Simulating Balatro Strategy Archetypes

Studying the Effect of the Joker and Shop Strategies using Monte Carlo Simulation

Donghyun Kim

Abstract

This project implements a simplified simulation of a Balatro-inspired poker roguelike game and uses Monte Carlo methods to assess several rule-based strategies. The model uses a 52-card deck, a five-card poker hand, deterministic scoring rules, and modifiers such as Jokers, Vouchers, and an interest-based economy. The gameplay progresses through Small, Big, and Boss blinds with chip requirements increasing with the ante. Five strategies were analyzed: Pair/Trips, Scaling Joker, Flush/Suit, Straight/Shortcut, and Pareidolia/Face. Each was simulated 1,000 times with the maximum ante of eight. The strongest performers were the Pairs/Trips and the Scaling Joker, which achieved a win rate of 10.8% and 10.3%, respectively. They consistently cleared more blinds while accumulating more money. Although the model captures the foundational structure of Balatro, a true recreation of the game would require substantially more complexity. Recommended improvements would include better deck mechanics, more Jokers and shop items, realistic scoring interactions, and various strategies that adapt dynamically to game situations.

Project Overview

Background

Balatro is a deck-building poker roguelike in which the player advances through a series of progressively demanding blinds by building poker hands that generate chips and score. After each round, players visit the shop where they can purchase Jokers, passive modifiers that can develop long-term synergies, as well as other items that can alter the deck, scoring engine, or the economy. Throughout a run, these decisions build a strategic identity, typically centered on scaling multipliers or unique combinations of effects that significantly increase the scoring potential.

Finding an optimal strategy in Balatro can be challenging because the game is both stochastic and combinatorial. Sources of randomness, such as card draws, shop rolls, and availability of certain items, can push a run in very different directions, and many powerful synergies only reveal their strength over a long sequence of rounds. As a result, the interactions between items are nonlinear and path-dependent, making analytical reasoning insufficient. Consequently, simulation is among the limited practical methods for evaluating strategic decisions and comparing archetypes under uncertainty.

Project Goals

The goal of this project is to build a simulation engine for a simplified version of Balatro and use it to compare how game strategies behave under certain controlled conditions. I relied on two sources to identify which strategies to examine: a Game Rant article outlining typical player approaches (Harris, 2025) and a community guide explaining how certain synergies evolve powerfully (DrBoomMD, 2023). These discussions inspired the selection of five strategies: Pairs/Trips, Scaling-Jokers, Flush/Suit, Straight-oriented, and Pareidolia/Face.

Related Work

Simulation-based analysis of card and strategy games has a long history in computational game research. Several previous studies apply Monte Carlo methods to assess decision-making in environments that combine randomness, complex state spaces, and long-term strategic dependencies. Although Balatro is a relatively new game with limited academic attention, similar approaches are found across related fields.

Monte Carlo techniques are widely used to model various games. Fitrianawati et al. (2022) used Monte Carlo sampling to evaluate the difficulty levels for various card combinations in the 24-card game. Neller et al. (2016) reviewed top-performing Poker Square agents that used Monte Carlo rollouts to estimate the value of different board placements under parameterized scoring systems. Zook et al. (2015) applied the Monte Carlo Tree Search (MCTS) to Scrabble to analyze game balance. These studies all show that simulation is a powerful tool for exploring strategy when analytical methods are impossible to implement.

Computational tools for Balatro exist within the player community. For example, the open-source Balatro Calculator (efhiii, 2024) computes a player's hand scores with different card and Joker combinations. This project aims to advance the Balatro community by developing a simplified, end-to-end Balatro-like simulator and applying Monte Carlo analysis to compare different high-level strategic archetypes.

Model Design and Implementation

Cards, Hands, and Base Scoring

The foundation of the simulator is a standard 52-card deck, a combination of thirteen ranks and four suits. For reproducibility, the deck can be shuffled using a pseudorandom number generator initialized with a user-provided seed. Whenever cards are needed, they are drawn from the top of the deck. If there are no more cards left, a new deck is generated.

