# Draw Thresholds in Ramsey Two-Player Games

Kelly Liu

**Abstract**

This paper investigates the growth and behavior of two-player achievement games based on Ramsey Theory on undirected graphs. We use a Monte Carlo, Python simulation to empirically study the growth patterns of Ramsey-type structures and draw thresholds that can serve as benchmarks for graph coloring algorithms. We define a win as the first player to create a monochromatic $k$-clique on a graph of $n$ nodes. Moreover, we find that, with more nodes, the graph better approximates a logistic growth pattern. Our results across simulations show a positive relationship between $n$ and its draw threshold, with a stepwise pattern that is best modeled logarithmically. We propose these draw thresholds as markers of meaningful zones of complexity for future research and help AI coloring agents, Ramsey solvers, and other related algorithms avoid trivial game setups. Finally, we hope that the dataset we created will serve as a valuable starting point for future studies to analyze and base new conjectures off of.

## 1 Introduction

Theorized by mathematician Frank Ramsey in 1930, Ramsey Theory is the study of the patterns and order that emerge in structures as they become larger and more complex. The field of study is more focused on discovering new Ramsey Numbers, $R(K_s, K_t) = n$, which state that in a graph of size $n$, a monochromatic $K_s$ or $K_t$ must exist. A monochromatic $K_s$ describes a clique or complete subgraph of size $s$ where all vertices are connected by edges of the same color. Two well-known Ramsey numbers are $R(3,3) = 6$ and $R(4,4) = 18$. Beyond discovering better thresholds for Ramsey Numbers, Ramsey Theory also has robust applications to other fields, including computer science, number theory and topology. We are most interested in its applications to game theory.

Since its conception, many variations of graph coloring games based on Ramsey Theory have arisen. One of the first such games, the "Game of Sim", was proposed by Gustavus Simmons in 1969[2], wherein two players attempt to two-color a hexagon and force their opponent to create a monochromatic triangle in the opposing color. Simmons' game is a classic example of an avoidance game because the goal is to avoid creating a monochromatic triangle in one's color. In contrast, in the achievement variation of Ramsey Games, as later introduced by Frank Harary in 1982 [3], the win condition was changed to be the first player to create a monochromatic triangle in their color. Multiple other versions have been proposed since then, including online, two-round, and offline variations [4], among others.

This paper focuses on two-player achievement games, and a win is defined as the first player to create a monochromatic $k$-clique on a graph of size $n$. We are interested in the draw threshold of such games, denoted by $D(n) = k_d$ where $n$ is the number of nodes in the graph and $k_d$ is the smallest clique size that results in all draws across all combinations of strategies. Similarly to Ramsey Theory, we hope to find a point where order supersedes chaos: regardless of what strategy each player uses and their respective advantages, the game will necessarily end in a draw. By identifying these transition points and examining trends among them, we hope to obtain the following: a) a fresh perspective on the growth and behavior of Ramsey-type structures and b) the natural performance limits that can serve as benchmarks for graph coloring algorithms. Additionally, the draw thresholds we propose could mark meaningful zones of complexity for future research and help AI coloring agents, Ramsey solvers, and other related algorithms avoid trivial game setups. Finally, we hope that our dataset can serve as a valuable starting point for future studies.

## 2 Methods

Our experiment uses a Monte Carlo, Python simulation to empirically study Ramsey two-player games on undirected graphs with $6 \leq n \leq 59$ and $3 \leq k \leq 11$. Note that the bounds of $k$ were adjusted at times to avoid running unnecessary long simulations because, although the problem is PSPACE complete [5], the time complexity still increases exponentially with $n$ and $k$. Each player was assigned one of four strategies:

| **Strategy** | **Abbreviation** | **Logic** |
|---|---|---|
| Win → Block → Random | W + B + R | if exists edge that allows you to win:<br>choose that edge to win<br>else if exists edge that stops opponent's win:<br>choose that edge to block<br>else:<br>choose a random edge |
| Win → Random | W + R | if exists edge that allows you to win:<br>choose that edge to win<br>else:<br>choose a random edge |
| Block → Random | B + R | if exists edge that prevents opponent's win:<br>choose that edge to block<br>else:<br>choose a random edge |
| Random | R | choose a random edge |

Table 1: Above are the strategies and psuedo-code assigned to each of the two players. All combinations of strategies (i.e. "P1: W+B+R vs P2: W+B+R", "P1: W+B+R vs P2: W+R", ... , "P1: R vs P2: B+R", "P1: R vs P2: R") were simulated across configurations of $(n, k)$.

