



New Seeding Strategies for the Influence Maximization Problem

Seok-Hee Hong^{1(✉)}, Juan Pablo Bonilla Ataidés¹, Rowena Kok¹,
Amyra Meidiana¹, and Kunsoo Park²

¹ University of Sydney, Camperdown, Australia

seokhee.hong@sydney.edu.au,

{jbon1036,rkok7745,amei2916}@uni.sydney.edu.au

² Seoul National University, Seoul, Korea

kpark@snu.ac.kr

Abstract. We present two new seeding strategies for the Influence Maximization Problem for Viral Marketing, based on graph connectivity and spectral graph theory. Specifically, the first approach CVSP uses the *cut vertices* and the *separation pairs* as the starting seeds. The second approach ER uses the vertex ranking based on the *effective resistance* values of the incident edges. CVSP and ER are efficient, and can be implemented in linear and near linear time, respectively.

Experiments using the Independent Cascade diffusion model with real-world data sets show that our new seeding strategies perform significantly better than the existing methods, such as centrality measures, k -core and the state-of-the-art IMM, in particular for the scale-free networks with globally sparse, locally dense clusters with small diameters, in the final influence spread. Moreover, visual analysis enables more refined comparison between the methods, demonstrating that our methods have more globally wide influence spread pattern than other methods with locally dense influence spread pattern.

Keywords: Influence Maximization · Viral Marketing · Network Visualization

1 Introduction

In today's fast developing technology and increasing information availability, network models are growing increasingly complex. Coupled with the substantial growth of social media in recent years, businesses have realized the potential of analyzing these social networks, particularly with the notion of Viral Marketing.

Through Viral Marketing, a business' marketing message is transmitted by word-of-mouth exchanges in an exponentially growing manner [10]. The Viral Marketing Strategy can be extremely effective because it is crucially based on trust within individuals' social network [28]. In this way, influence and product adoption can spread from a few key *seed* influencers to a large portion of users in the network. Analyzing the spread of influence of Viral Marketing may also

This work is supported by ARC grant DP190103301.

have applications in other areas, such as epidemiology, since Viral Marketing is considered analogous to the transmission of infectious disease in an epidemic.

Unfortunately, choosing these initial seed users such that influence is maximized, known as the Influence Maximization Problem (IMP), is shown to be NP-hard [11]. It is difficult to efficiently choose the optimal seed set, especially with a limited budget [5]. Nevertheless, a number of heuristics and algorithms are presented in the literature, see Sect. 2.3 for details.

In this paper, we introduce two new seeding strategies for the IMP problem. The first strategy CVSP is based on the *connectivity* of the graph, i.e., the decomposition from one-connected graphs into biconnected components, and then from biconnected graphs into triconnected components [6, 7]. Specifically, we use *cut vertices* as well as *separation pairs* as the starting seeds. The second strategy ER is based on the *spectral* graph theory. Specifically, we use a vertex ranking defined by the *effective resistance* values [24, 25].

Extensive comparison experiments using real-world data sets demonstrate that our new seeding strategies outperform the existing methods [3, 8, 14, 17, 20], such as various centrality measures, k -core index as well as the state-of-the-art IMM [26], in particular for the scale-free networks with globally sparse, locally dense clusters with short diameter.

Specifically, we experiment using the most widely used diffusion model, Independent Cascade (IC), and compare the final influence spread rate with existing seeding strategies. For scale-free graphs with small diameters, ER and CVSP outperform, with the final influence spread rate of 71.9% and 67.6% respectively, compared to IMM with 50.5%. For biological networks with long diameter, IMM performs the best with the final influence spread of 52.1%, followed by ER and CVSP with 35.1% and 29.5%, respectively.

Since cut vertices and separation pairs can be computed in linear time [6, 7], and the effective resistance values can be computed in near linear time [24], our new seeding strategies CVSP and ER are not only effective, but also efficient, matching the near linear time complexity of IMM [26]. Furthermore, visual analysis allows more sophisticated comparison between the methods, showing that our methods have more global influence spread pattern than other methods with local influence spread pattern.

