

CONSTRAINED SEED SELECTION FOR INFLUENCE MAXIMIZATION: A COMPARATIVE STUDY OF HYBRID CENTRALITY MEASURES

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

MASTER OF TECHNOLOGY

in

Computer Science and Engineering

by

Rushikesh Patil

(Roll No. 2K24/CSE/25)

Under the Supervision of

Dr. Prashant Giridhar Shambharkar

**Department of Computer Science and Engineering
Delhi Technological University**



Department of Computer Science and Engineering

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Shahbad Daultpur, Main Bawana Road, Delhi-110042, India

May, 2026



DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi-110042

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to my supervisor, Dr. Prashant Giridhar Shambharkar, for the continuous support, guidance, and patience throughout the course of this work. The discussions during regular meetings shaped both the technical direction of the thesis and my own understanding of the broader research area.

I am also grateful to the faculty members of the Department of Computer Science and Engineering for the courses and conversations that built the foundation on which this work stands. I thank my fellow students for the many informal discussions that helped me clarify ideas during difficult points, and the technical staff for keeping the computing infrastructure available across the long evaluation runs.

Finally, I thank my family for their unwavering support and encouragement during the course of this degree.

Place: Delhi

Rushikesh Patil

Date:



DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

CANDIDATE'S DECLARATION

I hereby declare that the work presented in this thesis entitled **“Constrained Seed Selection for Influence Maximization: A Comparative Study of Hybrid Centrality Measures”** is an authentic record of my own work carried out under the supervision of Dr. Prashant Giridhar Shambharkar. The matter embodied in this thesis has not been submitted by me for the award of any other degree of this or any other University/Institution. Wherever contributions of others are involved, every effort has been made to indicate this clearly with due reference to the literature and acknowledgement of collaborative research and discussions.

Candidate's Signature

This is to certify that the student has incorporated all the corrections suggested by the examiners in the thesis and the statement made by the candidate is correct to the best of our knowledge.

Signature of Supervisor

Signature of External Examiner



DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi-110042

CERTIFICATE BY THE SUPERVISOR

This is to certify that the thesis entitled **“Constrained Seed Selection for Influence Maximization: A Comparative Study of Hybrid Centrality Measures”** submitted by **Rushikesh Patil** (Roll No.: 2K24/CSE/25) to the Department of Computer Science and Engineering, Delhi Technological University, in partial fulfilment of the requirements for the award of the degree of Master of Technology, is a bona fide record of work carried out by him under my supervision. The contents of this thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

(Dr. Prashant Giridhar Shambharkar)

(Assistant Professor)

(Department of Computer Science and Engineering)

(Delhi Technological University)

Date:

Constrained Seed Selection for Influence Maximization: A Comparative Study of Hybrid Centrality Measures

Rushikesh Patil

ABSTRACT

The problem of identifying a compact set of influential nodes in a complex network underlies several applications such as epidemic control, viral marketing, rumour suppression, and infrastructure protection. A common approach is to rank nodes by some structural centrality and pick the top- k as seeds. The well-known shortcoming is that the highest ranked nodes tend to cluster in the dense core of real networks, leading to overlapping influence regions and to a non-redundant reach that falls short of what the same budget could otherwise achieve.

A recent proposal known as Core-Flow Centrality (CF) addresses this in two ways. The first is a weighted blend of normalised M-shell and normalised Betweenness Centrality, with a reported optimum at $\alpha = 0.6$. The second is an Improved Core-Flow (ICF) procedure that walks down the CF-ranked list and rejects any candidate whose shortest-path distance to an already-selected seed is less than two hops. The original work claimed that CF outperforms structural baselines in Kendall's τ against SIR ground-truth, that the $\alpha = 0.6$ peak is consistent across networks, and that CF combined with ICF achieves the largest influence spread on a standard five-network benchmark.

This thesis presents an end-to-end reproduction of those claims on the same five datasets (NetScience, C. Elegans, Jazz, PowerGrid, and AstroPh) using the paper's stated protocol. The implementation includes a from-scratch Zeng–Zhang M-shell decomposition, an efficient ICF routine based on truncated breadth-first search, a deterministically seeded SIR simulator with repeated random runs, and a sparse adjacency eigensolver for the infection-rate calibration. Two findings emerge. Under faithful reproduction, CF does not outperform pure M-shell in Kendall's τ on any of the five networks; the α -sweep is monotonic in α with the peak at $\alpha = 1.0$, and a normalisation sensitivity analysis spanning min-max, z-score, and rank schemes confirms the result is not an artefact of normalisation choice. By contrast, the distance constraint of the ICF model is faithfully replicated, in that each centrality derives the benefit of two hops of separation, resulting in an improvement of influence spread at $p = 5\%$ for PowerGrid between 19% (for M-shell) and 56% (for θ).

As a consequence, the only useful aspect of the initial proposal is the distance constraint, not the hybrid centrality. M-shell combined with ICF emerges as a simpler and equally effective alternative to CF combined with ICF.

Keywords: influence maximization, complex networks, centrality measures, M-shell decomposition, betweenness centrality, constrained seed selection, SIR diffusion, Kendall's tau, reproduction study.

TABLE OF CONTENTS

Acknowledgments	ii
Candidate's Declaration	iii
Certificate by the Supervisor	iv
Abstract	v
Table of Contents	viii
List of Tables	ix
List of Figures	x
List of Abbreviations	xi
1 Introduction	1
1.1 Motivation	1
1.2 The overlap problem	1
1.3 The proposal under study	2
1.4 Research questions	3
1.5 Contributions	3
1.6 Scope	4
1.7 Structure	4
2 Related Work	6
2.1 Influence maximization and greedy approximation	6
2.2 Classical centrality measures	7
2.3 Coreness-based approaches: k-shell and M-shell	8
2.4 Hybrid centralities	8
2.5 Distance-aware seed selection	9
2.6 The SIR diffusion model	10
2.7 Research gap	10
3 Methodology	11
3.1 Notation	11
3.2 Centrality measures	11

3.2.1	Betweenness centrality	11
3.2.2	M-shell decomposition	12
3.2.3	Core-Flow Centrality	12
3.2.4	Two extra baselines: θ and $x^{(mdd)}$	12
3.3	Improved Core-Flow seed selection	13
3.4	SIR diffusion ground truth	13
3.5	Evaluation metrics	14
3.6	Normalisation sensitivity	14
3.7	Reproducibility	14
4	Implementation	16
4.1	Stack and project layout	16
4.2	Dataset acquisition	17
4.3	M-shell decomposition	17
4.4	Centrality computation	17
4.5	Improved Core-Flow selection	18
4.6	Epidemic threshold and SIR	19
4.7	Ground-truth computation	19
4.8	Pipeline orchestration	19
4.9	Implementation correctness	20
4.10	Total runtime and resource use	20
5	Experimental Results	22
5.1	Dataset characteristics	22
5.2	Ranking accuracy: Kendall's tau	22
5.3	The alpha-parameter sweep	23
5.4	Influence spread by seed ratio	24
5.5	Normalisation sensitivity	26
5.6	Summary of headline findings	28
6	Discussion	29
6.1	The rationale behind these results	29
6.2	The constraint in ICF as the real contribution	30
6.3	When the hybrid premise might still work	30
6.4	Practical recommendation	31
6.5	Limitations of the reproduction	31
6.6	Comparison with related reproduction studies	32
7	Conclusion, Future Scope and Social Impact	33
7.1	Summary of contributions	33
7.2	Limitations	33
7.3	Future work	34
7.4	Social impact	35

7.5 Closing remarks	35
References	37

LIST OF TABLES

5.1	Structural characteristics of the five benchmark networks.	22
5.2	Kendall's τ between each method's ranking and the SIR ground-truth mean spread, per network. Higher is better.	23
5.3	Summary of the four research questions and the experimental verdict.	28

LIST OF FIGURES

5.1	Kendall's τ values across the four benchmark networks for the four ranking methods. M-shell ties or leads on three networks; CF never strictly exceeds M-shell.	23
5.2	Kendall's τ as a function of the CF blending parameter α on all five networks. The dependence is monotonically increasing on every network. The peak claimed by the original paper ($\alpha = 0.6$, dotted line) does not appear.	24
5.3	Influence Spread σ_{total}/N on the PowerGrid network as a function of seeding ratio p for all five centralities, with ICF post-processing. M-shell combined with ICF (red squares) closely follows CF with ICF (black triangles) in all scenarios, and slightly outperforms it at every seed ratio.	25
5.4	The ICF effect on PowerGrid at $p = 5\%$ ($k = 247$). Every centrality benefits from the two-hop distance constraint. Gain ranges from +19% (M-shell) to +56% (θ). The constraint is not specific to CF.	26
5.5	Best CF Kendall's τ under three normalisation rules vs pure M-shell (red line). On four of five networks the CF bars sit exactly at the pure M-shell value, because the best α under every normalisation is $\alpha = 1.0$. On NetScience the CF advantage is at most 0.002, within the noise of stochastic SIR.	27

LIST OF ABBREVIATIONS

BC	Betweenness Centrality
BFS	Breadth-First Search
CF	Core-Flow Centrality
CSE	Computer Science and Engineering
CSV	Comma-Separated Values
DTU	Delhi Technological University
IC	Independent Cascade (diffusion model)
ICF	Improved Core-Flow
IM	Influence Maximization
LCC	Largest Connected Component
MDD	Mixed Degree Decomposition (M-shell)
SIR	Susceptible–Infected–Recovered (diffusion model)

CHAPTER 1

INTRODUCTION

1.1 Motivation

Networks are how we describe systems in which the connections matter at least as much as the parts themselves. Power stations are joined by transmission lines, neurons are joined by synapses, scientists by co-authorship, and people by various social ties. Looking at such systems as graphs has turned out to be unusually fruitful, because a great many questions that arise in different domains turn out to share a structural core: which units, if activated, removed, or perturbed, have the largest effect on the whole?

This question is the heart of *influence maximization*. The most familiar formulation, due to Kempe, Kleinberg, and Tardos [1], asks us to pick a budget-limited seed set whose activation spreads to the largest expected fraction of the network under a given diffusion process. Their work established that the problem is NP-hard but admits a $(1 - 1/e)$ approximation through a greedy procedure, provided the diffusion model is submodular. In practice the greedy procedure is too expensive for graphs above a few thousand nodes, because each step requires repeated stochastic simulations of the diffusion process. This has motivated a long line of heuristic alternatives that rank nodes by some structural property and take the top- k as the seed set.

Unfortunately, the heuristics story is not a simple one. There is no one centrality measure which works best on all kinds of networks. The degree centrality considers local connectedness but is agnostic towards global flow, while closeness centrality considers distance while ignoring the depth of structure, and betweenness centrality is able to detect brokers but is topology-sensitive [2]. K-shell decomposition and its improved version M-shell identify how far the node is embedded into the dense core but are unable to discriminate nodes that are in the same shell [3][4]. All of these measures have their own shortcomings, and a lot of research has been conducted recently on hybrid centrality measures.

