ChessLab: Evaluating the Wisdom of Artificial Crowd in Chess

Mario Larsen

January 16, 2026

Abstract

Wisdom of crowds is a phenomenon stating that aggregated decisions of diverse, independent agents often outperform individual experts. However, crowds of casual chess players still struggle against strong opponents. Recent projects tend to show that the method by which a move is chosen by the crowd can significantly influence their performance. In this work, we introduce **ChessLab**, an open-source framework for studying artificial collective intelligence in sequential decision-making through chess. By replacing human crowds with heterogeneous ensembles of deterministic and learning-based chess engines, we began investigating questions such as: (1) Can voting ensembles of weak agents defeat stronger opponents? (2) How does diversity composition affect collective performance? (3) What voting mechanisms optimally aggregate engine decisions? This paper presents the framework, outlines our experimental methodology, and provides a comprehensive research roadmap for studies to be conducted during the next phase of this project.

Keywords— chess engines, wisdom of crowds

1 Introduction

Collective intelligence can in some circumstances exceed individual capability. One of the earliest demonstrations comes from Galton's famous ox-weighing experiment, where the median guess of approximately 787 individuals in a county fair was more accurate than expert estimators and nearly matched the actual weight of 1,197 pounds **?**.

This phenomenon has been tested in a variety of settings, such as prediction markets, where Atanasov et al. compared prediction markets with prediction polls in a large geopolitical forecasting tournament and found that carefully aggregated team forecasts could outperform market prices **?**. Question-answering game shows also provide natural experiments. In "Who Wants to Be a Millionaire", the "Ask the Audience" lifeline yields correct answers around 87-92% of the time, whereas phone-a-friend experts succeed only about 55-65% of the time, across many national editions of the show **?**. However, in competitive games like chess, crowds have historically struggled against strong opponents.

The 1999 Kasparov vs. The World match saw more than 50,000 human players collectively challenge the World Chess Champion Garry Kasparov **?**, reaching move 62 before conceding. More recently, in May 2025, over 143,000 players held World Champion Magnus Carlsen to a draw in a record-breaking game on Chess.com **?**, and in November 2025, over 200,000 players loose in a match against International Master Levy Rozman (GothamChess) **?**. However, these matches remain largely anecdotal: a single game provides minimal evidence for systematic collective intelligence, as outcome variance is high and learning effects are confounded.

A Fouloscopie project, led by Mehdi Moussaïd at the Max Planck Institute for Human Development, addressed this limitation by conducting the first rigorous, large-scale experiment on collective chess intelligence **?**. Approximately 25,000 participants were split into teams to play a total of 500 chess games against AI opponents. Using majority-vote aggregation of player moves, Moussaïd demonstrated that crowds achieved a performance level similar to a master while being composed predominantly of beginners and casual players. Importantly, by controlling for poll visibility, the results showed a significant effect of social influence on the crowd's level.

Yet even this landmark study leaves open critical questions:

- **Mechanism understanding**: What aspects of diversity (in skill, size, architecture) drive the ensemble advantage?
- **Optimization**: Can we identify decision-making processes that outperform simple majority aggregation?
- **Domain generality**: Does collective intelligence in chess rely on human-specific cognitive mechanisms, or do the principles apply to artificial agents?

We address these questions through **ChessLab**, an open-source framework for studying artificial collective intelligence in chess. By substituting human crowds with heterogeneous ensembles of chess engines, we gain precise control over experimental conditions, perfect reproducibility, and the ability to run experiments rapidly with minimal computational cost.

The main contributions of this paper are:

1. **Framework and methodology**: A modular, extensible system for running ensemble chess experiments at scale, with persistent storage of games, moves, evaluations, and statistical analyses.

2. **Theoretical grounding**: Clear hypotheses derived from Scott Page's Diversity Prediction Theorem and classical wisdom-of-crowds theory, adapted to the chess domain.

3. **Reproducible baselines**: Quantitative comparison against Fouloscopie human results, enabling assessment of whether artificial and human collective intelligence follow similar principles.

4. **Comprehensive research roadmap**: A structured plan for experiments addressing collective intelligence across different engine types, ensemble compositions, and voting strategies.

This paper focuses on framework description, experimental design, and hypothesis articulation. Results from additional experiments will be presented in subsequent publications as they are completed.

2 Related Work

2.1 Collective Intelligence and Wisdom of Crowds

The wisdom of crowds phenomenon emerged from early empirical observations: Galton's ox-weighing experiment demonstrated that collective judgments (median of 787 guesses) were more accurate than nearly all individual estimates **?**. Surowiecki systematized this observation, arguing that under certain conditions (diversity of opinion, independence of members, decentralization, and effective aggregation) groups reliably outperform experts **?**.

