Article

Not peer-reviewed version

Modeling and Experimental Analysis of Troop Pathfinding and Target Selection Algorithms in Clash of Clans–Style War Scenarios

Chimeng Ly *

Posted Date: 30 January 2026

doi: 10.20944/preprints202601.2386.v1

Keywords: game AI; pathfinding; target selection; game balance

Article

Modeling and Experimental Analysis of Troop Pathfinding and Target Selection Algorithms in Clash of Clans–Style War Scenarios

Chimeng Ly

Department of Computer Science, Paragon International University; cly5@paragoniu.edu.kh

Abstract

Pathfinding and target selection algorithms play a critical role in real-time strategy and mobile games, directly influencing gameplay balance, fairness, and player skill expression. Unlike traditional shortest-path algorithms such as A*, many commercial games intentionally employ simplified or constrained pathfinding to preserve strategic depth. This paper presents a modeling and experimental analysis of troop pathfinding and target selection behavior inspired by Clash of Clans, with a focus on Clan War attack scenarios involving troop movement toward Town Halls and defensive structures. Since the internal implementation of Clash of Clans is proprietary, this study proposes a behavioral approximation model based on observable in-game mechanics. A hybrid algorithm combining greedy nearest-target selection, local obstacle-aware movement, and priority-based cost functions is designed and evaluated. Multiple simulated base layouts with varying densities and wall configurations are tested. Results show that intentionally non-optimal pathfinding enhances game balance, prevents deterministic outcomes, and promotes strategic base design. The study follows the IMRAD structure and applies standard experimental game AI research methodologies.

Keywords: game AI; pathfinding; target selection; game balance

---

1. Introduction

Artificial intelligence (AI) algorithms are essential components of modern digital games, governing non-player character (NPC) behavior such as movement, combat decisions, and interaction with dynamic environments. In many contemporary games, especially those within the strategy genre, AI systems are responsible for automating complex actions that would otherwise require constant player input. Among these actions, troop movement and navigation are particularly critical, as they directly affect player experience, perceived fairness, and competitive balance [1]. Poorly designed movement algorithms can lead to frustrating gameplay, while overly efficient algorithms may remove strategic depth and reduce player agency.

In academic research, pathfinding is often studied through optimal algorithms such as A*, Dijkstra's algorithm, and their variants. These algorithms are designed to compute the shortest or least-cost path between two points in a graph-based environment. While such approaches are mathematically sound and widely used in theoretical studies, their direct application in commercial games is not always desirable. Perfect pathfinding can produce predictable and deterministic outcomes, making gameplay less engaging and limiting the impact of player strategy. As a result, many commercial games deliberately constrain or simplify pathfinding behavior to support balanced and enjoyable gameplay.

Clash of Clans is a globally popular mobile strategy game that combines base-building, resource management, and real-time combat. A defining feature of the game is its automated combat system, particularly during Clan Wars, where players deploy troops that independently navigate enemy bases. Once deployed, players have no direct control over troop movement; instead, all navigation and

 doi:10.20944/preprints202601.2386.v1

targeting decisions are handled by the game's AI system. Troops move through bases, interact with walls and defenses, and attack buildings such as defensive structures and the Town Hall based entirely on internal decision-making logic [2].

Unlike traditional player-controlled movement systems, troop behavior in Clash of Clans relies on a combination of pathfinding and target selection rules. Troops do not simply move toward the Town Hall along the shortest possible route. Instead, they prioritize nearby buildings according to troop-specific targeting rules, environmental constraints, and obstacle placement. This design choice allows players to influence troop behavior indirectly through base layout, wall configuration, and building placement, rather than through real-time control.

A key design philosophy underlying Clash of Clans is the prioritization of game balance over algorithmic optimality. By avoiding globally optimal pathfinding, the game encourages strategic base design and meaningful decision-making on both offense and defense. Walls, funnels, and layered defenses become critical tools for shaping troop movement, creating opportunities for skilled players to outplay opponents through intelligent layout design rather than relying solely on troop strength or numerical superiority.

