Stabilizing RL+Search for Imperfect-Information Games

Yinghao Li*
China Agricultural University
lyh010428@alu.cau.edu.cn

Abstract

Imperfect-information games model decision making when no player sees the whole state. Solving a large one is computationally prohibitive, because equilibrium-finding algorithms traverse the game tree, which grows exponentially with depth. Depth-limited solving sidesteps the cost by searching only a few moves ahead from the current state and estimating the value of the deeper nodes with a neural network, and the strongest programs for these games combine this search with reinforcement learning (RL+Search). In the Recursive Belief-based Learning (ReBeL) framework, every depth-limited subgame is solved with the same fixed number of counterfactual regret minimization (CFR) iterations. We show that this fixed budget is the dominant remaining source of seed-dependent noise in the recursive self-play loop. It forces the value network to fit policies of one solver maturity, and the resulting errors compound along the recursion. We propose **Stochastic Horizon Annealing (SHA)**, which draws the budget of each subgame from a truncated Gaussian centered at the same mean and leaves the rest of ReBeL's loop unchanged. The value network then regresses targets from a continuum of solving depths, which weakens the correlation of the recursive errors and acts as an implicit ensemble at zero expected cost. SHA is a one-line change to any ReBeL-style trainer. Our analysis gives three results. SHA preserves the convergence rate of linear CFR. Under a regularity condition on how a target responds to its budget, the variance it adds to a single target is a small fraction of the squared error of one solve. Under an explicit cross-depth error-correlation model, which we verify on trained networks, the ratio of SHA to fixed-budget target variance decreases monotonically with the recursion depth. The theory therefore predicts lower variance across training runs, growing with the depth of the game, and experiments bear this out. On three Liar's Dice variants and two Leduc variants, SHA narrows the across-run spread of final exploitability, the amount a best-responding opponent wins, by 34% to 61%, and within each family the reduction grows with the size of the game. Randomizing the horizon of the inner solver is thus a cheap stabilizer for recursive self-play with search.

1 Introduction

Imperfect-information games model sequential decision making when no player sees the whole state. Each player holds private information, so the value of each action depends on the distribution over what the opponent may hold, and that distribution depends on what the opponent believes about ours [Johanson et al., 2012]. A strategy is judged by its exploitability, the amount a best-responding opponent can win against it, and the aim is a Nash strategy that leaves no such opening. Solving a game directly requires traversing its whole tree, and the tree grows exponentially with depth. Heads-up limit Texas hold'em was essentially solved in 2015 [Bowling et al., 2015], and the games played by the agents that followed are far larger.

*Corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

Counterfactual regret minimization (CFR) is the workhorse for these games [Zinkevich et al., 2007]. It plays through the game repeatedly and shifts probability toward actions that have accumulated less regret at each information set. In two-player zero-sum games its running average converges to Nash at a rate that scales as the inverse square root of the number of iterations. CFR visits the whole tree on every iteration, so in a large game it is run only on a subgame around the current state. Depth-limited solving [Burch et al., 2014, Brown et al., 2018] looks a few moves ahead and replaces the rest of the game with a learned value function. The superhuman poker agents of the past decade are built on it, among them DeepStack and Libratus in no-limit hold'em and Pluribus in six-player poker [Moravčík et al., 2017, Brown and Sandholm, 2018, 2019b]. Chess and Go engines pair lookahead with a learned value function in the same way [Silver et al., 2018].

Recursive Belief-based Learning (ReBeL) [Brown et al., 2020] removed the hand-designed abstractions these systems relied on. It lifts the game to public belief states (PBS) [Nayyar et al., 2013], the common-knowledge distribution over the players' private information given the public actions. This turns an imperfect-information game into a perfect-information one over beliefs. A network supplies the value of a leaf belief, and CFR solves the depth-limited subgame below it. The next belief state is sampled from the policy at one iteration of the solve, chosen in advance and uniformly at random. One episode therefore solves a chain of nested subgames, and the solved value at each level becomes the leaf target of the level above. We call the length of this chain the recursion depth. It is set by the depth of the public tree, the tree of public observations, and is distinct from the lookahead depth of a single subgame.

A fixed budget makes this loop fragile. Every subgame is solved for the same number of CFR iterations, so every regression target comes from a single solver maturity. The network cannot tell whether a target came from a nearly converged solve or from a solver that stopped early, and it must absorb both into a single function. The recursion then compounds the error, because each target is the leaf value of the subgame above and a bias introduced at one level enters the next as input. In practice exploitability oscillates and a long tail of seeds converges poorly. ReBeL already randomizes which iteration supplies the next belief state. The budget is the last constant shared by all subgames, and we show below that it dominates the seed-dependent variance of the recursion.

We propose **Stochastic Horizon Annealing (SHA)**. Each depth-limited subgame draws its budget from a truncated Gaussian centered at the nominal budget, and the choice of that iteration stays uniform. The draw stays between half and one and a half times the nominal budget, so no solve becomes degenerate or disproportionately expensive. The symmetry of the distribution keeps the expected number of iterations within half an iteration of the nominal budget. Each value target then averages over a band of solver maturities, weakening the correlation between errors at different depths, the quantity the recursion amplifies. The effect is an implicit ensemble over solvers at no expected cost, and the change to a ReBeL trainer is a single line.

The analysis in Section 5 makes these statements precise. It shows that the randomized budget preserves the convergence rate of linear CFR and adds a small fraction of the squared solve error to each target. A model of how value errors correlate across recursion depths then gives the variance of the recursive target, and the SHA-to-ReBeL ratio there falls monotonically with depth, so deeper games gain more. Section 6 tests each prediction. On the three Liar's Dice variants the across-seed interval of final exploitability narrows by 34% to 61% and the mean falls by 7% to 17%, with the largest reductions on the largest variant. Leduc-30 and Leduc-40 extend the recursion depth, and there SHA cuts the across-run standard deviation by 40% and 53%. Probes of trained networks put the cross-depth error correlation at 0.373 under SHA against 0.749 under a fixed budget, as the model assumes.

Our contributions are:

- We introduce **Stochastic Horizon Annealing (SHA)**, a one-line, cost-neutral change to any ReBeL-style trainer that draws the budget of each depth-limited subgame from a truncated Gaussian centered at the usual budget. To our knowledge, SHA is the strongest depth-limited method for solving imperfect-information games.
- We give the first analysis of the fixed iteration budget as a variance channel in the recursion. We prove that SHA preserves the convergence rate of linear CFR and, under the stated regularity condition, that the variance it adds to a single target is a small fraction of the squared solve error. Under an explicit cross-depth error-correlation model, the ratio of SHA to fixed-budget target variance decreases monotonically with the recursion depth.

- We verify the mechanism on trained networks and the predicted stabilization on 30-seed self-play. The cross-depth error correlation falls from 0.749 to 0.373 and predicts the measured drop in nested root-value variance. On Liar's Dice, SHA shrinks the across-seed interval of final exploitability by 34% to 61%, with the largest reduction on the largest variant, as the depth dependence predicts, and lowers the mean by 7% to 17% alongside. On the deeper Leduc-30 and Leduc-40 it cuts the across-run standard deviation by 40% and 53%.

2 Related Work

CFR and depth-limited subgame solving. Sampling [Lanctot et al., 2009] and discounted weighting [Brown and Sandholm, 2019a] extend CFR to large games. ReBeL [Brown et al., 2020] is the closest relative of our work. Its inner loop already picks the iteration that supplies the next belief state uniformly, and SHA randomizes the only remaining static degree of freedom, the iteration budget itself. Accelerations such as RL-driven action abstraction [Li et al., 2024] and online pruning [Li and Huang, 2025] target a different bottleneck and compose with SHA unchanged.

Noise injection as a stabilizer for learning. Structured noise during training is a long-standing stabilizer. Dropout adds multiplicative noise to hidden units and acts as an implicit ensemble, later formalized as a regularizer with an explicit bias-variance trade-off [Wan et al., 2013, Srivastava et al., 2014]. Stochastic depth trains each layer under a distribution of effective depths [Huang et al., 2016], and label smoothing softens deterministic targets [Szegedy et al., 2016]. For exploration, parameter-space noise [Plappert et al., 2018] and noisy networks [Fortunato et al., 2018] serve the same purpose. SHA transfers the idea to the solver and is closest in spirit to randomized smoothing [Duchi et al., 2012], which convolves a non-smooth objective with a Gaussian at no asymptotic cost.

Neural equilibrium finding and RL+Search after ReBeL. A parallel line replaces tabular regret storage with function approximation. Deep CFR [Brown et al., 2019] regresses counterfactual advantages from sampled traversals, Double Neural CFR [Li et al., 2020] learns the average strategy as well, and DREAM [Steinberger et al., 2020] and ESCHER [McAleer et al., 2023] add baselines and history value functions that reduce the variance of sampled regrets. Dynamic Discounted CFR learns the discounting schedule itself [Xu et al., 2024, 2026], and later variants add pre-trained embeddings [Fu et al., 2026]. Policy-gradient methods with provable convergence [Liu et al., 2025, Kalogiannis and Farina, 2026] and reappraisals of their practical strength [Rudolph et al., 2026] form a related search-free family. These methods are our non-recursive reference point.

