# A Greedy Heuristic for Majority Target Set Selection in Social Networks

1st Braully Rocha da Silva
*DTI-Reitoria*
*Instituto Federal Goiano*
Goiânia-GO, Brazil
braully.silva@ifgoiano.edu.br

2nd Erika Morais Martins Coelho
*Instituto de Informática*
*Universidade Federal de Goiás*
Goiânia-GO, Brazil
erikamorais@inf.ufg.br

3rd Hebert Coelho da Silva
*Instituto de Informática*
*name of organization (of Aff.)*
Goiânia-GO, Brazil
hebert@inf.ug.br

4th Fábio Protti
*Instituto de Computação*
*Universidade Federal Fluminense*
Niterói-RJ, Brazil
fabio@ic.uff.br

***Abstract*—The influence of individuals in a network and its propagation is dealt with in several studies in the literature. A well-known model is the majority target set, in which if most of the neighbors of an individual in the network are influenced, then the individual is also influenced. Finding a majority target set of minimum size is an NP-hard problem for general graphs. This paper proposes a heuristic for this problem, which has faster runtimes and achieves better solution values than related works, both on small random instances and on large real social network graphs.**

***Index Terms*—graph, majority target set selection, contamination spread, social network, heuristic algorithm**

## I. Introduction

Social networks have rapidly become a fundamental tool for communication, marketing, and propagation of information, opinions, trends, and fashion news. With billions of users and millions of interactions daily, they are a phenomenon whose study and understanding need analysis and mathematical modeling.

Some facets of social network studies include the maximization of influence, content dissemination, and viral marketing [1; 2]. These topics can be modeled in a variety of ways and perspectives. The selection of target sets plays a central role in social network analysis. In this paper, we focus on modeling social networks as large graphs and influence propagation as an irreversible graph process that amounts to solving the *target set selection problem* [2; 3].

In a social network, individuals are naturally associated with the nodes of a graph, and interactions/relationships between them with the edges. The influence process can be modeled as follows: some individuals are initially influenced; next, the remaining individuals are influenced as a given set of their neighbors becomes influenced. Choosing a small initial set (*target set*) of individuals that maximizes the number of influenced individuals is an important variant of the target set selection problem.

Two factors are relevant in distinguishing variants of target set selection problems. The first is the contamination of the individual, which can be permanent or temporary. The second is the *activation bound* or *threshold function* that sets the threshold of neighbors required for a node to become influenced. This function can be a percentage of neighbors, a fixed number of neighbors, or a combination of both with a likelihood factor.

Besides these two factors, we also mention the search for target sets with higher weight [4; 5], with longer activation time [6], and with minimum cardinality [7; 8].

Studies on the target set selection problem are supported by a strong graph theoretic framework. In [3; 9; 10] it is shown that it is an NP-hard problem for general graphs, when the activation bound is at least 2. There are also polynomial-time algorithms and bounds for certain classes of graphs, such as trees, block-cactus graphs, and chordal graphs[3; 11; 12; 13; 14; 15], and studies on the parameterized complexity of the problem [7; 16; 17].

The majority target set selection problem considers the threshold function of a node $v$ to be $f(v) = \lceil \frac{d(v)}{2} \rceil$. Majority target set selection is an important and well-studied variant. It has applications in many areas [18; 19; 20] such as epidemiology, marketing, social networks, economics, and fault tolerance. Aa an example, consider a computer network with a set of servers, where the servers are sized to take over the work of half of their neighbors in case of failure. Another application example would be to disseminate an opinion about a controversial bill in a social network. If an individual suspects that the bill is bad, and most of his/her neighbors

*ASONAM '23*, November 6-9, 2023, Kusadasi, Turkey
© 2023 Association for Computing Machinery.
ACM ISBN 979-8-4007-0409-3/23/11…$15.00
http://dx.doi.org/10.1145/3625007.3627726

also have this opinion, then the individual will also adopt this opinion. Finding the minimum set of such individuals capable of influencing the whole community is also an NP-hard problem[3; 9; 21].

We performed experiments with the best target set heuristics, and realized that they perform poorly for the majority target set problem. We evaluate some parameters for a greedy heuristic on a random database, choosing the best parameter. Finally, we propose a greedy heuristic that is able to find better solutions than the previous heuristics in reasonable running times.

### A. Previous results

In [2; 22], greedy algorithms for the target set problem are described. With slight variations from each other, such heuristics choose at each iteration the next available node of maximum degree. Some formulations with meta-heuristics have also been proposed [23].

The well-known heuristics by [24] and [25] use simple and scalable algorithms, based on graph decomposition strategies, as the target set is built. Both present good performances for large graphs, such as social networks.

