The Shadow of the Past: Amortized Inference and Belief Revision using Chess as a Model System

Amarnath Dasaka (amarnath.d@research.iiit.ac.in)
Bapi Raju Surampudi
International Institute of Information Technology, Hyderabad, India

Abstract

How do humans allocate scarce cognitive resources across a stream of related decisions? Russek et al. (2025) predict reaction time on each chess move from the within-move *Benefit of Computation*, treating moves as if they were drawn independently. We ask whether those costs also depend on computation carried forward from the previous move. In chess, a player who has calculated a likely opponent reply can often reuse that work; a surprising reply should force revision. On 135,608 live-game tactical sequences (extracted from the Lichess puzzle database and re-timed against the original 2019 Lichess game database) played by 74,609 unique players, the within-sequence log-RT spike is **+0.309** (Cohen's $d = 0.235$; paired $t = 86$, N = 135,608; $BF_{10} > 10^{100}$). Hierarchical regression climbs from $R^2 = 0.011$ (Russek-static) to **0.282**; the amortization interaction is large when predictability is operationalized by the human-style Maia-2 ($\beta = -0.054$, $p < 10^{-27}$) and not significant under Stockfish best-move predictability. A three-parameter Bayesian Sampling Model with Cache recovers cache strength $s = 0.085$ [0.071, 0.099], with 100% of bootstrap draws positive. The H1 spike is roughly twice as large in solved sequences as in failed ones ($t = 26.2$, $p < 10^{-150}$); this pattern suggests successful tactical recognition followed by decisive recalculation rather than passive surprise. The findings replicate on a 2024 Lichess cohort and on FIDE World Rapid + Blitz 2024 over-the-board play.

Keywords: amortized inference; resource-rational analysis; chess; reaction time; Bayesian sampling; belief revision

Introduction

In a single online chess game a strong player may spend thirty seconds on one move and then dispatch the next three in under five seconds each. Classic models of decision-making attribute this variation to differences in the difficulty of the *current* problem and treat the rest of the trial as if the deliberation history did not matter. Recent work has begun to push back. Russek et al. (2025) showed that the time a human spends per chess move scales with the *Benefit of Computation* (ΔU_C): the value of running deeper search rather than retrieving a shallow heuristic. This is a powerful resource-rational account, but it analyses each move in isolation. Real planning is sequential: when a player has just spent thirty seconds working out the next three plies, the work invested in that plan should carry forward into what follows.

We test the hypothesis that human chess thinking time is dominated by *amortized inference* — the reuse of past computation (Dasgupta et al., 2018, 2020) — and that the cognitive cost of the current move depends on whether the opponent's previous move was *predictable* relative to the player's prior plan. When the opponent plays the move the player was already calculating, the cached plan carries forward and the next response is fast. When the opponent surprises, the plan must be revised and thinking time spikes. Subjectively, this is the familiar moment of cognitive reset: the mental tree collapses, the cached plan is discarded, and the player must actively rebuild their strategy from scratch. We call this pattern *the shadow of the past*.

Games preserve the structure of real sequential decisions while making large-scale measurement possible (Allen et al., 2024; Kuperwajs et al., 2023). Chess is especially useful because expertise, search, and time pressure have already been studied in detail (Sunde et al., 2022; van Opheusden et al., 2023). Online chess platforms timestamp every move at second-or-better resolution and expose engine evaluations that quantify the value of additional search; this lets us measure thinking-time fluctuations and link them to the computational structure of the decision. We make four contributions.

(H1) On a quality-filtered cohort of 135,608 live-game tactical sequences played by 74,609 unique players (timing recovered from the original 2019 Lichess games, not from the puzzle-training interface), we measure a robust within-sequence thinking-time spike at the first move that follows an opponent's surprising tactical setup.

(H2) The amortization signature appears only when predictability is operationalized by a *human-style* model (Maia-2) and not by an *engine-best* model (Stockfish). What gets amortized is the human's prior over likely opponent moves, not what an engine deems objectively best.

(H3) The thinking-time spike is not a passive surprise reaction but a marker of *recognition followed by recalculation*: players who succeeded had spikes roughly twice the size of those of players who failed.

(Model) We introduce the Bayesian Sampling Model with Cache (BSMC), a fitted three-parameter cognitive model that ties these findings together and recovers a measurable cache strength $s = 0.085$ from the data. The findings replicate on an independent 2024 Lichess cohort and on FIDE World Rapid + Blitz 2024 over-the-board games, indicating that the effect is a property of human chess cognition rather than a quirk of one snapshot.