2 Related Work

2.1 Influence Maximization Problem

The Influence Maximization Problem (IMP) [13] involves finding a set of seed vertices in a network such that the eventual spread of influence under a stochastic diffusion model is maximized. The problem is shown as NP-hard [11] and has specific applications in Viral Marketing [4, 19, 22].

Roughly speaking, the spread of influence in a network under a diffusion model is modelled as follows: (i) A set of k seed vertices are chosen and *activated*; (ii) A diffusion model, which imitates the spread of influence, repeatedly runs

through a new set of activated vertices at each diffusion step, until the spread converges (i.e., no new vertices are activated). Let $I(S)$ denote the number of activated vertices when the diffusion process finishes using the initial set of activated vertices, the *seed set* S . Let $G = (V, E, \omega)$ be a graph with a vertex set V and the edge set E , where $\omega(u, v)$ represents the weight of an edge (u, v) . Let M be a stochastic diffusion model with a positive integer k , representing the budget. Formally, the Influence Maximization Problem asks to find a seed set S^* such that $S^* \subseteq V \wedge |S^*| = k$ maximizes $\mathbb{E}[I(S^*)]$.

2.2 Diffusion Models

The most widespread influence propagation diffusion models are the Independent Cascade model and the Linear Threshold model, which are used as a general framework for IMP [12].

The *Independent Cascade (IC)* model generalizes the Susceptible-Infected-Recovered (SIR) model, one of the most ubiquitous diffusion models [23]. In these models, all vertices are initially susceptible (able to be infected) except for the infected vertices (the initial seed vertices), which only have a single chance to infect (influence) their neighbors.

At each time step k , an infected vertex u will independently infect each of its susceptible neighbors v with probability β in the SIR model, and $\beta_{u,v}$ in the IC model. At step $k + 1$, every infected vertex from the previous step enters the recovered state. This diffusion process ends once there are no more infected vertices and thus no more possible activation. The spreading influence of the initial seed set is defined as the number of resulting recovered vertices [20]. The IC model is often used to simulate influence propagation and it models, for example, how independent recommendations and reviews may induce a purchase [15].

In contrast, the *Linear Threshold (LT)* model describes a process in which all vertices that were active in the last time step remain active, and thus each vertex's tendency to become active increases monotonically [13]. This is in line with the notion that an individual's perceived value of a product increases as more people in their network adopt the product.

2.3 Seeding Strategies for the IMP

Kempe et al. [13] introduce the two diffusion models IC and LT, and then show that under these two models, the IMP problem is NP-Hard. They propose a greedy algorithm with a guarantee that the final influence spread is within $(1 - \frac{1}{e} - \epsilon)$, where ϵ is a tunable parameter. However, the greedy algorithm has poor scalability. Leskovec et al. [16] present the CELF algorithm, using the submodularity properties of the influence spread function for both IC and LT models to terminate the greedy algorithm early, with a 700 times speed up with the same performance guarantee.

Since then, a plethora of methods have been developed to solve the IMP, either heuristics or with performance guarantees for the sake of runtime. For example, IMM by Tang et al. [26] is one of the most well-known algorithms

considered as the state-of-the-art, based on the Reverse Influence Sampling (RIS) framework by Borgs et al. [2], which runs in near linear time, i.e., $O((k+l)(n+m) \log n/\epsilon^2)$, where l and ϵ are tunable parameters, $n = |V|$, $m = |E|$, $k = |S|$. IMM can return a $(1 - \frac{1}{e} - \epsilon)$ approximation with at least $1 - \frac{1}{n^t}$ probability.

Since the IMP is NP-hard, many heuristic algorithms are presented, where the typical method is to rank all vertices according to a certain measure and select the k highest ranking vertices as the initial seed set. With the goal of identifying the most important and connected vertices, natural measures to utilize are *centrality* measures and k -core index from social network analysis [27].

The *Degree centrality* of a vertex, defined as the number of adjacent vertices, is a simple but intuitive index used to identify a vertex's influence. Despite its simplicity and low computation cost, Liu et al. [17] report that when the spreading rate is small, Degree centrality outperforms Eigenvector centrality and the k -core index. However, Chen et al. [3] argue that Degree centrality has low accuracy and propose the *Semi-Local Centrality* (SLC) measure, which considers both the nearest and the next nearest neighbors of a vertex.