### B. Our results

Considering the majority target set problem, we realize that the heuristics in [24] and [25] do not perform well. They are slower and generate lower quality solutions than those in [2; 22]. In addition, the heuristic in [25] does not perform well when the activation function is a percentage of neighbors.

We evaluated parameters for a greedy heuristic, which maximizes the performance for the majority target set. We also identified a parameter that greatly improves the results when compared with the heuristics in [2; 24; 25].

We repeated the experiments of the abovementioned heuristics and compared the results with our approach. We obtained better results regarding solution quality and execution time for the majority target set problem.

We also performed experiments for the target set problem with activation functions ranging from 10% to 70% of the node degree. This kind of experiment had already been performed in [24], but not repeated in [25]. In this variant, our greedy heuristic also obtains better results than both.

## II. Preliminaries and Definitions

A social network is represented by a graph $G$, its individuals are the nodes in $V(G)$, and its relationships are the edges in $E(G)$. Let $v$ be a node of $V(G)$. Its neighbors form the set $N(v)$, the degree of $v$ is $d(v) = |N(v)|$, the degree of $v$ in $S$ is $d_S(v) = |N(v) \cap S|$, the open neighborhood of $v$ is $N(v) = \{w \in V(G) | vw \in E(G)\}$, and the value of its threshold function is $f(v)$.

Given a subset $S \subseteq V(G)$, the *activation process* is given by the sequence $I^p[S]$ where $I^0[S] = S$, $I^1[S] = I[S] \cup \{v \in V | d_{I^0}(v) \geq f(v)\}$, and $I^p[S] = I[I^{p-1}]$ for $p \geq 2$. When for some $p$ we have $I^q[S] = I^p[S]$ for all $q \geq p$, the stabilization of the activation process occurs, i.e., the process stops.

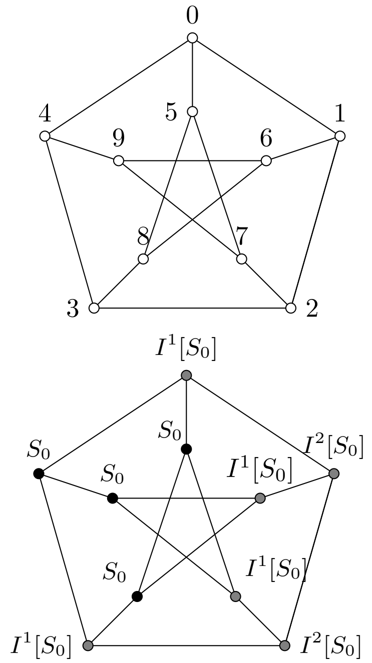

Figure 1. Petersen graph and its activation process in majority target set selection

According to the problem's specification, there are different threshold functions for the nodes; in this work we will only discuss the majority function: $f(v) = \lceil \frac{d(v)}{2} \rceil$.

Consider the graph $G$ in Figure 1 and the subset $S_0 = \{4, 5, 8, 9\}$. Since $G$ is a 3-regular graph, $f(v) = \lceil \frac{d(v)}{2} \rceil = 2$. At each step, we add the not-yet-influenced nodes that have exceeded the value of the threshold function, as follows:

- $I^0[S_0] = S_0 = \{4, 5, 8, 9\}$
- $I^1[S_0] = I[S_0] = \{0, 3, 4, 5, 6, 7, 8, 9\}$
- $I^2[S_0] = I[I^1[S_0]] = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$
- $I^3[S_0] = I[I^2[S_0] = I^2[S_0] = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$

Since $I^3[S_0] = V(G)$, $S_0$ is indeed a set that has activated all nodes of the graph, i.e., it is a target set.

In the graph of Figure 1, observe that there are target sets of cardinality smaller than $|S_0|$, for example $S_1 = \{4, 7, 8\}$.

## III. Algorithm

We propose a generic greedy algorithm that receives as input a graph $G = (V, E)$, a set of thresholds $t(v)$ for each node $v \in V$, and a precedence order $p$ of the parameters, to evaluate which node is chosen at each iteration.

The function $GREATER(x, y, p)$ evaluates whether the parameters of $x$ are greater than those of $y$, according to the precedence determined by $p$. For example, let $x = [degree = 3, delta = 2, distDelta = 1]$ and $y = [degree = 3, delta = 1, distDelta = 2]$. If $p = [degree, delta]$ then $GREATER(x, y, p) = False$, but if $p = [degree, distDelta]$ then $GREATER(x, y, p) = True$.