Formally, Scott Page's *Diversity Prediction Theorem* provides mathematical grounding:

$$\text{Collective Error} = \overline{\text{Individual Error}} - \text{Diversity} \tag{1}$$

where collective error is the mean squared error of the group prediction, $\overline{\text{Individual Error}}$ is the average of individual squared errors, and Diversity captures how different agents' predictions are **?**. This result implies that a team of common individuals can outperform a specialist, provided the group is large enough and exhibits sufficient diversity in cognitive approach.

However, diversity alone is insufficient. Studies show that social influence, information cascades, and herding can destroy the diversity necessary for wisdom of crowds. Almaatouq et al. demonstrated that agents strategically choosing whom to learn from (adaptive networks) better maintain collective intelligence than static networks **?**. Koriat and colleagues showed that *confidence* acts as a metacognitive signal: when individuals can estimate how well-calibrated their judgment is, confidence-weighted aggregation outperforms simple majority voting **?**, except when facing common misconceptions that result in misplaced confidence. Prelec extended this with the "surprisingly popular" voting method, which selects answers that are more popular than subjects predicted, revealing hidden consensus even when the majority is wrong **?**.

In summary, collective intelligence requires: (1) a sufficient number of participants, (2) diversity in decision-making, (3) independence of judgments, (4) appropriate aggregation mechanisms, and possibly (5) calibration of confidence when weighting is available.

[TODO add a passagge for this this article * **[Making the wisdom of crowds efficient](https://osf.io/preprints/psyarxiv/j9t6p$_v$1) $* * * * * Title : * * Makingthewisdomofcrowdsefficient^{-}withconfidence * * * Summary : * * Apreprintpaperdiscussingmethodstoimprovethe"wisdomofcrowds"b$

2.2 Collective Intelligence in Chess

Chess provides an ideal laboratory for studying collective decision-making in complex, sequential tasks. Unlike static judgment tasks (trivia answering, estimation), chess involves:

- **High-dimensional state space**: Approximately 10^{47} legal positions, making expert heuristics difficult and providing an extensive test dataset.
- **Strategic thinking**: Each player must evaluate not just the next move but plan for responses and counter-responses. This task allows deliberation to express emerging strategies.
- **Objective evaluation**: The outcome of a game is unambiguous, and thanks to modern engines, each move can be evaluated with precise numerical scoring. The performance level of a crowd can be estimated with arbitrary precision.
- **Cultural significance**: Chess remains a synecdoche for intelligence. Both humans and algorithms are expected to demonstrate competence at this task, providing abundant documentation and research literacy.

To evaluate the level of a chess player the comunity mainly rely on a number called the ELO. Given two players with Elo ratings Elo_A and Elo_B, the expected score (probability of A winning) is:

$$E_A = \frac{1}{1 + 10^{(\text{Elo}_B - \text{Elo}_A)/400}} \tag{2}$$

This is the standard Elo formula, calibrated for chess **?**.

While this metric captures relative strength between two players, its absolute interpretation depends heavily on methodology and context. Community conventions suggest that players understanding the basic rules rate between 600 and 900 Elo, casual players typically orbit around 1300 Elo, and from 2000 Elo onward, players enter the master and grandmaster territory. At his peak in May 2014, Magnus Carlsen achieved a classical rating of 2882, the highest in history. He also achieved 2909 in the newly-introduced Freestyle Chess (Chess960) rating system ?.

2.2.1 Empirical Studies of Crowd Chess

The Kasparov vs. The World match (1999) is the earliest documented crowdsourced chess challenge. Over four months, approximately 50,000 amateur and intermediate players voted on moves against the then-World Champion. Despite reaching an advanced position, the crowd eventually conceded after 62 moves when facing maximum resistance from Kasparov ?. While historically significant, this single-game format provides weak evidence: a single outcome cannot be reliably attributed to collective intelligence versus chance.

The Fouloscopie experiment rectified this limitation through large-scale repetition ?. Moussaïd and colleagues organized approximately 25,000 human participants to play a total of 500 games against chess engine opponents. Key design features included:

- **Diverse player population**: Participants self-reported their Elo rating; the population was approximately normally distributed with a mean around 1165 Elo.
- **Majority-vote aggregation**: Players had several hours to evaluate positions and vote on the next move; the most-voted move was played automatically.
- **Opponent variety**: Chess engine opponents (Maia models) ranged from weak (1100 Elo) to strong (1900 Elo), allowing measurement of performance across the strength spectrum.
- **Social influence manipulation**: Comparing voting-only versus poll-visibility conditions revealed how information visibility affects collective decision quality.