Each five-card hand is evaluated using conventional poker hand orders, ranging from high card to straight flush. The evaluation identifies patterns by counting repeated ranks, checking if all suits match, and verifying whether the ranks form a consecutive sequence. Within each category, the algorithm records the primary ranks and any kicker cards to fully specify the relative ordering of hands. This evaluation process is the basis for all scoring throughout the simulation.

Every hand type carries a base chip value and a base multiplier. Additional chips come from the summed rank values of the cards, while upgrades may further increase either chips or multipliers. The resulting score for a single hand is:

$$\text{Total Score} = (\text{base chips} + \text{upgrades} + \text{card values}) * (\text{base multiplier} + \text{upgrades})$$

This raw score is then modified by any active Jokers.

Progression through the game is organized into antes, each consisting of three blinds: Small, Big, and Boss. Every blind specifies a target chip amount, a limited number of playable hands, and a fixed number of discards. Targets scale with the ante level, making later blinds significantly more demanding. Clearing a blind award in-game currency, while failing one costs a life point.

Jokers act as passive scoring modifiers and are crucial to strategic differentiation. Although the real Balatro contains 150 Jokers, this simulation uses a curated subset aligned with the five archetypes under study: pair-oriented effects, straight-related effects, flush and suit synergies, face-card and Pareidolia interactions, and economy-scaling Jokers. Jokers stack multiplicatively or additively, depending on their design.

Between blinds, the player enters a shop that offers two items: one guaranteed Joker and either a second Joker or a Voucher that increases slot capacity or reduces the costs of items. Prices increase with the ante, and the strategy being tested determines what, if anything, to purchase.

At the end of each blind, the player earns interest proportional to their unspent money. This rewards conservative spending and enables strategies that rely on economy-driven multipliers.

During a blind, strategies may choose to discard hands before playing them. Discard behavior varies widely across archetypes: some insist on playing only pairs or better, others pursue flush or straight patterns, and some avoid extremely weak high-card hands.

Strategy Archetypes

The simulator implements five high-level archetypes to compare how different strategies perform under identical game conditions.

Pair/Trips

This archetype relies on the consistency of pair-based hands. The strategy prioritizes shop items that reward pairs, two pairs, three-of-a-kind, and full houses. It discards any hand that is not at least a pair. Early upgrades can create reliable chip gains as pairs occur more frequently than straights or flushes.

Scaling Joker

As the name suggests, this archetype targets Jokers that grow more powerful over time. Unlike other strategies that chase specific patterns, the Scaling Joker strategy aims to survive long enough to capitalize on the exponential scaling. The core idea is to bank steady chip gains and preserve money for interest. When executed successfully, this approach can produce extremely high multipliers in later blinds.

Flush/Suit

This archetype attempts to form flushes as frequently as possible. It favors Jokers that reward large suit counts or specific suits. The strategy discards all hands that do not have at least three cards of the same suit to increase the likelihood of hitting a flush. This approach is intuitive when suit-synergy items appear, but can be weak in their absence because flushes are relatively rare and discards can be costly.

Straight/Shortcut

This archetype commits to building straights or "near-straight" (four cards in a row) sequences. The strategy prioritizes Jokers that reward straight formation. To maintain the structural patterns needed for straights, the strategy aggressively discards hands that are unlikely to evolve toward the straight.

Pareidolia/Face

This archetype revolves around a specific Joker: Pareidolia. Pareidolia causes all cards to count as faces. Once this key effect is acquired, the strategy rarely discards, since nearly any hand becomes profitable under strong face-based multipliers. The archetype typically snowballs when face-count bonuses stack multiplicatively. However, this Joker must be acquired early to achieve its high payoff.

Experiment Design

The study uses a Monte Carlo simulation framework to evaluate how each strategy performs under identical stochastic conditions. Each run begins with a pseudo-random number generator that is initialized using a seed offset. For each of the five strategy archetypes, the simulator executes 1,000 independent runs.

A run is considered a "win" if the player successfully clears all three blinds up to the maximum ante of eight. The simulation records various statistics to compare the strategies: the proportion of successful runs, the average ante reached, the average number of blinds cleared, and the average remaining money at the end of the run. These metrics allow for a more nuanced assessment of overall performance.

