UNIVERSITY OF TECHNOLOGY SYDNEY
Faculty of Engineering and Information Technology

# On the Use of Network Control Techniques in Pursuit of Influence Spread in Complex Networks

by

**Abida Sadaf**

A Thesis Submitted
in Fulfillment of the
Requirements for the Degree

**Doctor of Philosophy**

Sydney, Australia

2024

# Certificate of Original Authorship

I, Abida Sadaf, declare that this thesis, is submitted in fulfilment of the requirements for the award of Doctor of Philosophy, in the Faculty of Engineering and Information Technology at the University of Technology Sydney. This thesis is completely my own work unless otherwise referenced or acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis. This document has not been submitted for qualifications at any other academic institution. This research is supported by the Australian Government Research Training Program.

Signature: Abida Sadaf

Date: March, 2023

# Abstract

Influence and control of complex networks is a very challenging problem within network science. One perspective suggests we can only fully understand a network if we have the ability to influence or control it and predict the results of the employed control mechanisms. A critical element in the process of control and influence spread is the selection of the nodes from which influence and control spreads. In the context of influence spread, those nodes are called seed nodes and the best seed nodes are those that enable the quickest spread of influence. In the control space they are called driver nodes and they enable control of the whole network. The central intuition of this thesis is that the role driver nodes play in the context of network control is closely related to the role of seed nodes in spreading influence and that approaches for one task may be applicable to the other. Thus, the main aim of this thesis is to utilise the concepts from the field of network control and apply those to improve the spread of influence in a network by using seed selection methods based on driver nodes. To be able to meet project aim and to develop more effective seed selection methods, first we need to understand the relationship between different global and local network structures and the number of driver nodes needed to control a given structure. This reveals what structures are easier to control and the resources needed to control them. The first component of the thesis highlights how differing structure in both real and synthetic social networks affects the number of driver nodes needed for control. We investigate a correlation between global structural measures and the number of driver nodes. Experiments show that there is a strong relationship between density and the number of driver nodes. Next, the thesis investigates how the number of driver nodes identified at the level of individual communities relates to the densities of those communities. This illustrates how local structures and their composition influence the number of driver nodes. This second study, in concert with the first, reveals that the total number of driver nodes for a given network when detected in communities individually tends to be smaller than when detected in the network as a whole. The reason is that communities are close-knit, high-density groups within the overall network. The identification of an optimal set of seed nodes that maximise influence spread is an important research area and a number of techniques for identifying seed nodes that can enable an efficient spread of influence in the network have been already proposed. Current research, however, shows limitations of these techniques in terms of effectiveness and efficiency in achieving maximum influence spread in the networks. The idea of utilising driver node selection methods from control theory in the context of seed selection has not been yet explored to its full extent, prompting the central work in this thesis. In alignment with the structural examination of effective driver node selection in the initial part of the thesis, we first use driver nodes identified at the global (i.e., whole network) level and exploit these nodes as part of the seed selection methods. We find a minimum dominating set to develop an initial set of driver nodes. Using this base set, we propose new methods based (i.e. Driver-Random, Driver-Degree, Driver-Closeness, Driver-Betweenness, Driver-Degree-Closeness-Betweenness, Driver-Kempe, Driver-Ranked) for selecting seeds. These methods make use of network centrality measures to rank the driver nodes in terms of their potential as seed nodes. As a result we get a small subset of driver nodes, that helps in improving influence spread. We compare the proposed methods to existing approaches using the Linear Threshold model on both real and synthetic networks. The experimental results show that the proposed methods consistently outperform the existing benchmarks. We conclude that using driver nodes as seeds in the influence spread results in faster and thus more effective spread than when applying traditional methods. Following from the demonstration that one needs fewer driver nodes to control a network when they are detected community-by community, rather in the network as a whole, the final study uses ‘divide and conquer’ approach to the time-consuming problem of driver node identification at the global level and instead identifies driver nodes within the communities, then using those driver nodes in the influence spread process. In this thesis we demonstrate the effectiveness of this approach in Random, Small-World and Scale-Free networks as well as real-world social networks. The process begins with identification of communities within the network, and then identification of driver nodes for each community separately. The driver nodes obtained are then ranked according to a range of common centrality measures (using similar protocol as with the whole network approach). We then compare the total number of nodes influenced as a result of utilising various seed selection methods based upon globally elected and ranked driver nodes and locally selected and ranked driver nodes. This approach is not only novel in its basic concept, but provides improved algorithmic outcomes alongside more effective influence spread. To summarize, in this thesis we bring together two fields – influence spread and control in complex networks. We proposed and tested new family of seed selection methods that utilize the concept of driver nodes from control field.

## Acknowledgements

ALLAH has been the biggest source of strength for me and I am enormously thankful for the blessings It has bestowed upon me.

I am obliged to pay my heartiest gratitude to my supervisors, Dr. Katarzyna Musial-Gabrys, and Dr. Luke Mathieson for they are the very reason, I am writing these acknowledgments. Their priceless suggestions and guidance helped in polishing my work. I am indebted to them for their role in inculcating professionalism into my emotional being through their kind, peaceful, and understanding demeanour. Kaska and Luke! I want to THANK you for bringing up the researcher, out of me.

I would like to thank my candidature assessment panel, Dr. Jaime Garcia Marin, Dr. Nico Pietroni, and Dr. Paul Kennedy for their positive, and timely reviews.

Vigorous gratitude towards my friends, Chrissie, Saman, Mark, and Michelle, for their love, and constant support in all the difficult times I have gone through, my world is incomplete without them. Thank you to my lovely little humans (Archie, Liam, and Miles), who kept me alighted and entertained.

I acknowledge the presence of my morning window, it taught me to show up, no matter what, like the rising sun everyday. I yearn to broadcast a word of thanks to all the trails and trains, that I travelled on, for keeping me alive.

Exuberant appreciation for my family for always having faith in me and my dreams. Sohaib, Shaam, Shafiq Bhai, Mansur and Bakhtawar, thank you for guarding my sanity through all these years. They say, *"It isn't a miracle that you reached at the end but miracle is, you had the courage to start"*. Dear Ghufran, for being that persistent courage, I am thankful to you, beyond infinity.

I dedicate this dissertation to my parents and my son, to them, I owe everything.

# Contents

Certificate of Original Authorship . . . . . . . . . . . . . . . . . . . . . . . . . ii

Abstract . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . iii

Acknowledgements . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . vi

List of Tables xii

List of Figures xvi

**1 Introduction 1**

1.1 Aims, Questions, Objectives and Significance . . . . . . . . . . . . . . . 3

1.2 Methodology . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8

1.3 Thesis Contributions . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 11

1.4 Thesis Organisation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 12

**2 Literature Review 14**

2.1 Complex Networks Basics . . . . . . . . . . . . . . . . . . . . . . . . . . 14

2.2 Network Types . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 14

2.2.1 Network Structural Measures . . . . . . . . . . . . . . . . . . . . 16

2.2.2 Network Models . . . . . . . . . . . . . . . . . . . . . . . . . . . . 20

2.2.3 Tasks on Network . . . . . . . . . . . . . . . . . . . . . . . . . . . 22

2.3 Background of Control, Controllability and Influence of Complex Networks . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 23

2.3.1 Control in a Complex Network . . . . . . . . . . . . . . . . . . . 24

2.3.2 Controllability of Complex Networks . . . . . . . . . . . . . . . 28

2.3.2.1 Quantifying Network Controllability . . . . . . . . . 29

2.3.2.2 Controllability Criteria: Kalman Rank Condition . . 31

2.3.2.3 Structural Controllability . . . . . . . . . . . . . . . 32
2.3.3 Influence in Complex Networks . . . . . . . . . . . . . . . . . . . . . . 37
2.3.3.1 Independent Cascade Model (ICM) . . . . . . . . . . 38
2.3.3.2 Linear Threshold Model (LTM) . . . . . . . . . . . . 40
2.3.3.3 Variations of IC Model . . . . . . . . . . . . . . . . . . 42
2.3.3.4 Variations of LTM Model . . . . . . . . . . . . . . . 42
2.4 Methods for Selecting Driver Nodes . . . . . . . . . . . . . . . . . . . 43
2.4.1 Maximum Matching: Hopcroft-Karp Algorithm . . . . . . . . 44
2.4.2 Ford Fulkerson Algorithm . . . . . . . . . . . . . . . . . . . . . . 46
2.4.3 Hungarian Algorithm . . . . . . . . . . . . . . . . . . . . . . . . 47
2.4.4 Minimum Dominating Set . . . . . . . . . . . . . . . . . . . . . 47
2.4.5 Control Profiles . . . . . . . . . . . . . . . . . . . . . . . . . . . . 49
2.4.6 Preferential Matching . . . . . . . . . . . . . . . . . . . . . . . . 50
2.4.7 Finding Driver Nodes: Analysis . . . . . . . . . . . . . . . . . . 51
2.5 Methods for Selecting Seeds for Influence Spread . . . . . . . . . . . 53
2.5.1 Random Seed Selection (R) . . . . . . . . . . . . . . . . . . . . 53
2.5.2 Degree Seed Selection (D) . . . . . . . . . . . . . . . . . . . . . 53
2.5.3 Closeness Centrality Seed Selection (C) . . . . . . . . . . . . . 54
2.5.4 Betweenness Centrality Seed Selection (B) . . . . . . . . . . . 54
2.5.5 Eigenvector Centrality Seed Selection . . . . . . . . . . . . . . 55
2.5.6 Kempe Seed Selection (K) . . . . . . . . . . . . . . . . . . . . . 55
2.5.7 PageRank . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 56
2.5.8 LeaderRank . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 57
2.5.9 ClusterRank . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 57
2.5.10 K-Shell decomposition . . . . . . . . . . . . . . . . . . . . . . . 58
2.5.11 TwitterRank . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 58
2.5.12 ShaPley value-based Influential Nodes (SPIN) algorithm . . . 59
2.5.13 Optimal Influencers . . . . . . . . . . . . . . . . . . . . . . . . . 59

2.5.14 ARL . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 60

2.6 Conclusion . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 60

**3 Network Structure and Driver Nodes 63**

3.1 Background . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 65

3.2 Research Methodology . . . . . . . . . . . . . . . . . . . . . . . 66

3.2.1 Description of Networks . . . . . . . . . . . . . . . . . . . 68

3.2.2 Network Profiles Used . . . . . . . . . . . . . . . . . . . . 68

3.2.3 Conducted Experiments . . . . . . . . . . . . . . . . . . . 68

3.3 Result and Analysis . . . . . . . . . . . . . . . . . . . . . . . . . 69

3.3.1 Results from Generated Networks . . . . . . . . . . . . . . 69

3.3.2 Results from Social Networks . . . . . . . . . . . . . . . . 70

3.4 Discussion and Conclusion . . . . . . . . . . . . . . . . . . . . . 77

**4 Driver Nodes in Communities 82**

4.1 Background . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 85

4.1.1 Community Detection . . . . . . . . . . . . . . . . . . . . 85

4.1.2 Driver Nodes . . . . . . . . . . . . . . . . . . . . . . . . . 86

4.2 Research Methodology and Experiment Design . . . . . . . . . . . 88

4.3 Results and Analysis . . . . . . . . . . . . . . . . . . . . . . . . 94

4.3.1 Results from Synthetic Networks . . . . . . . . . . . . . . 94

4.3.1.1 Community Density and Number of Driver Nodes in Communities in Random, Small World and Scale Free Networks . . . . . . . . . . . . . . . . . . . . . . 95

4.3.1.2 Difference Between Number of Driver Nodes in Networks (NDN) and Number of Driver Nodes in Communities (NDNC) in Random, Small World and Scale Free Networks . . . . . . . . . . . . . . . . . . 95

4.3.2 Results from Social Networks . . . . . . . . . . . . . . . . 98

4.3.2.1 Community Density and Number of Driver Nodes in Communities in Social Networks . . . . . . . . . . . 98

4.3.2.2 Difference Between Number of Driver Nodes in Networks (NDN) and Number of Driver Nodes in Communities (NDNC) in Social Networks . . . . . . 98

4.4 Discussion and Conclusion . . . . . . . . . . . . . . . . . . . . . . . . 99

**5 Influence Models and Driver Nodes 104**

5.1 Background . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 104

5.2 Research Methodology and Experiment Design . . . . . . . . . . . . . 107

5.2.1 Networks . . . . . . . . . . . . . . . . . . . . . . . . . . . . 109

5.2.2 Experiments . . . . . . . . . . . . . . . . . . . . . . . . . . . 110

5.3 Results and Analysis . . . . . . . . . . . . . . . . . . . . . . . . . . . 112

5.3.1 Results from Synthetic Networks . . . . . . . . . . . . . . . . 113

5.3.1.1 Impact of network density on number of nodes influenced . . . . . . . . . . . . . . . . . . . . . . . 119

5.3.1.2 Percentage of nodes influenced . . . . . . . . . . . 120

5.3.1.3 Critical Difference Diagram for Generated Networks 123

5.3.2 Results from Real Social Networks . . . . . . . . . . . . . . . 124

5.3.2.1 Percentage of influence gain by DDCB method over other methods in the social networks . . . . . . . . . 124

5.3.2.2 Percentage of nodes influenced . . . . . . . . . . . 125

5.3.2.3 Number of iterations needed to influence the network 130

5.3.2.4 Critical Difference Diagram for Social Networks . . . 131

5.3.2.5 Time Complexity and Execution Times . . . . . . . 132

5.4 Discussion and Conclusion . . . . . . . . . . . . . . . . . . . . . . . . 135

**6 Influence Models, Communities and Driver Nodes 139**

6.1 Background . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 139

6.2 Research Methodology and Experimental Design . . . . . . . . . . . . 143

6.2.1 Influence spread using global driver nodes as seeds . . . . . . . 143

6.2.2 Influence spread using local driver nodes as seeds . . . . . . . 145

6.3 Results and Analysis . . . . . . . . . . . . . . . . . . . . . . . . . . . . 149

6.3.1 Results From Generated Networks . . . . . . . . . . . . . . . . 149

6.3.1.1 What is the speed and reach of the influence spread? 149

6.3.1.2 How much advantage do community—level driver based seed selection methods give? . . . . . . . . . . 151

6.3.2 Results From Social Networks . . . . . . . . . . . . . . . . . . 157

6.3.2.1 What is the speed and reach of the influence spread? 157

6.3.2.2 Time Complexity and Execution Time . . . . . . . . 160

6.3.2.3 How much advantage do community-level driver based seed selection methods give? . . . . . . . . . . 161

6.4 Discussion and Conclusion . . . . . . . . . . . . . . . . . . . . . . . . 164

**7 Conclusion and Future Work 165**

7.1 Conclusion . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 165

7.2 Future Work . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 169

# List of Tables

2.1 Definitions of Network Structural Measures. . . . . . . . . . . . . . . 16

2.2 Control Methods to Select Driver Nodes . . . . . . . . . . . . . . . . 51

3.1 Trend-line Equations for R, SW, SF networks; Where $X$ denotes number of driver nodes and $Y$ denotes the network structural measures 73

3.2 Online Social Networks Profiles . . . . . . . . . . . . . . . . . . . . . 74

3.3 Random, Small-World, Scale-free Network Profiles Representing BC(Betweenness Centrality), EV(Eigenvector Centrality), CC(Closeness Centrality), $N_dD$(Driver Nodes Density), ND(Network Density) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 75

3.4 P-Value in Random, Small-world, Scale-free, P-Value denotes the probability measure of likely difference between all the groups of networks. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 77

4.1 Global Network Structure Measures i.e., Number of Nodes, Number of Edges and Network Density $ND$ with their corresponding Number of Driver Nodes Density i.e. $N_dD$ in random, small-world and scale-free networks. . . . . . . . . . . . . . . . . . . . . . . . . . 91

4.2 Difference between Number of Driver Nodes (NDN) in the Whole Network and the Number of Driver Nodes within the Communities (NDNC) of the Networks, i.e., random (R), small-world (SW) and scale-free (SF) (Diff.). Nodes and Edges of the networks are also presented. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 102

4.3 Social Networks and their Global and Local Structure Measures such as Nodes (N), Edges (E), Density (D), Number of Driver Nodes (NDN), Number of Communities (C), Number of Driver Nodes in Communities (NDNC) and Difference between Number of Driver Nodes in Networks and Number of Driver Nodes in Communities (Diff.)103

5.1 The average percentage of influenced nodes in all generated networks, with seed sizes up to 50% after 20 iterations. N - number of nodes, Net. - Network model, D - network density, L - low and M - medium density. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 114

5.2 Percentage of influenced nodes when the density is low and seed size is 1%. The highest percentage is bolded. . . . . . . . . . . . . . . . . . 116

5.3 Percentage of influenced nodes when the density is medium and seed size is 1%. The highest percentage is bolded. . . . . . . . . . . . . . . 118

5.4 Average percentage points gain of DDCB method over R, D, C, B, DCB, K, DR, DD, DC, DB, and DK methods in Random (R), Small-World (SW) and Scale-Free (SF) networks. Seed set size is expressed as % of the total number of nodes. Each cell in the table shows the average and standard deviation of percentage gain of DDCB over other methods. . . . . . . . . . . . . . . . . . . . . . . . 121

5.5 Gain of DDCB over R, D, C, B, DCB, K, DR, DD, DC, DB and DK in real social networks. Seed set size is 25% of the all nodes. Each cell in the table shows the percentage gain of DDCB over other methods after 100 iterations. . . . . . . . . . . . . . . . . . . . . . . . . . . . 126

5.6 Time Complexity of Calculating Different Measures . . . . . . . . . . 133

5.7 Total Execution Times (in hours – Hrs.) of Influence Model using each of the Seed Selection Methods in the Examined Social Networks when 100% Influence is Reached. . . . . . . . . . . . . . . . . . . . . . 134

5.8 Total Execution Times (in hours – Hrs.) of All Seed Selection Methods in Each Social Network with 1% Seed Size when 100% Influence is Reached. . . . . . . . . . . . . . . . . . . . . . . . . . . . 135

6.1 A percentage gain table shows the percentage gain of DDCBC method over other seed selection methods in influencing the nodes in Random, Small-World and Scale-Free networks when the seed set size is 1% after 20 iterations. $N$ is number of nodes, $E$ is number of edges, $C$ is number of communities and $CD$ is average community density. . 153

6.2 A percentage gain table shows the percentage gain of DDCBC method over other seed selection methods in influencing the nodes of the social networks. Average Community Densities of the networks are as follows: FB (0.06±0.02), ZKC (0.32±0.4), Twitter (0.00029±0.05), Diggs (0.00008±0.007), Youtube (0.000012±0.04), Ego (0.00034±0.05), LC (0.007±0.032), LF (0.0073±0.09), PF (0.015±0.54), MFb (0.001±0.43), DHR (0.00085±0.21), DRO (0.0005±0.4), DHU (0.0004±0.63), MG (0.0011±0.03), L (0.0019±0.54), FbAR (0.0014±0.03), FbA (0.0015±0.09), FbG (0.0075±0.05), FbN (0.0013±0.003), FbP (0.0049±0.003), FbPF (0.004±0.032) and Fbt (0.0051±0.05) . . . . . . . . . . . . . . . . . . 155

6.3 Time Complexity of Calculating Different Measures and Methods . . 161

6.4 Execution Times (in hours – Hrs.) of All Seed Selection Methods in Each Social Network with 1% Seed Size When 100% Influence is Reached. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 162

6.5 Total Execution Times (in hours – Hrs.) of All Seed Selection Methods in all the Social Networks when 100% Influence is Reached . 163

# List of Figures

1.1 Driver Nodes in a Simple Directed Network . . . . . . . . . . . . . . 4

1.2 Challenges to Research Objectives to Research Questions to Experiments . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7

1.3 Research Methodology : A Macro View . . . . . . . . . . . . . . . . . 10

2.1 Commonly considered types of complex networks with respect to edge direction, edge weight and node label . . . . . . . . . . . . . . . 16

2.2 Centrifugal Governor. Picture taken from: [137] . . . . . . . . . . . . 26

2.3 Kalman Rank Condition; Adapted from [76] . . . . . . . . . . . . . . 31

2.4 Structural Controllability of Complex Network . . . . . . . . . . . . 33

2.5 Structural Controllability[126] . . . . . . . . . . . . . . . . . . . . . . . 35

2.6 Network with Input Vector: adapted from [178] . . . . . . . . . . . . 36

2.7 An example illustrating the cascade diffusion process in the Independent Cascade Model in a network of $n$ activated nodes with independent cascade $p_1 = p_2$ . . . . . . . . . . . . . . . . . . . . . . . 39

2.8 An example illustrating the information diffusion process of the Linear Threshold Model, where node $A$ and node $B$ are activated. . . 41

2.9 Maximum Matching Algorithm, Adapted from [89] . . . . . . . . . . 44

2.10 Minimum Dominating Set (MDS) Model [149] . . . . . . . . . . . . . 48

3.1 Research Methodology : Chapter 3 . . . . . . . . . . . . . . . . . . . . 64

3.2 Density Vs Number of Driver Nodes. X-axis shows the network density and y-axis shows the number of driver nodes. . . . . . . . . . 78

3.3 Network Density Vs Driver Nodes Density of Random, Small-world, Scale-free and Social Networks, x–axis shows Network Density and y–axis shows Driver Nodes Density . . . . . . . . . . . . . . . . . . 79

3.4 Social Networks Representing Driver Nodes Density, and Network Density . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 79

3.5 A degree distribution of social networks i.e. FB and LF . . . . . . . 80

3.6 Cosine Similarity Matrix of Generated and Social Networks . . . . . . 80

4.1 Research Methodology: Chapter 4 . . . . . . . . . . . . . . . . . . . 83

4.2 Example of Grivan-Newman Algorithm . . . . . . . . . . . . . . . . . 87

4.3 Representation of Zachary Karate Club Network and Detected Communities . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 87

4.4 Methodology . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 88

4.5 Number of Driver Nodes in the Communities of Random, Small World and Scale Free networks versus their Community Densities . . 90

4.6 Difference between Number of Driver Nodes in Overall Network Verses Number of Driver Nodes in Communities found in (a) Random Networks, (b) Small World Networks and (c) Scale Free Networks . . 93

4.7 Number of Communities in Random, Small World and Scale Free Networks . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 94

4.8 Number of Driver Nodes in the Communities of LF, Z and FB networks verses their Community Densities . . . . . . . . . . . . . . . 97

4.9 Difference between Total Number of Driver Nodes (NDN) and Number of Driver Nodes in Communities (NDNC) in Social Networks 97

5.1 Research Methodology : Chapter 5 . . . . . . . . . . . . . . . . . . 105

5.2 Control meets influence. The general concept for evaluating usefulness of driver nodes selection methods in seed selection for influence spread problem. . . . . . . . . . . . . . . . . . . . . . . . . 109

5.3 Size vs. density in Random, Small-World and Scale-Free networks . . 110

5.4 Critical difference diagram for generated networks. . . . . . . . . . . . 124

5.5 The percentage of influenced nodes in each iteration (the trend-lines for all simulation cases) for different seed selection methods and for Z, Youtube, Diggs, and PF networks. Seed set size is 25%. . . . . . . 125

5.6 The percentage of influenced nodes in each iteration (the trend-lines for all simulation cases) for different seed selection methods and for LF, FB, FbT, LC, FbP, FbG, L, FbPF, and FbA networks. Seed set size is 25%. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 127

5.7 The percentage of influenced nodes in each iteration (the trend-lines for all simulation cases) for different seed selection methods and for MFb, FbN, Ego, Twitter, FbAR, DHU, DRO, DHR, and MG networks. Seed set size is 25%. . . . . . . . . . . . . . . . . . . . . . . . 128

5.8 Number of iterations needed to reach 100% of influenced nodes using different seed selection strategies: (a) Traditional Seed Selection, (b) Driver Seed Selection, (c) DDCB Seed Selection. Seed set sizes are 5%, 15% and 25%. . . . . . . . . . . . . . . . . . . . . . . . . . 130

5.9 Critical difference diagram for real networks . . . . . . . . . . . . . . 132

6.1 Research Methodology : Chapter 6 . . . . . . . . . . . . . . . . . . . 140

6.2 An example showing the process for selecting seed nodes set from the driver nodes identified in network communities . . . . . . . . . . . . 146

6.3 Number of Nodes Influenced in Random, Small-World and Scale-Free Networks: when the number of nodes (N) is 100 and the number of edges (E) is 800 (Figures a, b and c); when N is 300 and E is 12800 (Figures d, e and f); when N is 500 and E is 72000 (Figures g, h and i). A Comparison of all methods for 20 iterations when the seed size is 1% is presented. . . . . . . . . . . . . . . . . . . . . . . . . . . . . 148

6.4 Average Number of Nodes in Communities of Random, Small-World and Scale-Free Networks versus number of communities in those networks. The legend shows the Number of Nodes in communities of generated networks i.e. Random (R), Small-World (SW) and Scale-Free (SF). . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 150

6.5 Percentage of Number of Nodes Influenced in FB, Z, LC, LF, PF, FbG, FbP, FbPF and FbT Networks. A Comparison of all methods for 100 iterations. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 158

6.6 Percentage of Number of Nodes Influenced in MFb, DHR, DRO, DHU, MG, L, FbAR and FbA Networks. A Comparison of all methods for 100 iterations. . . . . . . . . . . . . . . . . . . . . . . . . . 159

6.7 Percentage of Number of Nodes Influenced in Youtube, Twitter, Diggs and Ego Networks. A Comparison of all methods for 100 iterations. . 160

Table of Acronyms

| **Notation** | **Definition** |
|---|---|
| $N$ | Nodes in the network |
| $N_dN$ | Number of Driver Nodes |
| $N_dNC$ | Number of Driver Nodes in Communities |
| $V$ | Vertices in a graph |
| $E$ | Edges in a graph |
| $N_c$ | Total number of controls require to fully control the network |
| $N_s$ | A set of source nodes |
| $N_e$ | Internal dilation points |
| $N_i$ | Internal dilation points |
| $P(D_i)$ | Probability of node i to be part of a minimum set of driver nodes |
| $G$ | A network represented by a graph |
| $V(G)$ | Node set of a graph $G$ |
| $E(G)$ | Edge set of a graph $G$ |
| $S$ | Dominating Set of nodes in a graph |
| $M$ | Always representing a matching in a graph |
| MDS | Minimum Dominating Set |
| $ND$ | Network Density |
| CDensity | Community Density |
| SCF | Structural Controlability Framework |

# Chapter 1

# Introduction

In recent years, network science has emerged as a multidisciplinary domain that brings together, among others, economics, finance, physics, sociology, biology and transport, and is a focus of many researchers in these fields. Network science plays an important role in understanding everyday problems and providing solutions to those in these domains. Researchers are working on providing solutions to stop the spread of diseases by studying the behaviours of human beings and their social infrastructure [210, 129, 126, 205, 40, 96, 139, 165, 222, 74]. This research domain works by studying complex systems, their behaviours and effects if and when they do not work properly. It is important to understand the phenomenon of complex systems to understand the role of network science in our daily lives. There are many natural complex systems in our lives, our universe being the biggest one of them all. The examples of such natural complex systems include the human nervous system; infrastructure systems like power grids, transportation systems, communication systems; biological systems; the global climate and ecosystems vital to human life on this earth. Potential damage in such systems may result in the spread of disease, epidemics, economic collapse, and social unrest. Therefore, in recent times, current research urges us to understand, model, predict and ultimately control or influence these complex systems.

Complex systems can be represented and modelled as complex networks, where elements of the systems and interactions between them correspond to nodes and edges respectively [153, 194]. Complex networks are of great importance for understanding complex systems, e.g. the statistical mechanics of network topology and dynamics enable us to understand the functioning of real systems [153, 194]. One current understanding of control in a complex network tells us that we do not know if and how the network structure correlates with the number of driver nodes. As driver nodes play a key role in achieving control of a complex network, identifying them and studying their correlation with network structure measures can bring valuable insights, such as what network structures are easier to control, and how we can alter the structure in our favour to achieve the maximum control over the network [182]. This research work focuses on both global and local structural measures and their relationship with number of driver nodes. We propose that communities (as a measure for local structure) are one of the most important features of networks, and detecting them enables us to analyse and explore further underlying structural features of the synthetic as well as real networks [64]. The idea is to detect communities and driver nodes within the communities to see how the number of communities influences the number of driver nodes. We divide the experimental work in four major studies.

1. The first study starts by proving the hypothesis that some network structures are easier to control. This study uncovers the underlying relationship between number of driver nodes and different network structural measures. We prove that, different network structural measures are in correlation with number of driver nodes e.g. network density. Increasing network density implies that there will be decreasing number of driver nodes and vice versa.

2. In the next study, we dig deeper into network structures by identifying communities within those networks. We identify the local driver nodes within the communities to analyse the relationship between number of driver nodes and number of communities in synthetic and real networks. We conclude that the number of driver nodes tends to decrease within locally structured communities in the networks because of increasing densities of the communities of the networks. These results correspond to the conclusions of the first study.

3. In the third experimental study, we extend the experiment further to see how

effective and efficient the driver based seed selection methods are in influencing the overall network. We discover that such methods influence more number of nodes in the network faster in terms of percentage of number of nodes influenced in the network.

4. We further extend the previous study by applying the seed selection methods based upon the driver nodes found in communities. We conclude that these methods even outperform their counterpart methods from the third study. This study concludes our research journey in a way that, now we know that, locally identified driver based seed selection methods are able to reach influence faster than the globally identified driver based seed selection methods and to a higher percentage of nodes in a network.

## 1.1 Aims, Questions, Objectives and Significance

*The aim of this thesis is to utilise the concepts from the field of network control and apply those to improve the spread of influence in the network by using seed selection methods based upon driver nodes.*

Influence and control of complex networks is one of the most challenging open problems within network science. One view says that we can only claim to fully understand a network if we have the ability to influence or control it and predict the results of the employed control mechanisms. Investigating and understanding global network structures like network density, centrality measures, or shortest paths and local structures like communities is an important space in many domains and disciplines, including the spread of news on social networks. To be able to develop more efficient seed selection methods, we need to understand the relationship between different global and local network structures and the number of driver nodes needed to control a given structure. This will allow understanding of which networks might be easier to control and the resources needed to control them.

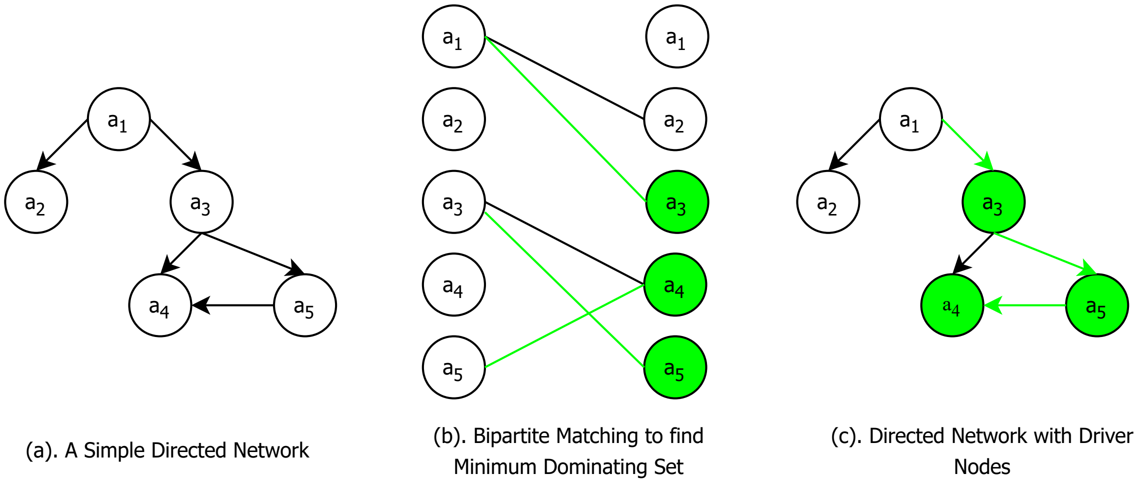

a1
a2
a3
a4
a5
(a). A Simple Directed Network
a1
a2
a3
a4
a5
a1
a2
a3
a4
a5
(b). Bipartite Matching to find Minimum Dominating Set
a1
a2
a3
a4
a5
(c). Directed Network with Driver Nodes

Figure 1.1 : Driver Nodes in a Simple Directed Network

We believe that control can be seen in networks, in many forms. One such example is influence spread in the network, where a set of nodes, commonly called as seed nodes can influence the other nodes in the network.

Finding a small subset of influential nodes to maximise influence spread in a complex network is an active area of research [40, 126, 40, 222, 73, 74]. We see that influence is an effective and softer form of control in complex networks. If we are able to influence a number of nodes in the network, we are controlling the network. Different methods have been proposed in the past to identify a set of seed nodes that can help achieve a faster spread of influence in the network. Understanding how influence is seeded and spreads through social networks is an increasingly important area of study. While there are many methods to identify seed nodes that are used to initialise a spread of influence, the idea of using methods for selecting driver nodes from control theory in the context of seed selection has not been yet explored. From Figure 1.1, we can see the detection of driver nodes in a directed network. The simple directed network in (a) can be converted into its bipartite network in (b), where green nodes are the matched nodes. In (c), we map the bipartite network back into the directed network, and the white nodes are the Minimum Dominating set or driver nodes. The notion that driver nodes play an important role in controlling all or part of the network, enables us to explore the correlations between local and global network structural measures and number of driver nodes. The correlations can reveal the network structures that are easier to control. We believe that Influence is a weaker form of control, so utilising the driver nodes to drive influence or spread influence through the network, to maximize the influence. From previous research, we see that, driver nodes are majorly used in different control methods to project control in the network [236, 27, 239, 91], so these could be a viable solution in maximizing influence in the network.

Therefore, we aim to propose seed selection methods based on driver node identification (locally and globally) from synthetic as well as real networks. Furthermore, the scope of the thesis includes utilising the seed selection methods combined with influence models to spread or maximise influence spread in the nodes of the network. Based upon the aim of the study, five main research challenges has been devised and presented in Figure 1.2. This diagram explains a relationship between research challenges, research questions, research objectives and the experiments that have been conducted to fulfil the objectives.

Research challenge 1 i.e., RC1, states that, "Understanding the research space to conduct a thorough research survey of Control and Influence in Complex Networks." The research initiates by understanding the research space to conduct a thorough research survey of the control and influence domain to find out potential gaps, which give birth to further research challenges to advance this research. Uptake of challenge 2 i.e., RC2 (i.e., Correlation of network structure measures with the number of driver nodes, to see the maximum control over a complex network.), includes a preliminary study based upon numerous synthetic network profiles to find out the relationship/correlation between network structure measures and number of driver nodes. Based on the finding of the promising results after conducting the experiment 1 (Exp1), which is Development and evaluation of a relationship between number of driver nodes and network structure measures in studying various network profiles for random, small-world, scale-free and real social networks. We find out that there exists a correlation between number of driver nodes and network structural measures. After that, we can put Research challenge 3 (RC3) in motion. While the premise of RC2 was global structure measures and their correlation with number of driver nodes, in RC3 we investigate in more detail the local network structures – communities. We detect communities in the networks with an aim of identifying the correlations between number of communities and number of driver nodes within those communities. Exp 2 concluded that if there are more number of communities in the network, we are likely to identify less driver nodes. In other words, number of driver nodes for the network is lower if we detect them within communities as opposed to when we detect them in the whole network. When it has been finally known that there indeed exists a relationship, we advanced our idea to use the identified driver nodes for the purpose of influence spread. The research advances to work on challenges 4 and 5 i.e., RC4 and RC5 respectively. RC4 and RC5 focus on exploring and proposing new seed selection methods, that can be utilized to spread influence efficiently and effectively throughout the networks. RC4 utilizes the driver based seed selection methods to compare the percentage of influence spread in various synthetic and social networks. RC5 extends this idea to include, locally identified driver nodes in communities to see the difference between the percentage of nodes influenced with globally identified driver-based methods verses locally identified driver-based seed selection methods. Following are the research questions derived from the challenges.

1. Research Question 1 (RQ1):How are the global network structural measures correlated with the number of driver nodes?

2. Research Question 2 (RQ2):How are the number of communities correlated with the number of driver nodes?

3. Research Question 3 (RQ3):How efficient and effective are driver based seed

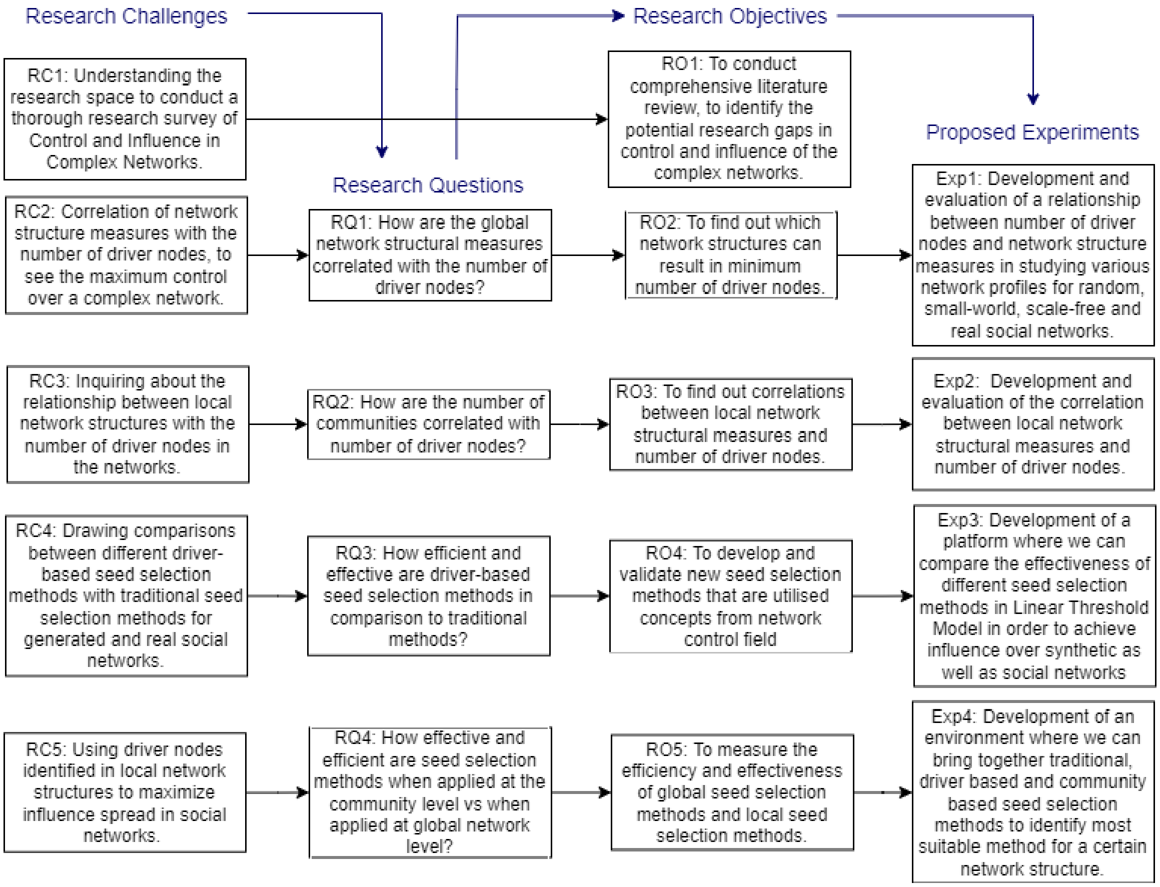

Research Challenges
Research Objectives
Research Questions
Proposed Experiments
RC1: Understanding the research space to conduct a thorough research survey of Control and Influence in Complex Networks.
RC2: Correlation of network structure measures with the number of driver nodes, to see the maximum control over a complex network.
RC3: Inquiring about the relationship between local network structures with the number of driver nodes in the networks.
RC4: Drawing comparisons between different driver-based seed selection methods with traditional seed selection methods for generated and real social networks.
RC5: Using driver nodes identified in local network structures to maximize influence spread in social networks.
RQ1: How are the global network structural measures correlated with the number of driver nodes?
RQ2: How are the number of communities correlated with number of driver nodes?
RQ3: How efficient and effective are driver-based seed selection methods in comparison to traditional methods?
RQ4: How effective and efficient are seed selection methods when applied at the community level vs when applied at global network level?
RO1: To conduct comprehensive literature review, to identify the potential research gaps in control and influence of the complex networks.
RO2: To find out which network structures can result in minimum number of driver nodes.
RO3: To find out correlations between local network structural measures and number of driver nodes.
RO4: To develop and validate new seed selection methods that are utilised concepts from network control field
RO5: To measure the efficiency and effectiveness of global seed selection methods and local seed selection methods.
Exp1: Development and evaluation of a relationship between number of driver nodes and network structure measures in studying various network profiles for random, small-world, scale-free and real social networks.
Exp2: Development and evaluation of the correlation between local network structural measures and number of driver nodes.
Exp3: Development of a platform where we can compare the effectiveness of different seed selection methods in Linear Threshold Model in order to achieve influence over synthetic as well as social networks
Exp4: Development of an environment where we can bring together traditional, driver based and community based seed selection methods to identify most suitable method for a certain network structure.

Figure 1.2 : Challenges to Research Objectives to Research Questions to Experiments

selection methods in comparison to traditional methods?

4. Research Question 4 (RQ4):How effective and efficient are seed selection methods when applied at the community level vs when applied at global network level?

Figure 1.2 couples each research question with corresponding research objectives. These research objectives provide the building blocks for achieving the aim of this thesis. The research objectives are given below:-

- Research Objective 1 (RO1): To conduct comprehensive literature review, to identify the potential research gaps in control and influence of the complex networks.
- Research Objective 2 (RO2): To find out which network structures can result in minimum number of driver nodes.
- Research Objective 3 (RO3): To find out correlations between local network structural measures and number of driver nodes.
- Research Objective 4 (RO4): To develop and validate new seed selection methods that are utilised concepts from network control field.
- Research Objective 5 (RO5): To measure the efficiency and effectiveness of global seed selection methods and local seed selection methods.

Below, in Section 1.2, the explanation of how the objectives and questions are linked with the thesis chapters is given.

## 1.2 Methodology

To achieve the objectives defined above, research methodology presented in Figure 1.3 is proposed and used. Figure 1.3 provides a macro view of all the research elements and how these are linked to all the objectives and questions to achieve the overall project's aim.

Firstly, a comprehensive literature survey is conducted to study control and influence in complex networks. The main focus of this part, is to fulfil RO1 as well as to find out the potential seed selection methods and ranking mechanisms for driver nodes in order to best utilise these for the further research. Various comparisons are included in the survey to understand the advantages and disadvantages of certain methods and approaches. This fulfils our first objective RO1, which was, "To conduct comprehensive literature review, to identify the potential research gaps in control and influence of the complex networks". The significance of this study is such that, it allows us to understand the control and influence space in the context of complex networks. This literature survey also enables us to find potential gaps in these domains, which are the vital part of the research challenges defined for this research work.

Based upon the thorough survey of methods and approaches in control and influence space the research gaps where identified and based on them the research challenges were formulated. Next we conduct preliminary study about correlation between number of driver nodes and global structural measures of different networks. This study will allow us to understand any relation between network structural measures and number of driver nodes, the analysis will guide us towards uncovering the network structures that are easier to control. We use randomly generated networks of various kinds (random, scale–free, small–world networks) and then conduct the same experiments on real social networks. This study concludes our second objective RO2.

We proceed, by further investigating into the local structures of the networks. This is done by identifying communities in the already generated networks and social networks. Within those communities, driver nodes are again identified to see if there exists a correlation between the number of driver nodes and number of communities in the networks. This helps in achieving RO3.

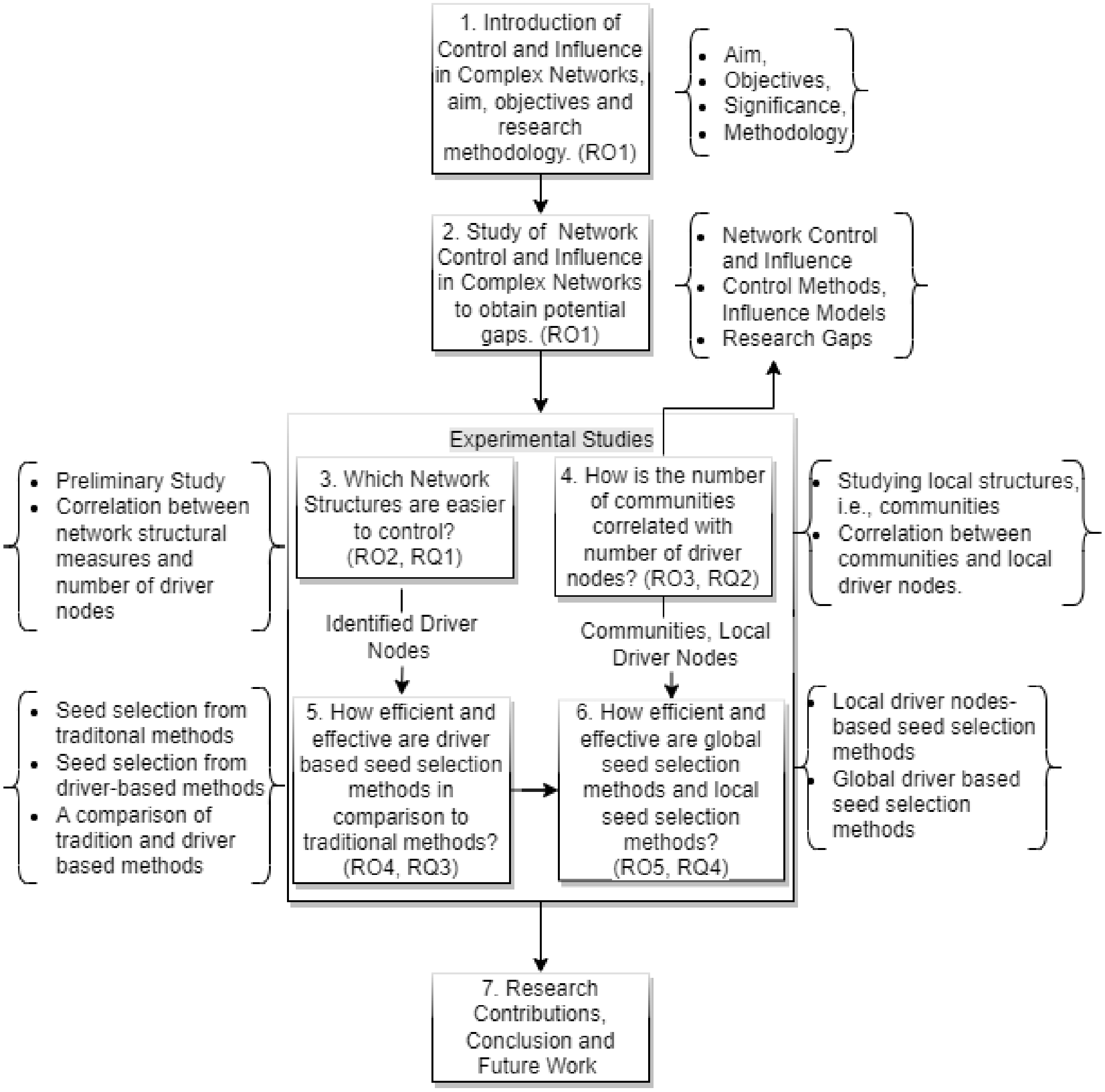

1. Introduction of Control and Influence in Complex Networks, aim, objectives and research methodology. (RO1)
Aim,
Objectives,
Significance,
Methodology
2. Study of Network Control and Influence in Complex Networks to obtain potential gaps. (RO1)
Network Control and Influence
Control Methods, Influence Models
Research Gaps
Experimental Studies
Preliminary Study
Correlation between network structural measures and number of driver nodes
3. Which Network Structures are easier to control? (RO2, RQ1)
4. How is the number of communities correlated with number of driver nodes? (RO3, RQ2)
Studying local structures, i.e., communities
Correlation between communities and local driver nodes.
Identified Driver Nodes
Communities, Local Driver Nodes
Seed selection from traditonal methods
Seed selection from driver-based methods
A comparison of tradition and driver based methods
5. How efficient and effective are driver based seed selection methods in comparison to traditional methods? (RO4, RQ3)
6. How efficient and effective are global seed selection methods and local seed selection methods? (RO5, RQ4)
Local driver nodes-based seed selection methods
Global driver based seed selection methods
7. Research Contributions, Conclusion and Future Work

Figure 1.3 : Research Methodology : A Macro View

For RO4 and RO5, we begin by proposing new methods for seed selection that utilise concept of driver nodes. Two sets of methods are developed: (i) global level driver nodes-based methods and (ii) community level driver nodes-based methods. The main aim of the experiments conducted in modules 5 and 6 (to address RC4 and RC5) is to find out the effectiveness and efficiency of proposed in this project driver based seed selection methods, both when driver nodes are identified locally at the communities level and globally at the network level. So, we use the previous outcomes from Exp1 and Exp2, and further introduce new experiments. These experiments, tests the proposed seed selection methods with both locally and globally identified driver nodes, hence achieve the research objectives 4 and 5.

The in depth methodologies for experimental studies conducted for Objectives 2, 3, 4 and 5 are given in Chapters 3, 4, 5 and 6 respectively.

## 1.3 Thesis Contributions

In this section, main contributions of this research work are highlighted as follows.

- **Potential Gaps in Control and Influence of Complex Networks**: A detailed literature survey of control methods in complex networks, and seed selection strategies used in propagating influence in the networks, is outlined. Survey laid out the current challenges, in the context of efficient strategies, to control or influence a complex network. For details, see Chapter 2.

- **Network Structural Measures and Driver Nodes**: An experimental study is conducted, to find out the correlations between network structural measures and number of driver nodes in an attempt to identify the network structures/profiles that are easier to control. Potential correlations were determined between global network structural measures (including number of nodes, number of edges, network density, closeness centrality, betweenness centrality, and eigenvector centrality) and number of driver nodes. We essentially find out the network structures that are easier to control or influence by a driver nodes set. For more details, see Chapter 3.

- **Driver Nodes in Communities**: Another key contribution from the study, "Driver Nodes in Communities", highlights the correlations between local network structural measures and number of driver nodes, see Chapter 4.
- **Influence Models and Control Methods**: Development and Validation of newly manufactured seed selection methods that helped bring together control and influence fields. The efficient and effective seed selection method(s) have been identified, see Chapter 5.
- **Influence Models, Communities and Driver Nodes** A successful comparison is provided for another set of newly developed seed selection approaches (based upon driver nodes in communities) with the approaches suggested previously (based upon driver nodes in the networks). For further details, see Chapter 6.

## 1.4 Thesis Organisation

This thesis is organised in the following chapters.

- Chapter *2* : This chapter presents a detailed literature review and relevant research work in the field of control and influence in complex networks. The literature revolves around the control, controllability and influence in complex networks in particular.
- Chapter *3* : This chapter describes an experimental study to find out the correlations between network structural measures and the number of driver nodes in a complex network. The study encompasses synthetic as well as real networks. This study has been published in ASONAM'21 [182].

- Chapter *4* : Chapter 4 presents an in depth study of when driver nodes are identified locally and globally. This research work incorporates identification of communities in the complex networks and their impact on the identification of number of driver nodes. The study presented in this chapter is in print as an extended research work in a book chapter by EB-ASONAM'21.

- Chapter *5* : Chapter 5 discusses the impact of different seed selection methods on networks, when using those in Linear Threshold Model to spread influence across the nodes of the network. The research work is submitted in the Journal of Applied Network Science for review [181].

- Chapter *6* : In this chapter, the final study of this extensive research work is presented. The study carefully describes the impact of driver nodes when identified locally and globally from different networks and then used as seed nodes after utilising different ranking mechanisms to spread influence across the various networks. This experimental study has been accepted and presented in SimBig'22 and in part at the Sunbelt'22 Conference [180].

- Chapter *7* : The final chapter includes a summary of the research work done, conclusions, research contributions, and potential future work.

# Chapter 2

# Literature Review

In this chapter, the literature review of control and influence in complex networks is provided. The chapter is subdivided into sections for control in complex networks and influences in complex networks, as the whole thesis revolves around those two fields. New seed selection methods proposed in this thesis are inspired by concepts coming from control in networks, which is why the review includes sections on methods for the selection driver nodes, ranking of driver nodes and seed selection methods.

## 2.1 Complex Networks Basics

A complex network is a representation of complex system. There are many systems of interest to scientists that are composed of individual parts or components linked together in some way. For example the Internet, World Wide Web [9], citation networks [63], or social networks [169] [145].

If we look into the connections between different components of any system, then this can be represented in the form of a network. The components of the system will be the network vertices (a.k.a. nodes) and the connections between them will be the edges (a.k.a. relationships) [153]. In this sense, a complex network can be seen as a representation of a complex system.

## 2.2 Network Types

Traditionally, with respect to the edges, complex networks can be modelled as directed/undirected as well as weighted/unweighted graphs. Nodes and edges can also be labelled. The label describes something about the relationship between the nodes [153]. It could name the relationship between people of the same family members for example father, mother, sister, cousin etc. Similarly, for a node, labels can be assigned based upon network topology and nodes characteristics [111], for example, the names of people in real life.

Undirected networks are those for which the relationship is reciprocated or when the direction is not significant from the perspective of analysis. On the other hand, in a directed network, relationships do not have to be reciprocated. Let's assume that we have a situation where one person sends an email to another, but the second one never replies. This can be modelled as an undirected relationship that is interpreted as "there exists any communication between two people" or as a directed connection where "a relationship exists from a person $x$ to $y$ if $x$ sends an email to $y$" [153].