In G. Boleda, I. Dautriche, M. Maldonado, C. Saldana and T. Torrent (Eds.), *Proceedings of the 48th Annual Conference of the Cognitive Science Society*

Background

Resource-rational analysis. Cognition operates under strict resource limits, and a growing literature treats human reasoning as the optimal use of that limited compute (Anderson, 1990; Callaway et al., 2022; Lieder & Griffiths, 2020). Russek et al. (2025) applied this framework to chess: thinking time at move t scales with the within-move benefit of search, ΔU_C, which they operationalize as the Stockfish $d = 15$ vs $d = 1$ evaluation gap. Callaway et al. (2022) provide a broader treatment of how humans rationally allocate planning resources across complex tasks. By construction the Russek specification treats each move independently of the moves around it — a simplification we relax in this paper.

Amortized inference. Gershman and Goodman (2014) introduced amortization as a way to make repeated probabilistic inference cheaper. Dasgupta (2019) and Dasgupta et al. (2018, 2020) formalize human cognition as approximate Bayesian inference in which past computation is *reused*. Their variational recognition model $Q_\varphi(h \mid d)$ supplies cheap, accurate posteriors for queries that resemble those it was trained on, while out-of-distribution queries pay a recalculation cost. Empirical work in this line has so far focused on one-shot probability judgments. Our contribution is to extend the framework to dynamic, sequential, adversarial planning, where the "training distribution" is the player's prior over likely opponent moves.

Chess expertise and chunking. A long line of work, from de Groot (1965) and Chase and Simon (1973) through template theory (Gobet & Simon, 1996) and the CHREST architecture (Bennett et al., 2020; Gobet & Lane, 2020), attributes expert chess perception to *chunks*: position-specific perceptual templates that supply candidate moves for fast retrieval. Eye-tracking shows that experts gravitate to promising squares within milliseconds (Küchelmann et al., 2024). Amortization in our framework is the *temporal* counterpart of chunking: not perceptual chunks of the *current* position, but cached deliberation from the *previous* move.

Distinct from policy compression. Lai and Gershman (2025) distinguish action chunking — the storing of multi-step policies in a compressed form — from amortization, which is the reuse of past inference. Our work concerns the latter; the predictions about RT structure differ.

Maia-2. McIlroy-Young et al. (2020) and Tang et al. (2024) train neural networks on millions of human Lichess games to predict what a *human* of a given Elo will play in a given position. Maia-2 is the natural human-style predictor; we use it to operationalize opponent predictability throughout this paper.

Methods

Data and cohort

Ecological validity of the puzzle database. The Lichess.org puzzle database (database.lichess.org; Lichess, 2026) contains about 5.9 million tactical positions extracted from completed online games. Lichess generates these algorithmically by scanning real games for moments where the position has a clear best move that is hard to find — the kind of critical decision point where a human, given enough time, can locate the right idea (Lichess, 2026) . What makes the database so useful for cognitive modeling is not just the positions themselves but the rich behavioral metadata that accumulates around them. Every time a player attempts a puzzle, the platform records whether they solved it, updates the puzzle's Elo-style difficulty rating using the Glicko system (Glickman, 1999)(Lichess, 2026), narrows its rating deviation, and accumulates a user-quality score (0–100) reflecting community feedback. Each puzzle also receives semantic tags describing its tactical motif (e.g., *fork*, *pin*, *mateIn2*, *middlegame*, *endgame*). After many attempts per puzzle, these fields provide a stable aggregate behavioral signal of how human players, in aggregate, respond to the position.

Reconstructing temporal dynamics. The puzzle database has one critical limitation for our purposes: it does not record *timing*. To recover the actual time players spent thinking when they encountered the position in real play, we used the fact that every puzzle originates from a specific game in the public PGN archive (Lichess, 2026) . We re-downloaded the 2019 monthly Lichess PGN dumps (the same year used in Russek et al. (2025)) and matched each puzzle, by game identifier and ply, back to its source game. The resulting cohort is the intersection of the puzzle database and the real-game PGN database: every entry has both the puzzle's algorithmic difficulty estimate and behavioral tags *and* the move-by-move clock, evaluation, and player-history information from the original game. This intersection is what lets us measure real human thinking time on positions of known, calibrated difficulty.

Puzzle-quality filter. The cascade described below includes a conservative metadata-based quality filter that uses the puzzle metadata Lichess accumulates from community play: Glicko rating deviation < 90 (excluding puzzles whose difficulty estimate is not yet stable), community user-quality score (popularity) $\geq 50/100$ (excluding puzzles humans flagged as uninteresting, incorrect, or otherwise low-signal), and at least 50 prior attempts (ensuring the rating estimate has converged). The filter is conservative by construction. It preserves the central mass of the puzzle distribution but excludes the long tail of low-popularity, weakly-rated, or rarely-attempted puzzles. We do not impose an extra per-player sequence cap: each unique player contributes their sequences as observed (median 1, p90 = 3 sequences), so the hierarchical regressions exploit between-player variability in skill without overweighting any one player.

