Measurement and Research Department Reports 2010-1

Adaptive Estimation: How to Hit a Moving Target

Matthieu J.S. Brinkhuis
Gunter Maris

Measurement and Research Department Reports 2010-1

Adaptive Estimation: How to Hit a Moving Target

Matthieu J.S. Brinkhuis

Gunter Maris

Cito
Arnhem, 2010

Abstract

An adaptive estimation procedure is presented for tracking changing parameters, such as abilities of students. The procedure is based on a Metropolis algorithm, a Markov chain Monte Carlo method involving an old state, a new state, and a stochastic innovation such that the Markov chain converges to an invariant distribution. The stochastic innovation is partially provided by responses of persons to items, with the Markov chain as a dynamic estimator of ability. Item responses are incorporated in two different places. First, item responses are incorporated as stochastic innovations of the Metropolis candidate distribution, with a discrete Markov chain as an alternative estimate of ability. Second, item responses are used as acceptance variables. A simulation study is provided to demonstrate some properties of these adaptive estimation procedures in monitoring student abilities. Several applications and extensions are discussed.

1. Introduction

The dominant tradition in statistical inference, especially parameter estimation, assumes that parameters are fixed unknown quantities, the value of which is to be estimated. In the field of educational measurement, however, it typically holds true that parameters change over time. For example, the point of providing education to students is exactly to change the value of their ability parameter. As long as observations are collected in a relatively short period of time, any changes in the abilities of students during testing can, probably, safely be ignored. However, when testing extends over a long period of time, changes in the ability of the students seem inevitable. In this paper we consider how parameter estimation may account for possible changes in the value of a parameter.

A straightforward approach to dealing with parameter change is to develop explicit models for exactly how the parameters change. That is, a (latent) growth model (Rao, 1958; Stoel, van den Wittenboer, & Hox, 2004) or other explicit models for change (Visser, Raijmakers, & van der Maas, 2009), can be considered. Although such models are certainly valuable, their scope is limited to situations in which a growth model of limited complexity is suitable. However, in many contexts it is not well defined how development or growth takes place (e.g., in a student monitoring system where students are frequently administered a test over period of a couple of years). In this paper we consider *tracking* systems that aim to follow development through time, rather than model it.

1.1. A simple example.

The type of system we will consider can be illustrated with a simple coin tossing example. To put the example in an educational measurement context we consider the situation where the same student repeatedly and independently answers the to same item. The response of the student with ability θ on the i-th occasion is denoted with the Bernoulli random variable X_i:

$$P(X_i = 1) = \frac{\exp(\theta)}{1 + \exp(\theta)} = p \ . \tag{1}$$

Formally, $\mathbf{X}$ is an *independent and identically distributed* (iid) sequence of Bernoulli random variables. The obvious estimate of the parameter p is the sample mean:

$$\overline{X}_n = \frac{1}{n}\sum_{i=1}^{n} X_i \xrightarrow{P} p \ , \tag{2}$$

which has many desirable statistical properties. It is unbiased, consistent, and best asymptotically normal, to name a few of these.

If we consider a situation where at some trial j the coin is replaced by a different one, or framed in the context of our educational measurement interpretation of the coin tossing experiment, the ability of the student changes:

$$P(X_i = 1) = \begin{cases} \frac{\exp(\theta)}{1+\exp(\theta)} = p^{<j} & \text{if } i < j \\ \frac{\exp(\theta-\delta)}{1+\exp(\theta-\delta)} = p^{\geq j} & \text{if } i \geq j \end{cases} \tag{3}$$

As a consequence, our naive estimator looses many of its desirable statistical properties. If j is large enough, it will take a very long time before the sample mean becomes even close to $p^{\geq j}$. To appreciate why this is the case we express the sample mean, based on a sample of size n, as a function of the sample mean, based on a sample of size $n-1$, and a new observation:

$$\overline{X}_n = \frac{n-1}{n}\overline{X}_{n-1} + \frac{1}{n}X_n \tag{4}$$

It is readily seen that the weight of new observations decreases with n, which is a good thing if the random variables involved are iid, but prevents the estimator from adapting to changes in the *true* value of the parameter.

An obvious way to overcome this problem is to give a fixed weight to both terms:

$$\widehat{X}_n = (1-\alpha)\widehat{X}_{n-1} + \alpha X_n \tag{5}$$

Such an estimator has very different operating characteristics. It is clearly *not* consistent, nor is it necessarily unbiased. If the value of the parameter does change, however, the estimator will quickly adapt to the new value, where the speed of adaptation is governed by the value of α. Figure 1 offers an illustration of how much faster $\widehat{\mathbf{X}}_n$ adapts to a parameter change than does $\overline{\mathbf{X}}_n$.

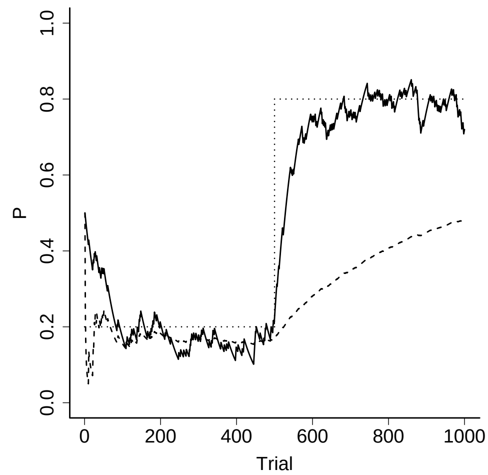

FIGURE 1. True parameter value (dotted line), sample mean (dashed line) and our alternative estimator (solid line) where $\alpha = .02$.