From the modelling perspective, relationships can also have assigned weights that reflect some property of a connection. In social networks, weight is usually used to express the strength of a relationship between two users. This can be expressed e.g. as number of emails between two users. In order to understand the characteristics of the networks and networked systems, we need to model them mathematically [153] [154]. When it comes to nodes, we usually talk about labels (or attributes) on nodes. Thus, networks can be labelled or unlabelled with respect to nodes and edges. In labelled, we can e.g. consider user's gender or age as node's labels. In a city map, street numbers can be edge's labels [228].

Figure 2.1 represents a multitude of possible types of networks and organises the landscape of networks with respect to the three named above dimensions (edge direction, edge weight and node label).

We can visualize a complex network by its graphical representation like in the Figure 2.1.

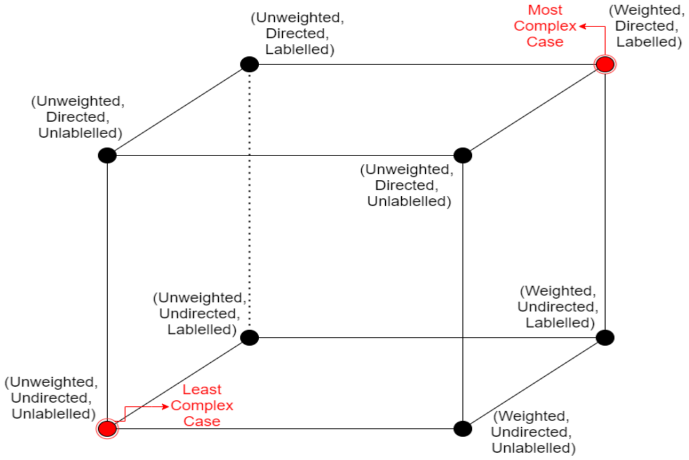

(Unweighted, Directed, Lablelled)
Most Complex Case
(Weighted, Directed, Labelled)
(Unweighted, Directed, Unlablelled)
(Unweighted, Directed, Unlablelled)
(Unweighted, Undirected, Lablelled)
(Weighted, Undirected, Lablelled)
(Unweighted, Undirected, Unlablelled)
Least Complex Case
(Weighted, Undirected, Unlablelled)

Figure 2.1 : Commonly considered types of complex networks with respect to edge direction, edge weight and node label

### 2.2.1 Network Structural Measures

Over the years, various network measures have been developed to understand the structure of the network and its functioning. The most commonly used is node degree distribution. Other measures include number of nodes and edges, density, clustering (both local and global), average path length, and different centrality measures (degree, closeness, betweenness centralities among others). These measures are also known as topological properties of a network, as they help quantify and interpret the structure of the network.

A short description of the most commonly used of the network structural measures is presented in Table 2.1.

Table 2.1 : Definitions of Network Structural Measures.

| Structural Measures | Definitions |
|---|---|
| No. of Nodes | In a graph $G(V, E)$, $\sum_{i=1}^{n} v_i$ denotes the sum of total number of nodes $v_i$ in $G$ [80]. |

| | |
|---|---|
| No. of Edges | In a graph $G(V,E)$, $\sum_{i=1}^{n} e_i$ denotes the sum of total number of edges $e_i$ in $G$ [80]. |
| Node Degree | In a graph $G(V,E)$, $\sum_{v \in V} deg(v) = 2\|E\|$, where $V$ is the set of vertices and $E$ is the set of edges of the graph $G$ [5]. |
| Max Node Degree | $\triangle G$ is the degree of node $v$ with maximum number of edges in a network [77]. |
| Degree Distribution | $Pdeg(k)$ = fraction of nodes in the network with degree $k$. [5]. |
| Average Path Length | $l_G = \frac{1}{n \cdot (n-1) \sum_{i \neq j} d(v_i, v_j)}$ [80]. |
| Network Diameter | Diameter, $D$, of a network having $N$ nodes is defined as the longest path, $p$, of the shortest paths between any two nodes $D = \frac{1}{4} max(min_p p_{ij} length(p))$ [211]. |
| Network Density | For undirected networks, $\eta = 2\|E\|\|V\|(\|V\|-1)$. For directed networks, $\eta = \|E\|\|V\|(\|V\|-1)$. Where $\|V\|$ is the number of vertices, and $\|E\|$ is the number of edges in the network. [211]. |
| Node Degree Centrality | $C_D(G) = \sum_{v \in G} \frac{\|deg(v^*) - deg(v)\|}{\|H\|}$, where $v^*$ is the vertex with highest degree. Where, $H = (\|V\|-1)(\|V\|-2)$. [211]. |
| Betweenness Centrality | $g(v) = \sum_{s \neq v \neq t} \frac{\sigma_{st} v}{\sigma_{st}}$, where $\sigma_{st}$ is the total number of shortest paths from node $s$ to the node $t$ and $\sigma_{st} v$ is the number of those paths that pass through $v$ [211]. |

| | |
|---|---|
| Closeness Centrality | $CC(i) = \frac{N-1}{\sum_j d(i,j)}$, where $i \neq j$, $d_{i,j}$ is the length of the shortest path between nodes $i$ and $j$, and $N$ is the number of nodes in the network. [211]. |
| Eigenvector Centrality | $x(v) = \frac{1}{\lambda} \sum_{t \in M(v)} xt = \frac{1}{\lambda} \sum_{t \in G} a_{v,t} x_t$, where $a_{v,t}$ is the adjacency matrix, i.e., $a_{v,t} = 1$ if vertex $v$ is linked to vertex $t$, and $a_{v,t} = 0$, otherwise. $M(v)$ is a set of neighbours of $v$ and $\lambda$ is a constant. [211]. |
| KATZ Centrality | For each node $i$, KATZ Centrality $x_i = \alpha \sum_j A_{ij} x_j + \beta$, where $A$ is the adjacency matrix of Graph $G$ with eigenvalues $\lambda$. $\beta$ controls the initial centrality and $\alpha < \frac{1}{\lambda_{max}}$ [211]. |
| Page Rank | $PR(P_i) = \frac{(d)}{n} + (1-d) \times \sum_{l_{j,i} \in E} \frac{PR(P_j)}{Outdegree(P_j)}$, where $d(DampingFactor) = 0.1\ 0.5$, $P_i$ is the Page Rank of node $i$ and $n$ is total number of nodes. [195]. |
| Group of vertices (sub-networks) | A network $G_1 = (V_1, E_1)$ is called a sub-network of a network $G(V, E)$ if $V_1(G)$ is a subset of $V(G)$ and $E_1(G)$ is a subset of $E(G)$ such that each edge of $G_1$ has same end vertices as in $G$. [47]. |
| Clique | A clique, $C$, in an undirected graph $G = (V, E)$ is a subset of the vertices, $C \subseteq V$, such that every two distinct vertices are adjacent. [133]. |
| $k$-plex | Let $G(V, E)$ be a graph. Then a subset $S$ of $V$ is a $k$-plex in $G$ if $deg_{G[S]}(y)$ is at least $\|S\| - k$ for every $v$ in $S$, where $V$ is the set of vertices, $S$ is a subset of vertices and $k$ is the number of plexes. [14]. |

| | |
|---|---|
| $k$-core | A $k$-core of a graph $G$ is the maximal sub-graph $H \subseteq G$, such that $\delta(H) \geq k$, where $\sigma(H)$ is the minimum degree of sub-graph $H$ and $k$ is any integer number [188]. |
| Global Clustering Coefficient, | $C = \frac{(NumberofTriangles)\times 3}{NumberofConnectedTripletsofNodes}$, where $C$ is Global Clustering Coefficient, Triangle is a set of 3 nodes, Connected Triplet is a connected Triangle [133]. |
| Local Clustering Coefficient | $C(v_i) = \frac{NumberofPairsofNeighboursofv_ithatareConnected}{NumberofPairsofNeighboursofv_i}$, where $C(v_i)$ is the Local Clustering Coefficient of node $v_i$ [213]. |
| Redundancy Coefficient | $rc = \frac{\lvert u,w \subseteq N(v), \exists v\,` \neq v, (v\,`,u) \in Eand(v\,`,w) \in E \rvert}{\frac{\lvert N(v)\rvert(\lvert N(v)\rvert - 1)}{2}}$, where $rc$ is the Redundancy Coefficient of the node $v$, $N(v)$ is the set of neighbours of $v$ in a graph $G$ [153] |
| Reciprocity | $r = \frac{1}{m}Tr(A^2)$, where $r$ is Reciprocity of the graph, and $Tr(A^2) = \sum_{i=1}^{n} A_{i,i}$ and $m$ is the number of edges in the network [155]. |
| Similarity (Structural Equivalence) | $n_{i,j} = \lvert N[i] \cap N[j] \rvert = \sum_k a_{i,k}a_{k,j} = a^2_{i,j}$, where $n_{i,j}$ is the count of common neighbours of the nodes $i$ and $j$ [60]. |
| Similarity (Regular Equivalence) | $\sigma_{i,j} = \alpha \sum_{kl} a_{ik}a_{jl}\sigma_{kl}$, where $\sigma_{i,j}$ is the product of the counts of the common neighbours and their neighbour's common similarity $\sigma_{kl}$ [79]. |
| Cosine similarity | $cos\alpha = \frac{n_{i,j}}{\sqrt{degi}\sqrt{degj}}$, where $n_{i,j}$ is the number of common neighbours of the nodes $i$ and $j$ [185]. |
| Pearson Correlation Coefficient, | $r_{i,j} = \frac{cov(a_i,a_j)}{var(a_i)^2}$, where $r_{i,j}$ is the Pearson Correlation Coefficient [18]. |

| | |
|---|---|
| Euclidean distance | $d(v_i, v_j) = \sqrt{(v_i - v_j)}^2$, where $v_i$ are the number of neighbours of vertex $i$ and $v_j$ are the number of neighbours of vertex $j$ [3]. |

From Table 2.1, we see all the network structural measures that are in use for the complex network analysis. In the context of this thesis, we use, many network structural measures, including, number of nodes, number of edges, network density, closeness centrality, degree centrality, betweenness centrality, eigenvector centrality, number of communities and community density. We use these measures later on, to rank driver nodes to form an optimal seed set, that can spread influence efficiently and effectively.

### 2.2.2 Network Models

One way to gain a deeper understanding about the structure, properties, and phenomena occurring in the real-world networks is to construct models of the underlying structure of a network and run controlled experiments over networks generated using those models. The models that can mimic the patterns existing in a real network help us understand the implications of these patterns [153].

Basic and well-known examples of complex network models are random models which include e.g. Erdős and Rényi model [45, 46], small world models [216], and scale free models [1]. Although those models are not very realistic, they enable to investigate some of the real-world phenomena, such as rich get richer (scale-free networks) or friend of a friend is my friend (small-world models). The network models descrbied below are used in this thesis because these are well-researched models in the network science community. Also, Erdős and Rényi model is used as the baseline.

- Erdős and Rényi model. It is a model in which some specific set of parameters is given fixed values but the generated network is random in all other respects.

One of the simplest examples of a random network is when the number of vertices $n$ and the number of edges $m$ are fixed. It means that we randomly place $m$ edges among $n$ vertices [46]. Alternatively, we can fix the number of nodes $n$ and probability $p$ that indicates the probability that edge will be created between any randomly chosen pair of nodes [62]. Random networks feature small average path, low clustering coefficient and Poisson node degree distribution for large $n$. There are two variations of the Erdős–Rényi random network. First is $G(n, M)$ model, where a graph is chosen uniformly at random from the collection of all graphs which have $n$ nodes and $M$ edges. Second is where $G(n, p)$ model is used, where a graph model is constructed by connecting labelled nodes randomly. After that each edge is included in the graph with a probability $p$, independently from every other edge.

- Small World model. In a small-world network majority of nodes are not neighbours of one another and the neighbours of any given node are likely to be neighbours of each other. It results in a structure in which majority of nodes can be reached from every other node by a small number of steps. A well-known example of small world model is the Watts-Strogatz model [216]. Mathematically a small-world network is defined as a network where clustering coefficient remains high when the average distance $L$ between two randomly chosen nodes grows proportionally to the logarithm of the total number of nodes $N$. Such that $L \propto logN$. The degree distribution is similar to this of random graph [216].

- Scale Free model. It is a network in which node degree distribution follows a power law distribution. It means that a network will include many nodes with small node degree and few nodes (a.k.a hubs) with very large number of connections. One of the very well known ways to generate scale free network is Barabási–Albert model [1]. Mathematically, a scale free network is the fraction $P(k)$ of nodes in the network that has $k$ connections to other nodes which goes to the larger values of k such that $P(k)$ $k^{-\gamma}$ [162].

There are many other network models that exist and the field is a very intensively researched and we refer interested readers to the surveys [52, 66] and [212].

### 2.2.3 Tasks on Network

There are several reasons for which complex networks are a useful representation of the underlying connected data. Various tasks on complex networks allow us to understand the structure and dynamics of complex networks in greater detail. Some of the most frequently explored tasks are explained below.

- Node Classification: it is related to the prediction of a class to which a node belongs to. For example in the telecom churn prediction we may be interested in assigning a customer to one of the classes: one class if a customer is predicted to churn and another if not [153, 165].

- Link Prediction: to identify if two nodes are likely to be linked together or not, for example friend recommendation on Facebook [*] [153, 165]. Dynamic link prediction is an extension of link prediction where various snapshots are used to train, validate and test [189].

- Community Detection: It is defined as the structural similarity between pair of nodes and pair of networks, for example to identify functional modules of the neurons [153, 165].

- Resilience: It is defined in terms of measuring the failure and recovery of networks [125, 153].

- Control: The ability to influence a complex network, such that to alter its output in a desirable outcome by driving the inputs [98].

[*] www.facebook.com

- Influence: Influence as a weaker form of control can be defined as the number of nodes that are activated by a specific number of seed set nodes using an influence reading model, such as Linear Threshold Model [103].

- Controllability: Controllability is the ability to control a given system to some extent. Furthermore, a system is called a controllable system, if by selecting suitable inputs of external signals we can drive the system from any initial state to any desired final state in a finite period of time. [98]

- Spreading processes: Dynamical processes over complex networks cover a range of applications from phase transitions and synchronisation in networks, through walking and searching on networks, to epidemics spread and collective behaviour enveloping social influence, rumour and information spread as well as opinion formation [154, 6, 153, 21].

This thesis is about bringing together, control, controllability and influence in complex networks. A background of control, controllability and influence of complex networks is presented in next sections.

## 2.3 Background of Control, Controllability and Influence of Complex Networks

Traditionally, control of complex systems has been done using control theory but now due to the emergence of complex networks, there are Structural Controllability, Exact Controllability, and Physical Controllability frameworks, and research in this area has recently attracted a lot of attention [124, 239, 153, 1, 194, 234, 196, 74, 222, 40, 96, 126, 165]. These frameworks suggests controllability conditions which needs to be satisfied for a complex network to be able to be controllable. Once a network is controllable it can be controlled by using control inputs. This has been explained in greater detail in the coming sections. The term "control" is frequently used in many disciplines and in various aspects. The kind of control we come across regarding the complex networks has its roots in the control theory which is a highly developed interdisciplinary branch of engineering and mathematics [8]. While the control is able to change the behaviour of the system in some respect, controllability is the ability of a system to be controllable to some extent. Some systems are partially while other can be fully controllable [124]. A key notion in control theory is the feedback process. The difference between the actual and desired output is applied as feedback to the system's input, forcing the system's output to converge to the desired output. Feedback control has deep roots in physics and engineering [137].

Previously, some models/methods have been proposed for the controllability and control in the complex networks related to specific domains, like interbank networks [40], protein interaction networks [222], and biological networks [74, 222, 40, 96, 126, 165, 139, 205, 210, 129]. One of the pioneer models/frameworks is based upon structural controllability [234, 126]. Other frameworks are based on exact controllability [231] and physical controllability [209]. One of the tasks mentioned in the previous section is network control and controllability. Control and controllability are interrelated, as the system needs to be controllable in order to be controlled. This section gives an introduction and background to the concepts of both the ability to control and influence the complex networks.

### 2.3.1 Control in a Complex Network

Control theory is a mathematically highly developed branch of engineering with applications in electronic circuits [120] and generally in the field of physics and electronics [137]. The term control is frequently used in various fields with diverse meanings, but here, as the starting point, the control is defined in its mathematical sense in the context of the control theory [82].

Control theory tells us how we can influence the behaviour of a dynamical system with the suitable inputs so that the system's output is able to follow a desired trajectory or reach final state in finite time. Thus, control is defined in terms of a state space representation (time-domain approach), where a control system is described by a set of inputs, outputs and state variables connected by a set of differential equations [98]. The state is defined as a mathematical entity that mediates between the inputs and the outputs of a dynamical system, while emphasising the notions of causality and internal structure[98]. Any state of a dynamical system can be represented as a vector in the state space whose axes are the state variables.

For example, the centrifugal governor presented by Maxwell as shown in Figure 2.2, one of the first practical control devices, has been used to regulate the pressure and distance between millstones in windmills since the 17th century and was used by James Watt to maintain the steady velocity of a steam engine. The feedback mechanism relies on a system of balls rotating around an axis, with a velocity proportional to the engine velocity. When the rotational velocity increases, the centrifugal force pushes the balls afar from the axis. This results in opening valves so that the vapour can get out. This lowers the pressure inside the boiler, slowing down the engine. James Maxwell in 1868 has provided the first definitive mathematical description of the centrifugal governor used in Watt's steam engine. This is one of the best known feedback control mechanisms in use today [137]. When the need to design controlled engineered systems has emerged, the mathematical control theoretical tools were developed, which are today widely applied in the design of electric circuits, manufacturing processes, communication systems, air-planes,spacecrafts and robots [124].

Complex networks are dynamical systems so to control them, we need to apply a set of inputs (control actions) to selected nodes and monitor how the network behaviour changes in response to these inputs. The feedback from that observation will allow us to check how far we are from the desired output and undertake corrective action if necessary. In general, a typical control over a complex network requires three steps.

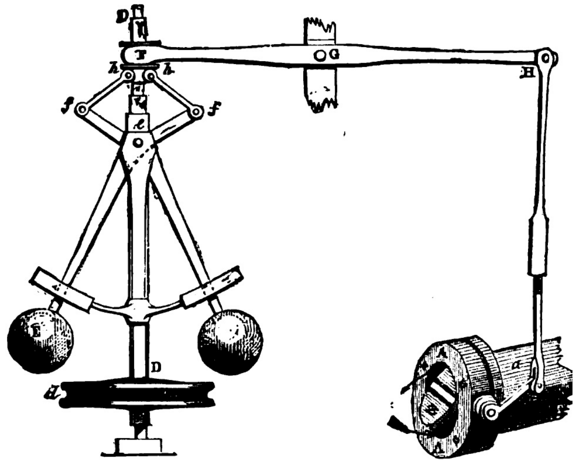


Figure 2.2 : Centrifugal Governor. Picture taken from: [137]

Step-1: Generate a model of a complex network (usually in a form of adjacency matrix that can be visualised as a graph) which is a representation of a given complex system.

Step-2: : Carry out a quantitative description of the dynamical laws that govern the temporal behaviour of each component.

Step-3: : Acquire an ability to influence the state and temporal behaviour of a selected subset of the components [124].

Some frameworks have been proposed to control complex networks related to specific domains, like interbank networks [40], protein interaction networks [222], and biological networks [74] [129]. However, it still becomes an open problem as to when and how the networks can be controlled in the real world scenarios. Especially challenging are social networks in which components are people and modelling their behaviour and its dynamics are inherently hard problems. Here are a few real life examples as how control can be applied in a real complex network.

1. Control from epidemic spread: Since the spread of coronavirus from the start of the year 2019 and up until now. Many countries have adapted a set of mech-

anisms to control its spread. This control is achieved after putting on a few restrictions and rules [161] [95]. So, we can say, that the restrictions and quarantine time are the control signals that can limit the spread of a pandemic. To control the spread of deadly viruses by imposing restrictions is not new. During 1918 Spanish flu spread with no vaccine to protect against influenza infection and no antibiotics to treat bacterial infections. To control its spread worldwide, some rules such as isolation, quarantine, good personal hygiene, use of disinfectants and limitations of public gatherings were enforced [143]. So, these control mechanisms were unwritten in some places as well as legally enforced in others, to control a complex system of virus spread. These mechanisms were found successful in those times and still relevant in recent pandemic incidents as well.

2. Influence people to change their behaviour: One of the important form of control is influencing the people in changing their behaviour. It is not the hard control but it has some proven effects. For example installations of speed cameras on highways are a major influence for a driver to slow down, where they will drive without concern for over-speed. Existing research consistently shows that speed cameras are an effective intervention in reducing road traffic collisions and related casualties [168].

3. Influencing people's habits by effective marketing: Driving the masses towards using a new product has always been a challenging task for organisations. One of the effective and efficient way to change people's opinions and habits in using a certain product is through marketing or advertising. The use of billboards on busy highways, TV commercials and push advertisements on social media websites such as YouTube[†] are very commonly used mechanisms in this regard. Nowadays, due to effective marketing and advertisement of the adverse effects of the use of plastic based products on the environment, many companies are

[†] www.youtube.com

attracting customers towards their organic products that produce zero to minimal carbon footprints. Changing people's habits from plastics to organically build materials could not have been possible without thorough marketing [28].

### 2.3.2 Controllability of Complex Networks

In this section, controllability of complex networks is described. We also learn that if a complex system is controllable at all. In order to find out control mechanisms for complex networks, we need to look into the structures of complex networks, if those can be controlled or not. The knowledge gained from the literature worked as a building block for further research on network structural measures and how different network structures are easier to control or not. Controllability is the ability to control a given system. For example, like a driver is assisting a car to move with the desired speed and in the desired direction by manipulating the pedals and the steering wheel [125].

Always, before applying control mechanisms, we analyse if it is at all possible to control a system. It means that we need to quantify the ability to steer a dynamical system to a desired final state in a finite time [137]. For example the act of balancing of a stick on our hand. We know from our experience that this is possible, suggesting that the system must be controllable [39].

Considering the controllability of complex systems there are two independent factors that contribute towards it. Both factors have a level of complication, which limits the advances in this field. One is the system's architecture, represented by the network encapsulating how the components interact with each other; and second one are the dynamical properties that depict the time-dependent interactions between the components. Hence, the controllability can be achieved only in the systems where both these perspectives are taken into account, for example, as it has been done in the case of control in biological networks [222]. Recent advances towards quantifying the topological characteristics of complex networks [194] [220] [154] have shed light on the role of system's architecture in its controllability. Dynamical change of links and nodes pose another challenge in understanding the control in complex networks [117] [88] [87] [113] [136] [202]. When the new components (nodes/edges) are added or deleted, they can completely change the control paradigm. So, there is a need to research in finding the efficient solutions to control the links/nodes dynamics [118].

#### 2.3.2.1 Quantifying Network Controllability

Network Controllability is quantified by using the traditional controllability conditions of complex systems. However, the complex networks we encounter in real life are not linear in nature. So, the controllability criteria defined for a complex linear system does not apply completely on a real life complex network. Initially, the controllability conditions defined for real life complex networks required them to be converted into a Linear Time Invariant (LTI) Systems first. So this section discusses the network controllability with respect to when a complex network is represented as an LTI system.

**Linear Systems** A linear system satisfies the property of linearity. Linearity is defined as differential equation of the relationship between input and output of the system. These differential equations should be utilising only linear operators. A system is linear if it satisfies the following two conditions.

- Additivity is represented by Equation 2.1
- Homogeneity is represented by Equation 2.2

$$\text{If } x_1(t) \ \rightarrow y_1(t) \text{ and } x_2(t) \ \rightarrow y_2(t), \text{ then } x_1(t) + x_2(t) \ \rightarrow y_1(t) + y_2(t) \tag{2.1}$$

$$\begin{gathered}\text{If } x_1(t) \ \rightarrow y_1(t)\text{ , then } a(x_1(t)) \ \rightarrow a(y_1(t)) \\ \text{Where } a \text{ is a constant}\end{gathered} \tag{2.2}$$

The additivity and homogeneity can be combined to form the principle of superposition, which implies as Equation 2.3. A system is linear if and only if it satisfies the principle of superposition [99].

$$(a_1(x_1(t))) + (a_2(x_2(t))) \ \rightarrow (a_1(y_1(t))) + (a_2(y_2(t))) \quad \text{Where } a_1 \text{ and } a_2 \text{ are constants} \tag{2.3}$$

**Linear Time Invariant Systems (LTI)** An LTI system satisfies two properties, one is linearity and other one is time invariance. Linearity has been defined before in Section 2.3.2.1. Time-invariant is defined as the property of a system where the output does not depend on a specific time $t$ when the input is applied [163]. For example, irrespective of the time when the input is applied the output will remain the same. That is, if the output due to input $x(t)$ is $y(t)$, then the output due to input $x(t-T)$ should be $y(t-T)$, where $t$ is the current time and $T$ is the time difference when the next input is applied. Such a system is time-invariant. Any system that can be modelled as a linear differential equation with constant coefficients is an LTI system. A very famous example of an LTI system is an electronic circuit constructed of capacitors, resistors and inductors [163].

A significant body of work in control theory focuses on linear systems, and linear time invariant (LTI) systems [97]. By looking into the dynamics of complex networks, we are able to define and quantify their controllability.

Let's look at the linear time-invariant control system (A,B):

$$\dot{x}(t) = fA(t)x(t) + B(t)u(t) \tag{2.4}$$

where the vector $x(t) = (x_1(t), ..., x_N(t))^T$ captures the state of a system of $N$ nodes at time $t$. For example, $x_i(t)$ can denote the amount of traffic that passes through a node $i$ in a communication network or transcription factor concentration in a gene regulatory network. The $(N \times N)$ matrix $A$ describes the interconnections between the nodes, for example the traffic on individual communication links or the strength of a regulatory interaction in genes. B is the $(N \times M)$ input matrix that identifies

$$A = \begin{bmatrix} a_{11} & 0 & 0 \\ a_{21} & a_{22} & 0 \\ a_{31} & 0 & a_{33} \end{bmatrix}; B = \begin{bmatrix} b_1 \\ 0 \\ 0 \end{bmatrix}; C = \begin{bmatrix} b_1 & b_1a_{11} & b_1a_{11}^2 \\ 0 & b_1a_{21} & b_1a_{21}(a_{11} + a_{22}) \\ 0 & b_1a_{31} & b_1a_{31}(a_{11} + a_{33}) \end{bmatrix}$$

$$N = 3, m = 1, RANK(C) = 3\ ; \mathit{Network\ is\ controllable}$$

Figure 2.3 : Kalman Rank Condition; Adapted from [76]

the nodes controlled by an outside controller. The system can be controlled using the time-dependent input vector $u(t) = (u_1(t)\ , ...\ u_M(t))^T$ imposed by the controller , where in general the same signal $u_i(t)$ can drive multiple nodes. If we wish to control a system, we first need to identify the set of nodes that, if driven by different signals, can offer full control over the network. These are called 'driver nodes'. The researchers have worked on finding the minimum set of driver nodes denoted by $N_D$, whose control is sufficient to fully control the system's dynamics.

The system described by Equation 2.4 is said to be controllable if and only if the matrix $C$ Equation 2.5 has full rank Equation 2.6.

$$C = [B, AB, A^2B, ..., A^{N-1}B] \tag{2.5}$$

$$rank(C) = N \tag{2.6}$$

This represents the mathematical condition for controllability, and is called Kalman's controllability rank condition [98]. Below is an example of Kalman controllability rank condition and how it is calculated.

#### 2.3.2.2 Controllability Criteria: Kalman Rank Condition

A condition for structural controllability is the Kalman rank condition as traditionally described [98] for linear time-invariant systems [97] represented by the Figure 2.3.

In order to calculate the Kalman Rank condition in the Figure 2.3. Matrix $A$ represents the adjacency matrix of the a complex network, while matrix $B$ is the input vector (containing the nodes randomly chosen as to which the control signals would be applied in order to control the network) that is the nodes controlled by an outside controller. $C$ is the controllability matrix whose form is given in Equation 2.5. We calculated the dot product $B \cdot A$, and then reduce it to the echelon form by performing row operations to calculate the rank of the final matrix $C$. Since the rank of the matrix $C$ is equal to the number of nodes $N$ in the complex network as shown in Figure 2.6, we infer that the network is controllable.

#### 2.3.2.3 Structural Controllability

The Structural Controllability proposes that, if there are no cycles in the complex network, a variable (or node) can control at most one of its neighbours in the structural interaction graph [126]. The effect or influence from an intervention on a node disseminates along the main directed path(s), where the number of necessary paths to cover the network dictates the minimum set of driver variables [126]. Cycles are considered to be self-regulatory and do not require an external control signal. Different algorithms have been proposed that can be used to identify the minimum driver nodes set. The most commonly used algorithm is maximum matching algorithm [53].

Traditionally structural controllability required a controllability condition to assess the controllability of a complex network, and that condition is known as Kalman Rank Condition [120], which has been described earlier in the Section 2.3.2.2.

Figure 2.4 shows how initial structural controllability paradigm which is equivalent to the Structural Controllability Framework (SCF) adapted for LTI systems. The advent of structural controllability framework for LTI systems is ground breaking as it eliminates the need to calculate the Kalman rank condition. The workflow within a general structural controllability framework for LTI systems is shown in Figure 2.4. The figure also shows the research landscape of the methods employed in different steps to define controllability and control over a complex network. To select a minimum number of driver nodes there are ranking mechanisms that can

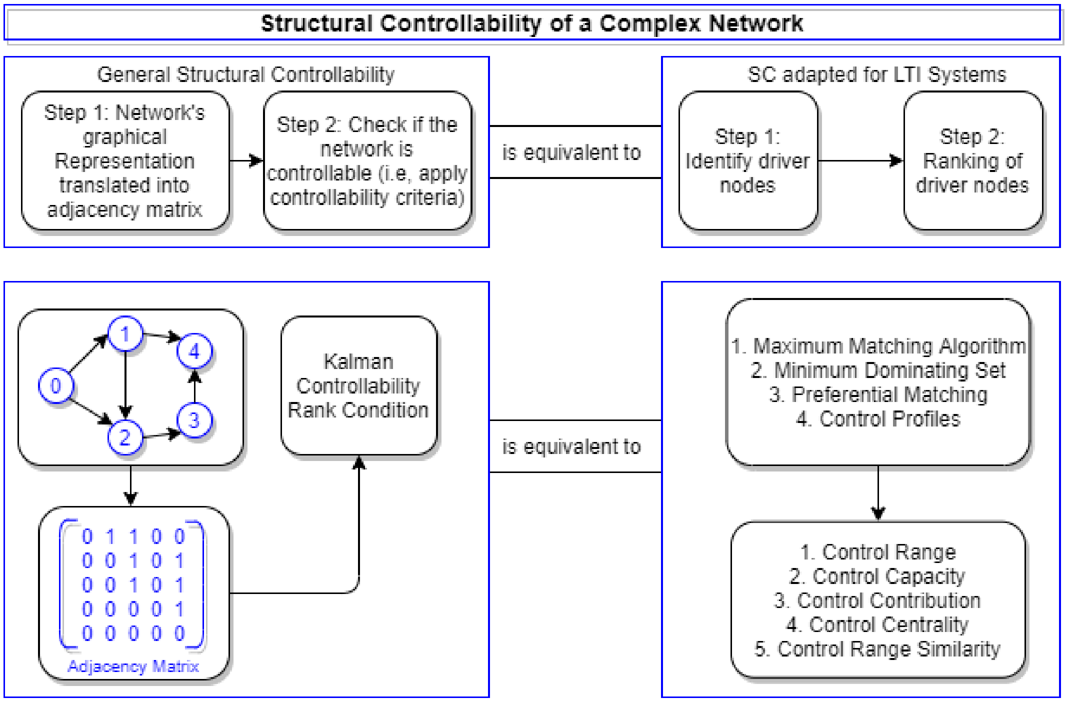

Structural Controllability of a Complex Network
General Structural Controllability
Step 1: Network's graphical Representation translated into adjacency matrix
Step 2: Check if the network is controllable (i.e, apply controllability criteria)
is equivalent to
SC adapted for LTI Systems
Step 1: Identify driver nodes
Step 2: Ranking of driver nodes
0
1
2
3
4
Kalman Controllability Rank Condition
0 1 1 0 0
0 0 1 0 1
0 0 1 0 1
0 0 0 0 1
0 0 0 0 0
Adjacency Matrix
is equivalent to
1. Maximum Matching Algorithm
2. Minimum Dominating Set
3. Preferential Matching
4. Control Profiles
1. Control Range
2. Control Capacity
3. Control Contribution
4. Control Centrality
5. Control Range Similarity

Figure 2.4 : Structural Controllability of Complex Network

rank the nodes that have more potential to become driver nodes given that they are connected to more nodes directly or indirectly. The details of the methods used in identification and ranking of the driver nodes are described in the further sections. Structural Controllability for LTI systems defined by [126], tells that in order to control the network we can find the minimum number of driver nodes, without actually calculating the Kalman rank condition. So the condition of finding the set of minimum number of driver nodes is actually equivalent to assessing a system's ability to be controlled previously done by using Kalman rank condition. It is hard to numerically verify Kalman's rank condition using fixed weights for a large network. The $rank(C)$ provides an idea of the controllable subspace of the system. The system is not completely controllable if $rank(C) < N$, it means that system can be decomposed into a controllable subsystem and an uncontrollable subsystem through a linear transformation [90]. This means, we can actually control a part of system which is called partial structural control. This is only a necessary test to be able to tell about structural controllability of a network but it is not a sufficient one because, practically it is really hard to convert a very large complex network into an adjacency matrix because we do not know all the weights of the edges in a network in case of a real world complex network which can grow infinitely, this poses a limitation when assessing the controllability of very large complex networks. Structural control, as described by [126] offers a framework to systematically avoid this limitation that was there in traditional structural control theory [120]. SCF leaves us with just finding a minimum number of input driver nodes that are needed to maintain full control of the network and is determined by the maximum matching in the network. Given a graph $G = (V, E)$, a matching $M$ in $G$ is a set of pairwise non-adjacent edges; that is, no two edges share a common vertex [53]. We gain full control of the directed network if and only if we directly control each unmatched node and there are directed path from the input signals to matched nodes [126].

In Figure 2.5 we present a graphical interpretation of the structural controllability framework as proposed by [126]. The unmatched nodes in Figure 2.5 are the driver nodes that are controlled by a control input signal. And these nodes in response can control each directed path to the matched nodes.

**Structural Controllability Observations** Here are some important deductions from the research work related to structural controllability framework.

1. A fundamental result from structural controllability states that the linear structured control system $(A, B)$ is structurally controllable if and only if the control-augmented graph $G(A, B)$ is spanned by cacti. A cacti is a connected graph where any two simple cycles in the graph have at most one node in common. Finding these cacti is equivalent to confirming the irreducibility condition on $[AB]$, since the cacti is a minimal structure such that removing any edge will render the system uncontrollable.

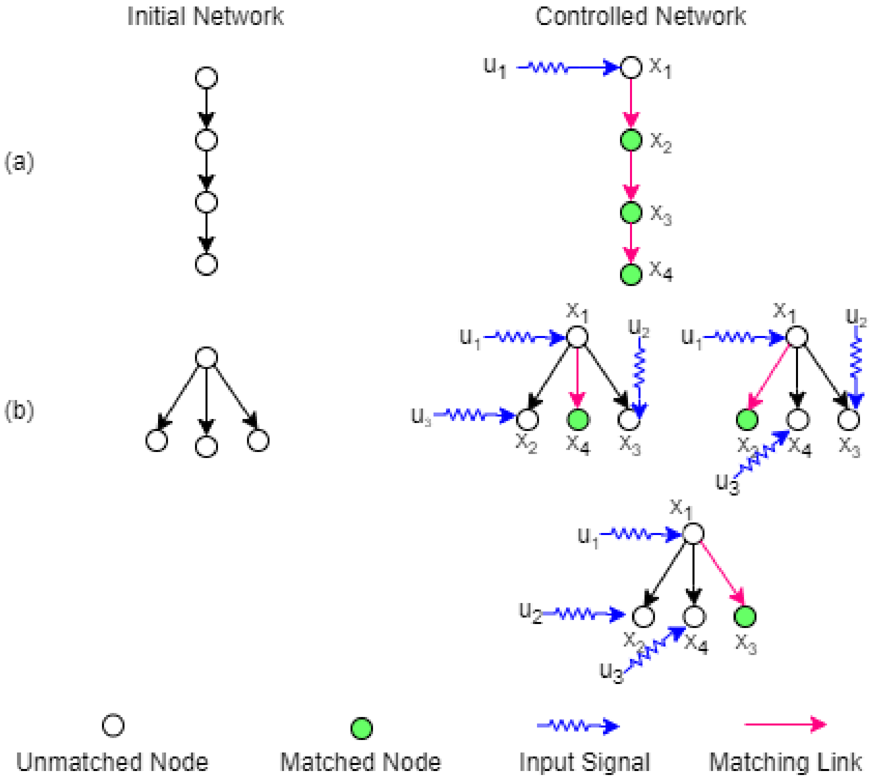


Figure 2.5 : Structural Controllability[126]

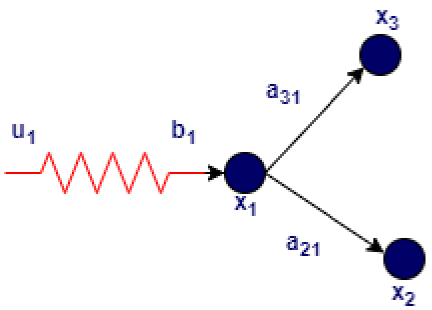


Figure 2.6 : Network with Input Vector: adapted from [178]

2. A node is inaccessible if there is no directed path reaching the node from any of the input nodes. Inaccessible nodes are nodes that are simply not reachable from the input nodes, hence it is not possible to exert a controlling influence over them.

3. A dilation exists in the graph $G(A, B)$ if a subset of nodes in $G(A)$, called $S$, can be found such that the number of nodes in the inbound neighbourhood set of $S$, given by $|T(S)|$ is smaller than the number of nodes in $S$, given by $|S|$. The inbound neighbourhood set of $S$ is the set of nodes with directed edges into $S$. Dilations imply an expansion in the network whereby there is not a sufficient number of independent inputs to control all nodes in $S$. There are at most $|T(S)|$ independent controls leading into $S$ and $|T(S)| < |S|$. The existence of cacti that span the graph $G(A, B)$ relies upon the control-augmented graph having no inaccessible nodes and no dilations.

4. Thus, the system $(A, B)$ is not fully structurally controllable if and only if it has inaccessible nodes or dilations [178].

In order to achieve control over a complex system we need to represent it in a form of a network which can be represented in a form of a graph to which we apply the procedure presented above. Consider the Figure 2.6 for an example network.

The small network represented in 2.6 can be controlled by an input vector $u$, allowing us to move it from its initial state to some desired final state in the state space.

### 2.3.3 Influence in Complex Networks

In this section, an overview of influence in complex networks and influence models is presented. Dynamical processes over complex networks can be used to create various events such as phase transitions and synchronization through walking and searching on networks, epidemics spread and collective behaviour that expand beyond social influence, rumour, information spread and opinion formation. Spread over a complex network, including its structural measures and dynamics have always been a potential area of research [21]. Social networks play a vital role in spreading ideas, behaviours and information. For example, medical and agricultural innovations can spread across the whole world [172], and information about new gadgets can spread via word of mouth or viral marketing [103]. Studies have observed different human emotional responses to real-life situations spreading across various social networks, such as happiness [54] and hate [171]. These processes are known as information diffusion and have traditionally been studied in the social sciences by [69] and [78]. In recent times, on the basis of initial information diffusion processes, many researchers have explored their applications in social network marketing [103] and recommender systems [160]. Spreading models are widely used to simulate the propagation of information, influence, opinion, content, and virus over a complex network to see how many nodes can be affected, and how fast they can be affected when different approaches are used [103]. Many models exist that implements the diffusion process to guide the public health measures by using epidemic models [104], for opinion formation by implementing voter models [35], and information diffusion by Independent Cascade [67], and Linear Threshold Model (LTM) [69, 78]. The two vital parts of spread analysis over a complex networks are the model of spread (diffusion model) and structure over which the propagation will take place. A large number of models exists and have been widely studies for that reason. Some examples are susceptible-infected (SI) susceptible-infected-susceptible (SIS), susceptible-infected-recovered (SIR) or threshold based models [69, 67, 166]. Diffusion models have been studied for many decades in epidemiology [104] and opinion dynamics [85]. The two most popular models for information diffusion in social networks are the Linear Threshold Model (LTM) [69, 78] and Independent Cascade (IC) Model [67]. These models have been used in diffusion prediction [191], influence maximisation [103], and estimating parameters for Independent Cascade models [184, 17]. For this research, we use LTM for influence spread in synthetic as well as real social networks. The same model is used across all the experiments to enable comparison of results across the whole study. Spreading process of LTM is similar to that of the infectious disease spread such as coronavirus. There is also a close proximity of information spread on social networks and infectious disease spread. For example active/inactive nodes in LTM model can be regarded as infectious/susceptible persons in an infectious disease spread process [227]. Regardless of the spreading model, at the beginning we need to select at least one node as a seed node which starts the spreading process. We can do it at random, like in the case of epidemic models, or we can use some heuristics. Some of the most commonly used methods, where top ranked influential nodes are selected, are Degree Centrality, Betweenness Centrality, Closeness Centrality, PageRank, LeaderRank, ClusterRank, K-Shell, Hill-climbing, HITS, ARL and Social Position [4, 146, 49] (see Section 2.5).

Below we discuss these two models in detail.

#### 2.3.3.1 Independent Cascade Model (ICM)

The main idea behind ICM is a common phenomenon defined in the field of behavioural economics and network theory, which occurs when a number of people make the same decision in a sequential order [67]. An information cascade model works in two steps: (i) first step is that an individual must encounter a scenario with a decision (yes or no) only then a cascade can begin; (ii) second step includes outside

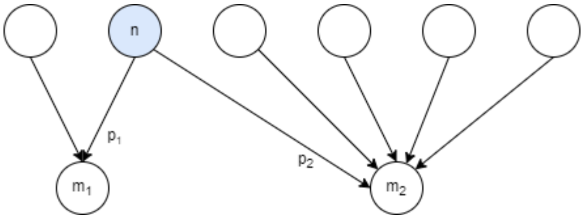


Figure 2.7 : An example illustrating the cascade diffusion process in the Independent Cascade Model in a network of $n$ activated nodes with independent cascade $p_1 = p_2$

factors, that can influence this decision [44]. In ICM, an active node $u$ attempts to influence all of its inactive neighbours but the success of the node $u$ in activating its inactive neighbour $v$ depends on the activation probability (a.k.a. propagation probability) of the edge from $u$ to $v$ (each edge can have its own value and the value $u \rightarrow v$ can be different from $v \rightarrow u$). Regardless of its success, the same node will never get another chance to activate the same inactive neighbour. The process ends when no further node gets activated.

**Quantifying Independent Cascade Model** In the Independent Cascade, each recently activated node $n$ will advance in activating each currently inactive neighbour $m$ with a fixed probability $p$, which is a global property of the system. $p$ is equal across all edges $n \rightarrow m$; when node $n$ has more than one neighbours, the attempts at activation are sequenced in a random order.

As an example, for the simple network in Figure 2.7, in which $A_0 = \{n\}$, the model has $p_1 = p_2 = p$ and, at time $t = 1$, nodes $m_1$ and $m_2$ are equally probable to become active. Given $p = 0.5$, the expected size of the set of active nodes at the end of the cascade propagation process is 2; this count includes the seed node $n$ itself. Another cascade model, Weighted Cascade, is a variation of ICM, such that it assigns non-uniform probabilities of activation to the edges: an edge $n \rightarrow m$ has probability:

$$\frac{1}{in - degree(m)}$$

of activating $m$ when $n$ is itself active. It simply means that, unlike for the Independent Cascade model, the expected number of neighbours which will succeed in activating any node equals 1 through weighted cascade [24].

#### 2.3.3.2 Linear Threshold Model (LTM)

In LTM the idea is that a node becomes active if a sufficient part of its neighbourhood is active. Each node $u$ has a threshold $t \in [0, 1]$. The threshold represents the fraction of neighbours of $u$ that must be active in order for $u$ to become active (e.g., how many of our friends have to switch to iPhone to push us to switching as well). At the beginning of the process a small percentage of nodes (seeds) is set as active in order to start the process. In the next steps a node becomes active if the fraction of its active neighbours is greater than its threshold [38] and the whole process stops when no node is activated in the current step. According to LTM, nodes can only become activated as activated neighbours increase. In practice, node thresholds are implemented by considering random or uniform thresholds [197] even though the propensity to be influenced can vary from individual to individual [201]. The linear threshold model (LTM) postulates that the thresholds are constrained by a linear relation to each other and therefore are completely defined by the first threshold $t_0$ and the linear increase $\delta$ as the sequence progresses [103]:

$$t_i + 1 = t_i + \delta . i. \tag{2.7}$$

**Quantifying Linear Threshold Model** Let $G = (V; E)$ denote an attributed social network, where $V$ is the set of nodes and $E$ is the set of edges between nodes. If two nodes $v, u \in V$ are connected by an edge, then $(v, u) \in E$, denotes an edge from $v$ to $u$. Define the set of neighbours of v to be $N(v) = \{u : u \in V; (u; v) \in E\}$. Each node $v \in V$ has an observed $m - \mathit{dimensional}$ vector of attributes, $X_v$, unobserved characteristics, $U_v$, and an outcome of interest $Y_v \in \{0, 1\}$, which is a binary indicator of whether the node is activated (e.g., whether an individual has donated to animal

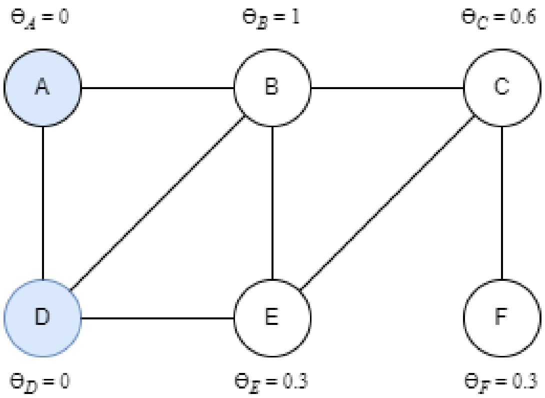


Figure 2.8 : An example illustrating the information diffusion process of the Linear Threshold Model, where node $A$ and node $B$ are activated.

charity). We define the set of activated nodes at time $t$ to be $D_t = \{v : Y_v = 1\}$. According to the Linear Threshold Model (LTM), each node $v$ has an activation threshold $\theta_v$. Given an initial set of activated nodes, $D_0 \subseteq V$, diffusion occurs in discrete steps, $t = 1, 2, ...T$. In each time step $t$, a node $v \in V \setminus D_i$ is activated if the activation influence, the weighted proportion of its activated neighbours, reaches the node's threshold $\theta_v$:

$$\sum_{u \in N(v)} w_u v Y_u^{t-1} \geq \theta_v \tag{2.8}$$

where $w_u v$ is the normalised influence weight of neighbour $u$ on $v$. According to LTM, nodes can only become activated as activated neighbours increase [197, 201]. For example, Figure 2.8 shows an illustration of a Linear Threshold Model, where nodes represent friend of friends circle in a social network. Initially, node $A$ and node $D$ are activated. Each node has their own threshold for example $\theta_A = 0.3$ means A's threshold is 0.3). The initial set of activations are the friends who have donated to an animal charity, which consists of two friends: $D_0 = \{A, D\}$. Assuming equal weights, $A$ will donate to an animal charity in the first time stamp since one of its three friends $D$ has donated to the said charity. No one else will be able to donate to the charity in subsequent steps, since $D$'s threshold is 1 and $C$'s threshold is 0.6.

#### 2.3.3.3 Variations of IC Model

Introduction of two cascades evolving simultaneously is another extension of basic IC model. In [25], they introduced a multi-campaign IC model. They further studied this idea of competing campaigns, in which the good campaign counteracts the effect of a bad campaign in a social network [25]. Different variations of IC Model involve time delay and time-critical constraints for influence diffusion. One such variation is described in [31], which proposes an extension to an IC model with meeting events, called IC-M model. IC-M model, works by assigning probabilities to the activated node so that these nodes can meet the inactive node. Compared to the basic IC model, the results from this model are more realistic and closer to actual situation, the only downside is the high execution time [31]. Another such research work adopted a novelty decay into the IC model [51]. Their findings suggest that the repeated exposures have reduced influence on users, hence development of a propagation path-based algorithm to assess the influence spread of seed nodes is a more feasible option. In this algorithm, there are two values on each edge of a social network, one is influence probability and other one is expected influence delay time [51]. In [140], researchers have thought about important time and trust factors. They managed to propose a trust-based latency-aware independent cascade (TLIC) model. In the TLIC model, a node can change its state (active or inactive) with different probabilities for a trusted neighbour node than for a distrusted neighbour [140].

#### 2.3.3.4 Variations of LTM Model

A variation to LTM is the use of Structural Causal Model (SCM) [167] to estimate individual thresholds in the LTM. In another research work competitive influence diffusion has been analyzed in different models that are based on general LTM [16]. In [141], they proposed a delayed LTM. In delayed LTM, nodes are supposed to have three states (i.e., active, inactive and latent active). In order to change the state from inactive to active, a node must be in a latent active state first. The results of the diffusion process of delayed LTM are better than traditional LTM, however the cost of accommodating a new node state is comparatively higher [141]. In [123], they considered the containment of competitive influence diffusion in social networks. The proposed extension to LT model, is about constructing the diffusion-containment (D-C) model, and traditional LT model is not applicable in case where both the diffusion and the containment of the influence are the main concerns. In this model, a node's state is defined as the activation probability which means, each node is only influenced by a neighbour with a higher probability, and the sum of the probabilities of possible node states is not greater than 1 [123].One of the most recent works has introduced methods for estimating heterogeneous treatment effects in networks [72]. Another extension to LTM is about calculating heterogeneous peer effect estimation and developing a structural causal model to identify and estimate peer effects. The two algorithms, developed for individual threshold estimation, are based on causal trees and causal meta-learners. The results on complex networks show that our proposed models can better predict individual-level thresholds in the Linear Threshold Model, these newly estimated thresholds help in predicting, which nodes will get activated over time [201].

## 2.4 Methods for Selecting Driver Nodes

Any network can be fully controllable if we control every single node but this is a very costly approach that in many cases is not feasible. Thus, the criteria of structural controllability of a complex system have been defined by determining the minimum number of driver nodes needed to control the whole system. To identify those nodes, the maximum matching algorithm was proposed and developed [89]. We can use different algorithms to find the maximum matching set of driver nodes in a bipartite graph, such as the Hopcroft-Karp algorithm [89], the Ford-Fulkerson algorithm [53],

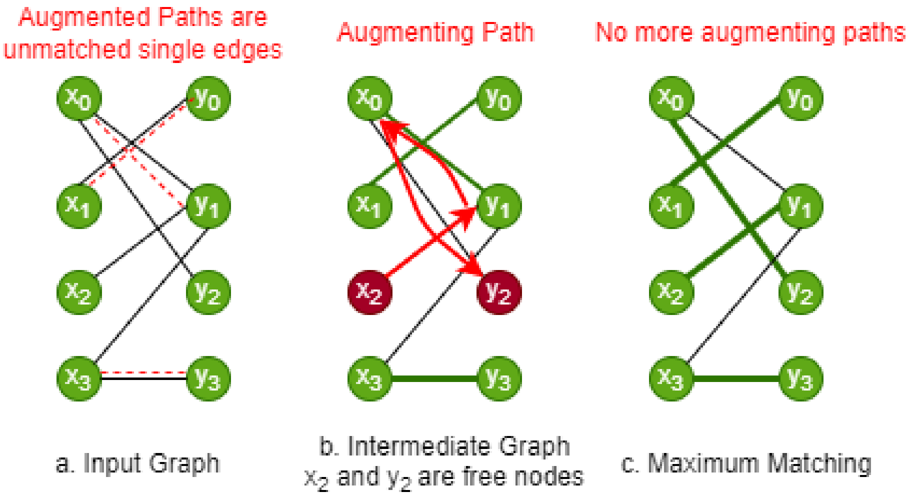


Figure 2.9 : Maximum Matching Algorithm, Adapted from [89]

and Hungarian algorithm [109]. Below is a description of how to find a set of driver nodes which are able to control the network. Although, if these nodes are physically able to control the network is yet to be seen [209].