Results

The results were computed with 95% confidence intervals (CIs) for all outcomes. Win-rates were treated as binomial proportions, while continuous outcomes such as ante reached, blinds cleared, and final money used nonparametric bootstrap CIs with 10,000 resamples. Distributional behavior was visualized using empirical cumulative distribution functions (ECDFs) and boxplots.

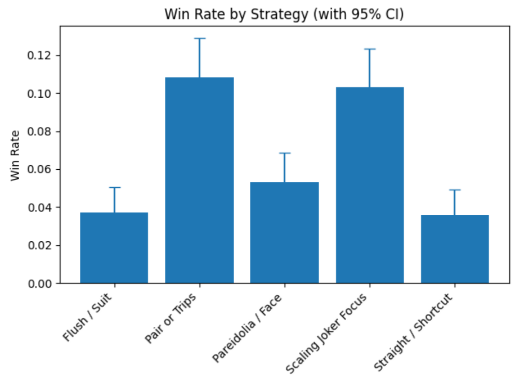
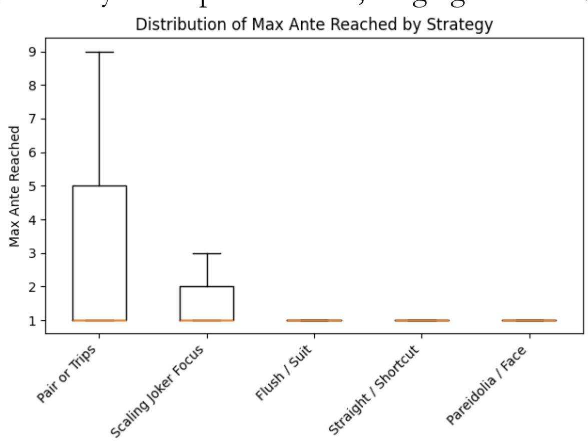


Figure 1. Boxplot showing Win Rate by Strategy

The Pair/Trips and Scaling Joker strategies achieved the highest success rates, with mean win probabilities of 10.8% (95% CI: 9.0-12.9%) and 10.3% (95% CI: 8.6-12.3%), respectively. The CIs for Pair/Trips and Scaling Joker overlap, suggesting that there is no statistically significant difference between the two top performers in this simulation. The remaining strategies had significantly lower performance, ranging from 3.6% (Straight/Shortcut) to 5.3% (Pareidolia/Face).



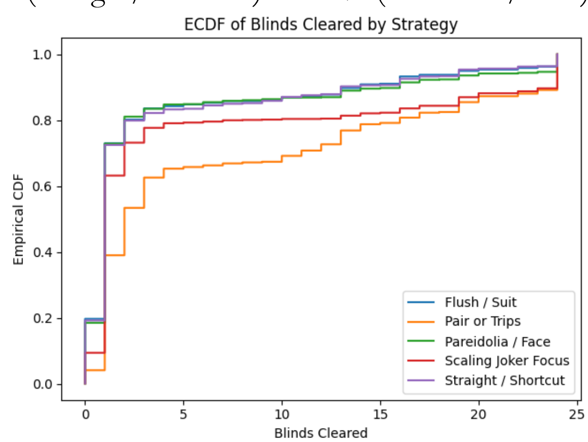


Figure 2. Distribution of Max Ante Reached by Strategy Figure 3. ECDF of Blinds Cleared by Strategy

Differences between strategies became more apparent when examining the run progression in depth. Pair/Trips reached an average ante of 3.05 (95 % CI: 2.88-3.23), substantially higher than all other strategies. Its ECDF curve (Figure 3) shows a pronounced right-tail, with a subset of runs pushing to antes 6-8. Although Scaling Joker Focus produced a comparable win rate, its median ante reached was much lower (mean=2.43, 95% CI: 2.26-2.60). The ECDF shows this. The curve is noticeably left-shifted compared to Pair/Trips. All other strategies remained clustered near ante 2, which is consistent with their lower success probabilities.