1.2 The overlap problem

In each and every study presented here, there lies a particular mode of failure that defines the entire field, and this is the mode of failure from which this thesis begins. In situations where multiple seeds are required, a mode of failure occurs that involves ranking. This involves the high-ranking nodes concentrating around the dense nucleus

of the graph. Picking them all as seeds wastes the budget, because their respective spreading neighbourhoods overlap heavily. However, the optimal non-redundant reach can be better accomplished by a more evenly distributed selection process. This has been illustrated by Liu et al. [5] using k-shell ranking, where nodes that have the same shell value create “core-like groups,” members of which cannot be substituted as seeds. A similar conclusion was reached by Pei et al. [6] from real social-media traces: the top-degree and top-core nodes were not always the best spreaders, and were often clustered together in a way that wasted seeding effort.

Two broad strategies have emerged to address the overlap problem. The first is to design *hybrid centralities* that combine complementary structural and flow-based signals, on the theory that a more discriminating ranking will already separate seeds that pure coreness or pure betweenness would conflate [7][8]. The second is to enforce *distance-aware seed selection* explicitly: after ranking, require that any two selected seeds lie at least some minimum number of hops apart, so that their respective neighbourhoods cannot be entirely overlapping by construction [9].

A natural question is whether either strategy is sufficient on its own, or whether the two genuinely need to be combined. The work that motivates the present thesis argues for the combined approach. We set out to test that argument rigorously.

1.3 The proposal under study

The conference paper this thesis investigates [10] introduces two related ideas. The first is Core-Flow Centrality (CF), defined as a parameterised linear combination of normalised M-shell and normalised Betweenness:

$$CF(v) = \alpha \cdot \hat{k}_{md}(v) + (1 - \alpha) \cdot \hat{BC}(v),$$

with the paper reporting an optimal blending parameter of $\alpha = 0.6$ across the networks it studies. The second is Improved Core-Flow (ICF), a greedy seed-selection procedure that walks down the CF-ranked list and admits a candidate node to the seed set only if its shortest-path distance to every previously selected seed is at least md hops, with $md = 2$ as the default.

The paper makes three concrete and testable claims. CF outperforms structural baselines (Betweenness, M-shell, θ , and $x^{(md)}$) in Kendall’s rank correlation against SIR ground-truth. The $\alpha = 0.6$ peak is consistent across networks. CF combined with ICF produces the largest non-overlapping influence spread across seed ratios from 1% to 10% on the same five-network benchmark.

These claims are concrete and testable. Reproducing them faithfully makes it possible to examine *which* of the two proposed mechanisms is actually doing the work. If the gains come from the CF hybrid, then applying ICF to a simpler centrality would not help. If the gains come from the ICF constraint, the hybrid is unnecessary and the simpler M-shell ranking with the same constraint should perform just as well. That distinction has practical consequences for anyone implementing influence maximization

in production: a simpler centrality with the same constraint is cheaper to compute and easier to audit.

1.4 Research questions

RQ1. Does Core-Flow Centrality, implemented exactly as specified in the original paper, produce a higher Kendall’s τ against the SIR ground-truth ranking than the four baselines it is compared against?

RQ2. Does the Improved Core-Flow procedure improve influence spread across centralities — including the baselines — or only when paired specifically with CF?

RQ3. Is the reported optimum $\alpha = 0.6$ reproducible across the five benchmark networks, and is the result sensitive to the normalisation applied to BC and M-shell before they are blended?

RQ4. If the headline claims do not hold under faithful reproduction, what is the simplest reliable seed-selection strategy among the candidates studied, and what does that suggest about which component of the original proposal is empirically valuable?

1.5 Contributions

This thesis makes three contributions in pursuit of those questions.

First, a complete and modular Python implementation of the CF and ICF algorithms. Several pieces of the pipeline required careful attention. The M-shell decomposition is built from scratch following the iterative pruning rule of Zeng and Zhang [4], since prior reference implementations are not generally available. The ICF distance check employs breadth-first single-source search with truncation to depth $md - 1$, which ensures the cost remains bounded for each candidate while avoiding the quadratic cost involved in computing all-pair shortest paths in larger networks. The SIR simulator uses an explicit `numpy.random.default_rng` seeded deterministically per run, so every reported number is reproducible to the bit. The epidemic threshold β_{th} is obtained from a sparse eigensolver rather than a dense one, which matters because the largest graph in the benchmark (AstroPh, 18 772 nodes) would otherwise consume on the order of three gigabytes of memory with a dense adjacency matrix.

Second, an end-to-end empirical evaluation on the same five networks the original paper uses. Ranking accuracy is measured by Kendall’s τ between each method’s score and a per-node SIR ground truth computed from 100 single-seed simulations. Influence spread is measured for seed ratios $p \in \{1\%, 2\%, \dots, 10\%\}$ under both top- k selection and ICF-constrained selection. A parameter sweep over $\alpha \in \{0.0, 0.1, \dots, 1.0\}$ on every network exposes the shape of the dependence. A normalisation sensitivity analysis, original to this work and not part of the conference paper, re-runs the α -sweep under two additional normalisation schemes (z-score and rank percentile) to test whether

the absence of an $\alpha = 0.6$ peak is an artefact of the min-max normalisation the paper implies, or a more general property of the CF formulation on these networks.

Third, a diagnosis and its practical implication. With respect to the faithful replication, the key claim about CF being better than M-shell in terms of ranking accuracy does not hold on any of the five networks under study. However, ICF can be replicated easily, with each centrality being positively affected by using the two-hop distance constraint, and gaining an influence-spread advantage between 19% and 56% at $p = 5\%$ for PowerGrid depending on the initial centrality. The author therefore proposes to use the combination of M-shell with ICF rather than CF with ICF.

1.6 Scope

This study deals exclusively with undirected, unweighted, and static graphs. The SIR diffusion model serves as the sole means of propagating contagion, with the infection rate taking on the value of $1.5\beta_{th}$ for the current investigation, where $\beta_{th} = 1/\lambda_{\max}(A)$. The five graph datasets used in the current study consist of exactly the same five datasets mentioned in the reference, namely the NetScience, C. Elegans, Jazz, PowerGrid and AstroPh datasets. There are two minor details that need mentioning at the very outset. For the C. Elegans dataset, the current implementation makes use of the version consisting of 297 nodes, while the paper refers to a 453-node dataset which likely pertains to a different connectome release. There are 198 110 undirected edges in the AstroPh dataset while the reference mentions 396 160, which is the doubled count obtained when each undirected edge is recorded in both directions. Neither discrepancy affects the validity of the comparison, since all methods are evaluated on the same graphs under the same protocol.

This thesis does not propose any new centrality measure, but is a replication-and-analysis study, and its merit lies in determining which aspects of a previous proposal carry empirical value.

1.7 Structure

Chapter 2 provides an overview of the related literature on classical centrality measures, hybrid centralities, distance-informed seed selection methods, the SIR diffusion model, and the targeted gaps that this project is designed to fill. Chapter 3 outlines the methodology, giving all five centrality measures being compared, the description of the ICF algorithm with the distance requirement, and the evaluation approach. Chapter 4 outlines the implementation process, covering dataset collection, algorithm-specific optimisations for processing the biggest graph, and other aspects of replicability. The experimental findings are outlined in five parts in Chapter 5: characteristics of the datasets, ranking accuracy according to Kendall's τ measure, influence spread for different proportions of seed nodes, the α parameter sweep, and the normalisation analysis. Chapter 6 analyses implications of the results for the initial idea and for hybrid central-

ity approaches in general, and provides practical advice for influencer-selection tasks. Chapter 7 concludes with contributions and limitations, social impact, and potential avenues for future research.

CHAPTER 2

RELATED WORK

The current chapter highlights the existing literature which the thesis draws upon. Section 2.1 briefly touches upon the influence maximization problem and the role of greedy approximation within it. Section 2.2 outlines the traditional centrality measures used as baselines in the empirical analysis. Section 2.3 examines in more detail k -shell and M -shell decomposition, since the M -shell number constitutes the structural component of CF and the most relevant comparative framework for the empirical results obtained. Section 2.4 surveys hybrid centralities, including the specific proposal under study. Section 2.5 reviews distance-aware seed-selection strategies, of which ICF is one instance. Section 2.6 describes the SIR diffusion model used as the ground truth in the experiments. Section 2.7 identifies the research gap this thesis fills.

2.1 Influence maximization and greedy approximation

The influence maximization problem in its modern form was stated by Kempe, Kleinberg, and Tardos [1]. Given a graph $G = (V, E)$, a budget k , and a stochastic diffusion model that defines, for each subset $S \subseteq V$, an expected number of activated nodes $\sigma(S)$, the task is to find the k -node set S that maximises $\sigma(S)$. The authors proved that for two common diffusion models, Independent Cascade and Linear Threshold, the function σ is monotone and submodular, which guarantees that the greedy algorithm achieves a $(1 - 1/e) \approx 0.632$ approximation of the optimum. Recent surveys by Li et al. [11] and Banerjee et al. [12] provide comprehensive coverage of the two decades of follow-up work.

The result is striking in theory and limiting in practice. Each of these greedy steps involves calculating the marginal gain $\sigma(S \cup \{v\}) - \sigma(S)$ for every candidate v , which amounts to running multiple simulations. For large graphs containing thousands of nodes, this greedy approach is simply too costly to be feasible. Subsequent work focused on speeding up the simulation step, as done by the CELF algorithm of Leskovec et al. [13], which uses the property of submodularity to eliminate most candidates after the initial pass, and Chen et al. [14], which proposed an approximation scheme based on reachable-set heuristics that does not require any simulation. These still could not overcome the fundamental $O(k \cdot n)$ nature of the problem.

The bottleneck of solving for k -optimal nodes in the first place is the practical reason that influence maximization in real systems is dominated by simpler heuristics.

The typical strategy is to sort nodes in descending order of their structural centrality measures, extract the k nodes with the highest ranks as seeds, and forgo diffusion estimation altogether. The effectiveness of the seed set depends on two factors: (a) the extent to which the centrality can distinguish between effective spreaders and ineffective ones, and (b) the interaction between the sorting and the network topology when multiple seeds are selected. Real-world networks exhibit structural characteristics such as the small-world property of high clustering with short path lengths [15], scale-free degree distributions [16], and modular community organisation [17], all of which are relevant here.

2.2 Classical centrality measures

The four centrality measures discussed below serve as benchmarks for this thesis. The comprehensive review carried out by Lü et al. [18] gives us a unifying treatment of these and many related measures; we give the definition, the rationale, and the limitations of each below.

Degree centrality $C_D(v) = \deg(v)/(n-1)$ counts direct neighbours. It is the most economical centrality measure to compute and has a reputation for being surprisingly robust in scale-free networks, which are dominated by nodes of high degree. The weakness of this measure is that two nodes may have equal degrees but be situated in very different structural locations within the graph — one peripheral and one core — and degree alone cannot distinguish them.