At the end of each iteration, the node with the maximum evaluation given by the $GREATER$ function is added to the set $S$. This cycle repeats until all nodes of the graph are influenced. At the end, the algorithm returns a set $S$ that is a majority target set.

**Algorithm 1** GenericGreedy

```
Require: $G = (V, E)$, Thresholds $t(v)$ for $v \in V$, precedence of parameters
Ensure: Majority Target Set $S$
$S = \varnothing$
$C = \varnothing$
while $|C| \leq |V|$ do
    $v = \varnothing$
    $max\varnothing$
    for all $u \in V \setminus C$ do
        $up = \varnothing$
        $uC = I^*[S \cup \{u\}]$
        $u\Delta = uC \setminus C$
        $up['degree'] = d(u)$
        $up['difficulty'] = t(u) - d_C(u)$
        $up['dist'] = (du - t(u)) - d_C(u)$
        $up['partial'] = |N(uC) \setminus N(C)|$
        $up['delta'] = |u\Delta|$
        $up['degreeDelta'] = |N(u\Delta)|$
        $up['difDelta'] = \sum_{i \in u\Delta}(t(i) - d_C(i))$
        $up['distDelta'] = \sum_{i \in u\Delta}((d(i) - t(i)) - d_C(i))$
        if $v = \varnothing \| GREATER(up, max, p)$ then
            $v = u$
            $max = up$
        end if
    end for
    $S = S \cup \{v\}$
    $C = I[S]$
end while
for all $v \in S$ do
    if $d_S(v) \geq f(v)$ then
        $S \leftarrow S \setminus \{v\}$
    end if
end for
Return $S$
```

### A. Parameter selection

Which combination of parameters produces smaller target sets? To answer this question, we generated a random database with node sizes from 50 to 150, and percentages from 0.1 to 0.9, tested with all combinations of size up to two, for all generated parameters, and compared the results with the algorithms in [24] and [25].

Table I shows the results for the parameters that did better than [24; 25]. We chose the parameter *difDelta* because it appears in the list of best results for percentage 50% (column "0.5") and the maximum sum of best results than the reference heuristics.

We can now build an optimized algorithm for the *difDelta* parameter. Despite the good results, this algorithm can be computationally expensive and have poor time performance. However, we can improve the execution time with caching and some pruning in the search for the bests vertices.

Table I
RESULT FOR EXPLORATION PARAMETERS

| **Algorithm** | **percentage of neighbors** | | | | | | | | **Total** |
|---|---|---|---|---|---|---|---|---|---|
| | ***0.5*** | ***0.1*** | ***0.2*** | ***0.3*** | ***0.6*** | ***0.7*** | ***0.8*** | ***0.9*** | |
| delta-difDelta | 101 | 234 | 171 | 0 | 117 | 117 | 127 | 127 | 994 |
| delta-difficulty | 99 | 234 | 171 | 0 | 117 | 120 | 128 | 127 | 996 |
| delta-dist | 100 | 243 | 179 | 149 | 74 | 0 | 0 | 0 | 745 |
| delta-distDelta | 101 | 243 | 179 | 149 | 77 | 0 | 0 | 0 | 749 |
| delta-partial | 118 | 245 | 188 | 0 | 126 | 115 | 116 | 110 | 1018 |
| difDelta | 112 | 234 | 168 | 0 | 125 | 133 | 140 | 140 | 1052 |
| DifDelta-degDelta | 105 | 234 | 171 | 0 | 123 | 127 | 137 | 142 | 1039 |
| difDelta-delta | 103 | 234 | 170 | 0 | 122 | 129 | 136 | 139 | 1033 |
| difDelta-difficulty | 102 | 234 | 169 | 0 | 120 | 136 | 140 | 144 | 1045 |
| difDelta-dist | 100 | 234 | 166 | 0 | 92 | 110 | 136 | 144 | 982 |
| difDelta-distDelta | 106 | 234 | 169 | 0 | 97 | 119 | 140 | 155 | 1020 |
| difDelta-grau | 105 | 234 | 171 | 0 | 123 | 127 | 137 | 142 | 1039 |
| difDelta-partial | 107 | 234 | 170 | 0 | 125 | 133 | 139 | 142 | 1050 |
| distDelta | 122 | 243 | 180 | 152 | 0 | 0 | 0 | 0 | 697 |
| DistDelta-degDelta | 114 | 243 | 180 | 152 | 0 | 0 | 0 | 0 | 689 |
| distDelta-delta | 114 | 243 | 179 | 152 | 0 | 0 | 0 | 0 | 688 |
| distDelta-difDelta | 108 | 247 | 187 | 148 | 0 | 0 | 0 | 0 | 690 |
| distDelta-difficulty | 103 | 247 | 187 | 148 | 0 | 0 | 0 | 0 | 685 |
| distDelta-dist | 120 | 243 | 180 | 152 | 0 | 0 | 0 | 0 | 695 |
| distDelta-grau | 114 | 243 | 180 | 152 | 0 | 0 | 0 | 0 | 689 |
| distDelta-partial | 112 | 243 | 180 | 153 | 0 | 0 | 0 | 0 | 688 |