### 2.4.1 Maximum Matching: Hopcroft-Karp Algorithm

Hopcroft-Karp algorithm [89] assumes a system is represented as a bipartite graph [81]. Maximum matching set is a maximum number of edges, no two of which meet at a common vertex [89]. A step by step process of finding a maximum matching set $M$ is given below:-

1. Initialize Maximal Matching $M$ as empty.

2. While there exists an Augmenting Path p

3. Remove matching edges of p from M and add not-matching edges of p to M

4. This increases size of M by 1 as p starts and ends with a free vertex

5. Return M.

The graph $G(V, E)$ is bipartite, if the set of vertices $V$ can be partitioned into two sets, $X$ and $Y$, such that each edge of $G$ joins a vertex in $X$ with a vertex in $Y$. An element of $X$ will be called a '$x_i$', and an element of $Y$, a '$y_i$'. Given a matching $M$, a node that is not part of matching is called free node. Initially all vertices as free nodes. In Figure 2.9, in second graph, $x_2$ and $y_2$ are free. In third graph, no vertex is free. Given a matching $M$, in the first step of the Maximum Matching algorithm, a maximal vertex-disjoint set of shortest augmenting paths relative to $M$ is found. All single edge paths that start and end with free vertices are augmenting paths. First we assign directions to the edges of $G$ in such a way that augmenting paths relative to $M$ become directed paths. This is done by directing each edge in $E - M$ so that it runs from a $v_i$ to a $u_i$, and each edge in $M$ so that it runs from a $x_i$ to a $y_i$ as we can see in the second graph of the Figure 2.9. Let $M$ be a matching in a bipartite graph $G$. In the next step, we extract a sub-graph from the initial graph, with the property that the directed paths of the sub-graph running from a free $y_i$ to a free $x_i$ correspond one-to-one to the shortest augmenting paths in $G$ relative to $M$. In the initial graph all single edges are augmenting paths, and we can pick in any order. In the middle stage, there is only one augmenting path. We remove matching edges of this path from the matching set $M$ and add non-matching edges. In final matching, there are no augmenting paths, so the matching is maximum [89, 240].

Unmatched nodes after applying maximum matching algorithm are called the 'driver nodes'. It is believed that these driver nodes can structurally control a complex network [126]. Maximum matching denotes the largest set of directed links without common nodes. That means, this set contains only node-disjoint directed paths, and directed cycles. In a maximum matching, a node is unmatched if no link in the maximum matching points at it [239]. The nodes of the original network that are connected directly to one of the input signals are called driven nodes, and the nodes of the input signals are called driver nodes. It is important to emphasise the distinction between driver and driven nodes because one driver node may drive more than one driven node [152].

### 2.4.2 Ford Fulkerson Algorithm

Ford Fulkerson algorithm [53] is described as follows: as long as there is a path from the source (start node) to the sink (end node), with available capacity on all edges in the path, we send flow along one of the paths. Then we find another path, and so on. A path with available capacity is called an augmenting path [37]. For example let $G(V, E)$ be a graph/network, for each edge from $U$ to $v$, let $c(u, v)$ be the capacity and $(f(u, v)$ be the flow. Maximal flow from the source $s$ to the sink $t$ can by find out by using Ford Fulkerson algoirthm. The input to the algorithm is the graph network $G(V, E)$ with flow capacity $c$, a source node $s$ and a sink node $t$. The algorithm computes a flow $f$ from $s$ to $t$ of maximum value. In the first step, flow $f(u, v) \rightarrow 0$ for all edges $(u, v)$. The second step repeats with the condition to find a path $p$ from $s$ to $t$ in $G_f$, such that $c_f(u, v) > 0$ for all edges $(u, v) \in p$. Where $G_f(V, E_f)$ is a residual network, it means its capacity is $c_f(u, v) = c(u, v) - f(u, v)$ with no flow. Under this condition, algorithm keeps finding the minimum capacity $c_f(p)$ and keep repeating the following two steps for each edge $(u, v) \in p$:

1. $f(u, v) \leftarrow f(u, v) + c_f(p)$ To send flow along the path.

2. $f(u, v) \leftarrow f(u, v) - c_f(p)$ Flow might be returned later.

In second step, if no more paths can be found, $s$ will not be able to reach $t$ in $G_f$. If $S$ is the set of nodes reachable by $s$ in $G_f$, then the total capacity in the original network of edges from $S$ to the remaining nodes in $V$ is equal to the total flow, found from $s$ to $t$. This represents an upper bound for all such flows. Which means that the flow, found is maximal [53, 41].

### 2.4.3 Hungarian Algorithm

The Hungarian method finds a perfect matching in a Bipartite graph and a potential such that the matching cost equals the potential value [23]. The Hungarian algorithm can also be executed by manipulating the weights of the bipartite graph in order to find a stable, maximum (or minimum) weight matching. This can be done by finding a feasible labelling of a graph that is perfectly matched, where a perfect matching is denoted as every vertex having exactly one edge of the matching. For example, a barpartite graph $G = (S, T; E)$ with $n$ worker vertices $(S)$, $n$ job vertices $(T)$ and the edges $(E)$. Each edge has cost/weight associated with it i.e, $c(i, j)$. Hungarian algorithm helps in determining the perfect matching with a minimum total cost. Mathematically, we can define it as follows:-

$$Afunctiony : (S \cup T) \rightarrow \mathbb{R}\, is\, a\, potential\, if\ y(i) + y(j) \leq c(i, j)\ for\, each\, i \in S.j \in T.$$

$$The\ value\ of\ potentially\ is\ the\ sum\ of\ the\ potential\ overall\ vertices\ \sum_{v \in S \cup T} y(v)$$

The cost of each perfect matching is at least the value of each potential: the total cost of the matching is the sum of the costs of all edges; the cost of each edge is at least the sum of the potentials of its endpoints; since the matching is perfect, each vertex is an endpoint of exactly one edge; hence the total cost is at least the total potential. The Hungarian method finds a perfect matching and a potential such that the matching cost equals the potential value [110].

### 2.4.4 Minimum Dominating Set

An equivalent to Structural Controllability is the optimisation procedure for undirected networks which can determine the minimum dominating set of nodes, which are required to control the network [147]. Minimum Dominating Set (MDS) starts from the assumption that each node can influence all of its neighbours simultaneously, but this signal cannot propagate any further. In it, the driver variables are identified by the minimal set such that every variable is separated by at most one

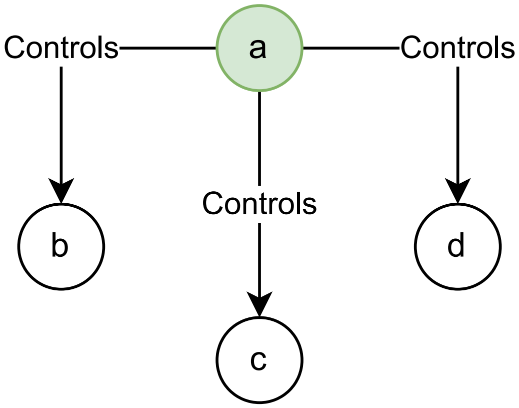


Figure 2.10 : Minimum Dominating Set (MDS) Model [149]

interaction [147, 148]. It has been used to identify control variables in protein interaction networks [222] and characterise how disease genes perturb the Human regulatory network [206]. Structural controllability assumes that only driver nodes can be controlled through external signals. MDS tells us that each driver node can control its associated edges independently. MDS further states that each non driver node is controllable if it is at least adjacent to a driver node. Also MDS based controllability is able to control the undirected network, and each node can control all of its outgoing edges separately. To understand MDS first we need to look at Dominating Set (DS) from graph theory. DS in a graph $G$ is a set of nodes $S$ (filled) in a graph $G$ is a dominating set if every node in $G$ is either an element of $S$ or adjacent to an element of $S$ [149]. The MDS approach states that a network is made structurally controllable by selecting an MDS (driver set) because each dominated node has its own control signal [147, 148], see Figure 2.10. A comparison of Maximum Matching (MM) and Minimum Dominating Set (MDS) reveals that, MM results in minimum number of driver nodes in Random Networks while MDS is more suitable to provide less number of driver nodes in scale-free networks where hubs are present [149].

In Figure 2.10, the network is structurally controllable by selecting a MDS because each dominated node has its own control signal. A maximum matching approach needs three driver nodes $v_1$, $v_2$, and $v_4$, assuming a matching link from $v_1$ to $v_3$. In contrast, the MDS only requires one node i.e, $v_1$. The labels , $(u_1)^1$, $(u_1)^2$ and $(u_1)^3$ indicate control. The MDS model has been widely applied to the control of complex networks, such as mobile ad-hoc networks (MANET), transportation routing, computer communication networks [193, 83, 2, 100]. MDS model has also been applied for the investigation of social influence propagation [102].

A greedy algorithm has been used to compare several types of artificial scale-free networks to look into the size of an MDS. It has been found out that a partial MDS set that dominates a fraction of nodes, exhibits the same scaling behaviour as MDS [142]. In another research work the applicability of the MDS approach proposed in [147] is demonstrated to the controllability of protein interaction networks. The results showed that the MDS of proteins were enriched with essential, cancer-related and virus-targeted genes. It means that, MDS protein set (of nodes) had a higher impact on network resilience than other hub proteins [222].

### 2.4.5 Control Profiles

Understanding the control properties of a complex system requires not just knowing how many controls are needed, but also characterising the functional origin of each control, because degree distribution correlation does not provide this information. For example, a finance system or a protein interaction system might require the same number of controls, but the structures within these networks can be very different. So, to practically implement a control method on a network depends on the interconnections of the control points as well [178]. Therefore, [178] describes, that we can examine the breakdown of the origin of controls in terms of amounts of source nodes, external dilation points, and internal dilation points for different classes of real-world networks, as well as for widely used generative network models [178]. They also discovered that knowing the full degree distribution is often unnecessary since, on average, the number of controls is dominated by only two points in the degree distribution: sources and sinks (i.e., nodes with in-degree = 0 or out-degree = 0, respectively) [178]. According to [178], control profiles of a complex network can be calculated by the minimum number of independent controls ($N_c$) required for full control of a complex network which is, the sum of the number of source nodes $N_s$, external dilation points $N_e$, and internal dilation points $N_i$, given by the equation below:

$$N_c = N_s + N_e + N_i \tag{2.9}$$

A dilation is formed where a path is branched into two or more paths in order to reach all nodes. Internal dilation is when a path is branched into two or more paths within a network. External dilation is when a path is branched towards sink nodes. The set of nodes can also be identified by maximum matching algorithm, as explained earlier in this section. Maximum matching algorithm provides us with accurate results but with expensive running time on large networks [178]. By counting source and sink nodes in linear time, we obtain a relatively good lower bound on the number of controls. In terms of time complexity, this approach is an improvement over the maximum matching algorithm [178].

### 2.4.6 Preferential Matching

In [238] authors proposed to use preferential matching to find driver nodes. For a directed network $G$, $V(G)$ is the node set and $E(G)$ is the edge set, where $N = |V|$ and $L = |E|$. A set of edges in $G$ is called a matching $M$ if no two edges in $M$ have a node in common. A node $v_i$ is matched by $M$ if there is an edge in $M$ pointing to $v_i$, otherwise $v_i$ is unmatched. A path $P$ is said to be $M - alternating$ if the edges of $P$ are alternately in and not in M. An $M - alternating$ path $P$ that starts and ends at the unmatched nodes is called an $M$ augmenting path. A matching with the maximum number of nodes is called a maximum matching $M^*$. A matching $M$ is called a perfect matching if all of the nodes of $G$ are matched by $M$. The minimum input theorem [190] says that if there is a perfect matching in a network, the number of driver nodes is one, otherwise the number of driver nodes is equal to the number of unmatched nodes with respect to any maximum matchings. The size of the maximum matching $M^*$ is denoted by $|M^*|$. According to [238], the minimum number of driver nodes is defined by the following Equation 2.10:

$$n_D = max(N - |M|^*, 1) \tag{2.10}$$

### 2.4.7 Finding Driver Nodes: Analysis

Table 2.2 shows a comparison of different methods which have been used to identify/select the set of driver nodes under structural controllability paradigm. The table defines the methods, their main concepts and whether they have been used with Linear Time Invariant (LTI) or Non-LTI complex systems.

Table 2.2 : Control Methods to Select Driver Nodes

| Methods | Definition and Main Concept | LTI/Non-LTI |
|---|---|---|
| Maximum Matching Algorithm | Maximum matching set is a maximum number of edges, no two of which meet at a common vertex. Set of unmatched nodes in a bipartite graph are called driver nodes. [89] | LTI Systems |
| Hopcroft-Karp Algorithm | Identifying edges with the property that no two edges share an endpoints. Identification of the number of driver nodes based on this algorithm is undefined. [89] | Not used before |

| Methods | Definition and Main Concept | LTI/Non-LTI |
|---|---|---|
| Ford Fulkerson Algorithm | The maximum flow from the source node to the sink node.Identification of the number of driver nodes based on this algorithm is undefined. [53] | Not used before |
| Hungarian Algorithm | Hungarian method finds a perfect matching of tight edges.Identification of the number of driver nodes based on this algorithm is undefined. [110] | Not used before |
| Control Profiles | It is defined as the minimum number of independent controls required for full control of a complex network. Control profile is the sum of the number of source nodes, external dilation points, and internal dilation points. [178] | LTI Systems |
| Preferential Matching | Identification of driver nodes by using preferential matching. Before identifying unmatched nodes, nodes sequence is sorted by nodes degree in ascending order. [238] | LTI Systems |
| Minimum Dominating Set (MDS) | Each node can influence all the neighbouring nodes simultaneously. Driver variables are identified by the minimal set such that every variable is separated by at most one interaction. [149] | LTI Systems |

## 2.5 Methods for Selecting Seeds for Influence Spread

In this section, various methods for seed selection for influence spread in a network are described. To initiate the influence process we need to select at least one node as a seed node which will start the spreading process. We can do it at random, like in case of some of the epidemic models [200], or we can use some heuristic to select the most optimal seed set which meets our needs, e.g. the total number of activated people will be the highest possible (advertisement campaign) or the total number of activated people will reach some threshold within some period (presidential campaign). Many different seed selection strategies have been developed to address different challenges, constraints and requirements. A brief description of most often used methods is included below.

### 2.5.1 Random Seed Selection (R)

In random seeds are selected Randomly from the node set of the network. Random seed selection is the baseline method to be used in comparing other seed selection methods.

### 2.5.2 Degree Seed Selection (D)

It starts by ranking the nodes according to degree centrality, and selecting a number of nodes with the highest values of degree measure [130]. Degree Centrality is defined as, the number of connections (degree) of a vertex in a network. An example can be, the influencers on social network sites, such as on Instagram[‡] who have many followers, these people can influence other people because of a large number of social contacts they contain. Mathematically, the node degree of node $i$ is defined as, $k_i = \sum_j^N e_{i,j}$, where $i$ is the focal node and $j$ represents the neighbours of $i$. The node degree is a fundamental indicator of a node's importance in the study of complex networks [217].

[‡] www.instagram.com

### 2.5.3 Closeness Centrality Seed Selection (C)

It dictates that a top percentage of number of nodes should be selected as seeds based upon their higher closeness centrality values [130]. Closeness Centrality is defined as the inverse sum of distances of a node to all other nodes in the network. Closeness Centrality of a node or an individual person in case of social networks, can be measured as, on average, how close or how far it lies from all other nodes in the network. The nodes with lower Closeness Centralities are desirable candidates, that are able to spread information because these nodes are closely connected to all other nodes in the network [7].

$$C(u) = \frac{n-1}{\sum_{v=1}^{n-1} d(v,u)} \tag{2.11}$$

Mathematically, it is defined in the Equation 2.11. Which describes that the Closeness Centrality of a node $u$ is the reciprocal of the average shortest path distance to $u$ overall $n-1$ reachable nodes. Where $d(v,u)$ is the shortest path distance between $v$ and $u$, and $n-1$ is the number of nodes reachable from $u$ [55].

### 2.5.4 Betweenness Centrality Seed Selection (B)

In this method a top percentage of number of nodes is selected as seeds based upon their higher betweenness centrality values [130]. It is calculated by analysing that, how often a node lies on the shortest path between any two pair of nodes in a network. High betweenness centrality for nodes that have a high difference from other nodes's betweenness values suggests that the network has pockets of densely connected nodes or communities. While, low betweenness centrality is an indication that the nodes of the entire network are well connected to each other which could imply the absence of well defined boundary structure for communities.

$$c_B(v) = \sum_{s,t \in V} \frac{\sigma(s,t|v)}{\sigma(s,t)} \tag{2.12}$$

Equation 2.12, shows the betweenness centrality of the node $v$. Where $V$ is the set of nodes $\sigma(s,t)$ is the number of $(s,t)$ shortest paths and $\sigma(s,t|v)$ is the number of those paths passing through some node $v$ other than $s,t$ [56].

Nodes with high betweenness centrality values are the ones that can form a connection between different communities. These same nodes are able to influence nodes from different communities increasing chances to influence more nodes in an overall complex network [56].

### 2.5.5 Eigenvector Centrality Seed Selection

In this method, a top percentage of number of nodes is selected as seeds based upon their higher eigenvector centrality values. Eigenvector centrality is a measure which defines the importance of a node in a network. It is calculated iteratively on nodes and where is assigned a relative score based on the idea that connections to high-scoring nodes contribute more to the score of the node under observation than equal connections to low-scoring vertices. A node is considered important if it is connected to other important nodes implying that a node with high eigen-vector centrality might not itself have high connections but relies on its neighbours to influence other nodes. [11].

### 2.5.6 Kempe Seed Selection (K)

It is a generalisation of hill-climbing algorithm where the seed set is constructed in the following way. For each node in the network, spreading process (or predefined number of steps of the process)is run and each node is evaluated on the basis of its potential to activate as many other nodes as possible. Then the best node is added to the seed set. For the following nodes the same process is followed, but then each node is evaluated on its potential in combination with all the nodes already in the seed set i.e. for the second node we have to check every combination of the first node in the seed set with all remaining nodes to find the best couple; for the third node we check every combination of the first two nodes in the seed set with all remaining nodes to find the best trio, etc. We continue adding nodes until we consume our seeding budget, i.e. reach the predefined size of the seed set. This approach on average produce the solution which is $(1 - \frac{1}{e})$ of maximum solution and outperforms centrality based methods like D, C or B. The disadvantage is that since we need to run the spreading process for $Nk$ times (where $k$ is the size of the seed set) it is very time-consuming, costly in terms of resources and hardly applicable for any real world solution [103]. The greedy algorithm by [103] is given below:

**Kempe Greedy Algorithm**: $N$ is a set of nodes, and $k$ is a positive integer such that $k \leq |N|$.

```
$A \leftarrow \phi$
for $i = 1$ to $k$ do
    Choose a node $n_i \in N \backslash A$ maximizing
    $\rho(A \cup n_i) - \rho A$
    Set $A \leftarrow A \cup n_i$
end for
```

### 2.5.7 PageRank

In this method a top percentage of number of nodes as seeds based upon their higher PageRank values [19, 164]. PageRank [164] have been widely known as a reputable way to obtain the authority score of a node based on network connectivity. For example, if $G = (V, E, W)$ be a directed network (general case), where $V = \{1, 2, ..., n\}$ is the node set and edge set $E$ represents all connections between nodes. $W = w_{i,j_{nn}}$ is the PageRank matrix, $w_{i,j}$ represents the strength of the endorsement from node $i$ to node $j$. The general PageRank values $x = \{x_1, x_2, ..., \ x_n\}$ of the nodes in a network can be represented as the following formula.

$$d = dWx + \frac{(1-d)}{n}e \tag{2.13}$$

Where $d \in (0, 1)$ is the damping factor, and $e = [1, 1, ..., 1]$ [122].

### 2.5.8 LeaderRank

In this method, a top percentage of number of nodes is selected as seeds based upon their higher LeaderRank values. For example, if we consider a network of $N$ nodes and $M$ directed links. Nodes correspond to users and links are established according to the relations between leaders and fans. To rank the users a ground node is introduced which may connect to every node in both directions. The resulting network will become densely connected and will consists of $N + 1$ nodes and $M + 2N$ links. Mathematically, LeaderRank method is equivalent to random walk on the directed network. It is described by a stochastic matrix P with elements $P_{i,j} = a_{i.j}/k_i^{out}$ which represents the probabilitiy that a random walker at $i$ goes to $j$ in the subsequent phase. $a_{i,j} = 1$ if node $i$ points to $j$ and 0 otherwise, while $k_i^{out}$ represents the out-degree which means number of leaders of $i$. This probability flow corresponds to the vote from fan $i$ to leader $j$. [131].

### 2.5.9 ClusterRank

In this method a top percentage of number of nodes is selected as seeds based upon their higher ClusterRank values [29]. In a directed social network, a link from $i$ to $j$ means is a follower of $i$, indicating that $j$ receives information from $i$. $\Gamma_i$ is the set of followers of $i$ and the density of interactions among $i$'s followers can be characterized by the local clustering coefficient of $i$. Based on the original definition of clustering coefficient [214], the clustering coefficient is given as in Equation 2.12:

$$c(i) = \frac{|e_{j,k} : j,k \in \Gamma_i|}{k_i^{out}(k_i^{out} - 1)} \tag{2.14}$$

where $k_i^{out}$ is the out-degree of $i$, that is, total number of followers of the node $i$. $|e_{j,k}|j,k \in \Gamma_i|$ is the links connecting two of $i$'s followers. Based upon clustering coefficient, ClusterRank $s_i$ is defined as:

$$s_i = f(c_i) \sum_{j \in \Gamma_i} (k_i^{out} + 1) \tag{2.15}$$

Where $f(c_i)$ is the effect $i$'s local clustering and $+1$ is in effect when $j$ is added to it [29].

### 2.5.10 K-Shell decomposition

This method starts by selecting a top percentage of number of nodes as seeds based upon their higher K-Shell values [106, 217, 128]. In K-Shell decomposition, nodes are assigned to K-shells according to their remaining degree, which is obtained by successively removing the smaller degree nodes with degrees less than the $k_s$ value of the current layer. The process initiates by pruning all nodes that have degree $k = 1$. When that is done, some nodes may be left with one link. So, the process of removing of the nodes keep going, iteratively until there is no node left with $k = 1$ in the network. The removed nodes, along with the corresponding links, are a part of a K-shell with index $k_s = 1$. Similarly, the next K-shell is iteratively removed for $k_s = 2$. The removal process continues to prune higher-k shells until all nodes are removed. Finally, each node is associated with one $k_s$ index, and the network are represented as the union of all K-shells [106].

### 2.5.11 TwitterRank

In TwitterRank, a top percentage of number of nodes is selected as seeds based upon their higher TwitterRank values [219]. TwitterRank is an extension of PageRank algorithm, designed to measure the importance of Twitter users taking into account similarity between users and the links between them. For example, consider a directed graph $G(V, E)$. It is formed with the twitterers and the "following" relationships among them. $V$ is the vertex set, which contains all the twitterers. $E$ is the edge set. An edge exists between two twitterers if there is "following" relationship between them, and the edge is directed from follower to friend. A random surfer model on graph $G$ computes the TwitterRank in the following steps [219]:-

1. A random surfer visits each twitterer with a probability by following the appro-

priate edge in $G$.

2. The random surfer performs a topic-specific random walk, i.e. the transition probability from one twitterer to another is topic-specific.

3. A topic specific relationship network is constructed among twitterers.

### 2.5.12 ShaPley value-based Influential Nodes (SPIN) algorithm

In ShaPley a top percentage of number of nodes is selected as seeds based upon their higher ShaPley value [151]. In [151], authors describe the process to determine SHaPley value by the use of the following example. A cooperative game with a transferable utility is defined as the pair $(N, v)$ where the set of players is defined as a set of, $N = \{1, 2, ...., n\}$. Real mapping of $v : 2^N \to \mathbb{R}$ is with $v(\phi) = 0$. $2^N$ is the set of all possible subsets of $N$. The mapping $v$ is called the characteristic function. Given that, any subset $S$ of $N$, $v(S)$ is called the value of coalition $S$ and represents the total transferable utility that can be achieved by the players in $S$, without any help from players in $N \backslash S$. The set of players $N$ is called the grand coalition and $v(N)$ is called the value of the grand coalition. ShaPley decide that a payoff $\phi_i(N, v)$ be allocated to a player $i$, which defines the relative importance of each player. The formula to calculate ShaPley value $\phi_i(N, v)$ of a player $i$ is given below:

$$\phi_i(N, v) = \sum_{C \subseteq N \backslash \{i\}} \frac{|C|!(n - |C| - 1)!}{n!} \{v(C \cup \{i\}) - v(C)\} \tag{2.16}$$

### 2.5.13 Optimal Influencers

In this method, optimal seeds are identified using optimal percolation, i.e. by evaluating the size of the giant connected component after the removal of the seed nodes [144]. For example, consider a network composed of $N$ nodes with $M$ links having an arbitrary degree distribution $P(k)$. If we remove a certain fraction $q$ of the total number of nodes. Percolation theory [20] describes that, if these nodes are chosen randomly,

a giant connected component from the network can become disconnected, i.e., $G = 0$. The optimal influence problem can be solved by determining the minimum fraction $q_c$ of influencers to divide the network: $q_c = min\{q \in [0,1] : G(q) = 0\}$
The optimal set $n^*$ of influencers $Nq_c$ can be obtained when the minimum of the largest eigenvalue reaches the critical threshold [144], given by the following equation:

$$\lambda(n^*; q_c) = 1 \tag{2.17}$$

### 2.5.14 ARL

In this approach, authors use Association Rule Learning (ARL). Thanks to the use of association rules and the simple assumption that people who often start a discussion, in which many other people then take part, are important for a given community, authors developed a new ARL method. It can find key people on "raw" data without the need to project users interaction towards objects (posts and comments) to the social network of interactions between users, which we need to use "traditional" methods to find key users such as node rank or PageRank. The evaluation showed that there is no statistically significant difference between the results achieved by ARL and PageRank, and by omitting the expensive network projection process, ARL is on average 36 times faster than the node degree and 70 times faster than PageRank (research was conducted on 108 different datasets coming from public Facebook pages) [49, 48].

## 2.6 Conclusion

Studies reveal that structure-only methods fail to properly characterise control, because there can be many different variations of possible dynamics that may occur in the networks [125, 239, 1, 194, 234, 196, 74, 222, 40, 96, 126, 165, 139, 205, 210, 129, 33, 61]. So, we not only need to consider/study the node behaviour, but also need to incorporate other factors, like role of links [96] and control profiles [178] in the controllability of the complex networks. There is a substantial amount of work that has been done regarding the structural controllability of complex networks. The following points emphasize the challenges that are part of this research area.

1. Structural controllability methods find the driver nodes to control the network. To find an optimal and energy efficient driver nodes still remains to be an area worth exploring further [224]. With structural controllability, researchers are also exploring MDS as the next model to control the complex networks. The general framework provided by [232] can be a direction to apply/experiment controllability with any multiplex network with an arbitrary architecture [232]. Network controllability helps us to identify the minimum set of driver nodes, MDS, needed to control the whole network. Practically, we might not have access to all of these driver nodes or only want to control the part of a network (sub-network). There is a need to work out an optimal solution to find a set of driver nodes that can be used to further propagate control/influence in the network.

2. Then, the complexity of choosing a smaller set of driver nodes arises. It means, given this number, the largest possible subset of the network can be controlled. If we have to restrict to this smaller set, we should have a ranking of driver nodes that allows us to pick those that have the largest impact on controlling the network. Existing measures for such a ranking, for example control capacity, and control range, are not best suited because they only focus on one aspect of driver nodes, either their probability to become a driver or the size of the sub-network they control. Control contribution combines both of these two aspects [239].

3. In the literature, a categorisation of techniques according to the type/kind of network they can control is still missing. To further elaborate this point, there is a need to look into the network structural measures and their relationship with different control measures. For example, a question that, “Which network structural measures are in correlation with the control measures such as driver nodes?” is still needed to

be explored.

4. From the literature survey, we find out that, an intersection of control methods and influence models needs to be explored further. We know that there are various seed selection methods i.e., traditional seed selection are already in use, when spreading the influence in overall network. But, a large amount of work is needed to find out an optimal seed set. We believe that by employing new ways, specifically driver nodes identification methods to identify driver nodes, and then rank those driver nodes by using seed selection methods and other criteria can be beneficial in maximizing the influence spread process in the overall network.

5. Many studies focus on how to quantify the influence of nodes in a complex network [71, 132] with the hope that if the most influential nodes are chosen to propagate a given phenomenon, then the spread of this phenomenon will be optimal.

# Chapter 3

# Network Structure and Driver Nodes

This chapter focuses on investigating how the network structure relates to the number of driver nodes and in turn the ability and effort needed to control a given network. The ultimate goal is, that we are able to determine which topologies are easier to control and which require more effort. This will help assess whether a given network can be easily or not controlled by simply looking at its characteristics and without the expensive process of determining the set of driver nodes. The initial challenge extracted from the literature survey begins by understanding the correlations between network structural measures and number of driver nodes. From Figure 1.2, Research Challenge RC2 states that, "Correlation of network structural measures and number of driver nodes, to see the maximum control over a complex network". We further explore and identify the Research Question RQ1 that will potentially help us in solving RC2. From Figure 1.2, RQ1 states that, "How are the global network structural measures related to number of driver nodes?". Answering RQ1 serves as the basis for the further research questions and experimental studies conducted, hence is an integral part of the thesis. We further devise an objective (RO2) to fulfill this question which states that, "To find out which network structures can result in minimum number of driver nodes". Figure 3.1 highlights the tasks that were accomplished in order to resolve RQ1 and RO2 so that RC2 can be accomplished in the context of whole thesis.

Our current understanding of control in a complex network is lacking the knowledge about how the network structure is related to the number of driver nodes. As driver nodes play a key role in achieving control of a complex network, identify-

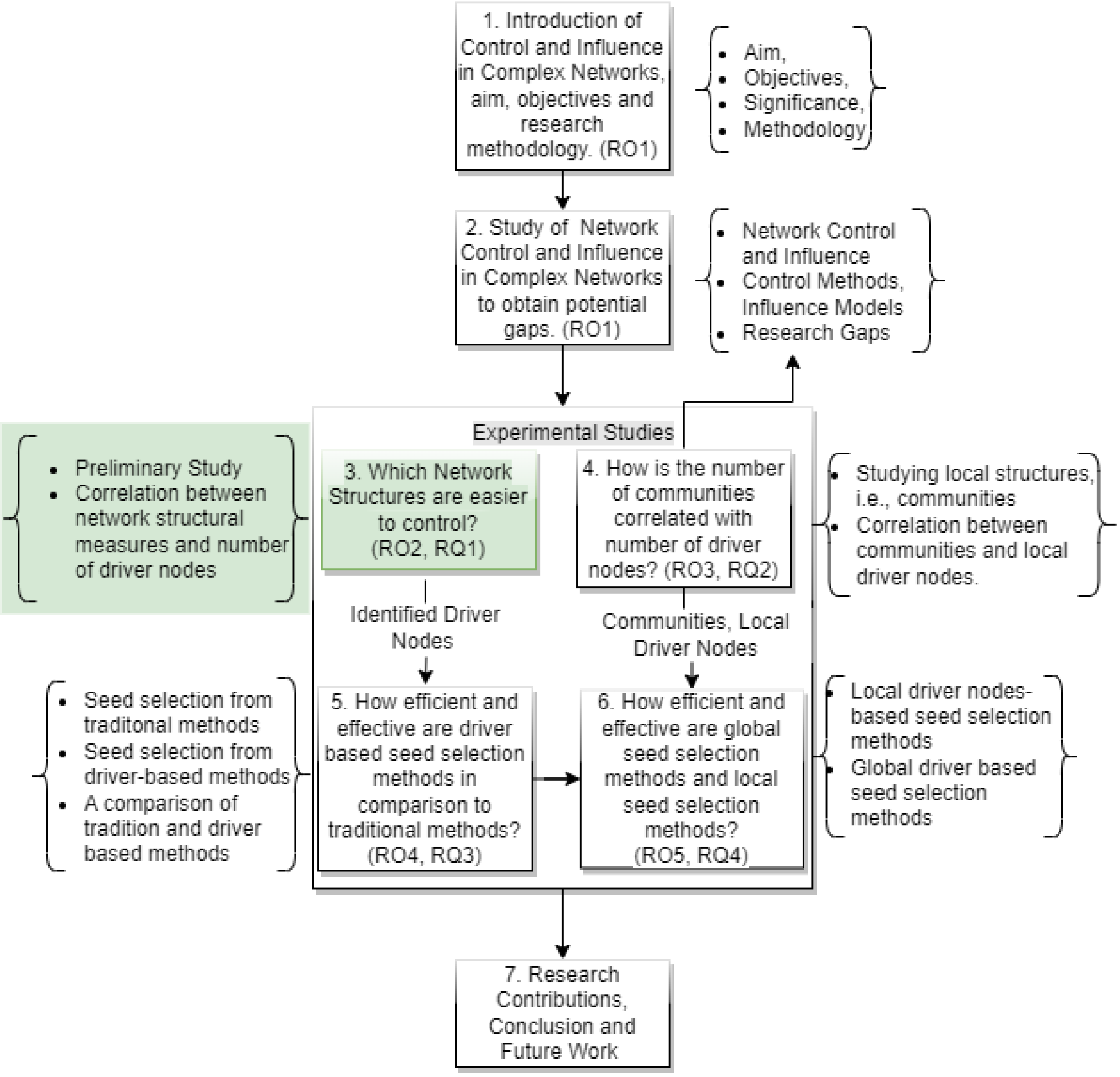

1. Introduction of Control and Influence in Complex Networks, aim, objectives and research methodology. (RO1)
Aim,
Objectives,
Significance,
Methodology
2. Study of Network Control and Influence in Complex Networks to obtain potential gaps. (RO1)
Network Control and Influence
Control Methods, Influence Models
Research Gaps
Experimental Studies
Preliminary Study
Correlation between network structural measures and number of driver nodes
3. Which Network Structures are easier to control? (RO2, RQ1)
4. How is the number of communities correlated with number of driver nodes? (RO3, RQ2)
Studying local structures, i.e., communities
Correlation between communities and local driver nodes.
Identified Driver Nodes
Communities, Local Driver Nodes
Seed selection from traditonal methods
Seed selection from driver-based methods
A comparison of tradition and driver based methods
5. How efficient and effective are driver based seed selection methods in comparison to traditional methods? (RO4, RQ3)
6. How efficient and effective are global seed selection methods and local seed selection methods? (RO5, RQ4)
Local driver nodes-based seed selection methods
Global driver based seed selection methods
7. Research Contributions, Conclusion and Future Work

Figure 3.1 : Research Methodology : Chapter 3

ing them and studying their correlation with network structure measures can bring valuable insights, such as what network structures are easier to control, and how we can alter the structure in our favour to achieve the maximum control over the network. The motivation behind this study is to understand if there is a strong or weak correlation between network structural measures and number of driver nodes. This information is necessary to find out which ranking mechanisms could work when choosing the best possible driver nodes as seed nodes. Since we see, influence as a type of control in the network, gaining this understanding is important to identify the influential nodes in the network, that can carry out the process of spreading influence to other nodes in the network efficiently. This chapter includes the following sections: Section 3.1 describes related work and the main research challenge that is the focus of this study; Section 3.2, describes the research methodology and experimental design in detail. Section 3.3, includes results and analysis of the experiments performed; finally the conclusions drawn from the experiments are discussed in Section 3.4.

## 3.1 Background

In the real world, many complex systems can be represented as complex networks [166, 58, 57, 59, 134]. Understanding how to control these networks is a critical, but still relatively unexplored research direction [121]. In order to take up this challenge, we first need to understand how the structural measures of different types of network influence their controllability and different control mechanisms. Identifying a set of driver nodes in complex networks plays a very important role in controlling a complex network. The full control of social networks is very hard to achieve due to their dynamics and complex human behaviour that cannot be fully controlled. However, we can still find out the potential relationships between the number of driver nodes and underlying structure of the network.

In this chapter, we aim to explore these possible dependencies between number of driver nodes, density of number of driver nodes and network structural measures.

For that matter, we employ minimum dominating set(MDS) method to determine number of driver nodes in generated and real networks.

## 3.2 Research Methodology

The research gap identified from prior work includes the identification of a network structure that can be controlled with the minimum number of driver nodes. The experimental objective is to find out the network structures that are easier to control.

The investigation is based on examining three commonly used network models: random, small-world, and scale-free and twenty two real social networks. First, the networks are generated with a varying number of nodes and edges, and then the structural measures are calculated giving the network profiles. We have used the minimum dominating set [147] method to calculate number of driver nodes. The number of edges is increased progressively *(*i.e., to increase network density) as described below and for the generated networks we calculate the number of driver nodes and density of driver nodes that may be needed to fully control the network using the minimum dominating set approach, which is described in Chapter 2.6, Section 2.4, Figure 2.10. In the Figure 2.10, the network is structurally controllable by selecting an MDS because each dominated node has its own control signal. The maximum matching approach needs three driver nodes $a$, $b$, and $d$, assuming a matching link from $a$ to $c$. In contrast, the MDS only requires one node i.e., $a$. Where $a$ can assume control of $b$, $c$ and $d$. However, finding a minimum dominating set is an NP-Hard problem in general, that means that it is not possible to calculate it in polynomial time. However different algorithms have been proposed [142] that help ameliorate this complexity barrier somewhat.

After the number of driver nodes is assessed for each network, the main goal is to check if there is any *and if so what* relationship between the number of driver nodes in a given network to the structure of this network. We express network structure as a collection of network measures, e.g, number of nodes [154], number of edges [154], network density [154], betweenness centrality [56], closeness centrality [179] and eigenvector centrality [12, 13].

We aim to compare different network structures *(random (R) [46], small-world (SW) [216], scale-free (SF) [1])* with the same number of nodes and edges and varying other network parameters as described in detail later in this section. We also use real networks as well. We investigate the patterns and relations between structural measures and the number of driver nodes $N_d$ in both artificial and real networks. We also calculate the ratio of $N_d$ to the total number of nodes $N$ in a network, which can be defined as $N_d \backslash N$, and called number of driver nodes density or driver nodes density. First, however, we turn to a detailed description of the dataset used in the experiments. The sizes of generated networks $(R, SW, SF)$ have been kept the same for all three network models to see if we are able to achieve minimum number of driver nodes $N_d$. In [127], Liu et al. argued and predicted the number of driver nodes based upon their degree distributions. However, when we analyse our networks (R, SW, SF) further by taking into account their density as the main measure for comparison, we see that as we increase the density of the network (i.e., by increasing number of edges) we can minimise the number of driver nodes as much as equal to 1. It proposes an idea that we can predict number of driver nodes $N_d$ based upon the density of the network.

Several statistical analysis tests, like t-test can be applied to measure the accuracy of the prediction results. Regression Analysis is very commonly used analysis method for the numerical prediction. In this regard, Pearson product-moment correlation coefficient $(r)$ [75], the coefficient of determination $(r^2)$ [75], the weighted $r$ [108], and mean squared error (MSE) [75] are the most widely used measures for assessing predictive models for numerical data. These measures can be used to calculate the accuracy of the prediction results [42]. Next section highlights the similarities and differences between randomly generated networks and social networks.

### 3.2.1 Description of Networks

The structure and formation of the randomly generated networks that have been used in this experimental study is explained in the Section 2.2.2. However a social network is a network of social interactions and personal relationships [169]. In a social network, every actor *i.e.,a person, a group, an organisation or a nation* is represented as a node. A relation is represented as a link between these nodes [135]. Social networks are somewhat behave in scale-free fashion [22].

### 3.2.2 Network Profiles Used

Network Profiles are instantiations of network models. These are composed of generated networks R, SW and SF with their corresponding structural measures calculated. The idea is to increase number of nodes and number of edges to increase network density in order to achieve desired/ideal number of driver nodes in network models. Each network with specific number of nodes and number of edges has been generated ten times, to get ten different profiles for the same network. Then average of each of the measures has been calculated to see the overall landscape. The network structure measures were then combined to make a network profile of the respected network. We have extended the experiments to include real social networks. The datasets have been downloaded from Stanford*. The network profiles of real networks include, number of nodes, number of edges and network density. We further complete these network profiles by calculating number of driver nodes $N_d$, and density of number of driver nodes $N_dD = N_d \backslash N$.

### 3.2.3 Conducted Experiments

The following three experiments are conducted on to achieve the overall goal of identifying the networks that are easier to control.

*https://snap.stanford.edu/data/

1. The first experiment was the calculation of different centrality measures and network density in generated and real networks.
2. The second experiment was the identification of number of driver nodes from all the types of networks and their profiles by using MDS method.
3. The third experiment was to find the correlation between network density and centrality measures with number of driver nodes in those networks.

## 3.3 Result and Analysis

This section covers the results obtained from all the experiments and their analysis. Table 3.3 shows network profiles generated for R, SW and SF networks. Table 3.2 shows social networks and their structural characteristics. Following are the comparisons which have been carried out to answer the research question. The correlation between different network structure measures and number of driver nodes using Power law.

### 3.3.1 Results from Generated Networks

According to the results the denser the network, whether R, SW, and SF, the smaller is the number of driver nodes needed. In a denser network we have more connections and that is why smaller number of driver nodes is needed to reach the whole network. We observe that number of driver nodes can be minimised as close to or equal to 1 by increasing the density and number of edges in all the networks. Figure 3.2 shows the correlation between density and average number of driver nodes for R, SW and SF networks where the number of nodes are 100, 200, 300, 400 and 500. We can clearly see from the trend-lines that for all three networks, the denser the network the smaller the number of driver nodes. We have performed a statistical t-test analysis on these networks, and found out that they are significantly similar to each other in terms of density of number of driver nodes $N_dD$. We can see equations of the above mentioned trend-analysis from Table 3.1. We can also see that there is not much difference between the density of number of driver nodes $N_dD$ in the networks, as all networks were able to minimise the number of driver nodes to 1 with increased edges and density. We can see from Table 3.3 that as centrality values closeness centrality (CC), betweenness centrality (BC) and eigenvector centrality (EV) are increased the number of driver nodes $N_d$ decreases. These centrality measures has been used because of their extensive use in the literature in relation to the complex network analysis and influence modelling over networks. These centrality measures, enables us to understand the basic network structure without adding too much complexity of calculating the more intricate structural measures of the network. It is possible to build seed selection methods with other measures, but there is a basis to select these centrality measures because of their calculation simplicity. Also the seed selection methods based upon the centrality measures have been used effectively in the past research. If the underlying task would be different not control or influence then, other structural measures might be effective but not in the context of this study.

These results are very encouraging as they tell us the dependence of minimum set of driver nodes on the structural measures of a network. Table 3.4 shows the results of Student's t-Test [221] to see how much variance is there in the density of number of driver nodes in R, SW and SF networks. Table 3.4 shows that, R and SF networks are significantly different from each other, still network density in both of the networks plays an important part in determining the number of driver nodes.

### 3.3.2 Results from Social Networks

For a general outlook at the characteristics of social networks, we reported number of nodes and edges in Table 3.2. Table 3.2 also shows the density of all of these networks. Based upon this representation in Table 3.2, Diggs-Friends(D)[†] dataset

[†] https://www.isi.edu/ lerman/downloads/digg2009.html

has lowest density of $2 \cdot 10^{-6}$ and Youtube(Y)[‡] dataset has second lowest density of $4 \cdot 10^{-6}$ and third lowest density is of Twitter(T)[§] dataset which is 0.0001. Highest density is of Facebook(FB)[¶] *i.e.,*0.0108 dataset after Zachary's Karate club(Z)[‖] which is 0.139. If we look into $N_dD$ of these networks we have highest $N_dD$ is of Z (0.382), followed by FB (0.123), D (0.086), T (0.04) and Y (0.032). We aim to replicate our results by showing that, if the network is denser, it takes less number of driver nodes to control as we have seen in generated networks earlier. If we look at Figure 3.4(a,b), this confirms our hypothesis because Z has higher density of number of driver nodes *i.e.,* $N_dD$ than Y. That means Y requires more number of driver nodes than Z to control it.

To examine the relationship between number of driver nodes and other network structural measures of real social networks, we took into consideration twenty two online social networks. To see how the structures of these networks effect the changes in number of driver nodes, we compared online social networks and randomly generated networks (i.e., R, SW, SF) (see Figure 3.3). The figure shows the network density on $(x - axis)$ against the number of driver nodes density *i.e.,* $N_dD$ on $(y - axis)$ in different networks. From Figure 3.4(a,b) we can see that where $N_dD$ is highest (see Z network) in Figure 3.4(a,b) and lowest (see D network) in Figure 3.4(a,b). This is in line with the results of previous experiments with generated networks. We also looked into degree distributions of social networks for further analysis. An interesting outcome is presented in Figure 3.5. It shows degree distribution of FB and LF networks. Power law degree distribution can give a glimpse into the structure of a network and distinguish different types of networks. We can see the degree distribution in FB follows a skewed pattern which means that some nodes have very high degree and there are obviously hubs present in the network (i.e., large degree nodes

[‡] https://snap.stanford.edu/data/com-Youtube.html

[§] https://snap.stanford.edu/data/ego-Twitter.html

[¶] https://snap.stanford.edu/data/ego-Facebook.html

[‖] http://networkrepository.com/soc-karate.php are referred to as hubs). While, the degree distribution captures only a small amount of the network structure, as it ignores how the nodes are connected to each other but, we can still see that if more number of nodes have high degree or not.Despite the change in the distribution, they follow the results of generated networks where when network density is higher, $N_dD$ are lower. FB networks has higher density and $N_dD$ is 0.12 i.e. than LF i.e. 0.003, where $N_dD$ is 0.39. That means, density of network does impact in determining number of driver nodes. In Figure 3.6 red blocks mean minimum similarity, green blocks mean the maximum similarity and light green blocks mean moderate similarity between the structures of the matrices in relation. The point of emphasis is that even moderate similarity values are high. As we have discussed earlier in this section that Z and FB datasets are quite similar structurally with R, SW and SF networks but R, SW are not structurally similar to SF as evident from T-test shown in the Table 3.4. Table 3.4 helps in emphasising the point of relationship between network density and number of driver nodes density. As most similarity is in those cases where, network density and driver nodes density is highest as compared to the other networks. We can see some other networks *i.e., T, D, Y, G, LF, DRO, MG, LA and FBA* which are also quite similar structurally the generated networks. While rest of the networks may not be ideally similar but they also lie on the spectrum of being quite similar with similarities in the range of 0.82 to 0.93, represented by red blocks in Figure 3.6.

