User Modeling and User-Adapted Interaction (2026) 36:4
https://doi.org/10.1007/s11257-025-09439-z

Balancing stability and flexibility: investigating a dynamic *K* value approach for the Elo rating system in adaptive learning environments

Hanke Vermeiren[1] · Abe D. Hofman[2,3] · Maria Bolsinova[4] · Han L. J. van der Maas[2] · Wim Van Den Noortgate[1]

Received: 30 January 2024 / Accepted in revised form: 12 November 2025

Abstract

In adaptive digital learning environments, it is essential to track learning trajectories. The Elo rating system, known for its computational simplicity, is frequently employed for this purpose. Current Elo-based systems cannot handle rapid changes in ability or are unable to balance accuracy and speed when updating player and item ratings. Changes in Elo ratings depend on the sensitivity parameter $\boldsymbol{K}$. Using fixed $\boldsymbol{K}$ values necessitates a trade-off: larger values facilitate the tracking of evolving ability levels but introduce greater rating volatility. Smaller values yield more stable estimates, but are slower to reflect actual ability levels. Existing modifications of the Elo system, which diminish $\boldsymbol{K}$ as the number of responses increases, are inadequate in scenarios characterized by considerable ability fluctuation, a common occurrence in digital learning environments. To address this challenge, we introduce a novel approach for dynamically adjusting $\boldsymbol{K}$ values in response to observed trends in rating changes. This method increases $\boldsymbol{K}$ during noticeable upward or downward shifts in ratings and reduces it otherwise. We present a computationally efficient implementation of this

✉ Hanke Vermeiren
hanke.vermeiren@kuleuven.be

Abe D. Hofman
a.d.hofman@uva.nl

Maria Bolsinova
M.A.Bolsinova@tilburguniversity.edu

Han L. J. van der Maas
H.L.J.vanderMaas@uva.nl

Wim Van Den Noortgate
wim.vandennoortgate@kuleuven.be

[1] Faculty of Psychology and Educational Sciences, and Imec Research Group Itec, KU Leuven, Kortrijk, Belgium

[2] Psychological Methods, University of Amsterdam, Amsterdam, The Netherlands

[3] Prowise, Amsterdam, The Netherlands

[4] Methodology and Statistics, Tilburg University, Tilburg, The Netherlands

Published online: 07 December 2025

idea and validate its superiority over existing K adjustment strategies through simulation studies. Additionally, we describe the implementation of this adaptive K model in a widely-used digital learning platform, Math Garden, which leverages both accuracy and response time in its assessments. By successfully integrating speed and precision, this innovative implementation enhances the effectiveness of digital adaptive learning environments.

Keywords Online learning · Computer adaptive practice · Elo rating system

1 Introduction

For several decades, extensive research has explored adaptive and personalized learning (Bernacki et al. 2021), including theoretical frameworks (Winne 1989; Hodson 1998), intelligent tutoring systems (Anderson et al. 1995; Nwana 1990), and early field implementations (Koedinger et al. 1997) that demonstrated the potential of tailoring instruction to individual learners. Widespread digitization has allowed to diverge from traditional forms of education that follow a traditional one-size-fits-all strategy. The emergence of adaptive learning systems enabled more personalized learning experiences without putting extra workload on teachers. By providing learners with experiences tailored to their needs, these environments aim to increase motivation and benefit learning outcomes. Adaptive or personalized learning is a broad concept and adaptivity in a digital learning environment can be aimed at different elements of the learning experience, based on stable or dynamic learner characteristics, such as cognitive style or ability levels (Van Schoors et al. 2021; Xie et al. 2019; Bernacki et al. 2021; Pelánek 2025). Most often, adaptive learning involves adjustments to the student's pace of learning by providing appropriate resources, feedback, and support. The benefits of personalized education have been recognized for a long time (Bloom 1984; Zheng et al. 2022; Tlili et al. 2024; Vanbecelaere et al. 2021; Debeer et al. 2021). However, over the years, interest in the field has increased due to the possibilities offered by modern technological advancements (Shemshack and Spector 2020; Kabudi et al. 2021; Er-Rafyg et al. 2024). In the computer adaptive practice (CAP) framework proposed by Klinkenberg et al. (2011), the focus is on adaptive item sequencing, where the degree of difficulty of the items is adjusted in accordance with the student's present level of proficiency. By providing appropriate challenge, the aim is to optimize the learning experience, enhance engagement and motivation, and ultimately foster improved educational outcomes (Vygotsky and Cole 1978; Ericsson et al. 1993; Deci and Ryan 2004; Csikszentmihalyi 2014). The CAP framework is inspired by the practice of computerized adaptive testing (CAT), which commonly uses item response theory (IRT) (van der Linden and Hambleton 2013), linking the probability that a subject answers an item correctly to the item's difficulty and the person's ability. In addition to determining item difficulties, it is essential in CAP to continuously monitor and estimate the ability levels of the learners to provide the items that are most appropriate for their ability levels. Various techniques for determining a learner's ability level have been developed over the years (Pelánek 2017; Pavlik Jr et al. 2009; Desmarais and Baker 2012; Bolsinova, Maris, et al. 2022; Šarić-Grgić et al. 2024; Bolsinova, Deonovic, et al. 2022; Jiang et al. 2023). The majority of them, however, are ill-suited for adaptive learning environments, because they may need pre-calibrated item banks, require intensive estimation techniques, and assume that the person's ability remains constant (Wauters et al. 2010a).

1.1 The Elo rating system

To tackle these issues, Brinkhuis and Maris (2009) and Klinkenberg et al. (2011) proposed the use of a paired comparison rating system that enables learner ability and item difficulty to be updated after each item-person interaction. The Elo rating system (ERS) was originally developed for ranking players in chess tournaments (Elo 1978). It is a straightforward self-correcting algorithm that is easily modified, resulting in several extensions and modifications throughout the years (Glickman 1999; Schölkopf et al. 2007). Due to its intuitive nature, the ERS has known widespread adoption in several domains including online gaming, economics, and social sciences (Véron et al. 2014; Yang et al. 2014; DeLong et al. 2011; Neumann et al. 2011; Sharabi 2022). Also in education, the ERS is often preferred because it is simple, fast, and updates ability estimates online after each response, making it well-suited for adaptive systems. Unlike IRT or Bayesian methods, it requires little computation, scales easily, and is easy to explain, while still giving reasonable accuracy. Although less precise than more complex models such as Bayesian knowledge tracing, for instance, when items are not independent, its transparency and efficiency make it a pragmatic choice for tracking ability in practice.

In an educational context, the ERS is used to update ratings for both the item difficulty and the learner's ability after the learner provides an answer to a new item. Updating ratings at time t relies on the previous rating at $t-1$, a weight (K) that determines the influence of the new observation, and the difference between the actual score and the expected score on the new observation. If a student successfully answers an item, the student wins, which implies the actual score is 1, but if the answer is incorrect, the item wins, meaning the score is 0. The ERS's objective is that when a student fails, his ability rating will drop, and when he wins from the item, the ability rating will rise. The item difficulty ratings are updated in the reverse direction. After both ratings are updated, an adaptive item selection function determines the next optimal item based on the new estimate of the learner's ability level to ensure that learners do not get demotivated. The formulas for updating the ratings of the learner (1) and item (2) parameters are similar. However, it is possible to implement a different K value for items and learners.

$$\theta_{i,t} = \theta_{i,t-1} + K_i \left(X_{ij,t} - E(X_{ij,t}) \right) \tag{1}$$

$$\beta_{j,t} = \beta_{j,t-1} - K_j \left(X_{ij,t} - E(X_{ij,t}) \right) \tag{2}$$

with $\theta_{i,t-1}$ and $\beta_{j,t-1}$ the current ratings of learner i and item j, respectively, $\theta_{i,t}$ and $\beta_{j,t}$ the updated ratings, K the weight assigned to a new response, $X_{ij,t}$ the observed outcome of person i on item j for the current trial and $E(X_{ij,t})$ the expected

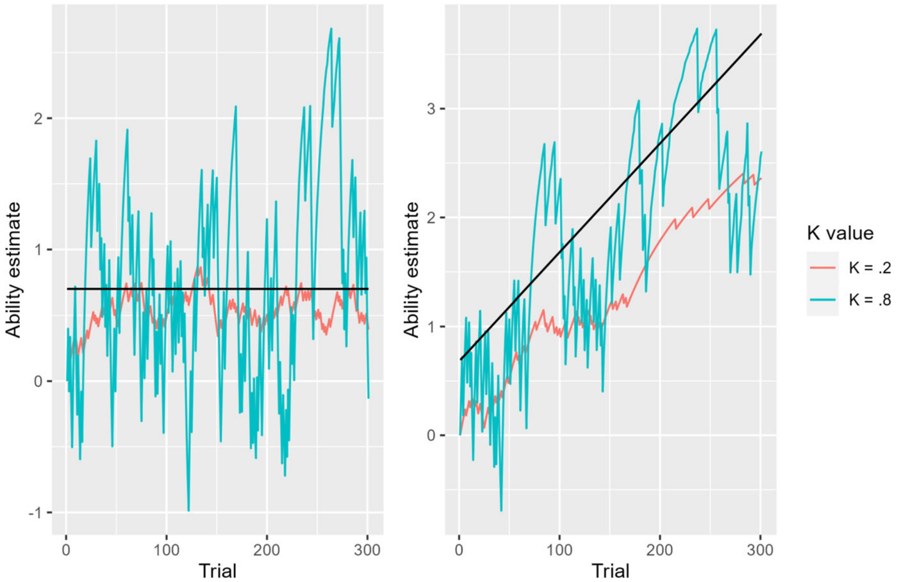


Fig. 1 Visualizations of the static K value problem for stable (on the left) and growing (on the right) ability (black line) using simulated data

outcome.The expected outcome $E(X_{ij,t})$, the probability of a win or a correct answer, is calculated using the Rasch model (3).

$$E(X_{ij,t}) = p\left(x_{ij} = 1|\theta_{i,t-1}, \beta_{j,t-1}\right) = \frac{e^{\theta_{i,t-1}-\beta_{j,t-1}}}{1 + e^{\theta_{i,t-1}-\beta_{j,t-1}}}. \quad (3)$$

In the basic ERS, the outcome is accuracy of the response (0 and 1 for an incorrect and correct response, respectively), but other outcomes can be considered, for example, including both accuracy and speed as suggested by Maris and Van der Maas (2012).

