Ranking Changes White Paper

Accuracy and Progression Improvements

Robert Petit

July 2022

Abstract

Previously in Skyweaver, we have used ELO simultaneously for matchmaking and for general rank. While this is a straightforward method for a ranking system, it presents several issues. In this white paper, we outline our chosen alternative system that uses, separately, a matchmaking rating (MMR), calculated using the Glicko system, and a rank score, calculated from match performance and play frequency. The end result is more accurate matchmaking and newly-created design space for ranked experience that does not infringe upon the integrity of matchmaking accuracy

1 Overview

Currently, the ranking system for Skyweaver takes place over a month-long season, players receive an ELO of 1500 at the start of the season and are ranked according to their forward-facing ELO score, which increases and decreases through play. Here, we outline the overhaul of this system: players are given a matchmaking rating and rank points separately from one another. Matchmaking rating is used to keep matches competitive and engaging, while the rank points are used to provide players with an easy-to-interperet comparison to between players. In practice, the matchmaking rating is a completley independent process where we emphasize *accuracy* above all else. In technical terms, we are using an estimator with low bias and trying to appropriately estimate, rather than minimize, the variance of matchmaking estimations. Rank points, on the other hand, are partially dependent on the matchmaking ranking, but are something that we can bias and tune to feel fair, fun, and rewarding. The resulting system offers substantial improvements in both subjective measures, like player enjoyment and reward scalability, and objective measures, like empirical validity and estimable tuning parameters.

This document serves three main purposes. First, we outline the calculations used at each step of the process. This first step is the most technical, so do not be dissuaded if the first look at this system is intimidating. Second, we provide some intuition behind the behaviors of the calculations to help simplify and make sense of the technical definitions. Lastly, we link those behaviors to the experience of each individual who will play on the ladder in both the short and long term. This last step is the crucial one that explains *precisely* what we get out of the system at each step.

The first of two systems we will cover is used for matchmaking, called the MMR system. We use a slightly-altered version of the Glicko System [1] to generate a rating and 'rating deviation', denoted r and RD, and dive deeply into the mechanics and dynamics of that system. Having a distinct and estimable rating deviation (whose interpretation is very near to a standard error for the estimated r) is perhaps the key addition of the entire system. The use of RD when updating r is what makes this system so superior to traditional ELO, and our ability to carry through RD to future calculations will offer exponential improvements for calibration and future alterations to any ranking structures.

The second system controls players rank points, denoted RP, their forward-facing score that judges their true game ranking, which is provided for players to directly compare their progression and issue rewards. This system is informed by the MMR, but does not inform MMR itself. This means that the ranking system is a recipient of accurate information in the form and scale that it is needed, and never needs to impede our ability to access accurate information directly. On top of the information from the unbiased, highly variable estimator, we can add in our own bias and scale to make this system more interpretable at a glance. We also use this to draw our end conclusions about how players are rewarded for their play, since we can control the properties of this value extremely closely.

The meat of this paper is the two systems, but we also outline a little extra information about how this interacts with the existing season structure, and how we will handle the migration from the previous ELO structure to the new one, since resetting all players completley in the middle of a season, which is the 'principled' approach, is a very bad option in our case. Our approach in this case focuses on simplicity and clarity, and we want to establish the technical validity of this approach.

Overall, we are meeting four clear goals: matchmaking that creates consistently competitive matches, a ranking ladder that emphasizes continued play with clear rewards for each match, a within-season process that creates weekly competition for top spots, and an inter-season process that creates long-term opportunities for climbing and improvement.

2 Matchmaking Rating

The Glicko rating system is somewhat math intensive, which can make it appear unintuitive at first. It represents a number of improvements over the ELO system, coming at the cost of calculation that is too unwieldy to be calculated by hand, but still simple programmatically. The core of the system is built on three layers. The first layer is the rating, r, of a player. When a player plays a match, we will move their rating according to their calculated Rating Deviation (RD), which can be interpreted similarly to a standard error of their rating. As time progresses, players will regularly have their RD increased according to a specific rating period. These rating periods are opportunities for players to play games at a reasonably-consistent level of skill, and a way for us to allow more movement along the ladder at opportune times; we will tie these to our season structure (1 week) to match them with our patch structure.

