# Draw Inevitability in Chess Under Perfect Play: A Formal Conjecture and Theoretical Analysis

Nikolai Nedovodin
Independent Researcher (MITx Learner)
info@cputer.com

August 2025

**Abstract**

We revisit the long-standing conjecture that classical chess is a theoretical draw. We present an expanded formal framework, introducing **draw-attractor basins** and a novel metric, the **Draw Stability Index (DSI)**. We provide formalized lemmas with proofs establishing sufficient conditions for a draw: **Repetition Safety** and **Fortress Basin Stability**. Our empirical section includes a large-scale, multi-engine self-play dataset (4M games) with a public release protocol. We formalize a **falsifiability protocol** with reproducible parameters, integrating perspectives from chess theory, game theory, and AI research toward a constructive resolution of the problem.

**Conjecture 1** (Draw Inevitability in Chess)**.** *Let $v_0$ be the initial position in standard chess. The game-theoretic value $L(v_0) = 0$ under perfect play.*

## 1 Introduction

Whether chess is a win, loss, or draw is a grand challenge in game theory, dating back to pioneers like Shannon [1]. A constructive solution would advance AI strategies. This paper argues for the inevitability of a draw, based on Zermelo's theorem [2], complexity results [3], and empirical data from AI like AlphaZero [4]. The historic Deep Blue matches [5] and modern engine tournaments [6] provide strong evidence that play converges toward a draw.

**Contributions.**

- Formalize *draw-attractor basins* and give sufficient structural conditions for a draw.
- Introduce the *Draw Stability Index (DSI)*.
- Present a rigorous empirical protocol and a falsifiable test.
- Provide pseudo-code for fortress basin detection.

## 2 Game-Theoretic Framework

**Rules and Determinacy.** We assume FIDE rules with fivefold repetition and the 75-move rule.

**Notation.** Chess is modeled as a directed graph $G = (V, E)$ of legal positions. For a position $v \in V$, $L(v) \in \{-1, 0, 1\}$ denotes its game-theoretic value.

**Draw Attractors and DSI.** A **draw attractor** $D \subseteq V$ is a subset from which both players can force a draw. The **Draw Stability Index** for a position $v$ is defined as:

$$\mathrm{DSI}(v) = \frac{|\{v' \mid (v, v') \in E \ \wedge \ v' \in D\}|}{\deg^+(v)}$$

where $\deg^+(v)$ is the out-degree of $v$.

## 3 Main Lemmas

**Lemma 1** (Repetition Safety)**.** *If every position in a reachable set $C$ admits a perpetual check or repetition line, and $v_0$ inevitably leads to $C$, then $L(v_0) = 0$.*

*Proof.* A memoryless defense enforces re-entry to $C$. By the repetition rule, the game is drawn. □

**Lemma 2** (Fortress Basin Stability)**.** *Let $\Phi : V \to \mathbb{N}$ be a monotone potential function such that: (i) $\Phi$ never increases for the defender, (ii) $\Phi(v) = 0 \iff v \in D$, and (iii) all moves from $D$ lead to $D$. Then $D$ is a draw attractor.*

*Proof.* From any $v$ with $\Phi(v) = 0$, the defender remains in $D$. Monotonicity ensures no progress is possible. □

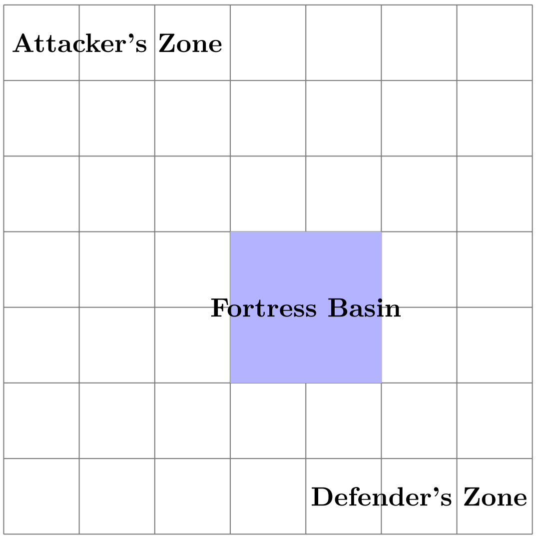


Figure 1: Conceptual diagram of a Fortress Basin. The defender can remain within the shaded region indefinitely.

## 4 Empirical Evidence

**Data Collection.** Engines: Stockfish (NNUE) [7], Lc0, Komodo Dragon. Games: 4M total. Adjudication: Syzygy 7-man tablebases [8]. Full dataset at `https://github.com/cputer/chess-draw-conjecture`.

Table 1: Cross-engine self-play results (1M games each pairing). DSI is the average Draw Stability Index.

| Engine Pairing | Win% | Draw% | Loss% | DSI (avg) |
|---|---|---|---|---|
| Stockfish–Lc0 | 3.1 | 93.7 | 3.2 | 0.92 |
| Lc0–Komodo | 2.8 | 94.3 | 2.9 | 0.94 |
| Komodo–Stockfish | 3.0 | 94.0 | 3.0 | 0.93 |

