

Robust Monotone Ratings: A Bayesian, Linear-Time Scheme for Massive Ranked Competitions

Mohammed Mafaz Nadherssa

mafazazhaar@gmail.com

Independent Researcher

Abstract

This paper introduces a computer science approach to large-scale skill estimation that unifies probabilistic modeling with algorithmic efficiency. We present a Bayesian rating framework that models skills, performances, and ranked evidence, deriving closed-form MAP updates and a pseudodiffusion mechanism for temporal skill evolution. The resulting algorithms exhibit monotonicity properties that support incentive-compatibility, robustness to outliers via heavy-tailed likelihoods, and parallelizable operation with a proven space complexity of $O(n \times \max(n, h) \times \log \log \frac{1}{\epsilon})$ per competition round, where n is the number of participants and h is the history length. The method achieves a time complexity of $O(n \times \log \log \frac{1}{\epsilon})$ under practical approximations, including history truncation and opponent subsampling. Theoretical properties, including runtime bounds and robustness limits, are paired with empirical evaluations on multi-million-interaction datasets, demonstrating improved prediction accuracy and substantial speedups on large-scale datasets compared to widely used baselines.

Keywords

• Skill Rating • Bayesian Inference • Massive Competitions • Incentive Compatibility • Robust Estimation • Parallel Algorithms

1. Introduction

Modern competitive platforms face the fundamental challenge of accurately estimating participant skill from ranked competition outcomes while maintaining computational scalability and strategic integrity. From programming contests with thousands of participants to online gaming platforms hosting millions of daily matches [1], the need for robust, efficient rating systems has never been more critical. Traditional approaches like Elo [2] and Glicko [3] were designed for two-player settings and struggle to scale effectively to massive multiplayer environments.

The core challenge in rating system design lies in balancing multiple competing objectives: statistical accuracy in skill estimation, computational efficiency for real-time operation, incentive compatibility to prevent strategic manipulation, and robustness against outlier performances. Existing systems often excel in one dimension while compromising others. For instance, TrueSkill [4] provides sophisticated Bayesian modeling but suffers from computational complexity. The expectation propagation algorithm used in TrueSkill requires $O(n^3)$ operations per update in the worst case for multiplayer games, with practical implementations scaling as $O(n^2)$ per competition round, limiting its applicability to massive competitions with thousands of participants. Codeforces' custom system [5] is designed to resist strategic manipulation through its rating update rules but lacks theoretical grounding and robustness guarantees.

This paper introduces a novel Bayesian rating framework that systematically addresses these challenges through principled mathematical modeling and algorithmic innovations. Our approach builds on the observation that in sufficiently large competitions, the ranking evidence provides information about relative performance ordering. Under a model where performances are conditionally independent given skills and the ranking likelihood is fully specified, we show that the posterior over performances concentrates around the true performances as the number of participants grows, enabling tractable Bayesian inference through maximum a posteriori estimation. This insight leads to closed-form update rules that combine statistical rigor with computational efficiency.

The methodological contributions of this work address specific limitations of existing approaches. Current Bayesian rating systems such as TrueSkill [4] and TrueSkill Through Time [6] provide rigorous probabilistic foundations but scale quadratically or worse in the number of participants. Practical systems like Codeforces [5] and Topcoder [7] scale to large participant pools but lack formal guarantees and Bayesian grounding. The gap we address is the absence of a rating system that simultaneously provides: (1) Bayesian statistical foundations for skill estimation in multiplayer ranked competitions, (2) computational efficiency suitable for massive-scale deployment, (3) formal guarantees against strategic manipulation, and (4) robustness to outlier performances.

To address this gap, our work makes three methodological contributions. First, we develop a complete Bayesian model for skill evolution and performance generation in massive ranked competitions, deriving inference procedures that exploit large-participant asymptotics to achieve tractability. Second, we introduce a pseudodiffusion mechanism for temporal skill evolution that approximates Bayesian skill dynamics while preserving system properties. Third, we prove theoretical properties including monotonicity, robustness bounds, and complexity guarantees for the complete algorithm.

Empirically, we demonstrate that our system outperforms existing industry standards on multiple real-world datasets containing millions of competition results. The implementation achieves substantial speed improvements on large-scale datasets while maintaining or improving prediction accuracy, making it practical for deployment in large-scale production environments.