Terminology

As a small aside on terminology, please note:

- Rating (r) is the term used for a players matchmaking rating.
- Rating deviation (RD) is the term used for the likely distance between their 'true' rating and our estimation of their rating.
- Rank is the term for a players overall rank on the ladder.
- Rank points (RP) is the more granular score that players increase to climb.

The similarity of these terms to each other is unfortunate, but changing them would make the language used here inconsistent with terminology used elsewhere, both in Skyweaver media and in the academic literature on this topic. We have chosen to adopt them and adhere to them as rigorously as possible.

The Process

Calculating a players change in ranting is two-step process that can be carried out in each rating period; first, we need to determine the RD of each player, then we update their ranks separately. This portion of the system is identical to the process laid out by Mark Glickman in his quintessential paper [2] on the topic. We give a brief treatment of it here, but strongly suggest that anyone interested in more information refer to his paper outlining the algorithm [2] and the more technical derivation of the system [1].

Variable List

- Rating r
  - r_{Opp} is the rating of their opponent
  - r' is the post-match adjusted rating
- Rating Deviation $RD \in [30, 350]$, where lower means that we are more certain of their 'true' skill
  - RD_{Old} is a players' RD from the previous rating period
  - RD_{Opp} is the RD of their opponent
  - RD' is their post-match adjusted RD
  - RD_{Init} which is the highest RD we assign to players about whom we have no recent data. Initially, this is 350, but empirically we may want to raise this in the future.
  - RD_{Min} is the minimum RD we allow
- Time (in rating periods) since last match t
- Time-sensitive uncertainty scalar c
- Match outcome $s \in \{1, \frac{1}{2}, 0\}$ for a win/draw/loss

For players who are just entering, they will start off play with an $RD = RD_{Init}$ and $r = 1500$

Step 1: Calculate RD

If a players' most recent match was not in this rating period, we expand their RD according to:

$$RD = min(\sqrt{RD_{Old}^2 + c^2 t}\,,\ RD_{Init}) \tag{1}$$

In practice, this is done once per period (currently 1 week) for all players at the start of the week.

Step 2: Calculate Post-match Rating change

After two players play a match, we will do these calculations for them separately. These refine the estimations of their rating, accounting for their relative ratings and certainties.

$$r' = r + \underbrace{\frac{q}{1/RD^2 + 1/d^2} \times g(RD_{Opp})}_{\Delta Scale} \times \underbrace{(s - E(s \,|\, r, r_{Opp}, RD_{Opp}))}_{\Delta Direction} \tag{2}$$

$$RD' = max\left\{\sqrt{(\frac{1}{RD^2} + \frac{1}{d^2})^{-1}},\ RD_{Min}\right\} \tag{3}$$

Extra Terminology Definitions

There are two variables and two functions listed above that are thus-far undefined.

$$q = \frac{ln(10)}{400} = 0.0057565$$

$$g(RD) = \frac{1}{\sqrt{1 + 3q^2(RD^2)/\pi^2}}$$

$$E(s \,|\, r, r_{Opp}, RD_{Opp}) = \frac{1}{1 + 10^{-(g(RD_{Opp})(r - r_{Opp})/400)}}$$

$$d^2 = \left(q^2 \times g(RD_{Opp})^2 \times E(s \,|\, r, r_{Opp}, RD_{Opp}) \times \left(1 - E\left(s \,|\, r, r_{Opp}, RD_{Opp}\right)\right)\right)^{-1}$$

Discussion

Intuition Behind Definitions

First and foremost, let's examine RD. It's initial and maximum value is 350, which basically correlates to 'we have no idea how a player will perform'. This serves as a soft cap on the rating change for a given player through its appearance in the second term of equation 2, and also attenuates the rating change of their opponent when it's high (which we will talk about at length when we discuss the definition of d^2). Our definition of RD is also time-dependant. We will regularly introduce uncertainty into the equation through step 1. This is a *huge* improvement over the ELO system, since we are functionally building in the opportunity for players to 'prove' that they have gotten better (or worse) over time, and devaluing the *quality* of rating that they have achieved in the past without devaluing the rating itself. This is not a completley accurate standard error for a players rating, but it functions much like one in many scenarios, to the point where we can issue a 95% confidence interval of a rating with $[r - 1.96RD, r + 1.96RD]$.

q is a universal constant meant to establish the scale of the rating system. In this case, we want a rating change of 400 to represent a 10-fold change in the likelihood of a player winning, so we set it such that the log-liklihood change we intend is scaled by 400.