Despite successful implementation in several digital learning environments (Klinkenberg et al. 2011; Pelánek et al. 2017), ERS-based systems still have some challenges. As can be seen in Eqs. (1) and (2), two factors influence the size of the rating update. The first is the difference between the expected outcome ($E(X)_{i,j}$) and the observed outcome ($X_{i,j}$). The second is the parameter K, which determines how much weight will be assigned to the last performance. K values play a key role in achieving a balance between stable skill estimates and the flexibility to dynamically track skill level changes. If the K parameter is too small, it will take the rating system a long time to converge. Too large K values, on the other hand, result in volatile rating estimates. As such, when choosing a K value, a trade-off needs to be made as can be seen in Fig. 1. Moreover, choosing an appropriate K value is a crucial step in dealing with the cold-start problem (Masthoff 2004). The cold-start problem refers to the phenomenon that for new learners or items in a learning environment, we do not have knowledge about their ability/difficulty levels. Therefore, initial starting ratings might differ from the true levels (Pankiewicz 2020; Park et al. 2019; Yudelson 2019; Pliakos et al. 2019). Typically, items receive an initial rating based on expert judgment or data-driven estimates (Park et al. 2019), which in practice works very well. However, another solution is to adapt the K value throughout the learning process allowing to balance the stability of smaller K with the flexibility of larger K values.

1.2 Study aim

In this study, we aim to demonstrate the superiority of an adaptive K value, which is sensitive to changes in ratings over time, compared to fixed K values. By allowing the K value to increase when a change in ability is detected and decrease when ability is believed to be stationary, this method aims to bypass the trade-off between accurate ratings and fast convergence that arises when choosing a fixed K value. We believe that this method allows accurate tracking abilities when they remain (somewhat) stationary, as well as tracking changes in ability levels rapidly. First, we start by providing a brief overview of a number of approaches to adapt the K value that have previously been proposed as alternatives for fixed K values. Next, we describe a computationally efficient function for adapting the K value based on changes in the ratings, and illustrate its use. Using simulation studies, the performance of this function is compared to the use of fixed K values and a K value that decreases over time. To conclude, we describe an extended version of this function as it is implemented in the adaptive learning environment, Math Garden.

2 Alternatives to a fixed *K* value

That fixed weights might be ineffective for the ERS is widely recognized. Elo Arpad (Elo 1978) originally proposed a K value that can take three different values depending on the number of games played. The ERS in the field of chess has seen a lot of adjustments throughout time, and more recent versions now let K values fluctuate based on player ratings and the quantity of games played (Glickman and Doan 2020). Another method to calculate ratings that gained popularity in the chess world and online gaming is the Glicko rating system (Glickman 1999). The Glicko rating system is similar to the ERS but takes into account the uncertainty of the rating by introducing a rating deviation (RD) term into the rating system, which influences the size of the update. If the RD is small, updates will be smaller because the rating is assumed reliable. A large RD, on the other hand, indicates high uncertainty and will result in larger updates. The size of the RD is governed by the number of games played and the time passed between games: the RD goes down the more games played, but goes up again when a large period of time has gone by since the last game. For the mathematics underlying the Glicko rating system, we refer to Glickman (1999, 2001).

In an educational setting, several approaches have been developed for the ERS that allow a dynamic K value. Most of these methods, however, adapt the K value only based on the number of parameter updates, therefore allowing the K value to decrease over time only. For example, Pelánek et al. (2017) proposed a nonlinear function (4) for K for updating the ability and item parameter, based on the number of times the ability and item parameters have been updated (n_i and n_j, respectively).

$$K_i = \frac{a}{1 + bn_i};$$
$$K_j = \frac{a}{1 + bn_j}. \tag{4}$$

The meta-parameters a and b can be determined using grid search.

Wauters et al. (2010b) proposed a logistic function that goes down in function of the number of items a person has answered. In this function, three parameters were varied: (1) the initial K value (K_0), (2) the a parameter influencing the slope of the logistic function, and (3) the b parameter that influences at what point the initial K value is reduced by half. For example, if b is 10, then the initial K value will be reduced by half after 10 item responses.

$$K_t = \frac{K_0}{1 + ae^{bn_j}}, \tag{5}$$

with n_{ij} the number of items completed by a person before answering item j.

Allowing the K value to decrease in function of the number of observations is comparable to the Glicko rating system in chess, where the RD decreases with the number of games played. Similar to the Glicko rating system (Klinkenberg et al. 2011) also include the time between sessions in the learning environment to model the uncertainty U of the rating estimates:

$$U_t = U_{t-1} - \frac{1}{40} + \frac{1}{30}D, \tag{6}$$

where U starts at 1 and D is the number of days between learning sessions. Using this formula, uncertainty for the learners will update to a lower value as more items are answered, but will increase again the more time passes between sessions in the learning environment. The same rationale is applied to the items; the more the items are played, the more the uncertainty value decreases. Uncertainty increases again if items are not played for some time. The logic for the incorporation of time intervals in between learning sessions is that we do not know what happens here to the ability parameter (learning progress, deterioration...), again increasing the uncertainty of the ability estimates. For items, the same function is used; however, it is less likely that a long time goes by before items are played again. As a result, item difficulties are likely to be more stable. Uncertainty is calculated both for item and learner estimates according to Eq. (6). Next, the K value for updating abilities and difficulties is updated using Eqs. (7) and (8), respectively. Note that both updating functions rely on different uncertainty values.

$$K_i = K(1 + K_a U_{i,t} - K_b U_{j,t}); \tag{7}$$
$$K_j = K(1 + K_a U_{j,t} - K_b U_{i,t}), \tag{8}$$

with K = 0.0075, the default value when there is no uncertainty, K_a = 4, and K_b = 0.5, the weights for the rating uncertainty of person i and item j.

3 Adaptive step size

Though an improvement over constant weights, current adaptive functions might still not be optimal. All functions (except the functions proposed by Klinkenberg et al. 2011; Glickman 1999) only update the K value in one direction, and might therefore not be sufficiently dynamic to track ability levels in learning environments, as these are expected to change over time. We advocate that the degree of uncertainty surrounding ratings should reflect changes in the underlying latent variables. To put it another way, we want the K value to go up when the learner's ability changes, but when no learning is occurring, the K value ought to decrease again in order to ensure reliable estimates. Just like Klinkenberg et al. (2011) and Glickman (1999), we propose the use of a K value that can both decrease and increase, but instead of relying on log data (e.g., number of learner-item interactions and time between sessions), we propose dynamic functions which allow adapting the value of K in response to variation in the ratings. Glickman (2022) proposed a similar idea in an improved version of the Glicko rating system. In Glicko-2, the updates of the ratings are still dependent on the rating deviation; however, now the rating deviation is in function of a new parameter named volatility. The volatility parameter represents the degree of expected fluctuation in a player's rating. If a player's performance is consistent, volatility will be low, which will result in lower RD and less change to the rating. Here, we propose a similar approach but aimed at adapting the K value in the ERS.

To demonstrate the principle of a dynamic K value sensitive to changes in ability, we describe a computationally efficient function to adapt K values that can be implemented in the basic ERS formula. The domain of exponential smoothing (Holt 1957; Billah et al. 2006; Ostertagová and Ostertag 2011) served as inspiration for the adaptive K function proposed here. Exponential smoothing is a popular forecasting technique used in time series data analysis aimed at short-term prediction of future events. Exponential smoothing is closely related to the use of the moving average. The difference, however, is that in exponential smoothing the time points of the time series are weighed unequally, with the influence of prior observations decreasing exponentially. While there are numerous variations, the most common technique is the simple exponential smoothing model (Brown 2004) (9), which is formulated as follows:

$$s_t = (1 - \alpha)s_{t-1} + \alpha x_t, \tag{9}$$

with s_t the smoothed value at time point t, α the smoothing factor limited between 0 and 1, x_t the current observation, and s_{t-1} the previously smoothed statistic. As can be seen, the formula only requires one parameter α, which determines the rate of exponential decay of the weights.