Filter cascade. Our analytical unit is the *live-game tactical sequence*: the four-ply window indexed as p_0 (OPP_SETUP, the deliberately surprising blunder, identified post-hoc via the Lichess puzzle database), p_1 (SOLVER_1, the player's first response), p_2 (OPP_2, the opponent's reply), and p_3 (SOLVER_2, the player's second move). Crucially, all timing data come from the original live games — the players were unaware they were facing what Lichess would later annotate as a "puzzle."

We applied a transparent filter cascade (N at each step):

Raw live-game tactical sequences (from 2019 mapping): **240,990**
→ RT in [0.5, 300] s at all four plies: ∼218,000
→ four-ply segment recoverable: ∼178,000
→ engine eval at all four plies: ∼158,000
→ player Elo + puzzle Rating non-missing: **135,608** played by **74,609 unique players**.

The RT bounds [0.5, 300] s exclude pre-moves and abandoned sequences. The four-ply recoverability filter ensures we have a complete OPP_SETUP → SOLVER_1 → OPP_2 → SOLVER_2 segment with valid clock parses on every ply. Engine-evaluation availability lets us compute ΔU_C and Maia-2 predictability features. The final filter retains only sequences with non-missing player Elo and puzzle rating, the two key moderators in our regressions.

Cohort descriptive statistics. The 135,608 sequences are distributed broadly across difficulty and skill (median puzzle rating 1,564, IQR 1,269–1,885, range 518–2,798; median player Elo 1,779, IQR 1,686–1,903, range 732–3,144). Puzzle quality is high: the median puzzle rating deviation (Glicko RD) is 77 (IQR 75–80), and the median user-quality score is 89/100 (IQR 84–93). Each puzzle had been attempted a median of 757 times by other users (IQR 158–2,778), meaning the difficulty estimates are stable. Time controls span rated rapid through classical, dominated by 600+0 (45.1% of sequences) and 900+15 (26.9%). The most common semantic tags are *middlegame* (54.2% of puzzles), *advantage* (44.5%), *endgame* (38.0%), *fork* (15.1%), *mate* (13.6%), *pin* (7.8%), and *sacrifice* (7.7%). The cohort is therefore not an artifact of a single tactical motif or one narrow skill range: it captures the full spread of how thousands of online players respond to a wide variety of tactical situations.

Player-level structure. 74,609 unique players contribute the 135,608 sequences (median 1 sequence per player, 90th-percentile 3). The breadth of the player population — rather than depth in any single player — means our hierarchical regressions exploit between-player variability in skill (z-scored Elo) without being dominated by individual idiosyncrasies. We follow the cohort-construction format introduced by McIlroy-Young et al. (2020) and re-extract from the same publicly hosted 2019 PGN files and the official puzzle database (Lichess, 2026).

Predictors and hierarchical regression

For each tactical sequence we extracted: log-transformed reaction times at OPP_SETUP, SOLVER_1, and SOLVER_2; ΔU_C (Stockfish 14, $d = 15$ vs $d = 1$ centipawn evaluation gap (Stockfish developers, 2026) — the Russek baseline); *opp_pred_maia* (binary: 1 if the Maia-2 top-1 move at the opp_setup position equals the opponent's actual move, conditioned on opponent and player Elo); *opp_pred_sf* (the same flag with Stockfish $d = 15$ best-move as the predictor, used as a comparison); the n-legal-move count at SOLVER_2 as a forcing control; and z-scored player Elo, puzzle Rating, and skill_gap (= player Elo − puzzle Rating). All multivariate models include log_clock and log_tc_base as time-pressure covariates.

We fit a sequence of OLS models on log $\mathrm{RT}_{\text{SOLVER\_1}}$, each adding one feature block. M0 is the intercept; M1 adds ΔU_C (the Russek-static baseline); M2 adds the shadow-of-past prior (log $\mathrm{RT}_{\text{OPP\_SETUP}}$); M3 adds amortization-relevant controls (*solver_recalc*, log n-legal moves at SOLVER_2); M5 brings in the Maia-defined amortization interaction and difficulty controls; M7 adds forcing, skill, and time-pressure controls; M9 adds tree-search complexity (PERFT) and Maia-distribution-entropy chunking interactions. (Intermediate specifications M4, M6, and M8 represent isolated main-effect variants and follow the same statistical trajectory; we omit them from the table for brevity.) We present this sequence of models to demonstrate that the jump in explained variance is robust to standard alternative explanations, such as position complexity. Regressions were fit in `statsmodels`; frequentist summaries and bootstrap checks used `Pingouin` where appropriate (Seabold & Perktold, 2010; Vallat, 2018).