On the search side, safe and nested subgame solving established how to re-solve subgames online [Brown and Sandholm, 2017, 2018]. Kovařík et al. [2023] characterize the value functions that make depth-limited solving sound, Šustr et al. [2021] probe how soundness is measured, and Kovařík et al. [2022] formalize the public-state models ReBeL relies on. Student of Games [Schmid et al., 2023] and the Stratego agent of Sokota et al. [2026] scale the same recipe, while DeepNash reaches superhuman Stratego with no search [Perolat et al., 2022]. Later ReBeL variants change the search itself. Zeng et al. [2025b] vary the subgame depth, and Kubíček et al. [2026a] refine the equilibrium inside the solver. Search without common knowledge appears in Zhang and Sandholm [2026], counter-strategies beyond the depth limit in Milec et al. [2025], and learned look-ahead models in Kubíček and Lisý [2026]. Others change the network and its data. A graph neural network serves as the value network [Zeng et al., 2025a], belief learning is accelerated [Li and Huang, 2026i], and belief embeddings feed the search [Solinas et al., 2025]. Generative opponent models [Li et al., 2023, 2025] and augmented self-play data [Lanier et al., 2026] supply the training signal. All keep the inner budget fixed, and SHA changes only its distribution.

Schedules, perturbation, and variance inside the solver. Schedules for hyperparameters and step sizes [Li and Huang, 2026c, Meng et al., 2026a, Zhang et al., 2026] and parameter-free, last-iterate-convergent regret matching [Cai et al., 2025, Meng et al., 2025b,c, 2026b, Li and Huang, 2026g] shape what the outer learner sees. So do divergence-regularized aggregation [Lu et al., 2025], meta-learned self-play regret minimization [Sychrovský et al., 2025], and perturbation or variance reduction inside sampled solvers [Masaka et al., 2025, Meng et al., 2025a, Li et al., 2026b].

3 Preliminaries

Imperfect-information extensive-form games. We consider an imperfect-information extensive-form game with agents $\mathcal{N} = \{1, \dots, n\}$, world state $w \in \mathcal{W}$, joint action space $\mathcal{A} = \times_i \mathcal{A}_i(w)$, and stochastic transitions $\mathcal{T}(w, \mathbf{a}) \in \Delta(\mathcal{W})$. Each agent observes a private signal O^i_{priv} and a public signal O_{pub}, leading to an infostate $s_i = (O^i_0, a^i_0, \dots, O^i_t)$. A behavioral strategy π_i maps each infostate to a distribution over legal actions. In two-player zero-sum games the exploitability of a profile π is $\text{expl}(\pi) = v_1^{\text{br}_1,\pi_2}(h_0) - v_1^{\pi_1,\text{br}_2}(h_0)$, with br_i a best response and h_0 the root.

Counterfactual Regret Minimization. CFR maintains, for every information set I and action a, a cumulative counterfactual regret $R^t(I, a) = \sum_{\tau \le t} r^\tau(I, a)$, where $r^\tau(I, a)$ is the instantaneous regret of the profile π^τ played at iteration τ. The next strategy is obtained by regret matching, $\pi^{t+1}(a \mid I) \propto \max\{R^t(I, a), 0\}$, with the uniform distribution when no regret is positive. The average strategy is the realization-equivalent one, $\bar{\pi}^T_i(a \mid I) = \sum_{t \le T} w_t q^t_i(I) \pi^t_i(a \mid I) / \sum_{t \le T} w_t q^t_i(I)$, with $q^t_i(I)$ player i's own contribution to the reach probability of I under π^t_i (uniform if the denominator is zero). With $w_t = 1$ the classical bound [Zinkevich et al., 2007] is $\text{expl}(\bar{\pi}^T) = O(|\mathcal{I}|\sqrt{|\mathcal{A}|/T})$ in two-player zero-sum games with perfect recall. Linear CFR [Brown and Sandholm, 2019a] uses $w_t = t$ in the regrets and in the averages and keeps the $O(T^{-1/2})$ rate; the linearly weighted value average obeys

$$v^t = \frac{t-1}{t+1} v^{t-1} + \frac{2}{t+1} \Delta_t, \tag{1}$$

where Δ_t is the value of the profile π^t at the root.

Public belief states and ReBeL. A public belief state (PBS) β_r is a common-knowledge distribution over histories conditioned on the public observation [Nayyar et al., 2013, Brown et al., 2020]. ReBeL constructs a depth-limited subgame at each β_r, evaluates leaf PBSs with a learned value network θ_v, and runs linear CFR for a fixed number of iterations N. The pivot $p \sim \text{Uniform}\{1, \dots, N\}$ fixes the iteration whose policy samples the next PBS β'_r. The value $V_{\beta_r} = \sum_{t=1}^N w_t \Delta_t / \sum_{t=1}^N w_t$ with $w_t \propto t$ is the regression target for θ_v at β_r, and solved values accumulate in a replay buffer D_v from which θ_v is trained.

Recursion depth. A self-play episode solves a chain of nested subgames $\beta_1 \to \beta_2 \to \cdots \to \beta_D$, where β_{d+1} is a leaf PBS of the subgame rooted at β_d and the target $V_{\beta_{d+1}}$ is what θ_v is trained to return at that leaf. We write D for the number of subgames in this chain and call it the recursion depth. For a fixed lookahead, recursion depth grows with the depth of the public tree; a deeper lookahead at each step leaves fewer nested subgames. The lookahead depth (the subgame depth that Zeng et al. [2025b] study empirically) is a separate design axis, held at the ReBeL default of 2 throughout. We replay 250 to 500 self-play episodes per trained network with depth-2 subgames and count the subgames solved per episode. The mean recursion depth is 1.6 to 2.0 on Liar's Dice 1×4f, 1×5f, and 1×6f, and 3.2 to 3.4 on Leduc-30 and Leduc-40.

4 Method: Stochastic Horizon Annealing

SHA replaces ReBeL's fixed iteration budget with a stochastic one and keeps the pivot uniform. At each depth-limited subgame rooted at a PBS β_r, the budget is drawn as

$$T \sim \mathcal{N}(N, \sigma^2) \text{ truncated to } [T_{\min}, T_{\max}], \qquad p \sim \text{Uniform}\{1, \dots, \lfloor T \rfloor\}, \tag{2}$$

where $T_{\min}, T_{\max}$ are loose practical bounds and the truncation renormalizes the Gaussian on that interval. All experiments use $T_{\min} = N/2$ and $T_{\max} = 3N/2$. Linear CFR then runs for $\lfloor T \rfloor$ iterations, the next PBS β'_r is sampled from the policy at iteration p, and the value-network target V_{β_r} is the linearly weighted average over all $\lfloor T \rfloor$ iterations. For integer N, $\sigma > 0$, and these symmetric bounds the expected number of iterations is $\mathbb{E}[\lfloor T \rfloor] = N - \frac{1}{2}$ (Lemma 5.2), so SHA preserves the expected number of iterations per solve up to half an iteration. Algorithm 1 gives the modified subgame solve, and Appendix A reproduces the original ReBeL procedure for comparison. The two differ only in the line that draws the budget.

Algorithm 1 ReBeL Depth-Limited Subgame Solving with SHA

Require: Root PBS β_r, value network θ_v, replay buffer D_v
Require: Nominal horizon N, annealing strength $\sigma > 0$, lookahead depth L, bounds $1 \le T_{\min} < T_{\max}$

while β_r is not terminal **do**
  Sample $T \sim \mathcal{N}(N, \sigma^2)$ truncated to $[T_{\min}, T_{\max}]$ ▷ SHA: budget randomized
  Sample $p \sim \text{Uniform}\{1, \dots, \lfloor T \rfloor\}$ ▷ ReBeL: pivot already uniform
  Build the depth-limited subgame G at β_r with lookahead L
  $R \leftarrow 0$; $\pi^1 \leftarrow$ uniform; $\bar{\pi} \leftarrow$ empty; $V_{\beta_r} \leftarrow 0$
  for $t = 1, \dots, \lfloor T \rfloor$ **do**
    Evaluate leaf PBSs of G under π^t via θ_v
    $\Delta_t \leftarrow$ value of π^t at the root of G
    $V_{\beta_r} \leftarrow \frac{t-1}{t+1} V_{\beta_r} + \frac{2}{t+1} \Delta_t$
    Add π^t to the reach-weighted average $\bar{\pi}$ with weight t
    if $t = p$ **then**
      Sample β'_r from the leaf distribution of π^t in G
    end if
    $\pi^{t+1} \leftarrow$ CFR update of G ▷ regrets of π^t, weight t
  end for
  Append (β_r, V_{β_r}) to D_v
  $\beta_r \leftarrow \beta'_r$
end while