For the ERS in a learning context, it can be argued that when the current rating accurately reflects the underlying ability level, positive and negative updates to the rating will be balanced in scenarios with a 50% success rate. As such, we can track changes in the sign of the updates to determine when the K value should increase. To do this, we let the K parameter (11) depend on a trend parameter (T) which is based on variation in the signs of the updates to the ratings (10). More specifically, if ratings are positively updated, the trend parameter of the rating will increase toward

1. On the other hand, when ratings are negatively updated, the trend parameter will go down toward -1. A downwards or upwards trend in the ratings indicates that the ratings do not accurately reflect the true ability levels. As such, when positive and negative updates alternate, the T value will oscillate around 0. The trend parameter starts from 0 for a new learner and is updated after every observation as follows

$$T_t = (1-\alpha)T_{t-1} + \alpha \times \mathrm{sign}(X_{ij,t} - E(X_{ij,t})) \tag{10}$$

with T_{t-1}, the trend value of the previous trial, with T_t, the newly updated trend value, and α, the parameter determining how large the updates to the trend value are. The part $\mathrm{sign}(X_{ij,t} - E(X_{ij,t})$ will always be either 1 or -1 depending on whether the learner answers the item correctly ($X_t = 1$) or incorrectly ($X_t = 0$) and determines whether the trend parameter gets updated upwards or downwards. Note that $E(X_{ij,t}$ is never exactly equal to 0 or 1, and therefore $(X_{ij,t} - E(X_{ij,t})$ will never equal 0 or 1. The K value at time point t is dependent on the value of the trend parameter at time point t and calculated according to (11).

$$K_t = |T_t| \tag{11}$$

The K value is function of the T value only. Then, the K value before any items are made (i.e., K_0 value) is set to 0. As soon as the first item is made, the K value to update the rating will be calculated. Given Eqs. 10 and 11, and since $T_0 = 0$, K_1 will always be .2 for every new learner.

By incorporating the $sign(X - E(X)$ component in the formula, we allow the T value to both increase or decrease depending on the current observation. The rationale is that when an individual's rating is around his actual ability level and item selection is aimed at a 50% success rate, they have an equal probability of providing a correct or incorrect response. As such, updates of the T value will alternate between positive and negative, allowing it to slowly converge to zero. Consequently, a small K value will be reached when no trend in the ratings is detected. On the other hand, when the learner's rating is higher or lower than their true ability level, we expect the percentage of correct or incorrect answers to diverge from .5. As a result, the T value will move toward 1 or -1 (depending on the direction of the trend in the ratings), resulting in an increase in the K value.

In exponential smoothing, the smoothing constant (α) is between 0 and 1, with larger values giving more weight to the most recent observations. If the smoothing parameter is one, only the most recent observation has weight and previous observations have no impact on the prediction of the model. As a result, the predictions of the smoothed time series are equal to the original data of the time series. If the smoothing parameter is 0, the series is smoothed completely flat. The larger the smoothing parameter, the faster it is able to respond to changing trends in the data. In our case, if α for the adaptive K function were 0, the trend parameter and as a result the K value would remain unchanged. Here, α determines how fast the T value and as a consequence the K value react to changes in the ratings. Larger α values result in larger updates to the trend value, which is a desirable trait for an adaptive method that is data-driven. However, larger α values also result in larger oscillations of T around zero, meaning that in moments of the ratings being stationary the K value will be larger, as visible

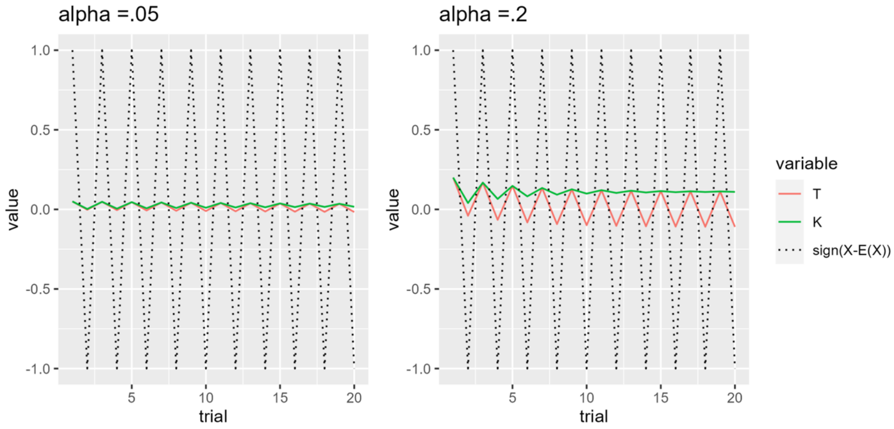


Fig. 2 Visualization of the effect of different alpha values on the updates of the T and K value

in Fig. 2. Note that in Fig. 2 a scenario is illustrated where responses continuously alternate between correct and incorrect. In real life, answering items might result in longer sequences of incorrect or correct answers, even when the rating accurately reflects the true ability level. This is another reason to be cautious with implementing large α values to avoid unwarranted changes in the K value.

4 Simulation study

4.1 Method

4.1.1 General setup

A simulation study was done in order to evaluate the newly proposed method and compare it with the performance of fixed K values. We simulate an adaptive learning environment where items are selected for the learners based on their current ratings, and ratings are updated after every response according to the basic ERS formula. For item selection, a success rate of 50% was implemented. To keep the simulation simple, item difficulties are assumed to be known (e.g., item ratings are not estimated throughout the simulation). Furthermore, ability is assumed to remain constant over time. Both item difficulties ($N = 800$) and learner abilities ($N = 500$) were simulated from a normal distribution ($N(0, \sigma^2)$). To ensure enough item options for learners toward the ends of the distribution, it was decided to implement a larger variance for the item distribution ($\sigma^2 = 3$) than for the learner distribution ($\sigma^2 = 2$). An algorithm implementing the ERS was set up, with the expected outcome of a person for a certain item calculated as in (3) using the Rasch model. To determine the outcome of the person-item interaction based on the expected outcome, values were sampled from a Bernoulli distribution with π equal to the expected score from the Rasch model. Updates of the learner rating estimates are implemented as in (1). Item selection was done using the beta distribution to ensure a success rate of .5. To do this, the following method is used:

a) Determine shape parameters a and b with μ = .5 and σ^2 = .01
b) Decide probabilities of the items using the beta density distribution $\beta \sim Beta(a, b)$
c) Use these probabilities to sample an item

To simulate a cold-start problem, initial ratings were randomly drawn from the same distribution as the true ability levels.

Simulations were run for a range of fixed K values as well as a range of α values for the adaptive method. For the fixed K value, we looked at values between .2 and .8. This is justified by the fact that K values higher than .8 will rarely be implemented in practice. Furthermore, since the goal of our adaptive method is to achieve a balance between convergence and accuracy, we are not interested in the most extreme values possible for K since those are the ones that by definition are not able to achieve this balance. To establish the superiority of the newly proposed method, we were more interested in the K values that are large enough to react to changes while still able to result in accurate ratings when the ability is stationary. For the α parameter of the adaptive function, values were limited between .005 and .25. As explained earlier, this parameter influences how large the updates to the trend parameter will be. Smaller α values result in smaller updates of T and as such a slow increase in the K value, larger values result in bigger updates, thereby also increasing the lower limit for the K value when ratings are stationary (see also Fig. 2). For all parameter values, the simulations were repeated 10 times with different seeds to obtain scenarios with different starting ratings, ability, and difficulty values.

4.1.2 Procedure

The performance of the fixed and adaptive K methods was compared in terms of convergence and accuracy of the estimates. It is expected that when using a fixed K value, larger K values will result in faster convergence but larger estimate errors, while lower K values will result in slower convergence and smaller error values. With this in mind, mean squared error (MSE) values were recorded for each trial. The MSE values were used to determine when the ratings become stable over time (e.g., when they converge). The simulation was designed to run until this point of convergence was reached. When convergence is reached, the convergence point and the measure error value at this point in the simulation are saved and can be used to compare the methods.

Mean square error To determine the accuracy of the rating at the point of convergence, the MSE is used. The MSE measures the squared difference between the estimated values (i.e., the Elo ratings, $\hat{\theta}$) and the true values (θ) averaged over persons:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(\hat{\theta} - \theta)^2. \tag{12}$$

The smaller the MSE, the smaller the error in the prediction made by the implemented ERS algorithm. Both θ and $\hat{\theta}$ are centered over persons before the MSE values for each time point are calculated.

Determining when to stop the simulation: The Kwiatkowski–Phillips–Schmidt–Shin test To determine the point where ratings become stable (i.d. there is no longer a downward or upwards trend) as well as when to stop the simulation, the Kwiatkowski–Phillips–Schmidt–Shin (KPSS) (Kwiatkowski et al. 1992; Shin and Schmidt 1992) test was used. This test was developed to test stationarity in time series data. The KPPS was implemented as follows in our simulation. After the first 20 trials, a moving window of 20 trials is implemented to test for stationarity on the MSE values. If the test is insignificant for 10 consecutive windows, convergence is reached and the simulation is terminated.

Measure of convergence Since the KPPS test was developed to determine stationarity of time series data, the test is too rigid for our purposes. In learning environments, we are interested in the time point where the rating converges for the first time, instead of the time point at which ratings have been stationary for some time (see also Fig. 3). As such, a second measure of convergence was implemented. For this, a more heuristic approach was adopted. Starting from the KPPS point of convergence, the mean and standard deviation of the MSE values from the last 20 trials were calculated. An upper limit is calculated by taking the mean plus two times the standard deviation. Next, we determine the earliest point in time for which the MSE is below this upper limit, which is stored as a point of convergence. We chose to implement this method for clarity. While the KPPS test overall gives the same general outcome, it results in more outliers and thus more variance. By using this measure in our further analysis, we aimed to enhance interpretability.

4.2 Results

4.2.1 Balance of accuracy and convergence

To determine the performance of the two methods, the interaction between the convergence measure and the MSE for the different parameter values in different scenarios was plotted. First and foremost, Fig. 4 illustrates the expected trade-off between convergence and estimation error for fixed K values. For the lower end of the range of K values, we find slower convergence but smaller MSE values, while the larger the K value gets, the faster the convergence but the larger the MSE values become.