The objective of this research is to model and experimentally analyze Clash of Clans–style troop pathfinding and target selection behavior using a formal algorithmic framework. Since the internal implementation of the game's AI is proprietary, this study focuses on behavioral approximation based on observable in-game mechanics. By designing controlled experiments and evaluating different base layouts and cost-function parameters, this paper demonstrates how intentionally non-optimal AI behavior contributes to balanced and engaging gameplay. Ultimately, the study aims to bridge the gap between academic pathfinding research and practical game AI design, highlighting the importance of balance-driven algorithms in commercial strategy games.

2. Background and Related Work

Artificial intelligence research in digital games has evolved significantly over the past decades, driven by increasing computational power and the demand for more engaging player experiences. In strategy games, AI systems are responsible for automating complex behaviors such as unit movement, target selection, and combat decisions. Unlike action games where player reflexes dominate, strategy games rely heavily on algorithmic decision-making to create meaningful challenges and maintain competitive balance [3] .

A major area of research in game AI focuses on pathfinding algorithms, which determine how units move through an environment while avoiding obstacles. Classical algorithms such as A*, Dijkstra's algorithm, and breadth-first search are frequently discussed in the academic literature due to their optimality guarantees [4]. However, these algorithms are often computationally expensive and may produce predictable outcomes when applied directly to commercial games. As a result, many game developers adopt simplified or heuristic-based approaches that trade optimality for performance and quality of the game.

Mobile strategy games introduce additional constraints compared to PC or console games. Limited processing power, real-time responsiveness, and the need for deterministic outcomes across devices require AI systems to be both efficient and robust. Consequently, mobile games often avoid global path computation and instead rely on local decision-making and greedy target selection [5]. This approach reduces computational overhead while still producing believable and strategically meaningful unit behavior.

2.1. Algorithms in Strategy and Mobile Games

Different game genres employ distinct artificial intelligence (AI) techniques depending on gameplay requirements, scale, and hardware constraints. Prior research in game AI emphasizes that algorithm selection is not solely a technical decision but also a design choice that directly affects player experience and game balance . The most common approaches in strategy-oriented games are summarized below:

 doi:10.20944/preprints202601.2386.v1

- **Real-Time Strategy (RTS)** Games typically involve controlling large numbers of units across expansive maps. To manage this complexity, AI systems often use global pathfinding algorithms such as A*, hierarchical pathfinding, or flow-field navigation. These approaches enable coordinated unit movement and efficient navigation at scale. However, academic studies note that globally optimal pathfinding can result in highly predictable behavior, which may reduce strategic diversity if applied directly without modification .
- **Mobile Strategy Games** prioritize computational efficiency, deterministic behavior, and stability due to limited processing power and the need for consistent performance across devices. Instead of computing globally optimal paths, these games frequently rely on simplified navigation combined with rule-based decision-making. This design ensures smooth real-time execution and predictable outcomes, which are essential for fairness in competitive mobile environments .
- **Tower Defense and Base Attack Games** Tower defense and base attack games commonly employ rule-based targeting systems combined with local navigation rather than full-map path planning. Units dynamically select nearby targets based on predefined priorities such as proximity or building type, and movement decisions are recalculated after each interaction. This approach allows level and base design to strongly influence unit behavior, enabling players to manipulate attack paths through strategic placement of walls, defenses, and structures .

Table 1. AI Pathfinding Approaches Across Strategy Game Genres.

Game Genre	Common AI Techniques	Path Optimality	Impact on Gameplay
RTS Games	A*, hierarchical, flow-field	High	Predictable large-scale movement
Mobile Strategy Games	Local, greedy navigation	Medium	Stable and fair gameplay
Base Attack Games	Rule-based + local movement	Low	Strong player influence
CoC Algorithms	Priority-based, wall-aware	Intentionally low	Balanced strategic play

These design principles are clearly illustrated in Clash of Clans, where troops do not follow globally optimal paths to the Town Hall. Instead, they prioritize nearby valid buildings according to troop-specific rules and retarget dynamically after each structure is destroyed. According to Millington and Funge, commercial games rarely implement “pure” AI algorithms; instead, they adopt hybrid systems tuned for gameplay experience and balance rather than strict mathematical optimality . Clash of Clans exemplifies this philosophy by intentionally constraining troop pathfinding to preserve strategic depth and fairness.

