J Neurophysiol 119: 1367–1393, 2018.
First published November 29, 2017; doi:10.1152/jn.00197.2017.

INNOVATIVE METHODOLOGY | *Control of Movement*

Estimating properties of the fast and slow adaptive processes during sensorimotor adaptation

Scott T. Albert and Reza Shadmehr
Laboratory for Computational Motor Control, Department of Biomedical Engineering, Johns Hopkins School of Medicine, Baltimore, Maryland

Submitted 20 March 2017; accepted in final form 24 November 2017

Albert ST, Shadmehr R. Estimating properties of the fast and slow adaptive processes during sensorimotor adaptation. *J Neurophysiol* 119: 1367–1393, 2018. First published November 29, 2017; doi:10.1152/jn.00197.2017.—Experience of a prediction error recruits multiple motor learning processes, some that learn strongly from error but have weak retention and some that learn weakly from error but exhibit strong retention. These processes are not generally observable but are inferred from their collective influence on behavior. Is there a robust way to uncover the hidden processes? A standard approach is to consider a state space model where the hidden states change following experience of error and then fit the model to the measured data by minimizing the squared error between measurement and model prediction. We found that this least-squares algorithm (LMSE) often yielded unrealistic predictions about the hidden states, possibly because of its neglect of the stochastic nature of error-based learning. We found that behavioral data during adaptation was better explained by a system in which both error-based learning and movement production were stochastic processes. To uncover the hidden states of learning, we developed a generalized expectation maximization (EM) algorithm. In simulation, we found that although LMSE tracked the measured data marginally better than EM, EM was far more accurate in unmasking the time courses and properties of the hidden states of learning. In a power analysis designed to measure the effect of an intervention on sensorimotor learning, EM significantly reduced the number of subjects that were required for effective hypothesis testing. In summary, we developed a new approach for analysis of data in sensorimotor experiments. The new algorithm improved the ability to uncover the multiple processes that contribute to learning from error.

NEW & NOTEWORTHY Motor learning is supported by multiple adaptive processes, each with distinct error sensitivity and forgetting rates. We developed a generalized expectation maximization algorithm that uncovers these hidden processes in the context of modern sensorimotor learning experiments that include error-clamp trials and set breaks. The resulting toolbox may improve the ability to identify the properties of these hidden processes and reduce the number of subjects needed to test the effectiveness of interventions on sensorimotor learning.

expectation maximization; motor learning; two-state model

INTRODUCTION

When people and other animals perform a movement that produces an unexpected outcome, they learn from the resulting error and retain a portion of this learning over time. Analysis of behavior in numerous contexts, including saccade paradigms (Ethier et al. 2008; Kojima et al. 2004), reach paradigms (Smith et al. 2006; Criscimagna-Hemminger and Shadmehr 2008; Pekny et al. 2011), vestibular paradigms (Colagiorgio et al. 2015), and classical conditioning paradigms (Stollhoff et al. 2005), has revealed an interesting behavioral property termed spontaneous recovery; following learning, washout, and then passage of time, behavior spontaneously reverts back to the previously learned state. That is, washout does not return memory to its baseline condition but appears to engage a process that masks the previously acquired memory. With passage of time, this mask appears to lift, resulting in reexpression of the learned behavior.

Spontaneous recovery is consistent with a mathematical model of learning where experience of error engages two (or more) independent learning processes, a fast process that learns strongly from error but forgets rapidly and a slow process that learns weakly from error but exhibits robust retention (Kording et al. 2007; Smith et al. 2006). It is possible that the putative learning processes represent interactions between distinct neural systems such as the cerebellum, the motor cortex, and the parietal cortex (Galea et al. 2011; Herzfeld et al. 2014b; Kim et al. 2015). The learning processes may also be represented in behavior as explicit and implicit processes (McDougle et al. 2015), body vs. world estimation (Berniker and Kording 2011), temporally labile vs. temporally stabile processes (Hadjiosif and Smith 2015), a memory of errors (Herzfeld et al. 2014b; Leow et al. 2016), and preparation-time dependent processes (Haith et al. 2015). In all of these approaches, experience of error engages multiple hidden processes that act in parallel, each responding to error with their own characteristic learning and retention properties and then combining their outputs to jointly influence behavior.