Bayesian Sampling Model with Cache (BSMC)

We fit a three-parameter cognitive model directly to the data. Motivated by Hick's law (which links RT to decision entropy) and sequential-sampling accounts (which standardly log-transform RT to handle right-skew) (Bogacz et al., 2006; Hick, 1952; Wald, 1945), we model log thinking time as scaling linearly with the prior entropy H of the player's distribution over candidate moves: $\log T = \alpha + \gamma H$. We use γ for the BSMC entropy-RT slope to keep it visually distinct from the regression coefficient β reported in the H2 model. We operationalize H as the Shannon entropy of Maia-2's predicted move distribution at each ply, computed from the model's softmax over legal moves: $H(p) = -\sum_i p_i \log p_i$, where $p_i = P_{\text{Maia-2}}(\text{move}_i \mid \text{position})$ conditioned on player and opponent Elo. At SOLVER_2, the player's prior is the *posterior* carried over from SOLVER_1, conditional on the opponent's actual move matching the cached prediction. This shrinks the effective entropy by a factor $(1 - s \times \mathit{opp\_pred\_maia})$, where $s \in [0, 1]$ is the *cache strength*. BSMC inherits its $\log T \propto H$ form from Hick's law and SPRT; our contribution here is this cache term, which couples the entropy form across the two-ply SOLVER_1 / SOLVER_2 system and lets us measure cache strength s directly from the joint RT pattern. We jointly fit (α, γ, s) on the two-equation system by profile likelihood (closed-form OLS on α, γ for fixed s, 1-D bounded search on s), and report 95% bootstrap CIs from 1,000 resamples.

Results

H1: within-sequence amortization

Thinking time at SOLVER_1 substantially exceeds thinking time at SOLVER_2. The mean log-RT difference is **+0.309** (Cohen's $d = 0.235$); a paired t-test on N = 135,608 sequences is decisive ($t = 86$, $p < 10^{-300}$, $\mathrm{BF}_{10} > 10^{100}$). A 2,000-resample bootstrap returns 100% positive draws. Figure 1 shows the

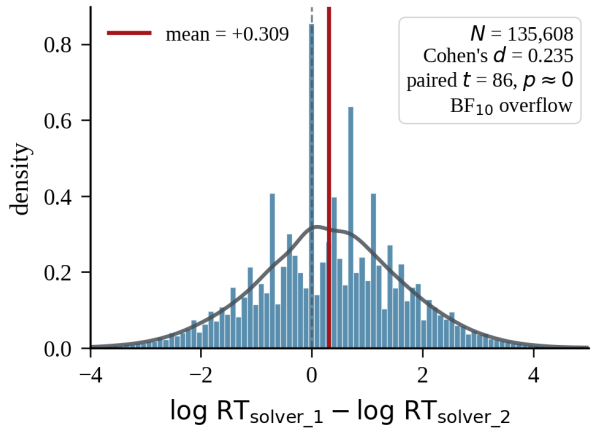

Figure 1: Within-sequence log-RT difference (SOLVER_1 − SOLVER_2) on N = 135,608 sequences. The distribution is sharply positive (mean +0.309, Cohen's d = 0.235); a non-parametric bootstrap returns 100% positive draws, consistent with a reliable within-sequence shift.

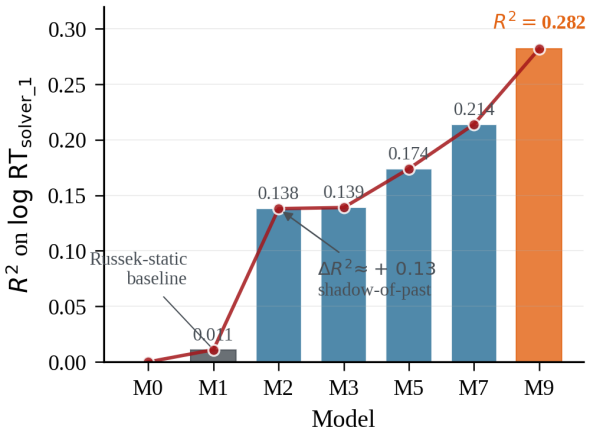

Figure 2: R^2 on log $\text{RT}_{\text{SOLVER\_1}}$ across the M0–M9 hierarchy. The largest single-step gain is M1→M2 (adding the shadow-of-past prior), which alone accounts for roughly half of the total explainable variance recovered by M9.

Table 1: Hierarchical regression on log $\text{RT}_{\text{SOLVER\_1}}$. The largest improvements are M1→M2 (shadow-of-past prior) and M3→M5 (Maia-defined amortization interaction and difficulty controls). All entries N = 135,608.

Model	R^2	ΔAIC
M0 (intercept only)	0.000	—
M1 (+ΔU_C, Russek-static)	0.011	−1,510
M2 (+ log $\text{RT}_{\text{OPP\_SETUP}}$)	**0.138**	−19,120
M3 (+ amortization controls)	0.139	−18,857
M5 (+*opp_pred_maia*×prior)	0.174	−25,180
M7 (full controls)	0.214	−31,250
M9 (+ PERFT + Maia entropy)	**0.282**	−43,391

distribution of per-puzzle log-RT differences, sharply offset to the right of zero.