2.2. *Pathfinding vs Target Selection*

In many strategy games, pathfinding and target selection are treated as two distinct but interconnected processes. Traditional **pathfinding** algorithms, such as A* or Dijkstra’s algorithm, assume that the destination is fixed and known in advance. Once a goal is defined, the algorithm computes a path that minimizes distance or cost while avoiding obstacles. This approach is common in academic studies and large-scale strategy games where units must travel efficiently across expansive maps.

Target selection, on the other hand, focuses on deciding what an agent should attack or move toward rather than how it should reach a specific destination. **Target selection** mechanisms typically evaluate potential objectives based on criteria such as proximity, priority, building type, or threat level. These decisions are often dynamic and may change frequently as the game state evolves.

In Clash of Clans, target selection is emphasized before pathfinding. Troops do not begin with a fixed global objective such as the Town Hall. Instead, they first identify the nearest valid target according to troop-specific rules, such as defensive structures or resource buildings. Once a target is

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 January 2026 doi:10.20944/preprints202601.2386.v1

selected, troops perform local movement toward that target, recalculating their decisions after each building is destroyed.

This design choice significantly reduces computational complexity, which is especially important for mobile platforms with limited processing power. By avoiding full-map path computation, the game ensures stable real-time performance even during large-scale attacks involving many troops. More importantly, prioritizing target selection allows base layout design to play a critical role in shaping troop behavior. Walls, funnels, and building placement can redirect troops toward less critical structures, delaying access to the Town Hall and increasing defensive effectiveness.

As you can see in Figure 1, the war has declared that participates of each clan have the same or similar Town hall to each other, so it will be fair for the each clan members to attack each other during the attack time. For this Figure 1 has show the path finder and the target selection that can make the game more realistic and enjoyable because both clans can attack each other fairly by their own ability or the strategy that they have.

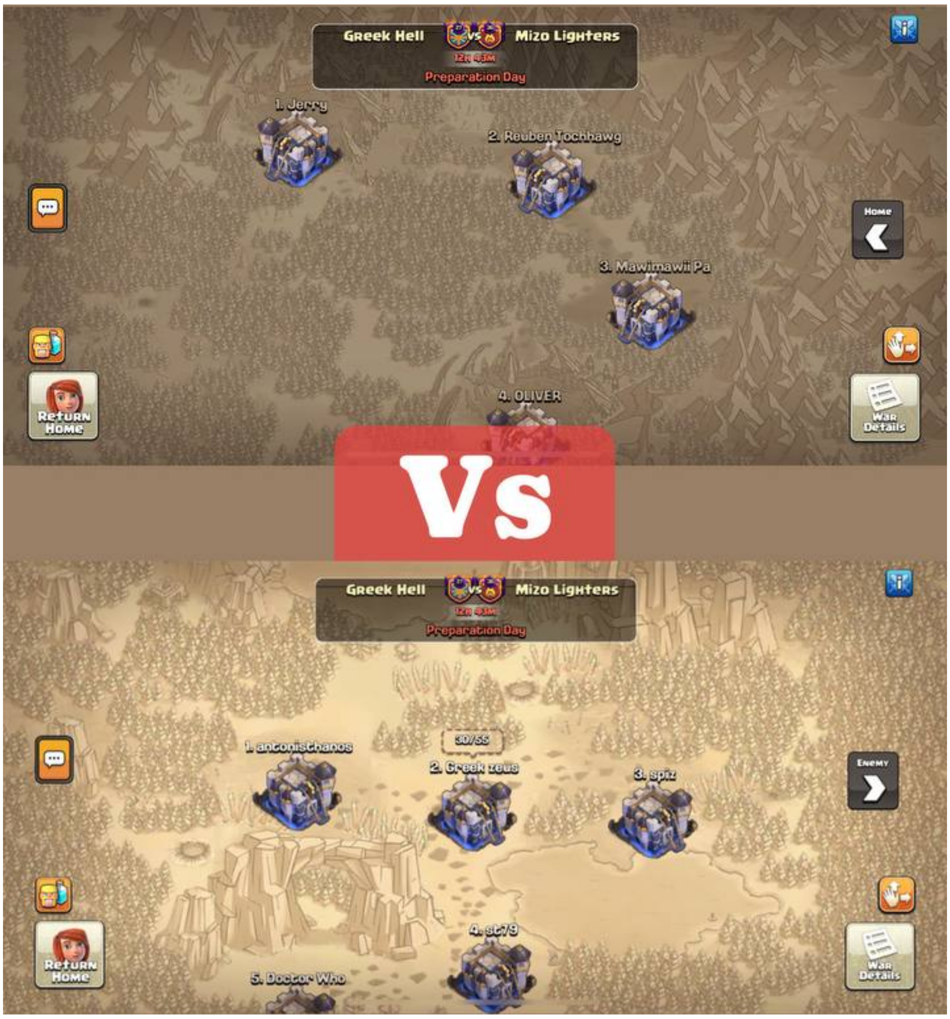

Figure 1. Clash of Clans troop behavior illustrating target selection before local movement [6].