$g()$ is a function that helps scale the outcome based on the rating certainty; it's primary purpose is to provide a regularized, sliding inverse of the RD for use elsewhere. If this seems complex, it is.

Intuitively, this just gives us a quick-reference for transforming 'high RD_{Opp}' $\rightarrow$ 'low r change' whenever we need it.

The expected outcome of the match, $E(s \,|\, r, r_{Opp}, RD_{Opp})$, gives a decimal that corresponds to the percentage chance of the player winning. This factor appears in two critical places: the rating change and the definition of d^2. For now, focus on the last term in EQ. 2 updating r'. We get the difference between the observed outcome and the expected outcome. This term, $(s - E(\cdot))$ is the only term in the entire definition of r' that is not strictly positive by definition, so if it is negative, the rating adjustment is unambiguously negative and vice versa. If $E(\cdot)$ is large, and the player wins, this term is small but positive. If $E(\cdot)$ is small and they lose, it is small but negative. The exact definition for $E(\cdot)$ is somewhat strange, but it quickly 'reverse engineers' the logarithmic logic built into q and $g(\cdot)$ to output the odds based on the rating.

d^2 is a term that we consistently use to scale the changes to r and RD. It relies on the product of the probabilities of the observed outcome, $E(s \,|\, r, r_{Opp}, RD_{Opp})$, and probability of the other outcomes, (1-$E(s \,|\, r, r_{Opp}, RD_{Opp})$). Note that, generally, $x \times (1-x) \mid x \in [0,1]$ is *maximized* at $x = 0.5$. Since d^2 is the inverse of these, it is *minimized* with respect to $E(\cdot)$ when the odds of the match are 50/50. We also grab the square of $g(RD_{Opp})$, to provide additional scaling. Remember that this term is less than one, and smaller when RD_{Opp} is larger. This scaling factor serves a dual purpose: it will make the rating change larger when the match has a more surprising outcome *and* when the opponent has a more certain rating. This makes our estimator more robust to matches against smurfs and new players, since it moves them *much* more quickly through the ratings and does not penalize the losing player who was mismatched due to low information.

Lastly, we can finally provide solid intuition behind the changed rating, r'. The section labeled $\Delta Scale$ is made up of a few factors. q, which sets the scale for the rating system as a whole; $1/RD^2$, which shrinks the rating change as our rating becomes more certain; $1/d^2$, which grows the rating change as the outcome of the match becomes more 'surprising'; and $g(RD_{Opp})$, which shrinks the rating change as our the deviation of our opponents rating increases. The $\Delta Scale$ is always going to be positive.

Multiplying with the scaling factor is the $\Delta Direction$ term. We set the direction (+/-) of the rating change according to the $\Delta Direction$ term, which is always negative with a loss, always positive with a win, and larger (in magnitude) when a lower-rated player beats a higher rated player.

Properties of the Rating System

This system has a number of properties that make it a vast improvement on the ELO system. First and foremost, it has an explicit Rating Deviation, which can allow us to make inferences about both the rating system and players themselves. Second, it uses both the relative ratings and the Rating Deviation of the player *and* their opponent to scale the rating change after a match. This makes the entry of new players into the system very gentle, and the entry is relatively robust to smurfs on both sides; players who encounter someone who is rated extremely incorrectly won't have their rating upset too heavily by a loss, and players who begin playing but easily beat their selected opponents climb very quickly.

As players begin to climb, the higher the skill gap between them and the opponents that they beat, the more rating they get and the larger their RD gets. Once they start to lose to players higher-rated than them, their RD shrinks, and the rating change shrinks. This is counteracted by the 'surprising result' factor (d^2) that is built into the calculation, which scales things up for a surprising result and scales things down for unsurprising results. This 'surprise' factor does not depend on a players' own RD at all, and instead gets set based on their opponent and their rating, so when there is a big surprise in the outcome, the rating may still move a lot, even if that players' RD was low to begin with.

Finally, we have come full circle, since the updated RD for a player depends on the surprise of their match outcome. If the match ended as it 'should', according to the rating, their RD shrinks for future matches and vice versa.