Results showed that crowds achieved a win rate exceeding 60% across all tested opponents. With poll visibility, win rates improved to approximately 77%, while visibility-hidden conditions showed approximately 64% win rates. Importantly, the crowd played measurably better than the average individual member, confirming classic wisdom-of-crowds predictions.

2.3 Chess Engines

Modern chess engines span multiple paradigms, each with distinct decision-making processes. We focus on three particularly relevant approaches:

2.3.1 Stockfish

Stockfish, as of 2025, is the world's strongest chess engine. It is an open-source tree-search algorithm combining classical techniques (alpha-beta pruning, endgame tables) with small neural networks for position evaluation ?. Its strength is calibrated via Elo ratings: users can set `UCI_Elo` to any value from 1320 to 3190. The engine deliberately weakens its play through stochastic evaluation perturbation (seeded random noise), ensuring exploration of different move trees at reduced strength levels.

2.3.2 Maia: Human-Mimetic Neural Networks

In contrast to optimal play, Maia learns to *predict human moves* from millions of Lichess games ?. Rather than computing optimal moves, Maia generates a probability distribution over legal moves, parameterized by player skill level. Maia-2, introduced at NeurIPS 2024, incorporates a skill-aware attention mechanism that dynamically integrates player skill level with board position encoding ?. This mechanism achieves approximately 50–52% top-1 move accuracy across human players rated 1100 to 1900 Elo. In practical play, Maia replicates human decision-making nearly indistinguishably, making it an excellent tool for evaluating crowd performance against human-like opponents.

2.3.3 Large Language Models

Recent work demonstrates that Large Language Models (LLMs) trained on chess game transcriptions can develop emergent internal representations of board state ?. This representation can be altered afecting the move predicted by the LLM. This represent a fundamentally different paradigm. They make decisions based on pattern recognition from training data, not explicit search or neural evaluation. Move quality is unstable and depends on a miriad of parameters (model, prompt with PGN notation, board representation, move history) that can be hard to replicate or interpret.

[TODO add paragraph about https://arxiv.org/pdf/2306.09983]

Carlini showed that GPT-3.5-turbo-instruct can achieve a playing strength of roughly 1,788 Elo under specific conditions ?. Dynomight systematically evaluated multiple LLM variants, finding that this success can partialy be replicated in newer chat orriented model given specific prompts ? [TODO add citation https://dynomight.net/more-chess/]. More recent transformer-based models have achieved grandmaster-level performance when trained directly on game transcriptions ?.

For ensemble experiments, LLMs offer the highest cognitive diversity and potential for open deliberation, but also present the most challenging engineering requirements (API management, prompt optimization, legal move filtering).

2.4 Aggregating Engines

Carvalho et al. studied majority voting among homogeneous groups of checkers engines. They found that group performance improves roughly logarithmically with group size, while outcome variance decreases **?**.

[TODO add a section with * **[Majority Voting in Chess (Scholar)](https://link.springer.com/chapter/10.1007/978-3-319-09165-5$_1$7) $* * * * * Title : * * Furtherinvestigationsof3 - membersimplemajorityvotingforchess * * * Summary : * * Researchinvestigatingwhetherasimplemajorityvoteofthreechessenginesprovidesbettermoveselectionthanasingleengine, touch$

3 Methodology and Framework: ChessLab

[Section to be completed in next revision]

3.1 System Architecture

ChessLab is a modular, open-source framework designed for large-scale chess engine evaluation. Key components:

3.1.1 Engine Abstraction Layer

A common interface (`BaseEngine`) supports multiple engine types:

- **Stockfish**: UCI protocol with Elo-limiting via `UCI_LimitStrength` and `UCI_Elo` options.
- **Maia**: Custom interface wrapping the Maia model; configurable skill level (1100–1900 Elo).
- **Random**: Legal move generator for baseline comparisons.
- **LLM**: HuggingFace integration with customizable prompts, legal-move filtering, and sampling strategies.
- **Voting ensembles**: Meta-engine combining other engines via various voting rules.

3.1.2 Game Execution Engine

Asynchronous game runner that:

- Manages chess state (FEN notation, legal moves, draws-by-repetition).
- Communicates with engines via UCI protocol or custom APIs.
- Records each move, response time, and engine evaluation.
- Handles parallel execution (configurable concurrency limit).
- Stores complete game records in PostgreSQL.

3.1.3 Persistent Storage

PostgreSQL schema captures:

- **Players**: Engine type, name, Elo rating, creation timestamp, configuration parameters.
- **Games**: White/Black player IDs, result (1-0, 0-1, 1/2-1/2), PGN, opening, phase metadata.
- **Moves**: Sequence number, SAN/UCI notation, FEN before move, evaluation, move quality (blunder/bad/dubious/ok/good/best).
- **Evaluations**: Depth-annotated Stockfish evaluation for positions; used in post-hoc move-quality analysis.
- **Requests**: Batch queue for asynchronous LLM inference.