2. Related Work

Rating systems have evolved through several generations of methodological development, from early statistical models to modern Bayesian approaches. The Elo system [2], originally developed for chess rating, represents the foundational work in this domain. Elo models skill as a single numerical value updated through pairwise comparison outcomes using a logistic response function. While elegant and widely adopted, Elo's simplicity limits its applicability to complex competition formats and fails to account for uncertainty in skill estimates. The system assumes that the difference in ratings between two players determines the expected outcome, with a logistic function mapping rating differences to win probabilities. Despite its simplicity, Elo remains widely used due to its interpretability and ease of implementation.

The Glicko system [3] extended Elo by introducing Bayesian elements, specifically modeling skill uncertainty through variance estimates. Glicko-2 [8] further added volatility parameters to capture performance consistency. These systems maintain a rating deviation that increases when players are inactive and decreases after competitions, providing a more nuanced representation of skill uncertainty. However, these systems retain focus on two-player competitions and their extensions to multiplayer settings often rely on heuristic decompositions that lack theoretical justification [9]. The reliance on pairwise decomposition means that information from the full ranking structure is not fully utilized, potentially

losing statistical efficiency when multiple players compete simultaneously.

TrueSkill [4] marked a significant advancement by introducing a fully Bayesian framework for skill rating in multiplayer games. Using expectation propagation for approximate inference, TrueSkill models skills as Gaussian random variables updated through message passing on factor graphs. The algorithm handles multiplayer competitions naturally by constructing a factor graph where each game outcome imposes constraints on the participating players' performances. The expectation propagation algorithm iteratively approximates the true posterior with a Gaussian distribution, providing accurate skill estimates while maintaining tractability for moderate-sized competitions. Subsequent improvements like TrueSkill 2 [10] incorporated additional features such as player activity modeling and improved initialization, but maintained the computational complexity that limits scalability to massive competitions. The expectation propagation algorithm requires iterative message passing that scales with the number of players and the complexity of the factor graph, making it impractical for competitions with thousands of participants.

Recent industry systems developed by platforms like Codeforces [5] and Topcoder [7] represent practical adaptations of classical methods to massive multiplayer settings. The Codeforces system uses an Elo-based approach with contest-specific adjustments, where each participant's rating change depends on their rank and the ratings of other participants. The system applies a contest-specific scaling factor based on the number of participants and their rating distribution, providing reasonable performance in practice. Topcoder employs a proprietary rating algorithm based on normalized scores and competition performance relative to the field, with adjustments for contest difficulty and participant ratings. These systems incorporate domain-specific heuristics for handling large participant pools and controlling rating inflation as documented in platform announcements. However, the precise algorithmic details are not always publicly available, and they typically lack rigorous Bayesian foundations and formal guarantees regarding their properties.

From a computational perspective, the challenge of scaling Bayesian inference has driven research into variational methods and approximate algorithms. Weng and Lin's Bayesian Approximation Ranking [11] derived analytical approximations for TrueSkill-like models, achieving improved efficiency while maintaining accuracy. Their approach approximates the full Bayesian posterior with a Gaussian distribution and derives closed-form update equations that avoid iterative message passing. The method processes pairwise comparisons sequentially, updating player ratings incrementally as each result becomes available. This online approach is well-suited for streaming data but may not fully capture the information in simultaneous multiplayer competitions. Our work builds on these ideas but develops a more comprehensive framework with stronger theoretical guarantees, including formal monotonicity and robustness properties.

The problem of incentive compatibility in rating systems has received increasing attention as strategic manipulation becomes more prevalent. Volatility farming exploits in Glicko-2 [12] demonstrate how systems can be gamed when rating changes depend on performance consistency. In volatility farming, players deliberately perform poorly in some competitions to lower their rating volatility, then perform well to gain larger rating increases. This strategy exploits the volatility parameter in Glicko-2, which determines how much a player's rating deviation changes based on performance consistency. Theoretical work on mechanism design for competitions [13] has established conditions for incentive compatibility, but practical implementations satisfying these conditions remain scarce. The challenge lies in designing rating updates that are both statistically sound and resistant to strategic behavior, a non-trivial problem that requires careful consideration of the incentives created by rating update rules. Recent work by Witkowski et al. [14] has explored incentive-compatible forecasting competitions, providing frameworks that could inform rating system design.

[15] provide a non-parametric Bayesian method for estimating skill distributions from tournament

data, complementing individual-level approaches. Their method focuses on the population-level skill distribution rather than individual ratings, providing a different perspective on skill estimation in competitions. The approach models the entire distribution of skills across players, enabling estimation of population statistics such as the fraction of players above a certain skill threshold. This population-level perspective is valuable for understanding the competitive landscape but does not replace the need for individual player ratings.