The values of $difDelta(v)$ may be cached for the next iteration if $v$ and none of its neighbors were contaminated, or partially contaminated, by the best vertex in the previous iteration. Finally, we can perform pruning on the evaluation of the vertices at each iteration, any vertex that has been contaminated by another vertex in the process of evaluating the best vertex in the current iteration can be discarded from the evaluation. For example, we are in iteration 1 of the algorithm and we have $v, u, \cdot, w$ vertices not activated, we compute then $difDelta(v)$ if $u \in \Delta(v)$ we can discard the evaluation of $u$ because $\Delta(u) \subseteq \Delta(v)$, consequently $difDelta(u) \leq difDelta(v)$.

With these optimizations, we implemented a new heuristic and compared our results with the algorithms in [24] and [25], using the same social network datasets of their research.

## IV. RESULTS

To compare our results with previous studies, we used a dataset of random graphs and a dataset of real social network graphs.

The random graph dataset is composed of [26] graphs, from 5 to 100 nodes and density ranging from 0.1 to 0.9. This dataset was used to find the best choice of parameters for the greedy heuristic.

We performed two sets of experiments with the social network datasets.

Table II
Result for Majority TSS

| Graph | n | TSSC | | TIP-Dec. | | GreedyDifTotal | | |
|---|---|---|---|---|---|---|---|---|
| | | *\|TSS\|* | *T(m)* | *\|TSS\|* | *T(m)* | *\|TSS\|* | *T(m)* | Δ% |
| BlogCatalog | 2093195 | 977 | 43 | 4598 | 44 | 353 | 22 | 64 |
| BlogCatalog2 | 1668647 | 527 | 44 | 4391 | 46 | 175 | 23 | 67 |
| BlogCatalog3 | 333983 | 274 | 44 | 834 | 46 | 127 | 23 | 54 |
| BuzzNet | 2763066 | 1445 | 45 | 7080 | 47 | 186 | 24 | 87 |
| Delicious | 145049 | 1725 | 46 | 4572 | 48 | 743 | 24 | 57 |
| Digg | 2011447 | 5554 | 2 | 5402 | 2 | 2905 | 4 | 46 |
| Douban | 327094 | 5155 | 48 | 32996 | 50 | 3657 | 27 | 29 |
| Flickr | 5899882 | 4352 | 3 | 6671 | 3 | 2984 | 42 | 31 |
| Foursquare | 1996522 | 36 | 5 | 2074 | 413 | 18 | 42 | 50 |
| Hyves | 2777176 | 50708 | 297 | 329883 | 714 | 22503 | 216 | 56 |
| Last.fm | 1043029 | 5913 | 49 | 6534 | 52 | 1934 | 33 | 67 |
| LiveJournal | 12816184 | 60523 | 1315 | 150551 | 1796 | 13045 | 927 | 78 |
| Livemocha | 2193083 | 3227 | 51 | 11279 | 53 | 1226 | 41 | 62 |
| YouTube | 76765 | 867 | 1315 | 1439 | 1796 | 365 | 928 | 58 |
| YouTube2 | 2990443 | 49861 | 1533 | 137315 | 2025 | 36807 | 1000 | 26 |
| ca-AstroPh | 198050 | 2476 | 51 | 2569 | 53 | 1523 | 42 | 38 |
| ca-CondMat | 93439 | 3586 | 51 | 4252 | 53 | 2617 | 42 | 27 |
| ca-GrQc | 14484 | 1060 | 51 | 1173 | 53 | 940 | 42 | 11 |
| ca-HepPh | 118489 | 1894 | 51 | 1911 | 53 | 1409 | 42 | 26 |
| ca-HepTh | 25973 | 1552 | 51 | 1796 | 53 | 1234 | 42 | 20 |
| ***Total time (min)*** | | 5095 | | 7400 | | 3586 | | 30 |

In the first experiment we used only the majority target set, with all the graphs used by [25] plus some used only by [24], available at [27; 28]. In table II we show the results of this experiment. Our heuristic produced lower values than the best results obtained by [25; 24] and in less time, on average 30%. For example, for the social network graph Buzznet[27] the proposed heuristic found a target set of size 186, while the best result between [24] and [25] was 1445, which gives an improvement of 87%.

