

Generating Dominating Sets Using Locally-Defined Centrality Measures

Hunter Rehm , Robert Kassouf-Short , Puck Rombach 

May 16, 2023

Abstract

The dominating set problem has many practical applications but is well-known to be NP-hard. Therefore, there is a need for efficient approximation algorithms, especially in applications such as ad hoc wireless networks. Most distributed algorithms proposed in the literature assume that each node has knowledge of the network structure. We propose a distributed approximation algorithm that uses two rounds of communication, and where each node has only local information, both in terms of network structure and dominating set assignment. First, each node calculates a local centrality measure to determine whether it is part of the dominating set D . The second round guarantees D is a dominating set by adding any non-dominated nodes. We compare several centrality measures and show that the Shapley value, introduced in game theory, is theoretically motivated and performs well in practice on several synthetic and real-world networks.

Keywords – Network, Dominating, Approximation, Shapley.

MSC – 05C69, 05C85 .

1 Introduction

A dominating set in a network is a subset of nodes so that every node is either in the dominating set or has a neighbor in the set. Dominating sets and their variations have many applications [21, 23, 37]. Often, the objective is to find a dominating set of minimum size.

Deciding whether or not a graph has a dominating set of size k is NP-complete [16]. The current fastest exact algorithm for identifying a minimum dominating set was introduced in 2011 by Van Rooij and Bodlaender [33] and runs in $O(1.4969^n)$ time. Certain classes of graphs admit polynomial-time approximation schemes for the minimum dominating set problem. One example of such a class is unit disk graphs. This class includes random geometric graphs, as well as many types of ad hoc wireless networks [9, 18, 27]. Instead of identifying a dominating set globally, there are also distributed algorithms where

decisions are made at the level of individual nodes with limited rounds of communications [6, 13, 17, 19, 22, 35]. In most of these distributed algorithms, it is assumed that each node has full global information regarding the structure of the network, but only local information regarding the dominating set assignment. In some applications, nodes may have none or limited global information. A pertinent example of such an application is ad-hoc and wireless sensor networks [15, 20, 31, 36].

The Space Communication and Navigation (SCaN) program office manages NASA's space communication activities. One of their goals is to understand satellite communication, including routing, radio technology, and network modeling. In the future of space science, autonomous robotic missions to other planets will be a primary means for humans to explore our solar system. In particular, satellite swarm missions akin to HelioSwarm have the opportunity to offer us great insight into the nature of other planets [29]. The power of satellite swarm missions comes from the wide variety of different roles that small satellites can serve. However, to enable interplanetary satellite swarm missions, it will be essential that satellites can autonomously organize and optimize the variety of roles available to the swarm.

One motivation for this paper comes from this consideration. Two general roles that satellites fall into are communications satellites and data collection satellites. Future swarms may have the flexibility to switch between these roles as the needs of the swarm demand. To enable the most efficient distribution of communications and data collection satellites, each satellite needs to be able to detect and adapt to the needs of the network autonomously. Communications satellites should form a dominating set over the swarm so that each data collector has the capacity to collate and transmit their data, but we also want to minimize the number of communications satellites so as to maximize the amount of data the swarm can collect.

Finding a minimum dominating set in these types of applications is often not feasible, because there may not be a central processor with access to the full network structure. Instead, the objective is to find an approximately minimum dominating set in a distributed manner where nodes can access only local information in the network.

Let $G(V, E)$ be a simple, undirected graph on n nodes, and let $\gamma(G)$ be the *domination number* of G ; the cardinality of a minimum dominating set. One of the most fundamental results in the theory of dominating sets is an upper bound in terms of the minimum degree $\delta(G)$ of G :

$$\gamma(G) \leq \frac{1 + \ln(1 + \delta(G))}{1 + \delta(G)} n. \quad (1)$$

This result was proved independently by [4, 5, 24, 28]. The proof given in [4] has become a standard example of the power of the probabilistic method. It gives a simple, two-step probabilistic algorithm that yields a dominating set whose cardinality is in expectation of the upper bound in Inequality 1. The algorithm works as follows. In the first round, each vertex is assigned to a set X with a

fixed probability p , independently of other vertices. In the second round, each vertex not dominated by X is added to a set Y . The output is an approximately minimum dominating set $X \cup Y$. This algorithm is both fast, as it is linear in the size of the graph, and highly local, as decisions are made at the vertex level with communication only to immediate neighbors.

Wu and Li [35], proposed a similar algorithm with a deterministic first step. It uses the local clustering coefficient to select a dominating set [34]. This algorithm assigns a node v to the dominating set if it has a clustering coefficient less than 1. In this case, a connected dominating set is guaranteed, and there is no need for a second round.

Inspired by this style of algorithm, we propose an algorithm for generating dominating sets with the help of local centrality measures. Our algorithm is not guaranteed to generate a minimum dominating set, but we show that it performs well in practice in finding small dominating sets. In the first step, the algorithm selects vertices based on a centrality measure meeting a certain threshold. In the second step, more vertices are added as needed to guarantee a dominating set.