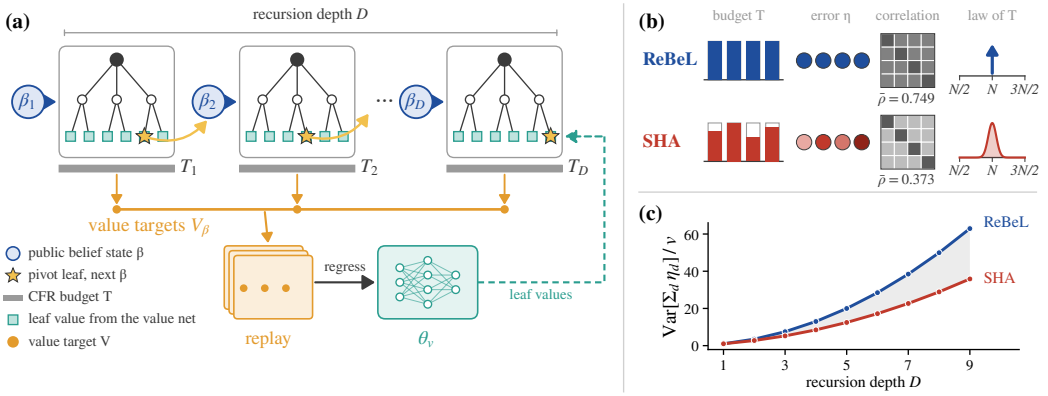


Figure 1: (a) One ReBeL self-play episode is a chain of D nested depth-limited subgames. CFR solves each one for T_d iterations with leaves evaluated by the value network θ_v, the solved root value is a regression target for θ_v, and the pivot leaf seeds the next public belief state. (b) A fixed budget solves every subgame to the same CFR maturity, so the value errors along the chain share a common component and their cross-depth correlation $\bar{\rho}$ is high; SHA draws each T_d from a truncated Gaussian and lowers $\bar{\rho}$ (both values measured in Section 6). (c) Variance of the summed error, $Dv(1 + (D-1)\rho)$ under Theorem 5.5, at the two measured correlations.

The change to a production ReBeL trainer is a single line, replacing the constant budget with a truncated Gaussian draw. Our experiments use $N = 1024$ and $\sigma = 64$, chosen so that $\sigma/N = 1/16$ and the truncation rate at the bounds stays below one in a thousand. The marginal cost over fixed-budget ReBeL is essentially zero. The expected budget is $N - \frac{1}{2}$ iterations and the draw costs microseconds. No forward or backward pass through the value network is added, and the wall-clock time per epoch differs by less than 1% between SHA and the baseline, well inside timing noise.

5 Theoretical Analysis

The section proves three results. The first two bound the cost of SHA, through its convergence rate and the variance it adds to a single target. The third gives the depth dependence of the benefit. Throughout, a subgame is a finite two-player zero-sum extensive-form game with perfect recall whose leaf values are produced by θ_v and held fixed for the duration of the solve. $V^\star_\beta$ is the Nash value of the surrogate, $V_\beta(H)$ is the linearly weighted target after H iterations of linear CFR, $\bar{\pi}^H$ is the realization-equivalent linearly weighted average strategy of Section 3, and $H = \lfloor T \rfloor \ge 1$ is

the realized budget. Utilities lie in an interval of length Δ, $|\mathcal{I}|$ counts the information sets of both players, and $|\mathcal{A}|$ is the largest action count at any of them. Var_T denotes variance over the budget draw with the subgame and θ_v fixed.

Lemma 5.1 (Value-target error bound)**.** *After H iterations of linear CFR,*

$$\mathrm{expl}(\bar{\pi}^H) \le \frac{K_\beta}{\sqrt{H}} \qquad \textit{and} \qquad |V_\beta(H) - V_\beta^\star| \le \frac{K_\beta}{\sqrt{H}}, \qquad K_\beta = \tfrac{2}{\sqrt{3}}\,\Delta\,|\mathcal{I}|\sqrt{|\mathcal{A}|}. \tag{3}$$

Proof. Write $w_t = t$ and $W_H = \sum_{t\le H} w_t = H(H+1)/2$. Linear CFR is regret matching run on the weighted instantaneous regrets $w_t r^t(I,a)$ [Brown and Sandholm, 2019a]. The potential argument for regret matching applies to any positive weights, since the regret-matching strategy makes the cross term $\sum_a (\sum_{s<t} w_s r^s(I,a))_+ w_t r^t(I,a)$ vanish, and gives $\max_a \sum_{t\le H} w_t r^t(I,a) \le \Delta\sqrt{|\mathcal{A}| \sum_{t\le H} w_t^2}$ at every information set; the counterfactual-regret decomposition [Zinkevich et al., 2007] sums these into a weighted external regret $R_i^{w,H} \le \Delta|\mathcal{I}_i|\sqrt{|\mathcal{A}| \sum_{t\le H} t^2}$ for each player i. Since $\sum_{t\le H} t^2 = H(H+1)(2H+1)/6$, we have $\sqrt{\sum_{t\le H} t^2}/W_H = 2\sqrt{(2H+1)/(6H(H+1))} \le \frac{2}{\sqrt{3}} H^{-1/2}$, so $R_i^{w,H}/W_H \le \frac{2}{\sqrt{3}}\Delta|\mathcal{I}_i|\sqrt{|\mathcal{A}|}\,H^{-1/2}$. In a two-player zero-sum game the weighted average profile has exploitability at most $(R_1^{w,H} + R_2^{w,H})/W_H$, which gives the first bound because $|\mathcal{I}_1|+|\mathcal{I}_2| = |\mathcal{I}|$. For the second, let $(\pi_1^\star, \pi_2^\star)$ be a Nash equilibrium. The external-regret bound against the fixed strategy $\pi_1^\star$ reads $\sum_t w_t u_1(\pi_1^\star, \pi_2^t) - \sum_t w_t u_1(\pi^t) \le R_1^{w,H}$, and $u_1(\pi_1^\star, \pi_2) \ge V_\beta^\star$ for every π_2, so $V_\beta(H) = \sum_t w_t u_1(\pi^t)/W_H \ge V_\beta^\star - R_1^{w,H}/W_H$. The symmetric argument for player 2 gives $V_\beta(H) \le V_\beta^\star + R_2^{w,H}/W_H$. □

The two theorems below average over the budget draw, so we record the moments of the truncated Gaussian first.

Lemma 5.2 (Moments of the truncated Gaussian)**.** *Let $T \sim \mathcal{N}(N,\sigma^2)$ truncated to $[N/2, 3N/2]$ with $\sigma > 0$, and $G \sim \mathcal{N}(0,1)$. Then (1) $T - N$ is symmetric about 0, so $\mathbb{E}[T] = N$ and every odd central moment of T vanishes; (2) $\mathbb{E}[|T-N|^k] \le \sigma^k\,\mathbb{E}[|G|^k]$ for every $k \ge 1$, hence $\mathrm{Var}[T] \le \sigma^2$ and $\mathbb{E}[|T-N|^3] \le 2\sqrt{2/\pi}\,\sigma^3$; and (3) for integer N, $\mathbb{E}[\lfloor T \rfloor] = N - \frac{1}{2}$.*

Proof. (1) The parent density and the truncation interval are both symmetric about N. (2) Let $Z = (T-N)/\sigma$, which is G conditioned on $|G| \le a$ with $a = N/(2\sigma)$. For $0 \le x \le a$, $\Pr(|Z| > x) = \big(\Pr(|G| > x) - \Pr(|G| > a)\big)/\Pr(|G| \le a) \le \Pr(|G| > x)$, because $\Pr(|G| > x) \le 1$ and $\Pr(|G| > a) \ge \Pr(|G| > x)\Pr(|G| > a)$; for $x > a$ the left side is 0. So $|Z|$ is stochastically smaller than $|G|$, $\mathbb{E}|Z|^k \le \mathbb{E}|G|^k$, and the constants are $\mathbb{E}[G^2] = 1$ and $\mathbb{E}|G|^3 = 2\sqrt{2/\pi}$. (3) For non-integer x, $\lfloor N + x \rfloor + \lfloor N - x \rfloor = 2N - 1$, and the law of $T - N$ is symmetric and atomless, so pairing x with $-x$ gives $\mathbb{E}[\lfloor T \rfloor] = N - \frac{1}{2}$. □

Theorem 5.3 (Convergence rate preservation)**.** *Let $T \sim \mathcal{N}(N,\sigma^2)$ truncated to $[N/2, 3N/2]$, $H = \lfloor T \rfloor$, and $N \ge 4$. Then*

$$\mathbb{E}_T\big[\mathrm{expl}(\bar{\pi}^H)\big] \le \frac{K_\beta}{\sqrt{N}}\Big(1 + \tfrac{1}{2}\big(\tfrac{\sigma}{N}\big)^2\Big)\Big(1 + \tfrac{2}{N}\Big). \tag{4}$$

For $\sigma \le N/8$ and $N \ge 1024$ the factor is at most 1.01, so SHA preserves the $O(N^{-1/2})$ rate of fixed-budget linear CFR.