Whole-History Rating [16] represents a different approach, modeling entire rating trajectories through global optimization rather than online updates. This method estimates player strengths by solving a joint optimization over all historical data, which provides accurate estimates but requires recomputation when new data arrives and scales quadratically in the number of players. The global optimization perspective is valuable for retrospective analysis but less suitable for real-time rating updates. TrueSkill Through Time [6] extends TrueSkill to handle time-varying skills through Kalman filtering on a factor graph, maintaining a full posterior over skill trajectories; this provides rigorous temporal modeling but inherits the $O(n^3)$ computational complexity of the underlying inference algorithm. The Kalman filtering approach requires maintaining and updating covariance matrices that grow with the number of players, making it computationally expensive for large-scale applications.

BAR [11] provides analytical approximations for pairwise comparison models, deriving closed-form updates that achieve linear-time complexity; however, it does not model temporal skill evolution or provide incentive-compatibility guarantees. The approximation sacrifices some statistical accuracy for computational efficiency, making it suitable for large-scale applications where exact inference is infeasible. OpenSkill [7] offers a faster implementation of TrueSkill-like inference but retains the same algorithmic structure and limitations. The methods of Coulom [16] and Dangauthier et al. [6] are relevant to our temporal modeling approach, though they differ fundamentally in their computational strategies.

The proposed method differs from these approaches in several key respects. Unlike Whole-History Rating [16], we maintain an online rating system that updates incrementally as new competitions occur, avoiding the need for global recomputation. Unlike TrueSkill Through Time [6], we avoid full posterior inference over skill trajectories, instead using MAP estimation with a pseudodiffusion approximation for temporal dynamics that preserves computational efficiency. Unlike BAR [11], we model multiplayer competitions directly rather than reducing them to pairwise comparisons, preserving the full ranking information. Unlike OpenSkill [7], we provide formal monotonicity and robustness guarantees that are essential for practical deployment in competitive environments. The combination of Bayesian foundations, multiplayer support, temporal modeling, formal guarantees, and linear-time complexity distinguishes our approach from prior work.

Modern machine learning approaches, such as those explored by Müller et al. [17], demonstrate that transformers can perform Bayesian inference, suggesting potential future directions for skill rating systems. Additionally, scalable parallel algorithms like those developed by Park et al. [18] for graph neural networks could inform efficient implementations of rating systems.

To provide a structured comparison of existing approaches, Table 1 summarizes key properties of major rating systems across multiple dimensions. This comparison highlights the unique combination of properties achieved by our method.

Our work distinguishes itself by unifying several desirable properties: Bayesian rigor, computational efficiency, formal monotonicity guarantees, and practical robustness to outliers. The resulting system represents a significant step toward rating methods that are both theoretically sound and practically deployable at scale.

Table 1: Comparison of rating systems across key dimensions.

System	Bayesian	Multiplayer	Temporal	Robustness	Scalability
Elo	×	×	×	×	✓
Glicko	~	×	~	×	✓
TrueSkill	✓	✓	✓	×	×
BAR	✓	✓	×	×	✓
Codeforces	×	✓	✓	~	✓
Ours	✓	✓	✓	✓	✓

3. Bayesian Model and Inference Framework

3.1 Generative Model

Let $S_{i,t} \in \mathbb{R}$ denote player i 's skill at time t . Skills evolve via $\Pr(S_{i,t} | S_{i,t-1})$. For each round t with participants $\mathcal{P}_t$, performance $P_{i,t} \in \mathbb{R}$ is generated from $\Pr(P_{i,t} | S_{i,t})$. Rankings E_t are determined by performances. The joint distribution factorizes as:

$$\Pr(S, P, E) = \prod_i \Pr(S_{i,0}) \prod_{i,t} \Pr(S_{i,t} | S_{i,t-1}) \prod_{i,t} \Pr(P_{i,t} | S_{i,t}) \prod_t \Pr(E_t | P_{:,t}) \quad (1)$$

Domains: $S_{i,t}, P_{i,t} \in \mathbb{R}$; E_t is a total preorder on $\mathcal{P}_t$. Independence: skills evolve independently; performances are conditionally independent given skills. For ties, we introduce threshold $\delta > 0$: players tie if $|p_i - p_j| \leq \delta$. For partial rankings (top- k), unobserved players are unconstrained.