Both $\widehat{\mathbf{X}}_n$ and $\overline{\mathbf{X}}_n$ form Markov chains. That is, their present value is independent of the past values conditionally on the previous value. However, $\overline{\mathbf{X}}_n$ converges to a point whereas $\widehat{\mathbf{X}}_n$ converges to a non degenerate invariant distribution. Some simple analysis shows that *if* the value of the parameter does not change, *then* all moments of this invariant distribution can be determined. Put differently, as long as the value of the parameter does not change, $\widehat{\mathbf{X}}_n$ converges to a known invariant distribution. Hence, if the Markov chain has converged to its stationary distribution and the value of the parameter has not changed, we may use the Markov chain to construct an unbiased and consistent estimator of the parameter. If on the other hand, the value of the parameter does change, the Markov chain will adapt to the new value and will provide us with information on the nature of the change in the parameter value. Later on we illustrate how these characteristics can be used in an educational measurement context.

1.2. **The Elo rating system.** Problems concerning changing abilities have appeared in other fields as well. A lot of work has been done in the field of measuring chess expertise. Already in 1929, Zermelo[1] introduced a model we now know as the BTL model (Bradley & Terry, 1952; Luce, 1959) to measure chess ability. The influential work of Elo (1978) describes several methods to deal with large number of chess games. His method was adopted by the World Chess Federation (FIDE) and several national chess federations, such as the United States Chess Federation (USCF). Chess games are regarded as paired comparisons, where two objects (players) are compared (play a match) and one is preferred over the other (wins or loses)[2]. Here we note the relation to an educational measurement context, where students answer items. Every answered item can be regarded as a student-item pair, where we might be interested in estimating either the ability of the students, the difficulty of the items, or both at the same time. With the use of such pairs, we can describe traditional examinations with a fixed item set, adaptive tests and tests that are administered over time. The relation to educational measurement allows us to use results from research that concerns the Elo rating system.

The Elo rating system is a numerical system in which each player receives a rating which may be converted into a winning probability. It consists of several different forms with corresponding formulas. Below the *Current Rating Formula for Continuous Measurement* is presented (1978, p. 25).

$$R_n = R_o + K(W - W_e) \tag{6}$$

R_n is the new rating after the event.
R_o is the pre-event rating.
K is the rating point value of a single game score.
W is the actual game score, each win counting 1, each draw 1/2.
W_e is the expected game score based on R_o.

The player's rating R can be updated after every match, allowing for continuous measurement. In Formula (6), the possible values of W are $\{0, 0.5, 1\}$, and the positive sign of W_e indicates that in every case a player gains points if he wins, and looses points if he loses. The amount of rating points that are at stake depends on the difference between the actual game score W and the expected game score W_e, multiplied by K. Consequently, a player competing a much higher rated opponent risks dropping few points when losing the game, with the possibility of gaining many points when

[1]We thank Gerhard Fischer for calling attention to the work of Ernst Zermelo.
[2]Draws are ignored for the sake of simplicity

winning. The opposite holds for the higher rated player competing with this lower rated opponent.

Properties of Elo's sytem and alternatives are extensively discussed by Batchelder and his co-workers in a series of papers (Batchelder & Bershad, 1979; Batchelder & Simpson, 1989; Batchelder, Bershad, & Simpson, 1992). Many extensions to Elo's system are available, for example by including parameters that vary with time (Fahrmeir & Tutz, 1994; Glickman, 1999), allow for sudden shifts in ability (Glickman, 2001), or include team ability estimations in a large scale gaming application (Herbrich, Minka, & Graepel, 2006).

Formula (6) bases its updated state on an observed outcome and a previous state, satisfying the Markov property. Specifically, it satisfies a discrete trial, inhomogeneous Markov process with on any trial two possible transitions, concerning a win or a loss (Batchelder et al., 1992, p. 194–195). A player's past is only represented in the previous ability estimate, which allows for quick adaptation to changing ability levels. Due to this property, estimates in the rating system do not converge to points, but remain noisy.

There are several desirable properties of Elo's system to follow development through time in an educational measurement context. It it a simple, self correcting system which simultaneously tracks the ability of two opposing chess players, or of students and items at the same time. In the field of chess ability estimation, it is widely accepted, applied and studied.

However, if we regard the rating system as a measurement model we find that properties we desire from an estimator, such as unbiasedness and consistency are not granted. Bias is discussed in Brinkhuis and Maris (2009), though the extent is limited in size and symmetric.

Batchelder and Bershad (1979) suggest an unbiased alternative to Elo's formula for sequential estimation, a simple linear difference system:

$$\widehat{X}_{t+1} = (1-a)\widehat{X}_t + aX_{t+1} \tag{7}$$

$$= \widehat{X}_t + a(X_{t+1} - \widehat{X}_t) \tag{8}$$

The linear difference system needs an unbiased estimator of ability to produce unbiased results. Formula (7) displays a linear difference equation with constant coefficients and random input X_{t+1} (1979, p. 53). It is rewritten in Formula (8) in the form of Elo's rating system in Formula (6). Rating $\widehat{X}_{t+1}$ is a new rating, being assigned to each player after a tournament. $\widehat{X}_t$ is an estimate of a players true rating, which might change over time due to update X_{t+1}. The parameter a is a weighing parameter, bounded between zero and unity.

The purpose of this paper is to apply well known methods from the field of *Markov Chain Monte Carlo* (MCMC) for the purpose of adaptive estimation. We will first discuss a Metropolis type of system as a flexible solution to adaptive estimation in the next paragraph. In the Metropolis system we have two logical places in which we can insert real item responses. We will discuss in two seperate paragraphs item responses as stochastic innovations, and item responses as acceptance variables in the Metropolis system. Both lead to different systems which we discuss and apply to simulations in the succeeding paragraphs. Finally, we will discuss the results of this paper, and suggest extensions and possible applications in several fields.