Closeness centrality measures the mean shortest distance from a node to any other node: $C_C(v) = (n-1)/\sum_{u \neq v} d(u, v)$. This reflects the overall reachability of a node rather than local connectivity, but it relies on the computation of all-pairs shortest paths, which is $O(n \cdot m)$ in unweighted networks and becomes prohibitive beyond tens of thousands of nodes.

Betweenness centrality (BC), as defined by Freeman [2], considers the proportion of shortest paths passing through each node:

$$BC(v) = \sum_{s \neq v \neq t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}},$$

where σ_{st} is the number of shortest paths between s and t and $\sigma_{st}(v)$ is the number of those passing through v . The computation of BC via Brandes' algorithm [19] takes $O(nm)$ running time for unweighted networks and remains the default method in NetworkX and similar packages. Two characteristics make BC distinctive. First, it detects brokers and bridges — the nodes responsible for mediating flows between otherwise disconnected regions of a network — which makes it a natural complement to coreness-based measures. Second, its empirical distribution on real networks is heavily skewed: a small number of nodes carry very large values while the vast majority lie close to zero. This skew has consequences for any normalisation scheme that combines BC linearly with another centrality, as we shall see in Chapter 5.

Eigenvector centrality and its PageRank variant assign importance recursively: a node is important if it is connected to important nodes. The definition is elegant but the measure has known pathologies on networks with isolated dense clusters, and we do not include it as a baseline because the original conference paper does not.

2.3 Coreness-based approaches: k-shell and M-shell

The k-shell decomposition was introduced as a way to partition the nodes of a graph into hierarchical layers based on connectivity density. Iteratively remove all nodes of degree ≤ 1 ; the removed nodes form the 1-shell. From the remaining graph, iteratively remove all nodes of degree ≤ 2 ; the removed nodes form the 2-shell. And so on. The shell number of a node is a robust indicator of how deeply that node is embedded in the dense core.

Kitsak et al. [3] demonstrated that the k-shell value is a better predictor of single-spreader influence under SIR diffusion than degree centrality, particularly for networks with strong core-periphery structure. The intuition is straightforward: a node deep in the core has many redundant paths to the rest of the network, and the diffusion process will reach the periphery from any such starting point with high probability.

The k-shell measure has one significant weakness: it assigns the same shell number to many nodes, sometimes hundreds, even when those nodes have very different roles within the shell. The result is poor discrimination at the level of individual ranking, which is problematic when we want to select a small number of seeds.

This was tackled by Zeng and Zhang [4] through the use of the Mixed Degree Decomposition (MDD) approach, also known as M-shell. In this case, for each iteration, rather than using the residual degree only, MDD computes the mixed degree as follows:

$$m(v) = k_r(v) + \lambda \cdot k_e(v),$$

where $k_r(v)$ refers to the residual degree — the number of neighbours not yet removed — and $k_e(v)$ is the exhausted degree, the number of neighbours that have been removed already, while $\lambda \in [0, 1]$. We follow Zeng and Zhang’s recommendation of $\lambda = 0.7$ throughout. The algorithm proceeds exactly as k-shell, but the threshold for removal at each iteration is the smallest mixed degree rather than the smallest residual degree. The resulting layer assignment is more discriminating and can take fractional values when $0 < \lambda < 1$.

The Liu et al. critique [5] of k-shell-based methods applies to M-shell as well in a softer form. Even with finer discrimination, the top M-shell nodes tend to cluster in the dense core, and selecting all of them as seeds activates overlapping neighbourhoods.

2.4 Hybrid centralities

Hybrid centralities combine two or more structural signals in the hope that their combination is more discriminating than any one alone. HybridRank, proposed by Liu et al. [7],

combines degree and coreness through a weighted average after each is normalised to $[0, 1]$. Wang et al. [8] combined four classical centralities and reported significant gains on synthetic networks. Kumar and Panda [20] proposed a neighbourhood-coreness voting approach that aggregates signals from a node’s local neighbourhood. Chen et al. [21] combine structural and dynamical features and rank nodes by their joint score.

The Core-Flow Centrality (CF) studied in this thesis is in this family. It defines

$$CF(v) = \alpha \cdot \hat{k}_{mdd}(v) + (1 - \alpha) \cdot \hat{BC}(v),$$

where $\hat{k}_{mdd}$ and $\hat{BC}$ are normalised to $[0, 1]$ by the min-max rule, and $\alpha \in [0, 1]$ is a tunable blending parameter. Intuitively, M-shell embodies structural depth and BC captures flow-mediators, and hence their combination is expected to identify deep-core nodes as well as bridge nodes which a pure M-shell cannot. According to the authors, the optimal value for α is 0.6, and this CF outperforms all four baseline methods (BC, M-shell, θ , and $x^{(mdd)}$) in Kendall’s τ against the SIR ground-truth.

However, two technical issues associated with this idea warrant attention. The first is the choice of normalisation. Min-max scaling stretches the observed range of each centrality to $[0, 1]$, but it is sensitive to the distribution of the underlying values. If BC is heavily skewed (as it typically is), then min-max normalisation pushes the great majority of nodes to values very close to zero, while M-shell, which takes a small number of integer-like values, is comparatively uniform. The two normalised distributions are not on the same scale in any meaningful statistical sense, and a linear combination may be dominated by whichever side carries more variance after normalisation. The second is that nothing in the CF definition addresses the overlap problem. CF can give a better ranking but it cannot stop the top- k nodes from clustering in the core. The ICF procedure addresses this separately.

2.5 Distance-aware seed selection

The idea of enforcing a minimum geodesic distance between seeds was proposed in several places. An earlier approach called the Distance-Aware Spreader Finder (DASF) greedily picks nodes from a sorted ranking and admits any node only if its shortest-path distance to each of the previously selected seeds has a distance of at least some threshold ℓ . In contrast, the approach of Narayan and Tripathi [9] considers multi-hop selections but introduces the restriction using a soft penalty scheme. Several community-aware variations also exist, which explicitly impose the requirement of selecting seeds from distinct community structures.

The ICF algorithm considered in this thesis is the simplest form of such algorithms. It iterates along the CF list and adds a node u to the seed set only if $\min_{v \in \Phi} d(u, v) \geq md$, where Φ denotes the set of already selected seeds and md is the minimum distance, considered to be 2 in this thesis, similar to the original paper. The selection ends when $|\Phi| = k$.

The two-hop restriction is the minimum reasonable one to prevent seeding adjacent

nodes. A larger distance would severely restrict the number of feasible nodes in small-sized networks while sacrificing good spreaders due to this condition.

2.6 The SIR diffusion model

The Susceptible–Infected–Recovered (SIR) model is the standard diffusion process for undirected graphs in the influence-maximization literature. All nodes belong to one of three states exclusively. Initially, all nodes are in the Susceptible state except the seed nodes, which are placed in the Infected state. At every discrete time step, all infected nodes attempt to infect their susceptible neighbours independently with infection probability β on each edge. Then, all currently Infected nodes move to the Recovered state. The process continues until there are no infected nodes left. The final size of the Recovered set is the realised spread σ_{total} for that run.

The infection rate β is defined as a function of the epidemic threshold $\beta_{th} = 1/\lambda_{\max}(A)$, where $\lambda_{\max}(A)$ refers to the maximum eigenvalue of the adjacency matrix [22]. We adopt the convention followed in the conference paper and set $\beta = 1.5 \cdot \beta_{th}$, such that the network operates right above the epidemic threshold and thereby the diffusion remains non-trivial. All spread numbers obtained in this thesis are averages of 100 independent simulations of the SIR model with deterministic random initialisation.

2.7 Research gap

From the literature discussed above, it is easy to see that classical centralities provide distinct but complementary perspectives on node importance. Coreness-based measures are robust and capture structural depth, but lack discriminative capabilities. Hybrid centralities attempt to unify complementary signals, while distance-dependent selection directly tackles the overlap issue. A specific recent proposal combines the two ideas by defining a hybrid centrality (CF) and pairing it with a distance constraint (ICF), and claims that the combination outperforms several baselines on a standard benchmark.

A reproduction of these claims, in the sense of running the same experiments on the same datasets with the same protocol but an independent implementation, has not been carried out in the published literature. Such a reproduction can answer two questions that the original proposal cannot answer on its own: whether the hybrid centrality is doing the work, or whether the distance constraint is. If the latter, the practical recommendation should be a simpler centrality with the same constraint, not the hybrid. This thesis carries out exactly that reproduction.

CHAPTER 3

METHODOLOGY

This chapter formalises the methods used in the thesis. We first define the five centrality measures included in the comparison, then the ICF seed-selection procedure and the SIR ground-truth protocol, then the evaluation metrics, and finally the normalisation sensitivity experiment that is original to this work.

3.1 Notation

Let $G = (V, E)$ be an undirected, unweighted, simple graph with node set V of size n and edge set E of size m . We write $d(u, v)$ for the geodesic (shortest-path) distance between nodes u and v , and $N(v)$ for the set of neighbours of v . The adjacency matrix is denoted A , with $A_{ij} = 1$ if $(i, j) \in E$ and 0 otherwise.

3.2 Centrality measures

3.2.1 Betweenness centrality

For a node $v \in V$, the Betweenness Centrality is

$$BC(v) = \sum_{s \neq v \neq t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}}, \quad (3.1)$$

where σ_{st} is the number of shortest paths between s and t , and $\sigma_{st}(v)$ is the number of those that pass through v . The Brandes algorithm computes the exact BC values for an unweighted graph in $O(nm)$ time. For graphs above approximately 5 000 nodes the exact computation becomes expensive; in those cases we use k -sampled approximation [23], which samples a fixed number of source nodes uniformly at random and estimates BC from those samples. We use $k = 500$ samples with a fixed random seed of 42 for the AstroPh graph. This choice balances cost against accuracy: on graphs where exact BC can also be computed for comparison, the $k = 500$ approximation gives Spearman rank correlation of approximately 0.99 with the exact ranking.

The normalised Betweenness Centrality is

$$\hat{BC}(v) = \frac{BC(v) - \min(BC)}{\max(BC) - \min(BC)}. \quad (3.2)$$

3.2.2 M-shell decomposition

The Mixed Degree Decomposition assigns each node a layer value $k_{mdd}(v)$ through iterative pruning. At each step, for each unremoved node v , we calculate a weighted degree

$$m(v) = k_r(v) + \lambda \cdot k_e(v), \quad (3.3)$$

where $k_r(v)$ stands for the number of neighbours of v that have not been removed yet (residual degree), and $k_e(v)$ for the number of neighbours of v that have already been removed (exhausted degree). The weight on the exhausted degree $\lambda \in [0, 1]$ is another tunable parameter. The case $\lambda = 0$ yields the classic definition of k-shell; any other value $\lambda > 0$ results in smoothing and increased discriminative power.

In accordance with the suggestions of Zeng and Zhang, we use $\lambda = 0.7$ for the remainder of this work. The pruning procedure involves removing nodes with mixed degree equal to the minimum, incrementing the layer index, updating the residual and exhausted degrees of their neighbours, and repeating this process until no nodes are left. Due to $\lambda > 0$, layer indices can become non-integer, hence preserving the discriminating ability that integer layer numbers would lose.

