CENTRAL ASIAN JOURNAL OF MATHEMATICAL THEORY AND COMPUTER SCIENCES

https://cajmtcs.casjournal.org/index.php/CAJMTCS

Volume: 07 Issue: 03 | July 2026 ISSN: 2660-5309

.

Article

Computational Simulation of Equitable Topological Graphs

Hanady F. Khudhair[1], Mohammed A. Abdlhusein[2]

1. Department of Mathematics, College of Education for Pure Sciences, University of Thi-Qar, Thi-Qar, Iraq
2. Department of Mathematics, College of Education for Women, Shatrah University, Thi-Qar, Iraq

Correspondence: hanady_fadel.kh@utq.edu.iq[1], mmhd@shu.edu.iq[2]

Abstract: This study presents a computational simulation of the equitable topological graph G_q, which is constructed from a finite non-empty set endowed with the discrete topology. The vertices of the graph correspond to all non-empty proper subsets of the underlying set, while adjacency is defined according to a complementary-cardinality rule. The main objective of this work is to provide computational verification of the theoretical properties previously established for this class of graphs. An exact python-based algorithm was developed to generate the graph and compute its principal parameters, including the order, size, degree distribution, number of connected components, clique number, radius, and diameter. In addition, growth behavior was analyzed for increasing values of n, and monte carlo sampling techniques were employed to estimate graph size in cases where exact construction becomes computationally expensive. The computational results show complete agreement with the theoretical formulas and structural decomposition of the equitable topological graph. The study confirms that vertex degrees depend solely on subset cardinality and that the graph is connected only for small values of n, becoming disconnected for larger values. Furthermore, the simulations verify the decomposition of G_q into complete bipartite components and, in the even case, an additional complete graph corresponding to the middle cardinality class. Monte carlo experiments produced highly accurate estimates with very small relative errors, demonstrating the effectiveness of sampling methods for large-scale instances. The findings highlight the usefulness of computational techniques in validating theoretical results and investigating the structural properties of topological graphs when direct analytical or manual construction becomes impractical.

Keywords: Equitable Topological Graph, Topological Graphs, Computational Simulation, Graph Theory, Python, Monte Carlo Methods

Citation: Khudhair, H. F., Abdlhusein, M. A. Computational Simulation of Equitable Topological Graphs. Central Asian Journal of Mathematical Theory and Computer Sciences 2026, 7(3), 175-182.

Received: 23rd Apr 2025
Revised: 28th May 2026
Accepted: 20th Jun 2026
Published: 13th Jul 2026

1. Introduction

This work presents a computational simulation of the equitable topological graph G_q. The purpose of the simulation is to support the theoretical results obtained previously through an exact Python implementation. The graph is generated from a finite non-empty set X equipped with the discrete topology, where the vertices are all non-empty proper subsets of X [1], [2], [3]. Two distinct vertices A and B are adjacent whenever the cardinality of B is equal to the cardinality of the complement of A. The computational simulation is not intended to replace mathematical proofs; rather, it provides numerical and visual verification of the structural properties of G_q. For small values of n, the graph can be constructed and visualized directly. However, as n increases, the number of vertices grows exponentially according to |V(G_q)|=2^n-2 making manual construction impractical. Therefore, computational methods become useful for verifying graph parameters such as order, size, degree sequence, connectivity, clique number, radius, diameter, and growth behavior [4]. In addition, computational analysis provides further confirmation of several graph-theoretical and domination-related properties studied in discrete topological graphs [5], [6], [7], [8] and in the broader graph theory literature [9], [10].

2. Materials and Methods

This section presents the methodology used to construct the graph G_q, compute and verify its theoretical parameters, estimate large cases, and generate visualizations for selected values of n. The procedure combines exact enumeration, analytical benchmarking, and probabilistic estimation under a reproducible computational design.

Research Design

The main purpose of the simulation is to verify the theoretical results concerning G_q by means of exact computation. The Python program constructs the vertex set, determines the edge set according to the complementary-cardinality rule, and then computes graph parameters such as the order, size, number of connected components, minimum degree, maximum degree, clique number, radius, and diameter. The simulation also provides visual representations for small values of n and growth plots for larger values. These outputs help clarify the behavior of the graph and show how quickly the number of vertices and edges increases as n becomes large. The implementation used Python as the programming language [11], NetworkX for graph representation and parameter computation [12], and Matplotlib for the generation of publication-quality plots [13].