The final property of this system to cover is the use of 'rating periods' to introduce turbulence across the entire system on a regular occasion. Before the first match in each period, we update every players' RD to be strictly higher than it was at the end of the last one. By doing this, we discount the accuracy of their previous rating without discounting the rating itself, so players who have had skill changes, or whose skill was narrow and is no longer applicable (for instance, someone who was playing a strong deck that received a corrective nerf) are allowed to move around the ratings with more freedom than they would have moved prior. It also keeps players who take long breaks from over-capitalizing on their previous play.

3 Ranking Points

The forward-facing portion of the ranking system are the Ranking Points. The purpose of Rank Points (RP) is to add more desirable behaviors to the MMR system for players to focus on. In particular, we want more interpretable numbers, a positive bias for progression, and room for changing/expanding the system in the future without compromising accuracy.

Table 1: Rank Values

Rank	**RP Minimum**	**Hard Floor**	**Soft Reset Value**	**Hard Reset Value**
Wanderer	200	Yes	-	-
Trainee I	300	Yes	-	300
Trainee II	400	-	-	350
Trainee III	500	-	-	400
Apprentice I	600	Yes	-	450
Apprentice II	700	-	-	500
Apprentice III	800	-	-	550
Expert I	900	Yes	-	600
Expert II	1000	-	-	700
Expert III	1100	-	-	800
Master	1200	Yes	1300 RP Cap	900
Grandweaver	Top 100 Masters	-	1400 RP Cap	1000

We overhaul the front-end for this and, instead of set values for winning and losing, your score is a combination of three factors: The MMR change, $\Delta r = r' - r$ after a match given by EQ. 2, a positive bias factor B and a win-streak bonus A The win streak bonus, A, is included in the formal definition, but set to $A = 0$ in the first drafts of the system to ease parameter tuning in the early days. The precise post-match update is given by

$$RP' = RP + \underbrace{\underbrace{\Delta r \times F}_{\text{Glicko Change}} + \underbrace{B \times s}_{\text{Positive Bias}} + \underbrace{A}_{\text{Win Streak Bonus}}}_{\Delta RP} \tag{4}$$

$$F = \underbrace{\frac{1000}{3000-1250}}_{\text{Range Transform}} \times \underbrace{min\{1, f_{min} + (1-f_{min}) \cdot \frac{(RP-200)}{1000}\}}_{\text{Rank-Dependence}}$$

$$B = max\{b_{min}, b_{max} - \left(\frac{(RP-200)}{1000}(b_{max}-b_{min})\right)\}$$

$$A = \left\{ \begin{array}{ll} a, & \text{if previous 2 matches are wins} \\ 0, & \text{otherwise} \end{array} \right\}$$

Where F is a tuning parameter that transforms the Glicko rating from the log-liklihood scale to the $RP \in [0, 1000]$ scale of this system and adjusts the importance of MMR changes at your rank, B is an additional reward to winning players (that is *not* mirrored for the loser, hence bias), $s \in \{1, \frac{1}{2}, 0\}$ is the win/loss/draw result of the game, and A is the win streak parameter.

Above, we choose B such that it starts at b_{max} when $RP = 0$ and slides continuously down to b_{min} when $RP = 1200$, and holds at b_{min} beyond that. $b_{max/min}$ are simply the bounds of that bias. F transforms $Domain\{r\} \to Domain\{RP\}$ and slides the importance of MMR up such that, by the time you hit $RP = 1200$, your change in rank is almost entirely dependent on your MMR change, but at lower ranks ΔRP is less dependent on your MMR. $f_{min} \in (0, 1)$ sets the initial importance of MMR in the equation as a percentage. A is the logic-gated representation of a, the discrete win-streak bonus.

In practice, we will round the post-adjustment RP since the outcome is going to include some fraction. Given the distribution of players that we will probably observe, the rounding here is likely to have a very small net positive effect on the system (it is slightly more likely to round up than to round down) but this will be ultimately trivial in terms of the whole system.

The result of this system is simply that, by winning a match you always go up some value; that value will be mostly based on B at lower ranks and mostly based on Δr at higher ranks. The change is also continuous, so while the interpretation of RP is different at different points on the ladder, it is always similar between any two close ranks.

As players progress through the rank, they will 'rank up' each time they hit a rank floor. This is a purely independent process until players achieve Master rank, at which point they are competing with other players on the ladder to be in the top 100 players in terms of RP. Those top 100 players, beyond $RP = 1200$, make up the entire Grandweaver rank. This is where the small rank inflation provided by B provides substantial systematic value.