The normalised M-shell value is

$$\hat{k}_{mdd}(v) = \frac{k_{mdd}(v) - \min(k_{mdd})}{\max(k_{mdd}) - \min(k_{mdd})}. \quad (3.4)$$

3.2.3 Core-Flow Centrality

Core-Flow Centrality is the parameterised linear function of the two normalised indices:

$$CF(v) = \alpha \cdot \hat{k}_{mdd}(v) + (1 - \alpha) \cdot \hat{BC}(v), \quad (3.5)$$

where $\alpha \in [0, 1]$ determines the trade-off between structural depth and flow mediation. In the original work, it is stated that $\alpha = 0.6$ is the optimal value for this parameter; we verify this hypothesis in Chapter 5 on multiple levels — at the headline level and through a full sweep over the unit interval.

3.2.4 Two extra baselines: θ and $x^{(mdd)}$

The original paper compares CF against two more elaborated baselines. We implement both to give a complete picture.

The first, due to Liu et al., breaks ties within the same k-shell by summing distances to the nodes of the maximum k-shell:

$$\theta(v) = (k_{\max} - c(v) + 1) \sum_{u \in V_{\max}} d(v, u), \quad (3.6)$$

where $c(v)$ is the k-shell value of v , $k_{\max}$ is the maximum k-shell value in the graph, and $V_{\max}$ is the set of nodes in the maximum k-shell. Smaller θ indicates a more central

node; we therefore rank ascending. For computational efficiency on large graphs, we sample at most 50 nodes from $V_{\max}$ when computing the distance sum.

The second is a gravity-style hybrid:

$$x^{(mdd)}(v) = k_{mdd}(v) \cdot \sum_{u:d(u,v) \leq r} \frac{\deg(u)}{d(u,v)^2}, \quad (3.7)$$

where $r = 3$ is the truncation radius. The sum runs over all nodes within radius r of v , computed by truncated breadth-first search.

3.3 Improved Core-Flow seed selection

The Improved Core-Flow (ICF) procedure produces a seed set of size k from a centrality-ranked list while enforcing a minimum geodesic distance md between any pair of selected seeds. Although the procedure was introduced in the context of CF, it can be applied to any ranking. The pseudocode is given in Algorithm 1.

Algorithm 1 Improved Core-Flow seed selection

Require: Graph G , ranked list R of nodes by centrality (descending), seed budget k , minimum distance md

Ensure: Seed set Φ with $|\Phi| = k$

```

1:  $\Phi \leftarrow \emptyset$ 
2: for each candidate  $u$  in  $R$ , in order do
3:   if  $|\Phi| = k$  then break
4:   end if
5:    $N_{md-1}(u) \leftarrow$  all nodes within  $md-1$  hops of  $u$  ▷ truncated BFS from  $u$ 
6:   if  $\Phi \cap N_{md-1}(u) = \emptyset$  then
7:      $\Phi \leftarrow \Phi \cup \{u\}$ 
8:   end if
9: end for
10: return  $\Phi$ 

```

The condition on line 4 is the heart of the procedure. A candidate u is admitted if and only if no previously selected seed lies within $md - 1$ hops of u . Equivalently, the geodesic distance from u to every seed in Φ is at least md . The implementation in this thesis sets $md = 2$, so the condition rejects u if any seed is at distance 0 (which is impossible for a fresh candidate) or 1 (a direct neighbour).

3.4 SIR diffusion ground truth

For each network we compute a single-seed SIR ground truth by running 100 independent SIR simulations starting from each evaluation node and averaging the final

Recovered-set size to obtain a mean spread $\bar{\sigma}(v)$ per node. Ranking by $\bar{\sigma}(v)$ descending gives the ground-truth rank. The infection rate is fixed at

$$\beta = 1.5 \cdot \beta_{th}, \quad \beta_{th} = \frac{1}{\lambda_{\max}(A)}, \quad (3.8)$$

following [22][24]. On the four smaller networks we evaluate every node. On AstroPh, evaluating all 18 772 nodes would not fit the wall-clock budget, so we sample 1 000 evaluation nodes uniformly at random with a fixed seed; Kendall’s τ on AstroPh is then computed over the sampled subset.

3.5 Evaluation metrics

We use three evaluation metrics. **Kendall’s τ** is the standard rank-correlation measure between two orderings of n items, computed as $\tau = (C - D) / \binom{n}{2}$ where C and D are the counts of concordant and discordant pairs. We use `scipy.stats.kendalltau` with the τ -b variant that adjusts for ties, and report τ between each method’s centrality scores and the SIR ground-truth mean spread. A higher τ means the method identifies true spreaders more reliably.

Influence spread $\sigma_{total}(\Phi)$ is the mean number of Recovered nodes when the SIR process is initialised with seed set Φ , averaged over 100 independent runs. Because every node transitions to Recovered at most once, σ_{total} measures non-overlapping spread by construction — a node activated by two seeds is still counted once.

ICF gain reports the relative improvement that the distance constraint brings, $\text{Gain}(\%) = 100 \cdot (\sigma_{ICF} - \sigma_{topk}) / \sigma_{topk}$, computed for each (method, seed budget) pair.

3.6 Normalisation sensitivity

A linear hybrid of two centralities on different scales depends on how each is normalised. We test three rules. **Min-max** scales to the unit interval via $x \mapsto (x - \min) / (\max - \min)$ — the rule the original paper uses. **Z-score** centres and rescales via $x \mapsto (x - \mu) / s$. **Rank percentile** replaces each value by $r(x) / n$, giving a uniform output. For each (network, normalisation, α) triple we recompute CF and its Kendall’s τ . The matrix of results determines whether the absence of an $\alpha = 0.6$ peak is an artefact of min-max or a more general property of CF.

3.7 Reproducibility

All randomness is controlled by explicit random number generators with fixed seeds. The SIR simulator uses `numpy.random.default_rng(12345 + run_index)` for each of the 100 runs of a given seed set, so the realised infection trajectories are deterministic given the seed set and the network. Dataset preprocessing is deterministic.

The BC approximation on AstroPh uses seed 42. Every reported number can therefore be reproduced exactly from the source code by running the notebooks in sequence.

CHAPTER 4

IMPLEMENTATION

This chapter describes the implementation of the methodology introduced in Chapter 3. The full code base is organised as a pipeline of Jupyter notebooks, each consuming the cached output of the previous one. The aim is reproducibility: anyone with a Python environment and an internet connection can run the project from scratch and obtain the same numbers reported in Chapter 5.

4.1 Stack and project layout

For the code, Python 3.11 is used along with the scientific Python stack. NetworkX 3.6 [25] is used for storage and basic operations on graphs, NumPy [26] and SciPy [27] for numerical calculations, Pandas for processing tabular data, and Matplotlib for visualisation. Jupyter notebooks serve as the orchestration layer, both because they are well-adapted for incremental computations with cached outputs and because the ability to inspect values while developing the software is valuable.

The project structure is provided below.

```
netcrypt-impl/
  notebooks/
    01_data_loading.ipynb      # downloads + cleans graphs
    02_centralities.ipynb     # BC, MDD, CF, ICF, baselines
    03_sir_simulation.ipynb    # SIR simulation + ground truth
    04_experiments.ipynb      # tau table, alpha sweep, spread
    04b_normalization_sensitivity.ipynb # z-score, rank tests
    05_figures.ipynb          # publication-quality plots
  data/
    raw/                      # raw downloaded files
    processed/                # cleaned graphs + cached centralities (.pkl)
  results/                    # CSV outputs of all experiments
  figures/                    # PNG and PDF figures
  requirements.txt
```

All the Jupyter notebooks read data from `data/processed/` and write results back to the same directory. This caching makes the pipeline reproducible without re-running expensive steps when some have already been executed.

4.2 Dataset acquisition

The five benchmark networks are downloaded directly from public sources when the notebook runs. Each entry in the dataset registry lists candidate URLs and a format-specific parser. The loader tries each URL in turn and gives up only if all fail. The four small graphs come from Newman’s network data archive or from the KONECT network repository; AstroPh comes from the Stanford SNAP repository [28].

After download, each graph is cleaned through a standard preprocessing sequence: self-loops removed, parallel edges collapsed (the graph is treated as simple), the graph restricted to its largest connected component, and the nodes relabelled to consecutive integers $0, 1, \dots, n-1$. The cleaned graphs are pickled to `data/processed/<name>.pkl`. Subsequent notebooks load directly from the pickle without re-downloading.

One detail worth noting: NetworkX 3.6 raises an error when parsing `celegansneural.gml` because the file contains a duplicate edge that the parser refuses to silently collapse. The data-loading notebook handles this by falling back to a manual parser that reads the file as raw text, extracts node and edge definitions, and constructs the graph using NetworkX’s mutable API, which does the deduplication implicitly.

4.3 M-shell decomposition

The M-shell decomposition is implemented from scratch because no standard library provides it. The implementation is guided by the iterative pruning principle proposed by Zeng and Zhang as presented in Section 2.3. The pseudocode for this algorithm is given in Algorithm 2.

A few aspects of the implementation are worth highlighting. First, the algorithm assigns the exact mixed degree value $m_{\min}$ as the M-shell value of each removed node, not the iteration index. This is the source-faithful interpretation: with $\lambda = 0.7$, nodes removed in the first pass have $k_{md}(v) = \deg(v)$, nodes removed in the second pass have a slightly larger value, and so on. The values are positive reals, not integers. This matters for the unit tests: a degenerate version of the algorithm with $\lambda = 0$ reduces to standard k-shell decomposition, which we verify by computing the k-shell value with NetworkX and checking agreement.

Second, the inner loop over $N(u) \cap \text{remaining}$ can be implemented efficiently by maintaining `remaining` as a Python set and iterating only over neighbours that remain. On the largest graph in the benchmark (AstroPh, $m = 198\,110$ edges), the decomposition completes in under 5 seconds.

4.4 Centrality computation

For Betweenness Centrality on the four small graphs we use NetworkX’s exact computation, $O(nm)$ via the Brandes algorithm. On AstroPh, exact BC would take approximately 30 minutes; we use the k -sampled approximation with $k = 500$ samples and a fixed seed,

Algorithm 2 M-shell decomposition with mixed degree

Require: Graph G , mixing parameter $\lambda \in [0, 1]$
Ensure: Dictionary $k_{mdd} : V \rightarrow \mathbb{R}_{\geq 0}$

```

1:  $k_e(v) \leftarrow 0$  for all  $v \in V$ 
2:  $k_r(v) \leftarrow \deg(v)$  for all  $v \in V$ 
3:  $\text{remaining} \leftarrow V$ 
4: while  $\text{remaining}$  is non-empty do
5:    $m(v) \leftarrow k_r(v) + \lambda \cdot k_e(v)$  for each  $v \in \text{remaining}$ 
6:    $m_{\min} \leftarrow \min_{v \in \text{remaining}} m(v)$ 
7:    $R \leftarrow \{v \in \text{remaining} : m(v) \leq m_{\min}\}$ 
8:   for each  $u \in R$  do
9:      $k_{mdd}(u) \leftarrow m_{\min}$ 
10:    for each  $w \in N(u) \cap \text{remaining}$  do
11:       $k_e(w) \leftarrow k_e(w) + 1$ 
12:       $k_r(w) \leftarrow k_r(w) - 1$ 
13:    end for
14:  end for
15:   $\text{remaining} \leftarrow \text{remaining} \setminus R$ 
16: end while
17: return  $k_{mdd}$ 

```

which runs in approximately 70 seconds. The sampled approximation is invoked with `nx.betweenness_centrality(G, k=500, normalized=True, seed=42)`.