Graph Construction and Exact Enumeration of G_q

The simulation was implemented in Python by following the mathematical definition of G_q exactly. For a given integer n, the program first constructs the finite set $X = \{1, 2, \ldots, n\}$. Then, it generates the power set $P(X)$ and removes the empty set and the full set X. The remaining subsets form the vertex set of $G_q : V(G_q) = P(X) - \{\emptyset, X\}$. After constructing the vertex set, the program checks every unordered pair of distinct vertices A and B. An edge is inserted between A and B if and only if the following adjacency condition is satisfied: $AB \in E(Gq) \Leftrightarrow |B| = |A^c| = n - |A|$. This means that a vertex of cardinality k is adjacent exactly to vertices of cardinality $n - k$. When n is even and $k = n/2$, the corresponding vertices form a complete graph, since every such vertex has complementary cardinality $n/2$. Otherwise, the vertices of sizes k and $n - k$ form a complete bipartite component. The exhaustive enumeration and pairwise adjacency procedure follow standard finite-set and graph-algorithm design principles [14].

Computational Environment and Simulation Parameters

Table 1. Simulation parameters

Parameter	Description
Programming language	Python
Graph construction method	Exact enumeration for small and moderate n
Vertex rule	All non-empty proper subsets of X
Adjacency rule	AB is an edge if and only if $\lvert B\rvert = \lvert A^c\rvert$
Exact verification range	$2 \leq n \leq 10$
Visualization range	$n = 2, 3, 4, 5$
Monte Carlo range	$n = 12, 14, 16, 18, 20$
Monte Carlo samples	200000 random vertex pairs for each n
Random seed	2026, for reproducibility

Analytical Benchmarks and Verification Criteria

The numerical outputs of the program were compared with theoretical formulas derived from the structural decomposition of G_q. For odd n, the graph is a disjoint union of complete bipartite graphs. For even n, it is a disjoint union of complete bipartite graphs together with one complete graph corresponding to the middle cardinality class.

For odd n, G_q is the disjoint union of the complete bipartite graphs K[C(n,i), C(n,n−i)] for i = 1, 2, ..., (n−1)/2.

For even n, the same disjoint union runs from i = 1 to n/2 − 1 and is supplemented by the complete graph K[C(n,n/2)].

Accordingly, the number of edges used for verification is computed as follows:

For odd n, ∣E(G_q)∣ equals the sum of [C(n,i)]² from i = 1 to (n−1)/2.

For even n, ∣E(G_q)∣ equals the sum of [C(n,i)]² from i = 1 to n/2 − 1, plus C(C(n,n/2), 2).

The degree of a vertex represented by a subset A with ∣A∣ = k depends only on the complementary cardinality n − k. Specifically, deg(A) = C(n,n − k) when k ≠ n/2, whereas deg(A) = C(n,n/2) − 1 when k = n/2; the subtraction excludes a loop in the middle complete component.

Monte Carlo Estimation Procedure

For values of n beyond the exact-construction range, the study estimated the edge count by sampling 200,000 unordered pairs of distinct non-empty proper subsets. The estimated adjacency probability p_hat was converted to an edge estimate using E_hat = p_hat N(N - 1)/2, where N = 2^n - 2. A fixed random seed of 2026 was used to make the estimates reproducible. This sampling strategy follows established Monte Carlo simulation principles [15].

Validation and Reproducibility

Validation was performed at three levels. First, exact vertex and edge counts were compared with the closed-form formulas. Second, the computed degree distribution, connected components, clique number, radius, and diameter were checked against the predicted component decomposition. Third, Monte Carlo estimates were compared with exact edge counts through relative error. The exact verification range was n = 2–10, visualizations were produced for n = 2–5, and Monte Carlo estimation was applied to n = 12, 14, 16, 18, and 20.

3. Results

Verification of Order, Size, and Main Parameters

Table 2 compares the exact theoretical values with the values obtained by the Python simulation. The results confirm that the computational construction produces the same order, size, and number of connected components predicted by the theoretical structure of G_q.

Table 2. Verification of the main parameters of G_q.

n	Vertices	Sim Vertices	Edges	Sim Edges	Components	Sim Components	Min degree	Max degree	Clique number
2	2	2	1	1	1	1	1	1	2
3	6	6	9	9	1	1	3	3	2
4	14	14	31	31	2	2	4	5	6
5	30	30	125	125	2	2	5	10	2
6	62	62	451	451	3	3	6	19	20
7	126	126	1715	1715	3	3	7	35	2
8	254	254	6399	6399	4	4	8	69	70
9	510	510	24309	24309	4	4	9	126	2
10	1022	1022	92251	92251	5	5	10	251	252

Degree Distribution Verification

The simulation also confirms that the degree of a vertex depends only on the cardinality of the subset represented by that vertex, not on the specific elements contained in the subset. Table 3 shows selected degree classes for several values of n.