2.3. Academic Research on Game Balance

Academic research on game artificial intelligence increasingly emphasizes that the effectiveness of an AI system should not be evaluated solely based on computational performance or optimal decision-making. Instead, researchers argue that AI behavior must be assessed in terms of its impact on player experience, fairness, and game balance. Yannakakis and Togelius highlight that game AI should support engagement, strategic diversity, and long-term replayability rather than merely demonstrating algorithmic efficiency . In this view, an AI system that is mathematically optimal may still be undesirable if it leads to repetitive or unbalanced gameplay outcomes.

Laird and van Lent further describe games as ideal environments for AI research because they combine real-time constraints with complex, interactive decision spaces . Unlike traditional benchmark problems, games require AI systems to operate under strict timing constraints while interacting continuously with human players. This makes games particularly suitable for studying the trade-offs between optimality, responsiveness, and player satisfaction. As a result, many researchers advocate for AI designs that intentionally incorporate limitations to preserve challenge and balance.

3. Algorithm Design and Formal Model

3.1. Behavioral Approximation and Hybrid Decision Model

The internal artificial intelligence implementation of Clash of Clans is proprietary and not publicly accessible. Consequently, this study does not attempt to reverse-engineer or replicate the game's source code. Instead, a behavioral approximation model is proposed, derived from observable in-game troop behavior, official documentation, and publicly available developer discussions. This approach is consistent with established academic practice when analyzing commercial games with closed-source implementations.

The proposed model represents troop behavior as a hybrid decision-making process that separates what a troop chooses to attack from how it moves toward that target. Rather than computing a global path to a fixed objective, troop behavior is modeled as a sequence of local decisions that evolve dynamically during an attack. Specifically, the model consists of three tightly coupled components: target selection, local movement, and dynamic retargeting.

In the target selection stage, troops evaluate nearby buildings according to predefined rules based on troop type and building category. Once a target is selected, the troop performs local, obstacle-aware movement toward that target. After the destruction of a building, the decision process repeats, allowing the troop to dynamically retarget based on the updated environment. This hybrid structure avoids global path planning while maintaining responsive and predictable behavior, aligning with design principles commonly observed in mobile strategy games. The proposed model uses a hybrid target selection and pathfinding approach. Each target t is evaluated using the cost function:

3.2. Target Cost Function and Local Movement Strategy

To formalize the target selection process, each potential target **t** is evaluated using a weighted cost function:

$$C(t) = \alpha D(t) + \beta W(t) + \gamma P(t) \tag{1}$$

where:

- $D(t)$ represents the Euclidean distance between the troop and the target
- $W(t)$ denotes wall resistance or obstacle cost encountered along the path
- $P(t)$ is a priority value assigned to the target (e.g., defensive structure, resource building, or Town Hall)
- α, β, γ are tunable weights that control the influence of each factor.

Targets with the lowest cost value are selected for engagement. This formulation captures the observed behavior in Clash of Clans, where troops do not simply attack the most important objective, such as the Town Hall, but instead prioritize nearby and accessible structures depending on troop-specific rules. By adjusting the weighting parameters, the model can represent different troop behaviors while maintaining deterministic and interpretable decision-making.

Once a target has been selected, troop movement is governed by a local obstacle-aware navigation strategy rather than full-map pathfinding. Troops advance toward the target while responding to immediate obstacles such as walls. If a wall blocks the direct path, the troop attacks the wall instead of searching for an alternative global route. After each building or obstacle is destroyed, movement direction is recalculated locally. This strategy significantly reduces computational complexity and ensures real-time responsiveness, which is critical for mobile platforms and large-scale attacks involving many units.

where $D(t)$ is distance, $W(t)$ represents wall resistance, and $P(t)$ denotes target priority.