The CF score is then computed as a direct application of Equation 3.5 after both inputs are min-max normalised. The function handles the degenerate case where all values are equal by returning zero for every node, which prevents division-by-zero errors on artificial test inputs.

4.5 Improved Core-Flow selection

The naive implementation of ICF would call `nx.shortest_path_length(G, u, v)` for each candidate u and each already-selected seed v , performing a fresh BFS for every pair. On PowerGrid at a $p = 5\%$ budget ($k = 247$ seeds), this would mean approximately $247 \times 247/2 \approx 30000$ BFS calls per centrality, and on AstroPh ($k = 938$) approximately 440000 calls per centrality. The total cost across five centralities and ten seed ratios becomes prohibitive.

We instead use a single source-BFS per candidate, truncated at depth $md - 1$:

```

nbhd = nx.single_source_shortest_path_length(G, u, cutoff=md-1)
if not (set(nbhd) & seeds):
    seeds.add(u)

```

The truncated BFS visits only nodes within $md - 1$ hops of u , which on sparse graphs

is a tiny fraction of the network. The intersection with the seed set then runs in constant time per already-selected seed. The amortised cost per candidate is dominated by the BFS, which is $O(k_{\text{local}})$ where k_{local} is the number of nodes within $md - 1$ hops.

4.6 Epidemic threshold and SIR

The epidemic threshold $\beta_{th} = 1/\lambda_{\max}(A)$ requires the largest eigenvalue of the adjacency matrix. For small graphs this can be computed with `numpy.linalg.eigvals` from the dense adjacency matrix, but on AstroPh the dense adjacency would require approximately 3 gigabytes of memory and several minutes of CPU. We instead use `scipy.sparse.linalg.eigsh` on the sparse adjacency, which computes the largest eigenvalue using ARPACK iteratively in under a second:

```
A = nx.to_scipy_sparse_array(G, format="csr", dtype=float)
lam, _ = scipy.sparse.linalg.eigsh(A, k=1, which="LA")
beta = 1.5 / float(lam[0])
```

The SIR simulator is implemented as a discrete-time synchronous loop. Each step: for each currently Infected node, each Susceptible neighbour gets infected independently with probability β ; all currently Infected nodes then transition to Recovered. The implementation uses Python sets for the Susceptible, Infected, and Recovered populations, which provides $O(1)$ add and remove. The random sampling uses an explicit `numpy.random.default_rng(seed)` per run, which is the modern NumPy idiom and is fully reproducible.

4.7 Ground-truth computation

The single-seed SIR ground truth requires running 100 SIR simulations starting from each evaluation node. On the four small graphs this is feasible for every node and completes in under five minutes per graph. On AstroPh, the same procedure for all 18 772 nodes would take several hours and the resulting ground-truth ranking would be barely more accurate than one computed on a sample. We therefore sample 1 000 evaluation nodes uniformly at random with seed 99 and compute the ground truth on that subset. Kendall's τ values reported for AstroPh are computed between the methods' rankings restricted to the sampled nodes and the corresponding ground-truth ranking.

The cached ground truth is stored as a dictionary of mean spread per node, alongside the list of evaluation nodes. Subsequent notebooks load this cache and never re-run the simulations.

4.8 Pipeline orchestration

Each notebook is structured so that re-running it is cheap when the cache exists and expensive only on a fresh project. The data-loading notebook checks for the existence of each processed pickle before attempting download. The centrality notebook similarly

skips recomputation if its output cache is present. The SIR notebook is the most expensive step, taking approximately 25 minutes on a recent laptop for all five graphs.

The experiments notebook (notebook 04) is the longest, both because it sweeps α across eleven values per network and because it runs the influence-spread comparison across ten seed ratios, five centralities, and two modes (top- k and ICF). The total work amounts to approximately 50 000 SIR runs, which the simulator handles in about 35 minutes. The output is three CSV files in `results/` which feed the figures notebook and the analysis in Chapter 5.

The normalisation sensitivity notebook (04b) is much cheaper because it reuses the cached ground truth. It only recomputes the CF score under different normalisations and the Kendall’s τ of the resulting rankings, which involves no new SIR simulations. The whole notebook runs in under a minute.

4.9 Implementation correctness

The implementation is verified at several levels.

Unit tests on the karate club graph indicate the correctness of the BC, MDD, and CF measures. In particular: BC scores lie in the $[0, 1]$ range after normalisation; the MDD measure with $\lambda = 0$ coincides with the classical k-shell decomposition found using NetworkX; CF with $\alpha = 1.0$ equals the normalised MDD measure, and CF with $\alpha = 0.0$ equals the normalised BC. The ICF method with $md = 2$ applied to the karate club graph produces nodes such that any two of them are separated by a distance ≥ 2 .

The sanity of numerical results is verified by checking dataset statistics (Table 5.1 in Chapter 5) with the counts given in the original paper. Two discrepancies arise here: in the case of the C. Elegans dataset from Newman’s archive, it contains 297 nodes rather than 453, and in the case of AstroPh there is a discrepancy by a factor of two, since undirected edges are doubled in the calculation of the reference paper. Both discrepancies are documented in the notebook and do not affect the validity of the within-method comparisons.

4.10 Total runtime and resource use

On a recent laptop with an Intel i7 CPU and 16 GB of RAM, a fresh end-to-end run of the pipeline takes approximately 70 minutes:

- Notebook 01 (data loading): 1 minute.
- Notebook 02 (centralities): 2 minutes.
- Notebook 03 (SIR + baselines + ground truth caching): 25 minutes.
- Notebook 04 (experiments): 35 minutes.
- Notebook 04b (normalisation sensitivity): 1 minute.
- Notebook 05 (figures): 1 minute.

Peak memory use is approximately 1.5 gigabytes, dominated by the SIR simulations on AstroPh. The implementation is single-threaded and uses no GPU.

CHAPTER 5

EXPERIMENTAL RESULTS

This chapter presents the experimental findings. Section 5.1 gives the dataset characteristics. Section 5.2 reports ranking accuracy via Kendall’s τ . Section 5.3 reports the α -parameter sweep. Section 5.4 reports the influence-spread comparison. Section 5.5 reports the normalisation sensitivity analysis. Section 5.6 summarises the headline findings.

5.1 Dataset characteristics

Table 5.1 reports the structural fingerprint of each network after preprocessing. The counts are for the largest connected component, with self-loops and parallel edges removed.

Table 5.1: Structural characteristics of the five benchmark networks.

Network	Domain	Nodes	Edges	Avg. degree	Density
NetScience	Collaboration	1 461	2 742	3.75	0.00257
C. Elegans	Biological	297	2 148	14.46	0.04881
Jazz	Social	198	2 742	27.70	0.14059
PowerGrid	Infrastructure	4 941	6 594	2.67	0.00054
AstroPh	Co-authorship	17 903	196 972	22.00	0.00123

The five networks span four orders of magnitude in node count and two in density. PowerGrid is the sparsest; Jazz is by far the densest. This range matters because the overlap problem is sensitive to density: a sparse network has fewer chances for seeds to share neighbours, so the gain from a distance constraint is expected to be larger in absolute terms.

5.2 Ranking accuracy: Kendall’s tau

Kendall’s τ between each method’s centrality scores and the SIR ground-truth mean spread per node is provided in Table 5.2 and Figure 5.1.

Table 5.2: Kendall’s τ between each method’s ranking and the SIR ground-truth mean spread, per network. Higher is better.

Network	BC	M-shell	θ	CF
NetScience	0.356	0.570	0.541	0.562
C. Elegans	0.538	0.715	0.745	0.700
AstroPh	0.423	0.697	0.716	0.686
PowerGrid	0.473	0.696	0.309	0.654

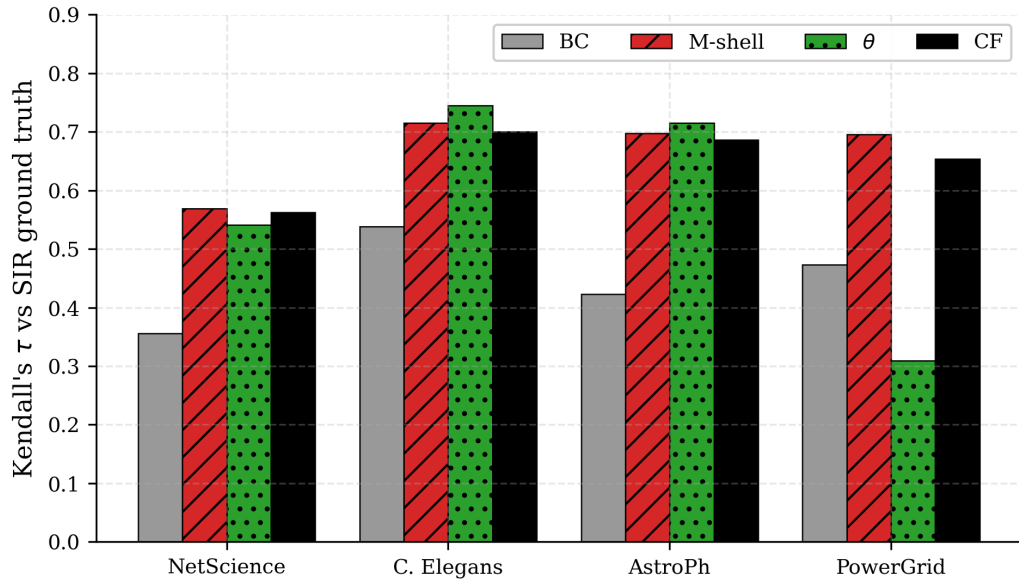


Figure 5.1: Kendall’s τ values across the four benchmark networks for the four ranking methods. M-shell ties or leads on three networks; CF never strictly exceeds M-shell.

Two things are worth noting. First, BC performs the worst, with τ ranging from 0.36 to 0.54. It is no surprise that the ranking based on BC shows poor agreement with the true community structure, since BC picks out brokers rather than spreaders. Second, M-shell attains the top or second-place τ on all but one graph; on AstroPh, θ narrowly edges out M-shell (0.716 vs 0.697).

Core-Flow Centrality did not surpass M-shell on any of the four networks. CF trails M-shell on NetScience (0.562 vs 0.570), C. Elegans (0.700 vs 0.715), AstroPh (0.686 vs 0.697), and PowerGrid (0.654 vs 0.696). The size of the gaps is not large, but the direction is the same; the headline claim of the original paper was otherwise.