For the adaptive K value method proposed here, we expect a better balance between convergence and MSE values. To demonstrate the superiority of our adaptive K method, simulation results for the adaptive K value method and fixed K were plotted together. If the adaptive method outperforms fixed K values, the results should be situated on the left side of the results from the fixed K values. Indeed, in Fig. 5 we can see that the new method outperforms fixed K values. The different dots represent the outcomes (e.g., the balance between convergence and accuracy) for different seeds and different K or α values. It is clear that in most instances, the adaptive function outperforms a fixed K value. To get a better insight into how the two methods compare

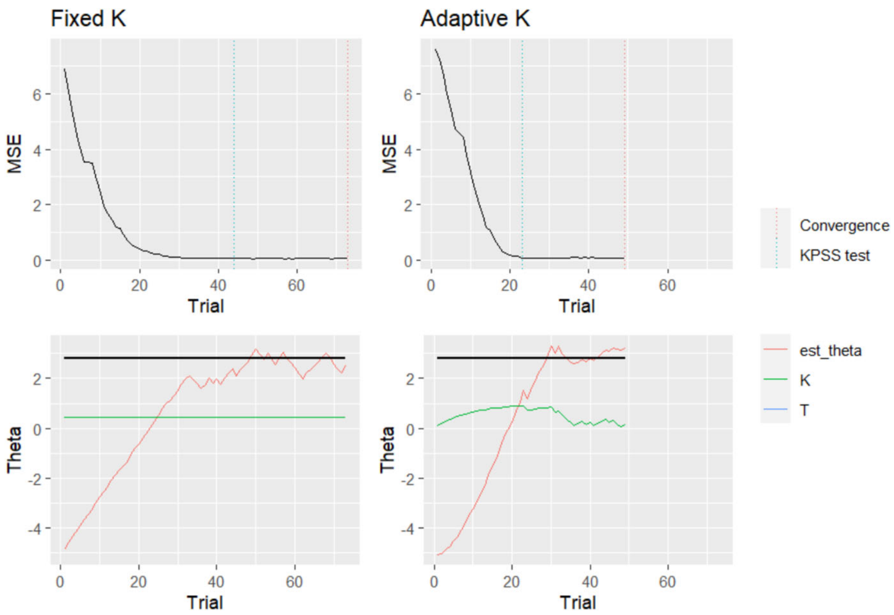


Fig. 3 Visualization of the trajectory of the MSE values across persons (row 1) and visualization of the trajectory of the rating toward true ability level (black line), K and T for one person (row 2)

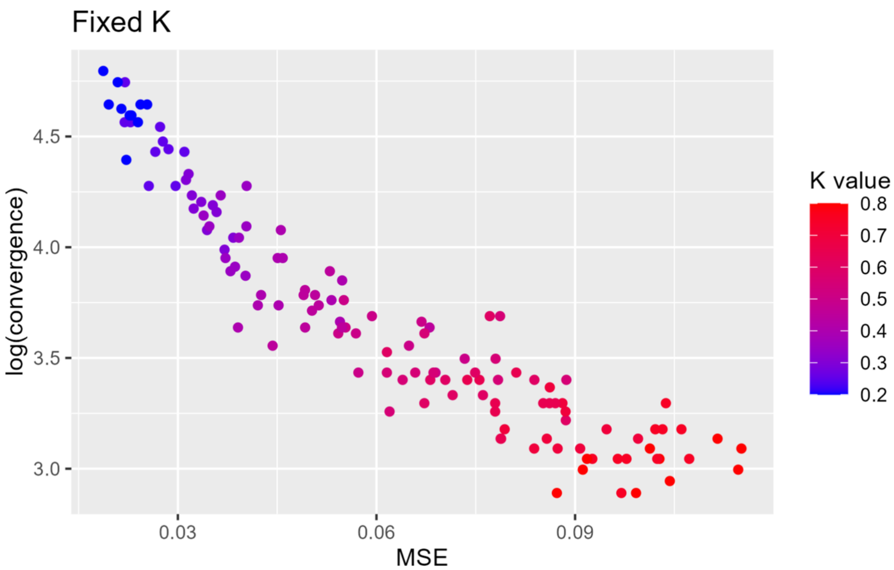


Fig. 4 Visualization of the effect of fixed K values on the balance between convergence and MSE for multiple iterations

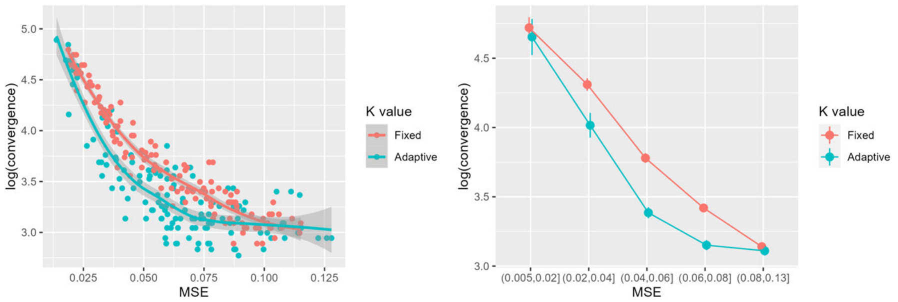
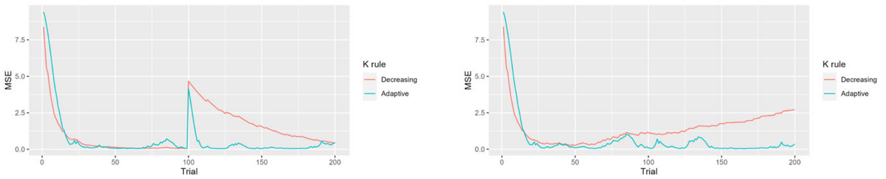


Fig. 5 Comparison of the performance of the adaptive method and performance of fixed K values in terms of convergence and MSE

to each other, we also look at the mean convergence value for different ranges of the MSE value. As can be seen, for most of the different MSE values, a clear difference can be found between the two methods with the adaptive method outperforming fixed K values. In addition, we performed a statistical test that examined whether the convergence differed between the two K value methods while controlling for differences in MSE using an analysis of covariance (ANCOVA). Convergence was the dependent variable, K value method was the fixed factor, and MSE was included as a covariate. The ANCOVA revealed that the K factor method had a significant effect on convergence after controlling for MSE, $F(1, 257) = 22.23$, $p < .001$. Estimated marginal means showed that at the same level of MSE, an adaptive K converged faster than a fixed K, with adjusted means of 35.6 (95% CI 32.9–38.2) and 44.6 (95% CI 41.9–47.3) iterations, respectively. Note that toward the end of the range of MSE values, the two functions show overlap. This is to be expected, since our simulation was designed in such a way that the earliest possible point of convergence is trial 21 (log(21) = 3.04). As such, our function can no longer outperform the fixed K values in terms of convergence.

Figure 3 compares both methods in terms of convergence and gives a clear illustration of the problem encountered when using the KPPS test for convergence. For the fixed K, a value of .45 was chosen. In Fig. 4, it can be seen that values between .4 and .5 reach the optimal balance between accuracy and convergence. For the adaptive method, the optimal parameter for α is around .1. It is clear from the upper panels (Fig. 3) that convergence is still a bit faster for the adaptive K function (both for convergence calculated according to the KPPS test and the more heuristic approach). The bottom panels illustrate the trajectory of the rating throughout the simulation for one person. It concerns the person with the start rating, farthest from their true ability level. It is apparent that in terms of convergence, both methods seem to do equally well, but the adaptive method maintains more stable ratings when converging. Furthermore, the functioning of the adaptive method is evident, as the ratings approach the true ability level, the trend value goes toward 0 resulting in smaller K values.

We did not include the previously proposed uncertainty functions in our simulation. The reason for this is that in the case of constant ability levels, a decreasing K value function (given optimal hyperparameter values) outperforms fixed K values and most likely also the proposed adaptive function. However, these functions have a substantial



(a) Scenario with a sudden change in ability level

(b) Scenario with a gradual change in ability level

Fig. 6 Visualization of the trajectory of the MSE for two different ability progression scenarios: **a** a sudden jump in ability at trial 100 and **b** a gradual increase of .02 for each trial

drawback in terms of implementation in learning environments. Since learners have a diversity of learning trajectories, it is difficult to determine optimal hyperparameter values that will work in every situation. The main obstacle is that at some point the K value becomes too small to keep track of changes in ability. To demonstrate this, Fig. 6a illustrates the evolution of the MSE in a scenario with a sudden change in ability for our adaptive function as well as a decreasing K function (4). For (4) a was set to 1 and to determine the optimal value for b, the same simulation as described in the method section was done. It was found that for b between .1 and .2, the best balance between accuracy and convergence was obtained. As such, it was decided to run the simulation for $b = .15$. For the adaptive function, α was set to .1. To simulate the sudden jump, for all persons from trial 100 on, the true ability value equals their initial value plus 2. It is evident that both functions perform equally well in responding to the cold start in the first part of the simulation. However, when halfway through the simulation a jump in ability is introduced, the decreasing K function takes notably more time to converge due to the small K values at this point. Note that the larger the change in ability and the smaller the K value at this point, the worse the performance of these functions. It is evident that when ability remains constant and ratings converge, implementation of a decreasing K function results in more stable estimates. The reason for this is the volatile nature of the adaptive function proposed, the function is implemented in such a way that it immediately reacts to changes in ability even when this is unwarranted. In Fig. 6b , we simulated a scenario where the ability level gradually increases by .02 at each trial. In terms of the initial cold start, both functions show a fast improvement in the MSE, however, as time progresses, the decreasing K function starts to again perform worse, while the adaptive function maintains a more stable MSE value. The reason for this is that the more items a person answers, the lower the K value becomes in the decreasing K function. However, since the learner continues to get better (i.e., a gradual increase in ability levels) the K value will become too small to keep up with the increasing ability levels.