Proof. Lemma 5.1 gives $\mathrm{expl}(\bar{\pi}^H) \le K_\beta H^{-1/2}$ pointwise, so it suffices to bound $\mathbb{E}_T[H^{-1/2}]$. Rounding first: $H \ge T - 1 \ge T(1 - 2/N)$ because $T \ge N/2$, and $(1-x)^{-1/2} \le 1 + x$ on $[0, \frac{1}{2}]$, where the difference is concave, vanishes at 0, and equals $1.5 - \sqrt{2} > 0$ at $\frac{1}{2}$, so $H^{-1/2} \le T^{-1/2}(1 + 2/N)$. For the continuous part write $T = N(1+u)$ with $|u| \le \frac{1}{2}$. By Lemma 5.2, u is symmetric about 0 and $\mathbb{E}[u^2] \le (\sigma/N)^2$. The binomial series $(1+u)^{-1/2} = \sum_{k\ge 0} c_k u^k$ with $c_k = (-1)^k\binom{2k}{k}4^{-k}$ converges uniformly on $|u| \le \frac{1}{2}$ (its terms are dominated by 2^{-k}), so expectation can be taken termwise. Odd terms vanish by symmetry; the even coefficients are positive

and decreasing, with $c_2 = 3/8$ and $c_{2k} \le 35/128$ for $k \ge 2$; and $u^{2k} \le 4^{-(k-1)}u^2$ on the support. Hence

$$\mathbb{E}_T\big[(1+u)^{-1/2}\big] \le 1 + \Big(\tfrac{3}{8} + \tfrac{35}{128}\sum_{k\ge 2} 4^{-(k-1)}\Big)\mathbb{E}[u^2] = 1 + \tfrac{179}{384}\,\mathbb{E}[u^2] \le 1 + \tfrac{1}{2}\big(\tfrac{\sigma}{N}\big)^2. \quad (5)$$

Multiplying the two factors proves the bound; at $\sigma/N = 1/8$ and $N = 1024$ it is $(1+\frac{1}{128})(1+\frac{1}{512}) = 66177/65536 < 1.01$. □

The variance that SHA adds to a single target depends on how fast the target responds to the budget. The error envelope of Lemma 5.1 fixes the width of the error, and the speed at which the realized target moves inside that envelope enters as a separate property. We state it as an assumption.

Assumption 1 (Regular budget response)**.** *There is a constant L_β such that $|V_\beta(H) - V_\beta(H')| \le L_\beta\,|H - H'|\,N^{-3/2}$ for all integers $H, H' \in [N/2 - 1,\ 3N/2]$.*

The scale $N^{-3/2}$ is that of the slope of the envelope $K_\beta/\sqrt{H}$ on this interval. The assumption therefore says the target varies with the budget no faster than a constant multiple of the rate at which its error bound shrinks. It places no restriction on the sign or size of the error itself, and L_β is a property of the subgame and network at the budget scale N under study. Without it, the value recurrence alone gives $|V_\beta(h+1) - V_\beta(h)| = \frac{2}{h+2}|\Delta_{h+1} - V_\beta(h)| \le 4\Delta/N$ on the interval, which is the assumption with $L_\beta = 4\Delta\sqrt{N}$.

Theorem 5.4 (Per-subgame target variance)**.** *Under the setting of Theorem 5.3 and Assumption 1,*

$$\mathrm{Var}_T[V_\beta(H)] \le \frac{L_\beta^2(\sigma+1)^2}{N^3}. \quad (6)$$

Proof. For an L-Lipschitz function f and independent copies X, X', $\mathrm{Var}[f(X)] = \frac{1}{2}\mathbb{E}[(f(X) - f(X'))^2] \le \frac{1}{2}L^2\mathbb{E}[(X-X')^2] = L^2\mathrm{Var}[X]$. Apply this with $f(H) = V_\beta(H)$ and $L = L_\beta N^{-3/2}$. Since $H = T - \varphi$ with $\varphi \in [0,1)$, the standard deviation of H is at most that of T plus $\frac{1}{2}$, and $\mathrm{Var}[T] \le \sigma^2$ by Lemma 5.2, so $\mathrm{Var}[H] \le (\sigma+1)^2$. □

Under a fixed budget this variance is zero, so Theorem 5.4 is the price of randomization. Without Assumption 1, the recurrence bound above gives $\mathrm{Var}_T[V_\beta(H)] \le 16\Delta^2(\sigma+1)^2/N^2$. The bound of Theorem 5.4 gains a factor of order N over it when L_β/Δ stays bounded as N varies. Relative to the squared error envelope K_β^2/N of a single solve the conditional bound carries the factor $(L_\beta/K_\beta)^2(\sigma+1)^2/N^2$, about $4\times10^{-3}(L_\beta/K_\beta)^2$ at $N = 1024$, $\sigma = 64$.

The third result concerns how per-subgame errors accumulate along the recursion chain of Figure 1(a). It rests on a model of the cross-depth error correlation, stated here and measured on trained networks in Section 6.

Assumption 2 (Cross-depth error correlation)**.** *Fix a self-play chain of D nested subgames and let η_d be the local error that the solve at depth d contributes to the root target: the depth-d target computed with θ_v at the leaves minus the same solve with the leaves replaced by their true continuation values, centered over training runs so that $\mathbb{E}[\eta_d] = 0$; its randomness comes from the training run through the shared parameters of θ_v. (a) The centered root-level error is $\sum_{d=1}^{D}\eta_d$. (b) The η_d have a common variance $v > 0$ and a common pairwise correlation $\rho \in [0,1)$. We write $(v_\mathrm{b}, \rho_\mathrm{b})$ under the fixed budget and $(v_\mathrm{s}, \rho_\mathrm{s})$ under SHA, with $\rho_\mathrm{s} < \rho_\mathrm{b}$.*

Part (a) models the root error as the unweighted sum of local contributions along the sampled chain. Reach weighting and bias stay outside the model, so the theorem below concerns variance. Part (b) is the decomposition $\eta_d = \bar\eta + \zeta_d$ into a shared component and idiosyncratic components that are pairwise uncorrelated, of common variance v_ζ, and uncorrelated with $\bar\eta$, which corresponds to $\rho = \bar v/(\bar v + v_\zeta)$. The ordering $\rho_\mathrm{s} < \rho_\mathrm{b}$ is a hypothesis about training, and Section 6 measures $\rho_\mathrm{b} = 0.749$ and $\rho_\mathrm{s} = 0.373$. Under a fixed budget every subgame is solved to the same CFR maturity, so the network fits one maturity-specific error. SHA trains it on a mixture of maturities. Shared weights induce correlation under both methods, and the measured ρ_s stays positive.

Theorem 5.5 (Recursive variance scaling)**.** *Under Assumption 2, the root-level target variance is*

$$\mathrm{Var}\Big[\sum_{d=1}^{D}\eta_d\Big] = D\,v\,\big(1+(D-1)\rho\big). \tag{7}$$

For fixed (v,ρ) this grows as $\Theta(D^2)$ when $\rho>0$ and as $\Theta(D)$ when $\rho=0$; lowering ρ lowers the coefficient of the quadratic term.

Proof. The sum has D variances equal to v and $D(D-1)$ covariances equal to ρv. □

Corollary 5.6 (Depth-dependent variance ratio)**.** *Under Assumption 2, the SHA-to-baseline variance ratio*

$$r(D) = \frac{v_{\mathrm{s}}}{v_{\mathrm{b}}}\cdot\frac{1+(D-1)\rho_{\mathrm{s}}}{1+(D-1)\rho_{\mathrm{b}}} \tag{8}$$

is strictly decreasing in D for fixed $(v_{\mathrm{s}}/v_{\mathrm{b}},\rho_{\mathrm{s}},\rho_{\mathrm{b}})$, from $v_{\mathrm{s}}/v_{\mathrm{b}}$ at $D=1$ to $v_{\mathrm{s}}\rho_{\mathrm{s}}/(v_{\mathrm{b}}\rho_{\mathrm{b}})$ as $D\to\infty$; it is $\Theta(1/D)$ exactly when $\rho_{\mathrm{s}}=0$. SHA lowers the root variance at depth D exactly when $r(D)<1$, which holds at every depth if $v_{\mathrm{s}}<v_{\mathrm{b}}$ and at every depth $D>1$ if $v_{\mathrm{s}}=v_{\mathrm{b}}$. For fixed model parameters the relative advantage of SHA grows with the chain length.

Proof. The derivative of $(1+(D-1)\rho_{\mathrm{s}})/(1+(D-1)\rho_{\mathrm{b}})$ in D has the sign of $\rho_{\mathrm{s}}-\rho_{\mathrm{b}}<0$; the limits are immediate. □

Discussion. The main result is Corollary 5.6. For fixed model parameters the variance ratio falls monotonically with the recursion depth. Its depth profile relative to $D=1$ is fixed by the two correlations, which Section 6 measures, and its level by the per-depth variance ratio $v_{\mathrm{s}}/v_{\mathrm{b}}$. Theorems 5.3 and 5.4 use only the symmetry, the support $[N/2,3N/2]$, and the second moment of the budget distribution (Lemma 5.2), so any symmetric distribution on that interval with variance at most σ^2 inherits them.

Scope. The analysis assumes a roughly constant per-iteration cost, as in ReBeL-style CFR, so a budget above N costs proportionally more. Where per-iteration cost grows super-linearly, the budget distribution has to allow for it. Leaf values are held frozen during a solve, and the implementation re-queries θ_v as the leaf beliefs move.

6 Experiments

The analysis makes three testable predictions. Under SHA the variance across training runs falls and the reduction grows with recursion depth (Corollary 5.6), the cross-depth error correlation is lower (Assumption 2), and the gain grows with the base budget. The paragraphs below test them in turn.