2. A Metropolis type of system

Basically, MCMC methods work in the following general way. A new state is determined from the old state and some stochastic innovation, in such a way that the resulting Markov chain converges to a, typically very complicated, invariant distribution. In adaptive estimation, an obvious approach is to let the stochastic innovation be determined, partially, by the responses of students to items. If the random variable X_i denotes the current state of our dynamic estimator, Y_i denotes the response to the i-th administered item, and we assume, for the moment, that the ability parameter θ does *not* change over time, we may schematically represent the estimation procedure as follows:

$$\begin{array}{ccccccccccc} & & \theta & & \theta & & \theta & & \dots & & \theta \\ & & \downarrow & & \downarrow & & \downarrow & & & & \downarrow \\ & & Y_1 & & Y_2 & & Y_3 & & \dots & & Y_\infty \\ & & \downarrow & & \downarrow & & \downarrow & & & & \downarrow \\ X_0 & \rightarrow & X_1 & \rightarrow & X_2 & \rightarrow & X_3 & \rightarrow & \dots & \rightarrow & X_\infty \end{array} \tag{9}$$

Our ultimate interest is in the distribution of X_∞. Specifically, we require that X_∞ has a known distribution for which θ is the location parameter:

$$P(X_\infty \leq x) = F(x - \theta) \equiv F_\theta(x) \tag{10}$$

where F is a distribution function that does not depend on any unknown parameters.

The Metropolis algorithm (Metropolis et al., 1953; Chib & Greenberg, 1995) from MCMC provides a flexible and powerful way to construct such a Markov chain with a given invariant distribution. Formally, the Metropolis-Hastings (Hastings, 1970) algorithm may be characterized as follows:

$$X_{t+1} \sim \begin{cases} X_t & \text{if } z_t = 0 \\ Y_t & \text{if } z_t = 1 \end{cases} \sim X_t \sim F_\theta(\cdot) \tag{11}$$

where X_t is our estimator of ability, a random variable from distribution f, and Y_t is a random variable from the proposal distribution g. The value of z_t determines whether the proposal value y_t is accepted or rejected. The algorithm is called the Metropolis algorithm if the distribution of Y_t conditional on X_{t-1} is symmetric about the value x_t, and the Hastings algorithm otherwise.

2.1. **A Formal Characterization.**

As a courtesy to the reader we work out the consequences of this formal characterization in some detail, for the case where X_t, and also Y_t, is a real-valued random variable[3]. The schematic

[3] If we regard discrete distributions, integrals in formulas are simply replaced by summations.

representation of the Metropolis-Hastings algorithm given above translates into the following formal characterization:

$$\begin{aligned} F(x-\theta) &= \int_{-\infty}^{x}\int_{-\infty}^{\infty} \pi(s,t) f(s-\theta) g(t|s) dt ds \\ &+ \int_{-\infty}^{\infty}\int_{-\infty}^{x} [1-\pi(s,t)] f(s-\theta) g(t|s) dt ds \end{aligned}$$

where $\pi(x_t, y_t) = P(z_t = 1|x_t, y_t)$ is the *acceptance probability* (i.e., the probability that we accept y_t as a draw from F_θ) and g denotes the proposal density, the distribution of Y_t conditionally on the value of x_t.

With a change in notation, and a change in the order of integration for the second integral we can express this equation as follows:

$$\begin{aligned} F(x-\theta) &= \int_{-\infty}^{x}\int_{-\infty}^{\infty} \pi(s,t) f(s-\theta) g(t|s) dt ds \\ &+ \int_{-\infty}^{x}\int_{-\infty}^{\infty} [1-\pi(t,s)] f(t-\theta) g(s|t) dt ds \end{aligned}$$

which can further be simplified to yield:

$$\begin{aligned} F(x-\theta) &= \int_{-\infty}^{x}\int_{-\infty}^{\infty} \pi(s,t) f(s-\theta) g(t|s) dt ds \\ &+ F(x-\theta) - \int_{-\infty}^{x}\int_{-\infty}^{\infty} \pi(t,s) f(t-\theta) g(s|t) dt ds \end{aligned}$$

from which we obtain the following condition:

$$\int_{-\infty}^{x}\int_{-\infty}^{\infty} \pi(s,t) f(s-\theta) g(t|s) dt ds = \int_{-\infty}^{x}\int_{-\infty}^{\infty} \pi(t,s) f(t-\theta) g(s|t) dt ds \tag{12}$$

Clearly, if the integrands on the left and right hand side are equal, the condition will hold, which gives us the following *detailed balance* condition (Chib & Greenberg, 1995):

$$\pi(s,t) f(s-\theta) g(t|s) = \pi(t,s) f(t-\theta) g(s|t) \tag{13}$$

It is important to observe that for given f and g the detailed balance condition does *not* admit of a single unique solution for π. For instance, both the functions:

$$\pi(s,t) = \min\left[1, \frac{f(t-\theta)g(s|t)}{f(s-\theta)g(t|s)}\right] \tag{14}$$

and

$$\pi(s,t) = \frac{f(t-\theta)g(s|t)}{f(t-\theta)g(s|t) + f(s-\theta)g(t|s)} \tag{15}$$

meet the detailed balance condition. Observe that if we consider a Metropolis algorithm, neither of these expressions for the acceptance probability π depend on the functional form of the proposal density g.

2.2. **Including item responses.** A Metropolis algorithm involves three distributions: the invariant distribution f, the distribution g of proposal values conditionally on the current state of the Markov chain, and the acceptance probability π conditionally on both the proposal value and the current state of the Markov chain. For the purpose of adaptive estimation, it is important to observe that only the distribution of item responses is given, and all three distributions needed for the construction of a Metropolis algorithm may be chosen freely, as long as they meet the symmetry condition on g and π satisfies the detailed balance condition. Since the invariant distribution f is assumed to be independent of the items that are administered, we are left with two choices for incorporating the effect of item responses. First, we may choose to use item responses as stochastic innovations, where they determine the proposal distributions g which yields discrete distributions for both f and g. Second, we may use item responses to determine acceptance or rejection directly, as acceptance variables π. Both options are developed below.