Table 3.1 : Trend-line Equations for R, SW, SF networks; Where $X$ denotes number of driver nodes and $Y$ denotes the network structural measures

| Nodes | Trendndlines | | |
|---|---|---|---|
| | Random | Small-world | Scale-free |
| N = 100 | y = 1.8725x-1.377 | y = 1.8605x-1.362 | y = 4.8481x-0.967 |
| N = 200 | y = 2.2748x-1.315 | y = 2.0849x-1.379 | y = 6.9247x-0.901 |
| N = 300 | y = 2.2748x-1.315 | y = 2.4338x-1.251 | y = 9.0069x-0.715 |
| N = 400 | y = 2.9337x-1.162 | y = 2.7753x-1.213 | y = 5.4731x-1.042 |
| N = 500 | y = 3.3727x-1.1 | y = 3.3727x-1.1 | y = 6.7544x-1.022 |

Table 3.2 : Online Social Networks Profiles

| Social Networks | Nodes | Edges | Density |
|---|---|---|---|
| Zachary's Karate Club (Z) [233] | 34 | 78 | 0.13903 |
| Facebook (FB) [138] | 4039 | 88234 | 0.01082 |
| Twitter (T) [138] | 23371 | 32832 | 0.00012 |
| Diggs-Friends (D) [84] | 1924000 | 3298475 | $2{\cdot}10^{-6}$ |
| Youtube (Y) [225] | 1134891 | 2987625 | $4{\cdot}10^{-6}$ |
| Ego-gplus (EG) [138] | 23629 | 39195 | 0.00014 |
| Librec-ciaodvdnetwork (LC) [112] | 4658 | 33116 | 0.00305 |
| Librec-filmtrust-trust (LF) [70] | 874 | 1309 | 0.00343 |
| petster-frienships-hamster-uniq (P) [173] | 1858 | 12534 | 0.00726 |
| musae-facebook-edges [175] | 22470 | 171002 | 0.00067 |
| Deezer-HR-edges (DHE) [176] | 54574 | 498202 | 0.00033 |
| Deezer-RO-edges (DRE) [176] | 41774 | 125826 | 0.00014 |
| Deezer-HU-edges (DHE) [176] | 47539 | 222887 | 0.00019 |
| musae-git-edges (MG) [175] | 37700 | 289003 | 0.00040 |
| lastfm-asia-edges (LA) [177] | 7624 | 27806 | 0.00095 |
| fb-artist-edges (FBA) [176] | 50516 | 819306 | 0.00064 |
| fb-athletes-edges (FBAT) [176] | 13867 | 86858 | 0.00090 |
| fb-government-edges (FBG) [176] | 7058 | 89455 | 0.00359 |
| fb-new-sites-edges (FBN) [176] | 27918 | 206259 | 0.00053 |
| fb-politician-edges (FBP) [176] | 5909 | 41729 | 0.00239 |
| fb-public-figure-edges (FBPF) [176] | 11566 | 67114 | 0.001003 |
| fb-tvshow-edges (FBT) [176] | 3893 | 17262 | 0.00228 |

Table 3.3 : Random, Small-World, Scale-free Network Profiles Representing BC(Betweenness Centrality), EV(Eigenvector Centrality), CC(Closeness Centrality), $N_dD$(Driver Nodes Density), ND(Network Density)

| Nodes | Edges | Random | | | | | Small-World | | | | | Scale-Free | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | BC | EV | CC | $N_dD$ | ND | BC | EV | CC | $N_dD$ | ND | BC | EV | CC | $N_dD$ | ND |
| 100 | 800 | 0 | 0.097 | 0.527 | 0.18 | 0.162 | 0 | 0.1 | 0.52 | 0.178 | 0.16 | 0.00979 | 0.088 | 0.514 | 0.22 | 0.149 |
| | 1600 | 0.00031 | 0.099 | 0.597 | 0.09 | 0.323 | 0 | 0.1 | 0.6 | 0.095 | 0.32 | 0.00298 | 0.092 | 0.58 | 0.17 | 0.272 |
| | 2400 | 0.00196 | 0.1 | 0.661 | 0.06 | 0.485 | 0 | 0.1 | 0.66 | 0.068 | 0.48 | 0.00379 | 0.095 | 0.617 | 0.12 | 0.368 |
| | 3200 | 0.00361 | 0.1 | 0.74 | 0.05 | 0.646 | 0 | 0.1 | 0.74 | 0.045 | 0.65 | 0.00455 | 0.095 | 0.649 | 0.09 | 0.44 |
| | 4000 | 0.00526 | 0.1 | 0.84 | 0.03 | 0.808 | 0.01 | 0.1 | 0.84 | 0.033 | 0.81 | 0.00572 | 0.093 | 0.675 | 0.07 | 0.56 |
| | 4800 | 0.00691 | 0.1 | 0.971 | 0.02 | 0.97 | 0.01 | 0.1 | 0.97 | 0.019 | 0.97 | 0.00644 | 0.095 | 0.777 | 0.05 | 0.71 |
| | 4950 | 0.00919 | 0.1 | 1 | 0.01 | 1 | 0.01 | 0.1 | 1 | 0.01 | 1 | 0.00751 | 0.098 | 0.877 | 0.03 | 0.877 |
| 200 | 2400 | 0.00468 | 0.07 | 0.52 | 0.133 | 0.12 | 0.00476 | 0.07 | 0.515 | 0.135 | 0.121 | 0.06136 | 0.509 | 1.974 | 0.145 | 0.113 |
| | 4800 | 0 | 0.07 | 0.57 | 0.076 | 0.24 | 0 | 0.07 | 0.569 | 0.08 | 0.241 | 0.06428 | 0.561 | 1.789 | 0.14 | 0.212 |
| | 7200 | 0.00018 | 0.07 | 0.61 | 0.054 | 0.36 | 0.00018 | 0.071 | 0.611 | 0.05 | 0.362 | 0.0654 | 0.617 | 1.633 | 0.125 | 0.367 |
| | 9600 | 0.00079 | 0.071 | 0.66 | 0.04 | 0.48 | 0.00079 | 0.071 | 0.66 | 0.045 | 0.499 | 0.06632 | 0.66 | 1.537 | 0.095 | 0.463 |
| | 12000 | 0.0014 | 0.071 | 0.72 | 0.028 | 0.6 | 0.0014 | 0.071 | 0.716 | 0.025 | 0.603 | 0.06654 | 0.67 | 1.518 | 0.075 | 0.482 |
| | 14400 | 0.002 | 0.071 | 0.784 | 0.023 | 0.724 | 0.002 | 0.071 | 0.784 | 0.02 | 0.724 | 0.06693 | 0.621 | 1.638 | 0.06 | 0.567 |
| | 16800 | 0.00261 | 0.071 | 0.866 | 0.017 | 0.844 | 0.00322 | 0.071 | 0.866 | 0.015 | 0.844 | 0.06743 | 0.654 | 1.543 | 0.04 | 0.654 |
| | 19200 | 0.00322 | 0.071 | 0.966 | 0.012 | 0.965 | 0.00383 | 0.071 | 0.966 | 0.01 | 0.965 | 0.06755 | 0.729 | 1.433 | 0.025 | 0.787 |
| | 19900 | 0.00383 | 0.07 | 1 | 0.005 | 1 | 0.02564 | 0.071 | 1 | 0.005 | 1 | 0.06763 | 0.777 | 1.343 | 0.02 | 0.898 |
| 300 | 12800 | 0.0024 | 0.058 | 0.583 | 0.047 | 0.285 | 0.00239 | 0.058 | 0.584 | 0.05 | 0.288 | 0.00223 | 0.054 | 0.605 | 0.09 | 0.337 |
| | 19200 | 0 | 0.058 | 0.636 | 0.031 | 0.428 | 0 | 0.058 | 0.636 | 0.03 | 0.428 | 0.0012 | 0.055 | 0.617 | 0.083 | 0.366 |
| | 22400 | 0.00024 | 0.058 | 0.667 | 0.027 | 0.499 | 0.00024 | 0.058 | 0.668 | 0.027 | 0.502 | 0.0015 | 0.055 | 0.628 | 0.063 | 0.392 |
| | 25600 | 0.00048 | 0.058 | 0.7 | 0.023 | 0.571 | 0.00048 | 0.058 | 0.699 | 0.03 | 0.569 | 0.00168 | 0.055 | 0.65 | 0.053 | 0.441 |
| | 28800 | 0.00072 | 0.058 | 0.737 | 0.02 | 0.642 | 0.00071 | 0.058 | 0.737 | 0.017 | 0.642 | 0.00176 | 0.057 | 0.584 | 0.05 | 0.456 |
| | 32000 | 0.00096 | 0.058 | 0.778 | 0.013 | 0.713 | 0.00086 | 0.058 | 0.795 | 0.013 | 0.742 | 0.00179 | 0.056 | 0.636 | 0.047 | 0.428 |
| | 35200 | 0.0012 | 0.058 | 0.823 | 0.013 | 0.785 | 0.0012 | 0.058 | 0.826 | 0.01 | 0.789 | 0.00187 | 0.057 | 0.668 | 0.037 | 0.502 |
| | 38400 | 0.00144 | 0.058 | 0.875 | 0.012 | 0.856 | 0.00145 | 0.058 | 0.874 | 0.01 | 0.856 | 0.00192 | 0.058 | 0.699 | 0.027 | 0.569 |
| | 41600 | 0.00168 | 0.058 | 0.933 | 0.01 | 0.928 | 0.00167 | 0.058 | 0.935 | 0.007 | 0.93 | 0.00204 | 0.058 | 0.737 | 0.023 | 0.642 |
| | 44850 | 0.00192 | 0.06 | 1 | 0.003 | 1 | 0.00192 | 0.058 | 1 | 0.003 | 1 | 0.00213 | 0.058 | 0.748 | 0.017 | 0.742 |

| Nodes | Edges | Random | | | | | Small-World | | | | | Scale-Free | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | BC | EV | CC | $N_dD$ | ND | BC | EV | CC | $N_dD$ | ND | BC | EV | CC | $N_dD$ | ND |
| 400 | 40000 | 0.00125 | 0.05 | 0.667 | 0.02 | 0.501 | 0.00125 | 0.05 | 0.667 | 0.023 | 0.501 | 0.00125 | 0.049 | 0.657 | 0.045 | 0.301 |
| | 44000 | 0 | 0.05 | 0.691 | 0.015 | 0.551 | 0 | 0.05 | 0.69 | 0.018 | 0.551 | 0.00032 | 0.05 | 0.67 | 0.038 | 0.351 |
| | 48000 | 0.00012 | 0.05 | 0.715 | 0.015 | 0.602 | 0.00012 | 0.05 | 0.715 | 0.015 | 0.602 | 0.00052 | 0.05 | 0.695 | 0.033 | 0.401 |
| | 52000 | 0.00025 | 0.05 | 0.742 | 0.015 | 0.652 | 0.00025 | 0.05 | 0.742 | 0.015 | 0.652 | 0.00065 | 0.05 | 0.722 | 0.03 | 0.451 |
| | 60000 | 0.00037 | 0.05 | 0.801 | 0.013 | 0.752 | 0.00037 | 0.05 | 0.801 | 0.013 | 0.752 | 0.00077 | 0.05 | 0.791 | 0.028 | 0.501 |
| | 64000 | 0.0005 | 0.05 | 0.835 | 0.01 | 0.802 | 0.0005 | 0.05 | 0.835 | 0.01 | 0.802 | 0.0009 | 0.05 | 0.815 | 0.023 | 0.675 |
| | 68000 | 0.00062 | 0.05 | 0.871 | 0.01 | 0.852 | 0.00062 | 0.05 | 0.871 | 0.008 | 0.852 | 0.00092 | 0.05 | 0.861 | 0.01 | 0.802 |
| | 72000 | 0.00088 | 0.05 | 0.911 | 0.008 | 0.902 | 0.00088 | 0.05 | 0.911 | 0.005 | 0.902 | 0.00098 | 0.05 | 0.901 | 0.008 | 0.852 |
| | 76000 | 0.001 | 0.05 | 0.955 | 0.005 | 0.952 | 0.001 | 0.05 | 0.955 | 0.005 | 0.952 | 0.001 | 0.05 | 0.935 | 0.005 | 0.902 |
| | 798000 | 0.00113 | 0.05 | 1 | 0.003 | 1 | 0.00113 | 0.05 | 1 | 0.003 | 1 | 0.00113 | 0.05 | 0.948 | 0.005 | 0.952 |
| 500 | 72000 | 0.00085 | 0.045 | 0.703 | 0.018 | 0.577 | 0.00087 | 0.045 | 0.699 | 0.018 | 0.569 | 0.0012 | 0.043 | 0.633 | 0.024 | 0.404 |
| | 76800 | 0 | 0.045 | 0.723 | 0.014 | 0.616 | 0 | 0.045 | 0.721 | 0.014 | 0.613 | 0 | 0.043 | 0.647 | 0.022 | 0.436 |
| | 81600 | 0.00024 | 0.045 | 0.743 | 0.012 | 0.654 | 0.00024 | 0.045 | 0.743 | 0.012 | 0.653 | 0.00024 | 0.043 | 0.682 | 0.016 | 0.5 |
| | 86400 | 0.00031 | 0.045 | 0.765 | 0.01 | 0.693 | 0.00032 | 0.045 | 0.765 | 0.01 | 0.693 | 0.00032 | 0.042 | 0.681 | 0.016 | 0.681 |
| | 91200 | 0.00039 | 0.045 | 0.788 | 0.01 | 0.731 | 0.00039 | 0.045 | 0.787 | 0.01 | 0.729 | 0.00039 | 0.041 | 0.699 | 0.016 | 0.721 |
| | 96000 | 0.00046 | 0.045 | 0.813 | 0.01 | 0.77 | 0.00049 | 0.045 | 0.802 | 0.01 | 0.754 | 0.001 | 0.04 | 0.71 | 0.018 | 0.771 |
| | 100800 | 0.00054 | 0.045 | 0.839 | 0.008 | 0.808 | 0.00054 | 0.045 | 0.837 | 0.008 | 0.806 | 0.00104 | 0.045 | 0.765 | 0.014 | 0.816 |
| | 105200 | 0.00062 | 0.045 | 0.865 | 0.006 | 0.843 | 0.00062 | 0.045 | 0.863 | 0.006 | 0.842 | 0.00113 | 0.045 | 0.799 | 0.012 | 0.842 |
| | 110000 | 0.00069 | 0.045 | 0.894 | 0.006 | 0.882 | 0.0007 | 0.045 | 0.894 | 0.006 | 0.882 | 0.00116 | 0.045 | 0.813 | 0.01 | 0.882 |
| | 124750 | 0.00077 | 0.045 | 1 | 0.002 | 1 | 0.00078 | 0.045 | 1 | 0.002 | 1 | 0.00136 | 0.045 | 0.857 | 0.01 | 0.898 |

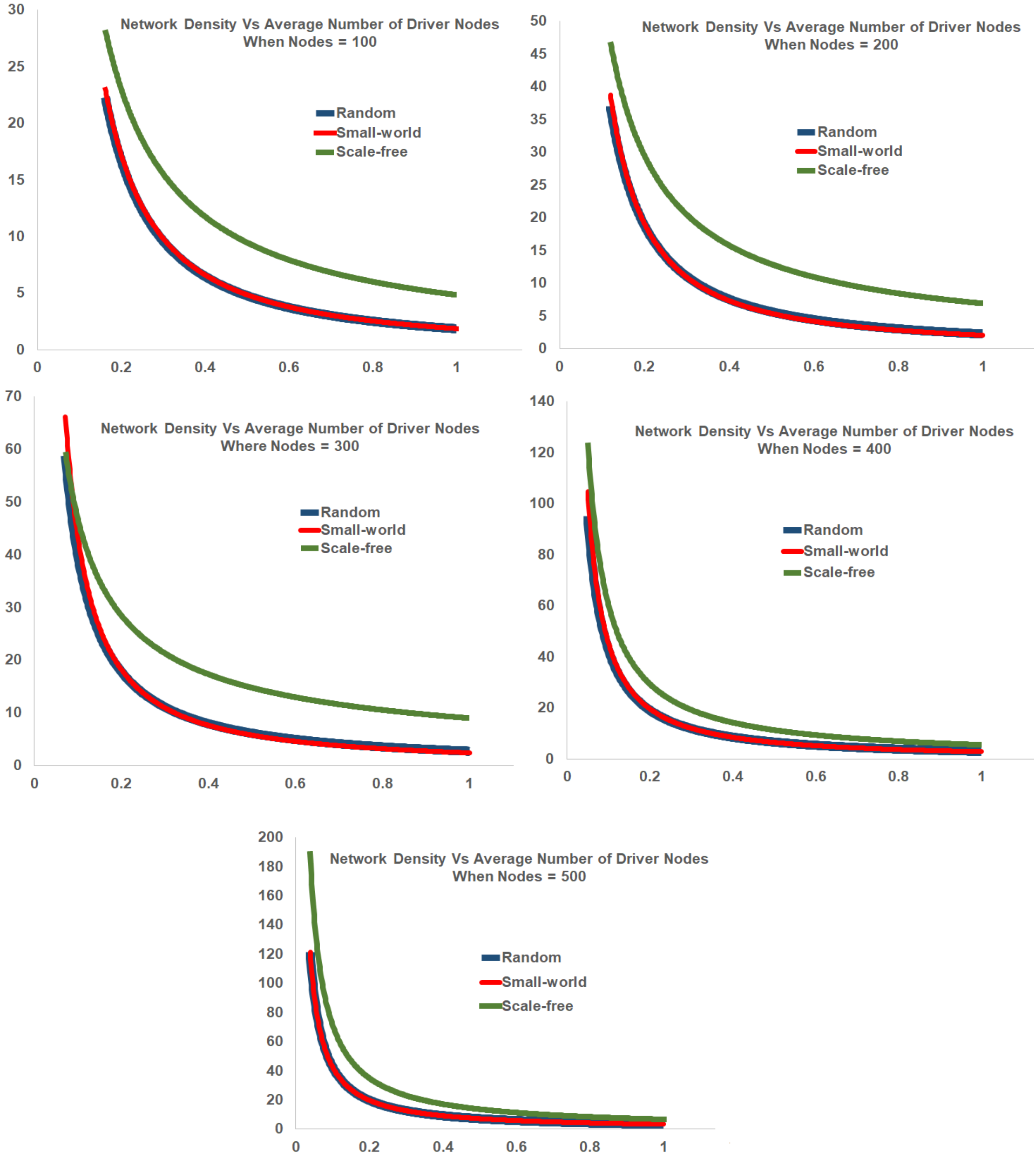
Table 3.4 : P-Value in Random, Small-world, Scale-free, P-Value denotes the probability measure of likely difference between all the groups of networks.

| t-Test (P-Value) | Random | Small-world | Scale-free |
|---|---|---|---|
| Random | | 0.966 | 0.001 |
| Small-world | 0.966 | | 0.287 |
| Scale-free | 0.001 | 0.287 | |

## 3.4 Discussion and Conclusion

The main goal of this study was to find out if there exists any correlation between network structural measures and the number of driver nodes needed for network control. This study completes the Research Challenge RC2, Research Question RQ1 and Research Objective RO2. RO2 states, "To find out which network structures can result in minimum number of driver nodes". The experiments were carried out for various network profiles *i.e., random, small-world, scale-free* and a wide range of numbers of nodes and edges in a network. A number of structural measures were computed in order to observe any correlation with the number of driver nodes. The relationship between structural measures and density of number of driver nodes has not been identified previously in relevant literature. Hence, we provide a novel understanding of which network structures are easier to control.

The main finding of the work is that the network structural measures do indeed correlate with the number of driver nodes. When the values of the investigated structural measures increase or decrease, this directly triggers the increase or decrease in the number of driver nodes. We found out that the denser the network, the smaller the number of driver nodes (see Figure 3.2). When density is equal to 1 (fully connected network), then the minimum number of driver nodes is reached. The same inference can be made from Table 3.3, where the number of edges is correlated with


Network Density Vs Average Number of Driver Nodes When Nodes = 100

Network Density Vs Average Number of Driver Nodes When Nodes = 200

Network Density Vs Average Number of Driver Nodes Where Nodes = 300

Network Density Vs Average Number of Driver Nodes When Nodes = 400

Network Density Vs Average Number of Driver Nodes When Nodes = 500

Figure 3.2 : Density Vs Number of Driver Nodes. X-axis shows the network density and y-axis shows the number of driver nodes.

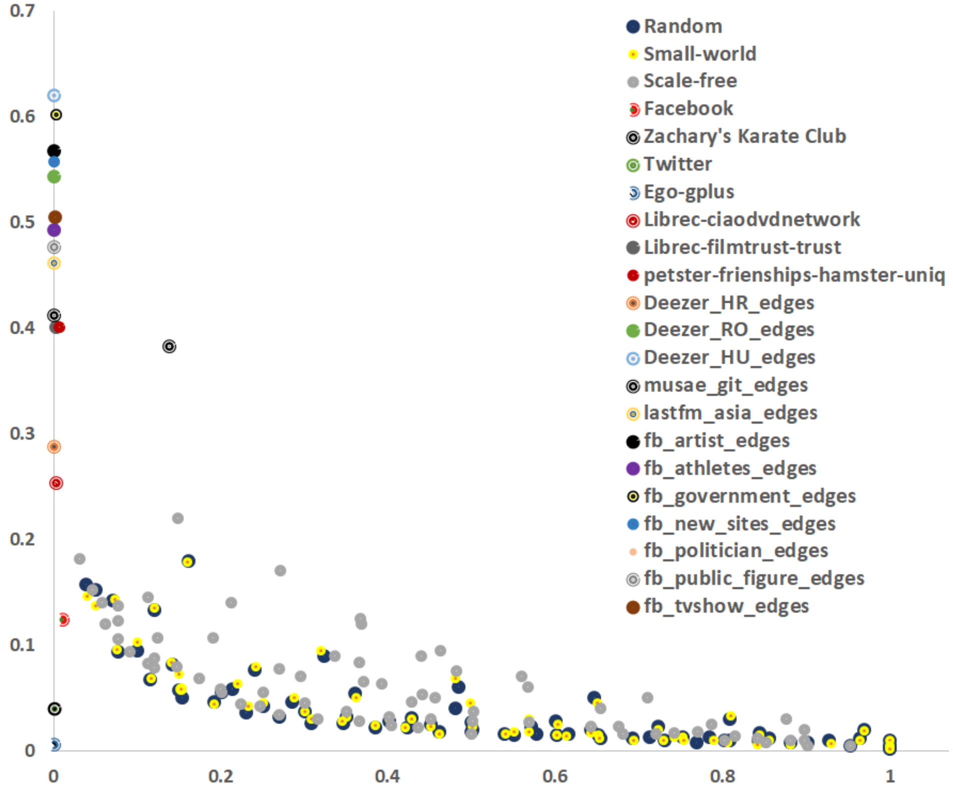


Figure 3.3 : Network Density Vs Driver Nodes Density of Random, Small-world, Scale-free and Social Networks, x–axis shows Network Density and y–axis shows Driver Nodes Density

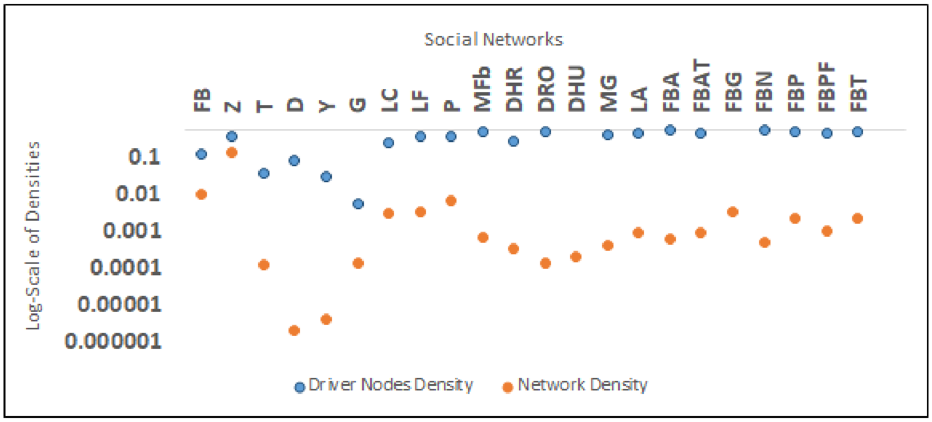


Figure 3.4 : Social Networks Representing Driver Nodes Density, and Network Density

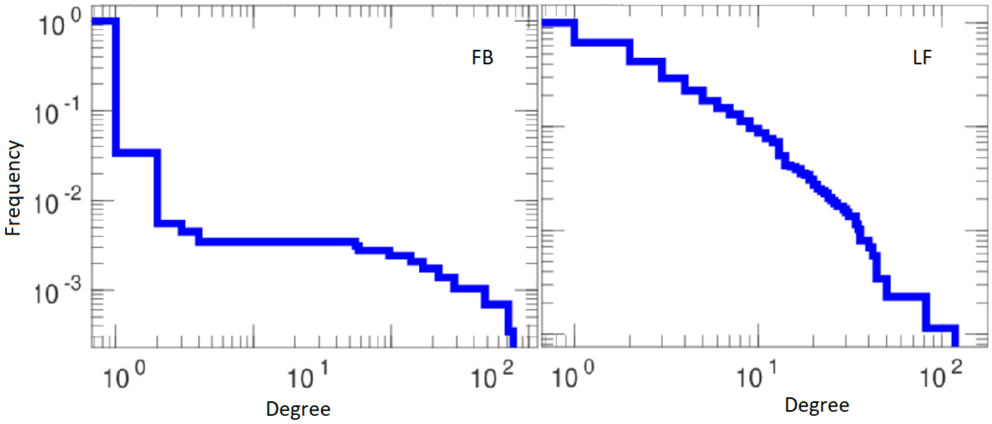


Figure 3.5 : A degree distribution of social networks i.e. FB and LF

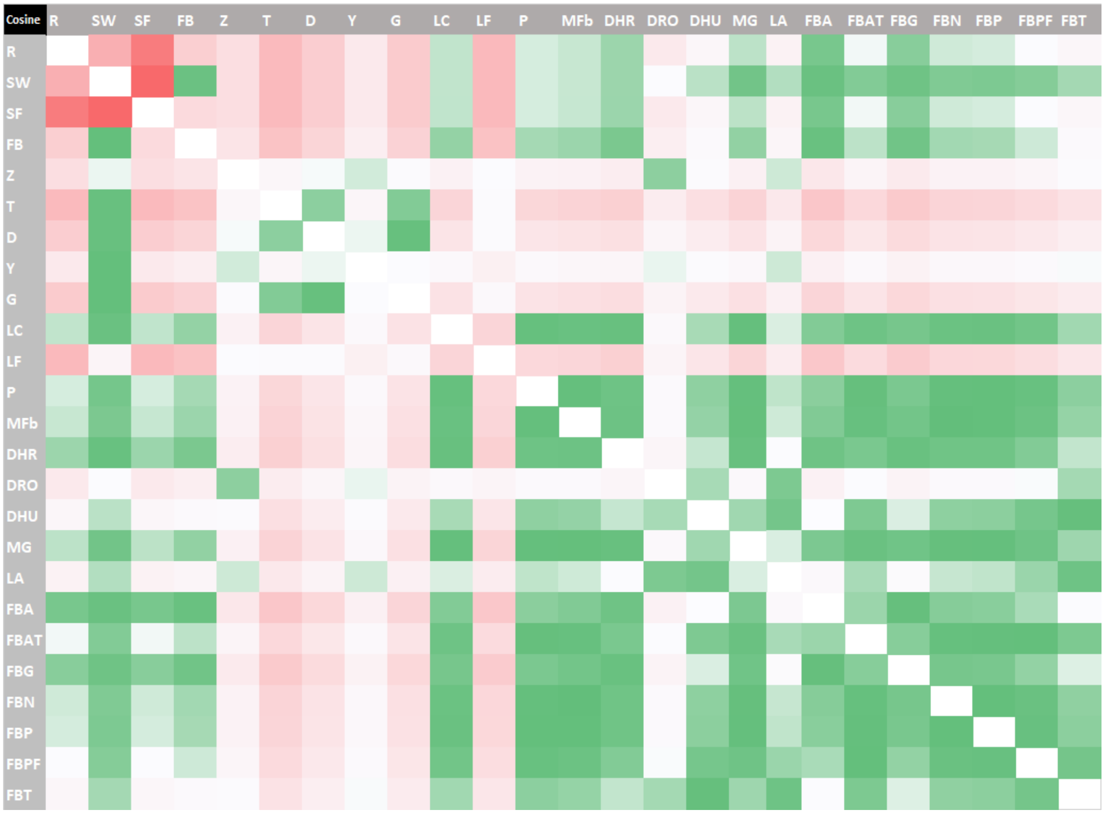


Figure 3.6 : Cosine Similarity Matrix of Generated and Social Networks

the number of driver nodes. Since increasing the number of edges directly correlates with the density of the networks we can see that the larger the number of edges the needed number of driver nodes decreases.

Correlations between the number of driver nodes and various centrality measures such as CC, EV, and BC show that the centrality measures are the indicators of how capable a node is in carrying and passing information to other nodes. For example, betweenness centrality quantifies the number of times a node acts as a bridge along the shortest path between two other nodes and is usually interpreted as a power of a node. It means that high values of centrality measures indicate that the nodes are capable of passing along the information to other nodes which can translate into being able to potentially influence and control their neighbourhood and in consequence the whole network.

We conclude that if we want to achieve control over a network with a minimum number of driver nodes then increasing the density of the network along with the other centrality measures can help in changing the structure of the network. Denser structure enables to achieve a smaller number of driver nodes. But, a natural next step will be to test what is the cost and purpose of building new relationships in different scenarios, e.g. social networks. We identified $N_d$ and $N_dD$ in various social networks, as can be seen in Figure 3.3. We can see from Figure 3.3 that social networks show different behaviours as compared to generated networks because of their very large sizes and low densities. Previously, researchers have suggested that a number of driver nodes greatly depend on node degree distribution rather than other structural measures [126]. However, we argue, based upon our experimental results, that it is enough to look at the global structural measures *i.e., network density* as they play an important role in determining if an underlying network is easy to control. Though the experimental results are promising, it is important to note that other existing methods for identifying driver nodes may yield different results.

# Chapter 4

# Driver Nodes in Communities

This chapter expands the goal of Chapter 3 by adding a new Research Challenge RC3, which states that, "Inquiring about the relationship between local network structures and the number of driver nodes in the networks." We begin by proposing research question RQ2 which states that, "How is the number of communities correlated with the number of driver nodes?". As we have seen in Chapter 2, the number of driver nodes is correlated with the global structural measures such as network density. So, it is plausible to consider the community structures of those networks in this regard. Communities in the network have potentially higher densities as compared to overall network density. So, finding that influential set of driver nodes from communities of the networks, can help carry control or influence efficiently and effectively. For this purpose, Research Objective RO3 has been defined as, "To find out correlations between local network structural measures and number of driver nodes.". In Figure 4.1 we can see the highlighted tasks that were carried out to fulfill RO3.

Due to the extensive use of online social networking sites, social networks are studied at great length in network science. In social networks, finding communities is a difficult task because of its structure that is complex, dynamically changing and the communities themselves can be overlapping [187]. In case of community detection, the methods to detect communities have been applied to identify terrorists organisations [183], recommending products for users [119], anomaly detection [208, 105], finding potential friends in social media [241] and analysing social opinions [207] to name a few.

It is not known yet that if and how the network structure correlates with the

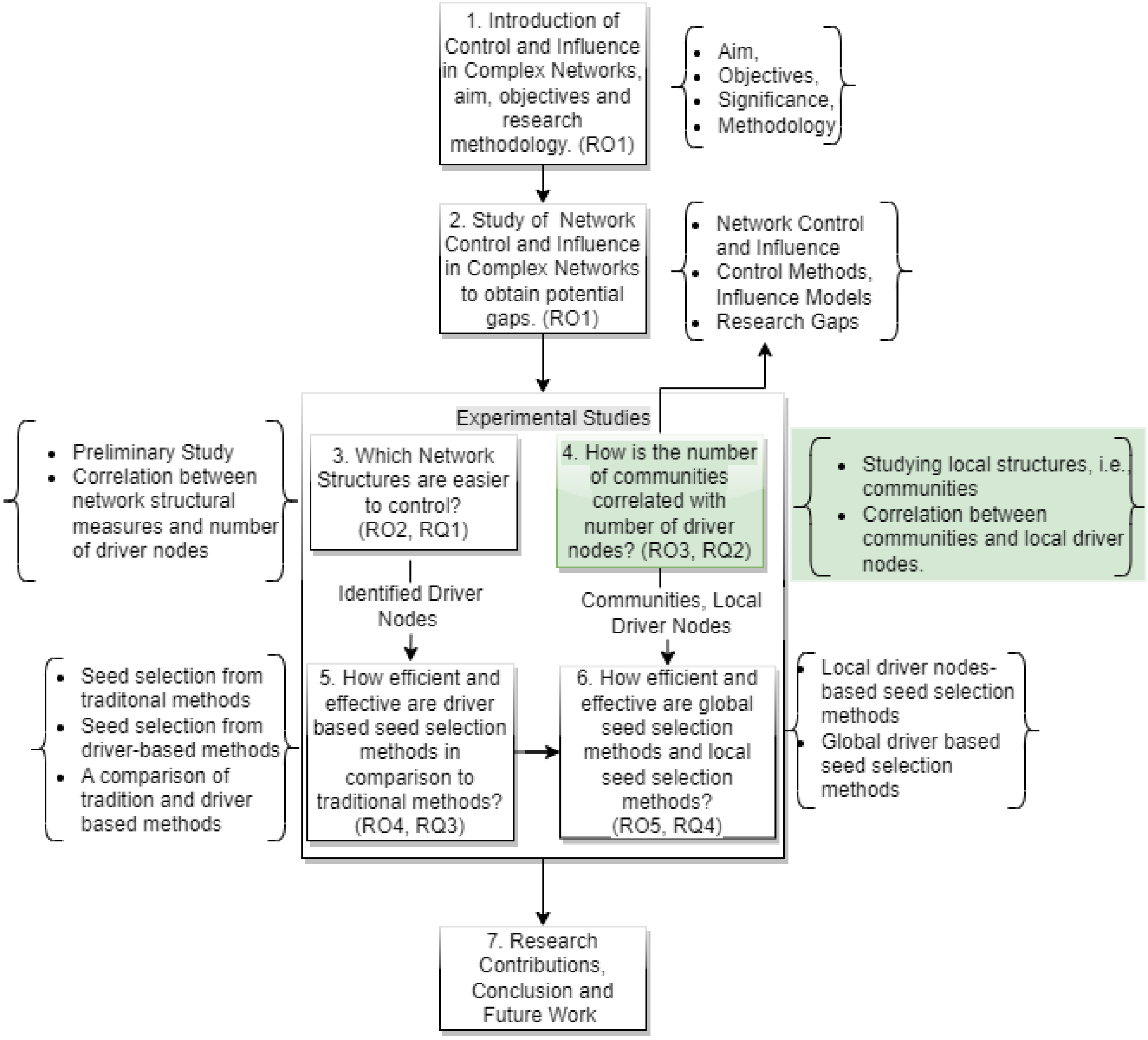

1. Introduction of Control and Influence in Complex Networks, aim, objectives and research methodology. (RO1)
Aim,
Objectives,
Significance,
Methodology
2. Study of Network Control and Influence in Complex Networks to obtain potential gaps. (RO1)
Network Control and Influence
Control Methods, Influence Models
Research Gaps
Experimental Studies
Preliminary Study
Correlation between network structural measures and number of driver nodes
3. Which Network Structures are easier to control? (RO2, RQ1)
4. How is the number of communities correlated with number of driver nodes? (RO3, RQ2)
Studying local structures, i.e. communities
Correlation between communities and local driver nodes.
Identified Driver Nodes
Communities, Local Driver Nodes
Seed selection from traditonal methods
Seed selection from driver-based methods
A comparison of tradition and driver based methods
5. How efficient and effective are driver based seed selection methods in comparison to traditional methods? (RO4, RQ3)
6. How efficient and effective are global seed selection methods and local seed selection methods? (RO5, RQ4)
Local driver nodes-based seed selection methods
Global driver based seed selection methods
7. Research Contributions, Conclusion and Future Work

Figure 4.1 : Research Methodology: Chapter 4

number of driver nodes [182]. As driver nodes play a key role in achieving control of a complex network, identifying them and studying their correlation with network structure measures can bring valuable insights, such as what network structures are easier to control, and how can we alter the structure to achieve the maximum control over the network. Our previous research work [182] determines the relationship between some of the global network structure measures and number of driver nodes. A systematic study presented in [182] builds an understanding of how global network profiles of synthetic (random, small-world, scale-free) and real social networks influence the number of driver nodes needed for control. In Chapter 3, we focused on global structural measures such as network density and how it can play an important role in determining how big or small the driver nodes set will be. Our results show that as density increases in networks like random, small world and scale free, the number of driver nodes tends to decrease.

This chapter focuses on both global and local structural measures and their relationship with number of driver nodes. We propose that communities are found to be one of the most important features of networks, and detecting them enables us to analyse and explore further underlying structural features of the synthetic as well as social networks. The idea is to detect communities and driver nodes within the communities to see how the number of communities influences the number of driver nodes. Based upon the review of the previous research work, we have formulated the following research challenge. Identifying a set of driver nodes in complex networks has always been very vital in control of a complex network and can start by detecting the potential relationships between number of driver nodes and underlying structural measures (global and local). This chapter contains the following sections: Section 4.1 describes related work and the main research challenge that is the focus of this study. Sections 4.2 and 4.3 describe (i) the research methodology and experimental design in detail and (ii) include results and analysis of the experiments performed respectively. Finally, the conclusions drawn from the experiments, future work and limitations are discussed in Sections 4.4.

## 4.1 Background

In social networks, identification of a minimum set of driver nodes is a potential research problem. This focuses on understanding how global and local network structures relate to the minimum number of driver nodes [40, 222, 126, 40, 73, 74, 236, 27, 239, 91]. In complex networks, small or large communities within the network are organically present. There are many definitions of communities in the networks, but in general they may be defined as a group of nodes which are densely connected with the other nodes in the group and sparsely connected to the nodes outside that group [64].

### 4.1.1 Community Detection

There are many community detection algorithms in use, for example GN [64], FN [64], CNM [64], LPA [170], EM [156], SCAN [223], Louvain [10], LFM [114], Infomap [174, 229] and NMF [237] to name a few.

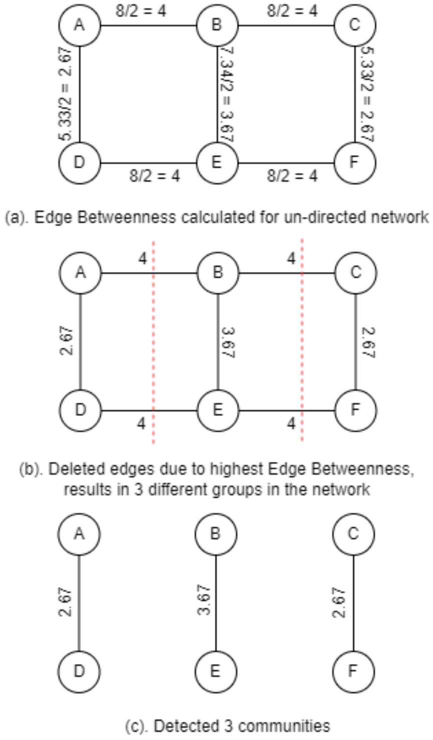
We start by utilising the widely used and tested algorithm, The Girvan-Newman algorithm (GN Algorithm). GN Algorithm is a benchmark in community detection and has been previously used to successfully identify communities in several different kinds of networks. The algorithm is simple and easy to implement. However it does have a trade-off in high time complexity. A brief description of the algorithm is given below, and an example of identified communities by using GN Algorithm from Zachare Karate Club is presented in Figure 4.3. The Girvan–Newman algorithm detects communities by progressively removing edges from the original graph. The algorithm removes the edge with the highest betweenness centrality, at each step. As the graph breaks down into pieces, community structures are exposed, and the result can be represented as a dendrogram. Below is the step-by-step process of the GN Algorithm.

1. Create a network of $N$ nodes and its edges.
2. Calculate the betweenness of all existing edges in the network.
3. Remove all the edges with the highest betweenness.
4. Recalculate the betweenness of all the edges that got affected by the removal of edges.
5. Repeat steps 3 and 4 until no edges remain.
6. Connected components are communities. [64]

For example, For a given network, we calculate the betweenness centrality for the edges. From Figure 4.2, highest betweenness centrality is between edges $AandB$, $BandC$, $DandE$, and $EandF$. So, we remove these edges in the next step and are left with three communities in the network.

### 4.1.2 Driver Nodes

Previously, some models/methods have been proposed for the identification of driver nodes that can potentially control the complex networks, for example; interbank networks [40], protein interaction networks [222], biological networks [126, 40, 73, 74], and real networks [236, 27, 239, 91]. Effects of local and global network structural measures including degree distribution have been explored in [192]. Many methods/algorithms have been proposed to identify a set of number of driver nodes from a network. These methods can be called control methods. The control methods that have been previously used to identify a set of driver nodes are based upon the algorithms, Maximum Matching Algorithm [89], Hopcroft-Karp Algorithm [89], Hungarian Algorithm [109] and Minimum Dominating Set [147]. Despite these algorithms, a complexity of choosing a smaller set of driver nodes still exists. If we have to restrict to this smaller set, we should have a ranking of driver nodes that allow us to pick those that have the largest impact on controlling the network. Existing measures for

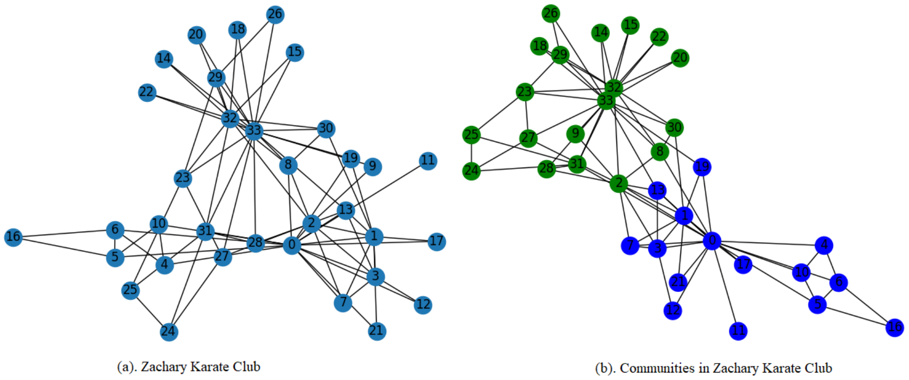


(a). Edge Betweenness calculated for un-directed network

(b). Deleted edges due to highest Edge Betweenness, results in 3 different groups in the network

(c). Detected 3 communities

Figure 4.2 : Example of Grivan-Newman Algorithm



(a). Zachary Karate Club

(b). Communities in Zachary Karate Club

Figure 4.3 : Representation of Zachary Karate Club Network and Detected Communities

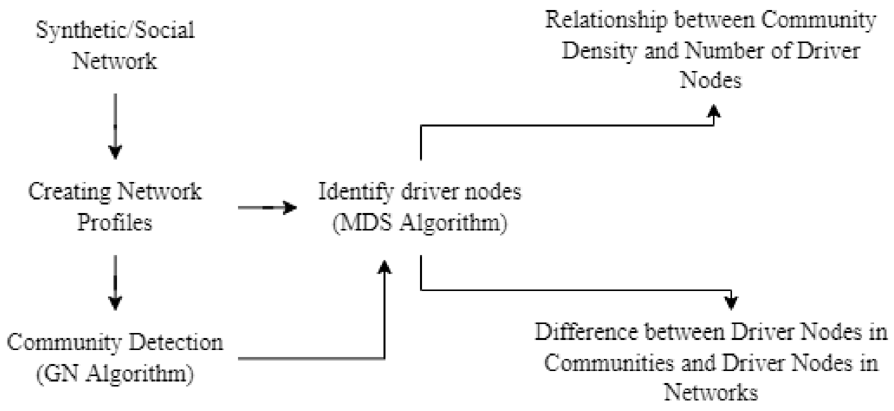

Synthetic/Social Network
Creating Network Profiles
Community Detection (GN Algorithm)
Identify driver nodes (MDS Algorithm)
Relationship between Community Density and Number of Driver Nodes
Difference between Driver Nodes in Communities and Driver Nodes in Networks

Figure 4.4 : Methodology

such a ranking, for example control capacity [96], and control range [205], are not suited because they only focus on one aspect of driver nodes, either their probability to become a driver or the size of the sub-network they control. Control contribution combines both of these two aspects [239]. In [239], researchers have claimed that control contribution will always be able to efficiently and effectively control the network, however, this requires further evaluation. In this section we will use the same MDS algorithm described in Chapter 2, Section 2.4, and Figure 2.10.

## 4.2 Research Methodology and Experiment Design

Figure 4.4 explains the methodology of the research work carried out to address RC3, RQ2 and RO3. In challenge RC2, the work revolved around the correlation between number of driver nodes and global network structure measures such as network density, as published in [182]. Below is the experiment design of this research work and a series of new experiments that were conducted to achieve the research questions mentioned in Section 4.1.

1. Detecting communities by using GN algorithm for all the generated and social networks.

2. A generalised framework has been developed which is suitable to detect communities from all networks. Nodes in the communities represent different sets.

We mean to analyse the following results from the above experiments. Firstly, correlation between community densities and number of driver nodes in those communities. Secondly, number of communities and number of driver nodes within those communities is identified to see if the driver nodes set is bigger or smaller than the set achieved by using Minimum Dominating Set method [147]. For the research questions identified in Section 4.1 and experiments proposed above, the following experimental setup has been adopted to conduct the experiments.

1. The same set of instances for random, small-world and scale-free networks are used as described in Chapter 3.

2. Calculate structural measures i.e., network densities (D) in generated as well as real networks as presented in Tables 4.3 and 4.1.

3. Identification of number of driver nodes ($N_dN$) in overall networks by using MDS method in both generated and social networks. $N_dN$ densities for generated networks are given in Table 4.1 and $N_dN$ for social networks are presented in Table 4.3.

4. Analyses the relationship between global structural measures, i.e., density with $N_dN$ as presented in [182].

5. Identify communities in synthetically generated networks and in social networks, the global structure measures of which are presented in Table 4.3. We utilised NetworkX library of Python programming language to generate networks. GN Algorithm has been used to detect communities in the network. A step-by-step process is given in Section 4.1. The algorithm and setup has been implemented in Python version 3.6. Also, the algorithm used a said threshold to stop, which is defined as the square root of the number of nodes in the network.

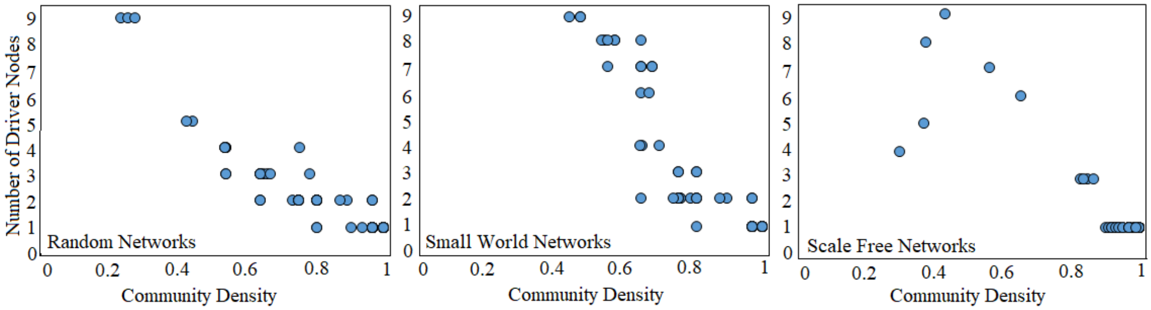


Figure 4.5 : Number of Driver Nodes in the Communities of Random, Small World and Scale Free networks versus their Community Densities

6. Identify driver nodes in communities in synthetic and social networks. We used Minimum Dominating Set Algorithm for identifying the driver nodes in communities. A description of the algorithm is given in the Section 2.4.

7. Correlation between community densities and number of driver nodes is done by obtaining densities of the communities and identifying number of driver nodes in those communities by MDS method.

8. The difference (Diff.) between total number of driver nodes identified in overall networks ($N_dN$) as compared to the number of driver nodes found in communities of those networks ($N_dNC$) is also obtained by obtaining results partially from the previous study [182], and largely from the current one. The Diff. tells us, the significance of identifying driver nodes within communities, like following a divide and conquer approach.

Table 4.1 : Global Network Structure Measures i.e., Number of Nodes, Number of Edges and Network Density $ND$ with their corresponding Number of Driver Nodes Density i.e. $N_dD$ in random, small-world and scale-free networks.

| Nodes | Random | | | Small world | | | Scale-free | | |
|---|---|---|---|---|---|---|---|---|---|
| | Edges | $ND$ | $N_dD$ | Edges | $ND$ | $N_dD$ | Edges | $ND$ | $N_dD$ |
| 100 | 800 | 0.18 | 0.162 | 800 | 0.178 | 0.16 | 800 | 0.22 | 0.149 |
| | 1600 | 0.09 | 0.323 | 1600 | 0.095 | 0.32 | 1600 | 0.17 | 0.272 |
| | 2400 | 0.06 | 0.485 | 2400 | 0.068 | 0.48 | 2400 | 0.12 | 0.368 |
| | 3200 | 0.05 | 0.646 | 3200 | 0.045 | 0.65 | 3200 | 0.09 | 0.44 |
| | 4000 | 0.03 | 0.808 | 4000 | 0.033 | 0.81 | 4000 | 0.07 | 0.56 |
| | 4800 | 0.02 | 0.97 | 4800 | 0.019 | 0.97 | 4800 | 0.05 | 0.71 |
| | 4950 | 0.01 | 1 | 4950 | 0.01 | 1 | 4950 | 0.03 | 0.877 |
| 200 | 2400 | 0.133 | 0.12 | 2400 | 0.135 | 0.121 | 2400 | 0.145 | 0.113 |
| | 4800 | 0.076 | 0.24 | 4800 | 0.08 | 0.241 | 4800 | 0.14 | 0.212 |
| | 7200 | 0.054 | 0.36 | 7200 | 0.05 | 0.362 | 7200 | 0.125 | 0.367 |
| | 9600 | 0.04 | 0.48 | 9600 | 0.045 | 0.499 | 9600 | 0.095 | 0.463 |
| | 12000 | 0.028 | 0.6 | 12000 | 0.025 | 0.603 | 12000 | 0.075 | 0.482 |
| | 14400 | 0.023 | 0.724 | 14400 | 0.02 | 0.724 | 14400 | 0.06 | 0.567 |
| | 16800 | 0.017 | 0.844 | 16800 | 0.015 | 0.844 | 16800 | 0.04 | 0.654 |
| | 19200 | 0.012 | 0.965 | 19200 | 0.01 | 0.965 | 19200 | 0.025 | 0.787 |
| | 19900 | 0.005 | 1 | 19900 | 0.005 | 1 | 19900 | 0.02 | 0.898 |
| 300 | 12800 | 0.047 | 0.285 | 12800 | 0.05 | 0.288 | 12800 | 0.09 | 0.337 |
| | 19200 | 0.031 | 0.428 | 19200 | 0.03 | 0.428 | 19200 | 0.083 | 0.366 |
| | 22400 | 0.027 | 0.499 | 22400 | 0.027 | 0.502 | 22400 | 0.063 | 0.392 |
| | 25600 | 0.023 | 0.571 | 25600 | 0.03 | 0.569 | 25600 | 0.053 | 0.441 |
| | 28800 | 0.02 | 0.642 | 28800 | 0.017 | 0.642 | 28800 | 0.05 | 0.456 |
| | 32000 | 0.013 | 0.713 | 32000 | 0.013 | 0.742 | 32000 | 0.047 | 0.428 |

| Nodes | Random | | | Small world | | | Scale-free | | |
|---|---|---|---|---|---|---|---|---|---|
| | Edges | $ND$ | $N_dD$ | Edges | $ND$ | $N_dD$ | Edges | $ND$ | $N_dD$ |
| | 35200 | 0.013 | 0.785 | 35200 | 0.01 | 0.789 | 35200 | 0.037 | 0.502 |
| | 38400 | 0.012 | 0.856 | 38400 | 0.01 | 0.856 | 38400 | 0.027 | 0.569 |
| | 41600 | 0.01 | 0.928 | 41600 | 0.007 | 0.93 | 41600 | 0.023 | 0.642 |
| | 44850 | 0.003 | 1 | 44850 | 0.003 | 1 | 44850 | 0.017 | 0.742 |
| 400 | 40000 | 0.02 | 0.501 | 40000 | 0.023 | 0.501 | 40000 | 0.045 | 0.301 |
| | 44000 | 0.015 | 0.551 | 44000 | 0.018 | 0.551 | 44000 | 0.038 | 0.351 |
| | 48000 | 0.015 | 0.602 | 48000 | 0.015 | 0.602 | 48000 | 0.033 | 0.401 |
| | 52000 | 0.015 | 0.652 | 52000 | 0.015 | 0.652 | 52000 | 0.03 | 0.451 |
| | 60000 | 0.013 | 0.752 | 60000 | 0.013 | 0.752 | 60000 | 0.028 | 0.501 |
| | 64000 | 0.01 | 0.802 | 64000 | 0.01 | 0.802 | 64000 | 0.023 | 0.675 |
| | 68000 | 0.01 | 0.852 | 68000 | 0.008 | 0.852 | 68000 | 0.01 | 0.802 |
| | 72000 | 0.008 | 0.902 | 72000 | 0.005 | 0.902 | 72000 | 0.008 | 0.852 |
| | 76000 | 0.005 | 0.952 | 76000 | 0.005 | 0.952 | 76000 | 0.005 | 0.902 |
| | 98000 | 0.003 | 1 | 98000 | 0.003 | 1 | 98000 | 0.005 | 0.952 |
| 500 | 72000 | 0.018 | 0.577 | 72000 | 0.018 | 0.569 | 72000 | 0.024 | 0.404 |
| | 76800 | 0.014 | 0.616 | 76800 | 0.014 | 0.613 | 76800 | 0.022 | 0.436 |
| | 81600 | 0.012 | 0.654 | 81600 | 0.012 | 0.653 | 81600 | 0.016 | 0.5 |
| | 86400 | 0.01 | 0.693 | 86400 | 0.01 | 0.693 | 86400 | 0.016 | 0.681 |
| | 91200 | 0.01 | 0.731 | 91200 | 0.01 | 0.729 | 91200 | 0.016 | 0.721 |
| | 96000 | 0.01 | 0.77 | 96000 | 0.01 | 0.754 | 96000 | 0.018 | 0.771 |
| | 100800 | 0.008 | 0.808 | 100800 | 0.008 | 0.806 | 100800 | 0.014 | 0.816 |
| | 105200 | 0.006 | 0.843 | 105200 | 0.006 | 0.842 | 105200 | 0.012 | 0.842 |
| | 110000 | 0.006 | 0.882 | 110000 | 0.006 | 0.882 | 110000 | 0.01 | 0.882 |
| | 124750 | 0.002 | 1 | 124750 | 0.002 | 1 | 124750 | 0.01 | 0.898 |

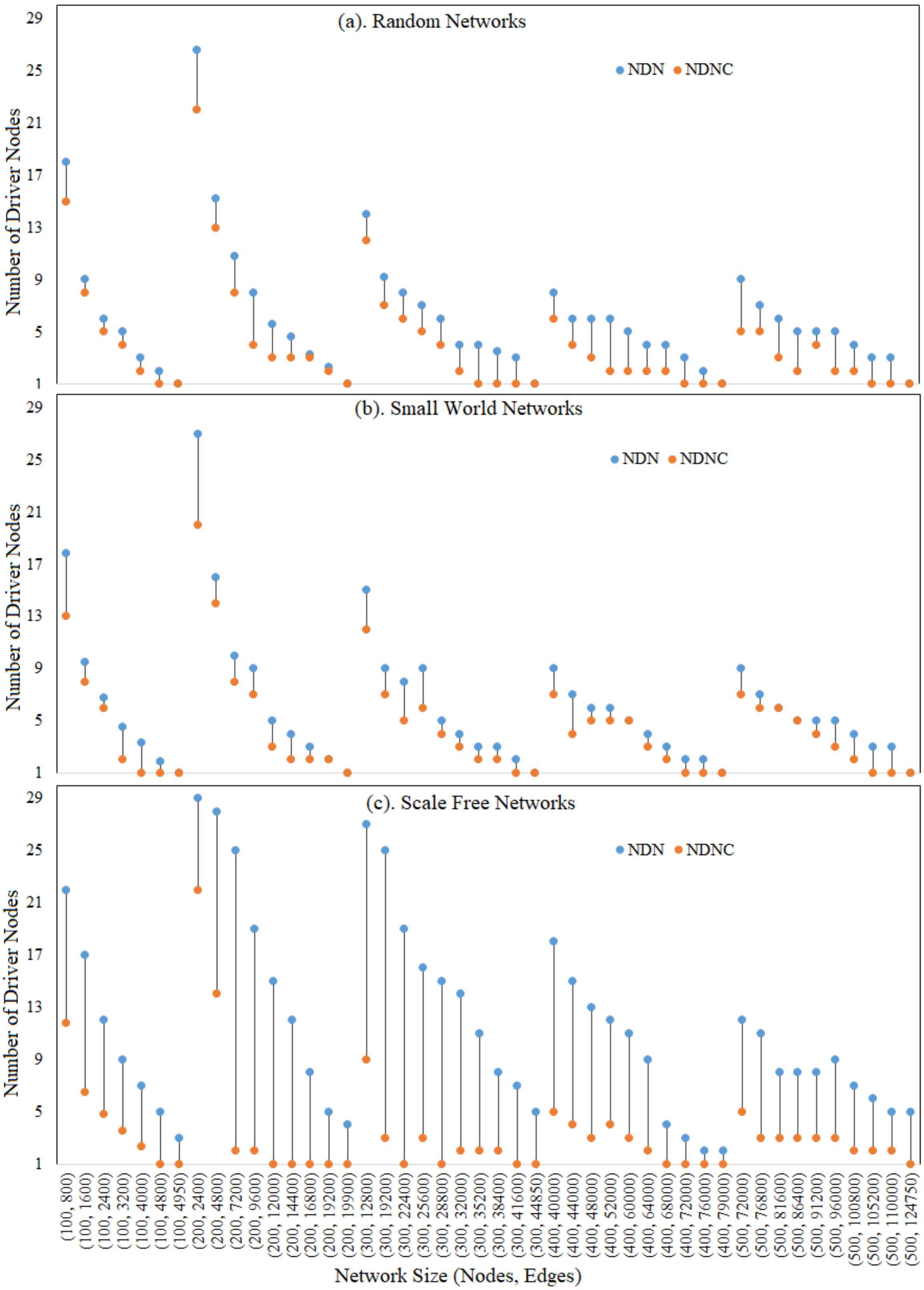


Figure 4.6 : Difference between Number of Driver Nodes in Overall Network Verses Number of Driver Nodes in Communities found in (a) Random Networks, (b) Small World Networks and (c) Scale Free Networks

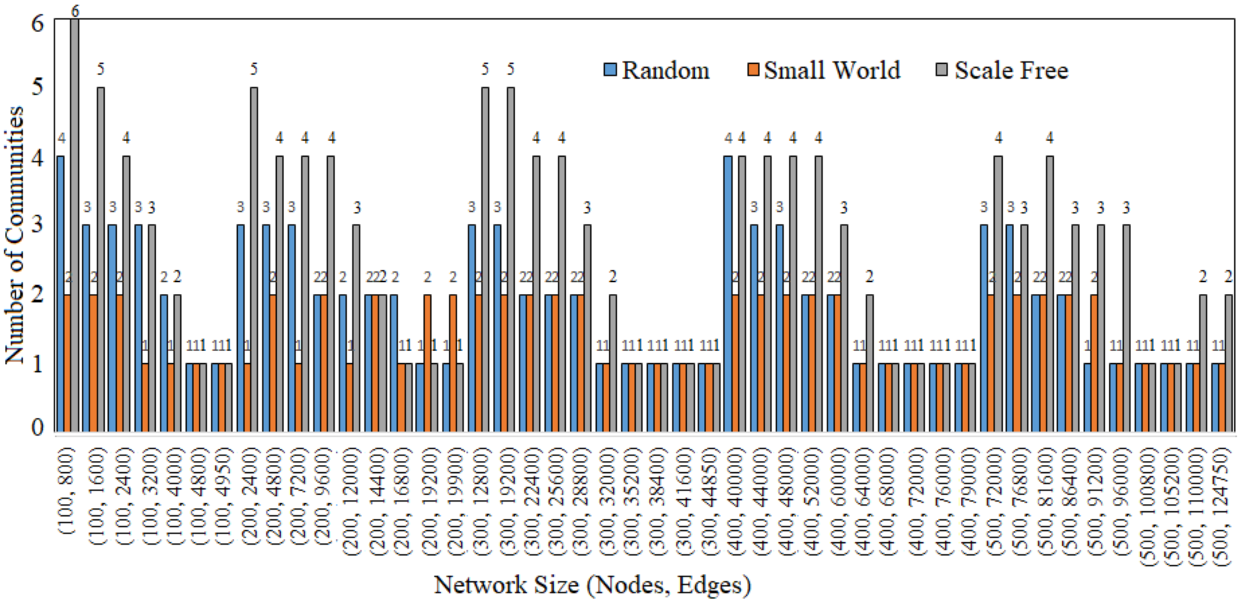


Figure 4.7 : Number of Communities in Random, Small World and Scale Free Networks

## 4.3 Results and Analysis

This section explains the results and analysis from the experiments that have been carried out in this research and experimental study. From Figure 1.2, we can see that Experiment Exp2 has been proposed to resolve the Research Objective RO4. This section is divided in two subsection, i.e., results from synthetic networks and results from real networks.