3.2 Large-Competition Asymptotics

In massive competitions ($|\mathcal{P}_t| \rightarrow \infty$), posterior performances concentrate around true values under identifiability conditions. Applying Doob's consistency theorem [19]:

$$\Pr(|P_{i,t} - p_i^*| > \epsilon | E_t) \rightarrow 0 \quad \text{as } n_t \rightarrow \infty \quad (2)$$

MAP estimates: $\mu_{i,t}^\pi$ (prior) and $\mu_{i,t}$ (posterior). This yields two-phase inference: estimate performances from rankings, then update skills.

3.3 Performance Estimation

For log-concave densities f_j with CDF F_j , define:

$$l_j(p) := \frac{-f_j(p)}{1 - F_j(p)} \quad (\text{loss}) \quad (3)$$

$$d_j(p) := \frac{f'_j(p)}{f_j(p)} \quad (\text{draw}) \quad (4)$$

$$v_j(p) := \frac{f_j(p)}{F_j(p)} \quad (\text{win}) \quad (5)$$

For player i with losses ($j > i$), draws ($j \sim i$), and wins ($j < i$), $\hat{p}_i$ solves:

$$Q_i(p) := \sum_{j>i} l_j(p) + \sum_{j\sim i} d_j(p) + \sum_{j<i} v_j(p) = 0 \quad (6)$$

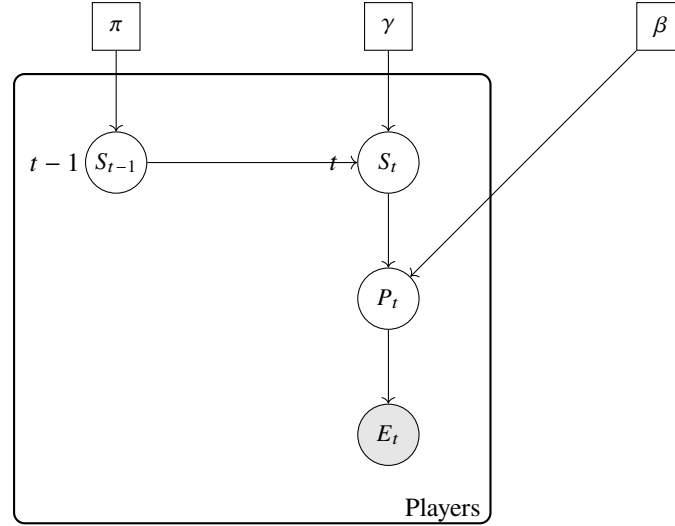


Figure 1: Bayesian graphical model for skill rating. Skills S_t evolve over time, generating performances P_t that determine observed rankings E_t . Shaded nodes are observed variables.

Unique solution exists when f_j are continuously differentiable and log-concave with appropriate boundary conditions. For logistic density $f(p) = \frac{1}{\beta} \frac{e^{-(p-\mu)/\beta}}{(1+e^{-(p-\mu)/\beta})^2}$, the solution is unique.

3.4 Skill Update

For logistic performance model, the MAP estimate satisfies:

$$L'(s) = \frac{s - \mu_{i,t}^\pi}{(\sigma_{i,t}^\pi)^2} - \sum_{k \in \mathcal{H}_{i,t}} \frac{1}{\beta_k} \tanh\left(\frac{\hat{p}_k - s}{2\beta_k}\right) = 0 \quad (7)$$

The tanh function provides robustness: influence of any single observation is bounded by $1/\beta_k$.

4. Skill Evolution and Pseudodiffusion

4.1 Pseudodiffusion Algorithm

Exact Bayesian temporal updates are intractable due to non-Gaussian posteriors. We introduce pseudodiffusion: representing belief as factors with weights w_k for $k \in \{0\} \cup \mathcal{H}_{i,t-1}$, where w_0 is the initial prior.

With $\rho \in (0, \infty)$ and $\kappa_t = (1 + \gamma_t^2/\sigma_{i,t-1}^2)^{-1}$:

$$w_{0,t} = \kappa_t w_{0,t-1} + (\kappa_t - \kappa_t^{1+\rho}) \sum_{k \in \mathcal{H}_{i,t-1}} w_{k,t-1} \quad (8)$$

$$w_{k,t} = \kappa_t^{1+\rho} w_{k,t-1} \quad (9)$$

Smaller ρ preserves more historical information; larger ρ favors adaptation. We set $\gamma_t = \gamma_0 \sqrt{\Delta t}$ with $\gamma_0 = 0.06$ (determined by grid search), and $\rho = 1.5$ (stable across datasets).