In Table II we show the results of the first experiment for the majority target set. We can observe that the results achieved by the proposed heuristic are better than the results obtained in [24] and [25], spending less computational time.

The results of the second experiment are shown in Figures 2 to 16. It was run on the intersection of the datasets from [25] and [24] (fourteen instances), and using a threshold function ranging from 10% to 70% of the number of neighbors. Figure 2 shows that the proposed heuristic was in average 30% faster. Moreover, it presented better solution values in all the fourteen instances (Figures 2 to 15).

## V. Conclusions

The heuristic proposed in this work proved to be faster and achieved better results than related works. It basically explores greedy strategies, which can occasionally lead to locally optimal solutions. As a next step, we can incorporate strategies based on random choices and simulated annealing, in order to escape from local optima.

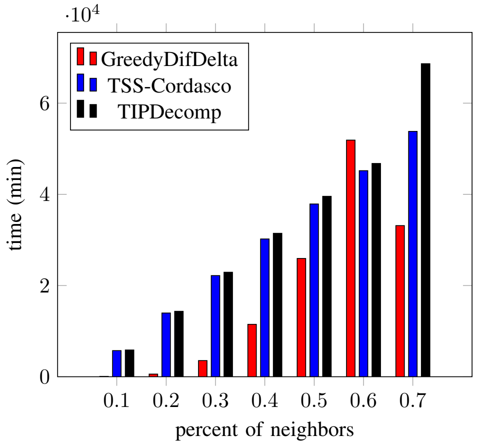

Figure 2. Total execution time