We compare several different centrality measures on synthetic and real-world networks. The best-performing measure in our analysis is the Shapley value, introduced in game theory [30]. It is used in various contexts to measure the importance of an actor in game theory [7, 8, 26, 32]. There are various definitions of the Shapley value in the literature. Here we use the definition from [8], which defines the Shapley value of a node as the sum of the reciprocals of the degrees of its neighbors. (These authors then subtract 1 from each value, which we will not do.)

In Section 2, we define the two-step algorithm and a few centrality measures that appear to effectively select small dominating sets. We compare their performance in the two-step algorithm on various synthetic and real-world networks in Section 3 and 4, respectively. Section 5 provides counterexamples to a few conjectures about the two-step algorithm that we developed from the simulations. Lastly, in Section 6, we discuss a few natural future directions.

2 Dominating set algorithms using thresholds

In Section 2.1 we outline the two-step algorithm explicitly. In Section 2.2, we define the centrality measures considered in this study with a few basic observations that motivate their use in this context.

2.1 The two-step algorithm

Let $c(v)$ be a locally computed node centrality measure on the node set of a network $V(G)$, and $\tau \geq 0$ a threshold constant. The two-step algorithm has two rounds: the first round selects all nodes v such that $c(v) > \tau$, and the second round selects all nodes v such that neither v nor any of its neighbors

were selected in the first round. The second step ensures that the final set of selected nodes forms a dominating set in G . Let $N(v) = \{u \in V | (u, v) \in E(G)\}$.

Two-step Algorithm

1. Let X be the set of nodes v with $c(v) > \tau$.
2. Let Y be the set of nodes v such that $(N(v) \cup \{v\}) \cap X = \emptyset$.
3. Let $D = X \cup Y$.

This algorithm can be executed in a distributed manner. To perform the first step, each node sends the necessary information to its neighbors so that each of them can calculate $c(v)$. Then, each node communicates to its neighbors whether they are in the set and carries out the second step. All of the measures we consider in Section 2.2 can be computed using one round of communication with the immediate neighborhood of each node, supposing each node knows its degree. However, local or not, any centrality measure can be used in the algorithm. If the algorithm is implemented in a distributed manner and a local centrality measure is used, it runs in time $O(\Delta(G))$, where $\Delta(G)$ is the maximum degree of the network. In the application of ad-hoc wireless networks, this often implies sublinear or even constant run time.

This algorithm performs best when X is close to a dominating set so that Y is small. Thus, the goal of the measure $c(v)$ is to capture nodes in X likely to appear in a minimum dominating set. In Section 2.2, we define a few centrality measures of interest. We compare their performance in the two-step algorithm on various synthetic and real-world networks in Section 3 and 4, respectively.

2.2 Centrality measures

Since we want our two-step algorithm to be local, we focus on centrality measures that can be determined locally. Specifically, we consider centrality measures that depend only on the immediate neighborhood of a node.

Uniform random measure (URM): Alon investigates which threshold minimizes the expected size of the set constructed in the two-step algorithm and gives an upper bound on the domination number [3]. Refer to Section 1 for more information.

Clustering coefficient (ICC): In [35], Li and Wu show that the collection of all nodes with two non-adjacent neighbors is a connected dominating set. This is quantified by finding the proportion of non-adjacent neighbors, known as the *clustering coefficient* [34] of a node v

$$CC(v) = \frac{t_v}{\binom{k_v}{2}} = \frac{2t_v}{k_v(k_v - 1)}.$$

Here, t_v is the number of triangles that node v is a part of, and k_v is the degree of node v .

Based on simulated data, we suspect that the nodes representative of stronger dominating set candidates tend to have lower values of $CC(v)$. For this reason, we let

$$ICC(v) = 1 - CC(v) = \frac{\binom{k_v}{2} - t_v}{\binom{k_v}{2}}.$$

It is worth noting that ICC is not a new centrality measure but rather our adjustment of the clustering coefficient that allows for more convenient comparisons with the other centrality measures, as they all tend to correlate positively with the likelihood of appearing in minimum dominating sets.

Using the language of our paper, Li and Wu show that the first step of two-step algorithms constructs a connected dominating set with measure ICC and a threshold of 0.

Relative neighbor degree (RND): Ai, Li, Su, Jiang, and Xiong define the neighbor-degree centrality of v as $ND(v) = \frac{\sum_{u \in N(v)} k_u}{k_i}$ [1]. We propose the *relative neighbor-degree centrality* of v as

$$RND(v) = \frac{k_v}{ND(v)} = \frac{k_v^2}{\sum_{u \in N(v)} k_u}$$

which measures how the average degree of nodes in $N(v)$ compares to the degree of v itself. If a node v has a relative neighbor degree of 1, then the average degree of the neighbors of v equals the degree of v . If v has a relative neighbor degree larger than 1, then v makes for a good dominating set candidate as its degree is larger than its neighbor's degree on average.