The *Closeness centrality* of a vertex is defined based on the sum of the shortest paths from the vertex to all the other vertices, to reflect how efficiently a vertex exchanges information with others [20]. Closeness centrality has been shown to outperform Degree and Betweenness centrality in terms of spreading influence [3]. However, computing the Closeness centrality requires significantly more time (i.e., $O(n^3)$ time [9]) than other measures, such as Degree centrality and k -core, its practical application can be limited.

The *Betweenness centrality* of a vertex is defined based on the number of the shortest paths that pass through the vertex, representing its potential power in controlling the information flow in a network. In contrast to Chen et al. [3], Lu et al. find that Betweenness centrality outperforms Closeness centrality in terms of the ratio of final recovered (infected) vertices [20]. Unfortunately, the high computational complexity of Betweenness centrality (i.e., $O(n^3)$ for general graphs; for sparse graphs, $O(nm)$ time for unweighted graphs and $O(nm + n^2 \log n)$ for weighted graphs [9]) means that for large networks, its scalability is infeasible.

The k -core of a graph G is the maximal subgraph where all vertices have degree at least k . Specifically, the k -core decomposition assigns an integer (coreness) k_s to each vertex, representing its location in the network based on its k -shell layer, where a vertex with larger coreness indicates that it is located in a more central, *core* position. The efficiency (i.e., linear time [1]) of k -core decomposition allows it to be applied to large-scale networks. Moreover, Kitsak et al. [14] find that the most efficient spreaders in the SIR model are those in the core of the network as identified by the k -core analysis, rather than vertices that are highly connected. However, Liu, et al. [18] find that due to the low granularity of k -core decomposition, higher k -core vertices often link locally to other vertices within the shell and, as such, are not good spreaders.

3 New Seeding Strategies

3.1 CVSP: Connectivity-Based Seeding Strategy

Our first strategy, CVSP (Cut Vertex Separation Pair), is based on the *connectivity* of the graph, specifically the decomposition from one-connected graphs into biconnected components, and then from biconnected graphs into triconnected components.

A *cut vertex* of a connected graph G is a vertex whose removal from G increases the components in G , i.e., disconnects [6]. Similarly, a *separation pair* of a biconnected graph G is a pair of vertices whose removal from G increases the components in G , i.e., disconnects [7]. Note that the cut vertices and separation pairs can be computed in linear time [6, 7].

Cut vertices and separation pairs are structurally important in terms of graph connectivity and topology, since the removal of such vertices disconnect the remaining graph, meaning that they disable communication and increase the vulnerability. In particular, cut vertices have been shown to be important actors in social network analysis, such as brokers, often with high betweenness centrality [27], indicating their potential use as seed vertices for the IMP problem.

To compare the performance of CVSP to other seeding strategies, such as various centrality measures, with resource restrictions, such as the budget which are conventionally modelled by the number of starting seed vertices k , we need to compute a vertex ranking by sorting the cut vertices and the separation pairs based on their degree. Specifically, CVSP can be described as follows:

1. Compute cut vertices of a one-connected graph G and sort them in the decreasing order of their degree to form the set CV . Ties are broken randomly to avoid introducing bias.
2. For each biconnected component B_i of G , compute the separation pairs to form the set SP . Sort the vertices in SP based on their degree, as in Step 1.
3. Compute the ordered *connectivity set* C by attaching SP at the end of CV . Select the top k vertices from C as the starting seed set.

3.2 ER: Spectral Seeding Strategy

Our second strategy, ER (Effective Resistance), is based on the spectral graph theory, which is concerned with the eigenvalues and eigenvectors of matrices associated with graphs. The *spectrum* of a graph is the list of eigenvalues of its Laplacian matrix L , which is defined using the Degree matrix D and Adjacency matrix A as $L = D - A$. The spectrum of a graph is closely related to a number of structural properties of graphs, such as the connectivity.