3.1.4 Statistical Analysis Module

Post-game analysis computes:

- Elo estimation using the methods described in Section **??**.
- Win rate and draw rate across opponent strengths.
- Confidence intervals via bootstrap resampling.
- Move accuracy (correlation between engine move and best move).
- Significance testing (two-sample t-test, Fisher's exact test for win rates).

3.2 Elo Rating Estimation

To compare ensembles against single engines and to Fouloscopie's human crowd, we estimate Elo ratings using the methods provided:

3.2.1 Single-Opponent Estimation

If an ensemble plays n games against a single opponent of known Elo, and achieves mean score s (where $1 =$ win, $0.5 =$ draw, $0 =$ loss), the ensemble Elo is:

$$\text{Elo}_{\text{ensemble}} = \text{Elo}_{\text{opponent}} + 400 \log_{10}\left(\frac{s}{1-s}\right) \tag{3}$$

This formula is derived by inverting Equation **??**, ensuring that the expected score at the estimated Elo matches the observed score.

3.2.2 Multiple-Opponent Estimation

If an ensemble plays against multiple opponents at different Elos, we fit a single Elo value that minimizes weighted squared error between observed and expected scores:

$$\hat{\text{Elo}}_{\text{ensemble}} = \arg\min_{\text{Elo}} \sum_{i=1}^{m} w_i \left(s_i - E(Elo, \text{Elo}_i)\right)^2 \quad (4)$$

where s_i is the mean score against opponent i, Elo_i is opponent i's rating, and w_i is the weight (typically the number of games against opponent i, normalized). This optimization is solved via bounded scalar minimization ($250 \leq \text{Elo} \leq 3000$) with precision 0.1 Elo points.

This approach is robust to mild non-transitivity in the results (e.g., A beats B, B beats C, but C sometimes beats A), and the fitting naturally produces confidence intervals via bootstrap resampling of game outcomes.

3.3 Experimental Design Principles

All experiments follow these principles:

3.3.1 Fair Time Allocation

Each engine in an ensemble receives equal wall-clock time per move. For Stockfish ensembles, if total time is 5 seconds per move and there are 5 engines, each receives 1 second. This ensures that the ensemble advantage comes from diversity, not resource allocation.

3.3.2 Deterministic Reproducibility

All random seeds are fixed and recorded. Stockfish's `UCI_Threads` is set to 1 (single thread) to ensure reproducibility across machines. Game variations are introduced through algorithm randomization (Stockfish's evaluation perturbation, Maia's sampling), not external noise.

3.3.3 Statistical Power

Experiments use sufficient games to detect meaningful effects (power = 0.8, $\alpha = 0.05$). Based on typical chess game outcomes:

- Comparing single-engine vs. N-engine ensemble: $n = 50$ games.
- Comparing two ensembles: $n = 100$ games.
- Measuring performance across strength range: $n = 20$ games per opponent level, ≈ 200 games total.

3.3.4 Opponent Strength Variation

Opponents span at least Elo 1200–2200 (amateur to master strength), with intermediate steps at ≈ 200 Elo increments. This enables fitting performance curves and testing predictions from collective intelligence theory.

4 Experiments

[Section to be completed in next revision]

This section outlines 10 key experiments, prioritized by importance and feasibility. Results will be presented in follow-up publications as experiments complete. For each experiment, we specify:

- **Hypothesis**: The theoretical prediction.
- **Design**: Independent and dependent variables.
- **Metrics**: Primary outcome measures.
- **Analysis**: Planned statistical tests.

4.1 Experiment 1: Homogeneous Stockfish Ensembles (Proof of Concept)

4.1.1 Motivation

Simple baseline establishing that ensemble voting provides measurable advantage over individual engines, even when engines are identical. This tests the most basic prediction: variance reduction through aggregation.

4.1.2 Design

- **Independent variable**: Ensemble size $N \in \{1, 3, 5, 10, 20\}$.
- **Engine configuration**: All Stockfish at Elo 1600 (intermediate strength), UCI_Elo limiting enabled.
- **Opponents**: Stockfish at Elo 1400, 1600, 1800 (opponent weaker, equal, stronger).
- **Games per condition**: 50 games (sufficient to detect win-rate difference of $\approx 5\%$, $p < 0.05$).