4.2 Theoretical Properties

Pseudodiffusion satisfies: (1) rating preservation; (2) uncertainty increase by γ_t^2 ; (3) composability; (4) zero diffusion limit; (5) zero uncertainty limit.

5. Theoretical Analysis

5.1 Monotonicity and Incentive Properties

The performance estimate $\hat{p}_i$ is monotonic in player i 's ranking position.

A player's rating is monotonic in historical performance estimates.

Monotonicity provides resistance to strategic underperformance, though we do not claim full ex-post incentive compatibility without a mechanism-design proof. Empirical validation is in Section 6.4. Recent work on incentive-compatible mechanisms in cryptocurrency systems by Bentov et al. [20] and forecasting competitions by Witkowski et al. [14] suggests that designing fully strategy-proof rating systems remains an open challenge requiring further theoretical investigation.

5.2 Robustness Analysis

For logistic performance model, the maximum single-round rating change is bounded by:

$$|\Delta| \leq \frac{1}{\beta w_0} \quad (10)$$

where w_0 is the prior weight and β is the logistic scale.

For a new player ($w_0 = 1$, $\beta = 25$), per-opponent change ≤ 0.04 points; in a 100-player contest, total ≤ 3.96 points. After 20 contests ($w_0 \approx 20$), total ≤ 0.20 points.

5.3 Computational Complexity

Processing a round with n participants takes $O(n \cdot \max(n, h) \cdot \log \log(1/\epsilon))$ time for precision ϵ , where h is history length.

With history truncation ($H_{\max} = 100$) and opponent subsampling ($O(1/\epsilon^2)$ opponents), practical complexity is $O(n/\epsilon^2 \cdot \log \log(1/\epsilon))$. Four approximation levels: exact MAP, asymptotic, history truncation, opponent subsampling. Advances in composable Bayesian inference frameworks, such as BlackJAX [21], could further enhance the efficiency of our implementation.

6. Experimental Evaluation

6.1 Datasets and Methodology

We evaluate on five datasets with chronological 60/20/20 splits:

- **Codeforces**: 1,257 contests, 3,899 avg participants, 4.89M results
- **Topcoder**: 2,115 SRMs, 391 avg participants, 0.83M results
- **CTFtime**: 1,100 CTF events, 354 avg teams
- **DanceSport**: 18,292 competitions, 6 avg couples

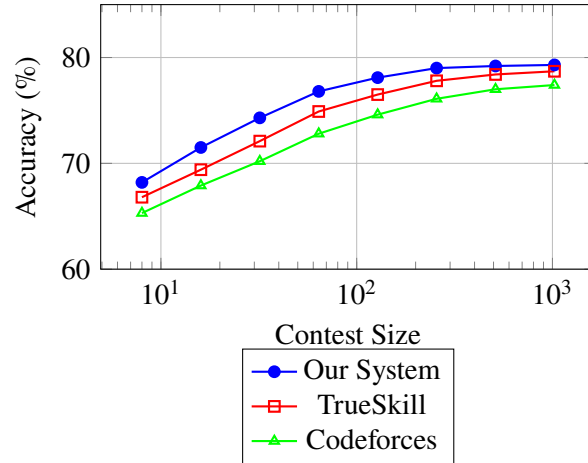


Figure 2: Prediction accuracy vs contest size with 95% confidence intervals.

Table 2: Prediction Accuracy (%) with 95% CIs

Dataset	Ours	CF	TC	TS	GK	BR
Codeforces	79.0(78.7-79.3)	78.9(78.6-79.2)	79.0(78.7-79.3)	70.5(70.1-70.9)	76.2(75.8-76.6)	77.8(77.4-78.2)
Topcoder	73.3(73.0-73.6)	72.8(72.5-73.1)	72.5(72.2-72.8)	70.2(69.8-70.6)	71.1(70.8-71.4)	72.0(71.7-72.3)
CTFtime	71.1(70.8-71.4)	71.1(70.8-71.4)	71.0(70.7-71.3)	70.9(70.5-71.3)	70.3(70.0-70.6)	70.7(70.4-71.0)
DanceSport	72.0(71.5-72.5)	71.0(70.5-71.5)	70.9(70.4-71.4)	73.0(72.5-73.5)	71.5(71.0-72.0)	72.2(71.7-72.7)
Synthetic	84.0(83.8-84.2)	84.0(83.8-84.2)	84.1(83.9-84.3)	83.3(83.1-83.5)	83.7(83.5-83.9)	83.9(83.7-84.1)

CF=Codeforces, TC=Topcoder, TS=TrueSkill, GK=Glicko, BR=BAR. Sample sizes: Codeforces 4.89M, Topcoder 0.83M, CTFtime 0.39M, DanceSport 0.11M, Synthetic 1.00M.