This effect is not driven by mechanical position differences: the n-legal-move count at SOLVER_2 is on average 30% *larger* than at SOLVER_1, so a forcing-only account would predict the opposite of what we observe. The H1 spike is a property of the time the player *chooses* to spend, not of position complexity.

H2: human-style predictability dominates

Hierarchical regression on log $\text{RT}_{\text{SOLVER\_1}}$ improves at every step. The Russek-static baseline (M1, ΔU_C alone) explains R^2 = 0.011. Adding the shadow-of-past prior (M2: log $\text{RT}_{\text{OPP\_SETUP}}$) jumps R^2 to **0.138** — past thinking time is by far the strongest predictor of current thinking time. Adding Maia-defined amortization features in M5 raises R^2 to 0.174. The full M9 specification reaches R^2 = 0.282 (Table 1), a 25-fold improvement over the Russek-static baseline. Figure 2 visualizes the climb model by model.

The *form* of the predictor matters as much as its presence. The amortization interaction (log $\text{RT}_{\text{OPP\_SETUP}}$ × *opp_pred*) is strong and negative when *opp_pred* is the Maia-2 top-1 match ($\beta = -0.054$, $p = 1.8 \times 10^{-28}$): when the opponent plays the move a Maia-trained predictor expects, the shadow-of-past slope flattens, the cached plan carries forward, and the next move is fast. The same interaction with Stockfish-best as the predictor is *not significant* (p = 0.18). What humans amortize against is what *humans* are likely to play, not what an engine evaluates as objectively best. This highlights a crucial theoretical distinction: although Stockfish at d = 15 is the natural normative baseline for ΔU_C, it is the wrong predictor of *predictability*. The human-aligned Maia-2 is the better match.

The shadow-of-past coefficient is also robust to all standard alternative explanations. Adding Maia-2 difficulty controls, puzzle-rating, player-Elo, branching factor, and recalc covariates moves $\hat{\beta}$ on log $\text{RT}_{\text{OPP\_SETUP}}$ from 0.340 in M2 to 0.323 in M5 — a 5% reduction with the t-statistic remaining above 100. The headline effect is therefore not an artifact of unmodeled difficulty.

The cache parameter is measurable (BSMC)

BSMC fitted on the full cohort returns α = 1.186, γ = 0.674, and **s** = **0.085** with 95% bootstrap CI [0.071, 0.099], 100% bootstrap-positive across 1,000 resamples. Refitting BSMC within player Elo bands shows a monotone trend: s rises from 0.058 (Elo 1500–1700) to 0.072 (1700–1900) to 0.118 (1900–2100) to **0.145** (2100+). Stronger players exhibit substantially more effective cache management — exactly the prediction chess-expertise theory makes about the temporal counterpart of chunk-based perception (Figure 3).

Beyond fitting cache strength as a single number, the BSMC parameters and the regression-recovered interaction tell quantitatively consistent stories about the same cache mechanism. Within BSMC, the cache reduces effective entropy at

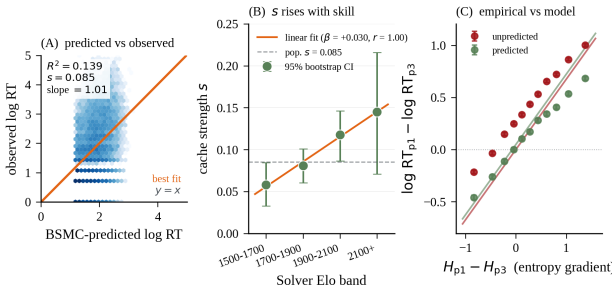

Figure 3: BSMC fit. **Left**: BSMC-predicted vs observed log RT on an 8,000-sequence subsample (the fit itself uses all 135,608 sequences); joint $R^2 = 0.139$, $s = 0.085$, best-fit slope 1.01 indicates no systematic bias. **Middle**: cache strength s rises monotonically with player Elo (95% bootstrap CIs). **Right**: empirical (markers, ± SEM) vs model-predicted (lines) RT gradient, stratified by opp_pred_maia.

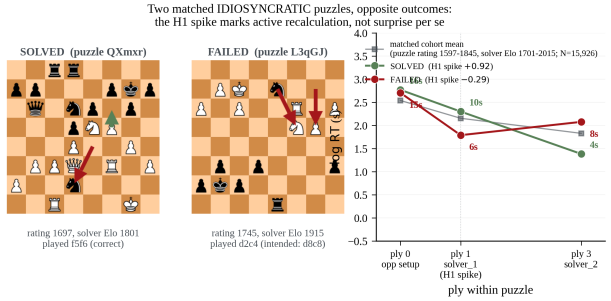

Figure 4: Two matched sequences, opposite outcomes. **Left board**: player invests 10 s at SOLVER_1, plays the correct move, and dispatches SOLVER_2 in 4 s (H1 spike +0.92). **Middle board**: matched player invests only 6 s at SOLVER_1, plays the wrong move, and sinks 8 s at SOLVER_2 (H1 spike −0.29). **Right panel**: log-RT trajectories for both example players, plotted alongside the *matched cohort mean* — the average across cohort sequences at similar puzzle rating and similar player Elo. The cohort-level failed mean is higher in absolute SOLVER_1 time, but this matched pair illustrates the stunted relative recalculation spike. The H1 spike marks recognition followed by recalculation, not passive surprise. Light pink arrows on the boards mark the opponent's setup move; bold arrows (green for the correct example, red for the failed one) mark the player's actual first move.