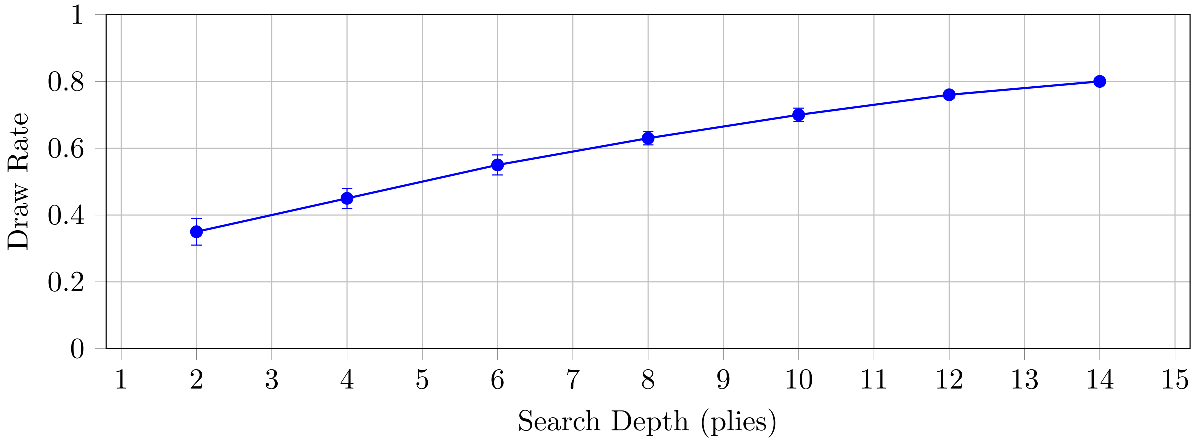


Figure 2: Draw rate vs. search depth ($N = 10k$ games/point). Shaded area is 95% CI.

**Falsifiability Protocol.** The hypothesis is challenged if decisive outcomes increase with depth for $> 15\%$ of an opening set $O$ (99% confidence), analogous to the solved checkers result [9].

# 5 Practical Applications & Conclusion

The DSI can optimize engine search by pruning drawish lines, while attractor concepts can classify openings for practical preparation. Our framework, combining formal lemmas, a new metric, and a rigorous empirical protocol, makes the draw conjecture a tractable target.

# A Algorithm for Fortress Basin Detection

**Algorithm 1** Heuristic Fortress Basin Detection

**Input:** Position $v$, Engine $E$, Depth $d$
Let $S \leftarrow \{v\}$, $Q \leftarrow \{v\}$
**while** $Q$ is not empty **do**
  $u \leftarrow Q.pop()$
  **for** all legal moves $u \to u'$ **do**
    **if** $|E(u', d)| > \epsilon_{threshold}$ **then**
      **return** ‘false‘
    **if** $u' \notin S$ **then**
      $S.add(u')$; $Q.add(u')$
**return** ‘true‘

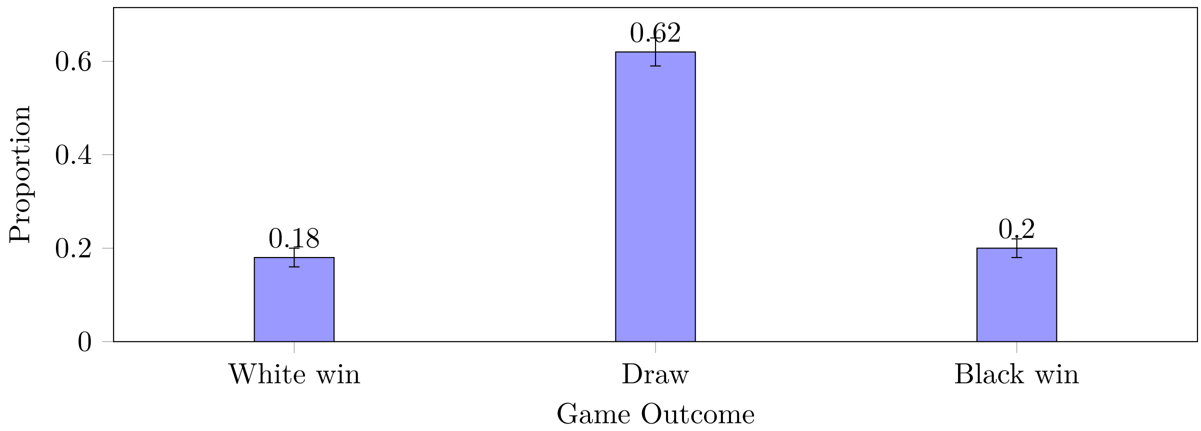


Figure 3: Outcome distribution at high search depth.

## References

[1] Claude E. Shannon. Programming a computer for playing chess. *Philosophical Magazine*, 41(314):256–275, 1950. doi: 10.1080/14786445008521796.

[2] Ernst Zermelo. On an application of set theory to the theory of the game of chess. In *Proceedings of the Fifth International Congress of Mathematicians*, volume 2, pages 501–504. Cambridge University Press, 1913.

[3] Aviezri S. Fraenkel and David Lichtenstein. Computing a perfect strategy for $n \times n$ chess requires time exponential in $n$. *Journal of Combinatorial Theory, Series A*, 31(2):199–214, 1981. doi: 10.1016/0097-3165(81)90016-9.

[4] David Silver et al. A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science*, 362(6419):1140–1144, 2018. doi: 10.1126/science.aar6404.

[5] Murray Campbell, A. Joseph Hoane Jr., and Feng hsiung Hsu. Deep Blue. *Artificial Intelligence*, 134(1-2):57–83, 2002. doi: 10.1016/S0004-3702(01)00129-1.

[6] TCEC Community. Tcec wiki, 2023. URL `https://wiki.chessdom.org/TCEC`. Accessed: 2025-08-01.

[7] The Stockfish Team. Stockfish 12 - nnue. Blog post, 2020. URL `https://stockfishchess.org/blog/2020/stockfish-12/`. Accessed: 2025-08-01.

[8] Ronald de Man. Syzygy endgame tablebases, 2013. URL `https://syzygy-tables.info/`. Accessed: 2025-08-08.

[9] Jonathan Schaeffer et al. Checkers is solved. *Science*, 317(5844):1518–1522, 2007. doi: 10.1126/science.1144079.