Table 3. Degree distribution by subset cardinality.

n	Subset size k	Number of vertices	Degree
4	1	4	4
4	2	6	5
4	3	4	4

5	1	5	5
5	2	10	10
5	3	10	10
5	4	5	5
6	1	6	6
6	2	15	15
6	3	20	19
6	4	15	15
6	5	6	6
8	1	8	8
8	2	28	28
8	3	56	56
8	4	70	69
8	5	56	56
8	6	28	28
8	7	8	8
10	1	10	10
10	2	45	45
10	3	120	120
10	4	210	210
10	5	252	251
10	6	210	210
10	7	120	120
10	8	45	45
10	9	10	10

Structural Property Verification

Table 4. Structural properties of G_q.

n	Connected?	Radius	Diameter	Component structure
2	Yes	1	1	*K_{2}*
3	Yes	2	2	*K_{3,3}*
4	No	∞	∞	*K_{4,4}* ∪ *K_{6}*
5	No	∞	∞	*K_{5,5}* ∪ *K_{10,10}*
6	No	∞	∞	*K_{6,6}* ∪ *K_{15,15}* ∪ *K_{20}*
7	No	∞	∞	*K_{7,7}* ∪ *K_{21,21}* ∪ *K_{35,35}*
8	No	∞	∞	*K_{8,8}* ∪ *K_{28,28}* ∪ *K_{56,56}* ∪ *K_{70}*
9	No	∞	∞	*K_{9,9}* ∪ *K_{36,36}* ∪ *K_{84,84}* ∪ *K_{126,126}*
10	No	∞	∞	*K_{10,10}* ∪ *K_{45,45}* ∪ *K_{120,120}* ∪ *K_{210,210}* ∪ *K_{252}*

The table verifies that G_q is connected only for $n = 2$ and $n = 3$. For $n \geq 4$, the graph becomes disconnected because the adjacency relation links only complementary cardinality classes. Therefore, the radius and diameter are infinite for $n \geq 4$ under the usual convention for disconnected graphs.

Growth Analysis for Larger Values of n

The growth analysis shows that the order and size of G_q increase rapidly as n increases. Although the component structure is theoretically clear, the direct construction of all vertices and all edges becomes increasingly expensive. This justifies the use of computational verification and sampling methods for larger values of n.

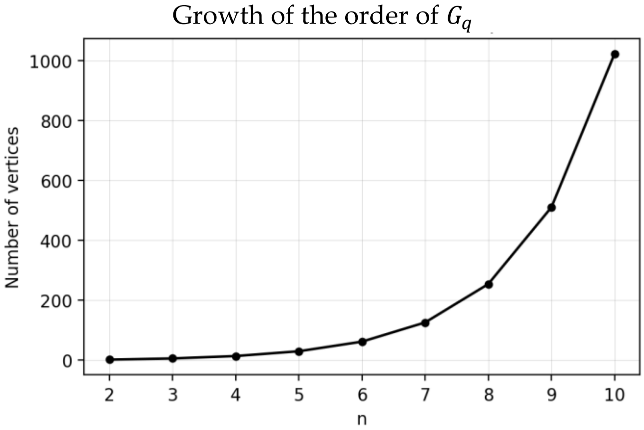

Figure 1. Growth of the number of vertices of G_q.

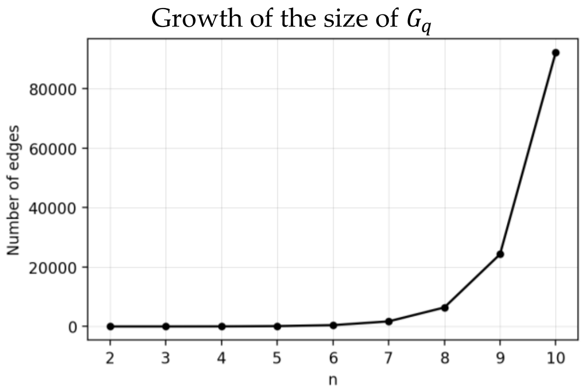

Figure 2. Growth of the number of edges of G_q.