Shapley value (SV): The Shapley value was originally proposed in [30], but was not introduced in network theory until later. Bozzo, Franceschet, and Rinaldi describe the Shapley value of a node v in a network as the sum of the reciprocals of the neighbors' degrees,

$$SV(v) = \sum_{u \in N(v)} \frac{1}{deg(u)} [8, 30].$$

A node with a high Shapley value is likely adjacent to many low-degree nodes.

For the remainder of the paper, we compare all four measures and specifically focus on how SV and RND, as they seem to perform best in this setting. From some simple analysis, these two measures are related. For instance, Proposition 2.1 shows that the number of nodes selected in the first round of the two-step algorithm with SV is always greater than that with RND.

Proposition 2.1. *For node v in a network N , if $RND(v) \geq \tau$, then $SV(v) \geq \tau$.*

Proof. Suppose that $RND(v) \geq \tau$. Then the average degree of a node in N_v is less than k_v/τ . By Jensen's inequality, the average of the set $\{1/k_u | u \in N_v\}$ is at least τ/k_v . Hence, $SV(v) = \sum_{u \in N_v} 1/k_u \geq k_v \frac{\tau}{k_v} = \tau$. \square

We give the expected value of SV in Proposition 2.2 below. This proposition shows that even though SV is a locally computed measure, we know how each

node will compare to the average without knowing anything about the network’s structure. The expected value of RND is not as easily described.

Proposition 2.2. *Let $G = (V, E)$ graph. The expected Shapley value of a node $v \in V$ is 1.*

Proof. Let A be the adjacency matrix of a network $N = (V, E)$ with $n = |V|$. Let M be the stochastic matrix with $M_{ij} = \frac{A_{ij}}{k_j}$ and $\mathbf{1}$ be the n -dimensional vector with 1 in each entry. Notice that $(M \cdot \mathbf{1})_i = SV_i$ for each $i \in V$. Since the stochastic matrix preserves the average of vectors, then the expected Shapley value is 1. \square

From Proposition 2.2, $\tau = 1$ is a natural choice of threshold if we use SV in our two-step algorithm. To round out this section, we give additional evidence suggesting that $\tau = 1$ is a reasonable choice of threshold in a vacuum.

Observation 2.3. *If $\deg(v) = 1$, then v is adjacent to a node u with $SV(u) \geq 1$.*

Observation 2.4. *If $\deg(v) \geq \deg(u)$ for all $u \in N(v)$, then $RND(v) \geq 1$.*

According to Observation 2.3, a threshold of $\tau = 1$ will, among others, select all nodes adjacent to degree 1 nodes in the first round of the algorithm. We will see in Section 3 and 5, other thresholds better find minimal dominating sets using the two-step algorithm.

3 Simulated data

We test our algorithm on simulated graphs generated from three random graph families. We generate 100 networks for each family with 100 nodes and an expected degree of 10. Below we describe the families and present the computational results.

Erdős-Rényi Model [14] In Figure 1, we compare the average size of a dominating set created by the two-step algorithm using a variety of thresholds over instances of Erdős-Rényi graphs. We sample each graph using 100 nodes and a probability $p = 0.1$.

Random Geometric Graphs [12] In Figure 2, we compare the size of the dominating set found by the two-step algorithm with different threshold values in random geometric graphs on the unit square with 100 nodes and a radius $r = 0.178$. The motivation for using this radius is that the average degree $100(0.178)^2$ is about 10, consistent across the other random graph families.

Chung-Lu Model [2, 10, 11] We use the Chung-Lu model to sample graphs with a longer-tailed degree distribution and create the degree sequence using a negative binomial degree distribution. Figure 3 compares the size of the dominating set for each threshold on 100 different graphs sampled using the Chung-Lu model. We create the degree sequence using the negative binomial distribution with $n = 1$ and $p = 0.1$.

In each of Figures 1, 2, and 3, SV and RND construct the smallest dominating sets across all thresholds, with SV slightly outperforming RND. Notice that the minimal dominating set in each family typically occurs at a threshold greater than 1 rather than precisely at 1. Because SV generally performed best, we focus on the Shapley value in Section 4.

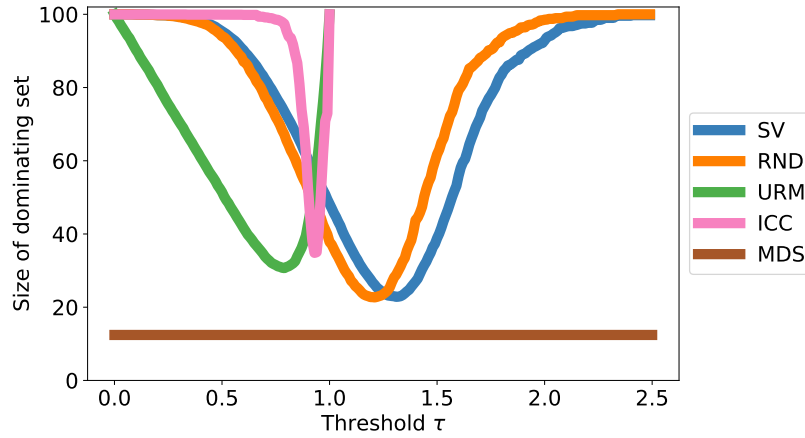


Figure 1: Erdős-Rényi Model with 100 nodes and probability 0.1.