Since players are going to be increasing their RP on average over time (the leader board is *not* zero-sum due to A and B) it is a strictly-dominant strategy to always play matches. This solves one of the key issues with our current system in which players who achieve very high ELO are actively disincentivized from playing since their ELO loss is guaranteed to be larger than their ELO gains if they are above average for their pool of opponents. This creates a long run output (through cascading decisions) that requires players to strongly believe that they are very under-ranked in order to play *at any level* where you are ranked purely off of ELO if relative rank is your sole goal. Rank inflation, and scaled results, solves that, since you can be above average and still have a positive expected value for a ranked game with even odds in terms of RP. We do have to account for this inflation eventually, but we allow it to persist in the time frame where players are competing for rank-based rewards.

Wanderer

The first 'rank' is Wanderer. Players will have this rank when they initially reach *account level* 15 and gain access to ranked queue. At this time, they will proceed ahead as normal. One note for future empirical analysis: we expect that the sample standard deviation r for each rank will be distributed by approximately χ_1^2, centered in the middle of the ladder: players around the middle of the ladder will exhibit the most variance in rating by that point. This relationship will not hold for Wanderer, since players who begin playing and immediately lose will drop rating, r, but not rank, RP, since they are already at the rank floor. In the future, this will mean that any estimation of ranking windows will have to intentionally omit wanderer for accuracy.

Hard Rank Floors

Players have some measure of demotion protection in the form of rank floors. If a rank floor is denoted 'hard' in Table 1 then players RP cannot decrease past this value once they hit it. This means that, if you climb to Apprentice I, you will *never drop below* Apprentice I for the season. These rank floors happen at the full ranks, not the inter-rank tiers. For the hard reset between seasons, these values are not in effect and players will be reset downward appropriately.

4 Season Structure

The proposed season structure is similar to what is already in place: 4 week seasons, punctuated by 'hard' reset at the end, with one-week sub-seasons marked with soft resets and rewards according to the structure already in place.

Weekly Soft Reset

The soft reset at the end of each week will have two changes: the *top* of the ladder will be truncated down to achievable values in order to counteract the RP inflation provided by B in EQ. 4 and keep the top-100 goal achievable for all Master-rank players. Master players are also truncated, though this will not affect those at the bottom of the leader board, in order to give the Grandweaver players some room for error in the first day or two of the week. These marks, provided in Table 1 are approximated based on no underlying theory, so in practice may need to be altered.

The soft reset will also serve as the roll-over for the ranking period, denoted t in EQ. 1, which will inject some artificial deviation into the ladder. This will make it easier for players at every level to make quick gains as soon as they return to playing in the new week.

Monthly Hard Reset

The hard reset will be much more impactful, and will line up with big changes issued by balance patches. The most notable is a sweeping RP change. Every player will have their RP set to a specific value based on their achieved rank that season (and have their rank updated accordingly). It will be less severe than the current practice of resting every player to (the rank formerly known as) Apprentice, and still leave players roughly distributed on the ladder.

The more substantial change is the change issued to MMR which will be carried out by calculating the following for each player:

$$r' = 1500 + \left(\frac{r - r_{lowest}}{r_{highest} - r_{lowest}} * 750\right) \quad (5)$$
$$RD' = 350 \quad (6)$$

The result of this will be redistributing MMR according to the current distribution, but fit to the range of 1500-2000. This will make the matchmaking system more generous without completley erasing the information we learned in the previous season and allow high-ranked players to still achieve the gains they need to get $RP > 1000$.

End of Season

At the end of each season, we will provide a single, season long rank for each player. For any player who does *not* reach Master, this is simply their rank since they are unaffected by rank resets. Players in Master+ will have their end-of-week RP scores averaged, and will be ranked on this number for the whole season. Players will see either their end-of-season rank or their average RP as their Season Ranking when they look at their past seasons performance.

5 Migration

To migrate from the current ELO system, we are just can simply 'back' the current information into the existing structure. Our three core values (r, RD, and RP) will need to be constructed from the singular ELO system. The 'technically accurate' way of doing this would be to reconstruct these values from the ground up, based on match info. This, however, is a prohibitively difficult task with relatively small payoffs (see [1] and [2]) so, instead, we are defaulting to the much more simplified version and offloading the information discovery onto the ranking process itself.