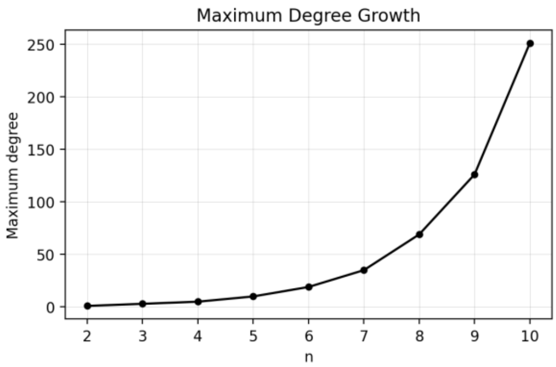

Figure 3. Growth of the maximum degree of G_q.

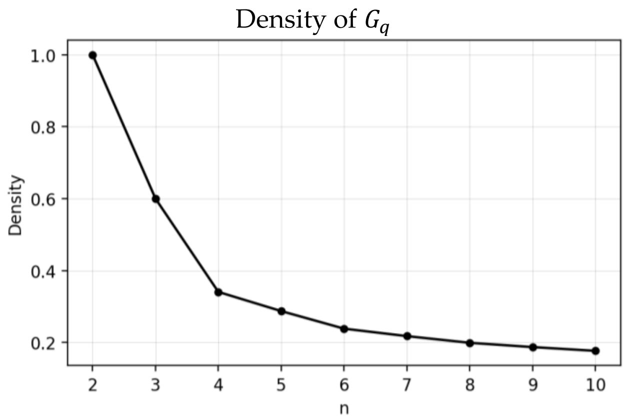

Figure 4. Density of G_q as n increases.

Monte Carlo Sampling for Large Cases

For larger values of n, monte carlo sampling was used to estimate the probability that two randomly selected distinct vertices are adjacent. In each trial, two non-empty proper subsets A and B of X were selected randomly. The pair was counted as adjacent whenever $|B| = n - |A|$. If p denotes the estimated adjacency probability and $N = |V(G_q n)|$, then the estimated number of edges is $|E(G_q)| \approx p\binom{N}{2}$. Monte carlo sampling is not a proof of the theoretical formulas. It is a numerical tool that provides an efficient approximation when exact construction is computationally costly.

Table 5. Monte Carlo estimation of the size of G_q.

n	Samples	Estimated p	Estimated Edges	Exact Edges	Relative Error (%)
12	200000	0.162115	1358260	1351615	0.4916
14	200000	0.148820	19968187	20056583	0.4407
16	200000	0.140425	301537384	300533759	0.3339
18	200000	0.132580	4555327226	4537543339	0.3919
20	200000	0.125745	69128715185	68923172031	0.2982

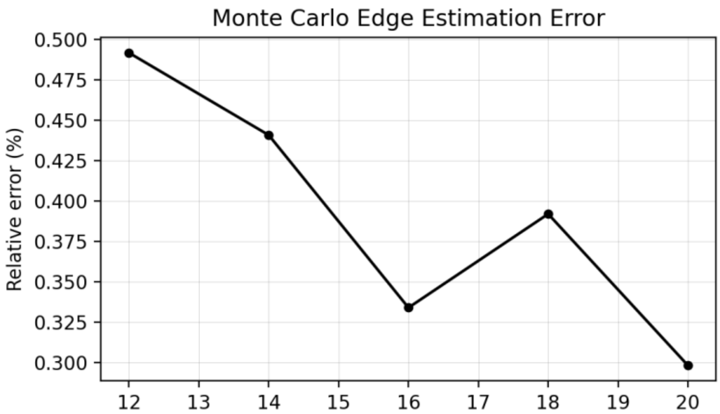

Figure 5. Relative error of Monte Carlo edge estimation.

Graph Visualizations

Figures 6-9 illustrate the structure of G_q for small values of n. The vertices are arranged by subset cardinality, so that the complementary-cardinality adjacency rule can be seen visually. For $n = 2$, the graph is P_2. For $n = 3$, it is $K_{3,3}$. For $n = 4$ and $n = 5$, the graph separates into disjoint components as predicted by the theoretical results.

Equitable Topological Graph G_q for $n = 2$

Figure 6. Visualization of G_q for $n = 2$.