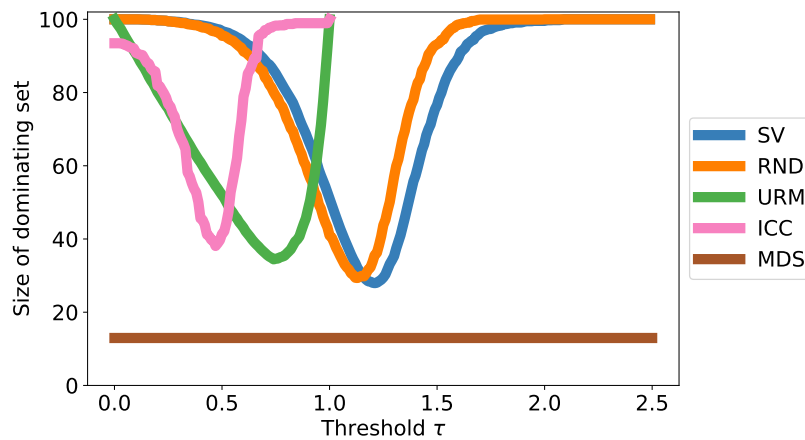


Figure 2: A comparison of the centrality measures in the local domination algorithm for random geometric graphs $G(100, 0.178)$.

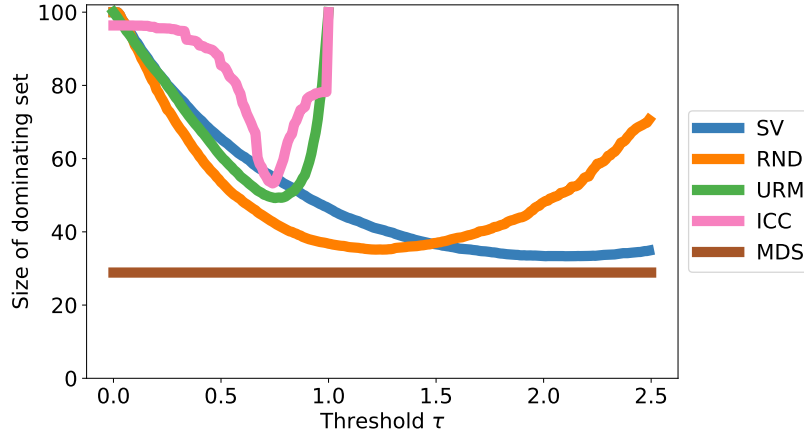


Figure 3: Chung-Lu model with 100 nodes and a degree distribution sampled from a negative binomial distribution.

4 Real-world data

We test the two-step algorithm using the Shapley value on two real-world networks: the Iridium satellite network and the European Natural Gas Pipeline network. We first show how the algorithm performs with $\tau = 1$. In the case of the Iridium network, we see a case where choosing a different threshold offers us a smaller dominating set.

4.1 Iridium Satellite Network

The Iridium satellites form a network in Low-Earth Orbit. Figure 4a indicates which nodes are selected in the two-step algorithm (in pink) with the Shapley value and threshold $\tau = 1$. Figure 4b shows the dominating set chosen by the two-step algorithm with the Shapley value when $\tau = 1.2$.

4.2 European Natural Gas Pipeline Network

The International Energy Agency (IEA) data, analysis, and policy recommendations have collected data from 31 participating countries to create the European natural gas network (ENGN). We use the Shapley value in the two-step algorithm to find a dominating set. In Figure 5, the nodes colored pink are selected by the two-step algorithm with SV centrality measure and a threshold of $\tau = 1$.

Figure 6 is a subgraph of the ENGN where the Shapley value scales each node. This visual represents how the Shapley value can vary quite a bit from node to node.

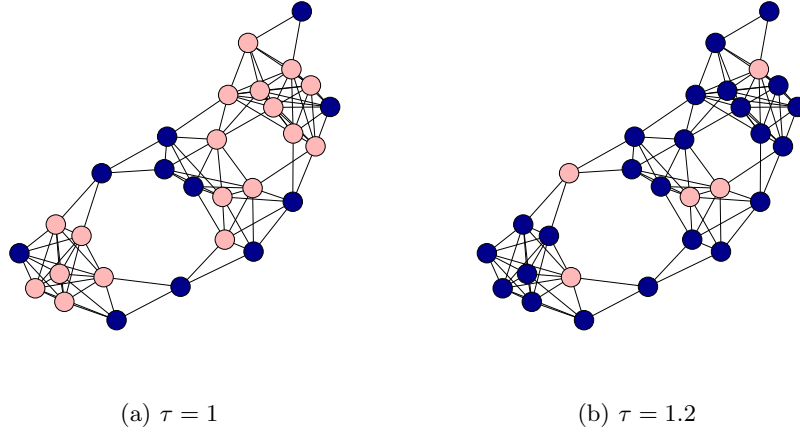


Figure 4: Comparing thresholds in the two-step algorithm with SV centrality measure on an instance of the Iridium satellite network. The color pink indicates a node in the dominating set.