Setup. We evaluate on five games. Liar's Dice 1×4f, 1×5f, and 1×6f use one die per player with 4, 5, or 6 faces, in the setting of ReBeL [Brown et al., 2020]. Leduc-30 and Leduc-40 are the stack-limited Leduc poker variants of Li et al. [2020], with public depth 17 for Leduc-40 counting chance edges, against 9 to 13 for the Liar's Dice variants. Every subgame uses lookahead depth 2 and $N=1024$ linear-CFR iterations, and SHA uses $\sigma=64$ with truncation $[512,1536]$ unless noted. The value network and optimizer follow the ReBeL reference implementation (Appendix A). The network has two hidden layers of 256 units with LayerNorm and GELU. Training uses Adam at 3×10^{-4} halved twice, Huber loss, gradient clipping at 5, batch size 512, and a replay buffer of 2×10^6 targets. Liar's Dice runs use 30 seeds per configuration and 2.5×10^7 samples per run. Leduc runs use 20 seeds per configuration and 1×10^7 samples per run. We report across-seed means with 95% confidence intervals and compare arms with Welch's t-test and a two-sided F-test. All experiments ran on a single server with four NVIDIA A100 GPUs.

Table 1: Final exploitability on Liar's Dice at epoch 100. ReBeL and SHA use 30 seeds, Deep CFR 3.

Game	ReBeL mean$\pm$CI$_{95}$	SHA mean$\pm$CI$_{95}$	Deep CFR mean$\pm$CI$_{95}$
1×4f	0.0167±0.0014	0.0155±0.0009	0.101±0.037
1×5f	0.0150±0.0013	0.0129±0.0006	0.148±0.006
1×6f	0.0238±0.0020	0.0197±0.0008	0.253±0.034

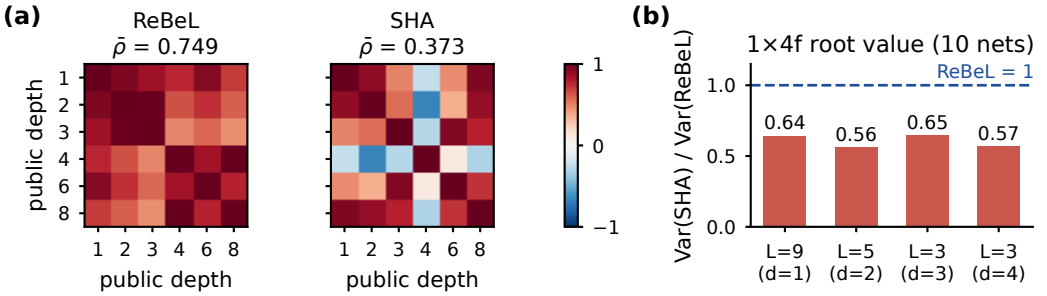


Figure 2: Test of Assumption 2 on trained value networks. (a) Across-run correlation of the value errors of public-depth groups on Leduc-40 ($\bar{\rho}$ is the mean off-diagonal entry). (b) Across-run variance of the root value in nested solves of chain length L on 1×4f, SHA relative to ReBeL.

Main results. Table 1 reports final exploitability. The primary effect is on the spread. At epoch 100 the across-seed interval narrows by 34% on 1×4f, 52% on 1×5f, and 61% on 1×6f. The mean also ends lower in all three variants, by 7%, 14%, and 17%. Both effects grow with the size of the variant. Larger variants have deeper public trees and longer recursion chains, the ordering in which Corollary 5.6 predicts the variance ratio to fall. At $N = 1024$ the variance SHA injects at a single subgame is tiny, and the gain comes from the recursion chain, where a lower cross-depth correlation shrinks the quadratic term. As a search-free reference, Deep CFR [Brown et al., 2019] trails the recursive methods by roughly an order of magnitude on the same games under the same exact evaluation. It uses external-sampling traversals, with 200 iterations of 300 traversals per player. The advantage networks are retrained from scratch for 1000 steps per iteration, and the final policy network is fit for 2000 steps with an MLP of the value network's size.

The cross-depth correlation assumption. Assumption 2 posits that fixed-budget training produces value errors that are correlated across recursion depths and that SHA lowers this correlation. We test the claim on trained networks in Leduc-40, whose public tree supports recursion chains of up to nine subgames. Each method contributes 4 independent runs under identical budgets, probed at 117 fixed public belief states drawn at public depths $\{1, 2, 3, 4, 6, 8\}$ with Dirichlet-random beliefs for both players. Ground truth is the value of the probed subgame solved to the end of the game with 8,192 iterations of linear CFR. The signed error of a network at a probe is its prediction minus this value.

Averaging the signed value errors within each public-depth group gives one error per depth per run, and correlating these across the 4 runs of each method yields the matrices of Figure 2(a). Under fixed-budget ReBeL the errors at different depths move together, with a mean off-diagonal correlation of 0.749 and a variance-coupling ratio $\mathrm{Var}[\sum_d e_d] / \sum_d \mathrm{Var}[e_d]$ of 3.17. Under SHA the correlation falls to 0.373 and the coupling ratio to 2.50. The ordering the assumption posits appears in runs that share a network, replay buffer, and optimizer. The residual correlation under SHA is the shared-weight component.

The same mechanism should show up in the across-run variance of a root value assembled through nested solves, which Figure 2(b) measures on 1×4f. Each of 10 trained networks per method serves as the leaf evaluator in a nested solve of the full game from the root. Lookahead depths $d = 1, 2, 3, 4$ with 1,024 iterations per solve give chains of length $L = \lceil \text{public depth}/d \rceil \in \{3, 5, 9\}$. We report the across-network variance of the root value, averaged over private hands. SHA's variance is 0.56 to 0.65 times ReBeL's at every chain length. Corollary 5.6 evaluated at the measured correlations, with equal per-depth variance under both methods, predicts ratios of 0.70, 0.62, and 0.57 at $L = 3, 5, 9$ and a limit of 0.50. The measured ratios lie between these predictions and the limit and stay flat in L, the profile Corollary 5.6 predicts for a positive residual correlation ρ_s. We pool the Leduc-40 probes

Table 2: Exact exploitability on Leduc-30 and Leduc-40 at the matched budget, $n = 20$ runs per arm.

Game	**ReBeL mean$\pm$CI$_{95}$**	**SHA mean$\pm$CI$_{95}$**
Leduc-30	0.228$\pm$0.036	0.242$\pm$0.021
Leduc-40	0.563$\pm$0.100	0.432$\pm$0.047

Table 3: Annealing strength across game sizes at $N = 1024$ (30 seeds).

Game	$\sigma = 0$	16	32	64	128	256
1×4f	0.0167	0.0165	0.0160	**0.0155**	0.0158	0.0167
1×5f	0.0150	0.0141	0.0134	**0.0129**	0.0131	0.0140
1×6f	0.0238	0.0224	0.0210	0.0197	**0.0195**	0.0213

Table 4: Base-budget ablation on 1×5f (30 seeds).

N	ReBeL	SHA	Mean↓	CI$_{95}$ ↓
256	0.0248	0.0228	8%	36%
512	0.0188	0.0166	12%	44%
1024	0.0150	0.0129	14%	52%
2048	0.0124	0.0103	17%	59%

by the number of nested depth-2 solves to the end, $L = 3$ to 9, dropping those adjacent to terminal nodes, where the ReBeL variance vanishes. The pooled variance ratio is 0.54, with per-length ratios of 0.36 to 1.25. No training is involved in this stage, so the reduction reflects the lower correlation of the recursive errors themselves.

Deeper public-belief-state domains. Liar's Dice has a shallow public tree, with recursion depths of 1.6 to 2.0. Leduc-30 and Leduc-40 add a different information structure and recursion depths of 3.2 to 3.4, testing Corollary 5.6 beyond the Liar's Dice range. Table 2 compares exploitability at the matched budget.

On Leduc-30 the means are statistically indistinguishable, while SHA cuts the across-run standard deviation by 40%. On Leduc-40 SHA lowers the mean by 23% and the standard deviation by 53%. Both variance reductions are significant at the 0.05 level under a two-sided F-test, and the Leduc-40 mean under Welch's test ($p = 0.02$). Variance reduction is the consistent effect, as Corollary 5.6 predicts; the mean gain appears only in the larger game.

Ablations. Figure 3 plots the 1×5f row of Table 3: exploitability falls from the fixed-budget baseline at $\sigma = 0$ to a broad minimum between $\sigma = 32$ and $\sigma = 128$ and rises again at $\sigma = 256$. Every value between $N/30$ and $N/8$ is near-optimal (Theorem 5.3). The optimum moves from $\sigma = 64$ on 1×4f to $\sigma = 128$ on 1×6f, and $\sigma = 64$ stays within 5% of the per-game best everywhere. Table 4 varies the base budget at fixed $\sigma = N/16$: the mean reduction grows from 8% at $N = 256$ to 17% at $N = 2048$ and the interval reduction from 36% to 59%.