Equitable Topological Graph G_q for $n = 3$

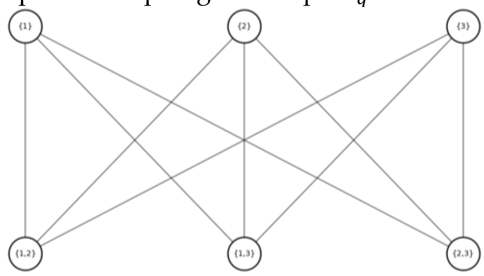

Figure 7. Visualization of G_q for $n = 3$.

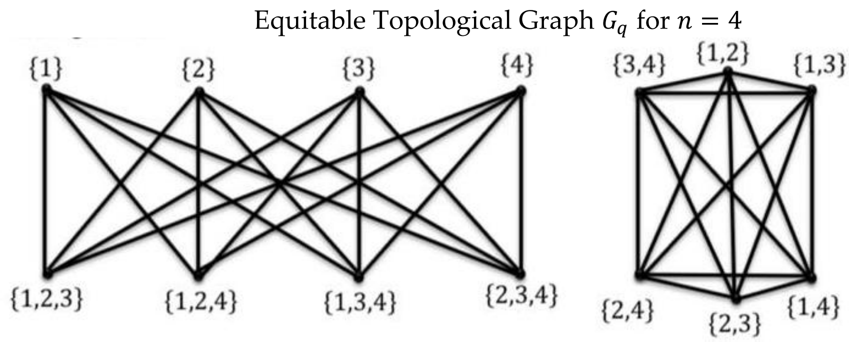

Figure 8. Visualization of G_q for $n = 4$.

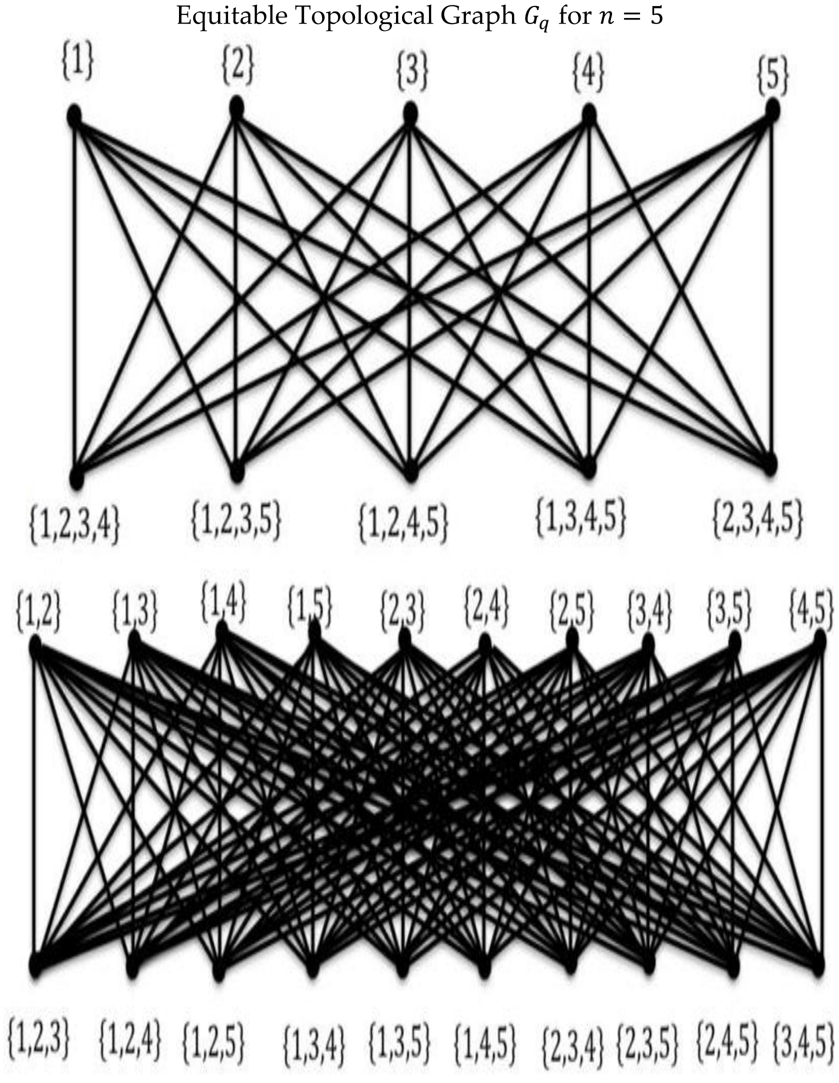

Figure 9. Visualization of G_q for n=5.