An example of a neural system that might implement such a learning model is the cerebellum and its principal cells, the Purkinje cells (P cells). Following experience of a visual error, some P cells prefer that error (Herzfeld et al. 2015; Kojima et al. 2010) and experience a strong modulation of their complex spikes. Experience of a complex spike in a P cell produces plasticity among some of the synapses, resulting in a reduction in the simple spikes that the P cell produces on the subsequent trial (Yang and Lisberger 2014). This resembles a learning process that adapts strongly from error. Interestingly, these

Address for reprint requests and other correspondence: S. Albert, Johns Hopkins University School of Medicine, 720 Rutland Ave., 416 Traylor Bldg., Baltimore, MD 21205 (e-mail: salbert8@jhmi.edu).

neurons also exhibit rapid forgetting, displaying little or no retention (in terms of change in their simple spikes) after 10 s of time has passed since experience of error (Yang and Lisberger 2014). Other P cells do not prefer that same error; for them, that error produces suppression of their complex spikes below baseline, resulting in a weak potentiation of their simple spiking rate. This resembles a learning process that adapts weakly from error. With repeated trials, the experience of these errors produces two time scales of change in the simple spikes, fast change in the P cells that prefer the error and slow change in the P cells that do not prefer the error (Yang and Lisberger 2014). Therefore, one potential neural mechanism for the multiple learning processes may be in the cerebellum, where the various P cells learn differently from a given error based on their preference for the direction of that error.

Here, our aim was to build a mathematical tool that could, in principle, extract the hidden processes from observed behavior. A common tool currently used for analysis of behavioral data in motor learning is a form of nonlinear optimization called least mean square error estimation (LMSE). This algorithm begins with a state space model of learning and then searches the model's parameter space to minimize the sum of squared differences between the observed behavior and the model's predictions. LMSE has been widely applied to analyze trial-by-trial changes of behavior during motor learning (Colagiorgio et al. 2015; Galea et al. 2015; McDougle et al. 2015; Trewartha et al. 2014). However, we found that when we applied LMSE to behavioral data collected during a typical adaptation experiment, the algorithm fit the measured data well, but for many subjects, it yielded unrealistic predictions about the properties of the underlying hidden processes. We speculated that this problem was due to a fundamental limitation of LMSE; in the context of error-based learning where the errors we make are influenced by the movements we generate, the LMSE algorithm is equivalent to a maximum likelihood estimator for a system that is ignorant of the stochastic nature of learning and moving.

Therefore, we wondered whether an algorithm that considered both of these sources of stochasticity, noise in the system that learned from error and noise in the system that produced the motor output, could improve our ability to estimate the hidden processes. We derived a canonical form of the two-state model that cast the learner in a framework where both learning from error and the production of a movement were stochastic processes (Cheng and Sabes 2006; Tanaka et al. 2012). In this framework, there was uncertainty in both the evolution of hidden states and the observation of movement (Kording et al. 2007).

To estimate parameters of this more general model of learning, we considered a maximum likelihood approach that was first applied to sensorimotor learning by Cheng and Sabes (2006) called expectation maximization (EM). Unfortunately, it is difficult to constrain EM to enforce traditional two-state dynamics. In addition, previous descriptions of the algorithm assumed time-invariant state space transitions (Ghahramani and Hinton 1996). In contrast, a typical motor control experiment relies on behavioral probes such as error clamp trials (Scheidt et al. 2000) and set breaks. The latter type of probe can make the state space transitions time dependent.