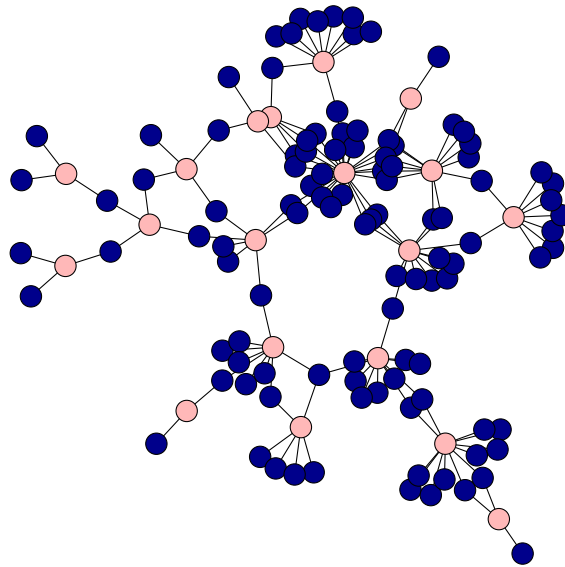


Figure 5: European Natural Gas Pipeline Network with dominating set.

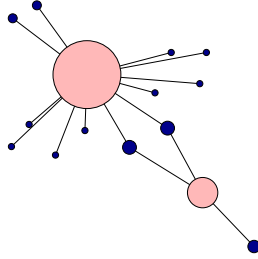


Figure 6: Subgraph of the European Natural Gas Pipeline Network. Each node size is scaled to its Shapley value.

5 Oddities

This section presents a set of counterexamples to natural conjectures regarding the two-step algorithm with the SV centrality measure. These counterexamples are enlightening, revealing exciting relationships between the measures we examined here and some general oddities surrounding the two-step algorithm.

With a threshold in the first filtering for the two-step algorithm, much of our questioning centered on choosing the best threshold. Inspired by Wu and Li [35], we initially sought a universal threshold for constructing a dominating set. If there were such a threshold for SV or RND, we could reduce our two-step algorithm to just one step: compute the measure and determine whether or not you are in the dominating set. If the threshold is 0, only one step is necessary, but perhaps a small non-zero threshold will also guarantee a dominating set in one step.

It turns out that there is no such non-zero threshold, but our construction to show this is unwieldy and unlikely to appear in graphs with relatively small numbers of nodes. We present this construction in the proof of the following proposition.

Proposition 5.1. *Given a threshold $\tau > 0$, let S_τ be the set of nodes such that $SV(v) > \tau$. There exists a graph G such that S_τ is not a dominating set of G .*

Proof. We construct a graph with node a that satisfies the following properties:

1. For node a , $SV(a) < \tau$.
2. For all nodes $b \in N(a)$, $SV(b) < \tau$.

Given $\tau > 0$, there exists some $N_1 \in \mathbb{N}$ such that $\frac{1}{N_1} < \tau$. So, we can construct a set of N_1 nodes $\{b_i\}_{i=1}^{N_1}$ such that each b_i is adjacent to a . Thus, $\deg(a) = N_1$, and $N(a) = \{b_i\}_{i=1}^{N_1}$.

Since $N_1, \tau > 0$, there exists $N_2 \in \mathbb{N}$ such that $N_2 > \frac{N_1}{\tau}$. For each b_i , construct $N_2 - 1$ nodes $\{c_{i,j}\}_{j=1}^{N_2-1}$ such that each is adjacent to b_i . Thus,

$\deg(b_i) = N_2$, and $N(b_i) = \{a\} \cup \{c_{i,j}\}_{j=1}^{N_2-1}$ for each i . Then, by construction, $\text{SV}(a) = \frac{N_1}{N_2} < \tau$.

Since $\frac{1}{N_1} < \tau$, $\tau - \frac{1}{N_1} > 0$, and since $N_2 \in \mathbb{N}$, $N_2 - 1 \geq 0$. Thus, there exists $N_3 \in \mathbb{N}$ such that $N_3 > \frac{N_2-1}{\tau - \frac{1}{N_1}}$. So, for each $c_{i,j}$, construct $N_3 - 1$ nodes $\{d_{i,j,k}\}_{k=1}^{N_3-1}$ such that each is adjacent to $c_{i,j}$. Thus, $\deg(c_{i,j}) = N_3$ for each j . So, by construction, for each b_i ,

$$\text{SV}(b_i) = \frac{1}{\deg(a)} + \sum_{j=1}^{N_2-1} \frac{1}{\deg(c_{i,j})} = \frac{1}{N_1} + \frac{N_2-1}{N_3} < \tau.$$

Thus, any graph containing this construction as a subgraph would satisfy the conditions above. \square

From the construction in Proposition 5.1, and the relationship between RND and SV given in Proposition 2.1, we get the following result.