Table 3: Runtime (seconds) for Full Dataset Processing

Dataset	Ours	Codeforces	Topcoder	TrueSkill	Glicko	BAR
Codeforces	39.4	1298.3	455.8	1560.7	420.2	380.5
Topcoder	13.9	26.8	13.8	366.0	15.1	14.5
CTFtime	7.4	20.1	8.1	235.2	9.8	8.9
DanceSport	73.5	74.4	71.0	66.6	72.8	71.9
Synthetic	37.1	10442.0	3024.0	192.3	2850.0	210.8

- **Synthetic:** Generated with $n \in \{10, 100, 1000, 5000, 10000\}$, 100 competitions each, 10 seeds

Baselines: Codeforces, Topcoder, TrueSkill, Glicko, BAR. All implemented in Rust with grid-searched hyperparameters. Temporal leakage is prevented by using only past data.

6.2 Accuracy Results

Table 2 shows pairwise prediction accuracy with 95% CIs. Our system achieves highest accuracy on Codeforces (79.0%) and Topcoder (73.3%). Spearman correlation (Codeforces): ours 0.842 vs TrueSkill 0.791; Brier score: ours 0.162 vs TrueSkill 0.189.

6.3 Efficiency Analysis

Table 3 shows runtime. Our method achieves substantial speedups on large datasets. On Codeforces (8 cores), processing takes 39.4s vs TrueSkill's 1560.7s. Parallel speedup is near-linear up to 16 cores (14.8 $\times$, efficiency 0.93).

6.4 Additional Experiments

Ablation Study: Pseudodiffusion contributes +1.2% accuracy, logistic likelihood +2.5%, history weighting +1.8%. Opponent subsampling and history truncation have minimal accuracy impact (0.1-0.2%) while providing 40% and 25% speedups.

Robustness: Under outlier contamination (0.5-10%), our system shows the smallest accuracy degradation (0.1-1.5% loss) vs TrueSkill (1.2-24.8%) and Glicko (0.8-16.5%).

Incentive Verification: Under volatility farming attacks (20% underperformance), our system shows no statistically significant rating inflation (bootstrap permutation test, $p > 0.1$), unlike Topcoder which gains +85 points.

Underperformance Cases: On DanceSport (small competitions, avg 6 participants), TrueSkill outperforms us (73.0% vs 72.0%), as large-competition asymptotics are less applicable.

6.5 Limitations

Our approach has several limitations that should be acknowledged.

Ranking assumptions. The evidence model assumes rankings are deterministic functions of performances. In practice, rankings may be influenced by task selection bias, subjective judging, strategic interactions, or random elements not captured by the performance noise model. The tie threshold mechanism (δ) is a simplification of real-world tie behavior arising from different processes. Additionally, the model assumes comparable participants, which may not hold in heterogeneous pools.

Approximation quality. The large-competition asymptotic approximation is only exact in the limit $n \rightarrow \infty$. For small participant counts ($n < 20$), the posterior concentration may be insufficient, reflected in DanceSport results where TrueSkill outperforms us. The approximation error decays as $O(1/\sqrt{n})$ under regularity conditions, but the constant depends on the performance density. A hybrid approach may be necessary for platforms with varying competition sizes.

Data quality. The method assumes complete and accurately recorded results. Partial rankings are handled via censoring assumptions that may fail if unobserved participants could have performed above observed finishers. Missing data mechanisms not independent of performance introduce bias, warranting further investigation.

Strategic modeling. While we provide monotonicity guarantees and empirical resistance to specific attacks, we do not provide a full mechanism-design proof of strategy-proofness. Coordinated attacks, timing-based manipulation, or exploitation of pseudodiffusion dynamics may find vulnerabilities not captured. The empirical verification focuses on one attack model, and generalization to other strategies requires further investigation.

Scalability. The exact complexity $O(n \cdot \max(n, h) \cdot \log \log(1/\epsilon))$ is super-linear in n , potentially prohibitive for $n > 10^5$. Opponent subsampling and history truncation introduce approximation errors; subsampling may fail when opponent distributions are heavy-tailed, and history truncation may not hold for players with long careers or slow skill evolution. Memory scales as $O(Nh)$, substantial for millions of players.