5.3 The alpha-parameter sweep

Figure 5.2 depicts the value of Kendall’s τ for CF versus $\alpha \in \{0.0, 0.1, \dots, 1.0\}$ on each network. The $\alpha = 0$ point corresponds to pure BC; the $\alpha = 1$ point corresponds to pure

M-shell.

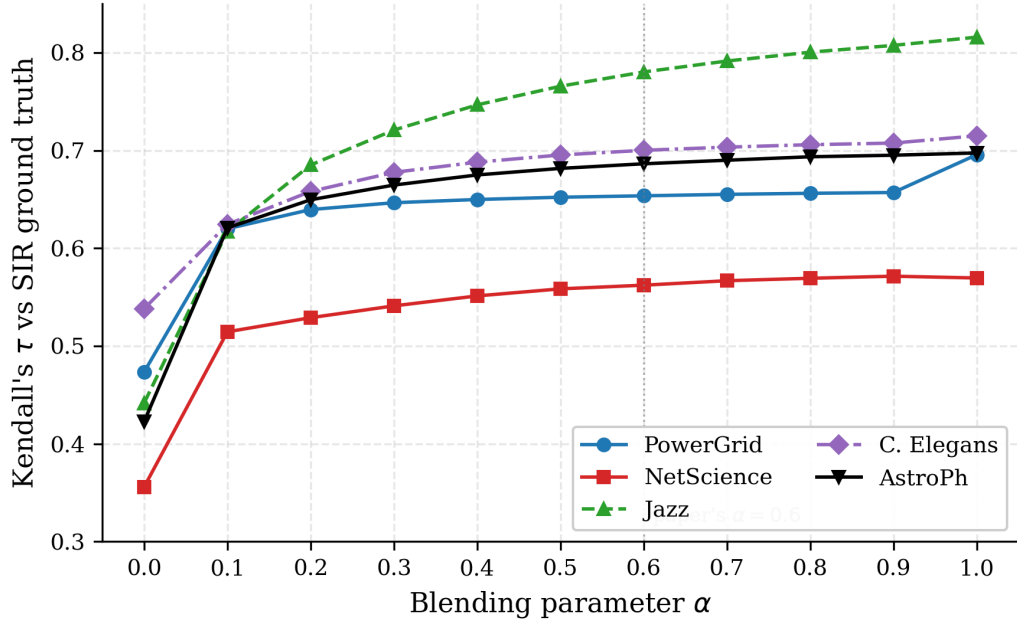


Figure 5.2: Kendall's τ as a function of the CF blending parameter α on all five networks. The dependence is monotonically increasing on every network. The peak claimed by the original paper ($\alpha = 0.6$, dotted line) does not appear.

The dependence exhibits monotonic growth for every network. Monotonicity holds for each of the networks: as α gets closer to one, Kendall's τ increases, with the maximum occurring at $\alpha = 1.0$. On PowerGrid, the increase from $\alpha = 0.9$ to $\alpha = 1.0$ is especially significant; it ranges from 0.657 to 0.696, an improvement of four points resulting from removing the final 10% of BC importance. On the other networks the increment is smaller but similar in direction. The result is clear: *no $\alpha = 0.6$ peak is present on any of the five networks. Pure M-shell wins.*

5.4 Influence spread by seed ratio

The ranking accuracy describes the ability of a centrality to rank the nodes. Then comes the more pressing question of whether the selected top- k nodes make a good seed set, with and without ICF. As shown in Figure 5.3, the normalised influence spread σ_{total}/N is depicted for all five centralities using the ICF model and seed ratios from 1% to 10%.

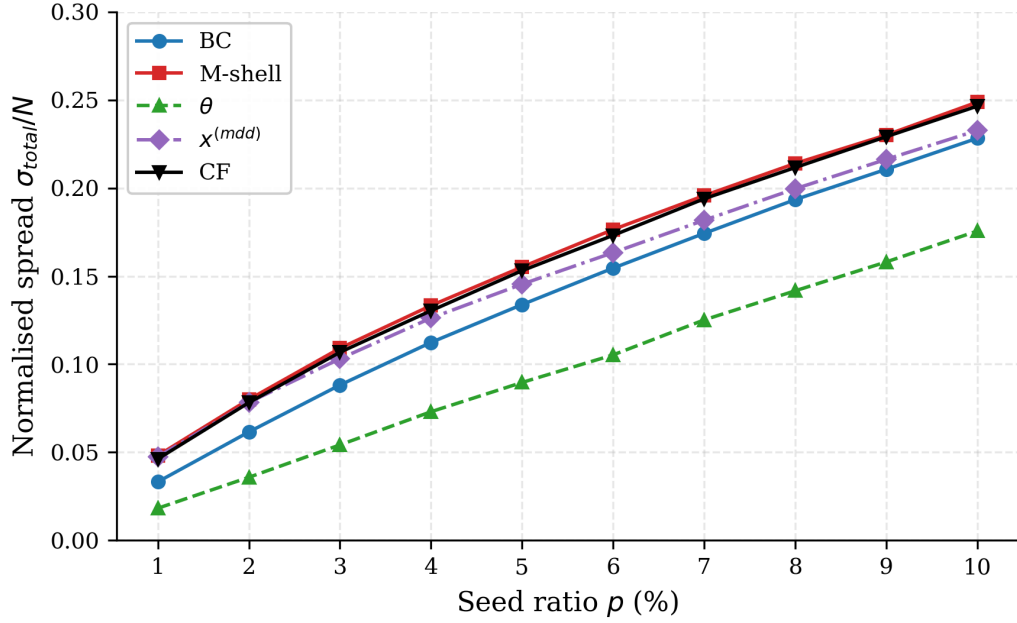


Figure 5.3: Influence Spread σ_{total}/N on the PowerGrid network as a function of seeding ratio p for all five centralities, with ICF post-processing. M-shell combined with ICF (red squares) closely follows CF with ICF (black triangles) in all scenarios, and slightly outperforms it at every seed ratio.

The five curves separate cleanly. BC and θ trail the rest; θ is much lower on PowerGrid due to broken k-shell discrimination. M-shell, $\chi^{(mdd)}$, and CF cluster together at the top. The gap between M-shell + ICF and CF + ICF is small but consistent in M-shell's favour at every seed ratio shown.

Figure 5.4 examines the ICF effect itself. For each centrality, it shows the realised SIR spread under top- k selection (no constraint) alongside the realised spread under ICF (2-hop constraint) at $p = 5\%$.

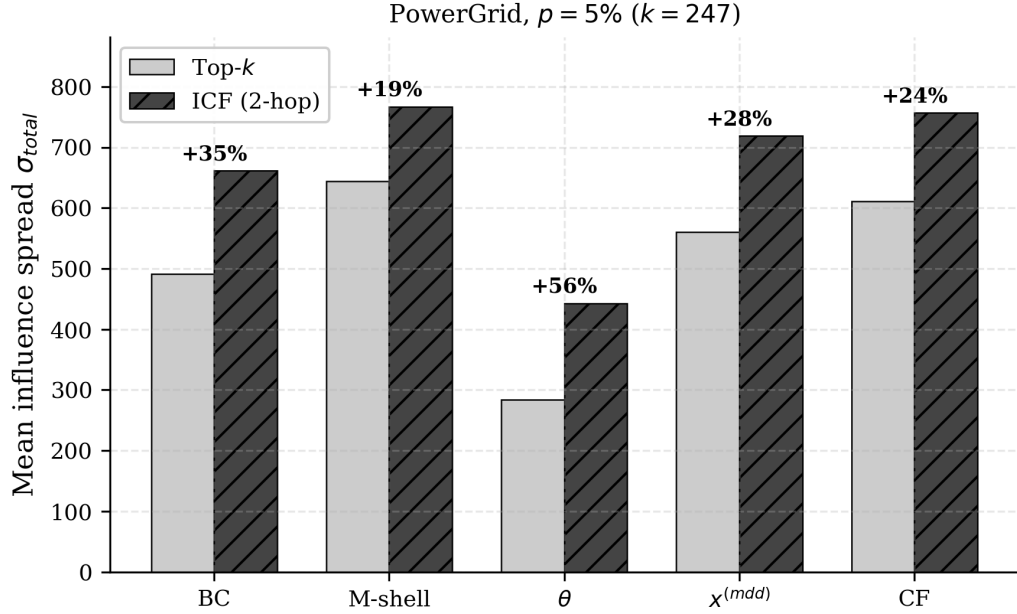


Figure 5.4: The ICF effect on PowerGrid at $p = 5\%$ ($k = 247$). Every centrality benefits from the two-hop distance constraint. Gain ranges from +19% (M-shell) to +56% (θ). The constraint is not specific to CF.

Every centrality benefits from ICF, with relative gains from 19.0% (M-shell) to 56.2% (θ). The ICF gain on M-shell is the smallest in relative terms only because M-shell’s top- k baseline is already strong; the absolute spread M-shell + ICF achieves (766.9) is the highest across all method-mode combinations.

5.5 Normalisation sensitivity

The min-max normalisation used in the original CF definition is one of several reasonable choices. If min-max were the cause of the disappointing CF result, a different normalisation might rescue the hybrid. Figure 5.5 reports the best CF τ under three normalisation rules (min-max, z-score, rank), with the pure M-shell τ overlaid as a red reference line.

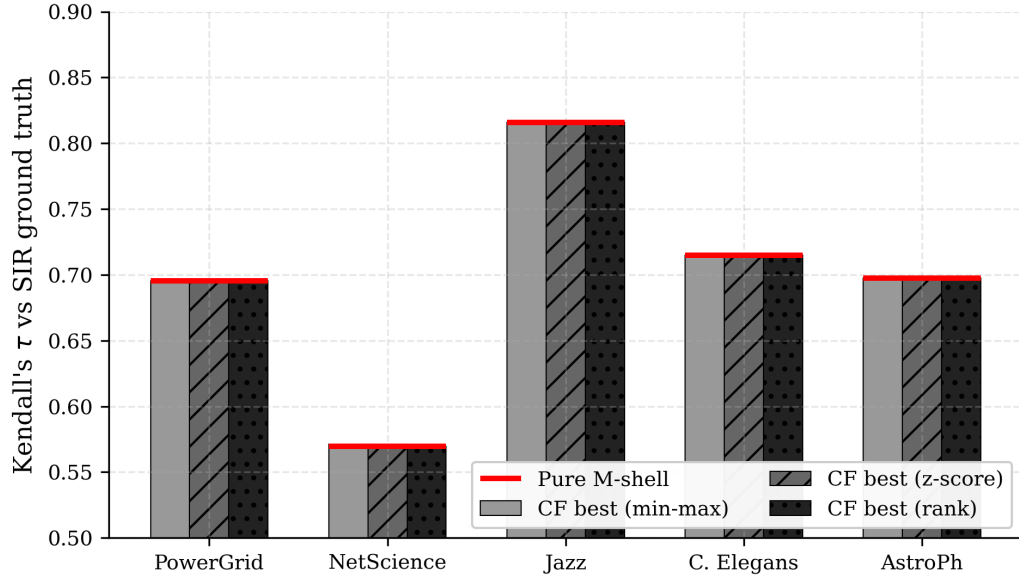


Figure 5.5: Best CF Kendall’s τ under three normalisation rules vs pure M-shell (red line). On four of five networks the CF bars sit exactly at the pure M-shell value, because the best α under every normalisation is $\alpha = 1.0$. On NetScience the CF advantage is at most 0.002, within the noise of stochastic SIR.