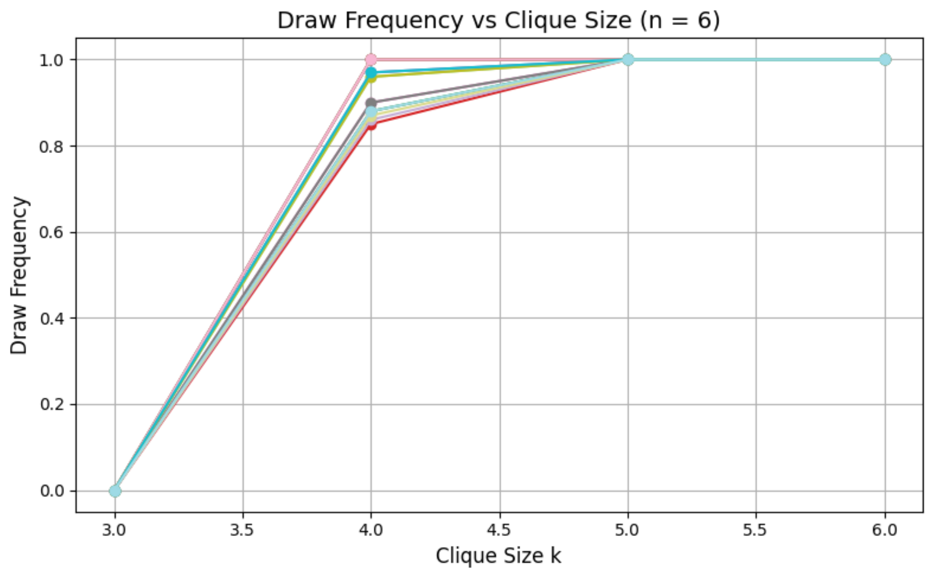
For each configuration of $(n, k)$, we tested all 16 combinations of the defined strategies (see Table 1) for Player 1 and Player 2. Each combination of strategies was simulated 100 times using different seeds, but this set of seeds was kept consistent across simulations to ensure data reproducibility. After the initial test, we revisited several configurations, specifically those that deviated from the general trends or appeared near a draw threshold and increased the number of simulations to

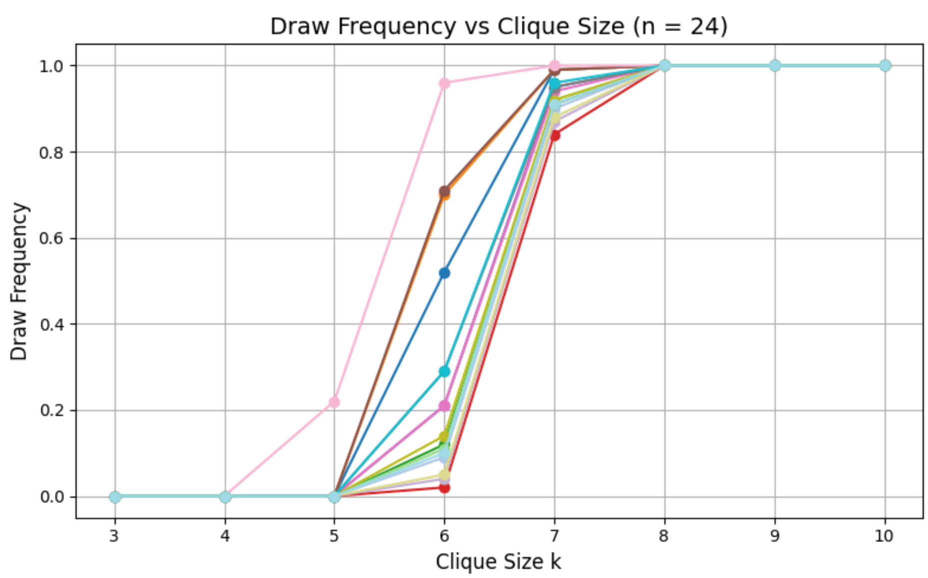
1000 to achieve more reliable results. We recorded the proportion of Player 1 wins, Player 2 wins and draws and pay special attention to the smallest $k$ such that, regardless of the combination of simulated strategies, trying to create a monochromatic $k$-clique within the $n$ nodes will always result in a draw. This was recorded as the draw threshold, $D(n)$.

# 3 Results

## 3.1 Draw Frequency vs Clique Size of Various Strategies

Keeping $n$ constant, we noticed that increasing target clique size $k$ increased the number of draws. The frequency of draws increased to 100% as we approached the draw threshold $D(n)$.