For example, a spectral sparsifier is a subgraph whose Laplacian quadratic form is approximately the same as the original on all real vector inputs. Spielman and Teng [25] proved that every n -vertex graph has a spectral approximation with $O(n \log n)$ edges. Specifically, *spectral sparsification* is a stochastic sampling method, where edges are sampled based on the *effective resistance* values, which is related to the commute distances in the graph.

Modelling a graph as an electrical network, the effective resistance of an edge is defined as the voltage drop across the edge and its value is equivalent to the probability of the edge to be included in a random spanning tree of the graph [24]. Note that the effective resistance values of edges in a graph can be computed in near linear time [24].

Since effective resistance values are closely related to the *commute distance* (i.e., the expected length of a random walk between two vertices), as well as *structural roles (positions)* of actors based on how they are embedded in a social network [27], these indicate potential use as seed vertices for the IMP problem. Specifically, ER can be described as follows:

1. Compute the effective resistance value $ER(v)$ of each vertex v as the sum of the effective resistance value $ER(e)$ of each incident edge e of v : i.e. $ER(v) = \sum_{e \in I(v)} ER(e)$, where $I(v)$ is the set of incident edges of v .
2. Compute a vertex ranking R in a decreasing order of $ER(v)$. Select the top k vertices from the vertex ranking R as the initial seed set.

4 Comparison Experiments

4.1 Experiment Design, Implementation and Data Sets

We compare our two new seeding strategies CVSP and ER against the following popular seeding strategies, described in Sect. 2.3: BC (Betweenness Centrality), CC (Closeness Centrality), DC (Degree Centrality), KC (k -core) and IMM.

We implement CVSP and ER, and diffusion simulations in Python. For the IC model, we use an implementation from [3], and initialize $\beta_{u,v}$ values to 0.5. We use NetworkX to compute centrality measures and k -core, and OGDF to compute the cut vertices, separation pairs and graph layouts. For IMM, we use implementation from [26] with the recommended setting of the tunable parameter $\epsilon = 0.5$.

We experiment with two types of real-world datasets: (1) *scale-free networks*: globally sparse, locally dense clusters with the Power law degree distribution and *short* diameters; (2) biological networks (RNA sequence graphs): globally sparse (i.e. path or almost tree structure), locally dense clusters with *long* diameters, called the *GION graphs* [21]. For the details of the data sets, see Table 1.

For each graph, we run the Independent Cascade (IC) diffusion model with each seeding strategy five times and take the average, due to the non-deterministic nature of the model. We set the budget (i.e., the number of starting seed vertices) as $k = |CV| + |SP|$.

4.2 Final Influence Spreading Rate Comparison

Table 1 and Fig. 1 show the *final influence spreading rate* (i.e., percentage of the activated vertices over the total number of vertices after IC converges). Clearly, ER performs the best on the scale-free graphs, followed by BC and CVSP, and DC performs on par with CVSP. Interestingly, IMM performs the worst for

Table 1. The average final influence spread rate (i.e., percentage of the activated vertices over the total number of vertices in the graph), where the number of starting seed vertices $k = |CV| + |SP|$, den means the density, and dia means the diameter of a graph. ER performs the best for scale-free graphs, followed by BC and CVSP; IMM performs the best for GION graphs, followed by ER and CVSP.

Dataset	BC	CC	DC	KC	CVSP	ER	IMM	$ V $	$ E $	den	dia	k
soc_h	58.8	53.2	54.8	51.2	53	60	33.8	2000	16097	8.05	10	326
yeastppi	83.8	74.4	81.6	73.8	83.6	85.6	59.4	2224	7049	3.17	11	674
offlights	81.2	65.4	75.6	68.2	76.4	81.4	44.2	2905	15645	5.39	14	709
facebook	49.2	47	45.6	9.8	46.8	49.2	52.6	4039	88234	21.8	8	109
p2p-Gnutella05	80.4	72.8	77.6	73.2	78.4	83.2	62.4	8842	31837	3.60	9	2636
Avg. Scale-free	70.68	62.56	67.04	55.24	67.64	71.88	50.48					
gion2	31	17.4	26.4	20.6	32.6	39.4	53.2	1159	6424	5.54	58	183
gion5	37.4	23.2	23.8	20.2	38.4	46.8	57.8	1748	13957	7.98	63	270
gion6	29.8	18.4	26.8	17.4	30.8	34	51.6	1785	20459	11.5	41	198
gion7	13.8	8.4	13.8	10.2	20.2	21.2	46.2	3010	41757	13.9	77	161
gion8	28.6	12.4	23.4	16.8	25.4	34	51.8	4924	52502	10.7	121	491
Avg. GION	28.12	15.96	22.84	17.04	29.48	35.08	52.12					