5 Extended implementation in prowise learning environments

5.1 Math garden data

The adaptive learning environments of Prowise (www.oefenweb.nl) implement a CAP framework developed by a team of researchers at the University of Amsterdam (Klinkenberg et al. 2011). The environments allow learners to practice language and mathematical skills with the goal of beneficial learning outcomes. To achieve this, the learning environments aim to provide learners with appropriate item difficulties to support learning outcomes and keep learners motivated. In a learning environment, the goal is to motivate learners to keep practicing their skills; therefore, often a success rate of .75 is aimed for. In the Prowise environments, children have some agency and can change the difficulty level they want to practice. Specifically, they can choose between three levels: easy (about 90% correct), medium (about 75% correct), or hard (about 60% correct). In other words, items are selected such that on average the probability of answering the items correctly equals the chosen success rate. Furthermore, game-based elements are implemented to keep learners motivated and the learning process more engaging. The learning environments consist of multiple games, each measuring one specific language or math skill. Since the basic ERS is unidimensional for each of these games, a separate rating scale is used. Each game consists of items with differing difficulty levels. In the CAP framework, the basic ERS formulas are extended since now the outcome and expected outcome are modeled, taking both accuracy and response time data into account. To do this, a scoring rule (Eq. (13)) balancing accuracy and speed is implemented that rewards quick correct answers and assigns a fixed domain-specific penalty for wrong answers.

$$S_{ij,t} = x_{ij,t}(d_j - t_{ij}) - (1 - x_{ij,t})P, \tag{13}$$

with $x_{ij,t} \in 0, 1$ the response accuracy, t_{ij} the response time of person i for item j, and d_j the deadline (which is scaled to 1 such that t_{ij} falls in between 0 and 1) associated with item j. When the item is answered correctly, the score S equals the remaining time $d_j - t_{ij}$. For example, if it takes 10 s to answer an item with a deadline of 20 s, t_{ij} and d_j would, respectively, be .5 and 1. Plugging these values into the formula gives you a score of .5. When an item is incorrectly answered (i.e., $x_{ij} = 0$), the score equals a fixed penalty $-P$. To calculate the expected score, (14) is used with $\xi_{ij} = \theta_{i,t-1} - \beta_{j,t-1}$:

$$E(S_{ij,t}) = \frac{-P\xi_{ij}^2 e^{-P\xi_{ij}} + \xi_{ij} e^{\xi_{ij}} - e^{\xi_{ij}} + 1}{\xi \left(e^{\xi_{ij}} - 1 + \xi_{ij} e^{-P\xi_{ij}}\right)}. \tag{14}$$

For more information on the derivation of this equation, we refer to Vermeiren et al. (2025). As a result, correct answers can also result in a decrease in the ability estimate if the response time is too slow and the score becomes smaller than the expected score. By implementing this in the ERS, this results in a more balanced ratio of positive and negative updates.

Since the Prowise learning environments uses a slightly different ERS algorithm as is used in the simulation (i.e., it does not rely only on accuracy and both item and learner ratings are updated), the previously described adaptive method was altered to accommodate these differences. Again, the K value is dependent on the absolute value of the trend parameter, which depends on response variations. When a learner scores better than expected ($S > E(S)$), T increases (toward 1). On the other hand, when a learner scores worse than expected ($S < E(S)$), T will decrease (toward -1). The trend parameter values are calculated separately for person i (15) and for item j (16):

$$T_{i,t} = (1-\alpha)T_{i,t-1} + \alpha \times \text{sign}(S_{ij,t} - E(S_{ij,t})); \tag{15}$$

$$T_{j,t} = (1-\alpha)T_{j,t} + \alpha \times \text{sign}(E(S_{ij,t}) - S_{ij,t}); \tag{16}$$

The difference with the adaptive function from the simulation is in how the K values are determined. Updating the K value is similar for ability (17) and difficulty (18) levels, except that for ability, for the 20 first trials of a new learner in the environment, an extra value (K_{start}) is added to the trend parameter in order to calculate $K_{i,t}$. This is done to better deal with an initial cold start. In addition, for both learners (17) and items (18) a minimum value for the K value is implemented:

$$K_{i,t} = \begin{cases} \max\left\{0.2,\ |T_{i,t}|^M + K_{\text{start}}\right\}, & \text{if } t \leq 20, \\ \max\left\{0.2,\ |T_{i,t}|^M\right\}, & \text{if } t > 20. \end{cases} \tag{17}$$

$$K_{j,t} = \max(.001, |T_{j,t}|^M), \tag{18}$$

with $\alpha = .2$, $K_{\text{start}} = .5$ and $M = 5$. α determines how fast the uncertainty adapts to rating deviations. M is used to suppress large absolute T values, resulting in smaller K values. K values are restricted between .001 (for items) or .2 (for learners) and 1[1]. T is restricted between 1 and -1 and starts initially with a value of 0. Note that the rationale of alternating between positive and negative updates does not hold up for adaptivity rules higher than .5 when only accuracy data are taken into account.

5.2 Illustration of the adaptive *K* method implemented in Math Garden

To demonstrate the performance of an extension of the adaptive K function, data of the game *addition* were used. The addition game consists of items where students are given a sum (e.g., 2+7) and need to type in the correct answer as can be seen in Fig. 7. We limited ourselves to data from September 2022, which is the first month after the summer school break in the Netherlands. Children generally neglect to log in to learning environments throughout school breaks. The skills of kids might improve or deteriorate during this time, which can cause estimates to become inaccurate. The dataset contains answers from students from different grades and consists of 26,694 learners.

Figure 8 illustrates how the T and K parameters of this function respond to variations in the update's sign. In the first plot, the data concern a learner new to the "*addition*" game. In the Math Garden, learner ability is initially estimated based on a number of variables (such as age, grade, etc.); however, in this particular example, the data suggest that the estimate might still have been too low. The items are too easy for the learner, which results in the updates consistently being positive. As a result, the trend value increases, resulting in a higher K value, which allows faster adjustment

[1] These values were determined using a/b testing in the learning environment and might not apply to every context.

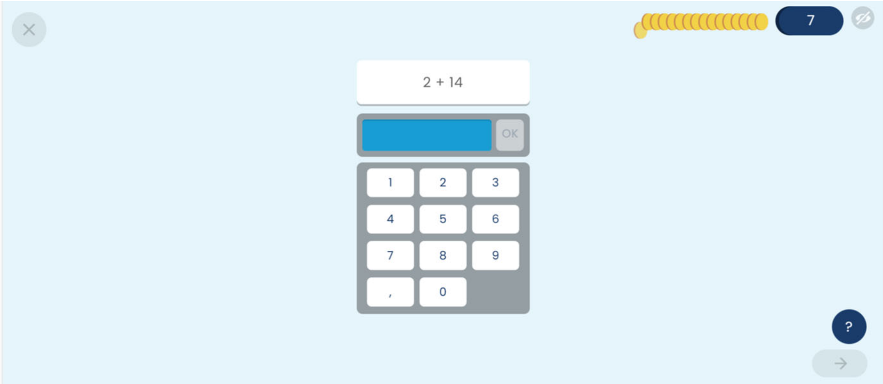
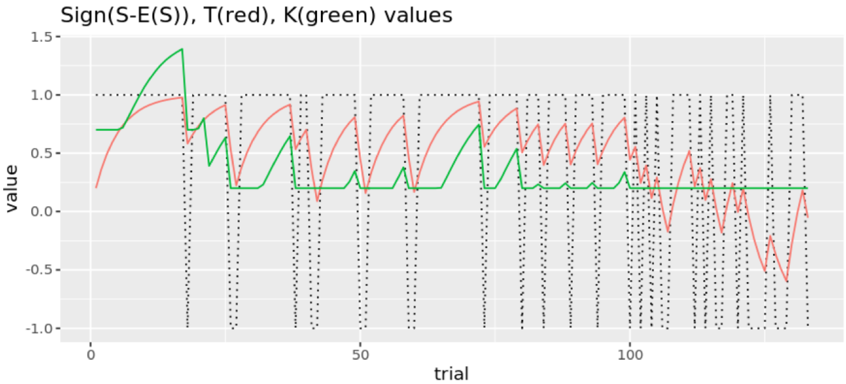


Fig. 7 Example of an item in the addition game of the Math Garden environment

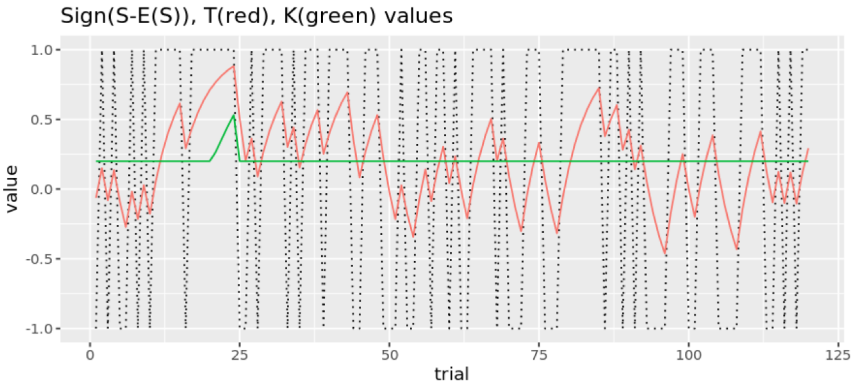
of the ratings. Additionally, the K value is boosted by an additional .5 for the first 20 trials because the learner is new to the learning environment. Finally, we observe that the positive and negative updates start to alternate more evenly after trial 100, suggesting that the rating is approaching the ability level.