Corollary 5.2. *Given a threshold $\tau > 0$, let R_τ be the set of nodes such that $\text{RND}(v) > \tau$. There exists a graph G such that R_τ is not a dominating set of G .*

The choice of threshold then depends on the graph structure we are examining. However, in the plots we generated, the graphs appeared to have a consistent shape, including a single local minimum. For the minimum to be unique, one condition would be that for any three thresholds $\tau < \sigma < \rho$, one could never see $f(\tau) < f(\sigma) > f(\rho)$.

Unfortunately, this is not true in general. Figure 7 depicts one counterexample of this idea. In particular, when we use SV with the two-step algorithm at three different thresholds ($5/6$, 1 , and $11/6$), the size of the dominating set generated goes from 19 to 20 and back to 19.

We conclude our catalog of oddities by focusing on the threshold of 1. Thanks to Proposition 2.2, we know that 1 holds a special place concerning the SV and RND values.

However, returning to the data represented in Figures 2,3, and 1, there was one further counterexample to consider. When we set the threshold at 1, the dominating set generated with RND was generally larger than the dominating set developed with SV. However, Figure 8 shows an example where the number of nodes selected in the two-step algorithm is smaller in SV (Figure 8a) than in RND (Figure 8b) using the threshold $\tau = 1$. Here, the black nodes are chosen in the first round and the pink nodes in the second.

There is undoubtedly more exploration in characterizing the function described in the plots. Both experimentally and theoretically, further investigation is warranted.

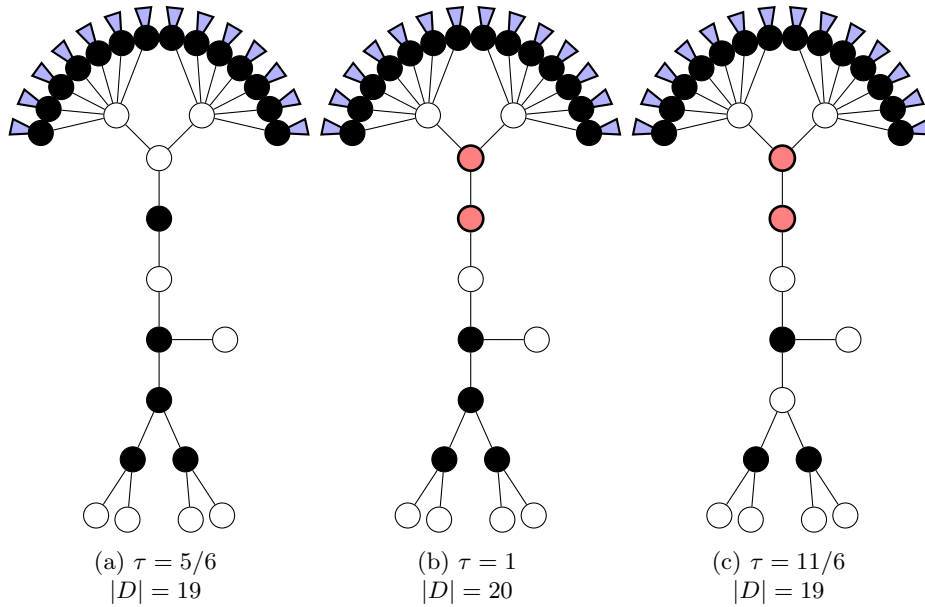


Figure 7: The size of the dominating set formed by the two-step algorithm with the Shapley value does not achieve a unique local minimum. The purple triangles represent 100 leaf nodes. A black node suggests that it was chosen in the first round of the two-step algorithm, and red suggests it was chosen in the second round.

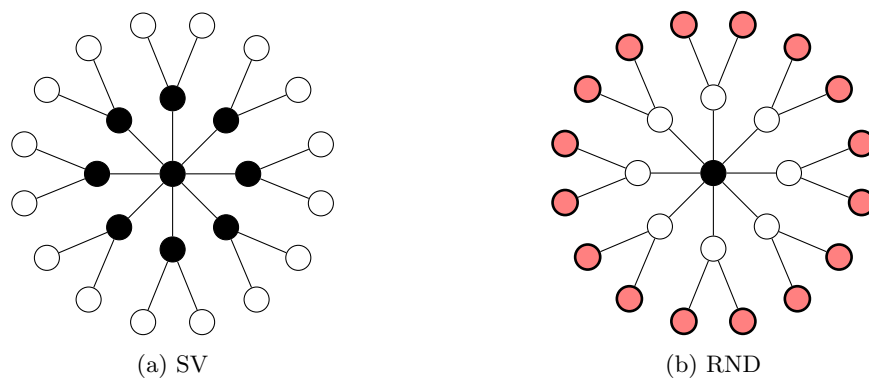


Figure 8: An example where the dominating set generated using RND is greater than the dominating set generated using SV.