3. Item responses as stochastic innovations

The first use of the Metropolis algorithm for adaptive estimation relates back directly to the coin tossing example from the introduction. The only difference is that in test administration, at each trial *a different coin* is being tossed. Specifically, the i-th item that is being administered has difficulty δ_i, and hence the probability of a correct response equals:

$$P(Y_i = 1) = \frac{\exp(\theta - \delta_i)}{1 + \exp(\theta - \delta_i)} \tag{16}$$

Put differently, the coin tossing example from the introduction involves *independent and identically distributed* (iid) replications of a Bernoulli random variable, whereas test administration involves *independent and non-identically distributed* (inid) Bernoulli random variables. We show how we may use these inid Bernoulli variables to construct a sequence of *dependent identically distributed* (did) Bernoulli random variables. That is, we turn independent answers to different questions into dependent answers to the same question using a Metropolis algorithm.

The state X_t of the Markov chain we construct is assumed to be governed by an invariant Bernoulli distribution with parameter:

$$p = \frac{\exp(\theta)}{1 + \exp(\theta)} \tag{17}$$

The t-th item response of a person is the proposal value Y_t, and is governed by the following Bernoulli distribution:

$$P(Y_t = 1) = \frac{\exp(\theta - \delta_{I(t)})}{1 + \exp(\theta - \delta_{I(t)})} \tag{18}$$

where $I(t)$ gives the index of the item that is administered at time t. Clearly, if Y_t equals X_t, the new state of the Markov chain will automatically be equal to its previous value, so that the choice of π in that case is of no consequence. If however, Y_t does not equal X_t the choice of π becomes important. If we consider the different ways the state of our Markov chain at time $t+1$ can be equal to one, and require that its distribution is Bernoulli with parameter p we obtain that:

$$
\begin{aligned}
(19) \qquad P(X_{t+1}=1) &= \frac{\exp(\theta)}{1+\exp(\theta)} \\
&= [1-\pi(1,0)]\frac{\exp(\theta)}{1+\exp(\theta)}\frac{1}{1+\exp(\theta-\delta_{I(t)})} \\
&+ \pi(0,1)\frac{1}{1+\exp(\theta)}\frac{\exp(\theta-\delta_{I(t)})}{1+\exp(\theta-\delta_{I(t)})} \\
&+ \pi(1,1)\frac{\exp(\theta)}{1+\exp(\theta)}\frac{\exp(\theta-\delta_{I(t)})}{1+\exp(\theta-\delta_{I(t)})}
\end{aligned}
$$

which, if $\pi(1,1)$ is chosen to equal unity, gives the following sufficient detailed balance condition:

(20)

$$\pi(0,1)\frac{1}{1+\exp(\theta)}\frac{\exp(\theta-\delta_{I(t)})}{1+\exp(\theta-\delta_{I(t)})} = \pi(1,0)\frac{\exp(\theta)}{1+\exp(\theta)}\frac{1}{1+\exp(\theta-\delta_{I(t)})}$$

which can be simplified to yield:

$$(21) \qquad \pi(0,1)\exp(-\delta_{I(t)}) = \pi(1,0)$$

We make the following choice, consistent with the detailed balance condition:

$$(22) \qquad \pi(1,0) = \min\left(1, \frac{\frac{\exp(\theta)}{1+\exp(\theta)}\frac{1}{1+\exp(\theta-\delta_{I(t)})}}{\frac{1}{1+\exp(\theta)}\frac{\exp(\theta-\delta_{I(t)})}{1+\exp(\theta-\delta_{I(t)})}}\right) = \min\left[1, \exp(\delta_{I(t)})\right]$$

and

$$(23) \qquad \pi(0,1) = \min\left(1, \frac{\frac{1}{1+\exp(\theta)}\frac{\exp(\theta-\delta_{I(t)})}{1+\exp(\theta-\delta_{I(t)})}}{\frac{\exp(\theta)}{1+\exp(\theta)}\frac{1}{1+\exp(\theta-\delta_{I(t)})}}\right) = \min\left[1, \exp(-\delta_{I(t)})\right]$$

We see that the acceptance probabilities do *not* depend on the unknown value of θ, which means that as long as the item difficulties for all items are known, we may actually generate realizations of the acceptance random variable Z_t.

In the coin tossing example from the introduction we used a linear difference system on the *iid* sequence as an alternative estimator for p. We can use the same linear difference system on the *did* sequence developed here. While the first moment of the estimator will be correct, results for other moments are not. Working out these moments for the *did* sequence developed here is left open for further research.

An implementation of the algorithm is expressed below in pseudo-code. For each person, we estimate ability by feeding dichotomous sequence x to a linear difference algorithm. We register item response y at time t to some selected item with difficulty parameter δ.

```
if y_t = x_{t-1} then x_t = x_{t-1}
else
   if x_{t-1} = 1 and y_t = 0 then
      z ~ Bernoulli(min[1, exp(δ_{I(t)})])
   if x_{t-1} = 0 and y_t = 1 then
      z ~ Bernoulli(min[1, exp(-δ_{I(t)})])
   if z = 0 then x_t = y_t
   if z = 1 then x_t = x_{t-1}
θ_t = (1 - α)θ_{t-1} + αx_t
```

First, if the dichotomous item response is the same as the value of x_{t-1}, we do not change our estimate x. However, if the value of y_t differs from x_{t-1}, we draw a value z from the appropriate Bernoulli acceptance distribution. Depending on the value of z, we either accept or reject our proposal value y_t, the item response. Finally, x_t updates our ability estimate θ_t by means of a linear difference equation with weight α.

4. Item responses as acceptance variables

The obvious alternative way to use item responses in a Metropolis-Hastings algorithm is to use the item responses a person gives as acceptance variables. Here we develop such an approach. We now denote the response of a person to item $\delta_{I(x,y)}$ as Z, the estimates as X and the proposal distribution as Y. Specifically, we choose

$$P(Z_t = 1 | X_t = x, Y_t = y) = \pi(x,y) \quad = \begin{cases} \frac{\exp(\theta - \delta_{I(x,y)})}{1+\exp(\theta-\delta_{I(x,y)})} & \text{if } y > x \\ \frac{1}{1+\exp(\theta-\delta_{I(x,y)})} & \text{if } y \le x \end{cases}$$

That is, if the proposal value is larger than the current value, a correct response implies that the new higher value will be accepted, whereas if the proposal value is smaller than the current value, an incorrect response implies the the new lower value will be accepted.