### 4.3.1 Results from Synthetic Networks

In this section, we analyse the results obtained from the conducted experiments. Below are the comparisons, that have been carried out to answer the research question. Some network structure measures related to network structure measures of random, small world and scale free networks are given in Table 4.1 along with number of driver nodes density ($N_dD$). The table has been adapted from our previous work presented in [182]. Following are the important results and their analysis.

#### 4.3.1.1 Community Density and Number of Driver Nodes in Communities in Random, Small World and Scale Free Networks

Figure 4.5, we correlated local structure measures such as community density with number of driver nodes within those communities. It is evident that as community densities start to approach 1, so are driver nodes. We can see that as community densities are higher, the number of drive nodes are low and vice versa. This result answers the first research question and also strengthens the results from previous research, where we structurally correlated the global measures i.e., network density with number of driver nodes [182]. It means that, networks that have denser communities have naturally less number of driver nodes. Also, Figure 4.5 shows that communities in the network may or may not have the different number of driver nodes.

#### 4.3.1.2 Difference Between Number of Driver Nodes in Networks (NDN) and Number of Driver Nodes in Communities (NDNC) in Random, Small World and Scale Free Networks

Firstly, from Table 4.1 it is clear that we are able to minimise the number of driver nodes (NDN) in the overall network, as we are able to increase the number of edges in all three generated networks. By increasing the number of edges, we automatically increase the density of the network. We can also see that there is not much difference between the density of number of driver nodes $N_dD$ in the networks, as all networks were able to minimise the number of driver nodes to 1 with increased edges and density. More details of these results are provided in [182].From Figure 4.6(a), 4.6(b) and 4.6(c), we can see the difference between NDN (number of driver nodes) and NDNC (number of driver nodes detected in communities) in random, small world and scale free networks. We can see a big difference between the plots of scale free networks in comparison to random and small world networks. From the figure, the conclusions from [182] again strengthens that, as the density tend to increase i.e., number of edges increase in the same node size network, number of driver nodes tend to decrease within the network as well as within the communities of those networks. For example, in a scale free network of (nodes = 400 and edges = 79000), only 1 driver node is required within the communities of that network. It is because network itself is really very dense, and communities are naturally denser than network itself by definition. So, that means, less number of driver nodes are required to spread influence in the overall network. Secondly, we know that, structurally random, small world and scale free networks are quite different from each other. We applied the correlation analysis on our generated network which indicates that, random, small world and scale free networks are quite different from each other. They behaved differently when identifying driver nodes from communities. This can be seen from Figure 4.6. Thirdly, Table 4.2 a huge difference between the number of driver nodes within the whole network as compared to within the communities of those networks. The table shows a heatmap of the difference between NDN and NDNC (Diff.). We can see from the map that the most difference is found in scale free networks, see Figure 4.6(c) and least difference is found in small world networks, see Figure 4.6(b) while in random networks the difference lines seems to be in the middle of what has been presented in small world networks and scale free networks . Which means, that, behaviour of random networks is changed from that of small world and scale free networks. We figure that, most real world networks have scale free properties, that is why difference is larger in the social networks as can be seen from Table 4.3. This inference strengthens our observations from the experiments. Lastly, Table 4.7 shows the number of communities in all generated networks. We can see that there is more variation in number of communities in scale free networks as compared to random and small world networks despite the same network sizes. As scale free networks are analysed to be closer in structure to real networks, they can have more communities in networks as compared to their small world and scale free counterparts.

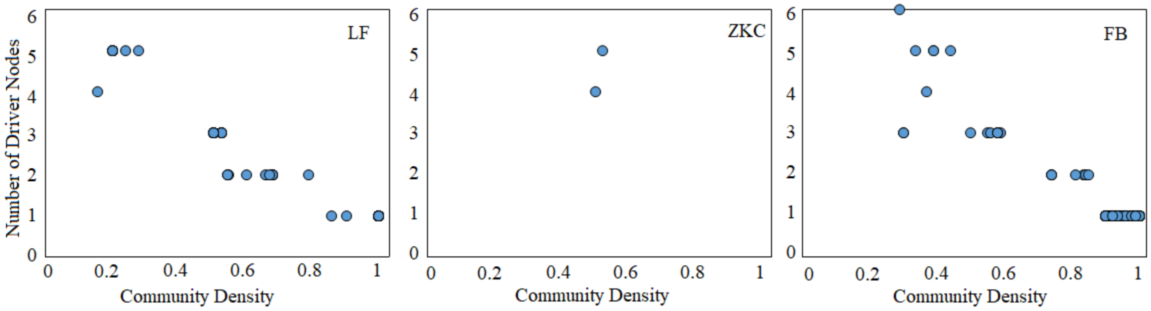


Figure 4.8 : Number of Driver Nodes in the Communities of LF, Z and FB networks verses their Community Densities

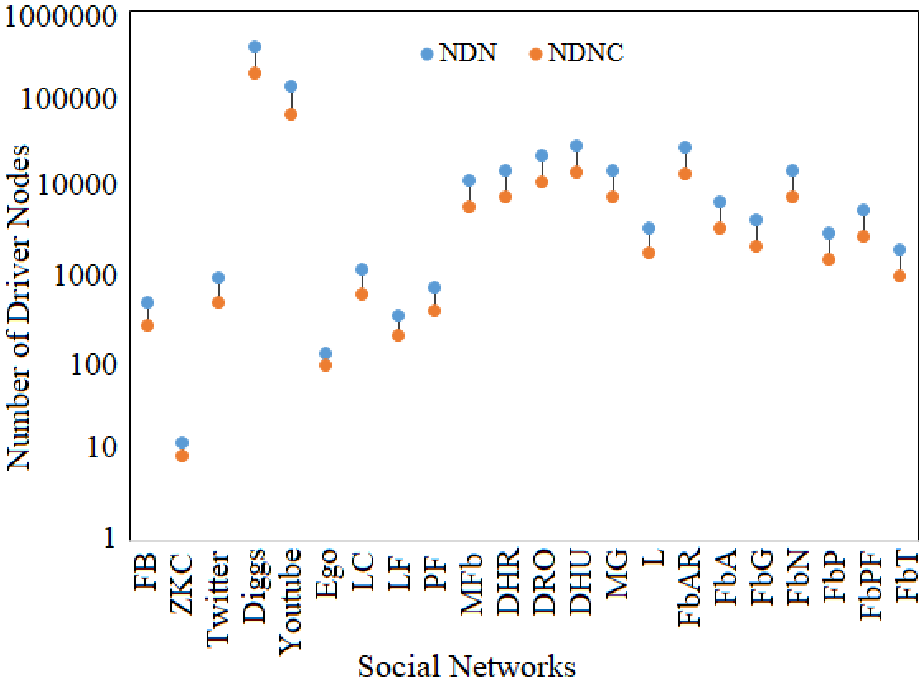


Figure 4.9 : Difference between Total Number of Driver Nodes (NDN) and Number of Driver Nodes in Communities (NDNC) in Social Networks

There are many different indicators to evaluate the importance of a node within a community. For example, node degree, betweenness centrality, closeness centrality etc. However, these measures only tell us different topological features about the node [92]. However, a driver node, when identified based upon these measures, becomes a node with the most number of connections and can be used to propagate control/influence within a community. Also, it has been noted earlier that communities with higher densities have lesser number of driver nodes. That is because that node(s) becomes the node with the most number of connections within that community. Therefore, our future research hypothesis in Chapter 6, suggests to spread influence through the driver nodes within communities of the networks.

### 4.3.2 Results from Social Networks

In this section, results and analysis from social networks are discussed in detail.

#### 4.3.2.1 Community Density and Number of Driver Nodes in Communities in Social Networks

From Figure 4.8 it is evident that communities, by definition, have high densities. This confirms our results from previous study that the denser the network (or community, as we have shown here), the smaller the number of driver nodes [182]. This is a strong reassurance showing that network structure has a strong influence on number of driver nodes. The same figures clearly show that, when the density approaches 1, the number of driver nodes decrease.

#### 4.3.2.2 Difference Between Number of Driver Nodes in Networks (NDN) and Number of Driver Nodes in Communities (NDNC) in Social Networks

We calculated the difference between total number of driver nodes in the whole network (NDN) versus number of driver nodes in the communities of the networks

(NDNC). The difference of both (Diff.) indicates that in all the social networks, number of driver nodes decrease when they are identified within communities. It strongly indicates that, the divide and conquer approach works for the networks. Also, it is easier to apply the process of identifying driver nodes within a smaller size community rather than a huge network with bigger size. By looking at Table 4.3, we can clearly see that, in large size networks for example, Diggs (nodes=1,924,000, edges=3,298,475) and Youtube (nodes=1,134,891, edges=2,987,625), NDN set reduces substantially in size. For Diggs, NDN drops from 481,000 to 198,967, and for Youtube, NDN drops from 283,722 to68,235. Even for small networks like ZKC (nodes=34, edges=78), LF (nodes=874, edges=1309) and PF (nodes=1858, edges=12534) results remain consistent. That means, irrespective the size of the network, when we detect communities and then identify driver nodes within those communities, the driver nodes set reduces to a great extent. We observed that in communities, densities are relatively higher naturally, hence the number of driver nodes in communities is less than the number of driver nodes in their corresponding overall networks.

We can see an overall picture for all the social networks from Figure 4.9, where the plot shows difference between NDN and NDNC values. Since we do not have overlapping communities due to the nature of the algorithm, we have at least one driver node within each community.

## 4.4 Discussion and Conclusion

In this chapter, we focused on Research Challenge RC3, Research Question RQ2 and Research Objective RO3. The main objective was to find out, the correlations between local network structural measures and number of driver nodes. One of the key findings in [182], was that the global network structural measures (i.e., community densities) do correlate with the number of driver nodes found in those communities. From our previous work, we found out that the denser the network, the smaller the number of driver nodes, meaning and those network structures are easier to control [182]. Through this study, we answer the research questions stated in Section 4.1. Communities themselves have pretty large densities as compared to the overall network. So, connecting from the previous results, it seems only plausible that, within those communities, we will find a minimum number of driver nodes with more potential to control the community and by controlling those communities, ultimately the overall network.

Our main contributions in this research work are given below:

1. The study of finding relationships between local structure measures of the network and the number of driver nodes. This has not been explored before. From this research, we contribute that local structure measures such as community densities correlated with the decrease or increase in number of driver nodes. It is easier to control the communities with higher densities because these communities require less number of driver nodes.

2. By detecting driver nodes within communities, we potentially decrease the total number of driver nodes. Hence, it is recommended to break the network down in communities to conquer the problem of identifying a minimum set of driver nodes. We can clearly see from Figures 4.6 and 4.9, that there is a difference in number of driver nodes when identified within communities as compared to when identified in overall networks. This result can help the researchers with the problem of identifying an optimal set of driver nodes.

3. MDS method to detect driver nodes is a very expensive process in very large networks, specially real social networks [147]. By dividing the networks into communities, we make the process of identifying driver nodes comparatively less time-consuming. By presenting this idea, we open another dimension to minimise the driver nodes set, which can still remain effective in controlling the overall network. Global as well as local structural measures of the networks can play an important role to figure out an efficient way to determine the potential

driver nodes set that can control a social network.

Furthermore, many more analyses can be done with different other kinds of networks with varying new structural measures to see the potential correlations.

A comparative analysis in [226], reveals that the GN algorithm might not be suitable for large networks because of its high computation time. But, it has never been actually tested with large scale networks. However, the implementation simplicity of the algorithm has an advantage over the more complex algorithms, hence can be employed to detect communities, even in large scale networks.There is however high computation costs attached in obtaining number of communities using GN algorithm.

Table 4.2 : Difference between Number of Driver Nodes (NDN) in the Whole Network and the Number of Driver Nodes within the Communities (NDNC) of the Networks, i.e., random (R), small-world (SW) and scale-free (SF) (Diff.). Nodes and Edges of the networks are also presented.

| Nodes | Edges | Diff. | | | | | | Edges | Nodes |
|---|---|---|---|---|---|---|---|---|---|
| | | R | SW | SF | R | SW | SF | | |
| 100 | 800 | 3 | 5 | 9 | 5 | 7 | 9 | 2400 | 200 |
| | 1600 | 1 | 2 | 9 | 2 | 2 | 10 | 4800 | |
| | 2400 | 1 | 1 | 6 | 3 | 2 | 17 | 7200 | |
| | 3200 | 1 | 3 | 7 | 4 | 2 | 12 | 9600 | |
| | 4000 | 1 | 2 | 6 | 3 | 2 | 12 | 12000 | |
| | 4800 | 1 | 1 | 4 | 2 | 2 | 10 | 14400 | |
| | 4950 | 0 | 0 | 2 | 0 | 1 | 6 | 16800 | |
| 300 | 12800 | 2 | 3 | 15 | 0 | 0 | 3 | 19200 | |
| | 19200 | 2 | 2 | 18 | 0 | 0 | 3 | 19900 | |
| | 22400 | 2 | 3 | 14 | 2 | 2 | 11 | 40000 | 400 |
| | 25600 | 2 | 3 | 10 | 2 | 3 | 11 | 44000 | |
| | 28800 | 2 | 1 | 11 | 3 | 1 | 8 | 48000 | |
| | 32000 | 2 | 1 | 11 | 4 | 1 | 7 | 52000 | |
| | 35200 | 3 | 1 | 9 | 3 | 0 | 6 | 60000 | |
| | 38400 | 3 | 1 | 6 | 2 | 1 | 6 | 64000 | |
| | 41600 | 2 | 1 | 6 | 2 | 1 | 2 | 68000 | |
| | 44850 | 0 | 0 | 4 | 2 | 1 | 2 | 72000 | |
| 500 | 72000 | 4 | 2 | 5 | 1 | 1 | 1 | 76000 | |
| | 76800 | 2 | 1 | 5 | 0 | 0 | 1 | 798000 | |
| | 81600 | 3 | 0 | 2 | 2 | 2 | 5 | 100800 | 500 |
| | 86400 | 3 | 0 | 3 | 2 | 2 | 5 | 105200 | |
| | 91200 | 1 | 1 | 4 | 2 | 2 | 4 | 110000 | |
| | 96000 | 3 | 2 | 2 | 0 | 0 | 4 | 124750 | |

Table 4.3 : Social Networks and their Global and Local Structure Measures such as Nodes (N), Edges (E), Density (D), Number of Driver Nodes (NDN), Number of Communities (C), Number of Driver Nodes in Communities (NDNC) and Difference between Number of Driver Nodes in Networks and Number of Driver Nodes in Communities (Diff.)

| Networks | N | E | D | C | NDN | NDNC | Diff. |
|---|---|---|---|---|---|---|---|
| FB [138] | 4039 | 88234 | 0.01 | 180 | 499 | 270 | 229 |
| Z [233] | 34 | 78 | 0.14 | 2 | 13 | 9 | 4 |
| Twitter [138] | 23371 | 32832 | 0.00012 | 350 | 939 | 489 | 450 |
| Diggs [84] | 1924000 | 3298475 | 0.000002 | 156432 | 398004 | 199037 | 198967 |
| Youtube [225] | 1134891 | 2987625 | 0.000004 | 54983 | 136520 | 68285 | 68235 |
| Ego [138] | 23629 | 39195 | 0.00014 | 75 | 132 | 96 | 36 |
| LC [112] | 4658 | 33116 | 0.003 | 517 | 1178 | 620 | 558 |
| LF [70] | 874 | 1309 | 0.0034 | 97 | 347 | 209 | 138 |
| PF [173] | 1858 | 12534 | 0.0073 | 206 | 745 | 398 | 347 |
| MFb [175] | 22470 | 171002 | 0.00067 | 2643 | 11955 | 6011 | 5944 |
| DHR [176] | 54574 | 498202 | 0.0003 | 6420 | 15678 | 7877 | 7801 |
| DRO [176] | 41774 | 125826 | 0.0001 | 4914 | 22680 | 11372 | 11308 |
| DHU [176] | 47539 | 222887 | 0.0002 | 5592 | 29479 | 14755 | 14724 |
| MG [175] | 37700 | 289003 | 0.0004 | 4435 | 15507 | 7775 | 7732 |
| L [177] | 7624 | 27806 | 0.0009 | 759 | 3518 | 1795 | 1723 |
| FbAR [176] | 50516 | 819306 | 0.0006 | 5943 | 28670 | 14372 | 14298 |
| FbA [176] | 13867 | 86858 | 0.0009 | 1383 | 6827 | 3449 | 3378 |
| FbG [176] | 7058 | 89455 | 0.0036 | 784 | 4245 | 2160 | 2085 |
| FbN [176] | 27918 | 206259 | 0.0005 | 3284 | 15558 | 7813 | 7745 |
| FbP [176] | 5909 | 41729 | 0.0024 | 562 | 2995 | 1530 | 1465 |
| FbPF [176] | 11566 | 67114 | 0.001 | 1051 | 5510 | 2792 | 2718 |
| FbT [176] | 3893 | 17262 | 0.0023 | 387 | 1966 | 1011 | 955 |

# Chapter 5

# Influence Models and Driver Nodes

This chapter addresses Research Challenge RC4, which states that, "Drawing comparisons between different driver based seed selection methods with traditional seed selection methods for generated and real social networks." In order to attain RC4, Research Question, RQ3 has been devised which states that,"How efficient and effective are driver-based seed selection methods in comparison to traditional methods?". Research Objective RO4 was defined to execute the experiment 3 (Exp3). RO4 states that, "To implement different traditional seed selection methods and driver-based methods to generate influence in synthetic and real networks". In Figure 5.1 we can see the highlighted tasks that were carried out to fulfil RO4 as part of the whole thesis.

In this chapter, Section 5.1 describes the related work including influence and control in complex networks. Section 5.2 describes the methodology and detailed experiment setup of the experiments being conducted to answer the research questions. Section 5.3 describes the results and their comprehensive analysis. Lastly, the conclusion and future work are discussed in Section 5.4.

## 5.1 Background

Since the beginning of social media, our online activities transformed the way we interact with others and this in turn has changed our social networks. Social media allow us to communicate and interact with others through sending direct messages, sharing opinions and information, as well as commenting on others content. Interactions over social media platforms may play an effective role in quick and worldwide proliferation

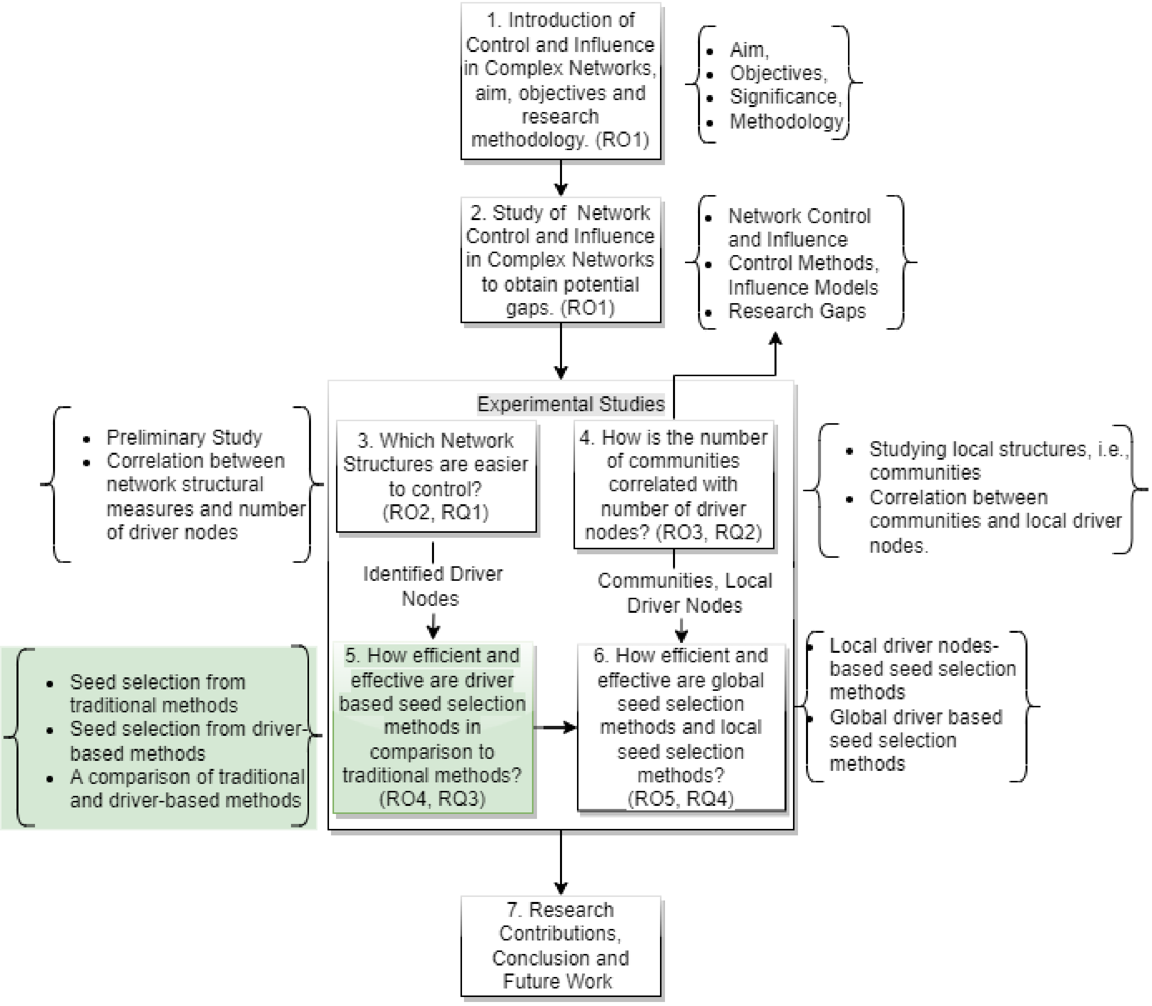

1. Introduction of Control and Influence in Complex Networks, aim, objectives and research methodology. (RO1)
Aim,
Objectives,
Significance,
Methodology
2. Study of Network Control and Influence in Complex Networks to obtain potential gaps. (RO1)
Network Control and Influence
Control Methods, Influence Models
Research Gaps
Experimental Studies
Preliminary Study
Correlation between network structural measures and number of driver nodes
3. Which Network Structures are easier to control? (RO2, RQ1)
4. How is the number of communities correlated with number of driver nodes? (RO3, RQ2)
Studying local structures, i.e., communities
Correlation between communities and local driver nodes.
Identified Driver Nodes
Communities, Local Driver Nodes
Seed selection from traditional methods
Seed selection from driver-based methods
A comparison of traditional and driver-based methods
5. How efficient and effective are driver based seed selection methods in comparison to traditional methods? (RO4, RQ3)
6. How efficient and effective are global seed selection methods and local seed selection methods? (RO5, RQ4)
Local driver nodes-based seed selection methods
Global driver based seed selection methods
7. Research Contributions, Conclusion and Future Work

Figure 5.1 : Research Methodology : Chapter 5

of news and can shape the opinions of users. Although social media proved to be an effective way to influence the public opinion, we know that not all users play the same role in this process. An example of that are ‘influencers’ who are seen as key players in the propagation of the information quickly and effectively [235]. Spread of influence, in particular, has gained a lot of attention in recent years as various research groups and commercial companies try to understand how people’s opinions and decisions can be influenced and potentially changed and to what extent we are receptive of others opinions. How the influence spreads in networks? is a question. This includes physics [26], ecology [50], biology [230] and network science [236].

Many studies focus on how to quantify the influence of nodes in a complex network [71, 132] with the hope that if the most influential nodes are chosen to propagate a given phenomenon, then the spread of this phenomenon will be optimal. One of the avenues to explore, in the search for more effective ways to assess the influence potential of a given node, is to look into the direction of control over complex networks.

There are a few notable works in this regard such as very recently, a recommender system to identify more influential nodes to increase the efficiency of spreading process [203] is one such example. Previously, in another related work, authors proposed a low-complexity heuristic algorithm to build a recommender system to achieve efficient coverage of nodes [107]. We used the control approaches in influence models to achieve the efficient coverage of nodes with a smaller set of influential seed nodes.

There are conceptual similarities between driver nodes in the network control space and seed nodes in the spreading processes, and the goal of this study is to explore the possibility of using driver nodes as seed nodes and proposing and developing new seed selection strategies for spread of influence inspired by driver node concept. Control can be seen as a "stronger" version of influence [215], so our hypothesis is that the influence can spread effectively (affect a larger number of nodes) through driver nodes than when using other, traditional seed selection strategies. We focus on maximising the number of nodes influenced in smaller number of iterations by utilising minimum seed size.

## 5.2 Research Methodology and Experiment Design

We see little work done in the space where techniques for finding driver nodes are used to support seed selection strategies; thus, we explore and address this research gap. Figure 5.2 depicts the research process employed in this study and defines the steps taken from defining research challenges by stating research questions and objectives to describing proposed experiments.

The main focus of RQ3 is to find out if it is feasible to use concepts from the field of network control in the context of influence spread and if so, how it can be done. To answer RQ3, we propose new methods that utilise driver nodes as seeds in influence spreading. First, we decide on the method that will be used to select driver nodes. This can be any of the approaches described before: *maximum matching [89], minimum dominating set [147], control profiles [178] and preferential matching [238].* To keep the consistency across all the experiments, we use Minimum Dominating Set approach. It is also considered a benchmark approach to identify driver nodes across various kinds of networks, . The next decision point is to select a technique to rank the identified driver nodes so an ordered list can be used in the seed selection process. This can be done using various approaches, including methods presented in the Chapter 2, Section 2.5, for both ranking the driver nodes or ranking seed nodes. We have used centrality measures to rank driver nodes, please see Section 5.2 for details. We also propose a new method based upon the centrality measures to rank the driver nodes, with the top nodes from the ranking being used as seeds. The answer to Research Question RQ3 and work that is done to answer it can be seen as an initial step to develop a framework where we use network control concepts (driver nodes selection and ranking) in the context of influence spread in networks. To further answer RQ3, we need to measure the effectiveness of that approach. When we run experiments, we assess to what extent the methods identified in this study are able to improve the influence spread over the traditional seed selection approaches. The evaluation of seeding strategies on different networks is done on the basis of how much influence the seed nodes are going to spread *a.k.a coverage of influence*, with respect to the spreading time [93]. So, the seed selection method which results in shorter spreading process time and/or has bigger coverage *influence over the network* can be regarded as more effective than the others. **"Control meets influence"** is an idea where we first identify driver nodes in a given network and apply them as seeds in influence model to see the result of influence spread in the network. Figure 5.2 presents the main stages of the research setup i.e. inputs, process, and outputs. We utilise traditional seed selection strategies such as random (R), degree centrality (D), betweenness centrality (B) and Kempe seed selection (K). The major outputs will include a comparison by network and a comparison by seed selection method. In network comparison, we observe the percentage of nodes influenced in each network. In method comparison, we compare the performance of seed selection strategies. The performance is measured on the basis of total number of iterations it takes for each method to obtain the highest influence in each network based upon a certain seed set size. Detailed experiment set-up is presented in the next section.

To bring concepts from control field into influence field, we propose to use Minimum Dominating Set (MDS) to identify driver nodes and then rank those driver nodes using the same ranking methods as in case of seed selection strategies. Additionally – Driver Degree Closeness Betweenness (DDCB) that first identifies MDS set and then rank the driver nodes on the basis of their average degree, betweenness and closeness centralities. A percentage of ranked driver nodes are then used as seeds. Linear Threshold model is used to simulate the spread of influence over both randomly generated networks (using random, small–world and scale–free models) and real social networks. A description of both randomly generated and real networks is included in the next subsection.

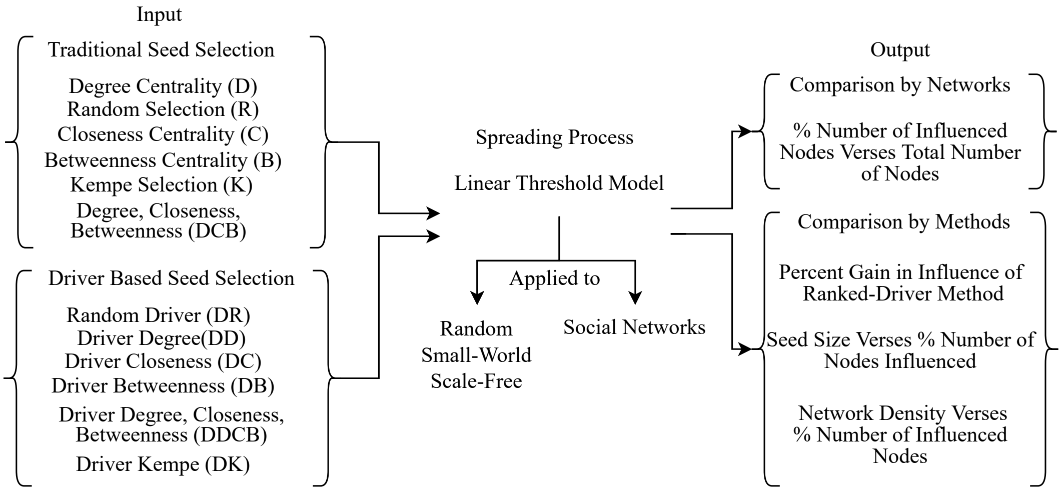

Input
Traditional Seed Selection
Degree Centrality (D)
Random Selection (R)
Closeness Centrality (C)
Betweenness Centrality (B)
Kempe Selection (K)
Degree, Closeness, Betweenness (DCB)
Driver Based Seed Selection
Random Driver (DR)
Driver Degree(DD)
Driver Closeness (DC)
Driver Betweenness (DB)
Driver Degree, Closeness, Betweenness (DDCB)
Driver Kempe (DK)
Spreading Process
Linear Threshold Model
Applied to
Random
Small-World
Scale-Free
Social Networks
Output
Comparison by Networks
% Number of Influenced Nodes Verses Total Number of Nodes
Comparison by Methods
Percent Gain in Influence of Ranked-Driver Method
Seed Size Verses % Number of Nodes Influenced
Network Density Verses % Number of Influenced Nodes

Figure 5.2 : Control meets influence. The general concept for evaluating usefulness of driver nodes selection methods in seed selection for influence spread problem.

### 5.2.1 Networks

This section includes the tables and figures describing the networks used during our evaluation.

- Figure 5.3 shows the Random, Small-World and Scale-Free networks sizes (i.e., number of nodes) plotted against their densities. We generated ten network profiles of each Random, Small-World and Scale-Free networks of size ranging from number of nodes equal to 100, 200, 300, 400 and 500. We kept the connections such that to make sure that networks are always connected. In total, 750 networks were generated. This includes the networks starting from smaller densities such as (0.05) to the highest density (1). Density is increasing for every network type when the nodes are from 100–500, due to increase in number of edges. Sometimes it took ten iterations to generate networks with varying sizes and densities, with the goal of achieving the highest density, i.e., 1.
- Table 3.2 includes information about twenty-two real social networks used dur-

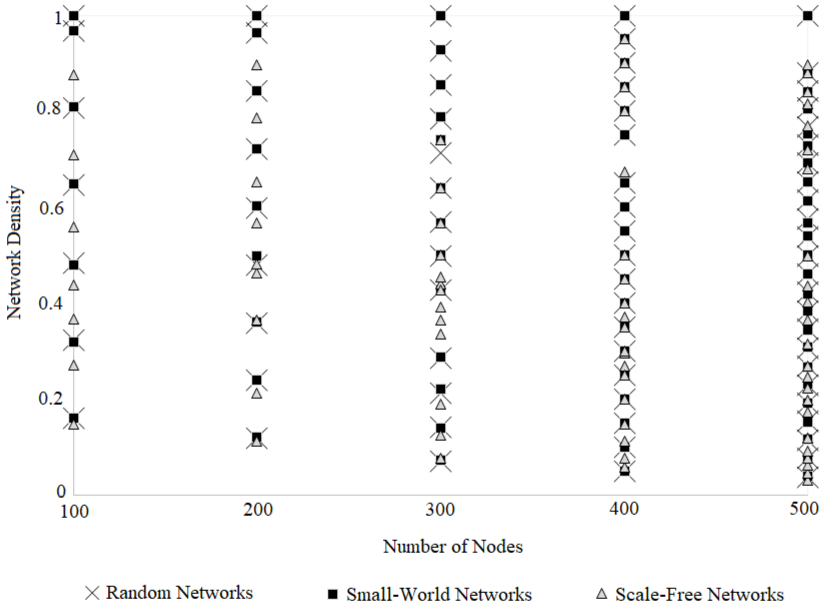


Figure 5.3 : Size vs. density in Random, Small-World and Scale-Free networks

ing our experiment and their network structure measures, i.e., the number of nodes, number of edges, and corresponding network density. The networks were downloaded from Stanford Large Network Dataset Collection repository [116].

- Figure 3.4 shows the densities of social networks with their number of nodes in a logarithmic scale chart.

### 5.2.2 Experiments

To be able to answer the research questions, we designed the following experiments.

1. Building Network Profiles: To enable systematic analysis of both traditional and driver–based seed selection strategies, the experiments are conducted on synthetic networks, including Random (R), Small-World (SW) and Scale-Free (SF) network models as well as real social networks. For comparison purposes, we generated all the networks with same number of nodes and edges. In order to

achieve that, we used the method previously applied in [204]. We generated 750 networks with 100, 200, 300, 400 and 500 each for Random, Small-World and Scale-Free. More details of the networks is also given in our previous work [182].

For social networks we used twenty two social networks available in SNAP library [116].

2. Traditional Seed Selection: The methods that are being used in this section are, random seed selection (R), degree seed selection (D), closeness centrality seed selection (C), betweenness centrality seed selection (B), Kempe seed selection (K) and additionally the degree, closeness and betweenness centrality seed selection (DCB) where a top percentage of number of nodes is selected as seeds based upon their average of higher degree, closeness and betweenness centrality values. R, D, C, B and Kempe are most commonly used seed selection methods.

3. Driver Seed Selection: One of the contributions of this study are novel driver-based seed selection strategies. It is a methodological advancement where network control concepts are used in the influence modelling space. The driver nodes are identified by using the Minimum Dominating Set (MDS) method. MDS has been calculated to show the number of driver nodes in the network using MDS method as described in [147]. Although MDS is a NP-hard problem, reduction rules are a great way to obtain a reduced minimum dominating set, a.k.a Branch and Reduce Algorithm [218]. By applying this algorithm, we get a reduced minimum dominating set a.k.a driver nodes. Ranking of driver nodes has previously shown that influential nodes often have higher centrality values [30]. We focus on ranking driver nodes using various centrality values. Additionally we use Kempe approach to rank driver nodes and as a baseline we use random approach. All proposed ranking strategies are outlined below. In Driver–Random Seed Selection (DR), we select nodes at random from all the driver nodes and they create seed set. In Driver–Degree Seed Selection (DD),

we rank the driver nodes in by their degree values and the top ranked nodes become seed nodes. In Driver–Closeness Seed Selection (DC), we rank the driver nodes by their closeness centrality values and the top ranked nodes become seed nodes. In Driver–Betweenness Seed Selection (DB), we rank the driver nodes by their betweenness centrality values and the top ranked nodes become seed nodes. In Driver–Degree–Closeness–Betweenness Seed Selection (DDCB), we rank the driver nodes by averaging the sum of each node's degree, closeness and betweenness centrality values. In Driver–Kempe Seed Selection (DK), we rank the driver nodes based upon their potential to influence the network. The node which is able to spread influence to a larger number of nodes is ranked higher and in each iteration every new node is evaluated together with those already in the seed set. At the end, the nodes which are able to spread influence to maximum number of nodes make it to the final seed set.

4. Simulating Influence Spread by using LTM: We use both traditional and driver-based seed selection methods to obtain the seed sets and we use those sets as the input to the LTM model to investigate how the spread progresses. In LTM, each agent activates if the number of its active neighbours is bigger or equal than its current activation threshold. We used Bootstrap Percolation to determine the thresholds for LTM. Bootstrap percolation is a process of spread of "activation" on a given network with a given number of initially active nodes. At each step those vertices which have not been active but have at least $\geq 2$ active neighbours become active as well [94].

## 5.3 Results and Analysis

Eleven seed selection methods (i.e. Random, Degree, Closeness, Betweenness, Degree–Closeness–Betweenness, Kempe, Driver–Random, Driver–Degree, Driver–Closeness, Driver–Betweenness, Driver–Kempe and Driver–Degree–Closeness–Betweenness) have been tested on synthetic and real world networks. LTM was used to ensure consistency of the results across the board. The findings are discussed from the perspective of (i) synthetic and (ii) real networks.

### 5.3.1 Results from Synthetic Networks

Results from synthetic networks include impact of network density on percentage of nodes influenced and a comparison of all seed selection methods with respect to number of nodes influenced within given time budget and time needed to achieve 100% coverage.

Table 5.1 : The average percentage of influenced nodes in all generated networks, with seed sizes up to 50% after 20 iterations. N - number of nodes, Net. - Network model, D - network density, L - low and M - medium density.

| Seed Size | | | 1% | 10% | 20% | 30% | 40% | 50% |
|---|---|---|---|---|---|---|---|---|
| N | Net. | D | Percentage of influenced nodes | | | | | |
| 100 | R | L | 20.0 | 33.0 | 49.0 | 67.0 | 78.0 | 89.0 |
| | | M | 59.0 | 88.0 | 97.0 | 100.0 | 100.0 | 100.0 |
| | SW | L | 30.0 | 49.0 | 58.0 | 64.0 | 82.0 | 93.0 |
| | | M | 57.0 | 92.0 | 99.0 | 100.0 | 100 | 100.0 |
| | SF | L | 18.0 | 51.0 | 67.0 | 79.0 | 87.0 | 100.0 |
| | | M | 62.0 | 100.0 | 100.0 | 100.0 | 100.0 | 100.0 |
| 200 | R | L | 16.0 | 30.0 | 42.0 | 53.0 | 78.0 | 87.0 |
| | | M | 67.0 | 79.0 | 82.0 | 98.0 | 100.0 | 100.0 |
| | SW | L | 16.5 | 40.0 | 54.0 | 67.0 | 78.0 | 89.0 |
| | | M | 67.0 | 91.0 | 100.0 | 100.0 | 100.0 | 100.0 |
| | SF | L | 15.5 | 40.0 | 56.0 | 67.0 | 79.0 | 98.0 |
| | | M | 88.0 | 100.0 | 100.0 | 100.0 | 100.0 | 100.0 |
| 300 | R | L | 25.0 | 46.0 | 57.0 | 68.0 | 78.0 | 82.0 |
| | | M | 64.0 | 88.0 | 97.0 | 100.0 | 100.0 | 100.0 |
| | SW | L | 26.0 | 46.0 | 67.0 | 73.0 | 84.0 | 92.0 |
| | | M | 60.0 | 84.0 | 98.0 | 100.0 | 100.0 | 100.0 |
| | SF | L | 24.6 | 49.0 | 57.0 | 66.0 | 79.0 | 87.0 |
| | | M | 87.0 | 96.0 | 100.0 | 100.0 | 100.0 | 100.0 |
| 400 | R | L | 21.0 | 44.0 | 55.0 | 61.0 | 87.0 | 92.0 |
| | | M | 77.0 | 100.0 | 82.0 | 97.0 | 100.0 | 100.0 |
| | SW | L | 22.0 | 45.0 | 56.0 | 68.0 | 72.0 | 88.0 |
| | | M | 71.0 | 82.0 | 98.0 | 100.0 | 100.0 | 100.0 |

| Seed Size | | | 1% | 10% | 20% | 30% | 40% | 50% |
|---|---|---|---|---|---|---|---|---|
| N | Net. | D | Percentage of influenced nodes | | | | | |
| | SF | L | 28.0 | 46.0 | 64.0 | 71.0 | 87.0 | 99.0 |
| | | M | 76.0 | 100.0 | 100.0 | 100.0 | 100.0 | 100.0 |
| 500 | R | L | 25.0 | 42.0 | 57.0 | 61.0 | 78.0 | 83.0 |
| | | M | 77.0 | 88.0 | 98.0 | 100.0 | 100.0 | 100.0 |
| | SW | L | 26.0 | 35.0 | 65.0 | 76.0 | 78.0 | 87.0 |
| | | M | 60.0 | 72.0 | 78.0 | 89.0 | 92.0 | 100.0 |
| | SF | L | 29.3 | 39.0 | 56.0 | 63.0 | 78.0 | 87.0 |
| | | M | 64.0 | 75.0 | 100.0 | 100.0 | 100.0 | 100.0 |

Table 5.2 : Percentage of influenced nodes when the density is low and seed size is 1%. The highest percentage is bolded.

| Nodes | | 100 | 200 | 300 | 400 | 500 |
|---|---|---|---|---|---|---|
| Networks | Methods | Percentage of influenced nodes | | | | |
| R | R | 20.0 | 16.0 | 25.0 | 21.7 | 14.6 |
| | D | 23 | 17.5 | 25.7 | 22.2 | 15.2 |
| | C | 25.0 | 17.5 | 23.7 | 22.7 | 15.8 |
| | B | 27.0 | 18.5 | 24.7 | 23.2 | 16.2 |
| | DCB | 27.0 | 18.5 | 25.67 | 23.25 | 16.2 |
| | K | 28.0 | 19.0 | 26.0 | 23.5 | 16.4 |
| | DR | 23 | 17.5 | 25.7 | 22.2 | 15.2 |
| | DD | 29.0 | 20.5 | 25.7 | 34.75 | 37.8 |
| | DC | 31.0 | 21.5 | 28.7 | 35.2 | 38.2 |
| | DB | 32.0 | 22.0 | 29.0 | 35.5 | 38.4 |
| | DK | 33 | 22.5 | 29.3 | 35.7 | 38.6 |
| | **DDCB** | **37.0** | **24.5** | **30.7** | **36.7** | **39.4** |
| SW | R | 27.0 | 16.5 | 26.0 | 22.0 | 15.2 |
| | D | 27.0 | 17.5 | 25.7 | 22.25 | 15.6 |
| | C | 27.0 | 18.5 | 24.3 | 23.25 | 16.2 |
| | B | 28.0 | 19.5 | 24.3 | 23.0 | 16.2 |
| | DCB | 28.0 | 19.5 | 26.3 | 23.0 | 16.2 |
| | K | 30.0 | 20.5 | 27.0 | 23.5 | 16.6 |
| | DR | 30.0 | 18.5 | 24.7 | 23.0 | 16.2 |
| | DD | 35.0 | 20.0 | 25.33 | 37.8 | 39.0 |
| | DC | 41.0 | 21.0 | 28.0 | 24.2 | 17.2 |
| | DB | 43.0 | 22.0 | 28.7 | 24.7 | 17.6 |
| | DK | 42.0 | 21.5 | 28.3 | 24.5 | 17.4 |

| Nodes | | 100 | 200 | 300 | 400 | 500 |
|---|---|---|---|---|---|---|
| Networks | Methods | Percentage of influenced nodes | | | | |
| | **DDCB** | **46.0** | **23.0** | **29.3** | **39.4** | **40.0** |
| SF | R | 18.0 | 15.5 | 24.7 | 28.0 | 13.6 |
| | D | 20.0 | 16.5 | 25.3 | 28.7 | 14.2 |
| | C | 21.0 | 17.0 | 23.3 | 21.75 | 14.6 |
| | B | 22.0 | 17.5 | 23.7 | 22.2 | 15.4 |
| | DCB | 22.0 | 17.0 | 25.7 | 21.0 | 14.0 |
| | K | 23.0 | 18.0 | 26.3 | 22.5 | 15.6 |
| | DR | 22.0 | 17.5 | 26.0 | 29.5 | 25.2 |
| | DD | 30.0 | 21.5 | 26.3 | 24.2 | 17.0 |
| | DC | 32.0 | 22.5 | 29.3 | 24.7 | 17.4 |
| | DB | 33.0 | 23.0 | 29.7 | 25.0 | 17.6 |
| | DK | 31.0 | 22.0 | 29.0 | 24.5 | 17.2 |
| | **DDCB** | **37.0** | **25.0** | **31.0** | **33.25** | **28.2** |

Table 5.3 : Percentage of influenced nodes when the density is medium and seed size is 1%. The highest percentage is bolded.

| Nodes | | 100 | 200 | 300 | 400 | 500 |
|---|---|---|---|---|---|---|
| Networks | Methods | Percentage of influenced nodes | | | | |
| R | R | 59.0 | 66.0 | 56.3 | 47.2 | 50.6 |
| | D | 61.0 | 67.0 | 56.7 | 48.0 | 51.4 |
| | C | 61.0 | 67.5 | 57.0 | 48.2 | 52.0 |
| | B | 63.0 | 68.5 | 57.7 | 48.7 | 52.4 |
| | DCB | 63.0 | 68.0 | 58.0 | 49.0 | 52.0 |
| | K | 64.0 | 69.0 | 58.0 | 49.0 | 52.6 |
| | DR | 62.0 | 68.0 | 57.7 | 48.5 | 52.2 |
| | DD | 70.0 | 71.5 | 61.3 | 95.2 | 75.8 |
| | DC | 72.0 | 72.5 | 62.0 | 95.7 | 76.2 |
| | DB | 73.0 | 73.0 | 62.3 | 96.0 | 76.4 |
| | DK | 74.0 | 73.5 | 62.7 | 96.2 | 76.6 |
| | **DDCB** | **78.0** | **75.5** | **64.0** | **97.2** | **77.4** |
| SW | R | 57.0 | 65.5 | 56.3 | 47.2 | 52.0 |
| | D | 62.0 | 67.0 | 56.7 | 48.0 | 52.4 |
| | C | 63.0 | 68.5 | 57.7 | 48.7 | 52.4 |
| | B | 64.0 | 69.0 | 57.7 | 48.5 | 53.0 |
| | DCB | 64.0 | 69.0 | 58.0 | 48.0 | 53.0 |
| | K | 66.0 | 70.0 | 58.3 | 49.0 | 53.4 |
| | DR | 62.0 | 68.0 | 57.7 | 48.5 | 53.0 |
| | DD | 65.0 | 69.5 | 58.7 | 49.2 | 59.0 |
| | DC | 67.0 | 70.5 | 59.3 | 49.7 | 59.4 |
| | DB | 69.0 | 71.5 | 60.0 | 50.2 | 59.8 |
| | DK | 68.0 | 71.0 | 59.7 | 50.0 | 59.6 |

| Nodes | | 100 | 200 | 300 | 400 | 500 |
|---|---|---|---|---|---|---|
| Networks | Methods | Percentage of influenced nodes | | | | |
| | **DDCB** | **71.0** | **72.5** | **60.7** | **50.7** | **60.2** |
| SF | R | 59.0 | 65.5 | 55.7 | 46.5 | 60.2 |
| | D | 61.0 | 66.5 | 56.3 | 47.2 | 60.4 |
| | C | 62.0 | 67.0 | 56.7 | 47.5 | 60.6 |
| | B | 63.0 | 67.5 | 57.0 | 47.7 | 61.0 |
| | DCB | 63.0 | 67.5 | 56.0 | 47.5 | 60.0 |
| | K | 64.0 | 68.0 | 57.3 | 48.0 | 61.2 |
| | DR | 63.0 | 67.5 | 57.0 | 47.7 | 61.0 |
| | DD | 71.0 | 71.5 | 59.7 | 49.7 | 62.6 |
| | DC | 73 | 72.5 | 60.3 | 50.2 | 63.0 |
| | DB | 74.0 | 73.0 | 60.7 | 50.5 | 63.2 |
| | DK | 72.0 | 72.0 | 60.0 | 50.0 | 62.8 |
| | **DDCB** | **78.0** | **75.0** | **62.0** | **51.5** | **64.0** |

#### 5.3.1.1 Impact of network density on number of nodes influenced

We analyzed the influence in networks through the lens of network density. Figure 5.3 shows size and density for all the generated networks. We can see that we have networks from lowest to highest densities for each of the given sizes i.e.*number of nodes.*Table 5.1 shows a comparison of the low and medium density networks with nodes from 100-500 with seed sizes 1%, 10%, 20%, 30%, 40% and 50% in all Random, Small-World and Scale-Free networks. The range of low densities is from 0.12 to 0.16, and the range for medium densities is from 0.60 to 0.64. We considered 20 iterations for LTM as a benchmark, because most of the networks reach 100% influence within 20 iterations for networks with medium densities. For complete graphs (density equal to 1), it is observed that all networks reached the maximum influence in less than 20 iterations, regardless of the seed selection method.

Table 5.2 shows the percentage of influenced nodes in Random, Small-World and Scale-Free networks when the density is low. We can see that for the lowest tested density, i.e., 0.1 the range of level of influence for different network sizes is between $15.5\% - 25\%$ when the seed size is 1%. For medium density networks it lies between 57% and 79% for all the methods when the seed size is 1% (Table 5.3) with $N = 100$. That means, more iterations are required with 1% seed size to achieve a 100% influence in all network types and sizes i.e. from 100 to 500 nodes.

Table 5.3 shows the percentage of nodes influenced for seed selection methods in R, SW, and SF networks when density is medium and seed size is 1%. For complete graphs all nodes in all networks are influenced. As we increase the seed set size we can see that the percentage of nodes influenced are started to increase regardless of density of the network. But for maximum seed size 50%, the 100% influence is reached in medium as well as low densities networks within 20 iterations. A few observations from the experiment suggest that firstly, it takes more iterations when the seed size is smaller i.e. 1% of the total number of nodes. Secondly, to achieve more influence when the densities are higher regardless of the network topology. Lastly, for complete graphs we need less iterations regardless the type of network or seed selection method used. From the above observations, we can say that density and seed set size play an important role in determining the efficiency of influence spread in terms of the percentage of influenced nodes. However, as we can see in the further results, seed selection method also matters.

#### 5.3.1.2 Percentage of nodes influenced

We present a comparison of seed selection methods in the form of the percentage of *gain* in influence of DDCB over Random, Small-World and Scale-Free networks in Table 5.4. The percentage of gain of influence (in percentage points) for the DDCB method is calculated by subtracting from the percentage of nodes influenced by DDCB the percentage of nodes influenced when using other methods (i.e. R, D, C, B, DCB, K, DR, DD, DC, DB, DK, DDCB). The overall gain for Random, Small-World and Scale-Free networks for a particular number of nodes is calculated by taking the average gain over all generated networks of one size (*i.e* $N = 300$). We compute the gain using the level of influence after 20 iterations, as this is the earliest point that the DDCB (and hence any) seed selection method reaches 100% influence. We noted that, as expected, the percentage points gain of DDCB method is the highest over Random seed selection method (*i.e.* 10.51%) in Random networks. However, Table 5.4 shows that the DDCB method outperforms all evaluated seed selection methods.