4. Discussion

The computational results support the theoretical analysis of equitable topological graphs. The exact Python simulation agrees with the theoretical order, size, component structure, degree parameters, clique number, radius, and diameter. The simulations also highlight the dependence of degree on subset cardinality: all vertices corresponding to subsets of the same cardinality have the same degree. Moreover, the visualizations confirm the decomposition of G_q into complete bipartite components and, for even n, an additional complete graph on the middle cardinality class. The Monte Carlo experiment provides an additional numerical perspective by estimating the probability of adjacency between two randomly selected vertices. This is particularly useful for large values of n, where constructing all possible edges may become expensive. The small relative errors obtained from 200000 random trials indicate that sampling can effectively approximate the global size behavior of G_q, while the exact mathematical formulas remain the primary reference.

5. Conclusion

This computational study verifies the principal theoretical properties of the equitable topological graph G_q using Python. The simulation confirms that the vertex set has order 2^n − 2 and that the edge set follows directly from the complementary-cardinality adjacency condition. The numerical results agree with the theoretical decomposition into complete bipartite and complete components. The growth tables, visualizations, and Monte Carlo estimates further demonstrate the usefulness of computational methods for studying G_q as n increases. Thus, the simulation provides consistent numerical and visual support for the mathematical results established in the theoretical analysis.

REFERENCES

[1] M. K. Idan and M. A. Abdlhusein, “Constructed discrete topological space from certain topological graphs,” Int. J. Acad. Appl. Res., vol. 7, no. 7, pp. 22–27, 2023.

[2] Z. N. Jwair and M. A. Abdlhusein, “Constructing new topological graph with several properties,” Iraqi J. Sci., vol. 64, no. 6, pp. 2991–2999, Jun. 2023, doi: 10.24996/ijs.2023.64.6.27.

[3] Z. M. Khalil and M. A. Abdlhusein, “New form of discrete topological graphs,” AIP Conf. Proc., vol. 3282, no. 1, Art. no. 040026, 2025, doi: 10.1063/5.0266555.

[4] R. Balakrishnan and K. Ranganathan, A Textbook of Graph Theory. New York, NY, USA: Springer, 2012.

[5] Z. M. Kalil, M. A. Abdlhusein, and M. R. Farahani, “Some dominating applications on discrete topological graphs,” J. Educ. Pure Sci., vol. 15, no. 1, pp. 40–46, 2025, doi: 10.32792/jeps.v15i1.678.

[6] Z. N. Jwair and M. A. Abdlhusein, “Some dominating results of the topological graph,” Int. J. Nonlinear Anal. Appl., vol. 14, no. 2, pp. 133–140, 2023, doi: 10.22075/ijnaa.2022.6404.

[7] M. K. Idan and M. A. Abdlhusein, “Different types of dominating sets of the discrete topological graph,” Int. J. Nonlinear Anal. Appl., vol. 14, no. 1, pp. 101–108, 2023, doi: 10.22075/ijnaa.2022.6452.

[8] M. K. Idan and M. A. Abdlhusein, “Some dominating results of the join and corona operations between discrete topological graphs,” Int. J. Nonlinear Anal. Appl., vol. 14, no. 5, pp. 235–242, 2023, doi: 10.22075/ijnaa.2022.6795.

[9] V. R. Kulli and S. C. Sigarkanti, “Inverse domination in graphs,” Natl. Acad. Sci. Lett., vol. 14, pp. 473–475, 1991.

[10] E. J. Cockayne, R. M. Dawes, and S. T. Hedetniemi, “Total domination in graphs,” Networks, vol. 10, pp. 211–219, 1980.

[11] G. van Rossum and F. L. Drake, Python 3 Reference Manual. Scotts Valley, CA, USA: CreateSpace, 2009.

[12] A. A. Hagberg, D. A. Schult, and P. J. Swart, “Exploring network structure, dynamics, and function using NetworkX,” in Proc. 7th Python in Science Conf. (SciPy 2008), Pasadena, CA, USA, 2008, pp. 11–15.

[13] J. D. Hunter, “Matplotlib: A 2D graphics environment,” Comput. Sci. Eng., vol. 9, no. 3, pp. 90–95, May–Jun. 2007, doi: 10.1109/MCSE.2007.55.

[14] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, Introduction to Algorithms, 4th ed. Cambridge, MA, USA: MIT Press, 2022.

[15] R. Y. Rubinstein and D. P. Kroese, Simulation and the Monte Carlo Method, 3rd ed. Hoboken, NJ, USA: Wiley, 2016, doi: 10.1002/9781118631980.