4.1.3 Hypothesis

Ensemble win rate will increase with ensemble size, following a curve of diminishing returns. Mathematically, if individual engines have variance σ^2 and uncorrelated errors, the ensemble variance should decrease as σ^2/N. This predicts:

- $N = 1$: baseline win rate (e.g., 45%).
- $N = 3$: win rate increase $\propto \sqrt{3} \approx 1.7$ (error reduction), e.g., 48% win rate.
- $N \to \infty$: plateau as opponents' skill becomes limiting factor.

4.1.4 Metrics

- Win rate (percentage of wins) and 95% confidence interval.
- Estimated ensemble Elo via multiple-opponent fitting.
- Move selection entropy (how often does ensemble choose the modal move vs. split votes).

4.1.5 Analysis

Logistic regression of win rate vs. ensemble size. Fitting a power law: $\mathrm{WinRate}(N) = a + b \log(N)$ or $\mathrm{WinRate}(N) = c - d/N$. Bootstrap confidence intervals on Elo estimates.

4.2 Experiment 2: Heterogeneous Elo Ensembles (Testing Page's Diversity Theorem)

4.2.1 Motivation

Test whether diversity in agent capability (not just replication) improves ensemble performance. Page's theorem predicts that cognitive diversity can outweigh individual competence; an ensemble of weak agents with diverse skills should outperform weaker homogeneous groups.

4.2.2 Design

- **Independent variables**:
  - Ensemble composition: $\{W, M, S\}$ representing weak (1400 Elo), medium (1600 Elo), and strong (1800 Elo) Stockfish.
  - Ratios tested: 100% W, 100% M, 100% S (homogeneous baselines); 50% W + 50% S, 75% W + 25% S, 33% W + 33% M + 33% S (heterogeneous).
- **Ensemble size**: Fixed at 5 engines per ensemble.
- **Opponents**: Fixed at Elo 1600 (to test scaling).
- **Games**: 100 games per condition.

4.2.3 Hypothesis

Ensemble performance follows an inverse-U relationship with diversity:

- Pure weak ensemble (100% 1400): Low performance (e.g., 25% win rate).
- Pure strong ensemble (100% 1800): High performance (e.g., 75% win rate).
- Optimally diverse ensemble (predicted $\approx$ 30–40% weak agents): Superior to homogeneous weak but inferior to homogeneous strong. However, the diversity bonus may make heterogeneous weak ensemble ($\approx$ 40% weak, 60% strong) outperform pure weak or weak-medium blends.

4.2.4 Metrics

- Win rate for each composition.
- Estimated ensemble Elo.
- Move diversity: entropy of vote distribution (e.g., how often is the vote split vs. unanimous).
- Error correlation: Do weak and strong engines make uncorrelated mistakes?

4.2.5 Analysis

Logistic regression with interaction terms for composition. Formal test of Page's theorem: test whether "diversity bonus" (predicted performance minus average individual performance) is positive and significant.

4.3 Experiment 3: Maia Ensemble (Human-Mimetic Diversity)

4.3.1 Motivation

Test whether diversity in *learned* decision-making (Maia) produces similar benefits to diversity in explicit Elo. Maia provides an alternative engine architecture and learned human-like heuristics.

4.3.2 Design

- **Independent variable**: Maia skill-level composition.
  - Homogeneous: 5x Maia-1600, 5x Maia-1400, etc.
  - Heterogeneous: Mix of Maia-1400, Maia-1600, Maia-1800.
- **Opponents**: Maia at Elo 1400, 1600, 1800 (to ensure fair comparison—playing against own architecture).
- **Games**: 100 per condition.

4.3.3 Hypothesis

Maia ensembles should exhibit similar diversity effects to Stockfish:

- Heterogeneous Maia ensemble outperforms homogeneous Maia ensemble of equivalent average strength.
- The diversity bonus magnitude differs from Stockfish (due to different decision-making architecture), but the principle holds.

4.3.4 Metrics

- Win rate and Elo estimates.
- Move variety: Do Maia models at different levels vote differently?
- Prediction accuracy: Correlation between Maia move choice and human-level move distribution (via Lichess data).

4.3.5 Analysis

Comparison of heterogeneous vs. homogeneous win rates (two-sample t-test). Estimation of diversity bonus in Elo points.

4.4 Experiment 4: Voting Strategy Comparison

4.4.1 Motivation

Optimize the aggregation rule. Does weighted voting outperform simple majority? Does surprisingly-popular voting provide theoretical improvement?

4.4.2 Design

- **Independent variable**: Voting rule.
  - Simple majority (baseline).
  - Weighted by Elo: Each engine vote counts for $10^{(\text{Elo}-1600)/400}$.
  - Weighted by evaluation score: Each engine vote counts for $\max(0, \text{evaluation\_score}/5)$ (normalized).
  - Surprisingly popular (LLM/future): Prediction-aware voting.
- **Ensemble**: Fixed heterogeneous Stockfish ensemble (1400, 1600, 1800, repeated).
- **Opponents**: Stockfish 1600, 1800.
- **Games**: 100 per condition.