Table 5.4 : Average percentage points gain of DDCB method over R, D, C, B, DCB, K, DR, DD, DC, DB, and DK methods in Random (R), Small-World (SW) and Scale-Free (SF) networks. Seed set size is expressed as % of the total number of nodes. Each cell in the table shows the average and standard deviation of percentage gain of DDCB over other methods.

| | | Seed Size | | | | | |
|---|---|---|---|---|---|---|---|
| N | M | 1% | 10% | 20% | 30% | 40% | 50% |
| R | R | 1.51±.11 | 8.2±.06 | 6.18±.03 | 4.87±.03 | 3.31±.03 | 1.49±.02 |
| | D | 9.74±.11 | 7.56±.06 | 5.64±.03 | 4.41±.03 | 2.97±.03 | 1.26±.02 |
| | C | 9.64±.10 | 7.74±.05 | 5.72±.03 | 4.49±.03 | 2.91±.02 | 1.21±.02 |
| | B | 9.08±.05 | 7.46±.02 | 5.43±.01 | 4.02±.01 | 2.64±.01 | 1.02±.01 |
| | DCB | 9.08±.01 | 7.74±.07 | 5.44±.04 | 4.49±.03 | 2.64±.06 | 1.21±.03 |
| | K | 8.79±.55 | 7.18±.53 | 5.15±.21 | 3.79±.45 | 2.46±.03 | .92±.02 |
| | DR | 8.95±.11 | 7.00±.06 | 5.11±.03 | 3.95±.03 | 2.64±.03 | 1.02±.02 |
| | DD | 2.26±.06 | 2.26±.06 | 2.05±.01 | 2.07±.01 | 1.07±.01 | .77±.01 |
| | DC | 1.69±.08 | 1.69±.09 | 1.54±.02 | 1.61±.04 | .77±.05 | .61±.51 |
| | DB | 1.41±.03 | 1.41±.04 | 1.28±.05 | 1.38±.09 | .61±.45 | .54±.35 |
| | DK | 1.12±.29 | 1.13±.45 | 1.02±.35 | 1.15±.85 | .46±.65 | .46±.07 |
| | R | 5.33±.05 | 3.97±.02 | 4.02±.02 | 4.59±.04 | 2.21±.03 | 1.08±.02 |

| | | Seed Size | | | | | |
|---|---|---|---|---|---|---|---|
| N | M | 1% | 10% | 20% | 30% | 40% | 50% |
| SW | D | 4.7±.05 | 3.41±.02 | 3.49±.02 | 4.13±.04 | 1.9±.02 | .95±.01 |
| | C | 4.08±.02 | 3.92±.01 | 5.41±.01 | 4.82±.02 | 3.26±.03 | .91±.01 |
| | B | 3.95±.03 | 3.11±.01 | 3.33±.01 | 3.81±.03 | 1.15±.01 | .71±.01 |
| | DCB | 3.95±.03 | 3.41±.03 | 5.41±.01 | 3.8±.02 | 1.15±.05 | .90±.04 |
| | K | 3.4±.04 | 2.51±.01 | 2.8±.02 | 3.30±.03 | .90±.05 | .54±.06 |
| | DR | 4.26±.05 | 2.85±.02 | 2.95±.02 | 3.80±.04 | 1.15±.02 | .70±.01 |
| | DD | 1.69±.06 | 1.71±.06 | 1.31±.01 | 2.05±.01 | 1.02±.01 | 0.54±.02 |
| | DC | 1.12±.91 | 1.12±.34 | 1.02±.04 | 1.64±.36 | 0.82±.34 | 0.38±.57 |
| | DB | .57±.21 | 0.57±.04 | 0.51±.11 | 1.23±.34 | .61±.28 | .23±.43 |
| | DK | 0.85±.04 | 0.85±.38 | 0.77±.08 | 1.43±.91 | 0.72±.54 | 0.31±.59 |
| SF | R | 6.97±.06 | 5.97±.03 | 5.13±.02 | 4.36±.03 | 3.43±.04 | 1.61±.02 |
| | D | 6.36±.06 | 5.41±.03 | 4.56±.02 | 3.90±.03 | 3.15±.04 | .14±.02 |
| | C | 5.79±.04 | 5.13±.03 | 4.00±.02 | 3.44±.02 | 2.99±.03 | 1.38±.02 |
| | B | 5.49±.06 | 4.85±.02 | 4.00±.01 | 3.46±.02 | 2.87±.03 | 1.15±.02 |
| | DCB | 5.79±.91 | 5.13±.65 | 4.00±.37 | 3.46±.91 | 0.87±.11 | .91±.31 |
| | K | 5.21±.55 | 4.57±.87 | 3.72±.84 | 3.23±.04 | 2.72±.06 | 1.05±.13 |
| | DR | 6.05±.06 | 4.85±.03 | 4.00±.02 | 3.46±.02 | 2.87±.03 | 1.15±.02 |
| | DD | 2.15±.06 | 2.15±.06 | 2.26±.05 | 1.79±.04 | .92±.01 | .61±.01 |
| | DC | 1.54±.04 | 1.54±.08 | 1.70±.93 | 1.30±.08 | .72±.11 | .50±.26 |
| | DB | 1.22±.33 | 1.23±.54 | 1.41±.09 | 1.02±.06 | .61±.73 | .40±.34 |
| | DK | 1.85±.11 | 1.84±.03 | 1.97±.93 | 1.54±.05 | .82±.17 | .53±.66 |

Additionally, the results in Table 5.2 and Table 5.3 show that the traditional seed selection methods do not perform as well as their ‘sibling’ driver based methods. By ‘sibling’ method, we denote a pair of methods where ranking is done using the same approach, but one is a driver-based method (only driver nodes are ranked) and the other is not (all nodes are ranked). None of the ranking methods incorporates the fact that even the high degree node can be clustered. As a cluster in a network is a set of densely connected nodes that is sparsely connected to other clusters in the network, so it is unnecessary to target all the highest degree nodes that may be in only one or few clusters. The driver nodes are selected in a way to enable control over the whole network and not only its parts, so they provide better coverage of the network. These observations suggest that if we rank driver nodes based on centrality measures when they are to be used as seeds, the influence spread process produces better results than the benchmark methods such as randomly generated seeds or most commonly used degree based methods. Table 5.4 shows the results when 1%, 10%, 20%, 30%, 40% and 50% of driver nodes are selected as seeds. With the increase of the seed set size, the difference in the percentage of nodes influenced by different methods becomes negligible. Thus, the advantage of the DDCB method is more critical when we have low budget for seed selection, and we can target only small number of nodes which, arguably, is a case in most situations.

#### 5.3.1.3 Critical Difference Diagram for Generated Networks

Figure 5.4 shows a comparison between all the methods used to generate influence over all of the generated networks. The critical difference diagram shows whether the results (expressed as % of nodes influenced) for various methods are significantly different from each other. The confidence level used is $\alpha = 0.05$. Critical difference diagrams use the Wilcoxon-Holm method [86] to determine the statistical significance of the results. The lower the rank (further to the right) the better performance of a model under the particular masking rate compared to the others on average. Horizontal line segments group together methods with ranks that are not significantly different in terms of spreading influence. The percentage of number of nodes influenced is calculated for each method for seed sizes 1%, 10%, 20%, 30%, 40% and 50%. The diagrams show that, in generated networks, driver based methods are

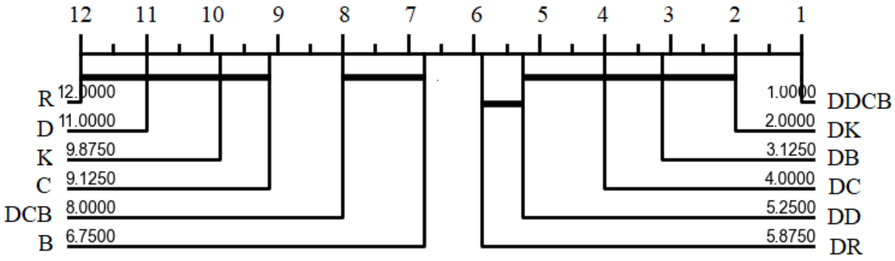


Figure 5.4 : Critical difference diagram for generated networks.

critically different from traditional seed selection methods in terms of percentage of influenced nodes. Moreover, the DDCB method consistently outperforms other methods and ranks as no. 1 across the board. Other driver–based methods, with exception of random approach (DR), although they outperform traditional methods, are not statistically significantly different between each other. When looking at the traditional methods there is more statistically significant difference between centrality based methods, e.g., degree and closeness centrality ranking methods are worse than betweenness centrality.

This indicates that the key to good seed selection method is rather the fact that we first select driver nodes and rank those than the ranking method itself. Selecting driver nodes enables to effectively reduce number of nodes to be ranked and in the same time ensures that selected nodes are good influencers as they can control the underlying structure.

### 5.3.2 Results from Real Social Networks

This section contains the results from real social networks.

#### 5.3.2.1 Percentage of influence gain by DDCB method over other methods in the social networks

Table 5.5 shows the percentage of influence gained over all other methods by DDCB method after 100 iterations. We can see that over Random method, percentage

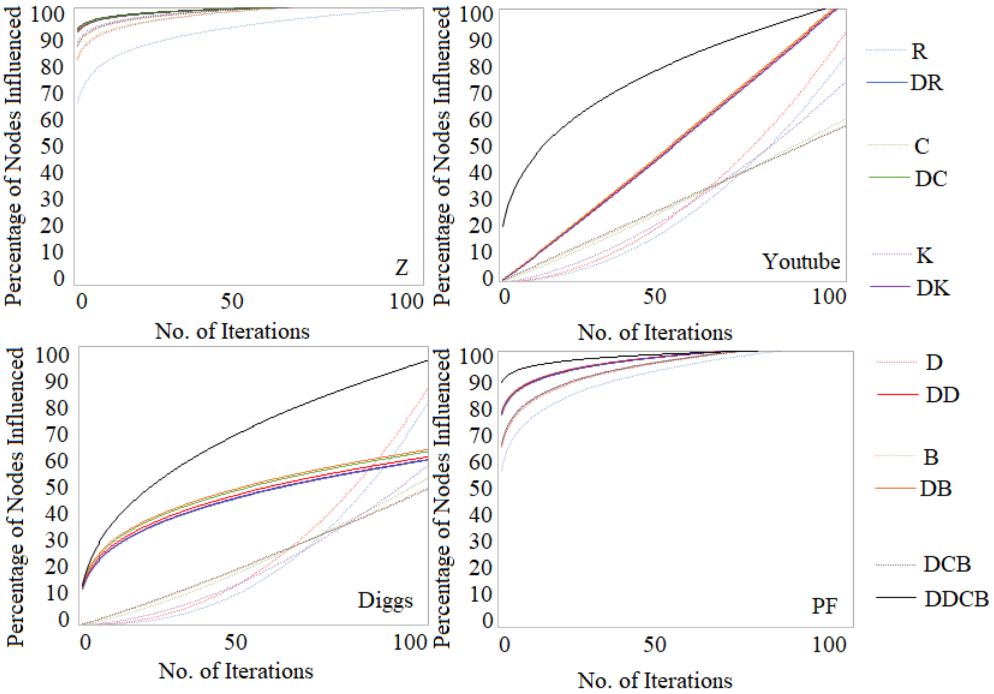


Figure 5.5 : The percentage of influenced nodes in each iteration (the trend-lines for all simulation cases) for different seed selection methods and for Z, Youtube, Diggs, and PF networks. Seed set size is 25%.

gain is the highest. It can be seen, that DDCB method gained more influence over traditional methods as compared to driver based methods. This means that driver based methods, regardless the applied ranking method, do increase the spread of influence over a network. We observe an increase of number of influenced nodes as the process progresses. This leads to the reduction of gain achieved by DDCB over other methods and eventually, when 100% of influenced nodes is achieved, there is no gain.

#### 5.3.2.2 Percentage of nodes influenced

We can see from the results that driver–based methods of seed selection (DR, DD, DC, DB, DK, DDCB) are able to achieve full influence i.e. *when all nodes are activated*

Table 5.5 : Gain of DDCB over R, D, C, B, DCB, K, DR, DD, DC, DB and DK in real social networks. Seed set size is 25% of the all nodes. Each cell in the table shows the percentage gain of DDCB over other methods after 100 iterations.

| Network | R | D | C | B | DCB | K | DR | DD | DC | DB | DK |
|---|---:|---:|---:|---:|---:|---:|---:|---:|---:|---:|---:|
| FB | 24.68 | 23.03 | 23.94 | 24.94 | 24.15 | 23.59 | 20.59 | 20.59 | 21.19 | 21.28 | 20.14 |
| ZKC | 8.18 | 2.00 | 1.82 | 1.09 | 0.95 | 0.73 | 0.18 | 0.18 | 0.27 | 0.00 | 0.00 |
| Twitter | 33.81 | 25.83 | 25.8 | 25.78 | 19.16 | 25.77 | 22.81 | 22.8 | 22.74 | 22.06 | 20.22 |
| Diggs | 38.49 | 36.05 | 35.76 | 35.47 | 35.37 | 38.21 | 19.11 | 17.89 | 16.67 | 15.53 | 18.85 |
| Youtube | 39.00 | 33.02 | 32.12 | 31.79 | 31.59 | 32.92 | 2.51 | 1.71 | 0.91 | 0.11 | 2.45 |
| Ego | 20.83 | 13.34 | 13.33 | 13.33 | 16.15 | 20.81 | 8.64 | 8.62 | 8.14 | 8.05 | 7.89 |
| LC | 30.84 | 24.62 | 24.61 | 24.61 | 24.52 | 30.81 | 21.4 | 21.23 | 20.98 | 20.65 | 21.06 |
| LF | 15.29 | 8.62 | 8.56 | 8.34 | 8.25 | 8.33 | 7.38 | 7.35 | 7.20 | 6.86 | 8.11 |
| PF | 7.62 | 4.66 | 4.43 | 4.21 | 4.13 | 4.25 | 1.94 | 1.78 | 1.64 | 1.60 | 1.71 |
| MFb | 21.44 | 20.16 | 20.11 | 20.11 | 20.10 | 20.11 | 14.07 | 13.80 | 19.70 | 19.43 | 13.80 |
| DHR | 36.77 | 33.42 | 32.21 | 31.00 | 30.90 | 33.20 | 5.78 | 5.26 | 4.73 | 4.21 | 5.01 |
| DRO | 38.43 | 33.74 | 33.42 | 33.22 | 33.12 | 33.45 | 12.50 | 12.40 | 12.13 | 11.94 | 33.18 |
| DHU | 42.4 | 33.77 | 33.52 | 33.33 | 33.13 | 37.85 | 25.35 | 25.02 | 24.84 | 24.61 | 24.33 |
| MG | 27.54 | 25.43 | 25.07 | 25.25 | 25.07 | 25.34 | 15.14 | 14.49 | 14.05 | 9.35 | 13.86 |
| L | 24.55 | 23.25 | 23.04 | 22.82 | 22.79 | 22.81 | 17.34 | 17.11 | 16.75 | 16.71 | 16.70 |
| FbAR | 37.97 | 30.40 | 30.18 | 29.95 | 29.93 | 30.30 | 28.43 | 28.14 | 27.85 | 27.56 | 28.28 |
| FbA | 45.29 | 30.45 | 30.05 | 29.55 | 39.87 | 44.89 | 31.28 | 30.83 | 30.28 | 29.46 | 31.01 |
| FbG | 19.95 | 18.22 | 17.93 | 17.71 | 18.18 | 18.20 | 12.97 | 12.75 | 12.39 | 12.13 | 12.68 |
| FbN | 28.82 | 21.03 | 21.00 | 20.95 | 20.96 | 21.01 | 11.85 | 11.64 | 11.18 | 11.10 | 11.40 |
| FbP | 26.73 | 20.90 | 20.76 | 20.40 | 20.87 | 20.89 | 14.89 | 14.47 | 14.15 | 13.90 | 14.31 |
| FbPF | 34.61 | 30.21 | 29.85 | 29.57 | 29.39 | 29.48 | 25.30 | 25.12 | 24.85 | 24.21 | 25.21 |
| FbT | 23.93 | 20.84 | 23.70 | 23.29 | 16.73 | 16.77 | 18.37 | 17.46 | 12.71 | 12.36 | 16.63 |

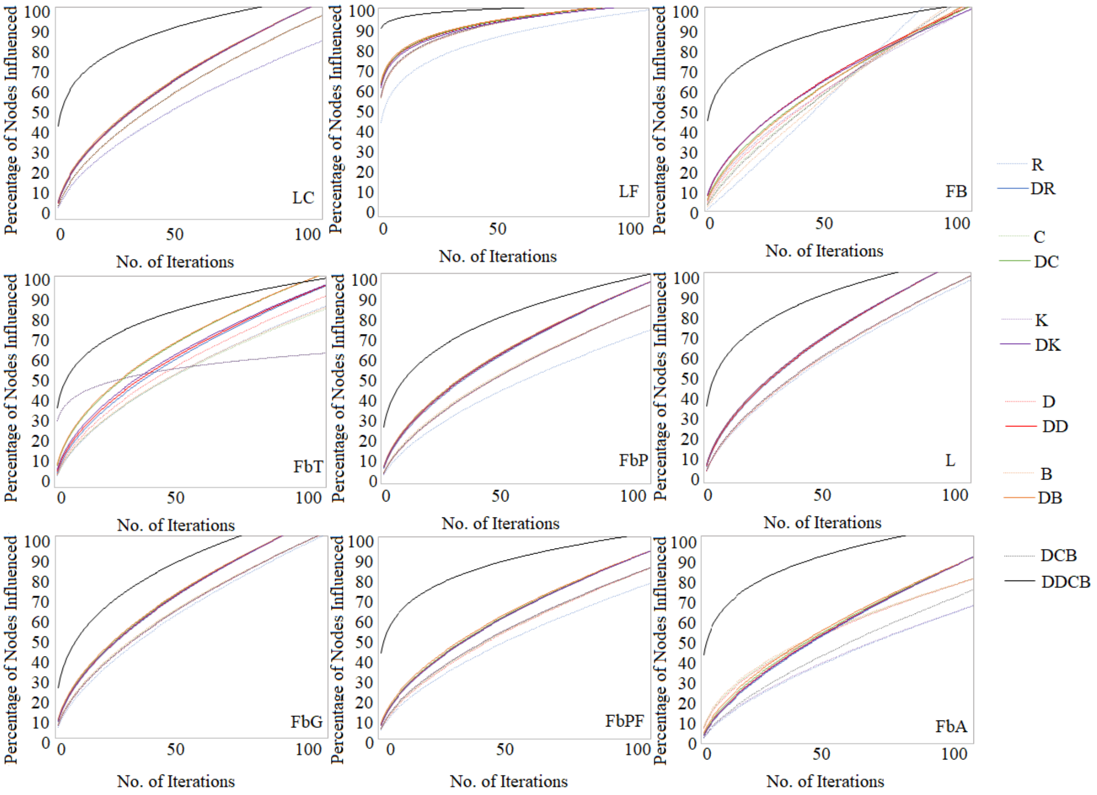


Figure 5.6 : The percentage of influenced nodes in each iteration (the trend-lines for all simulation cases) for different seed selection methods and for LF, FB, FbT, LC, FbP, FbG, L, FbPF, and FbA networks. Seed set size is 25%.

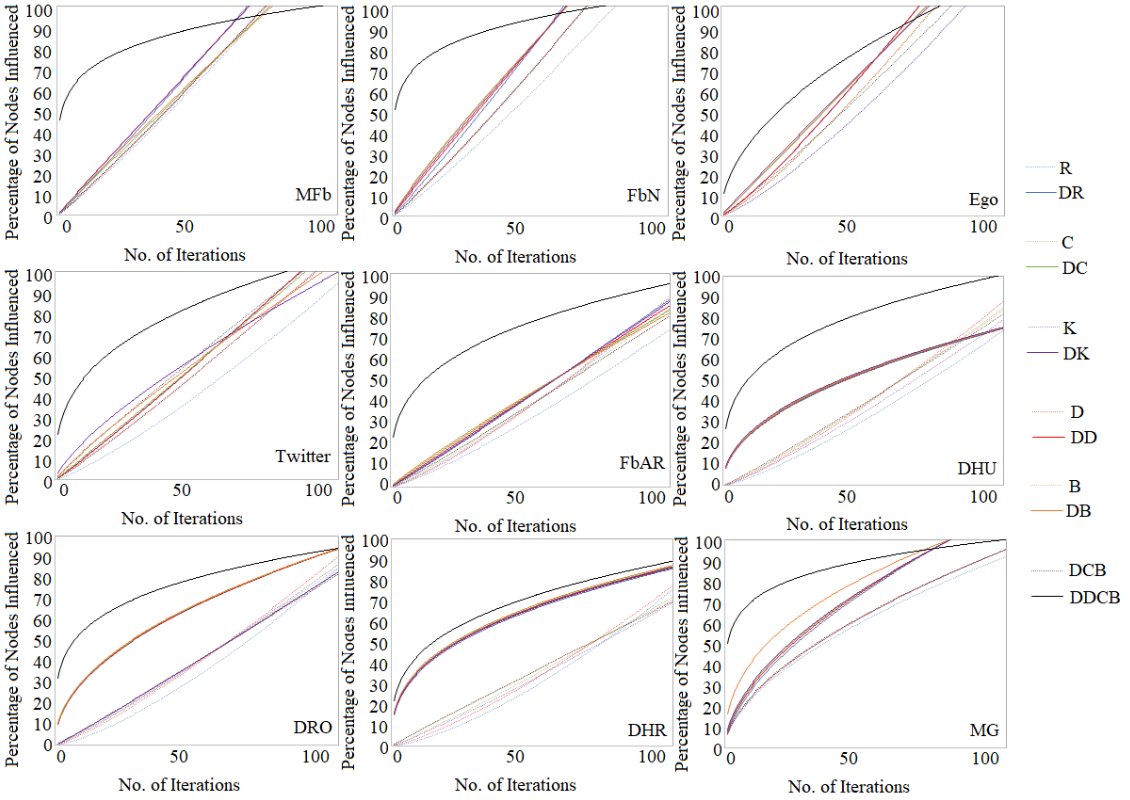


Figure 5.7 : The percentage of influenced nodes in each iteration (the trend-lines for all simulation cases) for different seed selection methods and for MFb, FbN, Ego, Twitter, FbAR, DHU, DRO, DHR, and MG networks. Seed set size is 25%.

in less iterations than traditional methods (R, D, C, B, K). With 25% nodes selected as seed nodes for DDCB method, in all the networks, all nodes are activated in 100 or fewer iterations. For R and D methods, for Twitter, FbN, DRO, DHU, FbA, DHR, YouTube and Diggs networks it took more than 100 iterations to achieve 100% influence. Additionally, for R method and Twitter, DRO, FbA, DHU, DHR, YouTube, and Diggs, it took more than 100 iterations to reach 100% influence. The networks are distributed in three figures, which are divided on the basis of their network densities. We can also notice in Figs. 5.6 and 5.7 that for such networks as YouTube, Diggs, DRO, DHU, FbT and FbN the spreading dynamic (trend-lines) based on R and D seed selection start slowly (below $y = x$ line) but then pick up as the number of iterations increases. This means that if we are looking for fast influence spread, in less number of iterations and with low seed size, we cannot just rely on Random or Degree based methods. Spreading dynamic for driver–based methods shows faster influence spread as compared to traditional seed selection methods. That means, driver nodes are influential and then ranking based on centrality measures enabled to extract the most influential ones. It means that driver nodes that were originally used as nodes that can control the network can also be used as seeds and provide faster influence spread.

The DDCB method shows promising results, but this could also be due to the network structural measures, e.g. density of the underlying social network. Thus, we look into the densities of the selected networks. We can see in case of lower density networks, such as Diggs (0.000002), YouTube (0.000004), Twitter (0.00012) and Ego (0.00014), the trend-line for all the methods starts lower than for the networks with high density such as FB (0.01) or ZKC (0.13). It is worth noting that for all networks, for seed size 25%, DDCB method achieved 100% influence in less than 100 iterations, which can be seen from the trendlines in Figures 5.5, 5.6 and 5.7. The point to be highlighted here is: networks with higher densities and smaller size (e.g. ZKC) show a trendline which depicts the quickest possible influence spread as compared

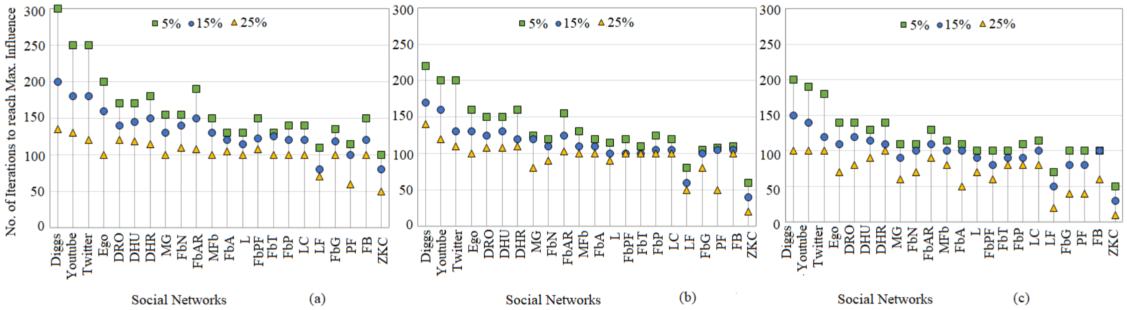


Figure 5.8 : Number of iterations needed to reach 100% of influenced nodes using different seed selection strategies: (a) Traditional Seed Selection, (b) Driver Seed Selection, (c) DDCB Seed Selection. Seed set sizes are 5%, 15% and 25%.

to the rest of the networks. Figures 5.5, 5.6 and 5.7 show the percentage of nodes influenced as a function of number of iterations of spreading process for different seed selection methods and seed set size of 25%. The results show that DDCB obtained momentum from the very start of influence spread in all the networks, despite their size. That means, that if we aim to achieve faster influence, then DDCB is the right choice. That will allow us to decrease our time (the number of iterations) needed to spread influence in the network, and we can achieve faster influence by using smaller seed set size. If we look at the shape of trend-lines from Figures 5.5, 5.6 and 5.7, we see a lift-off in trendlines of DDCB method.

#### 5.3.2.3 Number of iterations needed to influence the network

Fig. 5.8 shows different seed set sizes and the number of iterations each method needs to achieve 100% influence. We sorted the networks in ascending order of their densities to see clearly that the sparser networks needs more iterations to complete the process, irrespective of the seed set size. We worked on achieving the maximum influence by continuing the influence. Overall, the number of iterations reduce in all networks when the seed set size increases from 5% to 15% and to 25%. We can see a drop in the number of iterations as we increase the seed size and also as the density of a network increases. This means that some of the strategies to get 100% of nodes activated are (i) to increase the seed size or (ii) to run the process longer. But this is not always possible, for example due to resource limitation.

In Fig. 5.8, we can see that DDCB method, outperforms both driver based and traditional seed selection methods, and can be used when we want to see more nodes influenced in less time (iterations). We can also see that even driver method, where seeds are ranked in the highest degree, helps propagate influence faster than when using traditional seed selection methods. From the same figure, we can see that Diggs network is up to 300 iterations, when seed size is 5%, to achieve maximum influence when we use traditional seed selection methods. When we increase the seed size to 15% or 25%, we see a sudden drop in number of iterations needed. But still the number of iterations remain higher in Diggs than the rest of the networks. Since Diggs is the lowest density (0.000002) network, we can say our results from simulated networks are relevant here as well. Because, with dense networks, we achieve faster influence spread. We can see a drop in number of iterations for the same network from Fig. 5.8, where we compared different seed sizes for driver based seed selection methods. This result indicates that, even in networks with varying structures, driver based methods outperforms the traditional seed selection methods.

#### 5.3.2.4 Critical Difference Diagram for Social Networks

We see similar results in Figrue 5.9, as we have seen for the generated networks. We found out that all driver–based methods yield statistically better results as compared to their traditional counterparts. The critical difference diagram in Figure 5.9 shows the mean ranks of each method. The lower the rank (further to the right) the better performance of a method compared to the others on average. A line in each diagram indicates that there is no significant difference in performance among the models crossed by that particular line in terms of the Friedman test that compares the ranks of multiple methods. For example, there is a line connecting DB and DC

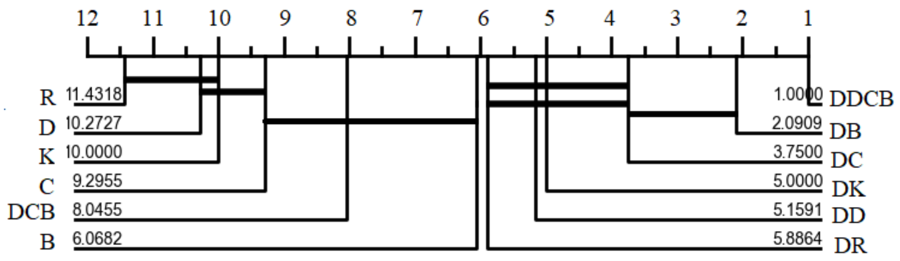


Figure 5.9 : Critical difference diagram for real networks

methods, DC and DR methods, R and K methods, D and C methods, and C and B. Which means, that there is no critical difference between the two methods that are connected in terms of their performances. The diagram also shows, a clear distinction between DDCB method and rest of the methods. DDCB method ranks higher on the right hand side, indicates that it is the most efficient method out of these all. The similarity line connecting the methods, means that the use of these methods in the context of the networks studied is indistinct. This means that, if we identify driver nodes first before selecting them as seeds, it increases their potential to influence more nodes in the network in less iterations as compared to traditional seed selection methods. Not all traditional methods are significantly different from each other. We can see high resemblance in the results of D, K and B methods. If we look at their counterparts methods based on driver nodes DR, DK and DB respectively, the results are significantly improved in them, due to the presence of driver nodes.

#### 5.3.2.5 Time Complexity and Execution Times

From the results, we observe that driver based methods can influence more number of nodes in the examined networks. However, there is time needed to calculate the driver nodes and assemble the ranking. Since our main focus is on calculating the percentage of the nodes influenced, that is why we focused so far on calculating the number of iterations to reach the maximum influence. However, to provide complete analysis, the execution time for all the seed selection methods in the biggest network, i.e., Youtube was recorded and compared.Table 5.6, shows the time it takes to execute the algorithms for the methods.

Table 5.6 : Time Complexity of Calculating Different Measures

| Centrality | Complexity |
|---|---|
| Degree | $O^2$ |
| Closeness | $O(N*E*d)$, where $d$ is the diameter |
| Betweenness | $O(N*M+N*2*logN)$ |
| Driver Nodes | $O(N^2.5)$ |

The most important observation is that when comparing all the methods in Table 5.8, DDCB method takes fewer iterations (i.e. 26, 20 and 16 at 5%, 15% and 25% seed nodes) to complete the influence process, which is to influence all the nodes. Also, all the driver based methods in comparison to traditional methods require almost half the iterations to influence 100% of the nodes of the networks. All the comparisons are done for 5%, 15% and 25% seed set sizes respectively. DR in comparison to R (77, 74 and 70) takes 37, 32 and 28 iterations to influence the nodes in the networks. DD in comparison to D (75, 74 and 60) takes 35, 30 and 28 iterations. DC in comparison to C (69, 63 and 62) takes 39, 35 and 30 iterations. DB in comparison to B (63, 59 and 55) takes 43, 40 and 32 iterations. DK in comparison to K (61, 57 and 49) takes 38, 32 and 25 iterations.

Table 5.7 shows the execution times of all seed selection methods for various sizes of the seed sets i.e. 1%, 5%, 15%, and 25%, when the maximum influence is reached. Driver based methods in comparison to the traditional methods, take less time in overall influence spread process. The calculation of the ranking and driver nodes is time consuming but, overall analysis tells us that, given that the driver based methods reach maximum influence in fewer number of iterations, decreases execution times.

In Table 5.8, execution times in each of the social networks for the driver based

Table 5.7 : Total Execution Times (in hours – Hrs.) of Influence Model using each of the Seed Selection Methods in the Examined Social Networks when 100% Influence is Reached.

| Execution Time (Hrs.): Maximum Influence | | | | |
|---|---|---|---|---|
| Seed Selection Methods | Seed Size | | | |
| | 1% | 5% | 15% | 25% |
| R | 166.95 | 125.67 | 112.45 | 98.32 |
| D | 156.39 | 120.54 | 109.32 | 108.63 |
| C | 150.88 | 119.52 | 109.45 | 95.43 |
| B | 144.75 | 120.75 | 107.64 | 94.32 |
| K | 129.36 | 98.73 | 86.21 | 80.41 |
| DCB | 134 | 82.12 | 76.34 | 73.32 |
| DR | 78.35 | 77.86 | 74.78 | 70.63 |
| DD | 72.52 | 70.64 | 64.13 | 60.52 |
| DC | 68.83 | 64.44 | 63.55 | 62.11 |
| DB | 66.22 | 60.21 | 59.96 | 55.63 |
| DK | 65.79 | 62.98 | 58.18 | 55.48 |
| DDCB | 58.15 | 55.65 | 51.15 | 49.71 |

methods such as DR, DD, DC, DB, DK, and DDCB are lower than their counterparts methods R, D, C, B, K, and DCB. The green colour represents the lower execution times and red shows the highest execution time.

Hence, comparing to all the methods, DDCB method is more efficient than any other method in terms of number of iterations that it takes to influence the overall network nodes. However, despite the fact that driver nodes identification methods have higher theoretical time complexity, they are useful in the context of the influence spread in complex networks because driver-based seed selection methods require fewer

Table 5.8 : Total Execution Times (in hours – Hrs.) of All Seed Selection Methods in Each Social Network with 1% Seed Size when 100% Influence is Reached.

| Networks | Execution Times (Hrs.) : Maximum Influence at 1% Seed Size | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | R | D | C | B | K | DCB | DR | DD | DC | DB | DK | DDCB |
| FB | 1.53 | 1.25 | 1.25 | 1.22 | 1.06 | 1.35 | 1.50 | 1.50 | 1.75 | 1.50 | 1.35 | 0.75 |
| ZKC | 0.75 | 0.75 | 0.75 | 0.75 | 0.16 | 0.73 | 0.25 | 0.22 | 0.21 | 0.21 | 0.69 | 0.16 |
| Twitter | 3.02 | 2.51 | 2.45 | 2.45 | 2.35 | 2.25 | 1.95 | 1.95 | 1.75 | 1.95 | 1.95 | 1.95 |
| Diggs | 55.32 | 52.45 | 50.98 | 50.56 | 48.78 | 48.23 | 19.01 | 18.99 | 15.92 | 16.45 | 16.25 | 16.00 |
| Youtube | 53.31 | 51.43 | 50.26 | 50.23 | 48.54 | 48.13 | 18.99 | 18.79 | 18.76 | 16.45 | 15.21 | 14.00 |
| Ego | 4.00 | 4.00 | 4.00 | 3.19 | 1.43 | 3.00 | 3.02 | 2.51 | 2.45 | 2.45 | 2.39 | 2.25 |
| LC | 1.53 | 1.25 | 1.15 | 1.12 | 0.94 | 1.50 | 1.50 | 1.50 | 1.50 | 1.50 | 1.11 | 0.69 |
| LF | 1.33 | 1.13 | 1.33 | 1.13 | 0.26 | 1.03 | 0.31 | 0.28 | 0.28 | 0.28 | 0.95 | 0.26 |
| PF | 1.42 | 1.42 | 1.42 | 1.42 | 0.29 | 1.00 | 0.37 | 0.35 | 0.35 | 0.31 | 0.95 | 0.29 |
| MFb | 4.00 | 3.75 | 3.15 | 2.50 | 2.25 | 2.05 | 3.02 | 2.51 | 2.45 | 2.45 | 1.95 | 2.05 |
| DHR | 3.35 | 2.55 | 2.55 | 2.45 | 2.45 | 2.45 | 2.00 | 2.00 | 2.00 | 1.51 | 0.99 | 1.51 |
| DRO | 4.50 | 3.15 | 3.35 | 4.13 | 2.35 | 4.13 | 3.02 | 2.51 | 2.45 | 2.45 | 3.94 | 2.25 |
| DHU | 4.50 | 4.50 | 4.05 | 3.15 | 2.35 | 3.05 | 3.02 | 2.51 | 2.45 | 2.45 | 2.75 | 2.25 |
| MG | 4.75 | 3.75 | 3.13 | 3.13 | 2.45 | 3.03 | 3.35 | 2.55 | 2.55 | 2.45 | 2.83 | 2.45 |
| L | 2.75 | 2.75 | 2.75 | 2.15 | 1.05 | 0.75 | 1.53 | 1.25 | 1.25 | 1.22 | 1.85 | 0.72 |
| FbAR | 4.00 | 4.00 | 3.39 | 3.15 | 2.45 | 2.45 | 3.35 | 2.55 | 2.55 | 2.45 | 1.95 | 2.05 |
| FbA | 3.02 | 2.51 | 2.45 | 2.45 | 2.35 | 2.25 | 1.91 | 1.91 | 1.87 | 1.91 | 1.11 | 1.21 |
| FbG | 2.75 | 2.15 | 2.15 | 2.65 | 1.05 | 0.75 | 1.53 | 1.25 | 1.25 | 1.22 | 1.97 | 0.72 |
| FbN | 3.45 | 3.45 | 3.15 | 2.35 | 2.35 | 2.25 | 3.02 | 2.51 | 2.45 | 2.45 | 1.85 | 2.25 |
| FbP | 2.33 | 2.33 | 2.23 | 1.23 | 1.13 | 0.68 | 1.15 | 1.12 | 0.99 | 0.99 | 1.03 | 0.95 |
| FbPF | 2.97 | 2.97 | 2.57 | 1.97 | 1.97 | 2.25 | 3.02 | 2.51 | 2.45 | 2.45 | 1.57 | 2.45 |
| FbT | 2.37 | 2.34 | 2.37 | 1.37 | 1.35 | 0.69 | 1.53 | 1.25 | 1.15 | 1.12 | 1.15 | 0.94 |

iterations to influence all nodes.

## 5.4 Discussion and Conclusion

In this chapter, we focused on Research Challenge RC4, Research Question RQ3 and Research Objective RO4. In this study, we used driver nodes selection methods as seed selection strategies in the influence spreading process to evaluate how they affect the spread time and the influence number of influenced nodes, both in generated and real social networks. This is the first research that brings the fields of control and influence together and proposes new seed selection methods that are inspired by concepts from control theory. This contribution addresses Research Objective RO4, that stated, "To implement different traditional seed selection methods and driver-based methods to generate influence in synthetic and real networks". We have compared traditional seed selection methods (R, D, C, B, DCB and K) with driver based seed selection methods as their sibling methods (DR, DD, DC, DB, DDCB and DK). We can draw very clear key contributions based upon the obtained results.

First, based on Section 5.3 we can say that all driver based seed selection methods outperforms the traditional seed selection methods in terms of percentage of influenced nodes in generated networks as well as real social networks. We further conclude that, if we have a better seed selection set at the beginning of the spreading process, it is high chance that the more number of nodes could be influenced as compared to when we just apply traditional seed selection methods. Moreover, even if we do random seed selection from driver nodes, they perform better than any of the traditional seed selection strategies. The main contribution here is the fact that, when applying driver based seed selection methods, even if the seed size is small those methods are able to achieve higher number of influenced nodes. Experiment results are similar in both generated and real networks. Secondly, we learn that, for sparse networks where density is very low, percentage of influenced nodes is higher in driver based seed selection methods as compared to traditional methods. We see this phenomenon for generated and real networks such as Youtube and Diggs, which are the lowest density networks. We see that even in these networks with small seed sizes driver based methods outperforms their sibling methods. If we complete graph, 100% influence can be achieved, regardless of the seed selection method the influence process is quick. When density is 1, all seed methods work in the same way – they become random. Important conclusion here is when we compared the percentage of influence in lowest and medium density networks for random and as well as social networks. From this comparison When we analysed the time complexity of seed selection methods, we see that, identifying driver nodes is a very complex task, but we do not need to calculate centrality measures for all nodes and the number of iterations required to reach the 100% influence in a network reduces when we use driver based methods. Thus, actually we are able to save time and resources. To conclude, 100% influence can be achieved, regardless of the seed selection method the influence process is quick. When density is 1, all seed selection methods work in the same way – they become random. The important thing to note is if the network density is very low, like in the case of Diggs network (0.000002), the driver based methods outperforms traditional methods in terms of number of iterations needed to achieve 100% coverage. For synthetic networks, we see the maximum gain that DDCB method has achieved over other techniques is 10.51% which is substantial average gain over Random seed selection method when seed size is 1% as shown in Table 5.4. The fact that DDCB method outperforms all others for small seed sizes, shows that it has great potential in situations with limited budget where only small number of nodes can be initially activated. This can be concluded based upon the percentage of influenced nodes in all generated networks. Those results are also confirmed by the experiments on real networks.

Our work identifies the relative performance of different seed selection methods in terms of influence spread in a wide variety of network structures, however further work can be done in identifying the characteristics of the individual nodes which lead to them serving as highly effective seed nodes. A deeper understanding of the structural contributions of individual nodes may lead to further improvements to seed selection methods. Driver based methods show improvement over traditional seed selection methods in both synthetic and real–world networks. Results for DDCB are very promising, as this method consistently outperforms other seed selection methods in both kinds of networks. The observed usefulness of our novel approaches addresses the research question "How can the concepts from network control be used in the spread of influence field?" of the research topic.

Finally, we can conclude that in order to achieve maximum influence in fewer iterations, not only density, but seed size and ranking of driver nodes is also important.

# Chapter 6

# Influence Models, Communities and Driver Nodes

This chapter explores the possibility of using community structure in social networks to reduce the cost of identifying driver nodes, and whether this remains a feasible approach for network control and influence spread methods. In Figure 6.1 we can see the highlighted tasks that were carried out to fulfil RC5 (i.e., Using driver nodes identified in local network structures to maximize influence spread in social networks.) and RO5 (i.e, To measure the efficiency and effectiveness of global seed selection methods and local seed selection methods.) as part of the whole thesis. This chapter contains the following sections: Section 6.1 describes related work and motivation behind this experimental study. Sections 6.2 and 6.3 describe (i) the research methodology in detail and (ii) include results and analysis of the experiments performed respectively. Finally, the conclusions drawn from the experiments and future work are discussed in Section 6.4.

## 6.1 Background

Due to the prevailing use of online social networking sites, social networks are very much a hot topic in network science. Nowadays, we have a good understanding of network structures and attention has shifted more towards their prediction, influence, and control. Full control of social networks is very hard to achieve due to their varying structures, dynamics, and the complexities of human behaviour. This study looks into how driver nodes, which enable complex network control, can be used in the context of influence spread in the social network space. We use driver nodes at both the global and community level to ‘divide and conquer’ the time-consuming

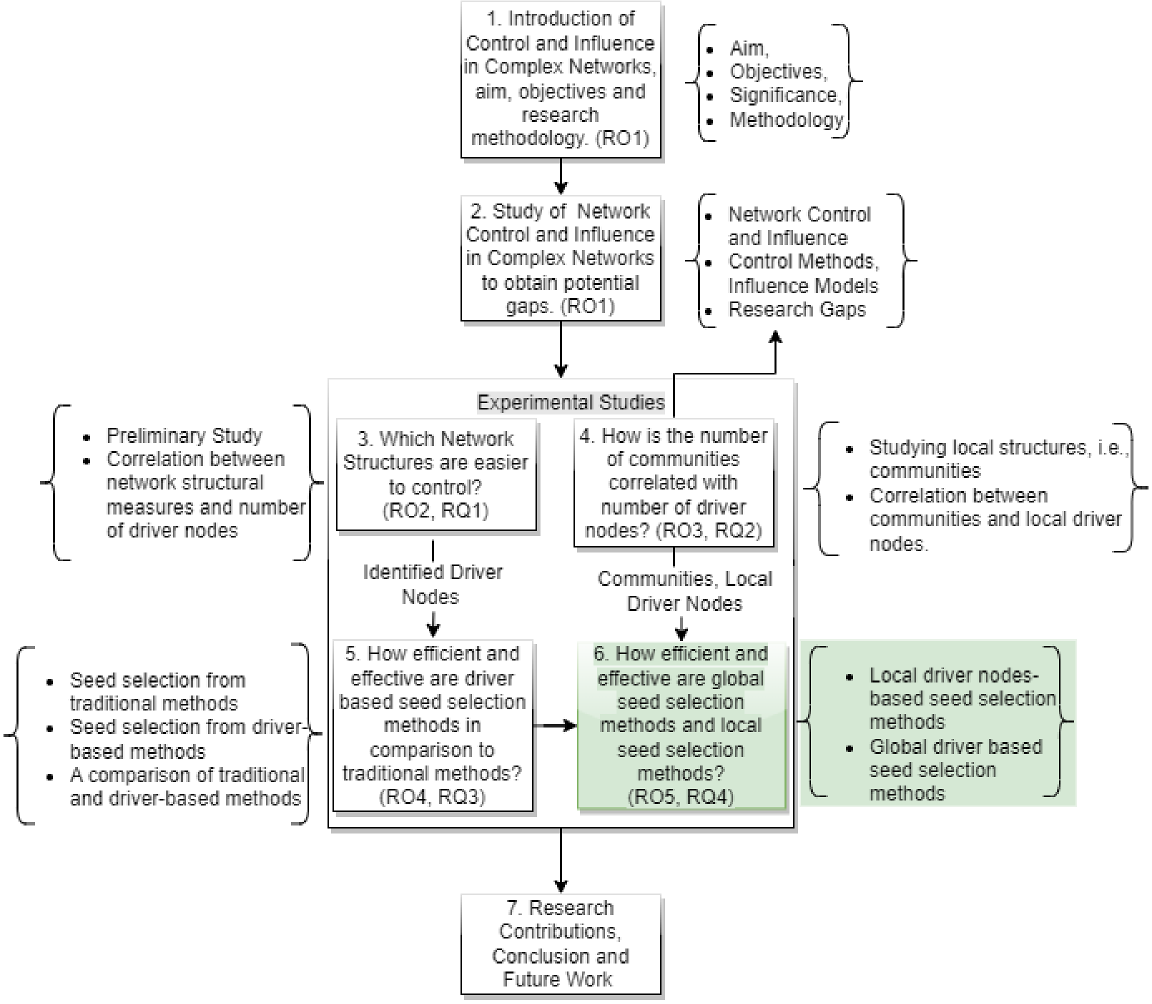

1. Introduction of Control and Influence in Complex Networks, aim, objectives and research methodology. (RO1)
Aim,
Objectives,
Significance,
Methodology
2. Study of Network Control and Influence in Complex Networks to obtain potential gaps. (RO1)
Network Control and Influence
Control Methods, Influence Models
Research Gaps
Experimental Studies
Preliminary Study
Correlation between network structural measures and number of driver nodes
3. Which Network Structures are easier to control? (RO2, RQ1)
4. How is the number of communities correlated with number of driver nodes? (RO3, RQ2)
Studying local structures, i.e., communities
Correlation between communities and local driver nodes.
Identified Driver Nodes
Communities, Local Driver Nodes
Seed selection from traditional methods
Seed selection from driver-based methods
A comparison of traditional and driver-based methods
5. How efficient and effective are driver based seed selection methods in comparison to traditional methods? (RO4, RQ3)
6. How efficient and effective are global seed selection methods and local seed selection methods? (RO5, RQ4)
Local driver nodes-based seed selection methods
Global driver based seed selection methods
7. Research Contributions, Conclusion and Future Work

Figure 6.1 : Research Methodology : Chapter 6

problem of driver node identification. Until recently, we did not know if and how the structure of social networks correlated with the number of driver nodes required to control the network [182]. As driver nodes play a key role in achieving control of a complex network, identifying them and studying their correlation with network structure measures can bring valuable insights, such as what network structures are easier to control, and how we can alter the structure in our favour to achieve the maximum control over the network. Our previous work [182] determines the relationship between some global network structure measures and the number of driver nodes. This study builds an understanding of how global network profiles of synthetic (random, small-world, scale-free) and real social networks influence the number of driver nodes needed for control. It focuses on global structural measures such as network density and how it can play an important role in determining the size of a suitable set of driver nodes. Our results show that as density increases in networks with structures exhibited by random, small world and scale free networks, the number of driver nodes tends to decrease. In this work we explore the potential that exploiting local structures (in this study we focus on communities) can offer in developing control of, and influencing, the network. Finding communities in a social network is itself a difficult task due to both dynamic and combinatorial factors [187].

The Influence Maximisation problem aims at discovering an influential set of nodes that can influence the highest number of nodes in social networks in the shortest possible time. A set of these nodes can be used to propagate influence in terms of social media news, advertising, etc. Several algorithms have been proposed to solve the influence maximisation problem that identify a set of nodes that is highly influential as compared to other nodes. For example Basic Greedy [103], CELF [115], CELF++ [68], Static Greedy [34], Nguyen's Method [159], Brog et al.'s Method [15],SKIM [36], TIM+ [199], IMM [198], Stop and Stare [157], Zohu et al.'s Method [241] and BCT [158] are some of those algorithms. Many algorithms have high run times when identifying a set of nodes to diffuse the influence through a social network, therefore there is a need to work on exploring different types of nodes if those can work towards achieving the high influence [101]. The problem of influence maximisation has high relevancy to the spreading of information on networks. The two most common network-based models are Independent Cascade model [103] and Threshold models [69]. In one of the previously proposed framework, the possible seed set has been identified by analysing the properties of the community structures in the networks. The CIM algorithm (i.e. Community-Based Influence Maximisation), utilises hierarchical clustering to detect communities from the networks and then uses the information of community structures to identify the possible seed nodes candidates, and at the end the final seed set is selected from the candidate seed nodes [32]. From the previous work such as [32] and [101], we can see, that by detecting communities and then selecting seed nodes from those communities can be an effective strategy to maximise influence.

From Chapter 3, and  [182], following main results were achieved, which are the basis for further new experiments that helps us in achieving Research Challenge RC5.

- Correlation between network density and number of driver nodes: For this purpose, network densities and number of driver nodes in those networks are plotted against each other to see the increase/decrease in number of driver nodes with the increase/decrease in the densities of the networks.

- Structural measures and density of driver nodes: In this step a comparison of structural measures like (Betweenness Centrality, Closeness Centrality, Nodes, Edges, Eigenvector Centrality and Clustering Coefficient) is presented with the density of number of driver nodes. Density of number of driver nodes is defined as total number of driver nodes divided by total number of nodes in the network.

In our proposed methods, we utilise driver nodes within the communities of networks for the influence spread using Linear Threshold Model. To make the driver nodes more influential, we propose different ranking mechanisms to see the number of nodes influenced after a certain time with a certain percentage of seed nodes in synthetic as well as real networks. The detail of network datasets has been presented in the later sections. We explain our method to select seed nodes from the communities in the next section.

## 6.2 Research Methodology and Experimental Design

This work springs from the Research Question RQ4, the main focus is in finding out, whether network control methods, in particular driver node selection, can be used to improve seed selection in influence models.

This prompts two possible approaches: (i) using driver nodes selected from the network as a whole, and (ii) using driver nodes selected at the community level as seeds. For all experiments, we used the Linear Threshold Model to model influence propagation. We used a set threshold of 0.5 for the network diffusion model. We have previously observed that a threshold value of at least 0.4 accelerates influence propagation [32]. The datasets description can be seen in Chapter 4 Section 4.2.

### 6.2.1 Influence spread using global driver nodes as seeds

The first experiment focuses on the seed selection process from the global perspective. Driver nodes are selected from the network as a whole, ranked, and finally used as seeds in the influence process. The below described approach has been proposed in [181]. As it outperforms other state-of-the art ranking methods, it serves in this study as a benchmark to show a difference between global- and local-level seed selection methods. The steps are as follows:

1. Minimum Dominating Set method [147] has been used to identify the number of driver nodes from the networks. More detail of this process can be found in [182]. DMS has been found by using greedy algorithm. At start, the dominating set is empty. Then in each iteration of the algorithm, a vertex is added to the set such

that it covers the maximum number of previously uncovered vertices. Then, if more than one vertex fulfils this criteria, the vertex is added randomly among the set of nominated vertices [186].

2. We ranked the nodes using different ranking mechanisms. The goal was to achieve an efficient set of nodes as seeds that can achieve maximum or full influence more quickly. The ranking mechanisms used are: Random, Degree Centrality, Closeness Centrality, Betweenness Centrality, Kempe Ranking, Degree-Closeness-Betweenness. We tested various seed set sizes: 1%, 10%, 20%, 30%, 40% and 50% of all detected driver nodes ranked these methods. In each of the methods, the driver nodes are ranked based on the following measures:

   - In Random (Driver Random – DR) we ranked the driver nodes randomly.
   - In Degree seed selection (DD) we ranked the driver nodes based on their degree in descending order.
   - For Closeness Centrality based seed selection method (Driver Closeness – DC), we ranked the nodes on the basis of their closeness centrality in descending order.
   - For Betweenness Centrality based seed selection method (Driver Betweenness – DB), we ranked the nodes on the basis of their betweenness centrality in descending order.
   - For Degree-Closeness-Betweenness method (Driver Degree Closeness Betweenness – DDCB), we ranked (in descending order) the driver nodes on the basis of the average of degree, closeness and betweenness centralities of each driver nodes.
   - For Kempe ranking (Driver Kempe – DK), we start by spreading influence through all the driver nodes as seed nodes. So we calculate the total number of nodes influenced by each driver node already in the seed set, and

then rank them in descending order. After ranking, we select a percentage of nodes that are required for a seed set.

- Linear Threshold Model (LTM) has been implemented for influence spread process. In LTM the idea is that a node becomes active if a sufficient part of its neighbourhood is active. Each node $u$ has a threshold $t \in [0, 1]$. The threshold represents the fraction of neighbours of $u$ that must be active in order for $u$ to become active. At the beginning of the process, a small percentage of nodes (seeds) is set as active in order to start the process. In the next steps a node becomes active if the fraction of its active neighbours is greater than its threshold, and the whole process stops when no node is activated in the current step [38].

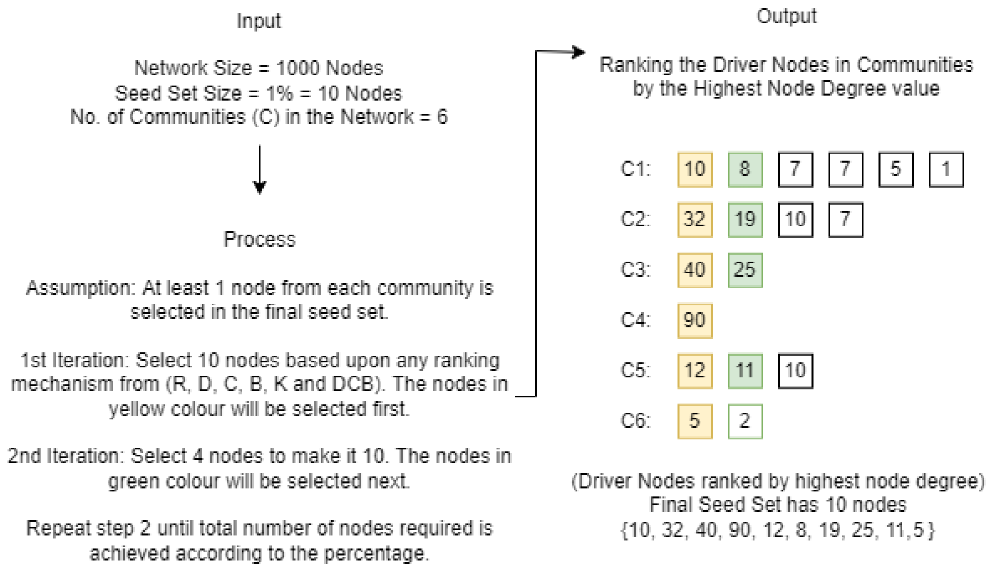
### 6.2.2 Influence spread using local driver nodes as seeds

The second experiment employs a new strategy: first identify communities in the network, and then identify driver nodes on a per-community basis.