4. Research Methodology

This section describes the experimental design used to evaluate the proposed Clash of Clans–style troop pathfinding and target selection model. The methodology focuses on creating a controlled simulation environment, defining experimental variables, and applying systematic testing procedures to ensure fairness, repeatability, and meaningful comparison across scenarios [7].

4.1. Please Add Title

A custom grid-based simulation environment was created to represent Clash of Clans–style war bases. The environment abstracts key gameplay elements while preserving the structural features that influence troop behavior. This approach allows controlled experimentation without relying on proprietary game code [8].

The simulated environment includes the following components:

Walls Walls are modeled as blocking structures that increase traversal cost and may require direct destruction before troops can proceed. Wall placement is a critical factor in shaping troop movement, as it determines whether troops attack obstacles or reach interior buildings directly.

Defensive Buildings Defensive structures represent high-priority targets that influence troop targeting behavior. These buildings are strategically placed to simulate realistic war base layouts and to observe how troops respond to nearby threats and defensive density.

Resource Buildings Resource buildings act as low- to medium-priority targets. Their inclusion allows the simulation to capture diversion effects, where troops are drawn away from the Town Hall due to proximity-based target selection.

Town Hall Core The Town Hall is positioned at the core of the base and represents the primary strategic objective. Its placement allows analysis of how base design and wall complexity delay or prevent direct access by attacking troops.

4.2. Experimental Variables and Testing Procedures

The experiments were designed by systematically varying base characteristics while keeping troop properties constant. This controlled approach enables clear observation of how environmental complexity influences troop behavior and algorithm performance.

The following experimental variables were tested:

Base Density Base density describes the overall compactness of buildings within the layout and was categorized as sparse, medium, or compact. Sparse bases provide more open paths, while compact bases increase the likelihood of wall interactions and target diversion.

Wall Complexity Wall complexity refers to how walls are arranged within the base. Low-complexity layouts use minimal walls, maze-like layouts create indirect paths, and layered layouts introduce multiple defensive rings. This variable directly affects wall resistance and local movement decisions.

Troop Type A generic melee troop was used in all experiments. This troop type prioritizes nearby buildings and directly attacks blocking walls, making it well suited for studying the interaction between target selection and obstacle-aware movement.

To evaluate algorithm behavior under these conditions, the study applied the following testing methods:

Black-box Testing The algorithm was evaluated based solely on observable inputs and outputs, without inspecting internal decision states. This mirrors real-world analysis of commercial games with proprietary AI implementations.

Scenario-based Testing Each base configuration represents a distinct scenario designed to test specific behaviors, such as funneling effectiveness or wall-induced path diversion.

 doi:10.20944/preprints202601.2386.v1

Stress Testing Compact bases with layered walls were used to examine algorithm performance under high-complexity conditions, where local decision-making is repeatedly triggered.

Repeatability Testing Each scenario was executed five times to ensure consistent outcomes and to reduce the impact of incidental variation. Results were averaged to improve reliability.

This combination of controlled variables and systematic testing methods ensures that observed outcomes can be attributed to algorithmic behavior rather than random variation or uncontrolled factors.

5. Experiments and Results

This section presents the results of the experimental evaluation of the proposed Clash of Clans–style troop pathfinding and target selection model. Each experiment focuses on a specific factor that influences troop behavior, including base layout density, target priority weighting, and the trade-off between balance and optimality. The results are presented in a simple and interpretable manner to highlight their impact on gameplay balance.

5.1. Experiment 1: Base Density Impact

This experiment examined how base density affects troop movement and attack outcomes. Three base layouts were tested: sparse, medium, and compact. In sparse bases, buildings were spread out with fewer walls. In compact bases, buildings were tightly packed and surrounded by multiple wall layers.

Table 2. Base Density vs Attack Outcome.

Base Type	Time to Town Hall	Walls Destroyed	Success
Sparse	Short	Low	Yes
Medium	Medium	Medium	Yes
Compact	Long	High	Sometimes

Observation:

Compact bases significantly delay troop progress and increase wall interactions. As base density increases, troops spend more time attacking walls and nearby buildings before reaching the Town Hall. This demonstrates how base layout design can strongly influence troop behavior without changing troop.