In the second plot, we have a student who has previously played the game. From the start, positive and negative updates alternate, and the K value is around .2 (the minimum K value for an individual), indicating that initially this player's rating seems to be accurate. However, in the second session (each session is 10 items) we start seeing a gradual increase in the trend value because the rating is consistently updated in an upward direction. After a while, trend values start to level out once more, and for the remaining part of the time, K stays at its lowest value. In the third example, we have a learner whose data suggest a decline in ability throughout the summer, indicated by multiple trials resulting in a downward update to the rating. As a result, the trend parameter decreases, however, not enough to have a strong effect on the K value. After some time, the dynamic reverses and a series of trials with positive updates occurs.

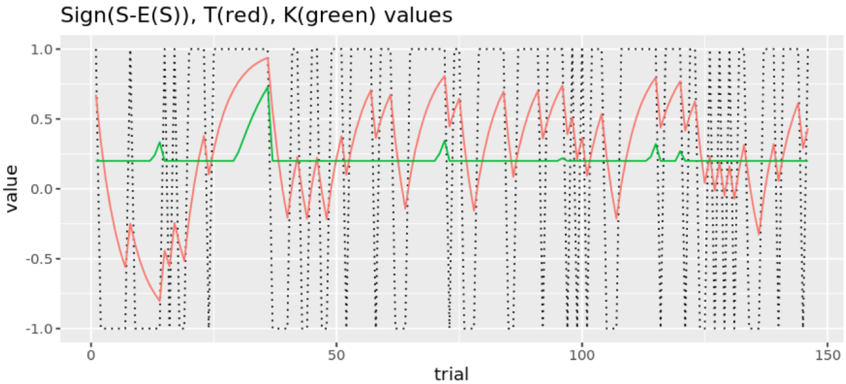
For Fig. 9, a random item and person from the dataset were sampled. The plots serve to demonstrate how the K function is responsive to variations in the parameter estimates. Note that item difficulty estimates remain relatively stable over time compared to ability estimates (plot A). The different effects of item and learner parameters on the K function are illustrated in plots B and C. The trend parameter in plot C shows considerable fluctuations for both item and person ratings; however, it is clear that for the item, these are more centered around zero, indicating that the item parameter estimates are reasonable. This difference, however, is less clear in the K values as a result of the way the adaptive function is implemented in Math Garden. For persons, a minimum K value of .2 is implemented, which gives the illusion of the K value remaining stationary in plot B. Lastly, plot D illustrates the dynamic interplay between the evolution of the ability estimates and the K values. Initially, high K values allow for faster convergence, and when ratings converge to the ability level, K values go down again, resulting in relatively stable ratings. We say relatively stable since the ERS is inherently noisy and the function implements a minimum K value of .2 for learners, which will still cause some minimal fluctuations.



(a) Learner who was new to the game.



(b) Learner who has previously played the game with an accurate rating estimate.



(c) Learner with a decline in ability throughout the summer.

Fig. 8 Visualization of the interplay between the sign of the rating update, the trend value, and the K value for three different students

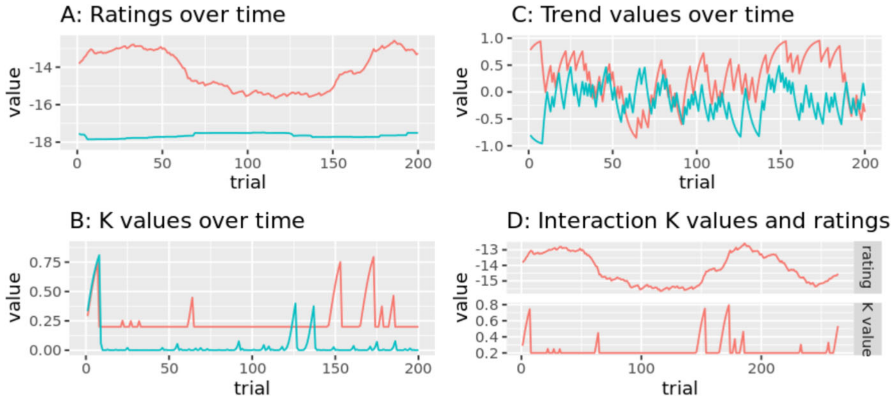


Fig. 9 Illustration of how the adaptive K function implemented in Math Garden works. Red indicates person parameters, blue indicates item parameters

As we observed in Fig. 9, the item's trend values appeared to oscillate around zero, resulting in low K levels. Since items are not expected to change much over time, this is the ideal situation. However, for some items it is found that trend parameters do not fluctuate around zero, causing more variability in the K value, as can be seen in Fig. 10. Despite this, large K values are suppressed by the power term in Eq. (18). Therefore, although the T and K parameters for items may fluctuate, item ratings continue to be rather stable. It is important to note that trend values for learners tend to be positive, whereas they are more inclined to be negative for items. One explanation is that, on the whole, learning occurs for most students, leading to more positive updates to learner ratings and more negative updates to item ratings. Another explanation for this might be that despite the speed-accuracy trade-off being used, positive and negative updates do not balance each other out evenly, and the higher success rates still have some influence on the ratio of positive to negative updates. Lastly, the student has ratings that have remained mostly unchanged over time. Since the learner is new to the game, a large K value is implemented for the first few trials. Later on in the learning process, however, the K values drop to .2 and hardly deviate from this, allowing stable ratings.

Lastly, in Fig. 11 we provide an illustration of how the trend values (T) from our proposed adaptive K function can be used for further analysis. Figure 11 demonstrates how the trend parameter behaves over different grades in different Math garden games. Here, we see the mean absolute trend for grade 3 (6-year-olds) to 8 (10-year-olds) for the following games: (1) Addition, (2) Subtraction, (3) Multiplication, (4) Division, and (5) a mix of items on all of these skills with a shorter response time. Analyzing this new parameter can tell us more about the trends we observe in learners' answers over time for different skills. It is clear from Fig. 11 that higher grades show higher mean absolute trends, which indicates more changes in ability levels for older learners. In addition, there seems to be some variation across games. For example, in the addition game, the mean absolute trend seems to stagnate over time. Note that this does not mean that the addition game is easier since the trend parameter only tells us something about changes in ability ratings and thus whether learning takes place. Rather, this pattern

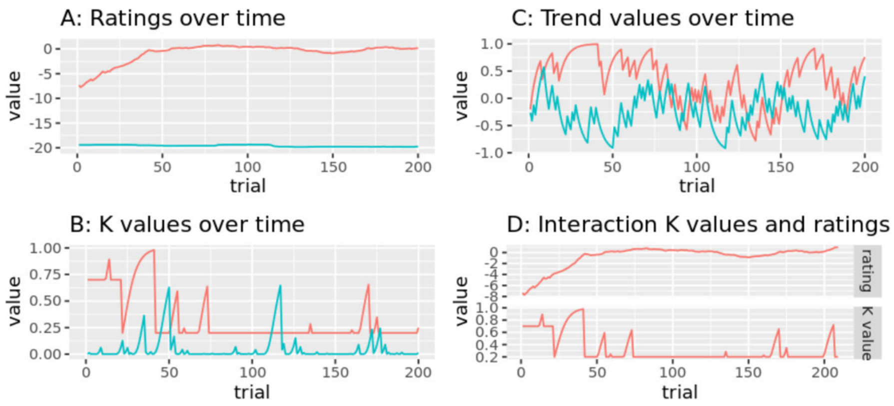


Fig. 10 Illustration of how the adaptive K function implemented in Math Garden works. Red indicates person parameters, blue indicates item parameters

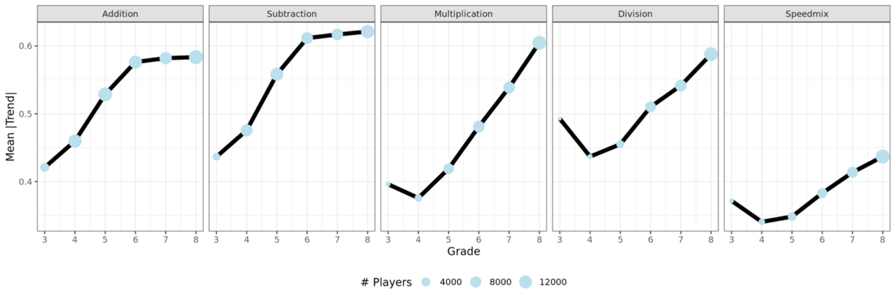


Fig. 11 Illustration of the mean absolute trend per grade for five games in Math Garden

is most likely due to a combination of more accurate ratings for older learners, and the fact that some of these learners already have reached high skill levels on this game and as such do not improve much anymore.

6 Discussion

In recent years, interest in the ERS as an effective method for parameter estimation in adaptive learning environments has grown. Despite some limitations such as assuming item independence, its simplicity and capability to continuously track both item difficulties and learner ability, makes it an ideal alternative to more intensive estimation techniques that often come with assumptions and requirements that are hard to uphold in an adaptive learning environment. Although the ERS has been successfully implemented, it still faces some challenges. One of these challenges is the rigidity of a fixed K value. The K value, a weight that governs the sensitivity of the rating update to the latest observation, plays a key role in achieving a compromise between fast convergence and obtaining stable ratings. Over the years, a number of approaches have been put forth to adapt the K value throughout the learning process in order to address this flexibility–stability trade-off. Using uncertainty functions, the K values are adapted based on the number of observations and the time spent between sessions in the learning environment. Mostly, those functions start with an initially large K value that decreases over time. The rationale behind this is that the more time the learner spends in the learning environment, the more information we have about their abilities, and the lower our uncertainty about the ratings is.