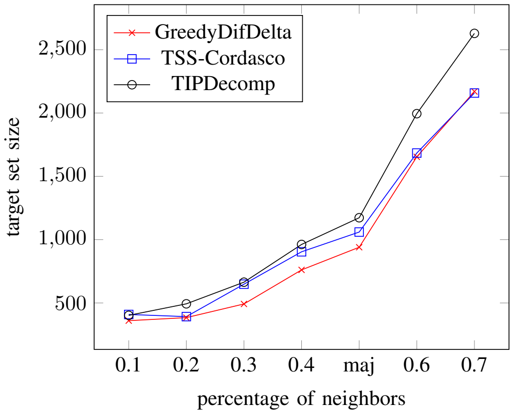

Figure 3. Result for ca-GrQc

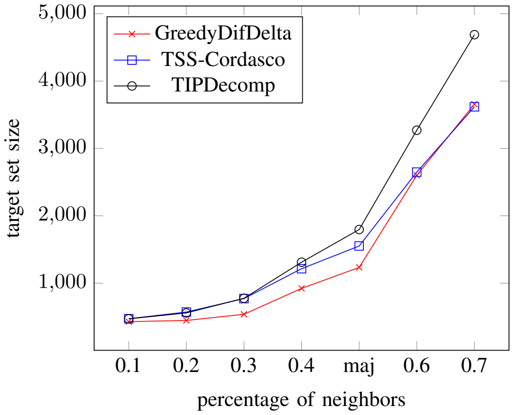

Figure 4. Result for ca-HepTh

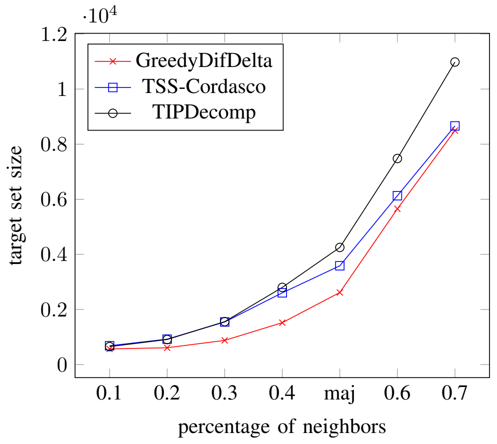

Figure 5. Result for ca-CondMat

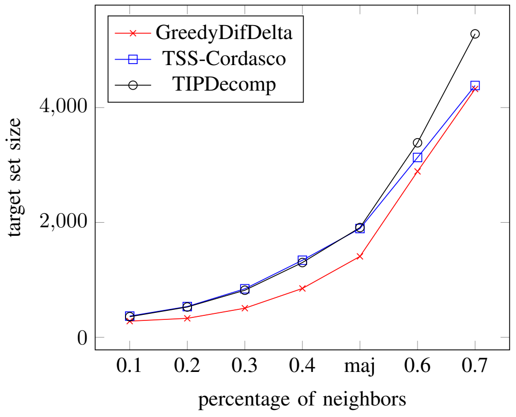

Figure 6. Result for ca-HepPh

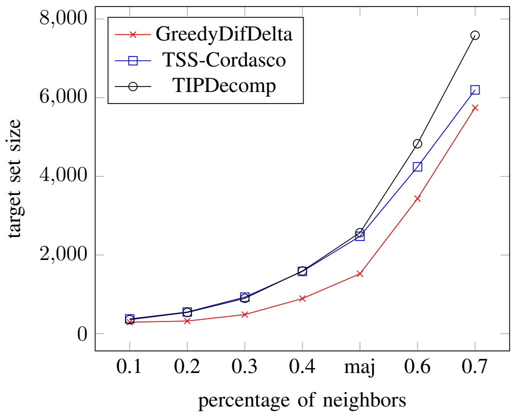

Figure 7. Result for ca-AstroPh

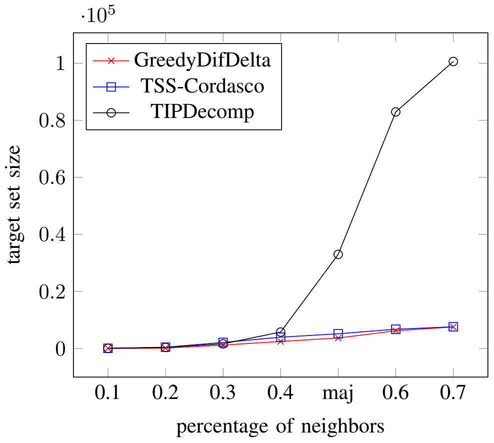

Figure 8. Result for Douban

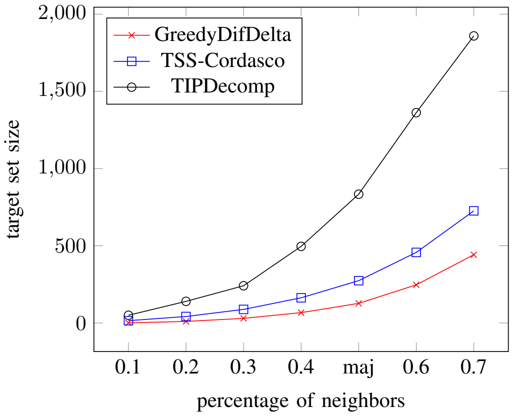

Figure 9. Result for BlogCatalog3

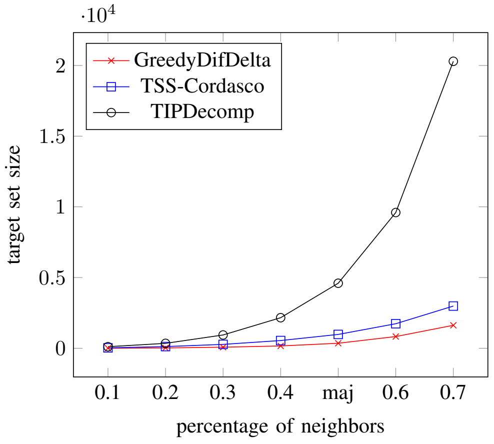

Figure 10. Result for BlogCatalog

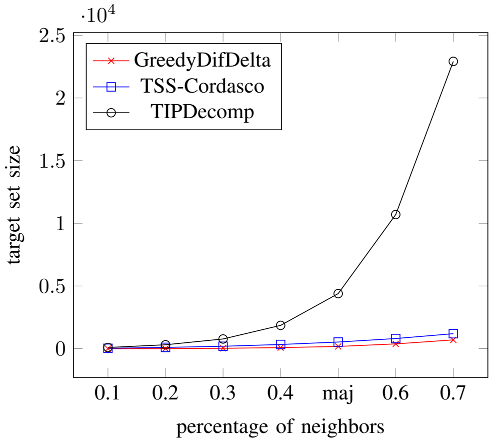

Figure 11. Result for BlogCatalog2

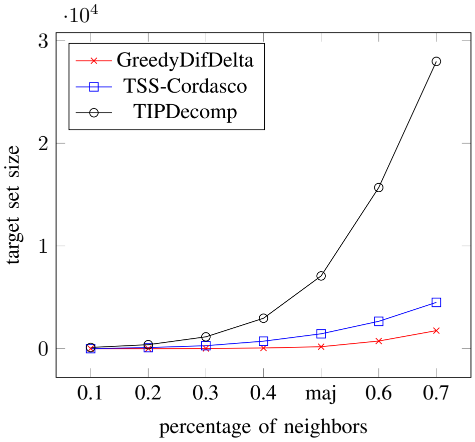

Figure 14. Result for BuzzNet

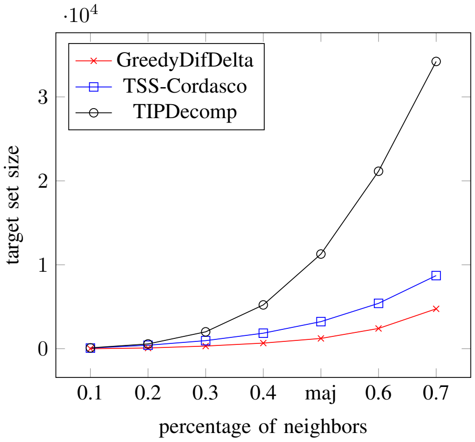

Figure 12. Result for Livemocha

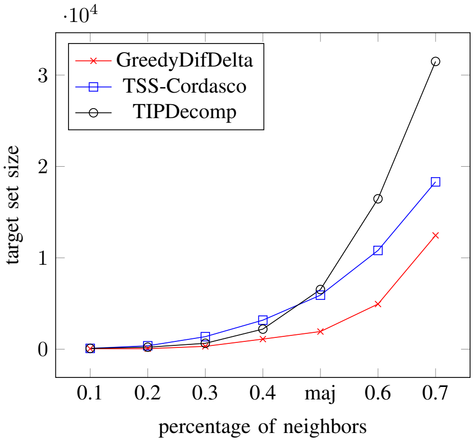

Figure 15. Result for Last.fm

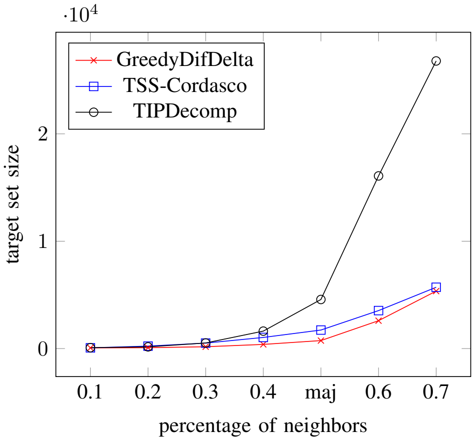

Figure 13. Result for Delicious

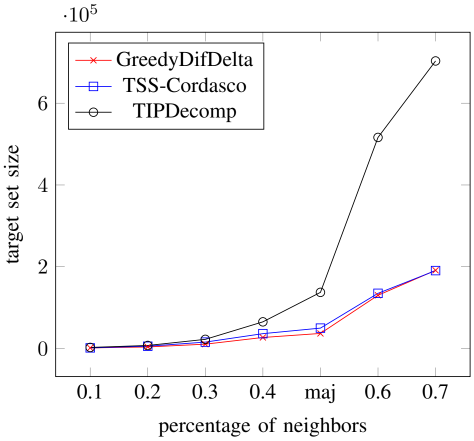

Figure 16. Result for YouTube2

## References

[1] M. Richardson and P. Domingos, "Mining Knowledge-Sharing Sites for Viral Marketing," *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2002.

[2] D. Kempe, J. Kleinberg, and E. Tardos, "Maximizing the spread of influence through a social network," in *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '03. New York, NY, USA: Association for Computing Machinery, 2003, p. 137–146.

[3] N. Chen, "On the approximability of influence in social networks," *SIAM Journal on Discrete Mathematics*, vol. 23, no. 3, pp. 1400–1415, 2009. [Online]. Available: https://doi.org/10.1137/08073617X

[4] S. Raghavan and R. Zhang, "A branch-and-cut approach for the weighted target set selection problem on social networks," *INFORMS Journal on Optimization*, vol. 1, no. 4, pp. 304–322, oct 2019. [Online]. Available: https://pubsonline.informs.org/doi/10.1287/ijoo.2019.0012

[5] ——, "Weighted target set selection on trees and cycles," *Networks*, vol. 77, no. 4, pp. 587–609, 2021.

[6] L. Keiler, C. V. Lima, A. K. Maia, R. Sampaio, and I. Sau, "Target set selection with maximum activation time," *Procedia Computer Science*, vol. 195, pp. 86–96, 2021.