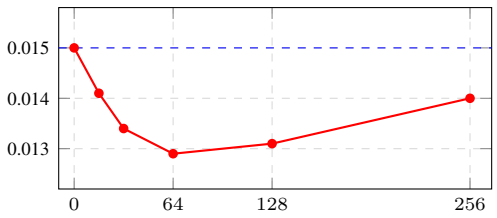


Figure 3: Exploitability versus σ on 1×5f (dashed: ReBeL).

7 Conclusion

Stochastic Horizon Annealing is a one-line, cost-neutral change to ReBeL that randomizes the last static degree of freedom in depth-limited solving. It preserves the convergence rate of linear CFR and keeps the variance it adds to each subgame target a small fraction of the squared solve error. Under a cross-depth error-correlation model tested on trained networks, it lowers the recursive target variance by a factor that decreases monotonically with depth. On Liar's Dice it narrows the across-seed interval of final exploitability by 34% to 61%, with the largest reduction on the largest variant, and lowers the mean by 7% to 17%. Probing shows the cross-depth correlation halved under SHA, and on Leduc-30 and Leduc-40 it cuts the across-run standard deviation by 40% and 53%. To our knowledge, SHA is the strongest depth-limited method for solving imperfect-information games.

Acknowledgments and Disclosure of Funding

This work was supported in part by the NeurIPS 2026 Financial Aid program.

References

Michael Bowling, Neil Burch, Michael Johanson, and Oskari Tammelin. Heads-up limit hold'em poker is solved. *Science*, 347(6218):145-149, 2015.

Noam Brown and Tuomas Sandholm. Safe and nested subgame solving for imperfect-information games. *Advances in neural information processing systems*, 30, 2017.

Noam Brown and Tuomas Sandholm. Superhuman ai for heads-up no-limit poker: Libratus beats top professionals. *Science*, 359(6374):418-424, 2018.

Noam Brown and Tuomas Sandholm. Solving imperfect-information games via discounted regret minimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 1829-1836, 2019a.

Noam Brown and Tuomas Sandholm. Superhuman ai for multiplayer poker. *Science*, 365(6456): 885-890, 2019b.

Noam Brown, Sam Ganzfried, and Tuomas Sandholm. Hierarchical abstraction, distributed equilibrium computation, and post-processing, with application to a champion no-limit texas hold'em agent. In *AAAI Workshop: Computer Poker and Imperfect Information*, volume 15, page 07, 2015.

Noam Brown, Tuomas Sandholm, and Brandon Amos. Depth-limited solving for imperfect-information games. *Advances in neural information processing systems*, 31, 2018.

Noam Brown, Adam Lerer, Sam Gross, and Tuomas Sandholm. Deep counterfactual regret minimization. In *International conference on machine learning*, pages 793-802. PMLR, 2019.

Noam Brown, Anton Bakhtin, Adam Lerer, and Qucheng Gong. Combining deep reinforcement learning and search for imperfect-information games. *Advances in neural information processing systems*, 33:17057-17069, 2020.

Neil Burch, Michael Johanson, and Michael Bowling. Solving imperfect information games using decomposition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 28, 2014.

Yang Cai, Gabriele Farina, Julien Grand-Clément, Christian Kroer, Chung-Wei Lee, Haipeng Luo, and Weiqiang Zheng. Last-iterate convergence properties of regret-matching algorithms in games. In *The Thirteenth International Conference on Learning Representations*, 2025.

John C Duchi, Peter L Bartlett, and Martin J Wainwright. Randomized smoothing for stochastic optimization. *SIAM Journal on Optimization*, 22(2):674-701, 2012.

Zhiyuan Fan and Gabriele Farina. Gae falls short in imperfect-information self-play reinforcement learning. *arXiv preprint arXiv:2605.19235*, 2026.

Meire Fortunato, Mohammad Gheshlaghi Azar, Bilal Piot, Jacob Menick, Ian Osband, Alex Graves, Vlad Mnih, Remi Munos, Demis Hassabis, Olivier Pietquin, et al. Noisy networks for exploration. In *International Conference on Learning Representations*, 2018.

Yanchang Fu, Shengda Liu, Pei Xu, and Kaiqi Huang. No-regret strategy solving in imperfect-information games via pre-trained embedding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 16906-16913, 2026.

Gao Huang, Yu Sun, Zhuang Liu, Daniel Sedra, and Kilian Q Weinberger. Deep networks with stochastic depth. In *European Conference on Computer Vision*, pages 646-661. Springer, 2016.

Michael Johanson, Nolan Bard, Neil Burch, and Michael Bowling. Finding optimal abstract strategies in extensive-form games. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 26, pages 1371-1379, 2012.

Fivos Kalogiannis and Gabriele Farina. Policy gradient methods converge globally in imperfect-information extensive-form games. *Advances in Neural Information Processing Systems*, 38: 29162-29224, 2026.

Vojtěch Kovařík, Martin Schmid, Neil Burch, Michael Bowling, and Viliam Lisý. Rethinking formal models of partially observable multiagent decision making. *Artificial Intelligence*, 303:103645, 2022.

Vojtěch Kovařík, Dominik Seitz, Viliam Lisý, Jan Rudolf, Shuo Sun, and Karel Ha. Value functions for depth-limited solving in zero-sum imperfect-information games. *Artificial Intelligence*, 314: 103805, 2023.

Vojtech Kovarik, Caspar Oesterheld, and Vincent Conitzer. Recursive joint simulation in games. *Synthese*, 208(2):67, 2026.

Ondřej Kubíček and Viliam Lisý. Look-ahead reasoning with a learned model in imperfect information games. In *The Fourteenth International Conference on Learning Representations*, 2026. URL `https://openreview.net/forum?id=NnBbr4hI8a`.

Ondřej Kubíček, Viliam Lisý, and Tuomas Sandholm. Equilibrium refinements improve subgame solving in imperfect-information games. In *IJCAI/ECAI*, pages 3452-3460. ijcai.org, 2026a.

Ondřej Kubíček, Viliam Lisý, and Tuomas Sandholm. Test-time reinforcement learning in imperfect information games, 2026b. URL `https://arxiv.org/abs/2608.30635`.

Marc Lanctot, Kevin Waugh, Martin Zinkevich, and Michael Bowling. Monte carlo sampling for regret minimization in extensive games. *Advances in neural information processing systems*, 22, 2009.

JB Lanier, Nathan Monette, Pierre Baldi, and Roy Fox. Data-augmented game starts for accelerating self-play exploration in imperfect information games. *arXiv preprint arXiv:2605.14379*, 2026.

Boning Li and Longbo Huang. Efficient online pruning and abstraction for imperfect information extensive-form games. In *The Thirteenth International Conference on Learning Representations*, 2025.

Boning Li and Longbo Huang. Abstraction agent: Llm-guided abstraction for imperfect-information games. In *Findings of the Empirical Methods in Natural Language Processing*, 2026a.

Boning Li and Longbo Huang. Agents that certify their own exploits: Confidence-scheduled restricted responses for safe opponent exploitation. *arXiv preprint arXiv:2607.28520*, 2026b.

Boning Li and Longbo Huang. Why alternation speeds up counterfactual regret minimization? *arXiv preprint arXiv:2610.21970*, 2026c.

Boning Li and Longbo Huang. Effective, efficient, and general information abstraction for imperfect-information extensive-form games. *arXiv preprint arXiv:2605.10900*, 2026d.

Boning Li and Longbo Huang. Gpu-cfr: 80x faster counterfactual regret minimization by compiling the game to static dataflow and cuda graph replay. *arXiv preprint arXiv:2609.11923*, 2026e.

Boning Li and Longbo Huang. Icm out! better tournament strategy from computed continuations, vs. solvers and llms. *arXiv preprint arXiv:2608.09586*, 2026f.

Boning Li and Longbo Huang. Why last-iterate scale-invariant regret matching converges linearly? *arXiv preprint arXiv:2609.38162*, 2026g.

Boning Li and Longbo Huang. Real-time parallel counterfactual regret minimization. *arXiv preprint arXiv:2605.19928*, 2026h.

Boning Li and Longbo Huang. Turborebel: 250× accelerated belief learning for large imperfect-information extensive-form games, 2026i. URL `https://openreview.net/forum?id=yMo7Z670f6`.

Boning Li and Longbo Huang. When are we near nash? certifying solutions in general-sum games. *arXiv preprint arXiv:2610.29300*, 2026j.

Boning Li, Zhixuan Fang, and Longbo Huang. Rl-cfr: improving action abstraction for imperfect information extensive-form games with reinforcement learning. *Proceedings of the Forty-first International Conference on Machine Learning*, 2024.

Boning Li, Yu Chen, and Longbo Huang. Av-aivat: 74x cheaper agent evaluation with certified anytime-valid stopping in imperfect-information games. *arXiv preprint arXiv:2608.06362*, 2026a.

Boning Li, Yu Chen, and Longbo Huang. Correlated chance sampling for monte carlo counterfactual regret minimization. *arXiv preprint arXiv:2607.27035*, 2026b.

Boning Li, Baoxiang Wang, and Longbo Huang. Pokerskill: Llms can play expert-level poker without training or solvers. *arXiv preprint arXiv:2605.30094*, 2026c.