SOLVER_2 by a factor $(1 - s \times opp\_pred\_maia)$; the implied per-trial log-RT savings on SOLVER_2 when $opp\_pred\_maia = 1$ scale with $-\gamma \times s \times \bar{H}_{p_3}$, which equals $-0.674 \times 0.085 \times \bar{H}_{p_3}$. With the cohort-mean SOLVER_2 entropy $\bar{H}_{p_3} \approx 1.55$ nats, this gives a cache effect of about −0.089 nats, comparable in magnitude to the M5 regression interaction ($\beta = -0.054$ on $\log \mathrm{RT}_{\mathrm{SOLVER\_1}}$). The two quantities live on different equations of the BSMC system, so we do not claim a one-to-one identity; we note only that the cognitive model and the regression converge on the same order-of-magnitude cache benefit. The mathematical link is structural: the interaction exists because the cache exists, and BSMC measures the cache parameter s directly rather than inferring it from a regression.

H3: the spike is a recognition-and-recalculation signal

For every tactical sequence we know whether the player played the engine's intended solution at SOLVER_1. The cohort solve rate is 67.0%. Splitting the H1 spike by outcome:

- Solved (N = 90,924): mean log H1 spike = **+0.375**
- Failed (N = 44,684): mean log H1 spike = **+0.174**

A Welch's t-test, solved vs failed, gives $t = 26.2$, $p = 3.6 \times 10^{-150}$. **Players who succeeded had spikes roughly twice the size of players who failed.**

Counterintuitively, failed sequences take longer in absolute seconds at SOLVER_1 (17.3 s vs 11.9 s) yet show a *smaller* log-RT spike. The arithmetic implication is direct: because absolute time at p_1 is longer for failed players, a smaller relative spike mathematically forces them to be sinking even more time into p_3, attempting to recover from a position they have already compromised. Failure is therefore characterized not by rushing the first move, but by a failure to decisively resolve the surprise.

The H1 spike, then, is *not* a passive consequence of opponent surprise. It is a marker of successful tactical recognition followed by decisive recalculation: players who register the changed board state produce a sharp recalculation spike, while players who miss the tactic diffuse their deliberation across both plies without ever triggering a clean cache-invalidation event. Figure 4 contrasts two matched sequences — same difficulty rating, similar player Elo — in which one player invested 10 s at SOLVER_1 and dispatched SOLVER_2 in 4 s (spike = +0.92, solved), while the matched player invested only 6 s at SOLVER_1 and then sank 8 s at SOLVER_2 (spike = −0.29, failed). The gray reference line in the right panel is a *matched cohort mean*: the average log RT at each ply across all sequences within ±100 of the example puzzles' rating range and within ±100 of the example players' Elo range, so the reference is matched on both puzzle difficulty and player skill.

External validity

The headline findings are not specific to the 2019 snapshot. (a) On a 47,982-puzzle cohort drawn from 2024-01 Lichess data with no overlap with the 2019 cohort, the H1 spike is +0.252 ($t = 46$, N = 47,982; the test statistic is well past any conventional threshold, with the standard p-value computation underflowing double-precision arithmetic); BSMC cache strength is $s = 0.104$ [0.081, 0.130], whose 95% credible interval overlaps the 2019 estimate. (b) Using what we call *blunder-puzzle synthesis* — treating each large evaluation drop in a competitive game as a synthetic puzzle event — we re-tested the H1 paired comparison on 6,768 blunder events drawn from FIDE World Rapid + Blitz Championship 2024 broadcast PGNs. The H1 spike replicates (+0.188, $t = 10.7$,

$p < 10^{-25}$); for the FIDE World Rapid Open division alone the spike (+0.322) actually *exceeds* that of the 2019 Lichess online cohort. The amortization signature is therefore a property of human chess cognition that survives time-control variation, year-of-data variation, and the transition from amateur online play to elite over-the-board tournament play.

Discussion

The single biggest predictor of current thinking time is *prior* thinking time, far stronger than the within-move benefit of computation that Russek et al. (2025) measured. This does not contradict their account; it extends it. Russek showed that humans spend time when computation will pay off. We show that the paying-off itself is amortized: the work done at one move can be reused at the next, modulated by whether the opponent plays a humanly-likely move.