Once driver nodes for each community are identified, they are then ranked using the same ranking mechanisms as in the first experiment, with seed sets chosen to cover all communities (detailed below). In detail, the approach is as follows:

1. Firstly, communities are identified in the network. This was done using Girvan-Newman algorithm [64]. The Girvan–Newman algorithm detects communities by progressively removing edges from the original graph in order of the highest betweenness centrality.

2. Within each community, candidate driver nodes were identified using the Minimum Dominating Set [147] approach as used with the whole network. Correlation between community densities and number of driver nodes is found by obtaining densities of the communities and identifying number of driver nodes in those communities by MDS method. Difference (Diff.) between total num-



Network Size = 1000 Nodes
Seed Set Size = 1% = 10 Nodes
No. of Communities (C) in the Network = 6

Assumption: At least 1 node from each community is selected in the final seed set.

1st Iteration: Select 10 nodes based upon any ranking mechanism from (R, D, C, B, K and DCB). The nodes in yellow colour will be selected first.

2nd Iteration: Select 4 nodes to make it 10. The nodes in green colour will be selected next.

Repeat step 2 until total number of nodes required is achieved according to the percentage.

Ranking the Driver Nodes in Communities by the Highest Node Degree value

| | | | | | | |
|---|---|---|---|---|---|---|
| C1: | 10 | 8 | 7 | 7 | 5 | 1 |
| C2: | 32 | 19 | 10 | 7 | | |
| C3: | 40 | 25 | | | | |
| C4: | 90 | | | | | |
| C5: | 12 | 11 | 10 | | | |
| C6: | 5 | 2 | | | | |

(Driver Nodes ranked by highest node degree)
Final Seed Set has 10 nodes
{10, 32, 40, 90, 12, 8, 19, 25, 11,5}

Figure 6.2 : An example showing the process for selecting seed nodes set from the driver nodes identified in network communities

ber of driver nodes identified in overall networks (NDN) as compared to the number of driver nodes found in communities of those networks (NDNC) is also obtained. The Diff. tells us, the significance of identifying driver nodes within communities, like following a divide and conquer approach.

3. To rank the nodes, we introduce a multi-round selection process. This process effectively ranks driver nodes within each community according to the ranking criterion, then selects one node per community per round, in the order given by the ranking, until the total percentage to be chosen is reached. This is perhaps better explained by the following example, illustrated in Figure 6.2. Consider a network with 1,000 nodes and 6 communities. Select a ranking method, in this case the node degree. Choose a target percentage of nodes to use as seed nodes, 1% in the example. Now, in order to choose 10 nodes from the driver nodes detected in the communities, we select 6 nodes at first – the highest degree node

from each community, marked in yellow in the figure. In the second round, we can select at most 4 nodes to reach the target of 10 – from each community, we take the node with the second-highest node degree and rank these nodes according to their degrees and take the 4 nodes with the highest degree. We choose the same ranking mechanism for all the community based driver nodes seed selection methods i.e., the highest node degree, apart from the original ranking that is different in each technique as explained previously.

4. Influence spread in the overall network using Driver Based Seed Selection Methods is done by following a series of steps. Starting from identification of driver nodes from the networks, ranking of driver nodes based upon Random, Node Degree, Closeness Centrality, Betweenness Centrality, Kempe Ranking, Degree-Closeness-Betweenness Centralities combined. After ranking of driver nodes, we selected our seed set on the basis of percentage of nodes from that set. We run our LTM for different seed sets, namely for example 1%, 10%, 20%, 30%, 40% and 50%.

5. Influence spread through Driver Nodes in communities of Networks is done by identifying driver nodes in communities. However, there was a challenge of getting the ultimate seed set that has representation from all the communities of the network. For this purpose, we devised our ranking approach that makes sure that at least one driver node is selected from each community of the network to make sure that the nodes in those communities can also be part of the influence process. For each of the driver based seed selection methods, we used one unified approach to further rank the nodes so that we are able to select at least one node from each of the communities.

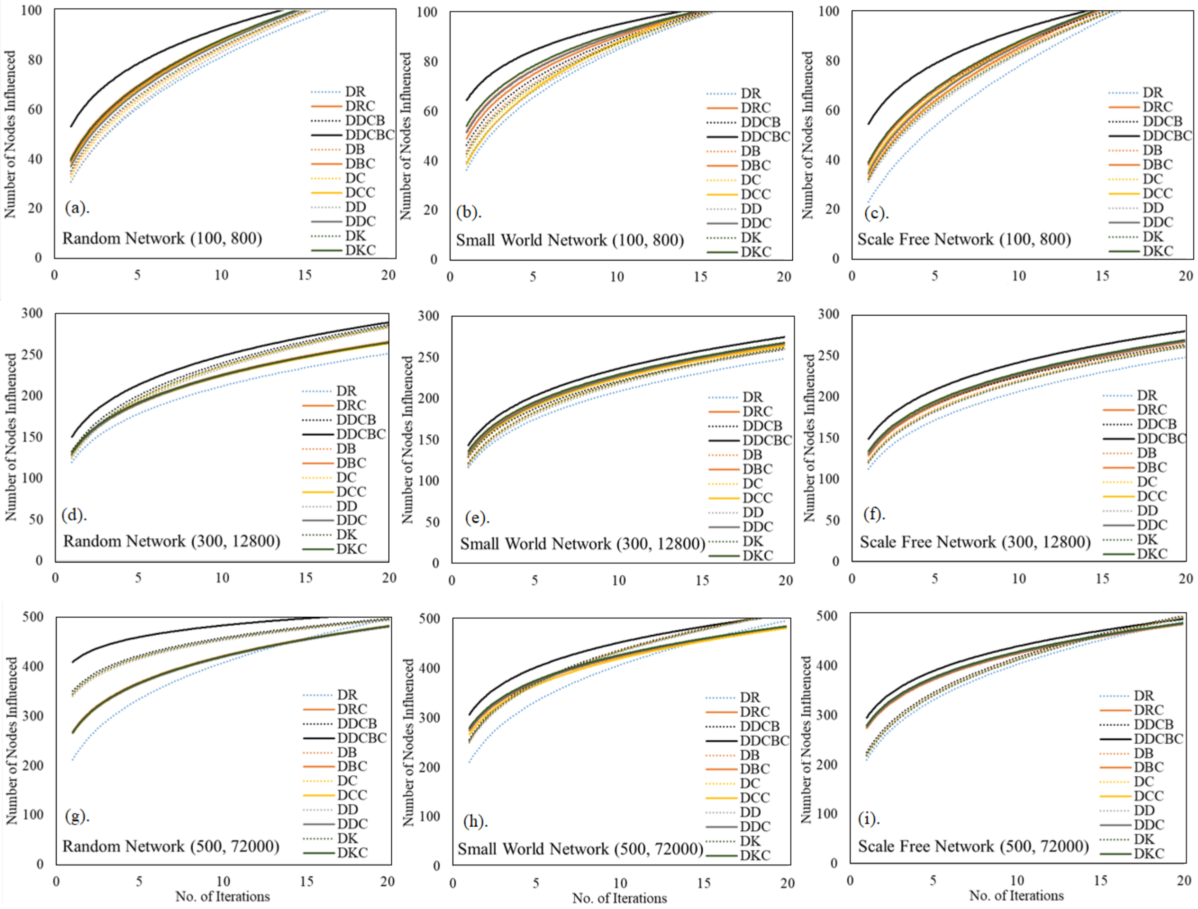


Figure 6.3 : Number of Nodes Influenced in Random, Small-World and Scale-Free Networks: when the number of nodes (N) is 100 and the number of edges (E) is 800 (Figures a, b and c); when N is 300 and E is 12800 (Figures d, e and f); when N is 500 and E is 72000 (Figures g, h and i). A Comparison of all methods for 20 iterations when the seed size is 1% is presented.

## 6.3 Results and Analysis

Six novel network level seed selection methods (i.e. Driver-Random (DR), Driver-Degree (DD), Driver-Closeness (DC), Driver-Betweenness (DB), Driver-Kempe (DK) and Driver-Degree-Closeness-Betweenness (DDCB)) have been proposed and tested on synthetic and real world networks before in [181] and the results show that those methods outperform their non-driver based counterparts. In this study, we use those methods but instead of selecting driver nodes from the global network, we propose a local approach where driver nodes are identified within the networks' communities. We name the new methods by adding C (for community) to the previously proposed methods (i.e, DRC - Driver-Random-Community, DDC - Driver-Degree-Community, DCC - Driver-Closeness-Community, DBC - Driver-Betweenness-Community, DKC - Driver-Kempe-Community and DDCBC - Driver-Degree-Closeness-Betweenness-Community). Below, we compare community based driver seed selection methods to network based driver seed selection methods.

### 6.3.1 Results From Generated Networks

This section covers the results and analysis of the experiments performed on generated networks.

#### 6.3.1.1 What is the speed and reach of the influence spread?

First, we compare the percentage of nodes influenced for global-level driver based seed selection methods and local-level (community) driver based seed selection methods. We perform the analysis iteration by iteration to see which seed selection methods enable to achieve the highest coverage the fastest.

In Figure 6.3, we can see trend-lines for all the seed selection methods (when seed set size is 1% of all the driver nodes) for random, small-world and scale-free networks. DDCBC method outperforms other methods in almost all the experimented cases. We

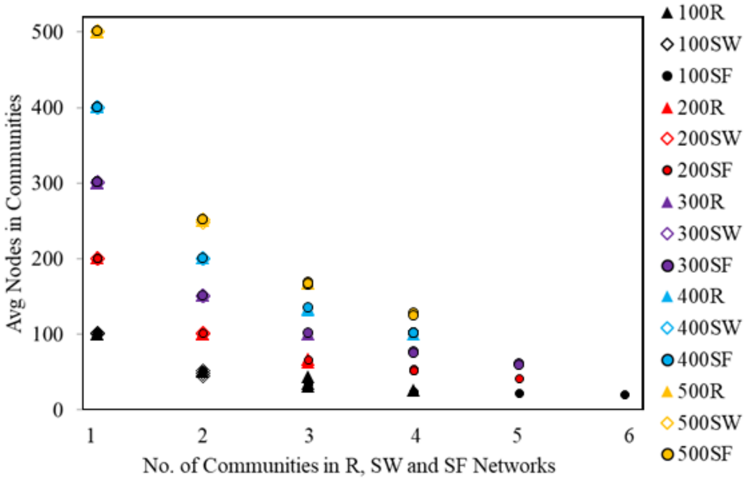


Figure 6.4 : Average Number of Nodes in Communities of Random, Small-World and Scale-Free Networks versus number of communities in those networks. The legend shows the Number of Nodes in communities of generated networks i.e. Random (R), Small-World (SW) and Scale-Free (SF).

can see a ‘head-start’ in the trend-line of DDCBC (represented in black colour) for all the networks when number of nodes in the network is 100 and number of edges is 800. This means that in only few iterations, DDCBC enables to influence more nodes than in the case of other seed selection methods.

Results in Figure 6.4 show that when the network is of small size, and density is approximately equal to 0.6, the influence spreads faster when using driver-community based seed selection methods than when the global-level driver based methods are employed. If we look at Figure 6.4, the network of smaller densities (i.e. 0.4), where number of nodes is 300 and number of edges is 2,800, the difference between the global-level driver based methods and community-level driver based methods is not so big. But we do see a gap between DDCBC method and other methods. Which tells us that, so far, DDCBC ranking of driver nodes in communities is working better than when we are using driver nodes of communities as seed nodes.

Although the comparison is done on a very small size of seed set (1% of all driver nodes), in DDCBC, we still achieve more influence earlier in the spreading process when using community-level driver based methods. It also gives us another insight regarding larger networks, their structures and densities, and how those are connected to spreading influence. We see that the spread is faster when density is higher than 0.5 as in the case of networks presented in the Figure 6.3 (network with 500 nodes and 72,000 edges). We can see that in those cases, the driver-community based method DRC, DDC, DBC, DKC and DDCBC outperforms their counterpart methods DR, DD, DB, DK and DDCB.

Based upon these observations, we conclude it does not matter which type of network it is, as long as its density is higher than 0.5 it will respond to the community-based seed selection methods better and the spread will be faster. Also, regardless of the network density, community-based method – DDCBC – outperforms all other methods Figure 6.3(a-f). This holds true for all the other settings as well. As when we have different edges for 100, 200, 300, 400 and 500 nodes networks.

#### 6.3.1.2 How much advantage do community—level driver based seed selection methods give?

Given a number of iterations $n$ and a method $X$, let $N_n^{infl}(X)$ denote the number of nodes influenced using the method $X$ after $n$ iterations. The Percentage Gain of method $A$ over method $B$ after $n$ iterations is then given by:

$$\frac{N_n^{infl}(A) - N_n^{infl}(B)}{N} \times 100 \tag{6.1}$$

where $N$ is the number of nodes in the network.

Table 6.1 shows the percentage gain of the DDCBC method over the global-level driver based methods. We represent only driver based methods (i.e. DR, DB, DC, DD, DK and DDCB), as the gain is higher over these methods as compared to other driver-community based methods (i.e. DRC, DBC, DCC, DDC and DKC) as well as they are our baseline for this study. Percentage gain is calculated by knowing the maximum number of nodes influenced after 20 iterations when seed size is 1%.

From Table 6.1 we can see the maximum gain in when the average density of the communities of the network is greater than 0.5. When the density reaches 1 all the methods perform very similar as spread in fully connected network behaves in a very similar way regardless of applied seed selection method. This highlights our previous point that density of network plays an important part in how effective a network is going to respond to the influence spread process. We can see the highest gain for DDCBC method in random networks, but DDCBC outperforms all global-level driver based methods in all the networks, except for the networks with densities equal or very close to 1.

From Figure 6.4, we can see the number of average nodes in communities versus the total number of communities in Random, Small-World and Scale-Free networks. The denser the network, the fewer communities we have, and those communities are denser than the previous ones. Hence, due to increase in community density, we see the higher percent gain in DDCBC method. The number of nodes influenced by DDCBC method increases, when there are fewer communities. Because when number of communities are less, they tend to be denser, hence the increase in number of nodes influenced. We see the difference in number of nodes influenced in DDCBC method which is bigger than compared to other methods.

Table 6.1 : A percentage gain table shows the percentage gain of DDCBC method over other seed selection methods in influencing the nodes in Random, Small-World and Scale-Free networks when the seed set size is 1% after 20 iterations. $N$ is number of nodes, $E$ is number of edges, $C$ is number of communities and $CD$ is average community density.

| N | E | C | CDensity | Random Networks | | | | | | Small-World Networks | | | | | | Scale-Free Networks | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Avg ± SD | DR | DB | DC | DD | DK | DDCB | DR | DB | DC | DD | DK | DDCB | DR | DB | DC | DD | DK | DDCB |
| 100 | 800 | 6 | 0.16±0.01 | 2.10 | 2.50 | 2.21 | 2.15 | 2.12 | 1.12 | 3.42 | 2.11 | 3.27 | 3.11 | 3.34 | 2.51 | 4.05 | 2.19 | 2.33 | 3.91 | 3.02 | 2.01 |
| | 1600 | 5 | 0.3±0.03 | 3.32 | 2.63 | 3.19 | 3.11 | 2.22 | 2.42 | 3.21 | 2.44 | 2.18 | 2.30 | 2.51 | 2.30 | 4.11 | 2.26 | 3.16 | 3.17 | 3.53 | 2.12 |
| | 2400 | 4 | 0.44±0.06 | 3.11 | 2.21 | 2.24 | 3.30 | 2.05 | 2.71 | 3.45 | 2.15 | 3.55 | 3.00 | 2.01 | 2.09 | 4.00 | 2.00 | 2.00 | 3.02 | 3.61 | 2.33 |
| | 3200 | 3 | 0.58±0.12 | 2.10 | 2.09 | 2.00 | 2.00 | 2.00 | 1.12 | 4.54 | 3.22 | 3.31 | 3.09 | 3.55 | 3.43 | 3.32 | 2.25 | 2.00 | 2.05 | 2.76 | 1.00 |
| | 4000 | 2 | 0.73±0.14 | 4.76 | 2.63 | 3.33 | 3.00 | 2.07 | 2.15 | 3.27 | 2.12 | 2.22 | 2.00 | 2.00 | 2.61 | 4.05 | 3.31 | 3.17 | 3.03 | 3.55 | 2.00 |
| | 4800 | 1 | 0.88±0.15 | 2.60 | 1.50 | 1.55 | 2.52 | 1.00 | 1.01 | 0.00 | 0.00 | 1.13 | 1.18 | 0.00 | 0.00 | 2.09 | 1.71 | 1.00 | 1.01 | 1.26 | 1.44 |
| | 4950 | 1 | 0.96±0.07 | 0.00 | 0.00 | 0.00 | 0.00 | 0.09 | 0.01 | 0.00 | 0.04 | 0.15 | 0.00 | 0.01 | 0.09 | 0.71 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 200 | 2400 | 5 | 0.12±0.01 | 4.00 | 4.00 | 4.00 | 5.02 | 4.04 | 4.48 | 5.32 | 5.11 | 5.09 | 5.44 | 5.08 | 4.00 | 5.09 | 4.62 | 4.51 | 4.39 | 4.82 | 4.00 |
| | 4800 | 4 | 0.23±0.02 | 3.00 | 20̇2 | 2.01 | 3.01 | 2.00 | 2.00 | 3.44 | 2.21 | 2.61 | 3.32 | 2.25 | 2.21 | 5.33 | 4.28 | 4.00 | 4.00 | 4.03 | 3.03 |
| | 7200 | 4 | 0.36±0.01 | 8.16 | 7.33 | 7.21 | 7.11 | 7.09 | 6.00 | 8.01 | 7.94 | 7.37 | 7.07 | 7.00 | 7.34 | 9.19 | 8.00 | 8.02 | 8.00 | 8.00 | 8.22 |
| | 9600 | 4 | 0.48±0.02 | 6.16 | 6.22 | 6.12 | 6.54 | 6.24 | 5.11 | 6.00 | 5.09 | 6.23 | 6.33 | 6.09 | 5.00 | 7.61 | 6.00 | 6.72 | 6.04 | 6.45 | 5.15 |
| | 12000 | 3 | 0.56±0.07 | 6.00 | 5.15 | 5.12 | 5.11 | 5.11 | 4.45 | 7.00 | 7.11 | 7.09 | 7.03 | 7.33 | 6.00 | 7.15 | 6.16 | 6.16 | 6.05 | 6.09 | 6.00 |
| | 14400 | 2 | 0.67±0.09 | 3.00 | 3.13 | 3.43 | 3.00 | 3.00 | 2.00 | 4.04 | 3.02 | 4.01 | 4.00 | 3.00 | 3.88 | 4.09 | 3.48 | 3.81 | 3.12 | 3.01 | 2.11 |
| | 16800 | 1 | 0.78±0.11 | 2.99 | 1.37 | 1.71 | 1.00 | 1.00 | 0.00 | 2.47 | 1.73 | 2.27 | 2.48 | 2.81 | 1.71 | 3.68 | 1.90 | 1.40 | 2.21 | 1.00 | 1.00 |
| | 19200 | 1 | 0.9±0.1 | 1.31 | 0.00 | 0.00 | 0.00 | 0.00 | 0.01 | 2.81 | 0.00 | 1.31 | 1.25 | 0.00 | 0.00 | 1.13 | 1.28 | 1.18 | 1.92 | 1.37 | 0.00 |
| | 19900 | 1 | 0.97±0.06 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 300 | 12800 | 5 | 0.31±0.03 | 4.11 | 3.00 | 3.66 | 3.05 | 3.51 | 2.00 | 4.00 | 3.71 | 3.00 | 3.00 | 3.14 | 2.84 | 5.00 | 3.00 | 3.00 | 4.17 | 4.91 | 3.73 |
| | 19200 | 5 | 0.41±0.03 | 4.00 | 3.00 | 3.62 | 3.22 | 2.94 | 2.27 | 3.13 | 2.34 | 3.12 | 3.33 | 3.91 | 2.20 | 4.83 | 3.01 | 3.00 | 3.82 | 3.63 | 3.49 |
| | 22400 | 4 | 0.46±0.06 | 4.00 | 3.00 | 3.08 | 3.00 | 2.25 | 2.47 | 4.41 | 3.58 | 3.45 | 3.29 | 3.11 | 2.20 | 5.00 | 3.02 | 3.00 | 4.00 | 3.00 | 3.00 |
| | 25600 | 4 | 0.53±0.08 | 4.14 | 2.27 | 2.09 | 3.65 | 2.37 | 2.18 | 3.99 | 2.02 | 3.73 | 3.00 | 3.00 | 2.00 | 4.07 | 3.16 | 3.72 | 3.81 | 3.00 | 2.91 |
| | 28800 | 3 | 0.58±0.1 | 3.00 | 2.00 | 2.00 | 3.02 | 2.91 | 2.00 | 2.00 | 2.18 | 2.69 | 2.03 | 2.15 | 1.70 | 3.38 | 2.00 | 2.18 | 2.16 | 2.42 | 2.71 |
| | 32000 | 2 | 0.63±0.17 | 6.00 | 4.00 | 4.00 | 4.00 | 4.02 | 3.05 | 4.11 | 3.62 | 3.51 | 3.02 | 3.00 | 3.95 | 5.47 | 4.45 | 4.15 | 4.13 | 4.00 | 3.11 |
| | 35200 | 1 | 0.69±0.16 | 10.00 | 8.00 | 8.07 | 8.62 | 8.05 | 8.33 | 5.16 | 5.00 | 5.62 | 5.71 | 5.43 | 4.11 | 6.04 | 5.05 | 5.18 | 5.00 | 5.09 | 4.00 |
| | 38400 | 1 | 0.76±0.17 | 13.04 | 6.16 | 7.43 | 7.12 | 7.63 | 7.55 | 10.00 | 3.37 | 3.28 | 3.63 | 3.00 | 2.00 | 13.41 | 4.16 | 4.09 | 4.01 | 4.00 | 4.00 |
| | 41600 | 1 | 0.83±0.17 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 1.00 | 0.03 | 2.04 | 2.00 | 2.23 | 2.84 | 1.71 |
| | 44850 | 1 | 0.91±0.15 | 0.01 | 0.82 | 0 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.12 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |

| N | E | C | CDensity | Random Networks | | | | | | Small-World Networks | | | | | | Scale-Free Networks | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Avg ± SD | DR | DB | DC | DD | DK | DDCB | DR | DB | DC | DD | DK | DDCB | DR | DB | DC | DD | DK | DDCB |
| 400 | 40000 | 4 | 0.43±0.12 | 23.02 | 21 | 21.28 | 22.00 | 21.56 | 21.38 | 5.00 | 4.05 | 4.15 | 4.15 | 4.00 | 4.22 | 5.00 | 4.18 | 4.04 | 4.05 | 4.00 | 3 |
| | 44000 | 4 | 0.48±0.12 | 25.21 | 22.63 | 22.00 | 22.84 | 22.32 | 22.38 | 4.32 | 4.53 | 4.00 | 4.00 | 4.00 | 3.00 | 6.00 | 4.15 | 4.72 | 5.73 | 4.00 | 4.17 |
| | 48000 | 4 | 0.53±0.12 | 25.00 | 21.91 | 21.36 | 21.00 | 21.00 | 21.37 | 9.03 | 8.05 | 8.00 | 8.32 | 8.26 | 7.16 | 10.11 | 8.31 | 8.00 | 9.04 | 8.14 | 8.48 |
| | 52000 | 4 | 0.58±0.12 | 22.04 | 12.18 | 12.44 | 13.00 | 12.32 | 12.11 | 12.63 | 11.00 | 11.00 | 11.05 | 11.33 | 11.64 | 13.00 | 11.04 | 12.00 | 12.04 | 12.00 | 11.17 |
| | 60000 | 3 | 0.67±0.14 | 18.09 | 12.45 | 12.52 | 12.11 | 12.23 | 12.45 | 10.00 | 9.54 | 9.75 | 10.63 | 10.12 | 9.18 | 12.00 | 10.32 | 10.46 | 11.10 | 11.47 | 10.82 |
| | 64000 | 2 | 0.76±0.07 | 13.01 | 8.05 | 9.15 | 9.55 | 8.16 | 9.27 | 7.28 | 7.56 | 7.99 | 7.02 | 7.48 | 6.18 | 8.17 | 7.00 | 7.25 | 7.57 | 7.11 | 7.03 |
| | 68000 | 1 | 0.83±0.03 | 8.69 | 6.00 | 6.03 | 6.05 | 6.16 | 6.37 | 5.16 | 4.82 | 4.93 | 4.91 | 4.00 | 4.04 | 7.88 | 5.00 | 6.00 | 6.00 | 6.00 | 5.00 |
| | 72000 | 1 | 0.88±0.03 | 4.52 | 1.00 | 1.27 | 2.29 | 1.00 | 1.00 | 5.15 | 4.12 | 4.18 | 4.10 | 4.11 | 4.73 | 4.38 | 3.00 | 3.17 | 3.00 | 3.00 | 3.00 |
| | 76000 | 1 | 0.93±0.03 | 1.00 | 0.00 | 0.02 | 0.00 | 0.04 | 0.03 | 1.19 | 1.00 | 1.00 | 1.00 | 1.00 | 0.00 | 1.00 | 2.00 | 2.00 | 3.00 | 2.00 | 2.00 |
| | 98000 | 1 | 0.98±0.03 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 500 | 72000 | 4 | 0.52±0.1 | 23.10 | 15.72 | 15.18 | 16.00 | 16.04 | 15.03 | 11.66 | 6.04 | 6.63 | 6.04 | 6.00 | 6.00 | 12.04 | 11.06 | 11.00 | 11.03 | 11.52 | 10.15 |
| | 76800 | 3 | 0.56±0.1 | 21.45 | 16.28 | 16.91 | 16.09 | 16.04 | 16.00 | 11.00 | 6.94 | 7.64 | 7.32 | 6.67 | 6.00 | 7.01 | 6 | 6.64 | 6.00 | 6.00 | 6.00 |
| | 81600 | 4 | 0.6±0.09 | 19.02 | 13.13 | 14.62 | 14.25 | 13.75 | 13.95 | 10.84 | 7.75 | 8.00 | 8.03 | 8.33 | 7.55 | 8.67 | 7.15 | 7.83 | 7.63 | 7.00 | 6.00 |
| | 86400 | 3 | 0.69±0.01 | 19.01 | 13.04 | 13.56 | 13.73 | 13.00 | 13.00 | 10.17 | 2.00 | 2.00 | 2.00 | 2.00 | 1.18 | 9.04 | 8.64 | 8.72 | 8.81 | 8.09 | 7.60 |
| | 91200 | 3 | 0.73±0.01 | 15.73 | 14.20 | 14.39 | 14.00 | 14.74 | 14.68 | 7.95 | 3.39 | 3.85 | 3.31 | 3.00 | 3.94 | 3.92 | 2.50 | 2.00 | 2.00 | 2.00 | 1.00 |
| | 96000 | 3 | 0.76±0.01 | 12.00 | 10.00 | 10.00 | 10.05 | 10.71 | 10.00 | 7.00 | 2.23 | 2.19 | 2.30 | 2.15 | 1.00 | 5.00 | 3.00 | 3.58 | 4.72 | 4.29 | 3.05 |
| | 100800 | 1 | 0.81±0.01 | 8.00 | 8.00 | 8.00 | 9.04 | 9.05 | 8.18 | 7.28 | 1.15 | 1.63 | 2.25 | 1.27 | 1.39 | 4.33 | 3.30 | 3.50 | 3.06 | 3.00 | 2.00 |
| | 105200 | 1 | 0.84±0 | 3.01 | 4.63 | 5.30 | 5.00 | 5.00 | 5>00 | 3.07 | 0.09 | 1.19 | 1.00 | 0.00 | 0.00 | 2.70 | 1.16 | 1.00 | 1.00 | 1.00 | 0.00 |
| | 110000 | 2 | 0.88±0 | 0.00 | 0.30 | 0.30 | 4.20 | 4.09 | 3.10 | 0.00 | 0.05 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 1.19 | 1.18 | 1.00 | 1.00 | 0.00 |
| | 124750 | 2 | 0.97±0.06 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |

Table 6.2 : A percentage gain table shows the percentage gain of DDCBC method over other seed selection methods in influencing the nodes of the social networks. Average Community Densities of the networks are as follows: FB (0.06±0.02), ZKC (0.32±0.4), Twitter (0.00029±0.05), Diggs (0.00008±0.007), Youtube (0.000012±0.04), Ego (0.00034±0.05), LC (0.007±0.032), LF (0.0073±0.09), PF (0.015±0.54), MFb (0.001±0.43), DHR (0.00085±0.21), DRO (0.0005±0.4), DHU (0.0004±0.63), MG (0.0011±0.03), L (0.0019±0.54), FbAR (0.0014±0.03), FbA (0.0015±0.09), FbG (0.0075±0.05), FbN (0.0013±0.003), FbP (0.0049±0.003), FbPF (0.004±0.032) and Fbt (0.0051±0.05)

| N | E | C | Networks | Seed Selection Methods (20% of all nodes) | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | DR | DD | DC | DB | DDCB | DK | DRC | DDC | DCC | DBC | DKC |
| 4039 | 88234 | 180 | FB | 28.68 | 25.03 | 24.94 | 25.94 | 25.15 | 24.59 | 21.59 | 21.59 | 22.19 | 22.28 | 21.14 |
| 34 | 78 | 2 | ZKC | 12.18 | 4.00 | 2.82 | 2.09 | 1.95 | 1.73 | 1.18 | 1.18 | 1.27 | 1.00 | 1 |
| 23371 | 32832 | 350 | Twitter | 37.81 | 27.83 | 26.80 | 26.78 | 20.16 | 26.77 | 23.81 | 23.80 | 23.74 | 23.06 | 21.22 |
| 1924000 | 3298475 | 156432 | Diggs | 42.49 | 39.05 | 36.76 | 36.47 | 38.37 | 39.21 | 20.11 | 18.89 | 17.67 | 16.53 | 19.85 |
| 1134891 | 2987625 | 54983 | Youtube | 42.00 | 38.02 | 35.12 | 32.79 | 32.59 | 33.92 | 3.51 | 2.71 | 1.91 | 1.11 | 6.45 |
| 23629 | 39195 | 75 | Ego | 24.83 | 15.34 | 14.33 | 14.33 | 17.15 | 21.81 | 9.64 | 10.62 | 11.14 | 9.05 | 8.89 |
| 4658 | 33116 | 517 | LC | 33.84 | 26.62 | 25.61 | 25.61 | 25.52 | 31.81 | 22.40 | 23.23 | 23.98 | 21.65 | 22.06 |
| 874 | 1309 | 97 | LF | 19.29 | 10.62 | 9.56 | 9.34 | 9.25 | 9.33 | 8.38 | 9.35 | 10.20 | 7.86 | 9.11 |
| 1858 | 12534 | 206 | PF | 10.62 | 6.66 | 5.43 | 5.21 | 5.13 | 5.25 | 2.94 | 3.78 | 4.64 | 2.60 | 2.71 |
| 22470 | 171002 | 2643 | MFb | 25.44 | 22.16 | 21.11 | 21.11 | 21.10 | 21.11 | 15.07 | 15.80 | 22.70 | 20.43 | 16.8 |
| 54574 | 498202 | 6420 | DHR | 39.77 | 35.42 | 33.21 | 32.00 | 31.90 | 34.2 | 6.78 | 7.26 | 7.73 | 5.21 | 6.01 |

| N | E | C | Networks | Seed Selection Methods (20% of all nodes) | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | DR | DD | DC | DB | DDCB | DK | DRC | DDC | DCC | DBC | DKC |
| 41774 | 125826 | 4914 | DRO | 42.43 | 35.74 | 36.42 | 34.22 | 34.12 | 34.45 | 13.50 | 13.40 | 13.13 | 12.94 | 34.18 |
| 47539 | 222887 | 5592 | DHU | 45.40 | 35.77 | 34.52 | 34.33 | 34.13 | 38.85 | 26.35 | 27.02 | 25.84 | 25.61 | 25.33 |
| 37700 | 289003 | 4435 | MG | 30.54 | 27.43 | 26.07 | 26.25 | 26.07 | 26.34 | 16.14 | 15.49 | 16.05 | 10.35 | 14.86 |
| 7624 | 27806 | 759 | L | 26.55 | 25.25 | 24.04 | 23.82 | 23.79 | 23.81 | 18.34 | 18.11 | 17.75 | 17.71 | 17.70 |
| 50516 | 819306 | 5943 | FbAR | 39.97 | 32.40 | 31.18 | 30.95 | 30.93 | 31.30 | 29.43 | 29.14 | 30.85 | 28.56 | 29.28 |
| 13867 | 86858 | 1383 | FbA | 47.29 | 32.45 | 31.05 | 30.55 | 40.87 | 45.89 | 32.28 | 31.83 | 33.28 | 30.46 | 32.01 |
| 7058 | 89455 | 784 | FbG | 21.95 | 20.22 | 18.93 | 18.71 | 19.18 | 19.20 | 13.97 | 13.75 | 15.39 | 13.13 | 13.68 |
| 27918 | 206259 | 3284 | FbN | 33.82 | 23.03 | 22.00 | 21.95 | 21.96 | 22.01 | 12.85 | 12.64 | 12.18 | 12.10 | 12.40 |
| 5909 | 41729 | 562 | FbP | 31.73 | 22.90 | 21.76 | 21.40 | 21.87 | 21.89 | 15.89 | 15.47 | 15.15 | 14.90 | 15.31 |
| 11566 | 67114 | 1051 | FbPF | 39.61 | 32.21 | 30.85 | 30.57 | 30.39 | 30.48 | 26.30 | 26.12 | 25.85 | 25.21 | 26.21 |
| 3893 | 17262 | 387 | FbT | 25.93 | 22.84 | 24.70 | 24.29 | 17.73 | 17.77 | 19.37 | 18.46 | 13.71 | 13.36 | 17.63 |

### 6.3.2 Results From Social Networks

The observation that real-world social networks tend to contain dense communities suggests that community based driver node selection would have a significant advantage over global selection. This relationship with density is also apparent in the generated networks. To verify whether this intuition is correct, we conduct similar analysis to this performed on generated networks. First, we analyze the percentage of nodes influenced by each method over 100 iterations with a seed set size of 20% of driver nodes. We have run the experiments for the seed set sizes from 1%, 10%, 20%, 30%, 40% and 50%. We show the comparison in case of 20% seed size, as it is the lowest seed set level to reach maximum influence in at most 100 iterations. We note however that there are also improvements at smaller seed set sizes.

#### 6.3.2.1 What is the speed and reach of the influence spread?

Figures 6.5, 6.6 and 6.7 show a comparison between global-level driver based seed selection methods and community-level driver based seed selection methods. We grouped the networks on the basis of their sizes and densities to analyse the results effectively. From Figure 6.5, we see a higher density of networks. The densities of these networks are: FB (0.01), Z ( 0.13), LC (0.003), LF (0.003), PF (0.007), FbG (0.003), FbP (0.002), FbPF (0.001) and FbT (0.002). Overall comparison tells us that, in these networks, there is less difference between the percentage of number of nodes influenced after 100 iterations. This indicates that when the network's densities are higher, then there is more chance that seed selection methods are able to achieve influence faster. If we look at the Fb network in Figure 6.5, its network density is 0.01 which is greater than the rest of the networks except the network Z which has the highest density of 0.14. If we compare the plots, we see that DDCBC method also works exceptionally better in most networks as compared to the rest of the methods. From Figure 6.6, we see the networks with densities ranging from 0.0001 to 0.0009. Densities of these networks are: MFb (0.0006), DHR (0.0003), DRO

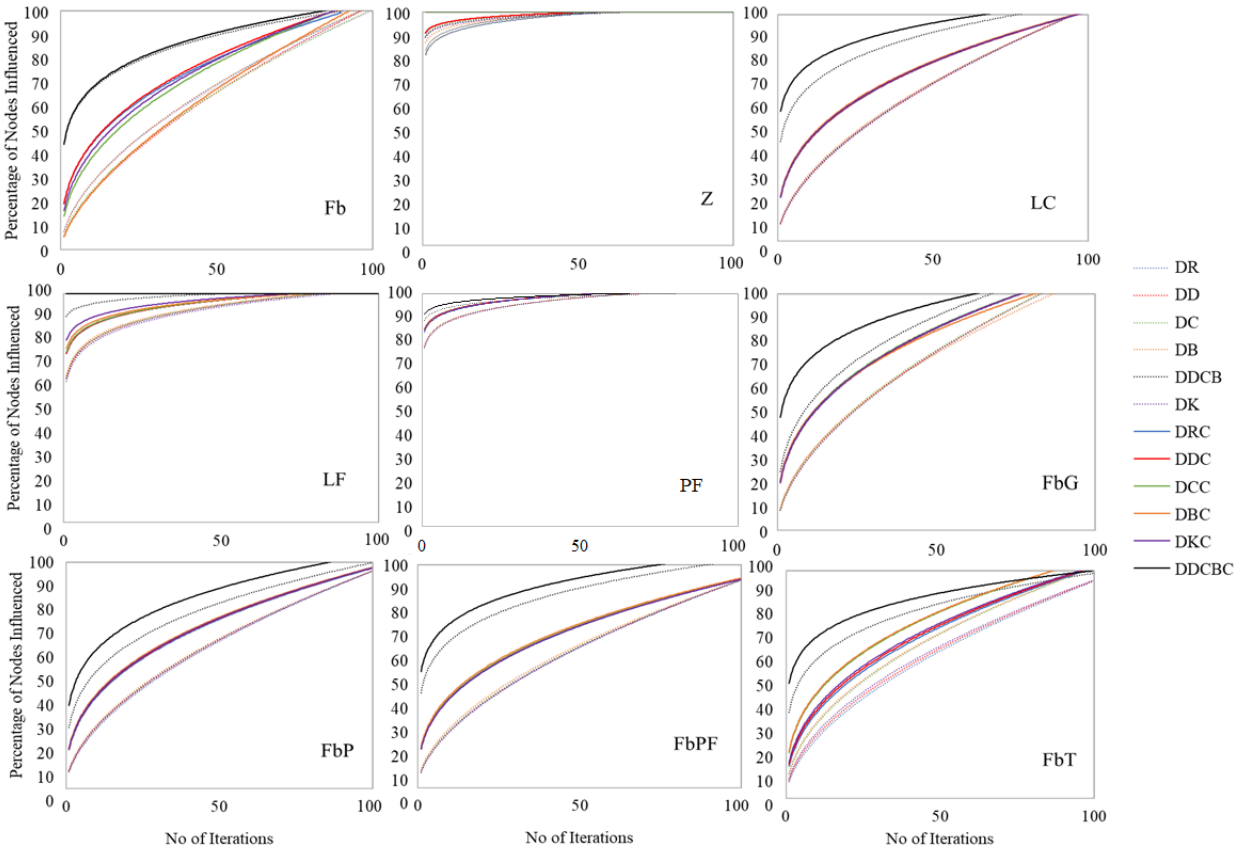


Figure 6.5 : Percentage of Number of Nodes Influenced in FB, Z, LC, LF, PF, FbG, FbP, FbPF and FbT Networks. A Comparison of all methods for 100 iterations.

(0.0001), DHU (0.0001), MG (0.0004), L (0.0009), FbAR (0.0006) and FbA (0.0009). With the lower density networks, we can see that the gain in driver community based methods is more prominent as compared to driver based methods. It means density of the network does play an important role to determine the total number of nodes influenced. From Figure 6.7, we see the networks with the lowest densities ranging from 0.000002 to 0.0001. Densities of these networks are: Youtube (0.000004), Twitter (0.00012), Diggs (0.000002) and Ego (0.00014). In these networks, we see a huge gap between DDCBC method and the rest of the methods. Which means, even in the lowest density networks, when we locally construct communities, the density tend to

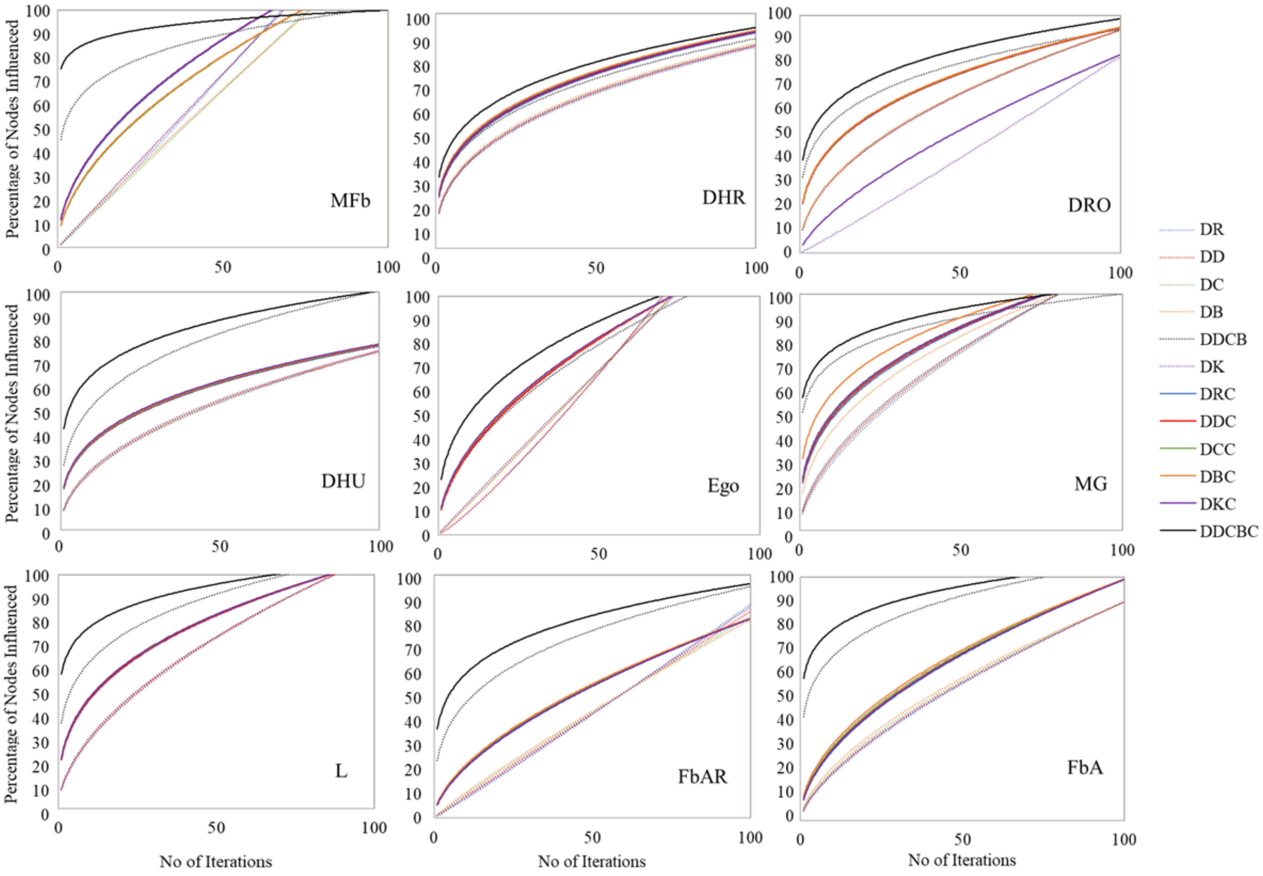


Figure 6.6 : Percentage of Number of Nodes Influenced in MFb, DHR, DRO, DHU, MG, L, FbAR and FbA Networks. A Comparison of all methods for 100 iterations.

increase as we can see from Table 6.2. Average community density of Youtube was calculated to be 0.000012±0.04, which means if we compare it to the overall network density of 0.000004, it is notably denser. That is why, even in these networks, driver-community based methods specially DDCBC method outperforms the driver based methods. This analysis justifies the use of network density as the strongest measure that is in direct relation with driver nodes, and communities in the network. However, for further deeper understanding of how the interconnections between driver nodes and nodes in the communities work, other useful measures like clustering coefficient, degree distribution, and centrality measures can be considered in future.

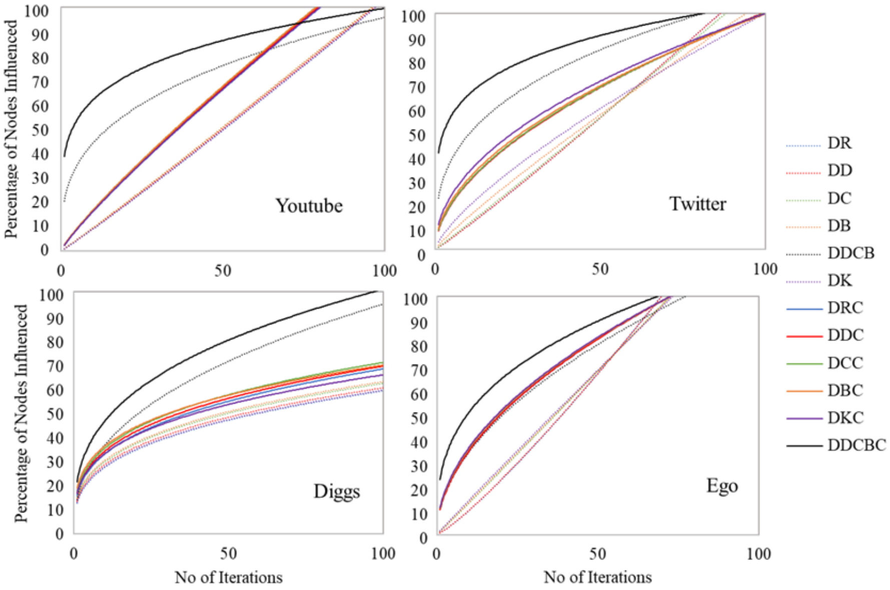


Figure 6.7 : Percentage of Number of Nodes Influenced in Youtube, Twitter, Diggs and Ego Networks. A Comparison of all methods for 100 iterations.

#### 6.3.2.2 Time Complexity and Execution Time

Table 6.3 shows the time complexity of calculating the structural measures that are used in seed selection methods. Despite driver nodes having high complexity, there is clear benefit of using driver nodes as seed selection methods based on them require less number of iterations to achieve a 100% influence over a network.

Table 6.5 shows the execution times of all seed selection methods for various sizes of seed set i.e. 1%, 5%, 15%, and 25%, when the maximum influence is reached. Driver community based methods, in comparison to the driver based methods, take less time in overall influence spread process. To identify communities in the network is a costly process but, overall analysis shows that, given that the community driver based methods i.e. DRC, DDC, DCC, DBC, DKC, and DDCBC reach 100% influence in fewer number of iterations, which ultimately results in a decrease in overall execution times.

Table 6.3 : Time Complexity of Calculating Different Measures and Methods

| Measures & Methods | Complexity |
|---|---|
| Degree (D) | $O^2$ |
| Closeness (C) | $O(N * E * d)$, where $d$ is the diameter |
| Betweenness (B) | $O(N * M + N * 2 * logN)$ |
| Kempe (K) | $O(logN)$ |
| Driver Nodes | $O(V1/2 * E)$ |
| DCB | $O^2 + O(N * E * d) + O(N * M + N * 2 * logN)$ |
| Community Detection | $O(m^2 n)$ with $m$ edges and $n$ nodes |

In Table 6.4, execution times in each of the social networks for the community driver based methods such as DRC, DDC, DCC, DBC, DKC, and DDCBC are lower than their counterparts methods DR, DD, DC, DB, DK, and DDCB. The green colour represents the lower execution times and red shows the highest execution time.

#### 6.3.2.3 How much advantage do community-level driver based seed selection methods give?

From Table 6.2, we see the percentage of gain that DDCBC has over other seed selection methods in terms of number of nodes influenced after 100 iterations when seed size is 20%. We can see from the table that DDCBC outperforms all methods, but the gain is bigger in terms of global-level driver based methods than the community-level driver based methods. We see this difference in gain mainly because of locally selected and then ranked driver nodes. Also, community creation plays an important role as, the communities are denser than the overall network. From Table 6.2 we can see that the biggest gain is achieved by DDCBC method over DK method which is 45.89% in FbA network. And the lowest gain is achieved by DDCBC method over DK method in ZKC network. The reason for lowest or lower gain is that ZKC has

Table 6.4 : Execution Times (in hours – Hrs.) of All Seed Selection Methods in Each Social Network with 1% Seed Size When 100% Influence is Reached.

| Networks | Execution Times (Hrs.) : Maximum Influence at 1% Seed Size | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | DR | DD | DC | DB | DK | DDCB | DRC | DDC | DCC | DBC | DKC | DDCBC |
| FB | 1.50 | 1.50 | 1.75 | 1.50 | 1.35 | 0.75 | 0.72 | 0.67 | 0.46 | 0.45 | 0.35 | 0.33 |
| ZKC | 0.25 | 0.22 | 0.21 | 0.21 | 0.69 | 0.16 | 0.11 | 0.15 | 0.15 | 0.15 | 0.11 | 0.05 |
| Twitter | 1.95 | 1.95 | 1.75 | 1.95 | 1.95 | 1.95 | 1.43 | 1.21 | 1.16 | 1.13 | 1.12 | 1.12 |
| Diggs | 19.01 | 18.99 | 15.92 | 16.45 | 16.25 | 16.00 | 15.55 | 15 | 14.79 | 14.69 | 13.99 | 12.98 |
| Youtube | 18.99 | 18.79 | 18.76 | 16.45 | 15.21 | 14.00 | 14.00 | 12.00 | 12.00 | 11.91 | 10.98 | 10.91 |
| Ego | 3.02 | 2.51 | 2.45 | 2.45 | 2.39 | 2.25 | 1.35 | 1.22 | 1.18 | 1.17 | 1.15 | 1.15 |
| LC | 1.50 | 1.50 | 1.50 | 1.50 | 1.11 | 0.69 | 0.69 | 0.65 | 0.44 | 0.43 | 0.33 | 0.32 |
| LF | 0.31 | 0.28 | 0.28 | 0.28 | 0.95 | 0.26 | 0.25 | 0.25 | 0.25 | 0.19 | 0.19 | 0.15 |
| PF | 0.37 | 0.35 | 0.35 | 0.31 | 0.95 | 0.29 | 0.28 | 0.27 | 0.26 | 0.21 | 0.21 | 0.18 |
| MFb | 3.02 | 2.51 | 2.45 | 2.45 | 1.95 | 2.05 | 1.43 | 1.21 | 1.16 | 1.13 | 1.12 | 1.12 |
| DHR | 2.00 | 2.00 | 2.00 | 1.51 | 0.99 | 1.51 | 1.53 | 1.31 | 1.26 | 1.23 | 1.22 | 1.22 |
| DRO | 3.02 | 2.51 | 2.45 | 2.45 | 3.94 | 2.25 | 1.43 | 1.21 | 1.16 | 1.13 | 1.12 | 1.12 |
| DHU | 3.02 | 2.51 | 2.45 | 2.45 | 2.75 | 2.25 | 1.43 | 1.21 | 1.16 | 1.13 | 1.12 | 1.12 |
| MG | 3.35 | 2.55 | 2.55 | 2.45 | 2.83 | 2.45 | 1.53 | 1.31 | 1.26 | 1.23 | 1.22 | 1.22 |
| L | 1.53 | 1.25 | 1.25 | 1.22 | 1.85 | 0.72 | 0.75 | 0.67 | 0.46 | 0.45 | 0.35 | 0.34 |
| FbAR | 3.35 | 2.55 | 2.55 | 2.45 | 1.95 | 2.05 | 1.53 | 1.31 | 1.26 | 1.23 | 1.22 | 1.22 |
| FbA | 1.91 | 1.91 | 1.87 | 1.91 | 1.11 | 1.21 | 1.43 | 1.21 | 1.16 | 1.13 | 1.12 | 1.12 |
| FbG | 1.53 | 1.25 | 1.25 | 1.22 | 1.97 | 0.72 | 0.75 | 0.67 | 0.46 | 0.45 | 0.35 | 0.34 |
| FbN | 3.02 | 2.51 | 2.45 | 2.45 | 1.85 | 2.25 | 1.43 | 1.21 | 1.16 | 1.13 | 1.12 | 1.12 |
| FbP | 1.15 | 1.12 | 0.99 | 0.99 | 1.03 | 0.95 | 0.66 | 0.59 | 0.41 | 0.39 | 0.34 | 0.33 |
| FbPF | 3.02 | 2.51 | 2.45 | 2.45 | 1.57 | 2.45 | 1.43 | 1.21 | 1.16 | 1.13 | 1.12 | 1.12 |
| FbT | 1.53 | 1.25 | 1.15 | 1.12 | 1.15 | 0.94 | 0.69 | 0.65 | 0.44 | 0.43 | 0.33 | 0.32 |

the highest network density and smallest size. In denser networks, we tend to see the less gain in DDCBC method. Which precisely can mean that, if we locally identify communities, those have denser structures as compared to the overall network. That is why community-driver based methods combined with ranking of DCB works better than the rest of the methods.