Hui Li, Kailiang Hu, Zhibang Ge, Tao Jiang, Yuan Qi, and Le Song. Double neural counterfactual regret minimization. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=lvWE53uu9dH.

Zun Li, Marc Lanctot, Kevin R McKee, Luke Marris, Ian Gemp, Daniel Hennes, Paul Muller, Kate Larson, Yoram Bachrach, and Michael P Wellman. Combining tree-search, generative models, and nash bargaining concepts in game-theoretic reinforcement learning. *arXiv preprint arXiv:2302.00797*, 2023.

Zun Li, Marc Lanctot, Kevin R McKee, Luke Marris, Ian Gemp, Daniel Hennes, Paul Muller, Kate Larson, Yoram Bachrach, and Michael P Wellman. Combining deep reinforcement learning and search with generative models for game-theoretic opponent modeling. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, pages 161-169, 2025.

Mingyang Liu, Gabriele Farina, and Asuman E. Ozdaglar. A policy-gradient approach to solving imperfect-information games with best-iterate convergence. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=ZW4MRZrmSA.

Runyu Lu, Yuanheng Zhu, and Dongbin Zhao. Divergence-regularized discounted aggregation: Equilibrium finding in multiplayer partially observable stochastic games. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=KD5nJUgeW4.

Wataru Masaka, Mitsuki Sakamoto, Kenshi Abe, Kaito Ariu, Tuomas Sandholm, and Atsushi Iwasaki. On the power of perturbation under sampling in solving extensive-form games, 2025. URL https://arxiv.org/abs/2501.16600.

Stephen Marcus McAleer, Gabriele Farina, Marc Lanctot, and Tuomas Sandholm. Escher: Eschewing importance sampling in games by computing a history value function to estimate regret. In *The Eleventh International Conference on Learning Representations*, 2023.

Linjian Meng, Wubing Chen, Wenbin Li, Tianpei Yang, Youzhi Zhang, and Yang Gao. Reducing variance of stochastic optimization for approximating nash equilibria in normal-form games. In *Forty-second International Conference on Machine Learning*, 2025a.

Linjian Meng, Tianpei Yang, Youzhi Zhang, Zhenxing Ge, Shangdong Yang, Tianyu Ding, Wenbin Li, Bo An, and Yang Gao. Efficient last-iterate convergence in solving extensive-form games. In *Advances in Neural Information Processing Systems*, 2025b. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/918c67cd7910578f1637beb56717f455-Abstract-Conference.html.

Linjian Meng, Youzhi Zhang, Zhenxing Ge, Tianyu Ding, Shangdong Yang, Zheng Xu, Wenbin Li, and Yang Gao. Last-iterate convergence of smooth regret matching$^+$ variants in learning nash equilibria. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025c. URL https://openreview.net/forum?id=v771lscnlS.

Linjian Meng, Tianpei Yang, Youzhi Zhang, Zhenxing Ge, and Yang Gao. Faster game solving via asymmetry of step sizes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 17161-17169, 2026a.

Linjian Meng, Youzhi Zhang, Shangdong Yang, Wenbin Li, Tianyu Ding, and Yang Gao. A faster parameter-free regret matching algorithm. In *The Fourteenth International Conference on Learning Representations*, 2026b. URL https://openreview.net/forum?id=JLllvi7dsg.

David Milec, Vojtěch Kovařík, and Viliam Lisý. Adapting beyond the depth limit: Counter strategies in large imperfect information games. In *Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems*, pages 2675-2677, 2025.

Matej Moravčík, Martin Schmid, Neil Burch, Viliam Lisý, Dustin Morrill, Nolan Bard, Trevor Davis, Kevin Waugh, Michael Johanson, and Michael Bowling. Deepstack: Expert-level artificial intelligence in heads-up no-limit poker. *Science*, 356(6337):508-513, 2017.

Ashutosh Nayyar, Aditya Mahajan, and Demosthenis Teneketzis. Decentralized stochastic control with partial history sharing: A common information approach. *IEEE Trans. Autom. Control.*, 58 (7):1644-1658, 2013.

Julien Perolat, Bart De Vylder, Daniel Hennes, Eugene Tarassov, Florian Strub, Vincent de Boer, Paul Muller, Jerome T. Connor, Neil Burch, Thomas Anthony, Stephen McAleer, Romuald Elie, Sarah H. Cen, Zhe Wang, Audrunas Gruslys, Aleksandra Malysheva, Mina Khan, Sherjil Ozair, Finbarr Timbers, Toby Pohlen, Tom Eccles, Mark Rowland, Marc Lanctot, Jean-Baptiste Lespiau, Bilal Piot, Shayegan Omidshafiei, Edward Lockhart, Laurent Sifre, Nathalie Beauguerlange, Remi Munos, David Silver, Satinder Singh, Demis Hassabis, and Karl Tuyls. Mastering the game of stratego with model-free multiagent reinforcement learning. *Science*, 378(6623):990-996, 2022. doi: 10.1126/science.add4679. URL https://www.science.org/doi/abs/10.1126/science.add4679.

Matthias Plappert, Rein Houthooft, Prafulla Dhariwal, Szymon Sber, Richard Y Chen, Dario Amodei, Pieter Abbeel, and Marcin Andrychowicz. Parameter space noise for exploration. In *International Conference on Learning Representations*, 2018.

Max Rudolph, Nathan Lichtlé, Sobhan Mohammadpour, Alexandre M Bayen, J Zico Kolter, Amy Zhang, Gabriele Farina, Eugene Vinitsky, and Samuel Sokota. Reevaluating policy gradient methods for imperfect-information games. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=vClBDezZUo.

Martin Schmid, Matej Moravčík, Neil Burch, Rudolf Kadlec, Josh Davidson, Kevin Waugh, Nolan Bard, Finbarr Timbers, Marc Lanctot, G Zacharias Holland, Elnaz Davoodi, Alden Christianson, and Michael Bowling. Student of games: A unified learning algorithm for both perfect and imperfect information games. *Science Advances*, 9(46):eadg3256, 2023.

David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science*, 362(6419): 1140-1144, 2018.

Samuel Sokota, Eugene Vinitsky, Hengyuan Hu, Zhiyuan Fan, J Zico Kolter, and Gabriele Farina. Scalable decision-making for games of imperfect information. *Nature*, 658(8134):55-59, 2026.

Christopher Solinas, Radovan Haluška, Pavel Yanushonak, David Sychrovský, Michael Buro, Martin Schmid, Nathan R. Sturtevant, and Michael Bowling. Belief embedding tree search. In *Reinforcement Learning and Video Games Workshop @ RLC 2025*, 2025. URL https://openreview.net/forum?id=ejeaOWLqbo.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929-1958, 2014.

Eric Steinberger, Adam Lerer, and Noam Brown. Dream: Deep regret minimization with advantage baselines and model-free learning. *arXiv preprint arXiv:2006.10410*, 2020.

Michal Šustr, Martin Schmid, Matej Moravčík, Neil Burch, Marc Lanctot, and Michael Bowling. Sound algorithms in imperfect information games. In *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems*, pages 1674-1676, 2021.

David Sychrovský, Martin Schmid, Michal Sustr, and Michael Bowling. Meta-learning in self-play regret minimization. *CoRR*, abs/2504.18917, 2025.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2818-2826, 2016.

Li Wan, Matthew Zeiler, Sixin Zhang, Yann Le Cun, and Rob Fergus. Regularization of neural networks using dropconnect. In *International Conference on Machine Learning*, pages 1058-1066, 2013.

Hang Xu, Kai Li, Haobo Fu, Qiang Fu, Junliang Xing, and Jian Cheng. Dynamic discounted counterfactual regret minimization. In *The Twelfth International Conference on Learning Representations*, 2024.

Hang Xu, Kai Li, Haobo Fu, Qiang Fu, Junliang Xing, and Jian Cheng. Deep (predictive) discounted counterfactual regret minimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 17284-17292, 2026.

Zelai Xu, Wanjun Gu, Chao Yu, Yi Wu, and Yu Wang. Learning strategic language agents in the werewolf game with iterative latent space policy optimization. In *Forty-second International Conference on Machine Learning*, 2025. URL `https://openreview.net/forum?id=N2m0BiSqhc`.

Weijun Zeng, Yinghao Li, Xiaosi Chen, Zijie Chang, and Fei Ge. Gnn-rebel: Enhancing neural belief representations for imperfect-information games. In *2025 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 7574-7579. IEEE, 2025a.

Weijun Zeng, Yinghao Li, Xiaosi Chen, Zijie Chang, and Fei Ge. An investigation of subgame depth in rebel: Impact on convergence and performance in imperfect-information games. In *2025 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 7580-7585. IEEE, 2025b.

Brian Hu Zhang and Tuomas Sandholm. General search techniques without common knowledge for imperfect-information games, and application to superhuman fog of war chess. In *The Fourteenth International Conference on Learning Representations*, 2026. URL `https://openreview.net/forum?id=afaakBqkvb`.

Naifeng Zhang, Stephen Marcus McAleer, and Tuomas Sandholm. Faster game solving via hyperparameter schedules. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 17319-17326, 2026.