The Bayesian Sampling Model with Cache makes this analytical: thinking time scales with the *effective* entropy of the player's prior, and the cache shrinks that entropy by a measurable factor $s = 0.085$. Importantly, the cache parameter s is directly estimated from behavioral data, bridging theoretical constructs with empirical measurement. It is the temporal generalization of Dasgupta's variational recognition-model framework (Dasgupta, 2019; Dasgupta et al., 2020) to dynamic sequential planning: Q_φ becomes a time-indexed prior over candidate moves, with cache strength quantifying how well it transfers across moves. The per-band climb in s aligns with the chess expertise literature: stronger players have denser, more transferable caches — the temporal counterpart of the dense perceptual chunks described by de Groot (1965), Chase and Simon (1973), and Gobet and Simon (1996).

The cache benefit is *human-aligned*: it is governed by Maia-defined predictability, not by Stockfish best-move. This says amortization tracks the regularities of human play, which is what we would expect if the underlying cognitive cache is built from a lifetime of training against other humans. The Stockfish-defined version of the test is predicted to fail; it does ($p = 0.18$).

The recognition-and-recalculation finding (H3) reframes what amortization is *for*. Failed puzzles do not show smaller spikes because players think *less overall*; they show smaller spikes because many of those players did not register the tactic, so no cache-invalidation event was triggered and deliberation remained diffused across the two plies. The signature distinguishes players who recognized the changed board state from those who did not. This connects amortization, an RT phenomenon, to the *outcome* of the planning episode in a way that earlier resource-rational accounts of thinking time have not. More broadly, the spike offers a simple behavioral marker of whether a person has registered the moment that requires revision, a possibility worth testing beyond chess. This failure to revise a cached line also connects to the classic Einstellung effect in chess, where an available plan can bias attention away from better alternatives (Bilalić et al., 2008).

Limitations and future work. Four limitations bound the claim. (i) Maia-2 was trained primarily on rapid-control online games; the amortization interaction holds in bullet, blitz, rapid, and classical sub-cohorts, with bullet showing the *largest* coefficient, arguing against a simple training-distribution artifact. (ii) We operationalize opponent predictability as a binary indicator (Maia-2 top-1 match). A graded version using Maia-2's full posterior $P(\textit{opponent move})$ as a continuous predictor is the natural extension; preliminary work suggests the binary collapse loses real information about marginal cases. We leave the graded specification as a promising direction for future work. (iii) We have not fitted BSMC hierarchically with non-linear $s(H)$; a preliminary diagnostic suggests an inverted-U in entropy space. (iv) The cache strength s is a *population* parameter; we have not yet estimated per-player s, which would let us ask whether individual differences in amortization predict skill development trajectories. This is a natural follow-up. Our blunder-puzzle synthesis on OTB games also defines blunder events at a fixed evaluation-drop threshold; sensitivity to this threshold is reported in supplementary material.

Acknowledgements

Code and processed data are available from the corresponding author on reasonable request. The raw inputs — Lichess game and puzzle databases — are publicly hosted at https://database.lichess.org (Lichess, 2026), so the analysis pipeline can be reconstructed from public sources. We thank the Lichess team for maintaining the public puzzle and broadcast databases, and the original Maia and Russek teams for releasing the artifacts that made this analysis possible.

References

Allen, K. R., Brändle, F., Botvinick, M., Fan, J. E., Gershman, S. J., Gopnik, A., Griffiths, T. L., et al. (2024). Using games to understand the mind. *Nature Human Behaviour*, *8*, 1035–1043. https://doi.org/10.1038/s41562-024-01878-9

Anderson, J. R. (1990). *The adaptive character of thought*. Lawrence Erlbaum Associates.

Bennett, D., Gobet, F., & Lane, P. C. R. (2020). Forming concepts of Mozart and Homer using short-term and long-term memory: A computational model based on chunking. *Proceedings of the Annual Meeting of the Cognitive Science Society*, *42*. https://escholarship.org/uc/item/2v5293gd

Bilalić, M., McLeod, P., & Gobet, F. (2008). Inflexibility of experts–reality or myth? quantifying the Einstellung effect in chess masters. *Cognitive Psychology*, *56*(2), 73–102. https://doi.org/10.1016/j.cogpsych.2007.02.001

Bogacz, R., Brown, E., Moehlis, J., Holmes, P., & Cohen, J. D. (2006). The physics of optimal decision making: A formal analysis of models of performance in two-alternative forced-choice tasks. *Psychological Review*, *113*(4), 700–765. https://doi.org/10.1037/0033-295X.113.4.700

Callaway, F., van Opheusden, B., Gul, S., Das, P., Krueger, P. M., Griffiths, T. L., & Lieder, F. (2022). Rational use of cognitive resources in human planning. *Nature Human Behaviour*, *6*, 1112–1125. https://doi.org/10.1038/s41562-022-01332-8

Chase, W. G., & Simon, H. A. (1973). Perception in chess. *Cognitive Psychology*, *4*(1), 55–81. https://doi.org/10.1016/0010-0285(73)90004-2