4.4.3 Hypothesis

- Simple majority: Baseline ($W_0 = 50\%$).
- Weighted Elo: Moderate improvement ($W_E \approx 52\%$) from reducing weak-engine influence.
- Weighted evaluation: Small improvement ($W_V \approx 51\%$) but risk of overfitting to position score.
- Surprisingly popular: Potential improvement ($W_S \approx 53\%$) if engines are overconfident or diverse in confidence.

4.4.4 Metrics

- Win rate for each voting rule.
- Move agreement: For each position, what fraction of ensemble members voted for the selected move?
- Robustness: Does voting rule perform consistently across opponent strengths?

4.4.5 Analysis

Pairwise comparisons (McNemar's test for win-rate differences). ANOVA for multi-way comparison.

4.5 Experiment 5: Cross-Architecture Ensemble (Maximum Diversity)

4.5.1 Motivation

Test ensembles combining fundamentally different architectures (search-based, neural learning-based, heuristic). This maximizes cognitive diversity in decision-making approach.

4.5.2 Design

- **Ensemble composition**: Stockfish-1600 + Maia-1600 + simple-heuristic-1600 (matched Elo when possible).
- **Ensemble variants**:
  - 5 engines: 2 Stockfish, 2 Maia, 1 heuristic.
  - 3 engines: 1 Stockfish, 1 Maia, 1 heuristic (minimal ensemble).
- **Opponents**: Stockfish 1400, 1600, 1800.
- **Games**: 100 per condition.

4.5.3 Hypothesis

Cross-architecture ensembles achieve higher Elo than same-architecture ensembles of equivalent average strength, due to uncorrelated decision-making errors.

4.5.4 Metrics

- Win rate and Elo.
- Move agreement analysis: How often do different architectures propose identical moves vs. different moves?
- Error correlation: Compute correlation of move quality rankings between architecture pairs.

4.5.5 Analysis

Comparison to single-best-engine baseline (Stockfish-1600) via two-sample t-test. Elo difference estimation.

4.6 Experiment 6: Fouloscopie Reproduction (Main Contribution)

4.6.1 Motivation

Directly reproduce the human Fouloscopie results using artificial ensembles. This is the keystone experiment: if artificial crowds match human crowd Elo, it suggests collective intelligence principles are domain-independent. If they diverge, it reveals how human and artificial cognition differ.

4.6.2 Design

- **Crowd composition**: Heterogeneous Stockfish ensemble (sizes: 10, 30, 50 agents), with Elo distribution spanning 1200–2000 (mimicking human skill variation in Fouloscopie).
- **Opponents**: Maia models at Elo 1200, 1400, 1600, 1800, 1900 (spanning weak to very strong).
- **Games**: 20 games per (crowd size, opponent) combination, total ≈ 300 games.
- **Voting rule**: Simple majority (matching Fouloscopie).

4.6.3 Hypothesis

Artificial crowds achieve comparable Elo to human crowds when ensemble size and diversity match. Specifically:

- Human Fouloscopie crowd: Estimated Elo ≈ 2200 (to be verified from dataset) with win rate $> 60\%$ across all opponent levels.
- Artificial crowd (50 Stockfish agents, mixed Elo): Estimated Elo ≥ 2000 (90% of human level) across all opponent strengths.
- The ensemble maintains stable performance across opponent range, suggesting robust collective intelligence.

4.6.4 Metrics

- Win rate curve across opponent strength.
- Estimated crowd Elo via multi-opponent fitting.
- Confidence intervals: Bootstrap 95% CI on Elo estimate.
- Comparison to human Fouloscopie: Relative Elo difference.

4.6.5 Analysis

Curve fitting: Logistic model of win rate vs. opponent Elo. Elo estimation via bounded optimization (described in Section **??**). Hypothesis test: Is artificial crowd Elo statistically indistinguishable from human crowd Elo? (Two-sample t-test on Elo estimates, with resampling).

4.7 Experiment 7: Social Influence Analog (Policy Transparency)

4.7.1 Motivation

Fouloscopie found that visible polling (showing vote counts before final voting) amplified collective intelligence. We test an analog: ensemble plays with/without revealing move-frequency counts during the voting phase.

4.7.2 Design

- **Independent variable**: Information visibility.
  - `Hidden`: Engines vote sequentially; no engine sees others' votes before committing.
  - `Visible`: After each engine commits, the cumulative vote count is revealed to remaining engines; they can adjust their strategy.
- **Ensemble**: 5 Stockfish agents at Elo 1600.
- **Opponents**: Stockfish at 1400, 1600, 1800.
- **Games**: 100 per condition.