Moreover, we assume that the acceptance probability has the following relation to the invariant distribution:

$$\pi(x,y) = \frac{f(y-\theta)}{f(x-\theta) + f(y-\theta)} \tag{24}$$

which means that the invariant distribution should satisfy the following functional equation:

$$\frac{f(y-\theta)}{f(x-\theta)} = \frac{\pi(x,y)}{1-\pi(x,y)} = \begin{cases} \exp(\theta - \delta_{I(x,y)}) & \text{if } y > x \\ \exp(-\theta + \delta_{I(x,y)}) & \text{if } y \le x \end{cases} \tag{25}$$

If for f the functional form of a normal distribution with variance σ^2 is assumed we readily obtain:

$$\frac{f(y-\theta)}{f(x-\theta)} = \exp\left[\frac{y-x}{\sigma^2}\left(\theta - \frac{y+x}{2}\right)\right] \tag{26}$$

in which we can recognize an item response model with two parameters, which means that items should be offered to the candidates with a discrimination parameter of $a = (y - x_t)/\sigma^2$ and $\delta = (y + x_t)/2$. An item bank with items with a large variety of discriminations and difficulty indices is required for this purpose. We can restrict $y - x$ to be equal to plus or minus σ^2 in order to restrict the acceptance probability to a Rasch model (Rasch, 1960). Hence, if we choose our proposal values from a symmetric distribution with half of its probability mass at the value $x + \sigma^2$ and half at the value $x - \sigma^2$ by selecting items with difficulty x_t plus or minus $\sigma^2/2$, then our variance parameter σ^2 equals the step size of the algorithm. Clearly, we end up with a discrete Markov chain.

We find, in contrast to the Markov chain developed in the previous section, that this algorithm implies a definite choice for the item that is to be administered at any given moment. Furthermore, the algorithm implies that the states that can be reached lie on a grid, where neighbouring values are σ^2 seperated from each other.

Again, we provide an example of the algorithm in pseudo-code.

$y \sim \mathcal{N}(x_t, \sigma)$
$a = (y - x_t)/\sigma^2$
$\delta = (y + x_t)/2$
`candidate generates` z_t
`if` $z_t = 1$ `then` $x_t = y$
`if` $z_t = 0$ `then` $x_t = x_{t-1}$

Based on the ability estimate x_t and variance σ^2, an appropriate proposal value is drawn from a normal distribution. An alternative distribution, as discussed herefore, can be used to obtain a Rasch model. Item parameters for item selecton are calculated, and a response z_t to this specific item is generated by the candidate. Specific for this algorithm is that the response to a specific item determines acceptance or decline of the candidate value y.

5. Simulation study

Here we describe the results of a simulation study to illustrate the adaptive estimation algorithms introduced in this paper and their applications. We discussed two places where item responses can be used in the Metropolis algorithm. First, we develop an example where we use item responses as stochastic innovations. Second, we develop an example where item responses are acceptance variables.

5.1. Item responses as stochastic innovations.

We have simulated 1000 students, who answer 500 questions in a test that stretches over some time. In this first study, where item responses are stochastic innovations, we have freedom in selecting what item to administer. We select random Rasch items with difficulties from a standard normal distribution. We illustrate how the algorithm performs if ability is kept stable, and if ability grows suddenly. Results are graphed for individual trace lines and for the group as a whole.

We start with the example from the introduction, where we observe a coin that undergoes a sudden change in its probability to fall tails. We simulated a thousand exchangable persons that undergo a similar sudden change. They are all offered the same first item, for which the probability of giving a correct response for everyone is $p = .2$, i.e. they all have the same ability of $\theta = \ln(p/(1 - p))$. Our Metropolis algorithm accepts or rejects responses on following items such that we obtain a *did* chain *as if* the first item was answered many times. After 250 trials, the ability of the candidates suddenly changes so that their probability to answer the first item correct changes to $p = .8$. We use a linear difference system to estimate the probabilities back form the dichotomous chains X that each of our simulated respondents produces.

In Figure 2, we plot the linear difference estimates of a single person, together with the underlying probability of answering the first item correct. The linear difference system needs a parameter α between zero and unity to weigh new observations. If we choose a very small number, new updates receive a small weight and changes are therefore quite slow. In Figure 2 the weight is $\alpha = .03$. If we choose the linear difference parameter to be larger, $\alpha = .1$, we observe more noise in the estimates but the ability to adapts quickly to the new value of $p = .8$, as seen in Figure 3.

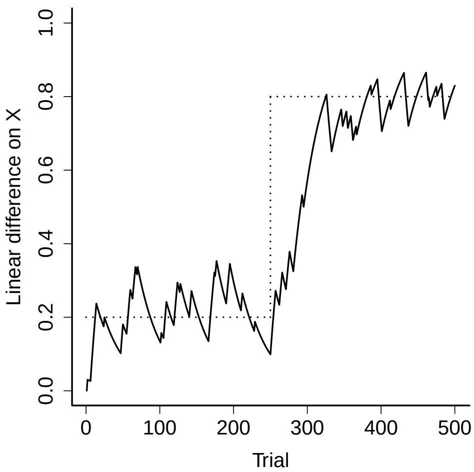

FIGURE 2. Linear difference estimates on a selected person, with $\alpha = .03$.