Here, we illustrate how a generalized EM algorithm can be used to estimate the hidden processes that may underlie a learning problem, even when constraints are applied to the model parameters, error clamp trials are included in the experimental paradigm, and the generative model of learning varies in time due to the occurrence of set breaks. The result is a new mathematical toolbox.

We demonstrate that EM fits observed human reaching behavior similarly to LMSE but uses an underlying parameter set whose likelihood is more likely to explain the observed behavior. To further evaluate EM, we consider several simulated sensorimotor learning paradigms. Unlike behavioral data measured in the laboratory, the fast and slow learning processes were explicitly known in the simulated data sets, allowing us to objectively quantify performance of EM.

Our work has two main results: *1*) behavior during sensorimotor learning is better represented by a generative model in which both the generation of movement and learning from error are stochastic processes, and *2*) in such a system, EM significantly improves the ability to uncover the hidden states of learning. The resulting algorithm has the practical implication of reducing the number of subjects that are needed for statistical testing of hypotheses.

METHODS

Overview. Our goal was to produce a mathematical toolbox that could robustly estimate the properties of a two-state learning process from data collected in a typical adaptation experiment. We employed a statistical algorithm known as EM. EM is an iterative parameter estimation technique that can be used for system identification in the presence of latent variables. As its name suggests, EM is composed of two separate steps. In the expectation step (E-step), a Kalman filter is used to provide the best estimate of the hidden states under the current estimate of the model parameters. In the maximization step (M-step), maximum-likelihood estimation is used to identify a set of model parameters that maximize an objective function known as the expected complete log-likelihood function. The E- and M-steps together are guaranteed to identify model parameters that improve the likelihood of observing the measured data. The E- and the M-steps are iterated until the likelihood of observing the measured data converges.

Current application of EM in the sensorimotor literature is limited to linear time-invariant (LTI) systems (Cheng and Sabes 2006), where the generative model assumes no constraints on the dynamics of the hidden learning processes. In this case, the E- and M-steps can be performed analytically via closed-form equations. Although closed-form equations simplify one's search for the optimal parameter set, performing this analytical formulation makes it difficult to enforce the conventional properties of a two-state model (e.g., the fast process forgets more rapidly than the slow process). In addition, closed-form expressions for the M-step of the algorithm cannot always be derived for complicated likelihood functions, as is the case when set breaks are included in the generative model. To rectify these issues, we considered a more general form of the EM algorithm, aptly named generalized EM (Dempster et al. 1977). Our version of this algorithm is similar to previous descriptions of EM in the sensorimotor literature, differing only in the implementation of the M-step. In our algorithm, numerical techniques are used to search for the maximum value attained by the expected complete log-likelihood function within a constrained parameter space. Our implementation is described in APPENDIX 1: MATHEMATICAL DESCRIPTION OF THE GENERALIZED EM ALGORITHM.

To assess the performance of our approach, we tested EM against a different technique for fitting state-space models to behavioral data, LMSE. LMSE is a technique in which one identifies model parameters that minimize the mean-squared-error between the measured behavior and the model predicted behavior. In an error-based learning model,

where movement results in errors that teach the motor learning system, LMSE is identical to a maximum likelihood estimator for a model that assumes that motor learning and movement production are deterministic processes, with uncertainty only arising in our measurement of the behavior of the subject, and is described in detail in APPENDIX 2: OVERVIEW OF THE LMSE ALGORITHM.

We applied our EM and LMSE algorithms to data recorded from human subjects performing a visuomotor rotation task. Additionally, we simulated various paradigms and generated data sets in which the fast and slow states were known. Using these data, we asked how accurately EM and LMSE could uncover the true states. Finally, we used EM and LMSE to perform a power analysis, estimating how many subjects were needed to robustly test the effectiveness of an intervention that modified error sensitivity and retention in a simulated population.