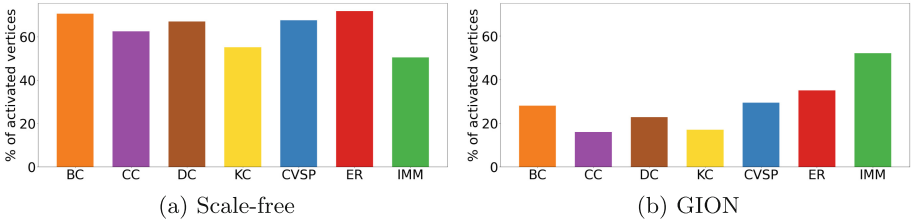


Fig. 1. Average final influence spread rate (i.e., percentage of activated vertices): ER performs the best for scale-free graphs; IMM performs the best for GION graphs.

scale-free graphs. On the other hand, IMM performs the best on the GION graphs, followed by ER and CVSP. Overall, experiments demonstrate excellent performance by ER and CVSP, in particular for scale-free graphs. Note that the time complexity of CVSP and ER is much faster than BC.

Figure 2 shows how fast the diffusion is carried out for different seeding strategies, and where the rate of diffusion peaks, by showing the number of newly activated vertices at each diffusion step. In terms of the influence rate, we observe that most of the diffusion occurs in the first few steps across all seeding strategies. The highest influence rate occurs at the second (or the third) diffusion step, and then drops quickly in particular for scale-free networks. Therefore, what determines which method will activate the most vertices after convergence is which method activates the most vertices in the first few diffusion steps. As such, the role of the starting seed vertices is critical.

Figure 3 shows the final influence rate after convergence, where the number of starting seed vertices k is varied. Clearly, the relative performances of seeding

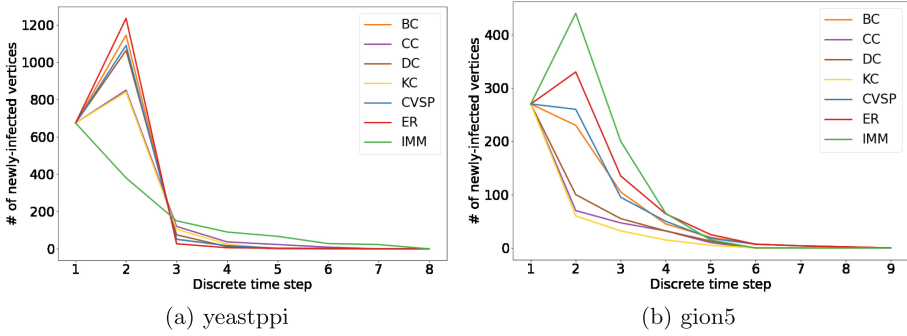


Fig. 2. Dynamics of diffusion: the number of newly activated vertices at each diffusion step. For all strategies, most of the diffusion occurs in the second step.

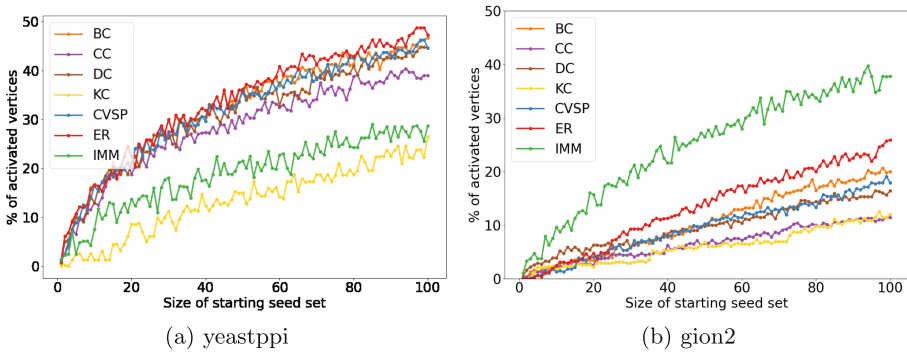


Fig. 3. Final influence rate after convergence, where the number of starting seed vertices k is varied. In general, the relative performances of seeding strategies do not change, based on the size of k .