Table 6.5 : Total Execution Times (in hours – Hrs.) of All Seed Selection Methods in all the Social Networks when 100% Influence is Reached

| Execution Time (Hrs.): Maximum Influence | | | | |
|---|---|---|---|---|
| Seed Selection Methods | Seed Size | | | |
| | 1% | 5% | 15% | 25% |
| DR | 78.35 | 77.86 | 74.78 | 70.63 |
| DD | 72.52 | 70.64 | 64.13 | 60.52 |
| DC | 68.83 | 64.44 | 63.55 | 62.11 |
| DB | 66.22 | 60.21 | 59.96 | 55.63 |
| DK | 65.79 | 62.98 | 58.18 | 55.48 |
| DDCB | 58.15 | 55.65 | 51.15 | 49.71 |
| DRC | 50.40 | 37.43 | 32.45 | 28.67 |
| DDC | 45.19 | 35.54 | 30.44 | 28.32 |
| DCC | 43.20 | 39.81 | 35.75 | 30.44 |
| DBC | 42.52 | 43.65 | 40.33 | 32.75 |
| DKC | 40.18 | 38.85 | 32.17 | 25.15 |
| DDCBC | 38.90 | 26.99 | 20.83 | 16.12 |

## 6.4 Discussion and Conclusion

In this study we focused on achieving the Research Challenge RC5, which states that, "Using driver nodes identified in local network structures to maximize influence spread in social networks". An idea of bringing the methods from control and influence fields together has been proposed in this research. In fact, we played with a research dimension that is at the intersection of both fields and fulfils the objectives of many research questions from both domains. We proposed, implemented and compared a list of new and novel seed selection methods with the traditional seed selection methods from influence domain and driver seed selection methods from influence meets control field. In this work, we introduced new seed selection methods, by utilising driver nodes in communities of the networks. The new methods outperformed the old ones. This opens up an avenue in the already existing research of control methods in complex networks. Our community-driver based methods show that, we can achieve maximum influence in fewer number of iterations and with a comparatively lower seed set size. Also, if we use ranking mechanisms based upon the centrality measures combining degree, betweenness and closeness, the driver nodes selected as seed nodes perform much better in that case as compared to when we rank them on the basis of individual centrality measures.

# Chapter 7

# Conclusion and Future Work

This section describes the major conclusions and potential future work.

## 7.1 Conclusion

The main aim of this thesis is to utilise the concepts from the field of network control and apply those to effectively and efficiently spread influence in a network by using seed selection methods based on the driver nodes set.

The research presented in this thesis started from an extensive literature survey of the domains of control, controllability and influence in complex networks. This review addresses the first challenge (RC1): "Conducting a thorough study based upon the previous research to work as a foundation of the current research thesis". We described and discussed current methods to identify and rank driver nodes, as well as seed selection methods that are needed to identify a set of potential seed nodes that can spread the influence through a network. The major findings of the literature survey are that, firstly, control in complex networks is a continuously evolving area of research, and there is much to explore when it comes to finding an influential set of driver nodes. Secondly, driver node identification and ranking is a complex process, requiring resource intensive computation. Thirdly, influence is a weaker form of control. There is still a need to develop new methods, for finding an optimal set of seed nodes that can spread influence in the network efficiently and effectively. The literature review resulted in identifying the gaps in extant research, which we formulated as Research Challenges (RC). The research conducted and results presented in this thesis addressed all the Research Challenges described in Figure 1.2.

To address RC2, which focused on investigation of the correlation of global network structural measures with the number of driver nodes with the aim of understanding what network structures are easier to control, we examined several structural metrics, such as network density, number of nodes, number of edges, betweenness centrality, eigenvector centrality, closeness centrality, and their correlation with number of driver nodes. One of the main conclusions from this experimental study was that networks with high values for the global network structural measures *number of edges*, and *network density* are easier to control than others - i.e. they inversely correlate with the number of driver nodes. The correlations revealed helped us in understanding that changes in global structural measures (e.g. network density) can impact the number of driver nodes. So, we were able to identify network structures which require a minimum number of driver nodes to establish control. The network structures with dense connections and higher density, having larger number of edges in ratio to number of nodes and less number of driver nodes are easier to control.

To deepen the understanding of the correlations between network structural measures and the number of driver nodes, we also performed analysis at the local network level. The investigation focused on communities and structural measures within communities. For example, we examined communities' density and its impact on the size of the set of driver nodes. We found out that local structural measures, such as the number of communities and community densities, have definite correlations with the number of driver nodes. When the density increases, it reduces the number of driver nodes because we have smaller, more efficient set of driver nodes to control the network. Local network structure measures such as number of communities and community densities have an important role to play in determining the minimum number of driver nodes. This study attained RC3, which stated, "Inquiring about the relationship between local network structures with the number of driver nodes in the networks."

After the two initial studies, summarised above and described in detail in Chapters 3 and 4 respectively, we move on to the effectiveness of driver node based seed selection methods in spreading influence in complex networks. This line of inquiry is codified in RC4, i.e., drawing comparisons between different driver node based seed selection methods and traditional seed selection methods for generated and real social networks, and RC5, i.e., using driver nodes identified in local network structures to maximise influence spread in social networks. To achieve that we proposed new seed selection methods based on driver node sets ranked on the basis of various global and local network structural measures. We validated and analysed these seed selection methods through experiments that focused on influence spread over various synthetic and real networks. We concluded that driver node based seed selection methods outperform traditional seed selection methods in spreading influence in complex networks. We developed an environment where we can bring together traditional, driver node based and community driver node based seed selection methods to identify the most suitable seed selection method(s) for a certain network structure in the context of influence spread in the complex networks (see Chapters 3, 4, 5, and 6).

To summarise, the main contributions of the thesis are as follows:

- We identified the main research gaps from the literature review conducted at the intersection of driver node selection methods from the complex network control space and seed selection methods from the complex network influence space. This gives the foundation of the thesis in which the gap between these two domains is bridged. (Attained RC1, and RO1)

- The main contribution of Chapter 3 is the identification of network structures that engender easier control. Network structures that have higher density, a greater number of edges, and a smaller number of driver nodes are generally easier to control. (Fulfilled RC2, RQ1, and RO2)

- In Chapter 4 we examined the role that the structure of communities played in the number of driver nodes needed and found correlations between local network structural measures and number of driver nodes. The findings show (Achieved RC3, RQ2, and RO3)

- In Chapter 5, we developed and validated new seed selection methods (i.e, R, DD, DB,DC, DK and DDCB) in an attempt to bridge the gap between control and influence fields. The new methods in comparison with traditional seed selection methods were more efficient and effective, comparing on the basis of speed and reach of influence spread. (Completed RC4, RQ3, and RO4)
- The main contribution of Chapter 6 is the comparison of newly proposed and developed seed selection approaches. The results indicate that driver nodes, when identified within communities of the networks, are able to spread influence faster and to a greater number of nodes as compared to when selected from the overall network, because of the changing network structural measures of the network. Community density is denser than the overall network. So, the speed and reach of seed selection methdos based upon driver nodes in communities (DRC, DDC. DBC, DKC, and DDBCC) is higher than the seed selection methods that are based upon driver nodes identified in overall network. (Procured RC5, RQ4, and RO5)

A discussion that considers a comparative analysis of local and global seed selection methods based upon driver nodes, enables us to identify effective measures and methods that can be used to reach the maximum influence in the network faster than the previously used methods. The existing body of knowledge does not take into account the important role of driver nodes in reaching maximum influence in the network due to the cost of the resources. But, given that, we have an optimal seed set, that can reach influence faster than the other methods, kind of overcome the limitation posed by the calculation time of the driver nodes based seed set.

Our initial research indicates that the structural measures such as network density can play a huge role in identifying the networks that are easier to control. Given that, we can identify the networks that will be controlled or influenced fully even before starting the process, which gives us an advantage to use the appropriate method(s) based upon the structural measures of the networks. Our research strengthens the already established premise that real networks are harder to influence or control. But, if we select an optimal seed set before going through the process it cuts down the execution time of the overall influence spread process. This gives us a strong indication to focus on finding more influential nodes rather than building more complicated spreading models.

By saying all that, we do not mean to eliminate the need to construct more efficient methods to identify driver nodes. Because, if we can somehow cut down the cost of identification methods, we can improve the overall process's execution time.

## 7.2 Future Work

In this thesis, many pivotal research problems have been studied and thoroughly examined. Although, many questions have been answered, the space is vast and there are many new research avenues that can be explored in the future. Possible future research directions are outlined as follows:

- Leveraging the obtained findings to estimate the number of driver nodes needed to control a network only by estimating the main network characteristics.

- Influence Models and Control Methods are used to identify driver nodes within communities of the network and target seed selection methods at the community rather than at the whole network level, which brings out the favourable outcomes, when it comes to speed or reach of influence spread process in the overall network. In future work, more local network structural measures can be analysed to explore relationships between number of driver nodes and those measures.

- Work remains to be done in the context of ranking of driver nodes by using other algorithms than the ones used in the thesis, for example Page Rank [19], Leader Rank [131], Cluster Rank [29] and K-Shell Decomposition [128].

- New methods such as Preferential Matching [238] can be used (instead of MDS) to identify driver nodes to improve the efficiency of the seed selection process.
- Another avenue for exploration is the effects of differing influence models, such as the Independent Cascade Model [43].

To summarise, the main achievements of this work include, but are not limited to, proposing and evaluating new seed selection methods in order to spread influence in different types of complex networks (i.e., synthetic and real). We study the correlation between different network structural measures (global as well as local) and the number of driver nodes, with the idea to control the overall network by using the minimum set of driver nodes. Influence, being the soft form of control, can also benefit from these nodes. The main experimental studies explained in Chapters 3, 4, 5, and 6 achieve effective and efficient influence spread in the complex networks by utilising newly proposed and developed seed selection methods.

There are many measures used to describe network structure, network density is a considerably basic and a very important indicator. Gnyawali and Madhavan [65] suggest that the number of network connections can greatly affect the communication and cooperation between individuals, so network density is an important factor affecting individual behaviours and effects. More importantly, network density has an important impact on the information diffusion process of the network. For example, in [150], researchers focused on the epidemic spreading and vaccination strategies in an urban environment, their results show that the network density plays a critical role on the information diffusion of both SIS and SIR epidemic processes. Also, changing the number of edges has important influence on interconnection between the components of the system.

Based upon this understanding this paper focused on analysing network density as a base measure for influence spread in the networks along with other centrality measures. Although, it poses a limitation on the analysis that has been done, it also provides an insight into the network structures that are easier to control given that we are able to change the density and number of connections in the network. For future, carrying out the influence spread analysis can also be correlated with slightly more robust measure such as the clustering coefficient, because it focuses on density, strength of relationships, as well as the correlation between nodes.

# Bibliography

[1] Réka Albert and Albert-László Barabási. Statistical mechanics of complex networks. *Reviews of modern physics*, 74(1):47, 2002.

[2] Khaled M Alzoubi, Peng-Jun Wan, and Ophir Frieder. New distributed algorithm for connected dominating set in wireless ad hoc networks. In *Proceedings of the 35th Annual Hawaii International Conference on System Sciences*, pages 3849–3855. IEEE, 2002.

[3] John Perry Ballantine and Arthur Rudolph Jerbert. Distance from a line, or plane, to a poin. *The American Mathematical Monthly*, 59(4):242–243, 1952.

[4] Suman Banerjee, Mamata Jenamani, and Dilip Kumar Pratihar. A survey on influence maximization in a social network. *Knowledge and Information Systems*, 62(9):3417–3455, 2020.

[5] A Barabsi and Rka Albert. Emerge of scaling in random networks. *Science*, 286:509512, 1999.

[6] Alain Barrat, Marc Barthelemy, and Alessandro Vespignani. *Dynamical processes on complex networks*. Cambridge university press, 2008.

[7] Murray A Beauchamp. An improved index of centrality. *Behavioral science*, 10(2):161–163, 1965.

[8] Stuart Bennett. *A history of control engineering, 1930-1955*. IET, 1993.

[9] Tim Berners-Lee, Dimitri Dimitroyannis, A John Mallinckrodt, and Susan McKay. World wide web. *Computers in Physics*, 8(3):298–299, 1994.

[10] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008, 2008.

[11] Phillip Bonacich. Factoring and weighting approaches to status scores and clique identification. *Journal of mathematical sociology*, 2(1):113–120, 1972.

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

[13] Phillip Bonacich and Paulette Lloyd. Eigenvector-like measures of centrality for asymmetric relations. *Social networks*, 23(3):191–201, 2001.

[14] Ronald V Book et al. Michael r. garey and david s. johnson, computers and intractability: A guide to the theory of *np*-completeness. *Bulletin (New Series) of the American Mathematical Society*, 3(2):898–904, 1980.

[15] Christian Borgs, Michael Brautbar, Jennifer Chayes, and Brendan Lucier. Maximizing social influence in nearly optimal time. In *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*, pages 946–957. SIAM, 2014.

[16] Allan Borodin, Yuval Filmus, and Joel Oren. Threshold models for competitive influence in social networks. In *Internet and Network Economics: 6th International Workshop, WINE 2010, Stanford, CA, USA, December 13-17, 2010. Proceedings 6*, pages 539–550. Springer, 2010.

[17] Simon Bourigault, Sylvain Lamprier, and Patrick Gallinari. Representation learning for information diffusion through social networks: an embedded cascade model. In *Proceedings of the Ninth ACM international conference on Web Search and Data Mining*, pages 573–582, 2016.

[18] Auguste Bravais. *Analyse mathématique sur les probabilités des erreurs de situation d'un point*. Impr. Royale, 1844.

[19] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual web search engine. *Computer networks and ISDN systems*, 30(1-7):107–117, 1998.

[20] Simon R Broadbent and John M Hammersley. Percolation processes: I. crystals and mazes. In *Mathematical proceedings of the Cambridge philosophical society*, volume 53, pages 629–641. Cambridge University Press, 1957.

[21] Piotr Bródka, Katarzyna Musial, and Jaroslaw Jankowski. Interacting spreading processes in multilayer networks: a systematic review. *IEEE Access*, 8:10316–10341, 2020.

[22] Anna D Broido and Aaron Clauset. Scale-free networks are rare. *Nature communications*, 10(1):1–10, 2019.

[23] Derek Bruff. The assignment problem and the hungarian method. *Notes for Math*, 20(29-47):5, 2005.

[24] Doina Bucur and Giovanni Iacca. Influence maximization in social networks with genetic algorithms. In *European conference on the applications of evolutionary computation*, pages 379–392. Springer, 2016.

[25] Ceren Budak, Divyakant Agrawal, and Amr El Abbadi. Limiting the spread of misinformation in social networks. In *Proceedings of the 20th international conference on World wide web*, pages 665–674, 2011.

[26] Sergey V Buldyrev, Roni Parshani, Gerald Paul, H Eugene Stanley, and Shlomo Havlin. Catastrophic cascade of failures in interdependent networks. *Nature*, 464(7291):1025–1028, 2010.

[27] Daniel A Burbano-L, Giovanni Russo, and Mario di Bernardo. Pinning controllability of complex stochastic networks. *IFAC-PapersOnLine*, 50(1):8327–8332, 2017.

[28] Dave Chaffey and Fiona Ellis-Chadwick. *Digital marketing: strategy, implementation & practice.* Pearson uk, 2019.

[29] Duan-Bing Chen, Hui Gao, Linyuan Lü, and Tao Zhou. Identifying influential nodes in large-scale directed networks: the role of clustering. *PloS one*, 8(10):e77455, 2013.

[30] Duanbing Chen, Linyuan Lü, Ming-Sheng Shang, Yi-Cheng Zhang, and Tao Zhou. Identifying influential nodes in complex networks. *Physica a: Statistical mechanics and its applications*, 391(4):1777–1787, 2012.

[31] Wei Chen, Wei Lu, and Ning Zhang. Time-critical influence maximization in social networks with time-delayed diffusion process. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 26, pages 591–598, 2012.

[32] Yi-Cheng Chen, Wen-Yuan Zhu, Wen-Chih Peng, Wang-Chien Lee, and Suh-Yin Lee. Cim: community-based influence maximization in social networks. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 5(2):1–31, 2014.

[33] Yu-Zhong Chen, Zi-Gang Huang, and Ying-Cheng Lai. Controlling extreme events on complex networks. *Scientific reports*, 4:6121, 2014.

[34] Suqi Cheng, Huawei Shen, Junming Huang, Guoqing Zhang, and Xueqi Cheng. Staticgreedy: solving the scalability-accuracy dilemma in influence maximization. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pages 509–518, 2013.

[35] Peter Clifford and Aidan Sudbury. A model for spatial conflict. *Biometrika*, 60(3):581–588, 1973.

[36] Edith Cohen, Daniel Delling, Thomas Pajor, and Renato F Werneck. Sketch-based influence maximization and computation: Scaling up with guarantees. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pages 629–638, 2014.

[37] Thomas H Cormen, Charles E Leiserson, Ronald L Rivest, and Clifford Stein. *Introduction to algorithms.* MIT press, 2009.

[38] Gianlorenzo D'Angelo, Lorenzo Severini, and Yllka Velaj. Influence maximization in the independent cascade model. In *ICTCS*, pages 269–274, 2016.

[39] G LUENBERGER David. Introduction to dynamic systems: Theory, models and applications, 1979.

[40] Danilo Delpini, Stefano Battiston, Massimo Riccaboni, Giampaolo Gabbi, Fabio Pammolli, and Guido Caldarelli. Evolution of controllability in interbank networks. *Scientific reports*, 3:1626, 2013.

[41] EA Dinic. Algorithm for solution of a problem of maximum flow in a network with power estimation, soviet math. doll. 11 (5), 1277-1280,(1970). *English translation by RF. Rinehart*, 1970.

[42] Norman R Draper and Harry Smith. Applied regression analysis . new york: John willey & sons, 1966.

[43] Wenjing Duan, Bin Gu, and Andrew B Whinston. Informational cascades and software adoption on the internet: an empirical investigation. *MIS quarterly*, pages 23–48, 2009.

[44] GuB DuanW. Whinstonab. *Informational Cascades and Software Adoption on the Internet*, 33(1):23, 2009.

[45] Paul Erdős and Alfréd Rényi. On the evolution of random graphs. *Publ. Math. Inst. Hung. Acad. Sci*, 5(1):17–60, 1960.

[46] Paul Erdös, Alfréd Rényi, et al. On random graphs. *Publicationes mathematicae*, 6(26):290–297, 1959.

[47] Paul Erdös and George Szekeres. A combinatorial problem in geometry. *Compositio mathematica*, 2:463–470, 1935.

[48] Fredrik Erlandsson, Piotr Bródka, and Anton Borg. Seed selection for information cascade in multilayer networks. In *International Conference on Complex Networks and their Applications*, pages 426–436. Springer, 2017.

[49] Fredrik Erlandsson, Piotr Bródka, Anton Borg, and Henric Johnson. Finding influential users in social media using association rule learning. *Entropy*, 18(5):164, 2016.

[50] Brian D Fath, Ursula M Scharler, Robert E Ulanowicz, and Bruce Hannon. Ecological network analysis: network construction. *Ecological modelling*, 208(1):49–55, 2007.

[51] Shanshan Feng, Xuefeng Chen, Gao Cong, Yifeng Zeng, Yeow Meng Chee, and Yanping Xiang. Influence maximization with novelty decay in social networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 28, 2014.

[52] Stephen E Fienberg. A brief history of statistical models for network analysis and open challenges. *Journal of Computational and Graphical Statistics*, 21(4):825–839, 2012.

[53] LR Ford and DR Fulkerson. Maximal flow through a network. *Canadian Journal of Mathematics*, 1956.

[54] James H Fowler and Nicholas A Christakis. Dynamic spread of happiness in a large social network: longitudinal analysis over 20 years in the framingham heart study. *Bmj*, 337, 2008.

[55] LC Freeman. Centrality in affiliation networks. *Social Networks*, 1:215–39, 1979.

[56] Linton C Freeman. A set of measures of centrality based on betweenness. *Sociometry*, pages 35–41, 1977.

[57] Zhong-Ke Gao, Peng-Cheng Fang, Mei-Shuang Ding, and Ning-De Jin. Multivariate weighted complex network analysis for characterizing nonlinear dynamic behavior in two-phase flow. *Experimental Thermal and Fluid Science*, 60:157–164, 2015.

[58] Zhong-Ke Gao and Ning-De Jin. A directed weighted complex network for characterizing chaotic dynamics from time series. *Nonlinear Analysis: Real World Applications*, 13(2):947–952, 2012.

[59] Zhong-Ke Gao, Yu-Xuan Yang, Peng-Cheng Fang, Ning-De Jin, Cheng-Yi Xia, and Li-Dan Hu. Multi-frequency complex network from time series for uncovering oil-water flow structure. *Scientific reports*, 5(1):1–7, 2015.

[60] Diego Garlaschelli and Maria I Loffredo. Patterns of link reciprocity in directed networks. *Physical review letters*, 93(26):268701, 2004.

[61] Alexander J Gates and Luis M Rocha. Control of complex networks requires both structure and dynamics. *Scientific reports*, 6:24456, 2016.

[62] Edgar N Gilbert. Random graphs. *The Annals of Mathematical Statistics*, 30(4):1141–1144, 1959.

[63] C Lee Giles, Kurt D Bollacker, and Steve Lawrence. Citeseer: An automatic citation indexing system. In *ACM DL*, pages 89–98, 1998.

[64] Michelle Girvan and Mark EJ Newman. Community structure in social and biological networks. *Proceedings of the national academy of sciences*, 99(12):7821–7826, 2002.

[65] Devi R Gnyawali and Ravindranath Madhavan. Cooperative networks and competitive dynamics: A structural embeddedness perspective. *Academy of Management review*, 26(3):431–445, 2001.

[66] Anna Goldenberg, Alice X Zheng, Stephen E Fienberg, and Edoardo M Airoldi. *A survey of statistical network models*. Now Publishers Inc, 2010.

[67] Jacob Goldenberg, Barak Libai, and Eitan Muller. Talk of the network: A complex systems look at the underlying process of word-of-mouth. *Marketing letters*, 12(3):211–223, 2001.

[68] Amit Goyal, Wei Lu, and Laks VS Lakshmanan. Celf++ optimizing the greedy algorithm for influence maximization in social networks. In *Proceedings of the 20th international conference companion on World wide web*, pages 47–48, 2011.

[69] Mark Granovetter. Threshold models of collective behavior. *American journal of sociology*, 83(6):1420–1443, 1978.

[70] Guibing Guo, Jie Zhang, Daniel Thalmann, and Neil Yorke-Smith. Etaf: An extended trust antecedents framework for trust prediction. In *2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2014)*, pages 540–547. IEEE, 2014.

[71] Quantong Guo, Yanjun Lei, Xin Jiang, Yifang Ma, Guanying Huo, and Zhiming Zheng. Epidemic spreading with activity-driven awareness diffusion on multiplex network. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 26(4):043110, 2016.

[72] Ruocheng Guo, Jundong Li, and Huan Liu. Learning individual causal effects from networked observational data. In *Proceedings of the 13th International Conference on Web Search and Data Mining*, pages 232–240, 2020.

[73] Wei-Feng Guo, Shao-Wu Zhang, Ze-Gang Wei, Tao Zeng, Fei Liu, Jingsong Zhang, Fang-Xiang Wu, and Luonan Chen. Constrained target controllability of complex networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2017(6):063402, 2017.

[74] Wei-Feng Guo, Shao-Wu Zhang, Tao Zeng, Yan Li, Jianxi Gao, and Luonan Chen. A novel structure-based control method for analyzing nonlinear dynamics in biological networks. *bioRxiv*, page 503565, 2018.

[75] Hoshin V Gupta, Harald Kling, Koray K Yilmaz, and Guillermo F Martinez. Decomposition of the mean squared error and nse performance criteria: Implications for improving hydrological modelling. *Journal of hydrology*, 377(1-2):80–91, 2009.

[76] Reza Haghighi and HamidReza Namazi. Algorithm for identifying minimum driver nodes based on structural controllability. *Mathematical Problems in Engineering*, 2015, 2015.

[77] SL HAKIMI. Onrealizability ofa set ofintegers as degrees ofthe vertices ofa lineargraph–landii. *J. Soc. Indust. Appl. Math*, 10:496–506, 1962.

[78] Henry Hamburger. Individuals and aggregates: Micromotives and macrobehavior. thomas c. schelling. norton, new york, 1978. 252 pp. cloth, 12.9s;,paper, 3.95. fels lectures on public policy analysis. *Science*, 205(4401):37–38, 1979.

[79] Robert A Hanneman and Mark Riddle. Introduction to social network methods, 2005.

[80] Frank Harary. Graph theory. Technical report, MICHIGAN UNIV ANN ARBOR DEPT OF MATHEMATICS, 1969.

[81] Frank Harary. Recent results on generalized ramsey theory for graphs. In *Graph Theory and Applications*, pages 125–138. Springer, 1972.

[82] Malo LJ Hautus. Controllability and observability conditions of linear autonomous systems. In *Indagationes Mathematicae (Proceedings)*, pages 443–448, 1969.

[83] Teresa W Haynes, Stephen Hedetniemi, and Peter Slater. *Fundamentals of domination in graphs*. CRC press, 1998.

[84] Tad Hogg and Kristina Lerman. Social dynamics of digg. *EPJ Data Science*, 1(1):1–26, 2012.

[85] Richard A Holley and Thomas M Liggett. Ergodic theorems for weakly interacting infinite systems and the voter model. *The annals of probability*, pages 643–663, 1975.

[86] Sture Holm. A simple sequentially rejective multiple test procedure. *Scandinavian journal of statistics*, pages 65–70, 1979.

[87] Petter Holme. Modern temporal network theory: a colloquium. *The European Physical Journal B*, 88(9):234, 2015.

[88] Petter Holme and Jari Saramäki. Temporal networks. *Physics reports*, 519(3):97–125, 2012.

[89] John E Hopcroft and Richard M Karp. An nˆ5/2 algorithm for maximum matchings in bipartite graphs. *SIAM Journal on computing*, 2(4):225–231, 1973.

[90] Shigeyuki Hosoe. Determination of generic dimensions of controllable subspaces and its application. *Research Reports of Automatic Control Laboratory, Faculty of Engineering, Nagoya University,(28): p78*, 83, 1981.

[91] Wenpin Hou, Peiying Ruan, Wai-Ki Ching, and Tatsuya Akutsu. On the number of driver nodes for controlling a boolean network when the targets are restricted to attractors. *Journal of theoretical biology*, 463:1–11, 2019.

[92] Shaobin Huang, Tianyang Lv, Xizhe Zhang, Yange Yang, Weimin Zheng, and Chao Wen. Identifying node role in social network based on multiple indicators. *PloS one*, 9(8):e103733, 2014.

[93] Jarosław Jankowski, Marcin Waniek, Aamena Alshamsi, Piotr Bródka, and Radosław Michalski. Strategic distribution of seeds to support diffusion in complex networks. *PloS one*, 13(10):e0205130, 2018.

[94] Svante Janson, Tomasz Łuczak, Tatyana Turova, and Thomas Vallier. Bootstrap percolation on the random graph $g_{n,p}$. *The Annals of Applied Probability*, 22(5):1989–2047, 2012.

[95] Christopher I Jarvis, Amy Gimma, Kevin van Zandvoort, Kerry LM Wong, and W John Edmunds. The impact of local and national restrictions in response to covid-19 on social contacts in england: a longitudinal natural experiment. *BMC medicine*, 19(1):1–12, 2021.

[96] Tao Jia and Albert-László Barabási. Control capacity and a random sampling method in exploring controllability of complex networks. *Scientific reports*, 3:2354, 2013.

[97] Thomas Kailath. *Linear systems*, volume 156. Prentice-Hall Englewood Cliffs, NJ, 1980.

[98] Rudolf Emil Kalman. Mathematical description of linear dynamical systems. *Journal of the Society for Industrial and Applied Mathematics, Series A: Control*, 1(2):152–192, 1963.

[99] Wilfred Kaplan. *Operational methods for linear systems*, volume 4. Addison-Wesley Reading, Mass., 1962.

[100] Amir Hassani Karbasi and Reza Ebrahimi Atani. Application of dominating sets in wireless sensor networks. *International Journal of Security and Its Applications*, 7(4):185–202, 2013.

[101] Farzaneh Kazemzadeh, Ali Asghar Safaei, and Mitra Mirzarezaee. Influence maximization in social networks using effective community detection. *Physica A: Statistical Mechanics and its Applications*, 598:127314, 2022.

[102] Laura L Kelleher and Margaret B Cozzens. Dominating sets in social network graphs. *Mathematical Social Sciences*, 16(3):267–279, 1988.

[103] David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 137–146, 2003.

[104] William Ogilvy Kermack and Anderson G McKendrick. A contribution to the mathematical theory of epidemics. *Proceedings of the royal society of london. Series A, Containing papers of a mathematical and physical character*, 115(772):700–721, 1927.

[105] Mohammad Reza Keyvanpour, Mehrnoush Barani Shirzad, and Maryam Ghaderi. Ad-c: a new node anomaly detection based on community detection in social networks. *International Journal of Electronic Business*, 15(3):199–222, 2020.

[106] Maksim Kitsak, Lazaros K Gallos, Shlomo Havlin, Fredrik Liljeros, Lev Muchnik, H Eugene Stanley, and Hernán A Makse. Identification of influential spreaders in complex networks. *Nature physics*, 6(11):888–893, 2010.

[107] Iordanis Koutsopoulos and Maria Halkidi. Efficient and fair item coverage in recommender systems. In *2018 IEEE 16th Intl Conf on Dependable, Autonomic and Secure Computing, 16th Intl Conf on Pervasive Intelligence and Computing, 4th Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech)*, pages 912–918. IEEE, 2018.

[108] Peter Krause, DP Boyle, and Frank Bäse. Comparison of different efficiency criteria for hydrological model assessment. *Advances in geosciences*, 5:89–97, 2005.

[109] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

[110] Harold W Kuhn. Variants of the hungarian method for assignment problems. *Naval research logistics quarterly*, 3(4):253–258, 1956.

[111] Sanjay Kumar, Sanidhya Chaudhary, Saksham Kumar, and Raj Kumar Yadav. Node classification in complex networks using network embedding techniques. In *2020 5th International Conference on Communication and Electronics Systems (ICCES)*, pages 369–374. IEEE, 2020.

[112] Jérôme Kunegis. Konect: the koblenz network collection. In *Proceedings of the 22nd international conference on world wide web*, pages 1343–1350, 2013.

[113] Renaud Lambiotte and Naoki Masuda. *A guide to temporal networks*, volume 4. World Scientific, 2016.

[114] Andrea Lancichinetti, Santo Fortunato, and János Kertész. Detecting the overlapping and hierarchical community structure in complex networks. *New journal of physics*, 11(3):033015, 2009.

[115] Jure Leskovec, Jon Kleinberg, and Christos Faloutsos. Graphs over time: densification laws, shrinking diameters and possible explanations. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pages 177–187, 2005.

[116] Jure Leskovec and Andrej Krevl. SNAP Datasets: Stanford large network dataset collection. http://snap.stanford.edu/data, June 2014.

[117] Aming Li, Sean P Cornelius, Y-Y Liu, Long Wang, and A-L Barabási. The fundamental advantages of temporal networks. *Science*, 358(6366):1042–1046, 2017.

[118] Aming Li and Yang-Yu Liu. Controlling network dynamics. *Advances in Complex Systems*, 22(07n08):1950021, 2019.

[119] Chaoyi Li and Yangsen Zhang. A personalized recommendation algorithm based on large-scale real micro-blog data. *Neural Computing and Applications*, 32(15):11245–11252, 2020.

[120] Ching-Tai Lin. Structural controllability. *IEEE Transactions on Automatic Control*, 19(3):201–208, 1974.

[121] Bo Liu, Tianguang Chu, Long Wang, and Guangming Xie. Controllability of a leader–follower dynamic network with switching topology. *IEEE Transactions on Automatic Control*, 53(4):1009–1013, 2008.

[122] Qi Liu, Biao Xiang, Nicholas Jing Yuan, Enhong Chen, Hui Xiong, Yi Zheng, and Yu Yang. An influence propagation view of pagerank. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 11(3):1–30, 2017.

[123] Weiyi Liu, Kun Yue, Hong Wu, Jin Li, Donghua Liu, and Duanping Tang. Containment of competitive influence spread in social networks. *Knowledge-Based Systems*, 109:266–275, 2016.

[124] Yang-Yu Liu and Albert-Laszló Barabási. Control principles of complex networks. *arXiv preprint arXiv:1508.05384*, 2015.

[125] Yang-Yu Liu and Albert-László Barabási. Control principles of complex systems. *Reviews of Modern Physics*, 88(3):035006, 2016.

[126] Yang-Yu Liu, Jean-Jacques Slotine, and Albert-László Barabási. Controllability of complex networks. *nature*, 473(7346):167, 2011.

[127] Yang-Yu Liu, Jean-Jacques Slotine, and Albert-László Barabási. Control centrality and hierarchical structure in complex networks. *Plos one*, 7(9):e44459, 2012.

[128] Ying Liu, Ming Tang, Tao Zhou, and Younghae Do. Improving the accuracy of the k-shell method by removing redundant links: From a perspective of spreading dynamics. *Scientific reports*, 5(1):1–11, 2015.

[129] Anna Lombardi and Michael Hörnquist. Controllability analysis of networks. *Physical Review E*, 75(5):056110, 2007.

[130] Linyuan Lü, Duanbing Chen, Xiao-Long Ren, Qian-Ming Zhang, Yi-Cheng Zhang, and Tao Zhou. Vital nodes identification in complex networks. *Physics Reports*, 650:1–63, 2016.

[131] Linyuan Lü, Yi-Cheng Zhang, Chi Ho Yeung, and Tao Zhou. Leaders in social networks, the delicious case. *PloS one*, 6(6):e21202, 2011.

[132] Pengli Lu and Chen Dong. Ranking the spreading influence of nodes in complex networks based on mixing degree centrality and local structure. *International Journal of Modern Physics B*, 33(32):1950395, 2019.

[133] R Duncan Luce and Albert D Perry. A method of matrix analysis of group structure. *Psychometrika*, 14(2):95–116, 1949.

[134] Jiawei Luo and Yi Qi. Identification of essential proteins based on a new combination of local interaction density and protein complexes. *PloS one*, 10(6):e0131418, 2015.

[135] Francesco Martino and Andrea Spoto. Social network analysis: A brief theoretical review and further perspectives in the study of information technology. *PsychNology J.*, 4(1):53–86, 2006.

[136] Naoki Masuda and Petter Holme. Detecting sequences of system states in temporal networks. *Scientific reports*, 9(1):1–11, 2019.

[137] J Clerk Maxwell. On governors. *Proceedings of the Royal Society of London*, 16:270–283, 1867.

[138] Julian J McAuley and Jure Leskovec. Learning to discover social circles in ego networks. In *NIPS*, volume 2012, pages 548–56. Citeseer, 2012.

[139] Giulia Menichetti, Luca Dall'Asta, and Ginestra Bianconi. Network controllability is determined by the density of low in-degree and out-degree nodes. *Physical review letters*, 113(7):078701, 2014.

[140] Rezvan Mohamadi-Baghmolaei, Niloofar Mozafari, and Ali Hamzeh. Trust based latency aware influence maximization in social networks. *Engineering Applications of Artificial Intelligence*, 41:195–206, 2015.

[141] Azadeh Mohammadi, Mohamad Saraee, and Abdolreza Mirzaei. Time-sensitive influence maximization in social networks. *Journal of Information Science*, 41(6):765–778, 2015.

[142] F Molnár, Sameet Sreenivasan, Boleslaw K Szymanski, and Gyorgy Korniss. Minimum dominating sets in scale-free network ensembles. *Scientific reports*, 3(1):1–10, 2013.

[143] David M Morens, Jefferey K Taubenberger, JK Taubenberger, et al. Influenza: the mother of all pandemics. *Emerging Infectious Diseases*, 12(1):15–22, 1918.

[144] Flaviano Morone and Hernán A Makse. Influence maximization in complex networks through optimal percolation. *Nature*, 524(7563):65–68, 2015.

[145] Katarzyna Musiał and Przemysław Kazienko. Social networks on the internet. *World Wide Web*, 16(1):31–72, 2013.

[146] Katarzyna Musiał, Przemysław Kazienko, and Piotr Brodka. User position measures in social networks. In *Proc. of 3rd workshop on social network mining and analysis*, pages 1–9, 2009.

[147] Jose C Nacher and Tatsuya Akutsu. Dominating scale-free networks with variable scaling exponent: heterogeneous networks are not difficult to control. *New Journal of Physics*, 14(7):073005, 2012.

[148] Jose C Nacher and Tatsuya Akutsu. Structural controllability of unidirectional bipartite networks. *Scientific reports*, 3:1647, 2013.

[149] Jose C Nacher and Tatsuya Akutsu. Structurally robust control of complex networks. *Physical Review E*, 91(1):012826, 2015.

[150] Matthieu Nadini, Lorenzo Zino, Alessandro Rizzo, and Maurizio Porfiri. A multi-agent model to study epidemic spreading and vaccination strategies in an urban-like environment. *Applied Network Science*, 5(1):1–30, 2020.

[151] Ramasuri Narayanam and Yadati Narahari. A shapley value-based approach to discover influential nodes in social networks. *IEEE Transactions on Automation Science and Engineering*, 8(1):130–147, 2010.

[152] Tamás Nepusz and Tamás Vicsek. Controlling edge dynamics in complex networks. *Nature Physics*, 8(7):568, 2012.

[153] Mark Newman. *Networks.* Oxford university press, 2018.

[154] Mark Ed Newman, Albert-László Ed Barabási, and Duncan J Watts. *The structure and dynamics of networks.* Princeton university press, 2006.

[155] Mark EJ Newman, Stephanie Forrest, and Justin Balthrop. Email networks and the spread of computer viruses. *Physical Review E*, 66(3):035101, 2002.

[156] Mark EJ Newman and Elizabeth A Leicht. Mixture models and exploratory analysis in networks. *Proceedings of the National Academy of Sciences*, 104(23):9564–9569, 2007.

[157] Hung T Nguyen, My T Thai, and Thang N Dinh. Stop-and-stare: Optimal sampling algorithms for viral marketing in billion-scale networks. In *Proceedings of the 2016 international conference on management of data*, pages 695–710, 2016.

[158] Hung T Nguyen, My T Thai, and Thang N Dinh. A billion-scale approximation algorithm for maximizing benefit in viral marketing. *IEEE/ACM Transactions On Networking*, 25(4):2419–2429, 2017.

[159] Huy Nguyen and Rong Zheng. On budgeted influence maximization in social networks. *IEEE Journal on Selected Areas in Communications*, 31(6):1084–1094, 2013.

[160] Athanasios N Nikolakopoulos, Dimitris Berberidis, George Karypis, and Georgios B Giannakis. Personalized diffusions for top-n recommendation. In *Proceedings of the 13th ACM Conference on Recommender Systems*, pages 260–268, 2019.

[161] Australian Institute of Health and Welfare. *The first year of COVID-19 in Australia: direct and indirect health effects.* Australian Institute of Health and Welfare, 2021.

[162] J-P Onnela, Jari Saramäki, Jorkki Hyvönen, György Szabó, David Lazer, Kimmo Kaski, János Kertész, and A-L Barabási. Structure and tie strengths in mobile communication networks. *Proceedings of the national academy of sciences*, 104(18):7332–7336, 2007.

[163] Alan V Oppenheim. Alan s. willsky. *Signals and Systems Second Edition Prentice Hall, New Jersey*, 7458, 1997.

[164] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab, 1999.

[165] Fabio Pasqualetti, Sandro Zampieri, and Francesco Bullo. Controllability metrics, limitations and algorithms for complex networks. *IEEE Transactions on Control of Network Systems*, 1(1):40–52, 2014.

[166] Romualdo Pastor-Satorras and Alessandro Vespignani. Epidemic spreading in scale-free networks. *Physical review letters*, 86(14):3200, 2001.

[167] Judea Pearl. *Causality*. Cambridge university press, 2009.

[168] Paul Pilkington and Sanjay Kinra. Effectiveness of speed cameras in preventing road traffic collisions and related casualties: systematic review. *Bmj*, 330(7487):331–334, 2005.

[169] Alfred Reginald Radcliffe-Brown. On social structure. *The Journal of the Royal Anthropological Institute of Great Britain and Ireland*, 70(1):1–12, 1940.

[170] Usha Nandini Raghavan, Réka Albert, and Soundar Kumara. Near linear time algorithm to detect community structures in large-scale networks. *Physical review E*, 76(3):036106, 2007.

[171] Manoel Horta Ribeiro, Pedro H Calais, Yuri A Santos, Virgílio AF Almeida, and Wagner Meira Jr. Characterizing and detecting hateful users on twitter. In *Twelfth international AAAI conference on web and social media*, 2018.

[172] EM Rogers. Diffusion of innovations. hohenheim, 2010.

[173] Ryan Rossi and Nesreen Ahmed. The network data repository with interactive graph analytics and visualization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29, 2015.

[174] Martin Rosvall and Carl T Bergstrom. Maps of random walks on complex networks reveal community structure. *Proceedings of the national academy of sciences*, 105(4):1118–1123, 2008.

[175] Benedek Rozemberczki, Carl Allen, and Rik Sarkar. Multi-scale attributed node embedding. *Journal of Complex Networks*, 9(2):cnab014, 2021.

[176] Benedek Rozemberczki, Ryan Davies, Rik Sarkar, and Charles Sutton. Gemsec: Graph embedding with self clustering. In *Proceedings of the 2019 IEEE/ACM international conference on advances in social networks analysis and mining*, pages 65–72, 2019.

[177] Benedek Rozemberczki and Rik Sarkar. Characteristic functions on graphs: Birds of a feather, from statistical descriptors to parametric models. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 1325–1334, 2020.

[178] Justin Ruths and Derek Ruths. Control profiles of complex networks. *Science*, 343(6177):1373–1376, 2014.

[179] Gert Sabidussi. The centrality index of a graph. *Psychometrika*, 31(4):581–603, 1966.

[180] Abida Sadaf, Luke Mathieson, Piotr Bródka, and Katarzyna Musial. Maximising influence spread in complex networks by utilising community-based driver nodes as seeds. *arXiv preprint arXiv:2212.11611*, 2022.

[181] Abida Sadaf, Luke Mathieson, Piotr Bródka, and Katarzyna Musial. A bridge between influence models and control methods[manuscript submitted for publication]. *Applied Network Science*, 2023.

[182] Abida Sadaf, Luke Mathieson, and Katarzyna Musial. An insight into network structure measures and number of driver nodes. In *Proceedings of the 2021 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 471–478, 2021.

[183] Firas Saidi, Zouheir Trabelsi, and Henda Ben Ghazela. A novel approach for terrorist sub-communities detection based on constrained evidential clustering. In *2018 12th International Conference on Research Challenges in Information Science (RCIS)*, pages 1–8. IEEE, 2018.

[184] Kazumi Saito, Ryohei Nakano, and Masahiro Kimura. Prediction of information diffusion probabilities for independent cascade model. In *International conference on knowledge-based and intelligent information and engineering systems*, pages 67–75. Springer, 2008.

[185] Gerard Salton. Automatic text processing: The transformation, analysis, and retrieval of. *Reading: Addison-Wesley*, 169, 1989.

[186] Laura A Sanchis. Experimental analysis of heuristic algorithms for the dominating set problem. *Algorithmica*, 33(1):3–18, 2002.

[187] K Sathiyakumari and MS Vijaya. Community detection based on girvan newman algorithm and link analysis of social media. In *Annual Convention of the Computer Society of India*, pages 223–234. Springer, 2016.

[188] Stephen B Seidman. Network structure and minimum degree. *Social networks*, 5(3):269–287, 1983.

[189] Joakim Skarding, Matthew Hellmich, Bogdan Gabrys, and Katarzyna Musial. A robust comparative analysis of graph neural networks on dynamic link prediction. *IEEE Access*, 10:64146–64160, 2022.

[190] David A Smith and Douglas R White. Structure and dynamics of the global economy: network analysis of international trade 1965–1980. *Social forces*, 70(4):857–893, 1992.

[191] Sandeep Soni, Shawn Ling Ramirez, and Jacob Joseph Eisenstein. Detecting social influence in event cascades by comparing discriminative rankers. In *The 2019 ACM SIGKDD Workshop on Causal Discovery*, pages 78–99. PMLR, 2019.

[192] Francesco Sorrentino. Effects of the network structural properties on its controllability. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 17(3):033101, 2007.

[193] Ivan Stojmenovic, Mahtab Seddigh, and Jovisa Zunic. Dominating sets and neighbor elimination-based broadcasting algorithms in wireless networks. *IEEE Transactions on parallel and distributed systems*, 13(1):14–25, 2002.

[194] Steven H Strogatz. Exploring complex networks. *nature*, 410(6825):268, 2001.

[195] Danny Sullivan. What is google pagerank? a guide for searchers & webmasters. *Search engine land*, 2007.

[196] Peng Gang Sun and Xiaoke Ma. Understanding the controllability of complex networks from the microcosmic to the macrocosmic. *New Journal of Physics*, 19(1):013022, 2017.

[197] Ashis Talukder, Md Golam Rabiul Alam, Nguyen H Tran, Dusit Niyato, Gwan Hoon Park, and Choong Seon Hong. Threshold estimation models for

linear threshold-based influential user mining in social networks. *IEEE Access*, 7(1):105, 2019.

[198] Youze Tang, Yanchen Shi, and Xiaokui Xiao. Influence maximization in near-linear time: A martingale approach. In *Proceedings of the 2015 ACM SIGMOD international conference on management of data*, pages 1539–1554, 2015.

[199] Youze Tang, Xiaokui Xiao, and Yanchen Shi. Influence maximization: Near-optimal time complexity meets practical efficiency. In *Proceedings of the 2014 ACM SIGMOD international conference on Management of data*, pages 75–86, 2014.

[200] Juliana Tolles and ThaiBinh Luong. Modeling epidemics with compartmental models. *Jama*, 323(24):2515–2516, 2020.

[201] Christopher Tran and Elena Zheleva. Heterogeneous peer effects in the linear threshold model. *arXiv preprint arXiv:2201.11242*, 2022.

[202] Giacomo Vaccario, Luca Verginer, and Frank Schweitzer. The mobility network of scientists: Analyzing temporal correlations in scientific careers. *Applied Network Science*, 5(1):1–14, 2020.

[203] Margarita Vitoropoulou, Konstantinos Tsitseklis, Vasileios Karyotis, and Symeon Papavassiliou. Cover: An information diffusion aware approach for efficient recommendations under user coverage constraints. *IEEE Transactions on Computational Social Systems*, 8(4):894–905, 2021.

[204] Akanda Wahid-Ul-Ashraf, Marcin Budka, and Katarzyna Musial. Netsim–the framework for complex network generator. *Procedia Computer Science*, 126:547–556, 2018.

[205] Bingbo Wang, Lin Gao, and Yong Gao. Control range: a controllability-based

index for node significance in directed networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2012(04):P04011, 2012.

[206] Bingbo Wang, Lin Gao, Qingfang Zhang, Aimin Li, Yue Deng, and Xingli Guo. Diversified control paths: A significant way disease genes perturb the human regulatory network. *PloS one*, 10(8):e0135491, 2015.

[207] Dong Wang, Jiexun Li, Kaiquan Xu, and Yizhen Wu. Sentiment community detection: exploring sentiments and relationships in social networks. *Electronic Commerce Research*, 17(1):103–132, 2017.

[208] Jing Wang and Ioannis Ch Paschalidis. Botnet detection based on anomaly and community detection. *IEEE Transactions on Control of Network Systems*, 4(2):392–404, 2016.

[209] Le-Zhi Wang, Yu-Zhong Chen, Wen-Xu Wang, and Ying-Cheng Lai. Physical controllability of complex networks. *Scientific reports*, 7:40198, 2017.

[210] Xiao Fan Wang and Guanrong Chen. Complex networks: small-world, scale-free and beyond. *IEEE circuits and systems magazine*, 3(1):6–20, 2003.

[211] Stanley Wasserman, Katherine Faust, et al. *Social network analysis: Methods and applications*, volume 8. Cambridge university press, 1994.

[212] Stanley Wasserman, Garry Robins, and Douglas Steinley. Statistical models for networks: A brief review of some recent research. In *ICML Workshop on Statistical Network Analysis*, pages 45–56. Springer, 2006.

[213] D Watts and S Strogatz. An undirected, unweighted network representing the topology of the western states power grid of the united states. *Nature*, 393:440–442, 1998.

[214] Duncan Watts and Steven Strogatz. The small world problem. *Collective Dynamics of Small-World Networks*, 393:440–442, 1998.

[215] Duncan J Watts and Peter Sheridan Dodds. Influentials, networks, and public opinion formation. *Journal of consumer research*, 34(4):441–458, 2007.

[216] Duncan J Watts and Steven H Strogatz. Collective dynamics of 'small-world'networks. *Nature*, 393(6684):440, 1998.

[217] Bo Wei, Jie Liu, Daijun Wei, Cai Gao, and Yong Deng. Weighted k-shell decomposition for complex networks based on potential edge weights. *Physica A: Statistical Mechanics and its Applications*, 420:277–283, 2015.

[218] Karsten Weihe. Covering trains by stations or the power of data reduction. *Proceedings of Algorithms and Experiments, ALEX*, pages 1–8, 1998.

[219] Jianshu Weng, Ee-Peng Lim, Jing Jiang, and Qi He. Twitterrank: finding topic-sensitive influential twitterers. In *Proceedings of the third ACM international conference on Web search and data mining*, pages 261–270, 2010.

[220] Andrew J Whalen, Sean N Brennan, Timothy D Sauer, and Steven J Schiff. Observability and controllability of nonlinear networks: The role of symmetry. *Physical Review X*, 5(1):011005, 2015.

[221] S William. The probable error of a mean. *Biometrika*, 6(1):1–25, 1908.

[222] Stefan Wuchty. Controllability in protein interaction networks. *Proceedings of the National Academy of Sciences*, 111(19):7156–7160, 2014.

[223] Xiaowei Xu, Nurcan Yuruk, Zhidan Feng, and Thomas AJ Schweiger. Scan: a structural clustering algorithm for networks. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 824–833, 2007.

[224] Gang Yan, Jie Ren, Ying-Cheng Lai, Choy-Heng Lai, and Baowen Li. Controlling complex networks: How much energy is needed? *Physical review letters*, 108(21):218703, 2012.

[225] Jaewon Yang and Jure Leskovec. Defining and evaluating network communities based on ground-truth. *Knowledge and Information Systems*, 42(1):181–213, 2015.

[226] Zhao Yang, René Algesheimer, and Claudio J Tessone. A comparative analysis of community detection algorithms on artificial networks. *Scientific reports*, 6(1):1–18, 2016.

[227] Shunyu Yao, Neng Fan, and Jie Hu. Modeling the spread of infectious diseases through influence maximization. *Optimization letters*, 16(5):1563–1586, 2022.

[228] Wei Ye, Linfei Zhou, Dominik Mautz, Claudia Plant, and Christian Böhm. Learning from labeled and unlabeled vertices in networks. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1265–1274, 2017.

[229] Paraskevas Yiapanis, Demian Rosas-Ham, Gavin Brown, and Mikel Luján. Optimizing software runtime systems for speculative parallelization. *ACM Transactions on Architecture and Code Optimization (TACO)*, 9(4):1–27, 2013.

[230] Lingchong You, Apirak Hoonlor, and John Yin. Modeling biological systems using dynetica—a simulator of dynamic networks. *Bioinformatics*, 19(3):435–436, 2003.

[231] Zhengzhong Yuan, Chen Zhao, Zengru Di, Wen-Xu Wang, and Ying-Cheng Lai. Exact controllability of complex networks. *Nature communications*, 4:2447, 2013.

[232] Zhengzhong Yuan, Chen Zhao, Wen-Xu Wang, Zengru Di, and Ying-Cheng Lai. Exact controllability of multiplex networks. *New Journal of Physics*, 16(10):103036, 2014.

[233] Wayne W Zachary. An information flow model for conflict and fission in small groups. *Journal of anthropological research*, 33(4):452–473, 1977.

[234] Jorge Gomez Tejeda Zañudo, Gang Yang, and Réka Albert. Structure-based control of complex networks with nonlinear dynamics. *Proceedings of the National Academy of Sciences*, 114(28):7234–7239, 2017.

[235] Ahmad Zareie and Rizos Sakellariou. Influence maximization in social networks: A survey of behaviour-aware methods. *arXiv preprint arXiv:2108.03438*, 2021.

[236] Jian-Xiong Zhang, Duan-Bing Chen, Qiang Dong, and Zhi-Dan Zhao. Identifying a set of influential spreaders in complex networks. *Scientific reports*, 6:27823, 2016.

[237] Shihua Zhang, Rui-Sheng Wang, and Xiang-Sun Zhang. Uncovering fuzzy community structure in complex networks. *Physical Review E*, 76(4):046103, 2007.

[238] Xizhe Zhang, Tianyang Lv, XueYing Yang, and Bin Zhang. Structural controllability of complex networks based on preferential matching. *PloS one*, 9(11):e112039, 2014.

[239] Yan Zhang, Antonios Garas, and Frank Schweitzer. Control contribution identifies top driver nodes in complex networks. *arXiv preprint arXiv:1906.04663*, 2019.

[240] Haijun Zhou and Zhong-can Ou-Yang. Maximum matching on random graphs. *arXiv preprint cond-mat/0309348*, 2003.

[241] Jiang Zhu, Bai Wang, Bin Wu, and Weiyu Zhang. Emotional community detection in social network. *IEICE Transactions on Information and Systems*, 100(10):2515–2525, 2017.