[7] P. Dvořák, D. Knop, and T. Toufar, "Target set selection in dense graph classes," *SIAM Journal on Discrete Mathematics*, vol. 36, no. 1, pp. 536–572, 2022. [Online]. Available: https://doi.org/10.1137/20M1337624

[8] E. Ackerman, O. Ben-Zwi, and G. Wolfovitz, "Combinatorial model and bounds for target set selection," *Theoretical Computer Science*, vol. 411, no. 44, pp. 4017–4022, 2010. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0304397510004561

[9] P. A. Dreyer and F. S. Roberts, "Irreversible k-threshold processes: Graph-theoretical threshold models of the spread of disease and of opinion," *Discrete Applied Mathematics*, vol. 157, no. 7, pp. 1615–1627, 2009. [Online]. Available: http://dx.doi.org/10.1016/j.dam.2008.09.012

[10] C. C. Centeno, M. C. Dourado, L. D. Penso, D. Rautenbach, and J. L. Szwarcfiter, "Irreversible conversion of graphs," *Theoretical Computer Science*, vol. 412, no. 29, pp. 3693–3700, 2011.

[11] A. Nichterlein, R. Niedermeier, J. Uhlmann, and M. Weller, "On tractable cases of target set selection," vol. 3, 12 2010, pp. 378–389.

[12] O. Ben-Zwi, D. Hermelin, D. Lokshtanov, and I. Newman, "Treewidth governs the complexity of target set selection," *Discrete Optimization*, vol. 8, pp. 87–96, 02 2011.

[13] M. Chopin, A. Nichterlein, R. Niedermeier, and M. Weller, "Constant thresholds can make target set selection tractable," vol. 55, 12 2012, pp. 120–133.

[14] C.-Y. Chiang, L.-H. Huang, B.-J. Li, J. Wu, and H.-G. Yeh, "Some results on the target set selection problem," *Journal of Combinatorial Optimization*, vol. 25, 11 2011.

[15] A. N. Zehmakan, "On the spread of influence in graphs," *Information and Computation*, vol. 281, p. 104808, 2021. [Online]. Available: https://doi.org/10.1016/j.ic.2021.104808

[16] M. Charikar, Y. Naamad, and A. Wirth, "On Approximating Target Set Selection," in *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2016)*, ser. Leibniz International Proceedings in Informatics (LIPIcs), K. Jansen, C. Mathieu, J. D. P. Rolim, and C. Umans, Eds., vol. 60. Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2016, pp. 4:1–4:16.

[17] T. A. Hartmann, "Target set selection parameterized by clique-width and maximum threshold," in *SOFSEM 2018: Theory and Practice of Computer Science*, A. M. Tjoa, L. Bellatreche, S. Biffl, J. van Leeuwen, and J. Wiedermann, Eds. Cham: Springer International Publishing, 2018, pp. 137–149.

[18] N. Linial, D. Peleg, Y. Rabinovich, and M. Saks, "Sphere packing and local majorities in graphs," in *[1993] The 2nd Israel Symposium on Theory and Computing Systems*, 1993, pp. 141–149.

[19] D. Peleg, "Size bounds for dynamic monopolies," *Discrete Applied Mathematics*, vol. 86, no. 2, pp. 263–273, 1998.

[20] E. Berger, "Dynamic Monopolies of Constant Size," *Journal of Combinatorial Theory, Series B*, vol. 83, no. 2, pp. 191–200, 2001. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0095895601920453

[21] D. Peleg, "Local majorities, coalitions and monopolies in graphs: A review," *Theoretical Computer Science*, vol. 282, pp. 231–257, 06 2002.

[22] 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*, ser. KDD '09. New York, NY, USA: Association for Computing Machinery, 2009, p. 199–208. [Online]. Available: https://doi.org/10.1145/1557019.1557047

[23] C. Wang, L. Deng, G. Zhou, and M. Jiang, "A global optimization algorithm for target set selection problems," *Information Sciences*, vol. 267, pp. 101–118, 2014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0020025513006750

[24] P. Shakarian, S. Eyre, and D. Paulo, "A scalable heuristic for viral marketing under the tipping model," *Social Network Analysis and Mining*, vol. 3, no. 4, pp. 1225–1248, 2013.

[25] G. Cordasco, L. Gargano, M. Mecchia, A. A. Rescigno, and U. Vaccaro, "Discovering Small Target Sets in Social Networks: A Fast and Effective Algorithm," *Algorith-*

*mica*, vol. 80, no. 6, pp. 1804–1833, 2018.

[26] E. N. Gilbert, “Random graphs,” *The Annals of Mathematical Statistics*, vol. 30, no. 4, pp. 1141–1144, 1959. [Online]. Available: http://www.jstor.org/stable/2237458

[27] J. Leskovec and A. Krevl, “SNAP Datasets: Stanford large network dataset collection,” http://snap.stanford.edu/data, Jun. 2014.

[28] R. Zafarani and H. Liu, “Social computing data repository at ASU,” 2009. [Online]. Available: http://socialcomputing.asu.edu