strategies do not change, based on the size of k . We observe that varying the number of starting seed vertices k has minimal effect on the results observed in Table 1. Roughly speaking, when a particular seeding strategy is strong relative to the rest, it remains strong for all k values, and when two seeding strategies are comparable, they remain comparable for all k .

4.3 Visual Analysis and Comparison

Visual analysis enables more refined comparison between the methods other than the efficiency and effectiveness. For example, in terms of efficiency, CVSP, DC and KC all run in linear time. For scale-free graphs, Table 1 shows that both CVSP and DC perform on par in terms of effectiveness (i.e., influence spreading rate). However, visual comparison in Fig. 4 reveals that the seed vertices by CVSP have a more global influence compared to the seed vertices by DC or KC with relatively local influence.

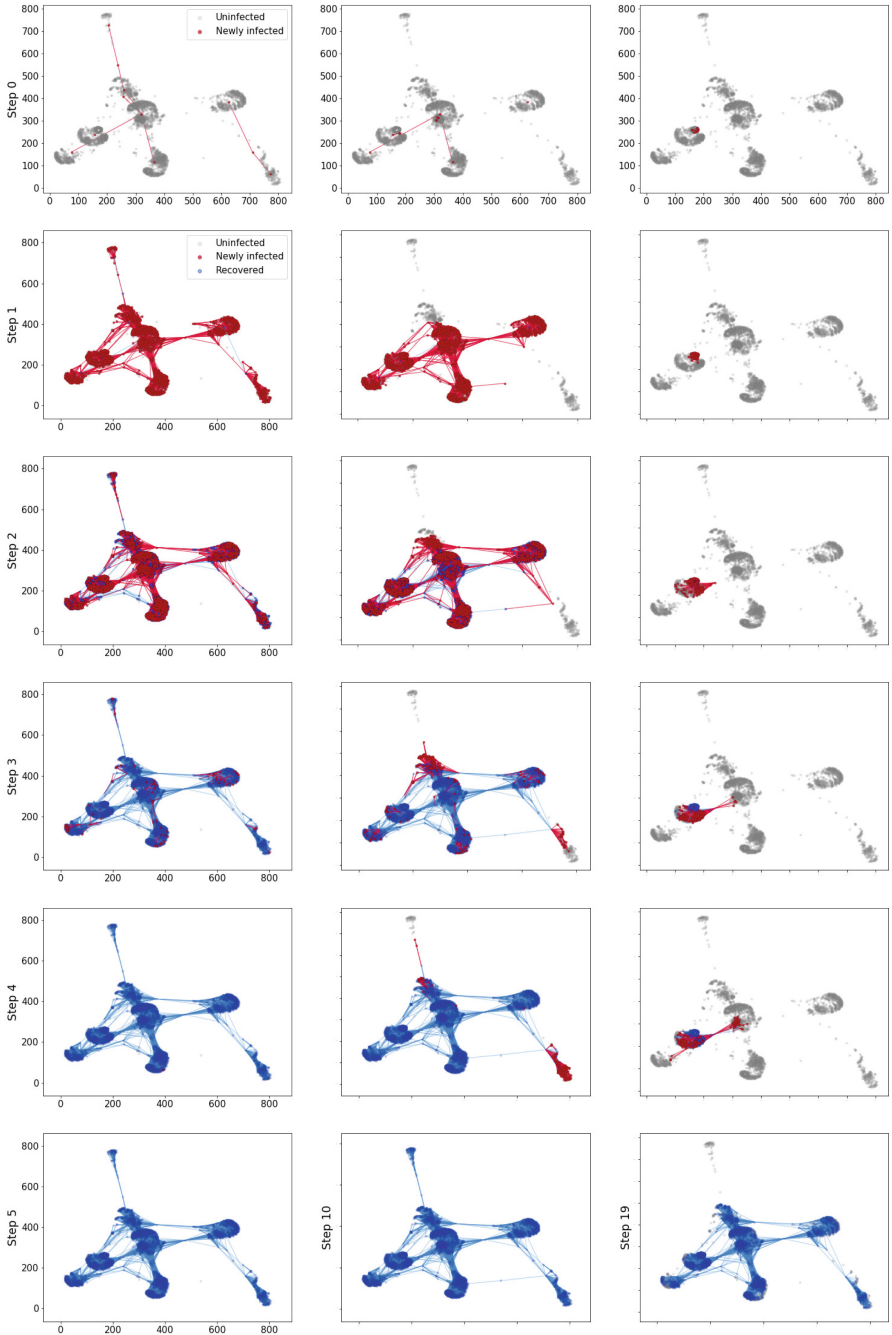


Fig. 4. Visual analysis and comparison of the influence spread by CVSP (left), DC (middle) and KC (right) on the facebook network. Note that the final diffusion step in the last row is 5 (CVSP), 10 (DC) and 19 (KC), respectively. Red (resp., Blue) means newly (resp., previously) activated vertices/edges, and Grey means not yet activated.

Specifically, we can observe that in the first two steps, CVSP produces more widely spread influenced vertices (in red), compared to more concentrated, sustained local influence spread by DC and KC, suggesting that their seed vertices are highly locally linked, leading to redundancy. Moreover, the final influence (in blue recovered vertices) of CVSP, DC and KC are achieved in 5, 10 and 19 diffusion steps respectively (in the last row), implying that the initial seed vertices by CVSP are more effective influencers than the seeds by DC and KC.

4.4 Summary and Recommendation