State space model of learning. Here, we derive a two-state model of learning. Our model possesses the same canonical form as previous models in the literature but differs in a fundamental way; we mathematically formalize the inclusion of error clamp trials and set breaks. This modification yields a more complicated time-varying form of the state-space equations, but also makes our approach compatible with general sensorimotor adaptation paradigms.

A learner is presented with a sequence of trials where she is instructed to make a movement towards a target. On trial n, she is presented with the target $g^{(n)}$. To achieve this target, she produces a movement, $u^{(n)}$, and observes the consequences of her action $h^{(n)}$. The consequence of her action (*Eq. 1*) is determined by her movement as well as any external perturbation to her movement, denoted by $r^{(n)}$:

$$h^{(n)} = u^{(n)} + r^{(n)} \tag{1}$$

The learner adjusts her movement toward the target according to her estimate of the perturbation $\hat{r}^{(n)}$. The movement she produces is altered by motor execution noise $\varepsilon_u^{(n)}$, which has a normal distribution with mean 0 and variance σ_u^2. We have

$$u^{(n)} = g^{(n)} - \hat{r}^{(n)} + \varepsilon_u^{(n)}. \tag{2}$$

In a two-state model, we assume that the learner estimates the perturbation via two independent states, referred to as the slow and fast states of learning. The values of the two states on trial n are represented by the vector $\boldsymbol{x}^{(n)} = [x_s^{(n)} \;\; x_f^{(n)}]^T$where x_s and x_f are the scalar-valued slow and fast states. The learner's estimate of the perturbation is related to these states according to

$$\hat{r}^{(n)} = \boldsymbol{c}^T \boldsymbol{x}^{(n)}. \tag{3}$$

Here, $\boldsymbol{c} = [1\ 1]^T$, meaning that the learner's estimate is equal to the sum of the fast and slow states.

Over time, the learner adjusts her estimate of the perturbation according to the errors she experiences. Error, denoted by $e^{(n)}$, is the difference between the observed outcome of the movement, and the target:

$$\begin{aligned} e^{(n)} &= h^{(n)} - g^{(n)} \\ &= r^{(n)} - \boldsymbol{c}^T \boldsymbol{x}^{(n)} + \varepsilon_u^{(n)}. \end{aligned} \tag{4}$$

Note that this error could be further manipulated by the experimenter. In some cases, the experimenter can add additional noise to the observed movement to increase feedback uncertainty (Wei and Körding 2010). One could also explicitly attempt to account for uncertainty in visual or proprioceptive transduction of error. We remark on these sources of noise further below.

A common experimental manipulation is the occurrence of an error clamp trial (Scheidt et al. 2000). On these trials, the learner is presented with an error that is independent of the movement she performed. Therefore, the error experienced by the learner can take different functional forms, depending on the trial type, according to

$$e^{(n)} = \begin{cases} r^{(n)} - \boldsymbol{c}^T \boldsymbol{x}^{(n)} + \varepsilon_u^{(n)}, & \text{not an error clamp trial} \\ e_c^{(n)}, & \text{error clamp trial.} \end{cases} \tag{5}$$

Here, the variable $e_c^{(n)}$ takes the value of the error imposed on trial n.

Two separate processes determine how the learner's estimate of the perturbation changes from one trial to the next: learning and forgetting. Together, learning and forgetting are captured by the state update equation:

$$\boldsymbol{x}^{(n+1)} = A\boldsymbol{x}^{(n)} + \boldsymbol{b}e^{(n)} + \varepsilon_{\boldsymbol{x}}^{(n)}. \tag{6}$$

The forgetting process is controlled by the matrix A, which encodes the rate at which states decay in the absence of error due to the passage of time. If we assume that each state evolves independently, we can represent A as a diagonal matrix of the form $A = \begin{bmatrix} a_s & 0 \\ 0 & a_f \end{bmatrix}$. Here, a_s and a_f are retention factors for the slow and fast states, respectively.