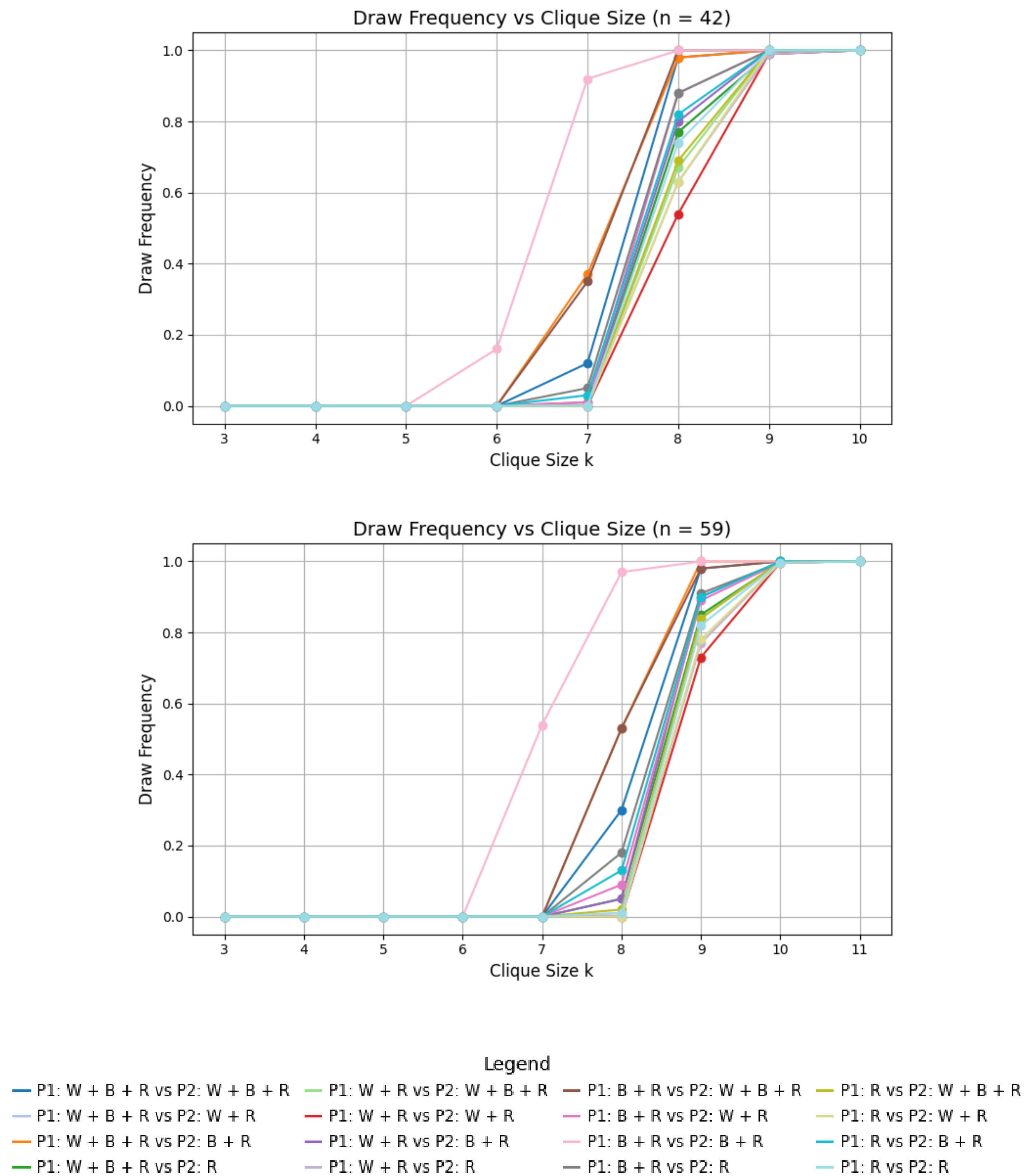


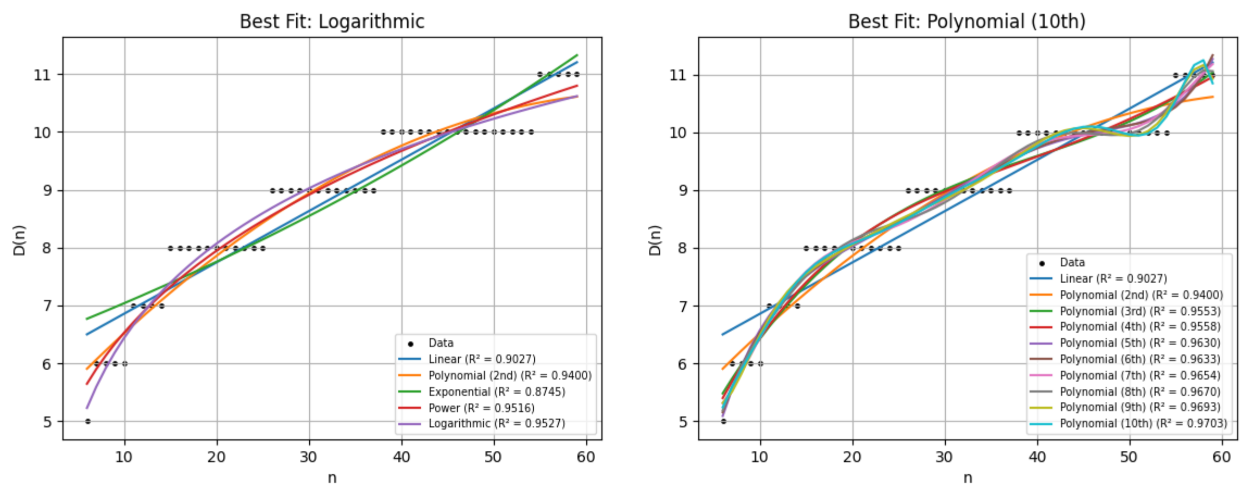

Figure 2: Comparison of draw frequencies for the 16 combinations of strategies across different clique sizes and numbers of nodes.

As the number of nodes increases, the graph shifts naturally to the right because the draw threshold increases. Moreover, by comparing the shape of the graph where $n = 6$ to that where

$n = 59$, we observe that the graph better approximates a logistic growth pattern for larger values of $n$. Shifting our focus to the specific combinations of strategies, we notice that "P1: B+R vs P2: B+R" (light pink) sticks out in every graph and deviates from general trend because it consistently results in higher draw frequencies than the other combinations of strategies. Additionally, the combinations of strategies "P1: B+R vs P2: W+B+R" (brown) and "P1: W+B+R vs P2: B+R" (orange) consistently overlay one another, indicating that they share similar draw frequency growth patterns.

### 3.2 Draw Threshold Function

Our results across simulations show a positive relationship between draw thresholds and $n$ and a stepwise, increasing pattern in draw thresholds as $n$ increases. We attempted to fit the data to various functions, such as exponential, power, logarithmic and polynomials, and used $R^2$ as our goodness of fit measure.




(a) Fit curves for a variety of different function types. The logarithmic function proved to be the best fit curve with the power function as a close second, while the exponential function performed the worst.

(b) Fit curves for polynomial functions of varying degrees (2-10). Fit increases as degree of the polynomial increases, indicating a positive correlation. However, this could be a result of unnecessary complexity.

Figure 3: Plots of $n$ vs $D(n)$ fitted with various candidate functional forms and evaluated by comparing their $R^2$ values.

While the data appears to increase stepwise, a stepwise function is ill-fit to generalize it because we are unable to provide sufficiently accurate bounds for each step - the steps increase in length inconsistently making it difficult to predict how they will grow. We find that a logarithmic function fits the data best among the selected functional forms in Figure 2(a). However, the logarithmic fit curve is outperformed by higher order polynomials (degree $> 3$). We observe a positive correlation between polynomial order and $R^2$. Fitting the data to a 10th-order polynomial yields $R^2 = 0.9703$, while a linear model only yields $R^2 = 0.9027$. We examine this correlation further using the equations and $R^2$ values for each of the functional forms above to check for unnecessary complexity.

We observe that the coefficients of the higher order polynomials (degree $\geq 3$) are often of magnitude $10^{-5}$ and smaller. This indicates that these functions are likely too complex for our