6 Future directions

Various algorithms aim to approximate a minimum dominating set for graphs [25]. Our approach of using a two-step algorithm allows for a larger variety of node measures than we propose for generating dominating sets. However, we focused on four local measures to highlight some methods that could be practical in robot swarm and autonomous system communications.

We see several directions to take this work into the future. First, an analysis of different measures and their properties when plugged into the two-step algorithm may be an excellent method for comparing the effects of other measures. Included with this, a more in-depth theoretical analysis of the two-step algorithm is warranted to explore its effectiveness further.

One concept we have toyed with is constructing bounds for the thresholds in various situations. For example, one might wonder whether there are certain classes of networks that yield natural bounds. For instance, with regular graphs, the Shapley value at each node is necessarily 1. In graphs that are regular or close to regular, it may be effective to combine for example the Shapley value with a random measure. Predicting optimal thresholds based on network structure alone or time-varying network structure alone is a powerful tool in robotic mission design. Moreover, if the ability to find bounds analytically proves intractable, passing this information into a neural network may be an exciting problem to pursue. A Graph Neural Network that intakes graph structure and outputs a substantial threshold for achieving a dominating set could enable more optimizations, especially if this could be computed locally in some way.

Acknowledgements

The authors would like to acknowledge the Space Communications and Navigation program at NASA for enabling helpful discussions and feedback throughout this work. Hunter Rehm is supported by the Vermont Space Grant Consortium (VTSGC) through the Graduate Fellowship Program.

References

- [1] Jun Ai, Linzhi Li, Zhan Su, Linhua Jiang, and Naixue Xiong. Node-importance identification in complex networks via neighbors average degree. In *2016 Chinese Control and Decision Conference (CCDC)*, pages 1298–1303. IEEE, 2016.
- [2] William Aiello, Fan Chung, and Linyuan Lu. A random graph model for power law graphs. *Experimental Mathematics*, 10(1):53–66, 2001.
- [3] Noga Alon. Transversal numbers of uniform hypergraphs. *Graphs and Combinatorics*, 6(1):1–4, 1990.

- [4] Noga Alon and Joel H Spencer. *The Probabilistic Method*. John Wiley & Sons, 2016.
- [5] Vladimir I Arnautov. Estimation of the exterior stability number of a graph by means of the minimal degree of the vertices. *Prikladnaya Matematika i Programirovanie*, 11(3-8):126, 1974.
- [6] Leonid Barenboim, Michael Elkin, and Cyril Gavoille. A fast network-decomposition algorithm and its applications to constant-time distributed computation. *Theoretical Computer Science*, 751:2–23, 2018.
- [7] Enrico Bozzo and Massimo Franceschet. A theory on power in networks. *Communications of the ACM*, 59(11):75–83, 2016.
- [8] Enrico Bozzo, Massimo Franceschet, and Franca Rinaldi. Vulnerability and power on networks. *Network Science*, 3(2):196–226, 2015.
- [9] Xiuzhen Cheng, Xiao Huang, Deying Li, Weili Wu, and Ding-Zhu Du. A polynomial-time approximation scheme for the minimum-connected dominating set in ad hoc wireless networks. *Networks: An International Journal*, 42(4):202–208, 2003.
- [10] Fan Chung and Linyuan Lu. The average distances in random graphs with given expected degrees. *Proceedings of the National Academy of Sciences*, 99(25):15879–15882, 2002.
- [11] Fan Chung and Linyuan Lu. Connected components in random graphs with given expected degree sequences. *Annals of Combinatorics*, 6(2):125–145, 2002.
- [12] Brent N Clark, Charles J Colbourn, and David S Johnson. Unit disk graphs. *Discrete Mathematics*, 86(1-3):165–177, 1990.
- [13] Fei Dai and Jie Wu. An extended localized algorithm for connected dominating set formation in ad hoc wireless networks. *IEEE Transactions on Parallel and Distributed Systems*, 15(10):908–920, 2004.
- [14] Paul Erdős. On random graphs. *Mathematicae*, 6:290–297, 1959.
- [15] Carlos F García-Hernández, Pablo H Ibarguengoytia-Gonzalez, Joaquín García-Hernández, and Jesús A Pérez-Díaz. Wireless sensor networks and applications: a survey. *IJCSNS International Journal of Computer Science and Network Security*, 7(3):264–273, 2007.
- [16] Michael R Garey. A guide to the theory of NP-completeness. *Computers and Intractability*, 1979.
- [17] Mohsen Ghaffari and Fabian Kuhn. Derandomizing distributed algorithms with small messages: Spanners and dominating set. In *32nd International Symposium on Distributed Computing (DISC 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018.