Martin Zinkevich, Michael Johanson, Michael Bowling, and Carmelo Piccione. Regret minimization in games with incomplete information. *Advances in neural information processing systems*, 20, 2007.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state three theoretical results and the empirical claims (34% to 61% narrower across-seed interval of final exploitability, with the mean 7% to 17% lower alongside, on Liar's Dice $1\times\{4,5,6\}$f with 30 seeds; reduced cross-depth error correlation on probed networks; transfer to Leduc-30 and Leduc-40), all supported by the experiments in Section 6. The per-subgame variance bound is stated under an explicit regularity assumption (Assumption 1) and the recursive variance theorem under an explicit correlation model (Assumption 2) that Section 6 tests directly.

   Guidelines:

- The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Section 5 contains an explicit scope paragraph for the analysis, covering the assumption of constant per-iteration cost in the inner solver, regimes where that assumption is violated, and the treatment of leaf values as frozen during a solve. Section 6 confines the empirical claims to Liar's Dice and two Leduc variants, with no multi-street poker or multi-player evaluation.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: All three theorems, the corollary, and the supporting lemma in Section 5 are stated with explicit assumptions (the game class, Assumption 1, and Assumption 2) and accompanied by complete proofs in the main text; the truncated-Gaussian moment bounds they use are proved there as well.

   Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Section 6 specifies all hyperparameters ($N = 1024$, $\sigma = 64$, truncation bounds, network architecture, optimizer settings, batch size, replay buffer size, total data budget, lookahead depth, evaluation protocol, seed counts for every experiment), and Appendix A gives the value-network architecture and reproduces the original ReBeL procedure; the Leduc-30/40 definition, the probing protocol, and the Deep CFR configuration are in Section 6.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The training code is released with the paper and builds on the open-source ReBeL codebase (MIT license). The release includes the one-line SHA modification, configuration files, and scripts for the main experiments.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: Section 6 and Appendix A provide the full experimental setting; the modification to ReBeL is a single line specified in Equation (2) and Algorithm 1.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: Liar's Dice results report mean $\pm$ 95% CI half-width over 30 random seeds, listed as mean$\pm$CI$_{95}$ in Table 1, which also gives the 3-seed Deep CFR reference. The Leduc comparisons use $n = 20$ runs per arm; Table 2 reports mean$\pm$CI$_{95}$ and the accompanying text gives the significance of the spread and mean differences; the ablation tables state their seed counts in their captions. The intervals are Student-t half-widths $t_{0.025,n-1} \cdot s/\sqrt{n}$ from the across-seed standard deviation s; Section 6 states that arms are compared with Welch's t-test and a two-sided F-test.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.

- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Section 6 states that all experiments were run on a single server with four NVIDIA A100 GPUs, and Section 4 reports the wall-clock cost difference between SHA and the baseline (under 1%).

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The work studies algorithmic improvements to depth-limited subgame solving for symbolic, fully synthetic Liar's Dice and Leduc poker games; no human subjects, no scraped data, no deployment risk. The research conforms with the NeurIPS Code of Ethics in every respect we identified.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [No]

Justification: The paper proposes a foundational algorithmic improvement for solving symbolic imperfect-information games; we identified no direct positive or negative societal impacts beyond those of equilibrium-finding methods generally.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [No]

Justification: No models or datasets with high misuse potential are released. The artifact is an algorithmic modification to the inner solver of ReBeL, instantiated on Liar's Dice and two Leduc variants.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: We cite the original ReBeL paper [Brown et al., 2020] and its open-source codebase (MIT license). The linear CFR variant [Brown and Sandholm, 2019a] and all noise-injection prior work invoked in Section 2 are properly cited.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: The released training code includes a README documenting build instructions, configuration files, and experiment-reproduction scripts. The algorithmic modification is fully described in Section 4 and Algorithm 1.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [No]

Justification: This work does not involve crowdsourcing or research with human subjects. All experiments are purely computational, running self-play on synthetic Liar's Dice and Leduc poker games.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [No]

    Justification: This work does not involve human-subject research. All experiments are synthetic game-solving simulations with no participants, no personal data, and no potential risk to individuals.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: Large language models were used to assist with manuscript polishing, figure-generation code, and experiment scripting. LLMs were not used in the core methodology (SHA algorithm design, theoretical analysis, or experimental execution); the scientific contributions are entirely human-designed.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.

A Value network and the original ReBeL procedure

Value network architecture. The value network follows the ReBeL reference implementation for Liar's Dice: two hidden layers of 256 units, each followed by LayerNorm and a GELU activation, and a linear output layer whose weights are scaled by 0.01 at initialization. Input is the PBS representation from the ReBeL codebase: $2 + (2n_{\text{dice}}n_{\text{faces}} + 1) + 2n_{\text{faces}}^{n_{\text{dice}}}$ dimensions (acting player, traverser, a one-hot over bids and the call action, and two belief vectors); the output is one value per private hand. Training settings are given in Section 6.

The original ReBeL self-play procedure. Algorithm 2 reproduces the self-play procedure of Brown et al. [2020] in the notation of that paper, including the optional policy-network warm start and the policy-network data that Algorithm 1 omits because our experiments train only the value network. The iteration budget T is a fixed constant shared by every subgame. SHA replaces that constant with the draw in Equation (2) and changes nothing else; the reference implementation we build on uses the linear weights shown in Algorithm 1 in place of the uniform averages written here.

Algorithm 2 ReBeL self-play as stated by Brown et al. [2020]

function SELFPLAY($\beta_r, \theta_v, \theta_\pi, D_v, D_\pi$) ▷ β_r is the current PBS
  while β_r is not terminal **do**
    $G \leftarrow$ CONSTRUCTSUBGAME(β_r)
    $\bar{\pi}, \pi^{t_{\text{warm}}} \leftarrow$ INITIALIZEPOLICY(G, θ_π) ▷ $t_{\text{warm}} = 0$ and π^0 uniform if no warm start
    $G \leftarrow$ SETLEAFVALUES($G, \bar{\pi}, \pi^{t_{\text{warm}}}, \theta_v$)
    $v(\beta_r) \leftarrow$ COMPUTEEV($G, \pi^{t_{\text{warm}}}$)
    $t_{\text{sample}} \sim$ Uniform$\{t_{\text{warm}} + 1, \dots, T\}$ ▷ T is a fixed constant
    for $t = t_{\text{warm}} + 1, \dots, T$ **do**
      if $t = t_{\text{sample}}$ **then**
        $\beta'_r \leftarrow$ SAMPLELEAF(G, π^{t-1})
      end if
      $\pi^t \leftarrow$ UPDATEPOLICY(G, π^{t-1})
      $\bar{\pi} \leftarrow \frac{t}{t+1}\bar{\pi} + \frac{1}{t+1}\pi^t$
      $G \leftarrow$ SETLEAFVALUES($G, \bar{\pi}, \pi^t, \theta_v$)
      $v(\beta_r) \leftarrow \frac{t}{t+1}v(\beta_r) + \frac{1}{t+1}$COMPUTEEV($G, \pi^t$)
    end for
    Add $\{\beta_r, v(\beta_r)\}$ to D_v
    for $\beta \in G$ **do**
      Add $\{\beta, \bar{\pi}(\beta)\}$ to D_π ▷ optional policy-network data
    end for
    $\beta_r \leftarrow \beta'_r$
  end while
end function

B Future work and broader context

Future work. Since SHA leaves the game tree, the action space, and the leaf evaluator untouched, it composes with action abstraction [Brown et al., 2015, Li et al., 2024], online pruning [Li and Huang, 2025], and leaf-network changes such as the GNN belief encoder of Zeng et al. [2025a]. Since Theorems 5.3 and 5.4 hold for any symmetric budget distribution on the interval with variance at most σ^2, σ could be adapted online from the dispersion of value targets. The variance-reduction mechanism uses no two-player zero-sum structure, which makes multi-player and general-sum games a natural target [Li and Huang, 2026j]. Search-free self-play faces the same variance concern [Fan and Farina, 2026], and whether the cross-depth correlation also governs policy-gradient methods is open. Finally, the variance ratio improves with recursion depth, so games with deep public trees, such as multi-round poker, are where the method should help most. SHA inherits any speedup of the inner solver, such as parallel CFR [Li and Huang, 2026h], GPU-compiled CFR [Li and Huang, 2026e], and information abstraction [Li and Huang, 2026a,d], which brings such games within reach.

Evaluation. Our across-seed statistics require many full training runs, a cost that variance-reduced evaluation [Li et al., 2026a] would lower. Test-time reinforcement learning [Kubíček et al., 2026b], strategies that reach beyond the search depth limit [Milec et al., 2025], and safe exploitation with certified restricted responses [Li and Huang, 2026b] raise a complementary question: whether the lower-variance value networks produced by SHA also give more reliable opponent-exploitation and continuation estimates [Li and Huang, 2026f].

Language-model agents and search. Large language models have begun to play imperfect-information games directly, with and without solver assistance [Xu et al., 2025, Li et al., 2026c], and game theory continues to study how agents recursively model one another [Kovarik et al., 2026]. When such agents are coupled to a search procedure with learned values [Kubíček and Lisý, 2026, Zhang and Sandholm, 2026], the inner solver's budget again becomes a hyperparameter shared across every nested call, so the analysis of Section 5 applies unchanged.