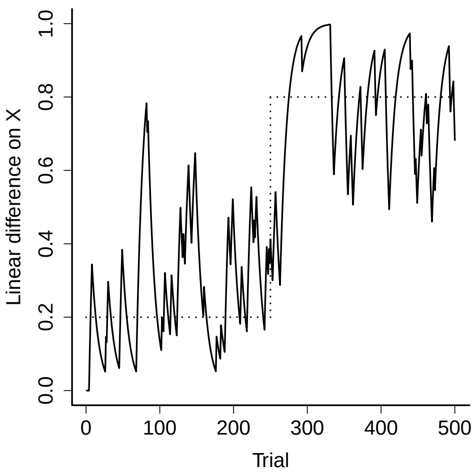

FIGURE 3. Linear difference estimates on a selected person, with $\alpha = .1$.

While we have graphed the estimates of a single person, we can also look at the whole group. At every trial, we have calculated the mean of the linear difference estimates of the persons. Again, we have used the two different levels of α, to evaluate the speed with which the algorithm adapts. It is clear that the algorithm in Figure 4 adapts slower than in Figure 5.

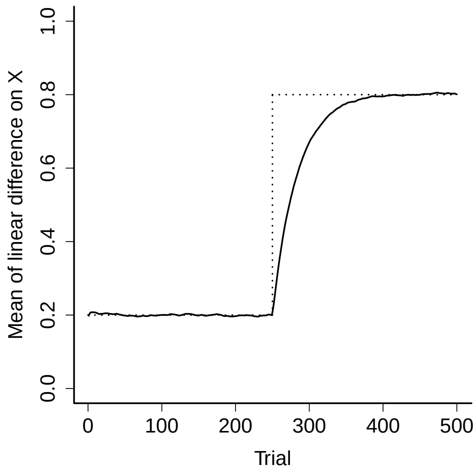

FIGURE 4. Mean linear difference estimates, with $\alpha = .03$.

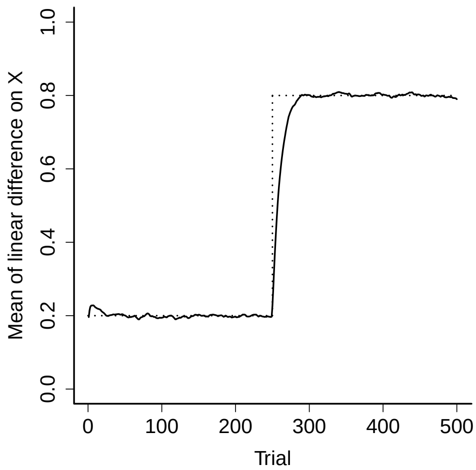

FIGURE 5. Mean linear difference estimates, with $\alpha = .1$.

We remark that regardless of the chosen parameter in the linear difference system, the mean of the estimates converges to its true value, as seen in Figure 4 and Figure 5. For considering estimates of individuals however, stepsize matters in interpreting results as can be seen in Figure 2 and Figure 3.

It is instructive to compare the performance of our Metropolis algorithm with an *iid* sample, following the example in the introduction. We simulate our persons again, only this time their ability does not change and is fixed at $p = .5$. We have also simulated a thousand fair coins that are thrown at every trial. Since abilities, or p-values, are stable in this simulation, we can use a running average to compare the two sequences. We note that since we estimate the probability correct of our first item, our Markov chain is already converged at the start and no burn-in period needs to be considered. Since this is true, the running averages of both sequences will be exactly correct at $p = .5$ if we let our numbers of persons increase ad infinitum.

The interesting difference between the two chains is their difference in standard deviations of the running means, over all 1000 persons. These are plotted in Figure 6.

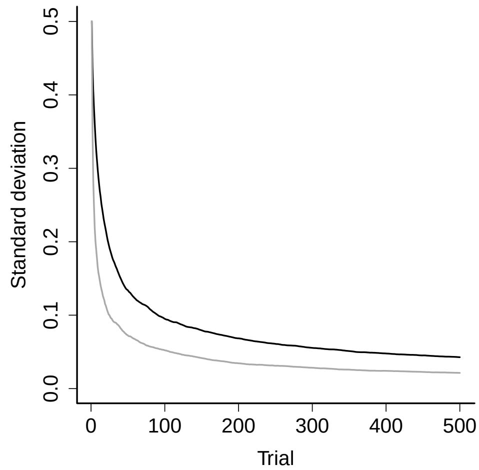

Figure 6. Standard deviation of the means of a Metropolis *did* sequence (black) and an *iid* sequence (grey) .

Intuitively, one would expect that the *iid* chain performs much better because observations are independent. However, Figure 6 shows that our generated *did* sequence does not perform much worse. By dividing these two sequences of standard deviations we obtain their relative efficiency, which is shown in Figure 7. The relative efficiency converges to about 2, which means that we need about 4 times as many observations to obtain a similar variance as the *iid* sample. The relative effiency of the *did* chain depends on many variables, and serves here merely as an indication of efficiency.

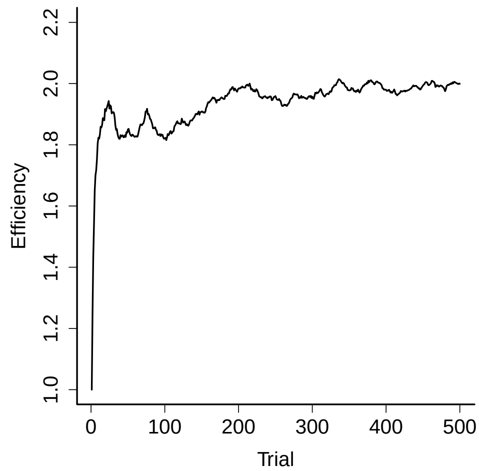

FIGURE 7. Relative efficience of our Metropolis *did* sequence compared to an *iid* sequence.

5.2. **Item responses as acceptance variables.** In this second study, where items act as acceptance variables, we perform a simular simulation as before, with 1000 students answering 500 items over some time. However, we now assume we have a calibrated item bank with known difficulty and discrimination parameters of all sorts to facilitate the item selection as proposed by the algorithm. Again several graphs are provided to demonstrate how the algorithm acts under stable and changing abilties.