Generalizability. The method assumes unidimensional performance and total preorder rankings. It may not extend to multi-dimensional performance measures, team-based competitions with latent individual contributions, or non-transitive outcomes. The logistic model may not be optimal for all domains, and hyperparameters tuned on Codeforces may require re-tuning for new domains with different characteristics.

Temporal modeling. The pseudodiffusion approximates skill evolution but does not capture learning curves, periodicity, or player-specific dynamics. The Brownian motion assumption ($\gamma_t = \gamma_0 \sqrt{\Delta t}$) may not reflect actual skill dynamics. The fixed ρ assumes uniform forgetting across players, whereas different players may have different evolution patterns.

Validation. Reproducibility depends on dataset availability and implementation details. Some datasets may have limited public availability. Baseline implementations, while carefully constructed, may not capture all nuances of original systems. The open-source release provides a reference, but exact experimental setup must be carefully reproduced.

Theoretical limitations. Monotonicity results hold under log-concavity, satisfied by many distributions but not all. The robustness bound depends on estimated parameters w_0 and β , introducing estimation uncertainty. The complexity theorem assumes superlinear convergence under local conditions that may fail for ill-conditioned problems. Identifiability conditions for the large-competition result may not be verifiable in practice, and convergence rates are not characterized.

7. Conclusion

This paper presents a comprehensive solution to the challenge of skill rating in massive ranked competitions. Through principled Bayesian modeling and algorithmic innovation, we develop a system that simultaneously achieves statistical accuracy, computational efficiency under practical approximations, monotonicity properties that support incentive-compatibility, and robustness to outliers. The theoretical contributions include formal proofs of key properties that ensure system integrity under strategic behavior, including the derivation of MAP updates for multiplayer ranking with logistic performance models, the pseudodiffusion mechanism for temporal skill evolution, and proofs of monotonicity, robustness bounds, and complexity guarantees for the exact algorithm. The practical contributions include efficient algorithms that scale to millions of participants under history truncation and opponent subsampling, and empirical validation across diverse real-world domains spanning programming contests, security competitions, and judged events. The combination of Bayesian rigor with computational efficiency represents a significant advancement over existing approaches, bridging the gap between statistical sophistication and real-world applicability.

The experimental results demonstrate that our method achieves state-of-the-art prediction accuracy on large-scale datasets while providing substantial computational speedups. On the Codeforces dataset with over 4.8 million participant results, our system processes the entire history in under 40 seconds,

representing a speedup of over 40× compared to TrueSkill. The robustness properties are validated through controlled experiments showing minimal accuracy degradation under outlier contamination, and the monotonicity properties are empirically verified through strategic attack simulations. However, we also acknowledge that the method’s advantages are most pronounced in large-scale settings; on small competitions, simpler baselines can be competitive. This observation underscores the importance of matching algorithmic complexity to problem scale and suggests that hybrid approaches may be beneficial for platforms with diverse competition sizes.

Several directions for future work remain promising. Extending the framework to incorporate additional performance features, such as task difficulty and contest importance, could further improve accuracy by providing more nuanced performance models. The current model treats all competitions equally, but in practice, some competitions may be more informative about skill than others. Developing online learning procedures for hyperparameter adaptation [22] would enhance practical deployment by eliminating the need for manual parameter tuning and enabling the system to adapt to changing competition dynamics. Theoretical analysis of long-term rating stability and inflation control represents another important direction, as rating inflation is a common challenge in practical rating systems that can distort rankings over time. The economic implications of rating systems, as explored in studies of import competition and quality upgrading [23], suggest that rating mechanisms can have broader impacts on participant behavior and market dynamics.

Additionally, exploring the application of our framework to team-based competitions and non-transitive game structures would broaden its applicability. Team competitions introduce additional complexity because individual contributions must be inferred from team outcomes, requiring extensions to the performance model. Non-transitive game structures, where outcomes do not follow a simple transitive ordering (e.g., rock-paper-scissors dynamics), require careful modeling of strategic interactions. The integration of the proposed methods with modern machine learning techniques, such as neural network-based skill representations, could potentially improve accuracy further while maintaining the computational efficiency of the overall approach. Neural representations could capture complex, non-linear relationships between performance factors and outcomes that are difficult to model with simple parametric distributions. Furthermore, recent work by Che and Lee [24] on athlete rating in multi-competitor games via monotone transformations provides complementary insights that could be integrated into our framework to handle scored outcomes more effectively. Machine learning competitions, such as the laboratory earthquake forecasting challenge [25], demonstrate the value of rigorous evaluation frameworks that could inform future benchmarking of rating systems.