| **Equation Type** | **Equation** | $R^2$ |
|---|---|---|
| Linear | $0.08889x + 5.963$ | 0.9027 |
| Polynomial (2nd) | $-0.001297x^2 + 0.1732x + 4.908$ | 0.9400 |
| Polynomial (3rd) | $6.072 \times 10^{-5}x^3 - 0.007217x^2 + 0.3391x + 3.686$ | 0.9553 |
| Polynomial (4th) | $-7.995 \times 10^{-7}x^4 + 1.647 \times 10^{-4}x^3 - 0.01179x^2 + 0.4164x + 3.285$ | 0.9558 |
| Polynomial (5th) | $2.269 \times 10^{-7}x^5 - 3.767 \times 10^{-5}x^4 + 0.002378x^3 - 0.07179x^2 + 1.13x + 0.4237$ | 0.9630 |
| Polynomial (6th) | $3.628 \times 10^{-9}x^6 - 4.807 \times 10^{-7}x^5 + 1.623 \times 10^{-5}x^4 + 3.538 \times 10^{-4}x^3 - 0.03298x^2 + 0.7773x + 1.578$ | 0.9633 |
| Polynomial (7th) | $-6.784 \times 10^{-10}x^7 + 1.58 \times 10^{-7}x^6 - 1.473 \times 10^{-5}x^5 + 7.022 \times 10^{-4}x^4 - 0.018x^3 + 0.2362x^2 - 1.193x + 7.044$ | 0.9654 |
| Polynomial (8th) | $-4.445 \times 10^{-11}x^8 + 1.088 \times 10^{-8}x^7 - 1.097 \times 10^{-6}x^6 + 5.9 \times 10^{-5}x^5 - 0.001842x^4 + 0.03426x^3 - 0.3805x^2 + 2.584x - 2.047$ | 0.9670 |
| Polynomial (9th) | $-3.982 \times 10^{-12}x^9 + 1.12 \times 10^{-9}x^8 - 1.344 \times 10^{-7}x^7 + 8.998 \times 10^{-6}x^6 - 3.686 \times 10^{-4}x^5 + 0.009526x^4 - 0.1538x^3 + 1.47x^2 - 7.156x + 18.68$ | 0.9693 |
| Polynomial (10th) | $-2.029 \times 10^{-13}x^{10} + 6.194 \times 10^{-11}x^9 - 8.173 \times 10^{-9}x^8 + 6.108 \times 10^{-7}x^7 - 2.846 \times 10^{-5}x^6 + 8.579 \times 10^{-4}x^5 - 0.01679x^4 + 0.2091x^3 - 1.586x^2 + 6.941x - 8.217$ | 0.9703 |
| Exponential | $6.382e^{0.009728x}$ | 0.8745 |
| Power | $3.39x^{0.2842}$ | 0.9516 |
| Logarithmic | $2.362\ln(x) + 0.9905$ | 0.9527 |

Table 2: Fitted equations and $R^2$ values for each model type.

data as the higher order terms contribute very little to the overall equation. In contrast, the coefficients for the other functional forms, like logarithmic, power and second-degree polynomial, have much more reasonable coefficients, indicating a more reasonable level of complexity that fits our data. Thus, based on our data, the draw threshold $D(n)$ is best approximated by a logarithmic function: $D(n) \approx 2.362\ln(n) + 0.9905$.

### 3.3 "Player 1: W+B+R vs Player 2: R" as a Meaningful $D(n)$ Marker

While the other strategies had interesting properties, "Player 1: W+B+R vs Player 2: R" was a key indicator of whether or not we were at a draw threshold. If Player 1 had a 0% win rate, it almost always indicated that all other strategies would result in all draws as well. In the cases that this wasn't immediately apparent, increasing from 100 to 1000 trials revealed this nuance. "Player 1: W+B+R vs Player 2: R" is the strategy where Player 1's advantage is most pronounced because Player 1 has first move advantage and uses a more complex strategy against Player 2's completely random approach to selecting nodes.

# 4 Discussion

Our results show interesting patterns in draw frequencies and thresholds as both $n$ and $k$ increase. Firstly, for larger $n$, the draw frequencies increase logistically with $k$. Additionally, the draw threshold $D(n)$ exhibits a stepwise increasing pattern that is best approximated with a logarithmic function. This aligns with classical Ramsey theory results which demonstrate that in sufficiently large graphs, complete disorder is impossible [6]. While the Ramsey Numbers seek to converge on the point where a $k$-clique must be present, our paper examines the point where a $k$-clique cannot arise. Thus, it makes sense that, while the Ramsey Numbers grow in an exponential pattern, our fit curve was the inverse of that, an logarithmic curve.

Moreover, looking back at our function for modeling the draw threshold: $D(n) \approx 2.362 \ln(n) + 0.9905$, we prove mathematically that this function makes sense. Recall that $2^{k/2} \leq R(k) \leq 2^{2k}$. Hence it is true that any 2-coloring of $K_{2^{2k}}$ has a mono k-clique. Let $n = 2^{2k}$. It follows that $\log_2(n) = 2k$ and $k = 0.5 \log_2(n)$. Since $\log_2(n)$ is approx $\ln(n)/0.7$ we have $k = (5/7) \ln(n)$, which is roughly $0.714 \ln(n)$. Thus, $D(n) \geq 0.714 \ln(n)$, which is true of our function.