In summary, extensive comparison experiments demonstrate that our new seeding strategies CVSP and ER are not only efficient with linear and near-linear time complexity, but also effective in terms of the influence spreading rate, as shown in Table 1. Overall, ER and CVSP outperform IMM for scale-free graphs with globally sparse, locally dense clusters with ultra small diameters. On the other hand, IMM performs the best for GION graphs, followed by ER and CVSP.

We also conducted experiments with the LT diffusion model, and obtained similar results. Based on our extensive empirical experiments, we conclude with the following recommendation and guidelines:

- For scale-free graphs with globally sparse, locally dense clusters and short diameters, we recommend ER for both effectiveness and efficiency.
- For graphs with globally sparse, locally dense clusters and long diameters, we recommend IMM for both effectiveness and efficiency.
- When efficiency is the most important consideration, we recommend CVSP over DC or KC, for global influence spread pattern, based on visual analysis and comparison.

References

1. Batagelj, V., Mrvar, A., Zaveršnik, M.: Partitioning approach to visualization of large graphs. In: GD 1999, pp. 90–97 (1999)
2. Borgs, C., Brautbar, M., Chayes, J., Lucier, B.: Maximizing social influence in nearly optimal time. In: SODA 2014, pp. 946–957 (2014)
3. Chen, D., Lü, L., Shang, M.S., Zhang, Y.C., Zhou, T.: Identifying influential nodes in complex networks. *Phys. A* **391**(4), 1777–1787 (2012)
4. Domingos, P., Richardson, M.: Mining the network value of customers. In: SIGKDD 2001, pp. 57–66 (2001)
5. Gui-sheng, Y., Ji-jie, W., Hong-bin, D., Jia, L.: Intelligent viral marketing algorithm over online social network. In: ICNDC 2011, pp. 319–323 (2011)
6. Hopcroft, J., Tarjan, R.: Algorithm 447: efficient algorithms for graph manipulation. *Commun. ACM* **16**(6), 372–378 (1973)
7. Hopcroft, J.E., Tarjan, R.E.: Dividing a graph into triconnected components. *SIAM J. Comput.* **2**(3), 135–158 (1973)
8. Iyer, S., Killingback, T., Sundaram, B., Wang, Z.: Attack robustness and centrality of complex networks. *PLoS ONE* **8**(4), e59,613 (2013)