5.2. Experiment 2: Priority Weight Variation

This experiment analyzed the effect of changing the **priority weight** (γ) in the target selection cost function. The priority weight controls how strongly troops prefer high-value targets such as defensive buildings or the Town Hall [9].

When the priority weight was increased, troops were more likely to ignore nearby buildings and instead move toward high-priority targets. While this behavior sometimes allowed faster access to important structures, it also caused troops to take longer paths and engage more defenses along the way.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 January 2026 doi:10.20944/preprints202601.2386.v1

Wall Breakers required to break through Walls

		Wall Level										
		1	2	3	4	5	6	7	8	9	10	11
Wall Breaker Level	1	1	2	2	2	3	5	6	7	9	12	15
	2	1	1	2	2	3	4	4	5	7	9	11
	3	1	1	1	1	2	3	3	4	5	6	8
	4	1	1	1	1	2	2	2	3	4	5	6
	5	1	1	1	1	1	2	2	2	3	3	4
	6	1	1	1	1	1	1	2	2	2	3	3

Figure 2. Priority-based target selection in Clash of Clans, where the troops wall breaker and the wall met the required to break it.

Result:

Higher priority weights increased attack time and troop losses, indicating that overly aggressive target selection can reduce overall attack efficiency. So if we played ground troop we need to use wall breaker to break the wall to let the other troop go to attack the defense without blocking from the wall so it will save the time and lead to get three stars [10].

5.3. Experiment 3: Balance vs Optimality

The final experiment compared balanced AI behavior with a purely optimal pathfinding approach. In this scenario, wall resistance (β) was set to zero, allowing troops to effectively follow the shortest possible path to the Town Hall.

Under this configuration, troops reached the Town Hall very quickly and with minimal interaction with walls or defensive structures [9].

Observation:

Although optimal pathfinding improved attack efficiency, it resulted in unbalanced gameplay. Defensive base design became less effective, and strategic wall placement lost its importance. This result explains why Clash of Clans intentionally avoids pure shortest-path algorithms in favor of constrained, balance-oriented AI behavior.

6. Conclusion

This study analyzed troop pathfinding and target selection algorithms inspired by Clash of Clans–style war scenarios, focusing on balance-oriented AI design rather than optimal pathfinding. By modeling observable troop behavior and conducting controlled experiments, the research showed that base layout, wall complexity, and target priority significantly influence attack outcomes.

The results demonstrate that intentionally constrained AI behavior enhances strategic gameplay by allowing players to influence troop movement through base design. Comparisons with pure shortest-path navigation further revealed that optimal pathfinding can lead to unbalanced gameplay and reduce defensive relevance. Overall, this work supports the view that effective game AI prioritizes player experience and balance over mathematical optimality and provides a practical framework for analyzing AI behavior in commercial strategy games.

References

1. Millington, I.; Funge, J. *Artificial Intelligence for Games*, 2 ed.; CRC Press: Boca Raton, FL, USA, 2009.
2. Clash of Clans Wiki: Game Mechanics Overview. https://clashofclans.fandom.com. Accessed: 2026-01-18.
3. Yannakakis, G.N.; Togelius, J. *Artificial Intelligence and Games*; Springer: Cham, Switzerland, 2018.
4. Hart, P.E.; Nilsson, N.J.; Raphael, B. A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Transactions on Systems Science and Cybernetics* **1968**, *SSC-4*, 100–107.

 doi:10.20944/preprints202601.2386.v1

5. Holmgård, C.; Togelius, J.; Yannakakis, G.N. Game Balance and Player Experience. In Proceedings of the Proceedings of the IEEE Conference on Computational Intelligence and Games (CIG), 2014, pp. 1–8.
6. Supercell Developers. Clash of Clans AI Design. YouTube. Accessed: 2026-01-18.
7. Designing AI for Strategy Games. https://www.gdcvault.com. Accessed: 2026-01-18.
8. Laird, J.E.; van Lent, M. Human-Level AI's Killer Application. *AI Magazine* **2001**, *22*, 15–26.
9. Russell, S.; Norvig, P. *Artificial Intelligence: A Modern Approach*, 3 ed.; Pearson: Upper Saddle River, NJ, USA, 2016.
10. Game Maker's Toolkit. Why Game AI Cheats. YouTube. Accessed: 2026-01-18.