Dasgupta, I. (2019). *Algorithmic approaches to ecological rationality in humans and machines* [Doctoral dissertation, Harvard University]. https://nrs.harvard.edu/URN-3:HUL.INSTREPOS:42029476

Dasgupta, I., Schulz, E., Goodman, N. D., & Gershman, S. J. (2018). Remembrance of inferences past: Amortization in human hypothesis generation. *Cognition*, *178*, 67–81. https://doi.org/10.1016/j.cognition.2018.04.017

Dasgupta, I., Schulz, E., Tenenbaum, J. B., & Gershman, S. J. (2020). A theory of learning to infer. *Psychological Review*, *127*(3), 412–441. https://doi.org/10.1037/rev0000178

de Groot, A. D. (1965). *Thought and choice in chess*. Mouton.

Gershman, S. J., & Goodman, N. D. (2014). Amortized inference in probabilistic reasoning. *Proceedings of the Annual Meeting of the Cognitive Science Society*, *36*. https://escholarship.org/uc/item/34j1h7k5

Glickman, M. E. (1999). Parameter estimation in large dynamic paired comparison experiments. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, *48*(3), 377–394. https://doi.org/10.1111/1467-9876.00159

Gobet, F., & Lane, P. C. R. (2020). The CHREST architecture of cognition: From chess to chunking. In *Cognitive architectures and human-machine interaction*. Routledge.

Gobet, F., & Simon, H. A. (1996). Templates in chess memory: A mechanism for recalling several boards. *Cognitive Psychology*, *31*(1), 1–40. https://doi.org/10.1006/cogp.1996.0011

Hick, W. E. (1952). On the rate of gain of information. *Quarterly Journal of Experimental Psychology*, *4*(1), 11–26. https://doi.org/10.1080/17470215208416600

Küchelmann, T., Reinhard, I., Schläfer, A., & Velichkovsky, B. M. (2024). Expertise-dependent visuocognitive performance of chess players in mating tasks: Evidence from eye movements during task processing. *Cognitive Processing*. https://doi.org/10.1007/s10339-024-01194-0

Kuperwajs, I., Schütt, H. H., & Ma, W. J. (2023). Using deep neural networks as a guide for modeling human planning. *Scientific Reports*, *13*, 20269. https://doi.org/10.1038/s41598-023-46850-1

Lai, L., & Gershman, S. J. (2025). Action chunking and the compositionality of policy abstractions. *Cognition*, *255*, 106018. https://doi.org/10.1016/j.cognition.2024.106018

Lichess. (2026). Lichess.org open database [Accessed May 9, 2026].

Lieder, F., & Griffiths, T. L. (2020). Resource-rational analysis: Understanding human cognition as the optimal use of limited computational resources. *Behavioral and Brain Sciences*, *43*, e1. https://doi.org/10.1017/S0140525X1900061X

McIlroy-Young, R., Sen, S., Kleinberg, J., & Anderson, A. (2020). Aligning superhuman AI with human behavior: Chess as a model system. *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 1677–1687. https://doi.org/10.1145/3394486.3403219

Russek, E., Acosta-Kane, D., van Opheusden, B., Mattar, M. G., & Griffiths, T. L. (2025). Time spent thinking in online chess reflects the value of computation. *Nature Human Behaviour*. https://doi.org/10.1038/s41562-024-02077-2

Seabold, S., & Perktold, J. (2010). Statsmodels: Econometric and statistical modeling with Python. *Proceedings of the 9th Python in Science Conference*. https://doi.org/10.25080/Majora-92bf1922-011

Stockfish developers. (2026). Stockfish: A free and strong UCI chess engine [Version 14 used in analyses; accessed May 9, 2026].

Sunde, U., Zegners, D., & Strittmatter, A. (2022). *Speed, quality, and the optimal timing of complex decisions: Field evidence* (CESifo Working Paper No. 9546). CESifo. https://doi.org/10.48550/arXiv.2201.10808

Tang, Z., Jiao, D., McIlroy-Young, R., Kleinberg, J., Sen, S., & Anderson, A. (2024). Maia-2: A unified model for human-AI alignment in chess. *Advances in Neural Information Processing Systems*, *37*, 20919–20944. https://doi.org/10.52202/079017-0659

Vallat, R. (2018). Pingouin: Statistics in Python. *Journal of Open Source Software*, *3*(31), 1026. https://doi.org/10.21105/joss.01026

van Opheusden, B., Galbiati, G., Kuperwajs, I., Bnaya, Z., & Ma, W. J. (2023). Expertise increases planning depth in human gameplay. *Nature*, *618*, 1000–1005. https://doi.org/10.1038/s41586-023-06124-2

Wald, A. (1945). Sequential tests of statistical hypotheses. *The Annals of Mathematical Statistics*, *16*(2), 117–186. https://doi.org/10.1214/aoms/1177731118