The learning process is controlled by the vector $\boldsymbol{b}$, which encodes the learning rates of the fast and slow states. The parameter $\boldsymbol{b}$ is a 2×1 vector of the form $\boldsymbol{b}=[b_s \; b_f]^T$, where b_s and b_f are the error sensitivities of the slow and fast states, respectively. These error sensitivities determine the rate at which each state learns from error.

The entire process of updating the learner's estimate of the perturbation, like the process of generating a movement, is affected by noise, represented by $\boldsymbol{\varepsilon}_x$. This noise source represents the combined effect of many sources of noise that accumulate in afferent pathways involved in learning from error. These include but are not limited to noise in the proprioceptive and visual transduction of error, noise in an error stimulus itself, and noise in the synaptic mechanisms that contribute to learning from a given error, etc. We will refer to the collection of these processes as state update noise and assume that it is distributed according to a multivariate normal distribution with mean $[0\ 0]^T$ and variance-covariance matrix Q. In accord with our assumption that the two states evolve independently, we require that their covariance be equal to zero, implying that Q is a 2×2 diagonal matrix. To simplify the model, we assumed that the fast- and slow-state update variances were equal, yielding a variance-covariance matrix of the form $Q = \begin{bmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_x^2 \end{bmatrix}$, where σ_x^2 represents the cumulative state update variance described above. In a set of control analyses, we also considered a model of learning where the fast and slow states had different noise variances (see *Different noises in the fast and slow adaptive processes*).

Equation 6 treats the errors experienced in error clamp and non-error clamp trials the same. That is, we assume that the learner does not differentiate from an error that was produced by her own behavior and an error that was presented to her in an error clamp trial. The validity of this assumption is currently under debate. Although blocks of error clamp trials have been used extensively in the literature to assess decay properties of motor memory, two recent reports have found evidence that in some cases error clamp blocks appear to contain contextual cues that can affect the process of learning (Vaswani and Shadmehr 2013; Vaswani et al. 2015). In contrast, another report has found evidence that learning in error clamp trials remains consistent with learning in non-error clamp trials (Brennan and Smith 2015) or that differences in learning become evident only after long passages of time away from the task (Pekny and Shadmehr 2015). Given that this question remains unanswered, here we chose the simplest model wherein all errors were treated equally.

We do not directly observe the states of the fast and slow processes. Instead, we measure the movement of the subject on each trial. In many experiments, to normalize across different targets that may be presented to the subject, it is useful to define the subject's movement relative to the target location, $y^{(n)}$:

$$\begin{aligned} y^{(n)} &= g^{(n)} - u^{(n)} \\ &= \boldsymbol{c}^T \boldsymbol{x}^{(n)} - \varepsilon_u^{(n)}. \end{aligned} \tag{7}$$

Substitution of the motor action in *Eq. 7* into our expression for error in *Eq. 5* yields the following simplification:

$$e^{(n)} = \begin{cases} r^{(n)} - y^{(n)}, & \text{not an error clamp trial} \\ e_c^{(n)}, & \text{error clamp trial.} \end{cases} \quad (8)$$

Finally, we account for set breaks in our generative model of learning, noting that set breaks result in significant forgetting of previously learned behavior (Ethier et al. 2008). Therefore, we imagined that set breaks could be modeled as additional decay that elapses after the conclusion of the trial preceding a set break:

$$\boldsymbol{x}^{(n+1)} = D^{(n)}\left[A\boldsymbol{x}^{(n)} + \boldsymbol{b}e^{(n)} + \boldsymbol{\varepsilon}_{\boldsymbol{x}}^{(n)}\right] \qquad \boldsymbol{\varepsilon}_{\boldsymbol{x}}^{(n)} \sim N\left(\begin{bmatrix}0 & 0\end{bmatrix}^T, Q\right)$$
$$\text{where} \quad D^{(n)} = \begin{cases} A^d = \begin{bmatrix} a_s^d & 0 \\ 0 & a_f^d \end{bmatrix}, & \text{trial n is followed by a set break} \\ \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, & \text{trial n is not followed by a set break.} \end{cases} \quad (9)$$