The need for balancing the choice of faster escaping the cold-start problem or accurate ability estimates is solved by previously proposed functions that allow the K value to decrease over time. However, these functions might encounter problems when implemented in learning environments where sudden changes and trends in the data are expected. As such, a continued decrease in the K over time might ultimately result in a too-low weight to keep track of the true ability levels. In order to enable the K value to respond to changes in the learner's underlying ability, we set out to create a novel approach. To accomplish this, we rely on the variation in the signs of the updates to the rating. When item selection is aimed at a success rate of 50%, updates alternating between positive and negative indicate that the rating is close to the actual level of ability. We propose a method inspired by the method of exponential smoothing that allows the K value to increase when a change in ability is detected and to decrease when ability remains constant. Our simulation study demonstrated a superior performance of the function compared to fixed K values in terms of achieving a balance between convergence to true ability levels and accuracy of the ratings.

We acknowledge that there are still some limitations to the design of the simulation study. First and foremost, we want to emphasize that the method we propose here is a simple and intuitive approach to illustrate our message. Extensions and alterations might be needed to deal with certain problems that might be encountered in practice or specific scenarios. Concerning the simulation study, it was decided to keep the item difficulties fixed. This was done for practical considerations such as simulation time and clarity. We, however, do not expect that estimating item difficulties will result in different results. In the implementation of Math Garden, both for item and learner ratings, K values are determined based on a rating-sensitive method. Note that the method we propose is not possible for a player–player system (such as a chess system), as it will result in inflated K values for those at the ends of the distribution. As those players will only be matched to players that are better/worse than them, the K value will keep increasing unwarranted. In learning environments, this problem is resolved by ensuring a large enough data bank with enough item options for those at the ends of the distribution.

Next, in our main simulation, ability levels were assumed constant, and item selection was done with a .5 adaptivity rule. In regard to the former, our results show that the proposed method is able to adapt the K value in response to changes in ability levels, as indicated by larger trends and K values resulting from the cold start at the start of the simulations. Furthermore, the Math Garden implementation, as well as a simple simulation comparing the adaptive K and the decreasing K function shows that our method is better equipped to track learning trends. A success rate of .5 was chosen to illustrate the performance of the proposed methods. We acknowledge that higher success rates will require some alterations of this method, since the number of correct and incorrect responses will not balance each other out when ratings approximate ability levels. As such, in such scenarios, the trend value will not be able to converge to zero when the ability remains constant. A solution to this problem might be to involve reaction times in calculating the expected and observed outcome. By punishing slow correct responses, the scoring rule implemented in Math Garden aims to balance positive and negative updates to the ratings.

In our main simulation study, the proposed method of rating-sensitive K values is only compared to constant values and not the decreasing K functions proposed in the past. We propose the adaptive method as an alternative to a fixed K value, which allows balancing a fast escape of the cold start with more stable estimates later on. It is true that decreasing K functions are proposed for the same reason. If these functions are well defined (e.g., optimal values for the meta-parameters are chosen), they might outperform our adaptive method in a scenario with a fixed ability. The reason for this is that while the decreasing K function results in a K value that gets smaller over time, in the adaptive method the K value might from time to time increase again in response to small trends that do not necessarily reflect a real change in ability. However, in a learning environment, where learners have a variety of learning trends, differ in elapsed time between sessions, time spent working on the skills outside the environment, and numerous other factors, it becomes troublesome to decide on parameter values for these methods that will result in desirable results. By relying on variation in the ratings to adapt the K value, the newly proposed method bypasses these problems. A simple simulation (Fig. 6) where a sudden/gradual change in ability is implemented shows that indeed our proposed method can keep track of changes in ability, which can cause problems for the decreasing K function when the K value becomes too small. We acknowledge that our proposed function is not able to reach the level of accuracy of a decreasing K function when the ability is constant. However, it might be possible to extend or adapt the function proposed here to take this into account. A possible way to do this is to take into account more trials when detecting a trend in the responses to avoid significant changes to the K values as a result of trivial trends in the data. Lastly, the results reported here were for a specific simulation scenario. However, the simulation was repeated for different scenarios (different start values, larger item and person pool, and different variations for the ability and difficulty distributions) with similar results. Readers can find the results and the simulation code on OSF. It was found that a smaller variation in the distribution of ability levels results in a smaller difference between the methods; however, one could argue that the smaller the variation gets, the less advantageous it becomes to implement adaptivity in the learning environment.

The objective of this study was to evaluate the effectiveness of a new adaptive K approach that alters the K value in response to changes in the learner's ratings. Our study highlights the pitfalls of depending on fixed K values for adaptive learning implementations. We acknowledge that the proposed function, which solely relies on accuracy data, still has limitations, particularly when modeling systematic change in adaptivity scenarios with a success rate higher than .5. However, the performance of an extended version used in Math Garden demonstrates that our suggestion of modeling change in the K values based on change in the parameter estimates is fruitful and likely better suited than existing methods. Furthermore, since the trend values are updated after every observation, and answering items still has some randomness to it, unwarranted increases of the K value are possible. An option for this might be to develop a method that looks at a larger sequence of ratings before changing the K value. However, methods that apply a (moving) window are likely to be more complicated to implement in practice.

The idea of letting the size of the rating updates depend on fluctuations in the ratings is also implemented in the Glicko-2 for rating chess. A future direction might be to compare our proposed method with the Glicko-2. Note that this would require some alterations to make it suitable for learning environments (Park 2021). The benefits of the method proposed here are its simplistic and intuitive nature, as well as only a single parameter (α) that needs to be estimated. We hypothesize that there are several possible approaches to implement our proposal of using rating-sensitive K values in the ERS. Future research avenues should be aimed at testing rating-sensitive K values in different scenarios and learning environments, as well as investigating possible extensions and alterations to deal with the limitations it encounters.

Author Contributions All authors were involved in the conceptualization and design of the study. H.V. did the analysis and visualization and wrote the original manuscript. The others reviewed and edited the original manuscript. All authors were involved in the revision of the manuscript.

Funding This work was supported by the Research Foundation Flanders, G0D4122N (Wim Van Den Noortgate) European Research Council advanced grant, 101053880 CASCADE (Han L. J. van der Maas) Dutch Research Council VI.Veni.221G.056 (Maria Bolsinova).

Data Availability R scripts with simulation data are available on OSF: https://osf.io/n975b/overview?view_only=8fe1e9c93ba94bb5b9f0cbed3eaaf2dc. Data from Math Garden will not be made available, but can be obtained from Oefenweb (www.oefenweb.nl).

Declarations

Conflicts of interest The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: One of the researchers involved in the project is employed one day a week by Prowise, the company that develops the learning software.

References

Anderson, J.R., Corbett, A.T., Koedinger, K.R., Pelletier, R.: Cognitive tutors: lessons learned. J. Learn. Sci. **4**(2), 167–207 (1995)

Bernacki, M.L., Greene, M.J., Lobczowski, N.G.: A systematic review of research on personalized learning: personalized by whom, to what, how, and for what purpose (s)? Educ. Psychol. Rev. **33**(4), 1675–1715 (2021)

Billah, B., King, M.L., Snyder, R.D., Koehler, A.B.: Exponential smoothing model selection for forecasting. Int. J. Forecast. **22**(2), 239–247 (2006)

Bloom, B.S.: The 2 sigma problem: the search for methods of group instruction as effective as one-to-one tutoring. Educ. Res. **13**(6), 4–16 (1984)

Bolsinova, M., Deonovic, B., Arieli-Attali, M., Settles, B., Hagiwara, M., Maris, G.: Measurement of ability in adaptive learning and assessment systems when learners use on-demand hints. Appl. Psychol. Meas. **46**(3), 219–235 (2022a)

Bolsinova, M., Maris, G., Hofman, A.D., van der Maas, H.L., Brinkhuis, M.J.: Urnings: A new method for tracking dynamically changing parameters in paired comparison systems. J. R. Stat. Soc.: Ser. C: Appl. Stat. **71**(1), 91–118 (2022b)

Brinkhuis, M.J., Maris, G.: Dynamic parameter estimation in student monitoring systems. Measurement and Research Department Reports (Rep. No. 2009-1). Arnhem: Cito, 146 (2009)

Brown, R.G.: Smoothing, Forecasting and Prediction of Discrete Time Series. Courier Corporation, New York (2004)

Csikszentmihalyi, M.: Flow and education. In: Applications of Flow in Human Development and Education: The Collected Works of Mihaly Csikszentmihalyi, pp. 129–151 (2014)

Debeer, D., Vanbecelaere, S., Van Den Noortgate, W., Reynvoet, B., Depaepe, F.: The effect of adaptivity in digital learning technologies modelling learning efficiency using data from an educational game. Br. J. Educ. Technol. **52**(5), 1881–1897 (2021)

Deci, E.L., Ryan, R.M.: Handbook of Self-Determination Research. University Rochester Press, New York (2004)

DeLong, C., Pathak, N., Erickson, K., Perrino, E., Shim, K., Srivastava, J.: Teamskill: modeling team chemistry in online multi-player games. In: Advances in Knowledge Discovery and Data Mining: 15th Pacific-Asia Conference, Pakdd 2011, Shenzhen, China, May 24–27, 2011, Proceedings, part ii 15, pp. 519–531 (2011)

Desmarais, M.C., Baker, R.S.d.: A review of recent advances in learner and skill modeling in intelligent learning environments. User Model. User-Adap. Inter. **22**, 9–38 (2012)

Elo, A.E.: The Rating of Chessplayers, Past and Present. Arco Pub, Bangalore (1978)

Ericsson, K.A., Krampe, R.T., Tesch-Römer, C.: The role of deliberate practice in the acquisition of expert performance. Psychol. Rev. **100**(3), 363 (1993)