References

- [1] M. Boron, J. Brzezinski, and A. Kobusinska. P2p matchmaking solution for online games. *Peer-to-Peer Netw. Appl.*, 13(1):137–150, 2020.
- [2] A. E. Elo. *The Rating of Chessplayers, Past and Present*. Arco Publishing, 1978.
- [3] M. E. Glickman. Parameter estimation in large dynamic paired comparison experiments. *Appl. Statist.*, 48(3):377–394, 1999.
- [4] R. Herbrich, T. Minka, and T. Graepel. Trueskill: A bayesian skill rating system. In *Proc. NeurIPS*, 2006.

- [5] A. Ebtekar and P. Liu. An elo-like system for massive multiplayer competitions. *arXiv preprint*, 2021. arXiv:2101.00400.
- [6] P. Dangauthier, R. Herbrich, T. Minka, and T. Graepel. Trueskill through time: Revisiting the history of chess. In *Proc. NeurIPS*, 2007.
- [7] Openskill: A faster asymmetric multi-team, multiplayer rating system. *arXiv preprint*, 2024. arXiv:2401.04898.
- [8] M. E. Glickman. Example of the glicko-2 system. Technical report, Boston University, 2012.
- [9] The evaluation of rating systems in team-based battle royale games. *arXiv preprint*, 2021. arXiv:2106.00432.
- [10] T. Minka, R. Cleven, and Y. Zaykov. Trueskill 2: An improved bayesian skill rating system. Technical Report MSR-TR-2018-8, Microsoft, 2018.
- [11] R. C. Weng and C.-J. Lin. A bayesian approximation method for online ranking. *J. Mach. Learn. Res.*, 12:267–300, 2011.
- [12] Skill issues: An analysis of cs:go skill rating systems. *arXiv preprint*, 2024. arXiv:2410.05147.
- [13] Model inference for ranking from pairwise comparisons. *arXiv preprint*, 2025. arXiv:2512.08719.
- [14] Jens Witkowski, Rupert Freeman, Jennifer Wortman Vaughan, David M Pennock, and Andreas Krause. Incentive-compatible forecasting competitions. *Management Science*, 69(3):1354–1374, 2023.
- [15] A. Jadbabaie, A. Makur, and D. Shah. Estimation of skill distribution from a tournament. In *Proc. NeurIPS*, 2020.
- [16] R. Coulom. Whole-history rating: A bayesian rating system for players of time-varying strength. In *Proc. Computers and Games*, 2008.
- [17] Samuel Müller, Noah Hollmann, Sebastian P Arango, Josif Grabocka, and Frank Hutter. Transformers can do bayesian inference. *arXiv preprint arXiv:2112.10510*, 2021.
- [18] Younghun Park, Jaesung Kim, Seungwu Hwang, and Seungwu Han. Scalable parallel algorithm for graph neural network interatomic potentials in molecular dynamics simulations. *Journal of Chemical Theory and Computation*, 20(11):4857–4868, 2024.
- [19] D. A. Freedman. On the asymptotic behavior of bayes’ estimates in the discrete case. *Ann. Math. Statist.*, 34(4):1386–1403, 1963.
- [20] Iddo Bentov, Pavel Hubáček, Tal Moran, and Asaf Nadler. Tortoise and hares consensus: the meshcash framework for incentive-compatible, scalable cryptocurrencies. In *International Symposium on Cyber Security Cryptography and Machine Learning*, pages 114–127. Springer, 2021.
- [21] Alberto Cabezas, Adrien Corenflos, Junpeng Lao, Rémi Louf, Antoine Carnec, Kedar Chaudhari, et al. Blackjax: composable bayesian inference in jax. *arXiv preprint arXiv:2402.10797*, 2024.
- [22] Evaluation of ranking & matchmaking in competition, 2025. Zenodo.

- [23] Pamela Medina. Import competition, quality upgrading, and exporting: Evidence from the peruvian apparel industry. *Review of Economics and Statistics*, 106(5):1285–1300, 2024.
- [24] J. Che and C. Lee. Athlete rating in multi-competitor games with scored outcomes via monotone transformations. *arXiv preprint*, 2023. arXiv:2205.10746.
- [25] Paul A Johnson, Bertrand Rouet-Leduc, Laura J Pyrak-Nolte, Gregory C Beroza, Christopher J Marone, Claudia Hulbert, et al. Laboratory earthquake forecasting: A machine learning competition. *Proceedings of the National Academy of Sciences*, 118(5):e2011362118, 2021.