The parameter d in *Eq. 9* is a decay factor that parametrizes elapsed time between trials to account for additional forgetting of the fast and slow states (Joiner and Smith 2008) across a set break. A value of $d=0$ means that a set break results in no further forgetting beyond that which accompanies an experimental intertrial interval (ITI). A positive value of d indicates that a set break results in more forgetting than an experimental ITI. Here, we assumed that all set breaks are an equal length of time, although *Eq. 9* could be easily modified to allow for set breaks of variable length by replacing d by the product of d with the ratio of the duration of a set break to the average ITI.

The form of *Eq. 9* assumes that the rate of decay of the fast and slow states follows the retention properties that are observed trial by trial in the absence of set breaks. Our two-state model can now be represented as the following system of state-space equations that account for both error clamp trials and set breaks:

$$\begin{aligned} &\boldsymbol{x}^{(n+1)} = A^{(n)}\boldsymbol{x}^{(n)} + \boldsymbol{b}^{(n)}e^{(n)} + \boldsymbol{\varepsilon}_{\boldsymbol{x}}^{(n)} && \boldsymbol{\varepsilon}_{\boldsymbol{x}}^{(n)} \sim N\left(\begin{bmatrix}0 & 0\end{bmatrix}^T, Q^{(n)}\right) \\ &y^{(n)} = \boldsymbol{c}^T\boldsymbol{x}^{(n)} + \boldsymbol{\varepsilon}_u^{(n)} && \boldsymbol{\varepsilon}_u^{(n)} \sim N\left(0, \sigma_u^2\right) \\ &A^{(n)} = \begin{cases} A & \text{no set break} \\ A^{d+1} & \text{set break} \end{cases} && Q^{(n)} = \begin{cases} Q & \text{no set break} \\ A^d Q A^{d^T} & \text{set break} \end{cases} \\ &\boldsymbol{b}^{(n)} = \begin{cases} \boldsymbol{b} & \text{no set break} \\ A^d\boldsymbol{b} & \text{set break} \end{cases} && e^{(n)} = \begin{cases} r^{(n)} - y^{(n)} & \text{not an error clamp trial} \\ e_c^{(n)} & \text{error clamp trial} \end{cases} \end{aligned} \quad . \quad (10)$$

In *Eq. 10*, the "no set break" condition indicates that trial n is not followed by a set break. The "set break" condition indicates that trial n is followed by a set break. The "not an error clamp trial" condition indicates that trial n was not an error clamp trial. The "error clamp trial" condition indicates that trial n was an error clamp trial. Note that the sign of the motor noise in *Eq. 10* was flipped; because this noise is Gaussian with mean zero, changing its sign describes an equivalent system.

Experimental procedure. We recruited $n = 20$ healthy right-handed subjects (ages 17–59 yr, 8 male) to perform a visuomotor adaptation study. All subjects signed a consent form approved by the Johns Hopkins University School of Medicine Institutional Review Board before participating in the experiment.

Subjects were seated in a chair and held the handle of a planar robotic manipulandum. The arm of the subject was obscured from view, and the position of the hand was represented by a white cursor projected onto the screen situated directly on top of the hand. The x- and y-positions of the manipulandum (i.e., the subject's hand) were recorded at 200 Hz from optical encoders at a resolution of >0.1 mm using custom C++ code. Subjects were instructed to move their hand from a starting circle through a target circle (radius = 1 cm). The target circle was presented in one of eight positions in the workspace at a displacement of 10 cm. On some trials, the subject was provided no visual feedback of the cursor. We term these trials "no-feedback trials." Apart from these no-feedback trials, subjects had continual visual feedback of their hand position during the outward reach. Our experiment employed a single perturbation condition. The perturbation was a 30° counterclockwise rotation to the cursor position, about the starting position. The subject was awarded a point each time the cursor passed through the target within 190–290 ms following movement onset. Subjects were instructed to obtain as many points as possible during the experiment.