Er-Rafyg, A., Zankadi, H., Idrissi, A.: Ai in adaptive learning: challenges and opportunities. In: Modern Artificial Intelligence and Data Science 2024: Tools, Techniques and Systems, pp. 329–342 (2024)

Glickman, M.E.: Parameter estimation in large dynamic paired comparison experiments. J. R. Stat. Soc.: Ser. C: Appl. Stat. **48**(3), 377–394 (1999)

Glickman, M.E.: Dynamic paired comparison models with stochastic variances. J. Appl. Stat. **28**(6), 673–689 (2001)

Glickman, M.E.: Example of the Glicko-2 system. http://glicko.net/glicko/glicko2.pdf (2022)

Glickman, M.E., Doan, T.: The US chess rating system. https://new.uschess.org/sites/default/files/media/documents/the-us-chess-rating-system-revised-september-2020.pdf (2020)

Hodson, D.: Teaching and Learning Science: Towards a Personalized Approach. McGraw-Hill Education, London (1998)

Holt, C.C.: Forecasting trends and seasonals by exponentially weighted moving averages. ONR Memorandum **52**(52), 5–10 (1957)

Jiang, S., Xiao, J., Wang, C.: On-the-fly parameter estimation based on item response theory in item-based adaptive learning systems. Behav. Res. Methods **55**(6), 3260–3280 (2023)

Kabudi, T., Pappas, I., Olsen, D.H.: Ai-enabled adaptive learning systems: a systematic mapping of the literature. Comput. Educ. Artif. Intell **2**, 100017 (2021)

Klinkenberg, S., Straatemeier, M., van der Maas, H.L.: Computer adaptive practice of maths ability using a new item response model for on the fly ability and difficulty estimation. Comput. Educ. **57**(2), 1813–1824 (2011)

Koedinger, K.R., Anderson, J.R., Hadley, W.H., Mark, M.A.: Intelligent tutoring goes to school in the big city. Int. J. Artif. Intell. Educ. **8**, 30–43 (1997)

Kwiatkowski, D., Phillips, P.C., Schmidt, P., Shin, Y.: Testing the null hypothesis of stationarity against the alternative of a unit root: How sure are we that economic time series have a unit root? J. Econom. **54**(1–3), 159–178 (1992)

Maris, G., Van der Maas, H.: Speed-accuracy response models: Scoring rules based on response time and accuracy. Psychometrika **77**(4), 615–633 (2012)

Masthoff, J.: Group modeling: Selecting a sequence of television items to suit a group of viewers. In: Targeting Programs to Individual Viewers, Personalized Digital Television, pp. 93–141 (2004)

Neumann, C., Duboscq, J., Dubuc, C., Ginting, A., Irwan, A.M., Agil, M., Engelhardt, A.: Assessing dominance hierarchies: validation and advantages of progressive evaluation with elo-rating. Anim. Behav. **82**(4), 911–921 (2011)

Nwana, H.S.: Intelligent tutoring systems: an overview. Artif. Intell. Rev. **4**(4), 251–277 (1990)

Ostertagová, E., Ostertag, O.: The simple exponential smoothing model. In: The 4th International Conference on Modelling of Mechanical and Mechatronic Systems, Technical University of Košice, Slovak Republic, Proceedings of Conference, pp. 380–384 (2011)

Pankiewicz, M.P.: A warm-up for adaptive online learning environments–the elo rating approach for assessing the cold start problem. In: International Conference on Computers in Education, pp. 324–329 (2020)

Park, J.: Online estimation of student ability and item difficulty with glicko-2 rating system on stratified data. In: International Educational Data Mining Society (2021)

Park, J.Y., Joo, S.-H., Cornillie, F., van der Maas, H.L., Van den Noortgate, W.: An explanatory item response theory method for alleviating the cold-start problem in adaptive learning environments. Behav. Res. Methods **51**(2), 895–909 (2019)

Pavlik Jr, P.I., Cen, H., Koedinger, K.R.: Performance factors analysis—a new alternative to knowledge tracing. In: Proceedings of the 2009 Conference on Artificial Intelligence in Education. IOS Press (2009)

Pelánek, R.: Bayesian knowledge tracing, logistic models, and beyond: an overview of learner modeling techniques. User Model. User-Adap. Interact. **27**, 313–350 (2017)

Pelánek, R.: Adaptive learning is hard: challenges, nuances, and trade-offs in modeling. Int. J. Artif. Intell. Educ. **35**(1), 304–329 (2025)

Pelánek, R., Papoušek, J., Řihák, J., Stanislav, V., Nižnan, J.: Elo-based learner modeling for the adaptive practice of facts. User Model. User-Adap. Interact. **27**, 89–118 (2017)

Pliakos, K., Joo, S.-H., Park, J.Y., Cornillie, F., Vens, C., Van den Noortgate, W.: Integrating machine learning into item response theory for addressing the cold start problem in adaptive learning systems. Comput. Educ. **137**, 91–103 (2019)

Šarić-Grgić, I., Grubišić, A., Gašpar, A.: Twenty-five years of Bayesian knowledge tracing: a systematic review. User Model. User-Adap. Interact. **34**(4), 1127–1173 (2024)

Schölkopf, B., Platt, J., Hofmann, T.: Trueskill: a Bayesian skill rating system. In: Advances in Neural Information Processing Systems, vol. 19, pp. 569–576. MIT Press (2007)

Sharabi, L.L.: Finding love on a first data: matching algorithms in online dating. Harvard Data Sci. Rev. **4**(1), 1–11 (2022)

Shemshack, A., Spector, J.M.: A systematic literature review of personalized learning terms. Smart Learn. Environ. **7**(1), 1–20 (2020)

Shin, Y., Schmidt, P.: The kpss stationarity test as a unit root test. Econ. Lett. **38**(4), 387–392 (1992)

Tlili, A., Salha, S., Wang, H., Huang, R., Rudolph, J., Weidong, R.: Does personalization really help in improving learning achievement? A meta-analysis. In: 2024 IEEE International Conference on Advanced Learning Technologies (icalt), pp. 13–17 (2024)

Vanbecelaere, S., Cornillie, F., Sasanguie, D., Reynvoet, B., Depaepe, F.: The effectiveness of an adaptive digital educational game for the training of early numerical abilities in terms of cognitive, noncognitive and efficiency outcomes. Br. J. Edu. Technol. **52**(1), 112–124 (2021)

van der Linden, W.J., Hambleton, R.K.: Handbook of Modern Item Response Theory. Springer, Berlin (2013)

Van Schoors, R., Elen, J., Raes, A., Depaepe, F.: An overview of 25 years of research on digital personalised learning in primary and secondary education: a systematic review of conceptual and methodological trends. Br. J. Edu. Technol. **52**(5), 1798–1822 (2021)

Vermeiren, H., Kruis, J., Bolsinova, M., van der Maas, H.L., Hofman, A.D.: Psychometrics of an elo-based large-scale online learning system. Comput. Educ. Artif. Intell. **8**, 100376 (2025)

Véron, M., Marin, O., Monnet, S.: Matchmaking in multi-player on-line games: studying user traces to improve the user experience. In: Proceedings of Network and Operating System Support on Digital Audio and Video Workshop, pp. 7–12 (2014)

Vygotsky, L.S., Cole, M.: Mind in Society: Development of Higher Psychological Processes. Harvard University Press, New York (1978)

Wauters, K., Desmet, P., Van Den Noortgate, W.: Adaptive item-based learning environments based on the item response theory: possibilities and challenges. J. Comput. Assist. Learn. **26**(6), 549–562 (2010)

Wauters, K., Desmet, P., Van Den Noortgate, W.: Monitoring learners' proficiency: weight adaptation in the Elo rating system. Educ. Data Min. 2011 (2010)

Winne, P.H.: Theories of instruction and of intelligence for designing artificially intelligent tutoring systems. Educ. Psychol. **24**(3), 229–259 (1989)

Xie, H., Chu, H.- C., Hwang, G.- J., Wang, C.- C.: Trends and development in technology-enhanced adaptive/personalized learning: a systematic review of journal publications from 2007 to 2017. Comput. Educ. **140**, 103599 (2019)

Yang, L., Dimitrov, S., Mantin, B.: Forecasting sales of new virtual goods with the Elo rating system. J. Revenue Pricing Manag. **13**, 457–469 (2014)

Yudelson, M.: Elo, i love you won t you tell me your k. In: European Conference on Technology Enhanced Learning, pp. 213–223 (2019)

Zheng, L., Long, M., Zhong, L., Gyasi, J.F.: The effectiveness of technology-facilitated personalized learning on learning achievements and learning perceptions: a meta-analysis. Educ. Inf. Technol. **27**(8), 11807–11830 (2022)

Hanke Vermeiren is a PhD student at the Faculty of Psychology and Educational Sciences and imec research group itec at the KU Leuven, researching the Elo rating system algorithm for adaptive learning environments.

Abe Hofman is an Assistant Professor at the Psychological Methods group at the University of Amsterdam. He conducts research on algorithms for adaptive learning and models the data collected by these systems aiming to understanding when learning happens. He co-leads the edaptiv.org research group, and works as a data scientist for Prowise Learn.

Maria Bolsinova is an assistant professor at the Methodology and Statistics department at Tilburg University in the Netherlads. Her research interests are psychometrics for (adaptive) learning and technology-based assessment.

Han van der Maas is a Professor of Psychological Methods at the University of Amsterdam. His research centers on the formalization and empirical evaluation of psychological theories, with applications in cognition, expertise, developmental processes, attitudes, and intelligence.

Wim Van den Noortgate is Professor of Statistics at the Faculty of Psychology and Educational Sciences and imec research group itec at the KU Leuven. His main research interests include meta-analysis and learning analytics.