- [18] Harry B Hunt III, Madhav V Marathe, Venkatesh Radhakrishnan, Shankar S Ravi, Daniel J Rosenkrantz, and Richard E Stearns. \mathcal{N} -approximation schemes for NP-and PSPACE-hard problems for geometric graphs. *Journal of Algorithms*, 26(2):238–274, 1998.
- [19] Lujun Jia, Rajmohan Rajaraman, and Torsten Suel. An efficient distributed algorithm for constructing small dominating sets. *Distributed Computing*, 15(4):193–205, 2002.
- [20] Shafiqullah Khan, Al-Sakib Khan Pathan, and Nabil Ali Alrajeh. Wireless sensor networks: Current status and future trends. 2016.
- [21] Mark Sh Levin. On combinatorial optimization for dominating sets (literature survey, new models). *arXiv preprint arXiv:2009.09288*, 2020.
- [22] Ben Liang and Zygmunt J Haas. Virtual backbone generation and maintenance in ad hoc network mobility management. In *Proceedings IEEE INFOCOM 2000. Conference on Computer Communications. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (Cat. No. 00CH37064)*, volume 3, pages 1293–1302. IEEE, 2000.
- [23] Zhuo Liu, Bingwen Wang, and Lejiang Guo. A survey on connected dominating set construction algorithm for wireless sensor network. *Information Technology Journal*, 9(6):1081–1092, 2010.
- [24] László Lovász. On the ratio of optimal integral and fractional covers. *Discrete Mathematics*, 13(4):383–390, 1975.
- [25] Gang Lu, Ming-Tian Zhou, Yong Tang, Zhen-Qiang Wu, Guo-Yong Qiu, and Liu Yuan. A survey on exact algorithms for dominating set related problems in arbitrary graphs. *Chinese Journal of Computers*, 33(6):1073–1087, 2010.
- [26] Tomasz P Michalak, Karthik V Aadithya, Piotr L Szczepanski, Balaraman Ravindran, and Nicholas R Jennings. Efficient computation of the Shapley value for game-theoretic network centrality. *Journal of Artificial Intelligence Research*, 46:607–650, 2013.
- [27] Tim Nieberg, Johann Hurink, and Walter Kern. Approximation schemes for wireless networks. *ACM Transactions on Algorithms (TALG)*, 4(4):1–17, 2008.
- [28] Charles Payan. Sur le nombre d’absorption d’un graphe simple. 1975.
- [29] Raj Thilak Rajan, Shoshana Ben-Maor, Shaziana Kaderali, Calum Turner, Mohammed Milhim, Catrina Melograna, Dawn Haken, Gary Paul, Vedant, V. Sreekumar, Johannes Wepler, Yosephine Gumulya, Riccardo Bunt, Asia Bulgarini, Maurice Marnat, Kadri Bussov, Frederick Pringle, Jusha Ma, Rushanka Amrutkar, Miguel Coto, Jiang He, Zijian Shi, Shahd Hayder,

- Dina Saad Fayez Jaber, Junchao Zuo, Mohammad Alsukour, Cécile Renaud, Matthew Christie, Neta Engad, Yu Lian, Jie Wen, Ruth McAvinia, Andrew Simon-Butler, Anh Nguyen, and Jacob Cohen. Applications and potentials of intelligent swarms for magnetospheric studies. *Acta Astronautica*, 193:554–571, 2022.
- [30] Lloyd S Shapley. Cores of convex games. *International Journal of Game Theory*, 1(1):11–26, 1971.
- [31] Manish Kumar Singh, Syed Intekhab Amin, Syed Akhtar Imam, Vibhav Kumar Sachan, and Amit Choudhary. A survey of wireless sensor network and its types. In *2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN)*, pages 326–330. IEEE, 2018.
- [32] N Rama Suri and Yadati Narahari. Determining the top-k nodes in social networks using the Shapley value. In *Proceedings of the 7th international Joint Conference on Autonomous Agents and Multiagent Systems-Volume 3*, pages 1509–1512, 2008.
- [33] Johan MM Van Rooij and Hans L Bodlaender. Exact algorithms for dominating set. *Discrete Applied Mathematics*, 159(17):2147–2164, 2011.
- [34] Duncan J Watts and Steven H Strogatz. Collective dynamics of ‘small-world’ networks. *nature*, 393(6684):440–442, 1998.
- [35] Jie Wu and Hailan Li. A dominating-set-based routing scheme in ad hoc wireless networks. *Telecommunication Systems*, 18(1):13–36, 2001.
- [36] Jennifer Yick, Biswanath Mukherjee, and Dipak Ghosal. Wireless sensor network survey. *Computer Networks*, 52(12):2292–2330, 2008.
- [37] Jiguo Yu, Nannan Wang, Guanghui Wang, and Dongxiao Yu. Connected dominating sets in wireless ad hoc and sensor networks—a comprehensive survey. *Computer Communications*, 36(2):121–134, 2013.