An epoch consisted of eight trials, one to each of the eight targets around the circle, chosen in a random sequence. The task began with a baseline period of 30 epochs where subjects reached without any perturbation (Fig. 1*A*). We interspersed three epochs of no-feedback trials within this baseline period to familiarize subjects with this condition. The baseline period was followed by a block of 30 perturbation epochs. After the perturbation period, visual feedback was removed for 15 epochs. After this no-feedback period, feedback was reinstated and the perturbation removed (washout trials) for 30 epochs.

Our analysis focused on the hand end-point error, which was taken as the angular displacement of the hand from the target when the subject's hand displacement exceeded the 10-cm target displacement.

Epoch vs. trial-by-trial analysis of behavior. The two-state model described in *Eq. 10* is readily applied to trial-by-trial data where only a single target is presented during the experiment. However, in our visuomotor rotation task, as in many experiments, the paradigm consisted of multiple targets. In paradigms with multiple targets, the learning process should be represented with different states for each target, that is, separate fast and slow states of learning for each target in the workspace. This expansion of the hidden state dimensionality is further complicated by generalization of learning across targets. That is, the error experienced when the subject moves toward one target generalizes to other targets in the workspace, resulting in differential amounts of learning across the workspace (Krakauer et al. 2000).

In our task, we did not directly probe this generalization function. Therefore, we attempted to minimize the effect of generalization on the learning process by averaging behavior across the eight trials/targets visited in each epoch of the experiment. In this case, whereas