First, we illustrate a single student with the adaptive estimates in Figure 8. We observe a stable ability throughout the trials, with estimates that are clearly noisy. The step size of the algorithm is determined by σ^2, which is the variance of the candidate generating density g. If we reduce σ^2, the stepsize and thus the amount of noise are reduced. We have shown that if

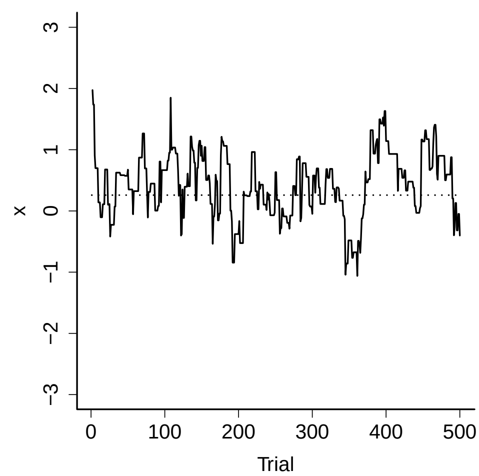

FIGURE 8. A stable ability with true parameter value (solid line) and adaptive estimate (dashed line).

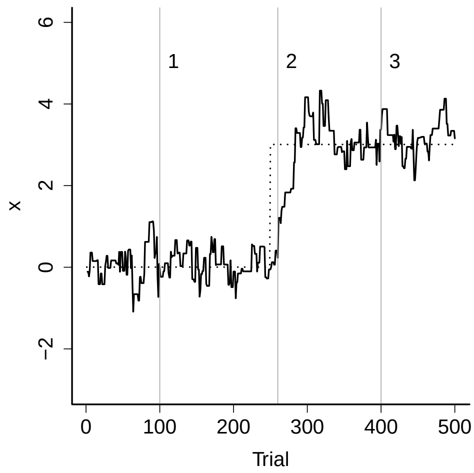

FIGURE 9. A changing parameter value (dotted line) and adaptive estimate (solild line).

ability is stable, the estimates converge to a known invariant distribution centered around the student's true ability.

More interesting is the case where there is growth of ability. The size of the variance is more important here, since it determines whether we have a quick or slow adaptation to changes in true ability. True abilities of the 1000 students are drawn from a standard normal distribution. In our simulation we allowed halfway for a rather large and sudden increase in true abilities for every student, also the spread in true abilities was increased. Figure 9 shows the estimates for a selected student. Though there is some obvious lag between the change in true ability and the adaptive estimates, the system adapts itself quite quickly to the new situation. We have selected three occassions in which we make a cross-section of all the students' abilities, noted by three vertical lines in Figure 9. For each occassion, we can plot cumulative empirical distributions of both the true and estimated abilities of the 1000 students. The estimates of the algorithm contain normal distributed noise with a fixed and determined variance σ^2 for all persons. These estimates can be compared with the true abilities either by a deconvolution of the estimates and the normal distributed error, or by adding a normal distributed random error to the true abilities. The latter option is applied to obtain true cumulative distributions with error and compare them to the estimates with the same distributed error in Figure 10. On the first and last occassion the cumulative distributions of the adaptive estimates resemble the true distribution with error added, while the occassion directly after the sudden shift in ability shows lag in the adaptive estimates as expected.

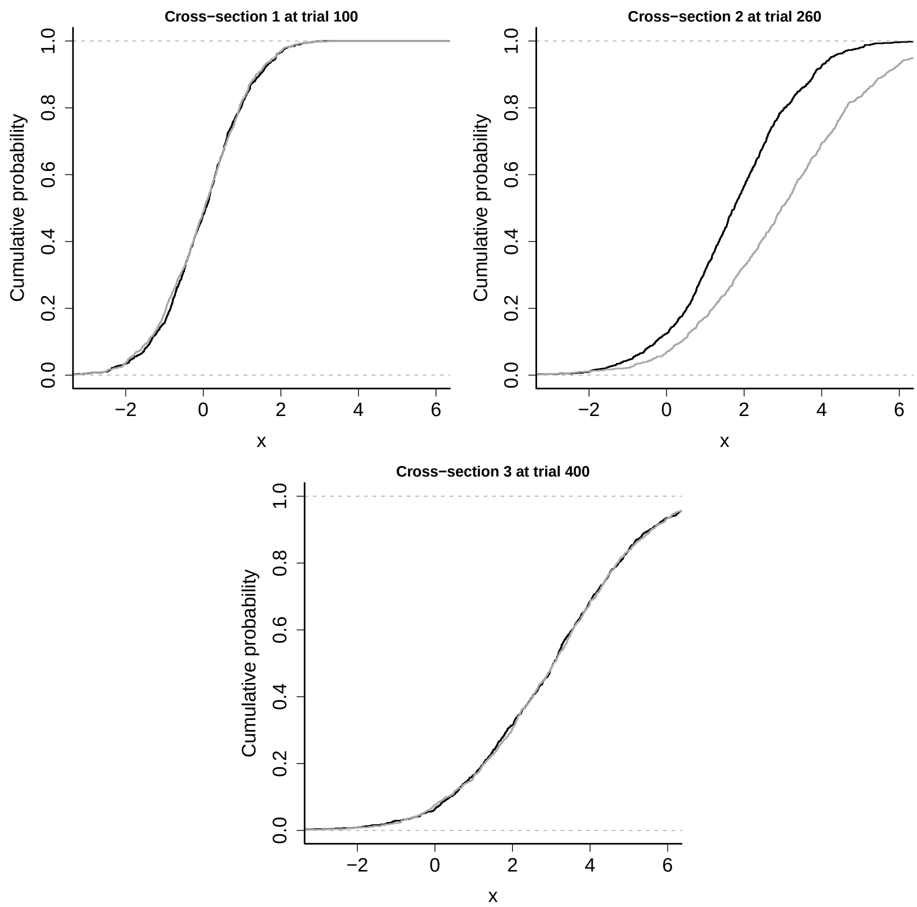

FIGURE 10. Cumulative empirical distributions of true abilities with error (black) and adaptive estimates (grey) at cross-section 1,2 and 3.