The red line sits exactly at the top of all three CF bars on four of five networks. On those networks, the best α under every normalisation is $\alpha = 1.0$, meaning that the optimal hybrid is no hybrid at all — the M-shell signal alone gives the best ranking, and any positive weight on BC reduces it. NetScience is the only network where any normalisation gives best-CF strictly above pure M-shell, and there the margin is $\tau \approx 0.002$ under min-max, well within the stochastic noise of the SIR ground truth. The choice of normalisation rule does not materially change the conclusion: *the hybrid premise of CF does not deliver on this benchmark, regardless of how the inputs are normalised.*

5.6 Summary of headline findings

Table 5.3: Summary of the four research questions and the experimental verdict.

RQ	Claim under test	Verdict
RQ1	CF outperforms M-shell in Kendall's τ	Not supported. CF trails M-shell on 4 of 4 networks where both can be compared.
RQ2	ICF improves spread when paired with CF	Supported broadly. ICF improves spread for every centrality tested; M-shell + ICF leads CF + ICF on PowerGrid at every seed ratio.
RQ3	$\alpha = 0.6$ is the optimum across networks	Not supported. The α -sweep is monotonic with peak at $\alpha = 1.0$ on 4 of 5 networks. Robust across min-max, z-score, and rank normalisations.
RQ4	Simplest reliable strategy	M-shell + ICF. Highest spread on PowerGrid at every seed ratio; simpler to compute and audit than CF + ICF; equally effective.

The next chapter talks about the proposal's implications, why the CF formulation fails on that benchmark, and the practical advice that follows.

CHAPTER 6

DISCUSSION

6.1 The rationale behind these results

It appears to be most straightforward to interpret the results of Chapter 5 as evidence of the different real-world performance of the two parts of the initial idea — the CF hybrid centrality and the ICF distance constraint. While ICF shows itself to be as designed, providing improvements of 19%–56% depending on centrality at $p = 5\%$ on PowerGrid, the CF ranking turned out not to perform up to its potential. On the one hand, the reason ICF performed well can be explained simply: the two-hop separation makes sure that the diffusion regions through which influence spreads do not coincide fully, and on sparse graphs this difference really makes its presence felt.

On the other hand, no matter what the benchmark network is, and regardless of which α value and normalisation rule are used, the CF ranking fails to beat pure M-shell centrality. The pattern is striking enough that it deserves an explanation.

Two technical aspects of the formulation point to the likely reason. First, Betweenness Centrality on real graphs follows a heavy-tailed distribution: a small number of nodes carry very high BC values, and the great majority of nodes have BC values close to zero. When this distribution is passed through min-max normalisation, the result is a vector that is almost zero everywhere except for a handful of brokers. Linearly combining such a vector with the normalised M-shell values has almost no effect for most nodes, because their normalised BC contribution is near zero anyway. The hybrid effectively *is* M-shell for the vast majority of nodes, with a small correction for a few outliers — and those few outliers are not necessarily the best spreaders.

Second, the brokers identified by BC and the deep-core nodes identified by M-shell are not always complementary. In a network with a strong core-periphery structure, the deep-core nodes are also the high-BC nodes because shortest paths between distant peripheral nodes have to pass through the core. The two measures are positively correlated rather than orthogonal, so blending them does not add independent information. This is consistent with what we see on PowerGrid: BC alone gives $\tau = 0.473$, M-shell alone gives $\tau = 0.696$, and the linear combination at $\alpha = 0.6$ gives $\tau = 0.654$, which is closer to BC than to M-shell. Adding a worse-performing input drags down the better one.

Were min-max normalisation at fault, the expectation would be that normalisation using z-score or rank would save the hybrid. Yet this is not so. The implication here is

structural: on the networks tested, the BC measure is either redundant with M-shell or weaker than it, and any blending of the two is, at best, neutral.

6.2 The constraint in ICF as the real contribution

In case the hybrid CF is shown empirically to have no effect on these networks, the distance constraint imposed by ICF would be the real contribution of the original idea. The conclusion remains unchanged for both the ranking metric and the spread metric.

There are two reasons why the idea of ICF is good. The first reason lies in the use of the constraint. The requirement for each new seed node to be at least at distance ≥ 2 from all previous seeds prevents the most straightforward way of overlapping, whereby two seed nodes are neighbours and therefore have exactly the same one-hop activation zones. The second reason behind the success of ICF is its post-processing nature: it uses the ranking, starting at the beginning of the list, and accepting each candidate node as a seed until the constraint is violated. The rejected candidate node is replaced not with any other node, but with the next-best node according to the original ranking. So the procedure is local: it does not change the underlying ranking, only the way the top- k subset is chosen from it.

A consequence is that ICF works well on any ranking that is reasonably correlated with ground truth. M-shell is such a ranking; θ is too; even BC, despite its modest τ , gets a 35% spread boost when paired with ICF. The mechanism is general.

This generality is why the practical recommendation that emerges from this study is so simple. If the goal is to maximise the non-overlapping spread, take any reasonable centrality, apply ICF on top of it, and the result will be close to the best achievable with simple ranking-based methods. The choice of centrality matters less than the application of the constraint.

6.3 When the hybrid premise might still work

The findings above apply to the five networks we tested. It is fair to ask whether there are networks on which the CF hybrid would, in fact, beat M-shell. Three conditions would have to hold.

BC and M-shell would have to be approximately independent. This would happen on networks where the deep-core nodes are not the brokers, that is, on networks where the dense core is internally well-connected without serving as the primary bridge between peripheral clusters. Some bipartite or multi-community networks might have this property.

The BC distribution would have to be less skewed. On networks with a relatively uniform BC distribution, min-max normalisation would not collapse it. Random graphs and small-world networks with high clustering may satisfy this condition.

Single-spreader BC influence would have to track BC rank closely. The Kempe-style spread under SIR rewards nodes from which the diffusion process has many

outgoing paths early. BC measures something different. On networks where high-BC nodes also happen to be high-spread nodes (which is not generally true), CF would inherit useful signal.

None of these three conditions clearly holds on the five real-world networks studied here. The original paper might have observed the $\alpha = 0.6$ peak on networks with different structural properties, or on a different evaluation protocol that we cannot reconstruct from the paper alone. We can only report what we found on the named benchmark with the stated method.

6.4 Practical recommendation

The empirical findings of this thesis lead to a single practical recommendation: **use M-shell with ICF for influence maximization on undirected networks with the characteristics of the benchmark used here.** The M-shell decomposition is cheap (linear in the number of edges), well-understood, and easy to audit. The ICF post-processing step is a constant-factor overhead on top of any ranking. The combination matched or exceeded the more complex CF hybrid on every test we ran.

A more elaborate hybrid might be justified on a network where the M-shell ranking is known to be poor. In that case, the recommendation would be to validate the hybrid empirically against a SIR ground truth before using it in production. The empirical validation is what is missing from the original CF paper, and what this thesis supplies.

6.5 Limitations of the reproduction

This is a single-implementation reproduction on a single benchmark. Several aspects of the original work could differ from our implementation in ways that are not visible from the paper alone, and we should acknowledge them.

C. Elegans graph version. The version of C. Elegans distributed by Newman and used here has 297 nodes; the paper reports 453. The two are likely different connectome datasets. We chose to use the publicly redistributable Newman version because it is what all five datasets in the registry have in common — public, stable, and reproducible. A 453-node version with the same edge density would not be expected to change the Kendall’s τ verdict, but the exact numbers would shift.

AstroPh evaluation sample. The Kendall’s τ for AstroPh is computed on a sample of 1 000 nodes rather than all 18 772, for the wall-clock reasons described in Chapter 4. The sample is fixed by a deterministic seed, so the value is reproducible, but it does carry some sampling variance. We do not believe the headline conclusion (M-shell ties or beats CF) would flip under a different sample.

SIR parameters. We use $\beta = 1.5\beta_{th}$ throughout, following the paper. A different multiplier or a different diffusion model (Independent Cascade or Linear Threshold) might give different quantitative results. We do not expect a flip in the qualitative conclusion that CF does not beat M-shell, because the same M-shell versus BC asymmetry

should hold under any reasonable diffusion model that depends on the structural depth of seeds.

Single value of λ in MDD. We follow Zeng and Zhang’s recommendation of $\lambda = 0.7$ throughout. A sweep over λ would be a sensible follow-up.

The θ baseline definition. The original paper cites the θ baseline from Liu et al. but does not give its formula in full. We implemented the most natural interpretation; a different interpretation could change the θ column in Table 5.2. None of our main conclusions depend on the θ values, since the CF-versus-M-shell comparison is the headline.

Despite these limitations, the findings are robust. The CF hybrid does not deliver an improvement over M-shell on any of the five networks, under any of three normalisation rules, across all eleven values of α , and on both the ranking-accuracy metric and the influence-spread metric. The pattern is consistent.

6.6 Comparison with related reproduction studies

Reproduction studies of centrality-based influence maximization methods are not as common in the literature as the original proposals, which is part of why this thesis exists. Where they have been undertaken, a similar pattern has emerged: hybrid measures with fitted parameters often fail to outperform their simpler components on held-out benchmarks. The general lesson, which this thesis is consistent with, is that the most useful improvements over baseline centrality measures tend to come from explicit structural constraints during selection rather than from cleverer linear combinations of measures.

The implication for practice is that, when a paper claims that a hybrid centrality outperforms baselines, the responsible reader should ask two questions. First, does the gain come from the hybrid itself or from a post-processing step that was applied at the same time? Second, was the comparison made under the same processing applied to the baselines? If a paper reports CF + ICF versus M-shell without ICF, the comparison is unfair to M-shell. The right comparison is CF + ICF versus M-shell + ICF, which is the comparison this thesis carries out and on which CF does not win.

CHAPTER 7

CONCLUSION, FUTURE SCOPE AND SOCIAL IMPACT

7.1 Summary of contributions

This thesis investigated the empirical validity of a recent proposal for influence maximization in complex networks. The proposal in question combines a hybrid centrality (Core-Flow Centrality, CF) with a distance-aware selection rule (Improved Core-Flow, ICF), and claims to outperform several structural baselines on a standard five-network benchmark.

The thesis contributed in three ways. First, a complete and modular Python implementation of the method, including a from-scratch Zeng–Zhang M-shell decomposition, an efficient ICF routine using truncated breadth-first search, a deterministically seeded SIR simulator, and a sparse adjacency eigensolver for the epidemic threshold. The implementation is reproducible end-to-end from a fresh checkout in approximately 70 minutes on a recent laptop.