The economic performance largely mirrored progression depth. Pair/Trips generated the highest final-money average (37.6, 95% CI: 34.2-41.0), followed by Scaling Joker Focus (28.2, 95% CI: 24.9-31.5). These patterns naturally correspond closely to the average number of blinds cleared (7.11 vs. 5.17). Figure 2 shows that Flush/Suit, Straight/Shortcut, and Pareidolia/Face all displayed narrow boxplots near the minimum possible values, indicating that in most runs, these strategies failed early.

Limitation

Several limitations of the simulator likely contributed to the observed strategy hierarchy. For example, the shop offers only two items per visit, with no rerolls, no rarity tiers, with little variety of items. This drastically reduces the diversity of viable builds and disproportionately suppresses strategies needing multi-card synergies for scaling. Additionally, some of the Joker effects were simplified to avoid complex interactions; for instance, Smeared was modeled as a flat flush multiplier rather than a true suit-merging mechanic, and Shortcut rewarded straights (and "almost-straights") with a multiplier boost rather than allowing 1 rank gap to build straights. Most importantly, the ante and blind scaling parameters are approximations rather than reflecting the actual Balatro difficulty curves, which may have weighed early survival differently than the actual game.

These limitations mean that the simulation is not intended to reproduce Balatro exactly. Rather, it provides an environment for testing how high-level strategic archetypes perform under probabilistic constraints.

Conclusion

This study demonstrates that even a simplified model of a Balatro-style scoring environment can reveal meaningful differences in strategic behavior. Pair/Trips emerged as the most consistent performer, and Scaling Joker followed closely, achieving a similar win rate but with outcomes that were more dependent on early access to key Jokers. In contrast, strategies that relied on forming straights, flushes, and face-card synergies rarely gained momentum. This shows that these archetypes rely on item diversity and more flexible card manipulation than the current model provides.

While the simulator captures the broad structure of Balatro, its various constraints shape which strategies can succeed. Even so, this experiment highlights that consistent, low-variance approaches tend to outperform high-synergy builds when resources are scarce and randomness is uncompromising.

Future Work

Several future developments could improve the realism and predictive power of this simulator. First, in order to enable strategies that depend on deck shaping, deck manipulation mechanics should be put into place. Second, the shop could be expanded to include rerolls, item variety, and rarity tiers; this will allow complex Joker synergies to emerge more frequently. Third, the model should have a more realistic hand selection system, rather than a fixed 5-card draw. Finally, calibrating blind difficulty and scaling functions using Balatro data would yield more accurate performance curves. These extensions to the model could support a more meaningful exploration of optimal strategy comparison and help in evaluating the relative strength of archetypes under more practical circumstances.

Works Cited

DrBoomMD. (2023). How to win chips and influence mult (A thorough guide to beating Balatro) [Online forum post]. Reddit. https://www.reddit.com/r/balatro/comments/1bbh75a/how_to_win_chips_and_influence_mult_a_thorough/

Balatro Calculator. efhiii. https://efhiii.github.io/balatro-calculator/

Fitrianawati, M., Aliansyah, Z., Peni, N. R., Farid, I. W., & Hakim, L. (2022). Monte Carlo method at the 24 game and its application for Mathematics Education. *Journal of Honai Math*, *5*(2), 83–94. https://doi.org/10.30862/jhm.v5i2.250

Harris, S. (2025, February 2). *Balatro: Best Strategies*. Game Rant. https://gamerant.com/balatro-best-strategies/

Neller, T. W., Yang, Z., Messinger, C. M., Anton, C., Castro-Wunsch, K., Maga, W., Bogaerts, S., Arrington, R., & Langley, C. (2016). Monte Carlo approaches to parameterized poker squares. *Lecture Notes in Computer Science*, 22–33. https://doi.org/10.1007/978-3-319-50935-8_3

Zook, A., Harrison, B., & Riedl, M. O. (2019). Monte-Carlo tree search for simulation-based strategy analysis. arXiv. https://arxiv.org/abs/1908.01423