Rating r will be set by just copying over ELO one-to-one. There are some obvious issues with this, since the ELO approximation is constructed differently and both are sensitive to the relative 'true' ratings of the rest of the pool (and thus the relative estimations, which have different consistency properties) but these scores are both built on the same underlying assumptions and is probably the best case scenario. There is one nuance here that lets this work: the expected range for Glicko is actually larger than the expected range for ELO. This is from by the constant value of $E(win)$ with respect to rating differences in Glicko that ELO does not exhibit; ELO allows for less numerical distinction at the top levels of play. Because of this, we don't introduce unexpected inaccuracy to the system; the MMR algorithm can just increase ranks as appropriate and the players will spread out, and skew, appropriately.

Deviation RD will be set to $RD = 350$ for everyone. Since we have no concept of certainty embedded, we will just default to generous mobility. This has the benefit of allowing the ranking process as much freedom as possible to handle the initial information discovery that we are relying on to set the initial rating r.

Current Ranking	RP Transform	New Ranking
Currently Grandmaster	$\rightarrow RP = 1200$	Grandweaver
Currently Master	$\rightarrow RP = 1100$	Master
Currently Prophet	$\rightarrow RP = 1000$	Expert II
Currently Scholar	$\rightarrow RP = 700$	Apprentice II
Currently Apprentice	$\rightarrow RP = 400$	Trainee II
Below Apprentice	$\rightarrow RP = 200$	Wanderer

Table 2: The transformation from ELO is based on the achieved ranking. The conversion from ELO to ranking is not so simple as to use ELO for this directly.

Rank RP will be converted by a 'discrete-rounded linear interpolation' (we're copying values in batches) that mimics future season resets. We are using values *observed on the current ladder* as the benchmarks for moving people, so the underlying theory here is as simple as "ELO is better than nothing". Critically, this also means that players currently on the ladder are exactly as eligible for rewards as they were before the migration.

Conclusion

Though the dynamics of r and RP are complicated, the improvements to user experience and balance practices are clear and easy to see. We can now revisit our original goals, more competitive and accurate matchmaking, rewarding consistent play across a season, weekly competition for top spots, and easy across-season climbing. The competitiveness of matches that result from this are immediately improved by shifting to a more accurate calculation than ELO *and* by not completley resetting the matchmaking scores on a monthly basis; we also open the door for long-term improvements by introducing more observable player characteristics across the board, which will help with future balance efforts. The net-positive RP inflation, through B, and more dynamic post-match RP update will allow players to quickly climb to an appropriate rank and continue to make *playing* a match more rewarding than not playing one. Weekly soft resets at the top of the leader board and artificially inflated RD will make climbing always possible, avoiding weekly leader stagnation. Finally, scaling hard resets and completley redrawn RD will increase mobility and give players a consistent and achievable goal at the start of each season, without forcing the ranked ladder into a state that is so chaotic it is difficult to interact with.

There are some notable limitations here; the dynamic nature of the separate systems requires careful tuning. In many cases, these tuning parameters are impossible to estimate in a principled manner. As an example, increasing f_{Min} increases the variability across players in number of matches to climb, primarily at low ranks; this is a non-linear relationship (since it is associated with variance), entirely subjective, and the 'correct' value depends on observed variance in match results of players, given their 'true' rank, which we do not have a reliable estimator for currently. The information to observationally determine a good value for this score simply does not exist; we can (and have) done simulation exercises to find a decent value, but simulation is imperfect, and the large number of tuning parameters like this means that MMR and RP will both be behaving slightly suboptimally in the early days. Nonetheless, these limitations are small and short-lived, especially if we get substantial community feedback to pair with the quantitative understanding of the systems.

Overall, this system should provide a robust, future-proofed way to rank players and continue to provide value in the long run. Though the dynamics are complex, but the maintenance can become easy, even rote, after the initial phase of adjustments. This will result in a ranked leaderboard that players enjoy interacting with and matchmaking system that can help ensure a high quality match at every opportunity for every player.

References

[1] Mark E Glickman. Parameter estimation in large dynamic paired comparison experiments. *Journal of Applied Stastics*, 48(3):377–394, 1999.

[2] Mark E Glickman. The glicko system. *The Glicko System*, Sep 2016.