While exploring various combinations of strategies wasn't the focus of this paper, we mentioned some interesting patterns that arose earlier. In general, "P1: B+R vs P2: B+R" resulted in a higher proportion of games that ended in draws. This conclusion makes sense as both players always choose to block one another, but don't necessarily choose the winning edge when it is available (that is left up to chance if they end up randomly choosing an edge). Moreover, it makes sense that "P1: W+B+R vs P2: B+R" and "P1: B+R vs P2: W+B+R" would result in similar draw frequencies , but both less than that of "P1: B+R vs P2: B+R". Since one player is programmed to win when possible, it makes sense that there are less draws than when neither player is programmed to do so. However, they nonetheless exhibit similar draw frequencies because the odds of either player becoming advantaged enough to win (i.e. not being blocked by the other and successfully setting up an almost-monochromatic $k$-clique through randomly selecting edges) decreases as $n$ and $k$ increase.

While we focused on draws, the dataset our simulations produced can serve as a springboard for future studies and conjectures about strategy performance and underlying patterns. Although our dataset includes data for $3 \leq k \leq 12$ for up to $n = 59$, which is a more robust dataset than previously proposed by other studies, our optimized algorithm was limited by computational constraints on larger graphs. Given more computing power, we likely could have ran 1000 simulations instead of 100 for all configurations and increased n to be larger to find a better fitting approximation of $D(n)$. Additionally, based on the observation made in Results section 3.3, we could have also ran simulations on just "Player 1: W+B+R vs Player 2: R," using it as our indicator of draw thresholds. This would have greatly cut down on computing time and allowed us to increase the number of nodes $n$, gather more data points and create a more accurate function for $D(n)$.

Future studies could look into simulating games on larger graphs or ones with more complex strategies that are more calculated than randomly choosing an edge in the worst case. Win conditions could be changed to be not just cliques, but cycles or specific shapes, like stars. This could then be extended to more practical, real-world applications such as social networks and tournament brackets. We are also interested in potentially training AI players to play the game and exploring how our proposed draw frequencies aid their training process by avoiding trivial configurations of $n$ and $k$. An open problem remains how to train AI models, using reinforcement learning, deep neural networks or other techniques, to play Ramsey Games. The strategies used in our study were simple compared to how decision making often works in real-time game play. For example, future studies could use strategies that are more nuanced than just ”R” (Random); they could have the players pick edges that include nodes with the highest/lowest degree and examine how that affects $D(N)$ and overall performance.

In conclusion, by quantitatively modeling Ramsey Games, we provide novel insights into the growth patterns of draw frequencies and thresholds. Our study improves upon known patterns within such games and lays the groundwork for future investigation into optimal Ramsey Game strategy and how that changes as $n$ and $k$ get bigger.

## References


[1] Rosta, Vera. “Ramsey Theory Applications.” The Electronic Journal of Combinatorics, vol. 1000, Dec. 2004, p. DS13: Dec 7. Semantic Scholar, https://doi.org/10.37236/34.

[2] Simmons, Gustavus J., editor. “The Game of SIM.” Mathematical Solitaires and Games, Routledge, 1968, p. 66.

[3] Harary, Frank. “Achievement and Avoidance Games for Graphs.” North-Holland Mathematics Studies, edited by Béla Bollobás, vol. 62, North-Holland, 1982, pp. 111–19. ScienceDirect, https://doi.org/10.1016/S0304-0208(08)73554-0.

[4] Friedgut, Ehud, et al. “Ramsey Games Against a One-Armed Bandit.” Combinatorics, Probability and Computing, vol. 12, no. 5–6, Nov. 2003, pp. 515–45. Cambridge University Press, https://doi.org/10.1017/S0963548303005881.

[5] Slany, W. “Graph Ramsey Games.” Electron. Colloquium Comput. Complex., Nov. 1999. Semantic Scholar,
https://www.semanticscholar.org/paper/c5411fbded7c7a7db01b20b0a7df689e8c295ea0.

[6] Erdös, P., and G. Szekeres. “A Combinatorial Problem in Geometry.” Compositio Mathematica, vol. 2, 1935, pp. 463–70. eudml.org, https://eudml.org/doc/88611.