6. DISCUSSION

For the purpose of estimating a parameter that may change during the course of the data collection, we developed adaptive estimation procedures based on the Metropolis-Hastings algorithm. It should be clear from the specific algorithms considered here that these are surely not the only ones that could be developed. Neither the choice for the Rasch model nor the choice for the Metropolis-Hastings algorithm are necessary.

Applications of adaptive estimation are obvious in the field of monitoring student progress and well-suited for a whole range of different situations. The most obvious one, and the one that served as guiding example throughout the paper, is that of estimating ability in situations where the ability may change with time. A different type of application can be found on the item side, where it may be of interest to find out whether item parameters are invariant over time. A third variant that could be developed involves the joint estimation of person *and* item parameters, both of which might change with time.

Another type of application becomes clear if we look at the characteristics of the Markov chains that have been constructed. The state of the Markov chain, assuming it has converged to its invariant distribution (10), differs between persons only in the value of its location parameter. We can obtain that accross persons $x_\infty \sim \int_{-\infty}^{\infty} f(x - \theta_p)h(\theta_p)d\theta_p$, where f is fully known. We can use this to recover the unknown distribution of ability h from $\overline{x}$. This property is useful in applications of survey research. If the chains are converged, estimates from a cross-section at some time point can be used in a simple regression analysis without the introducion of bias, since the variance of estimators do not depend on ability. Here we have made the claim of classical test theory true: an observed score equals a true score plus an error.

This paper dealt with adaptive estimation. Specifically, based on the assumption of a fitting IRT model we developed methods to estimate its parameters. An interesting open problem however concerns the evaluation of this assumption. That is, how does one evaluate model fit in a situation where all the parametes involved in the model may be changing with time. We propose to leave this topic for further research. A second problem, that is particulary relevant in situations where both person and item parameters are being estimated is that of parameter identification. It is known from experience with the Elo rating system, the values of the ratings tend to inflate with time. This inflation is due to the fact that chess players enter the system with a low rating and usually leave the system again at some point in time with a much higher rating. From the Elo update it is clear that the rating system maintains the sum of the ratings.

We began by noting that often an explicit growth model is not available. In situations where a tentative growth model is available we can extend the adaptive estimation procedures to both provide estimates for the parameters of the growth model and to track individual students.

A final note concerns the title of this report. In the adaptive estimation procedures as discussed, we are able to track a moving target. However, since there is always lag between the actual change of the targets and our adaptive estimates, we can only hit the target if it stops moving for some time.

References

Batchelder, W. H., & Bershad, N. J. (1979). The statistical analysis of a Thurstonian model for rating chess players. *Journal of Mathematical Psychology*, *19*(1), 39-60.

Batchelder, W. H., Bershad, N. J., & Simpson, R. S. (1992). Dynamic paired-comparison scaling. *Journal of Mathematical Psychology*, *36*, 185-212.

Batchelder, W. H., & Simpson, R. S. (1989). *Rating systems for human abilities: The case of rating chess skill* (Modules in Undergraduate Mathematics and its Applications No. 698). Arlington, MA: COMAP. Available from `http://www.comap.com/product/?idx=134`

Bradley, R. A., & Terry, M. E. (1952). Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, *39*(3/4), 324–345.

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

Chib, S., & Greenberg, E. (1995). Understanding the Metropolis-Hastings algorithm. *The American Statistician*, *49*(4), 327–335.

Elo, A. E. (1978). *The rating of chess players, past and present.* London: B.T. Batsford, Ltd.

Fahrmeir, L., & Tutz, G. (1994). Dynamic stochastic models for time-dependent ordered paired comparison systems. *Journal of the American Statistical Association*, *89*(428), 1438–1449.

Glickman, M. E. (1999). Parameter estimation in large dynamic paired comparison experiments. *Applied Statistics*, *48*, 377–394.

Glickman, M. E. (2001). Dynamic paired comparison models with stochastic variances. *Journal of Applied Statistics*, *28*(6), 673–689.

Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, *57*(1), 97–109.

Herbrich, R., Minka, T., & Graepel, T. (2006). TrueSkill: A Bayesian skill rating system. In B. Schölkopf, J. Platt, & T. Hoffman (Eds.), *Advances in neural information processing systems* (Vol. 19, pp. 569–576). Cambridge, MA: MIT Press. Available from `http://books.nips.cc/nips19.html`

Luce, R. D. (1959). *Individual choice behavior: A theoretical analysis.* New York: John Wiley & Sons, Inc.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., & Teller, E. (1953). Equation of state calculations by fast computing machines. *Journal of Chemical Physics*, *21*(6), 1087–1092.

Rao, C. R. (1958). Some statistical methods for comparison of growth curves. *Biometrics*, *14*(1), 1–17. Available from `http://www.jstor.org/stable/2527726`

Rasch, G. (1960). *Probabilistic models for some intelligence and attainment tests.* Copenhagen: The Danish Institute of Educational Research. (Expanded edition, 1980. Chicago: The University of Chicago Press.)

Stoel, R. D., van den Wittenboer, G., & Hox, J. J. (2004). Methodological issues in the application of the latent growth curve model. In

K. van Montfort, H. Oud, & A. Satorra (Eds.), *Recent developments in structural equation modeling: Theory and applications* (pp. 241–262). Amsterdam: Kluwer Academic Publishers.

Visser, I., Raijmakers, M. E. J., & van der Maas, H. L. J. (2009). Hidden Markov models for individual time series. In J. Valsiner, P. C. M. Molenaar, M. C. D. P. Lyra, & N. Chaudhary (Eds.), *Dynamic process methodology in the social and developmental sciences* (pp. 269–289). New York: Springer.

Zermelo, E. (1929). Die berechnung der turnier-ergebnisse als ein maximumproblem der wahrscheinlichkeitsrechnung. *Mathematische Zeitschrift*, *29*, 436–460.