9. Jacob, R., Koschützki, D., Lehmann, K.A., Peeters, L., Tenfelde-Podehl, D.: Algorithms for centrality indices. In: Brandes, U., Erlebach, T. (eds.) *Network Analysis*. LNCS, vol. 3418, pp. 62–82. Springer, Heidelberg (2005). https://doi.org/10.1007/978-3-540-31955-9_4
10. Kaplan, A.M., Haenlein, M.: Two hearts in three-quarter time: how to waltz the social media/viral marketing dance. *Bus. Horiz.* **54**(3), 253–263 (2011)
11. Kempe, D., Kleinberg, J., Tardos, É.: Maximizing the spread of influence through a social network. In: *SIGKDD 2003*, pp. 137–146 (2003)
12. Kempe, D., Kleinberg, J., Tardos, É.: Influential nodes in a diffusion model for social networks. In: *ICALP 2005*, pp. 1127–1138 (2005)
13. Kempe, D., Kleinberg, J., Tardos, É.: Maximizing the spread of influence through a social network. *Theory Comput.* **11**(4), 105–147 (2015)
14. Kitsak, M., et al.: Identification of influential spreaders in complex networks. *Nat. Phys.* **6**(11), 888–893 (2010)
15. Leskovec, J., Adamic, L.A., Huberman, B.A.: The dynamics of viral marketing. *ACM Trans. Web (TWEB)* **1**(1), 5–es (2007)
16. Leskovec, J., Krause, A., Guestrin, C., Faloutsos, C., Faloutsos, C., VanBriesen, J., Glance, N.: Cost-effective outbreak detection in networks. In: *SIGKDD 2007, KDD 2007*, pp. 420–429 (2007)
17. Liu, J.G., Lin, J.H., Guo, Q., Zhou, T.: Locating influential nodes via dynamics-sensitive centrality. *Sci. Rep.* **6**(1), 1–8 (2016)
18. Liu, Y., Tang, M., Zhou, T., Do, Y.: Core-like groups result in invalidation of identifying super-spreader by k-shell decomposition. *Sci. Rep.* **5**(1), 1–8 (2015)
19. Long, C., Wong, R.C.W.: Viral marketing for dedicated customers. *Inf. Syst.* **46**, 1–23 (2014)
20. Lü, L., Chen, D., Ren, X.L., Zhang, Q.M., Zhang, Y.C., Zhou, T.: Vital nodes identification in complex networks. *Phys. Rep.* **650**, 1–63 (2016)
21. Marnier, M.R., Smith, R.T., Thomas, B.H., Klein, K., Eades, P., Hong, S.H.: Gion: interactively untangling large graphs on wall-sized displays. In: *GD 2014*, pp. 113–124 (2014)
22. Richardson, M., Domingos, P.: Mining knowledge-sharing sites for viral marketing. In: *SIGKDD 2002*, pp. 61–70 (2002)
23. Shakarian, P., Bhatnagar, A., Aleali, A., Shaabani, E., Guo, R.: *Diffusion in Social Networks*. SCS, Springer, Cham (2015). <https://doi.org/10.1007/978-3-319-23105-1>
24. Spielman, D.A., Srivastava, N.: Graph sparsification by effective resistances. *SIAM J. Comput.* **40**(6), 1913–1926 (2011)
25. Spielman, D.A., Teng, S.H.: Spectral sparsification of graphs. *SIAM J. Comput.* **40**(4), 981–1025 (2011)
26. Tang, Y., Shi, Y., Xiao, X.: Influence maximization in near-linear time: a martingale approach. In: *SIGMOD 2015*, pp. 1539–1554 (2015)
27. Wasserman, S., Faust, K., et al.: *Social network analysis: methods and applications* (1994)
28. Zhu, T., Wang, B., Wu, B., Zhu, C.: Maximizing the spread of influence ranking in social networks. *Inf. Sci.* **278**, 535–544 (2014)