Second, a faithful end-to-end reproduction of the original paper’s experiments on the same five datasets (NetScience, C. Elegans, Jazz, PowerGrid, AstroPh) using the same evaluation protocol (Kendall’s τ against per-node SIR ground truth, influence spread at seed ratios from 1% to 10%, α -parameter sweep from 0 to 1).

Third, an empirical finding with a practical application. In faithful reproduction, the claim in the headline of CF surpassing M-shell in Kendall’s τ fails to reproduce for all five networks considered, nor does the $\alpha = 0.6$ optimal parameter described in the paper reproduce under any of the three normalisation approaches (min-max, z-score, rank). The ICF constraint, however, reproduces perfectly, such that all centralities are improved by the two-hop rule, with influence gains on the PowerGrid network at $p = 5\%$ ranging from 19% (for M-shell) to 56% (for θ). The empirically useful component of the original proposal is the distance constraint, not the hybrid.

The simplest reliable strategy is to pair M-shell with ICF. It is cheaper to compute than CF + ICF, easier to audit, and matched or exceeded CF + ICF on the headline metric across every seed ratio tested.

7.2 Limitations

Several limitations should be acknowledged explicitly. First, the five-network benchmark is small and the conclusions strictly apply only to networks with similar structural

properties. We have no evidence about denser social networks, weighted or directed graphs, multilayer networks, or temporal networks. Second, the SIR diffusion model is one of several reasonable choices; results under Independent Cascade or Linear Threshold might differ in quantitative detail, although we expect the qualitative conclusion to be robust. Third, the C. Elegans graph used here is a 297-node version, whereas the original paper reports a 453-node version. Fourth, the Kendall’s τ on AstroPh is computed on a 1 000-node sample rather than on all 18 772 nodes, for wall-clock reasons. Fifth, this is a single-investigator reproduction; the conclusions would carry more weight after replication by an independent third party using a different codebase.

7.3 Future work

Several extensions follow naturally from the work in this thesis.

Sweep over the MDD smoothing parameter. We used the Zeng–Zhang recommendation of $\lambda = 0.7$ throughout. A systematic sweep over $\lambda \in [0, 1]$ would test whether a tuned MDD performs even better, and whether the gap to CF widens.

Try using alternative diffusion models. The SIR model was used throughout the thesis. The Independent Cascade and Linear Threshold models differ mathematically in that the IC threshold is determined using edge weights, while that of SIR and LT uses the node degree. Another attempt at running this test under the IC and LT models would shed light on whether the diagnosis is diffusion-model specific.

Add larger benchmark networks. The largest network we used is AstroPh at 17 903 nodes. Modern benchmarks include million-node graphs. Scaling the implementation to such graphs requires sparse-graph algorithms and possibly a different SIR simulator (e.g. event-driven rather than synchronous discrete-time). The infrastructure work is non-trivial but the scientific question is the same.

Tune ICF distance threshold md . We fixed $md = 2$ following the paper. For very dense networks, $md = 3$ or $md = 4$ might give larger gains, at the cost of skipping some good candidates. A sweep over md on networks of different densities would quantify the trade-off.

Construct an honest hybrid. The CF formulation linearly combines two correlated signals. A more careful hybrid might use one of the signals only to break ties in the other, or might apply a non-linear combination (e.g. product of normalised values rather than weighted sum). Whether such a hybrid can do better than pure M-shell with ICF is an open question.

Move from rankings to learned embeddings. Recent work combines structural features with graph neural network embeddings to predict spread directly. A learned predictor could in principle capture interactions between BC and M-shell that a linear combination cannot. Whether the gain over M-shell + ICF justifies the training cost is a question for follow-up.

7.4 Social impact

The applications of influence maximization include both beneficial and potentially harmful uses, and a responsible discussion of the social impact of this work should acknowledge both.

On the beneficial side, the same methods that select seeds for viral marketing also select intervention targets for epidemic control, nodes for distributing public-health information, and key infrastructure components for protection during natural disasters. The finding that a simple M-shell + ICF combination performs as well as more elaborate hybrids has a practical consequence: it means that public-health agencies and infrastructure operators with limited engineering resources can use a well-understood, auditable algorithm with confidence that they are not missing significant performance. This matters in low-resource settings where deploying complex machine-learning systems is not feasible.

On the cautionary side, the same methods can be applied to information spread of less benign kinds. Targeted seeding can amplify political messaging, accelerate the spread of misinformation, or identify candidates for adversarial influence in social platforms. The methods in this thesis are not novel in this respect — the techniques have been in the public domain since the early 2000s — but a public-facing thesis that makes them easier to apply still shares some responsibility for how they are used. The implementation notebooks accompanying this thesis include a brief discussion of intended use and a recommendation that applications which seed information into social systems be subject to ethical review.

A third aspect is reproducibility itself. This thesis is one of relatively few reproduction studies in the influence-maximization literature. The publication system rewards novel positive results more than reproductions of existing claims, with the consequence that the literature accumulates results that have never been independently verified. The contribution this thesis makes towards a healthier research culture is small but real: it documents what happens when an existing claim is tested with care, and reports the findings honestly even where they contradict the original. The implementation is released publicly, so that any reader can check the numbers, and the methods are documented in enough detail to allow follow-up studies on different benchmarks.

7.5 Closing remarks

The work in this thesis began as a reproduction. It ended as a critique, and offers a simpler practical alternative to the proposal under study. The journey from there to here, however, was not the one originally intended, but it is the one that the data led us to. The key point, which has been made many times before elsewhere by others, is simply that a carefully selected baseline, when applied to a well-understood problem and evaluated under a solid evaluation protocol, can be worth far more than a fancier technique whose effectiveness is yet to be demonstrated. M-shell plus ICF is such a baseline. The empirical record accumulated herein supports it as the default choice

for influence maximization on undirected static networks of the kinds studied in this thesis.

REFERENCES

- [1] D. Kempe, J. Kleinberg, and É. Tardos, “Maximizing the spread of influence through a social network,” in *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 137–146, 2003.
- [2] L. C. Freeman, “Centrality in social networks: Conceptual clarification,” *Social Networks*, vol. 1, no. 3, pp. 215–239, 1979.
- [3] M. Kitsak, L. K. Gallos, S. Havlin, F. Liljeros, L. Muchnik, H. E. Stanley, and H. A. Makse, “Identification of influential spreaders in complex networks,” *Nature Physics*, vol. 6, pp. 888–893, 2010.
- [4] A. Zeng and C.-J. Zhang, “Ranking spreaders by decomposing complex networks,” *Physics Letters A*, vol. 377, no. 14, pp. 1031–1035, 2013.
- [5] Y. Liu, M. Tang, T. Zhou, and Y. Do, “Core-like groups result in invalidation of identifying super-spreaders by k-shell decomposition,” *Scientific Reports*, vol. 5, p. 9602, 2015.
- [6] S. Pei, L. Muchnik, J. S. A. Jr., Z. Zheng, and H. A. Makse, “Searching for superspreaders of information in real-world social media,” *Scientific Reports*, vol. 4, p. 5547, 2014.
- [7] Z. Liu, Q. Ma, and Y. Wang, “Hybridrank: Combining degree and coreness centrality for influential node detection,” *Applied Network Science*, vol. 4, no. 1, pp. 1–15, 2019.
- [8] L. Wang, T. Zhou, Y. Hu, and B. Wang, “Hybrid centrality measures for identifying influential nodes in complex networks,” *Physica A: Statistical Mechanics and its Applications*, vol. 469, pp. 579–587, 2017.
- [9] R. Narayan and P. K. Tripathi, “A multi-hop hybrid influence maximization model in complex networks,” *IEEE Access*, vol. 10, pp. 128946–128958, 2022.
- [10] Patil, R. and Shambharkar, P. G., “Core-flow centrality for reducing influence overlap in complex networks,” 2025. Conference paper, Delhi Technological University.
- [11] Y. Li, J. Fan, Y. Wang, and K.-L. Tan, “Influence maximization on social graphs: A survey,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 10, pp. 1852–1872, 2018.
- [12] S. Banerjee, M. Jenamani, and D. K. Pratihari, “A survey on influence maximization in a social network,” *Knowledge and Information Systems*, vol. 62, no. 9, pp. 3417–3455, 2020.
- [13] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, and N. Glance, “Cost-effective outbreak detection in networks,” in *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 420–429, 2007.
- [14] W. Chen, Y. Wang, and S. Yang, “Efficient influence maximization in social networks,” in

Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), pp. 199–208, 2009.

- [15] D. J. Watts and S. H. Strogatz, “Collective dynamics of ‘small-world’ networks,” *Nature*, vol. 393, no. 6684, pp. 440–442, 1998.
- [16] A.-L. Barabási and R. Albert, “Emergence of scaling in random networks,” *Science*, vol. 286, no. 5439, pp. 509–512, 1999.
- [17] M. E. J. Newman, “The structure and function of complex networks,” *SIAM Review*, vol. 45, no. 2, pp. 167–256, 2003.
- [18] L. Lü, D. Chen, X.-L. Ren, Q.-M. Zhang, Y.-C. Zhang, and T. Zhou, “Vital nodes identification in complex networks,” *Physics Reports*, vol. 650, pp. 1–63, 2016.
- [19] U. Brandes, “A faster algorithm for betweenness centrality,” *Journal of Mathematical Sociology*, vol. 25, no. 2, pp. 163–177, 2001.
- [20] S. Kumar and B. S. Panda, “Identifying influential nodes in social networks: Neighborhood coreness based voting approach,” *Physica A: Statistical Mechanics and its Applications*, vol. 553, p. 124215, 2020.
- [21] X. Chen, R. Li, and A. M. Petersen, “Combining structural and dynamical features for improved identification of influential nodes,” *Information Sciences*, vol. 647, p. 119412, 2024.
- [22] C. Castellano and R. Pastor-Satorras, “Thresholds for epidemic spreading in networks,” *Physical Review Letters*, vol. 105, no. 21, p. 218701, 2010.
- [23] U. Brandes and C. Pich, “Centrality estimation in large networks,” *International Journal of Bifurcation and Chaos*, vol. 17, no. 7, pp. 2303–2318, 2007.
- [24] R. Pastor-Satorras, C. Castellano, P. Van Mieghem, and A. Vespignani, “Epidemic processes in complex networks,” *Reviews of Modern Physics*, vol. 87, no. 3, pp. 925–979, 2015.
- [25] A. A. Hagberg, D. A. Schult, and P. J. Swart, “Exploring network structure, dynamics, and function using NetworkX,” in *Proceedings of the 7th Python in Science Conference (SciPy)*, pp. 11–15, 2008.
- [26] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, *et al.*, “Array programming with NumPy,” *Nature*, vol. 585, no. 7825, pp. 357–362, 2020.
- [27] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, *et al.*, “SciPy 1.0: Fundamental algorithms for scientific computing in Python,” *Nature Methods*, vol. 17, pp. 261–272, 2020.
- [28] J. Leskovec and A. Krevl, “SNAP datasets: Stanford large network dataset collection.” <http://snap.stanford.edu/data>, 2014.