4.7.3 Hypothesis

- Against weak opponents: Visibility increases win rate (engines align on safe, strong moves).
- Against strong opponents: Visibility decreases win rate (engines converge on popular but suboptimal moves, losing diversity).

4.7.4 Metrics

- Win rate by opponent strength.
- Vote concentration: Entropy of move distribution (do engines converge or diversify?).
- Move quality: Do visible-voting ensembles choose objectively worse moves (lower evaluation) in tough positions?

4.7.5 Analysis

Two-way ANOVA (visibility × opponent strength) on win rate. Interaction plot: Does visibility effect reverse with opponent strength?

4.8 Experiment 8: Scaling Laws (Crowd Size)

4.8.1 Motivation

Characterize performance as a function of ensemble size. Are there diminishing returns? Can we fit a power law?

4.8.2 Design

- **Independent variable**: Ensemble size $N \in \{1, 3, 5, 10, 20, 50, 100\}$.
- **Ensemble**: Heterogeneous Stockfish (representative mix of Elo 1200–2000).
- **Opponents**: Fixed at Elo 1600 (pivot point).
- **Games**: 50 per size.

4.8.3 Hypothesis

Performance scales logarithmically or as a power law:

$$\text{Elo}(N) = \text{Elo}(1) + c\log(N) + \epsilon \tag{5}$$

or

$$\text{Elo}(N) = a - bN^{-\alpha} \tag{6}$$

with diminishing returns (slowing improvement at large N).

4.8.4 Metrics

- Estimated Elo vs. ensemble size.
- Goodness-of-fit: R^2 for power law and logarithmic models.

4.8.5 Analysis

Non-linear least-squares fitting of both models. AIC/BIC model comparison. Extrapolation: At what size do we expect $\approx 90\%$ of asymptotic performance gain?

4.9 Experiment 9: LLM Ensemble (Orthogonal Diversity, Future Work)

4.9.1 Motivation

Large Language Models represent a distinct paradigm. Testing them in ensembles addresses questions about whether collective intelligence principles apply beyond traditional game-playing systems.

4.9.2 Design

- **Engines**: HuggingFace LLM models (Llama-2, Mistral, etc.) with legal-move filtering and sampling.
- **Ensemble**: 5 LLM instances (possibly different models) with shared prompt template.
- **Prompts**: Systematic variation of prompt templates (FEN-only vs. move history vs. candidate moves).
- **Opponents**: Stockfish at 1400, 1600, 1800.
- **Games**: 50 per condition (LLMs slower, so lower sample size initially).

4.9.3 Hypothesis

- Single LLM: Expected Elo $\approx$ 1200 (untrained/weakly fine-tuned models).
- LLM ensemble: Elo $\approx$ 1350 (diversity provides $\approx$ 150 Elo improvement, but still weaker than Stockfish).
- Prompt variation: Providing legal-move lists improves ensemble Elo significantly.

4.9.4 Metrics

- Win rate and Elo.
- Legal-move compliance: % of moves that are legal.
- Move diversity: Do different LLM instances/prompts suggest different moves?

4.9.5 Analysis

Comparison to Stockfish baseline. Ablation: Impact of prompt template on Elo.

4.10 Experiment 10: Time Control Sensitivity (Robustness Check)

4.10.1 Motivation

Verify that results generalize beyond specific time allocations. Elo ratings in chess are calibrated to specific time controls; our experiments must ensure conclusions are robust.

4.10.2 Design

- **Independent variable**: Time control.
  - Blitz: 3+0 (3 seconds, no increment).
  - Rapid: 10+0 (10 seconds).
  - Classical: 30+0 (30 seconds) or depth-limited (10 moves, depth 20).
- **Ensemble**: Fixed 5-engine heterogeneous Stockfish.
- **Opponents**: Stockfish 1600.
- **Games**: 50 per time control.

4.10.3 Hypothesis

Ensemble advantage is robust across time controls: relative Elo difference between ensemble and single engine remains constant (≈ 50–100 Elo) regardless of time control.

4.10.4 Metrics

- Win rate and Elo for each time control.
- Ensemble advantage (Elo gain) constancy.

4.10.5 Analysis

ANOVA for Elo across time controls. Test null hypothesis: Ensemble advantage is independent of time control.

5 Results

[Section to be completed in next revision]

This section outlines expected findings based on theory and preliminary understanding. These are **placeholders** to be replaced with actual experimental results.

5.1 Exp 1: Homogeneous Ensembles

PLACEHOLDER: We expect win rate to increase from $\approx 50\%$ (single engine) to $\approx 52\%$ ($N = 3$), $\approx 53\%$ ($N = 5$), with diminishing returns at larger N. Corresponding Elo gain: $[BASELINE\_ELO] \rightarrow [BASELINE\_ELO + 50] \rightarrow [BASELINE\_ELO + 80]$, approaching asymptote. **Statistical significance**: Ensemble of 5 will be significantly better than single engine ($p < 0.05$).

5.2 Exp 2: Heterogeneous Elo

PLACEHOLDER: Optimal diversity at $\approx$ 30–40% weak agents. Expected Elo:

- 100% weak (1400): $[ELO\_1400]$ (homogeneous baseline).
- 50% weak + 50% strong: $[ELO\_MID] \approx$ Average or slightly above (diversity bonus).
- 33% weak + 33% mid + 33% strong: $[ELO\_OPTIMAL]$ (predicted to be highest).
- 100% strong (1800): $[ELO\_1800]$ (homogeneous strong baseline).

5.3 Exp 6: Fouloscopie Reproduction

PLACEHOLDER: Artificial crowd (50 heterogeneous Stockfish) estimated Elo: [CROWDFISH_ELO_EST] (to compare to human Fouloscopie crowd Elo of [HUMAN_CROWD_ELO] from dataset). Win rate curve: [CURVE_DESCRIPTION]. Confidence interval on Elo: $[ELO\_EST] \pm [MARGIN\_ERROR]$.

6 Discussion and Roadmap

[Section to be completed in next revision]

While this paper focuses on framework and methodology, we can anticipate key insights:

6.1 Implications for Collective Intelligence Theory

If experiments confirm that artificial ensembles replicate human Fouloscopie results, it would suggest:

1. **Domain independence**: Collective intelligence principles (diversity, independence, aggregation) apply across human and artificial agents.
2. **Mechanism transparency**: Using artificial agents, we can precisely isolate which factors drive collective benefit (e.g., variance reduction vs. diversity vs. correction of biases).
3. **Optimizability**: We can systematically test hypotheses about optimal voting rules, ensemble composition, and information structures that would be intractable with human subjects.

6.2 Limitations and Future Work

6.2.1 LLM Integration

Currently marked as future work due to engineering complexity (API rate limits, prompt optimization, legal-move filtering). However, LLMs represent the fastest-moving frontier in AI and deserve careful study.

6.2.2 Opponent Diversity

Current experiments use same-architecture opponents (e.g., Maia ensemble vs. Maia opponents). Future work should include cross-architecture matches (e.g., Stockfish ensemble vs. Maia opponents) to test robustness.

6.2.3 Game Phase Analysis

Preliminary work should analyze whether ensemble advantage varies by game phase (opening, middlegame, endgame), as opening knowledge and endgame technique may distribute differently across agent types.

6.2.4 Surprising Popular Moves

Post-hoc analysis of games should identify moves where the ensemble (or majority) chose a move that was initially surprising (unexpected by others in the ensemble) but turned out to be optimal. This could reveal mechanisms by which diversity prevents groupthink.

6.2.5 Theoretical Extensions

Page's Diversity Theorem assumes independence; in ensemble voting, engines share the same position. Future work should explore models of *conditional* independence given the board state, and how architecture diversity affects this.

7 Conclusion

[Section to be completed in next revision]

This paper introduces ChessLab, a framework for studying collective intelligence in games through systematically engineered ensemble experiments. By replacing human crowds with controlled, diverse AI systems, we gain the ability to:

1. Run hundreds of experiments with perfect reproducibility.

2. Test hypotheses from collective intelligence theory in a complex, sequential domain.

3. Isolate mechanisms (diversity vs. bias correction vs. variance reduction) through controlled manipulation.

4. Compare artificial and human collective intelligence quantitatively (via Elo ratings).

The roadmap of 10 experiments provides a structured progression from simple proof-of-concept (homogeneous ensembles) through sophisticated tests of theory (Page's theorem) and optimization (voting mechanisms) to direct reproduction of human results (Fouloscopie analog).

Results from these experiments will appear in follow-up publications. We anticipate that this work will:

1. **Validate or refine** collective intelligence theory in a new domain.

2. **Provide actionable guidance** on ensemble design for game-playing systems.

3. **Lay groundwork** for studying collective intelligence in other complex domains (trading, scientific collaboration, etc.).

4. **Contribute** to understanding how human and artificial cognition achieve collective goals.

We release ChessLab as open-source software, with the hope that researchers can extend, replicate, and build upon this work.

Disclaimer

The assistance of Large Language Models (LLMs) was used in a limited and controlled manner during the writing process of this paper. All scientific claims, experimental designs, theoretical contributions, and the final version of this paper remain the responsibility of the authors.