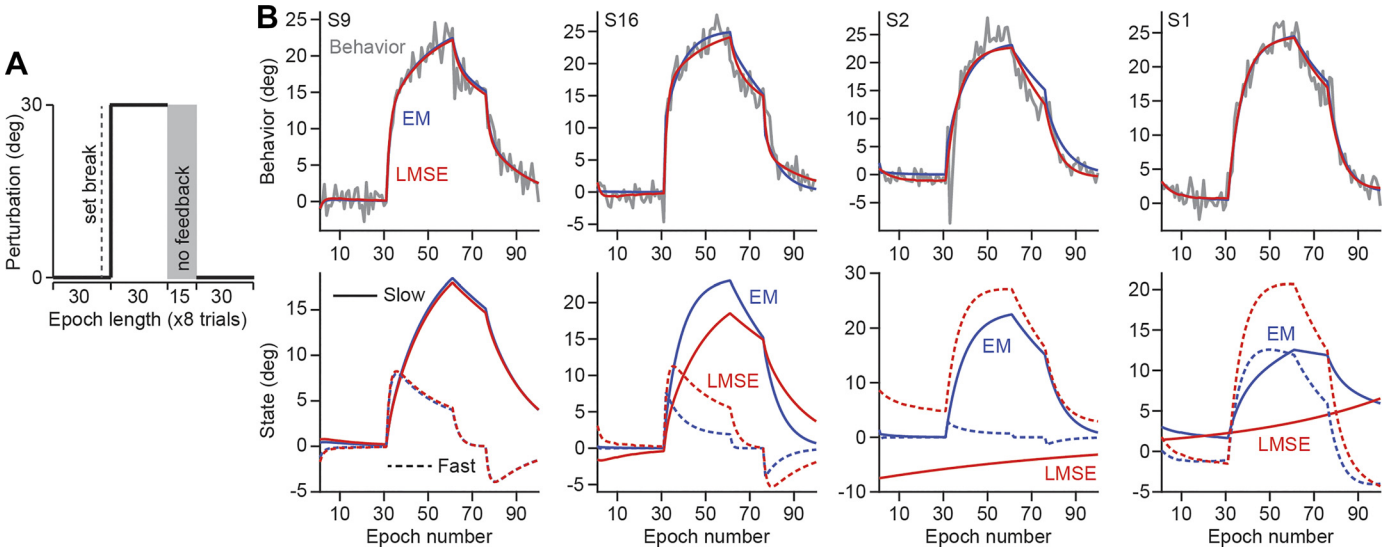

Fig. 1. Experimental paradigm: expectation maximization (EM) and least mean square error estimation (LMSE) algorithms uncover different hidden processes. *A*: subjects ($n = 20$) participated in a reach adaptation task. There were 8 targets in total, each chosen pseudo-randomly and presented once in epochs of 8 trials. Following a no-perturbation baseline period, a 30° counterclockwise rotation was applied to the cursor representing the subject's hand position. After 30 epochs of this perturbation, visual feedback was removed for 15 epochs. Finally, visual feedback was reinstated during a washout block of 30 epochs. *B*: single-subject behavior. We fit the epoch-by-epoch data (reach direction) of each subject with EM (blue lines; *top*) and LMSE (red lines; *top*). Both provide good fits to the measured data. Each algorithm estimated the fast and slow processes that produced the measured behavior (*bottom*). For *subject S9*, these time courses agreed across algorithms. For *subject S16*, EM and LMSE time courses exhibited reasonable two-state behavior but had differing learning dynamics. For *subjects S2* and *S1*, the EM and LMSE predictions diverged completely.

the same target may not be visited from one trial to the next, the same set of targets is visited from one epoch to the next, reducing the effect of generalization on the recorded epoch-by-epoch behavior. Unless otherwise noted, all analysis of subject behavior in our visuomotor rotation task is based on application of the two-state model in *Eq. 10* to epoch-by-epoch behavior.

However, we also considered two-state model fits to the raw trial-by-trial data. To apply our two-state model to the trial-by-trial data, we first had to decide how best to deal with generalization. We considered two extreme cases: *1*) where learning generalized completely from one target to all other targets and *2*) where there was no generalization of learning to any other targets. We will refer to these trial-by-trial models as our full-generalization and no-generalization models, respectively. In the full-generalization model, we assumed that the learner used a single fast and slow state to account for the perturbation for all targets as in *Eq. 10*. The fast and slow states were ignorant of the target location and fully generalized learning that occurred from one target to all other targets. Therefore, in the full-generalization model, we applied *Eq. 10* to the trial-by-trial data as if the same target had been visited on each trial.

In the no-generalization trial-by-trial model, learning from one target did not generalize to other targets. This trial-by-trial model had 16 states, one fast and one slow state for each of the 8 targets. All fast/slow states experienced forgetting on every trial according to a common fast/slow retention factor. However, if target k was visited on trial n, then only the fast and slow states associated with target k learned from the error experienced on that trial (all fast/slow states had a common fast/slow error sensitivity). Furthermore, all fast and slow states were subject to a non-zero state noise on each trial. In APPENDIX 3: MULTIPLE TARGET STATE-SPACE MODEL OF LEARNING, we provide a complete description of this model and show how it can be extended to account for generalization that might extend beyond the width of a single target.

To summarize, the majority of our analysis of behavioral data were averaged in epochs of eight trials where all eight targets were visited once in the epoch. In a set of control analyses (see *Trial-by-trial analysis of behavior*), we considered two trial-by-trial models that covered the extremes of generalization, one where learning generalized completely across targets and one where learning did not generalize at all across targets. For the simulations (see *Simulating realistic data*), we never performed averaging of the simulated behavior.

Simulating realistic data. To test our algorithm, we simulated realistic data using the two-state system described in *Eq. 10*. We simulated four paradigms (Fig. 3*A*) commonly encountered in the literature. These paradigms differed with respect to the inclusion or exclusion of error clamp trials and set breaks as well as the manner in which the perturbation was introduced (i.e., abruptly vs. gradually).

All paradigms began with a baseline period in which the learner was simulated for many trials in the absence of any external perturbation to her movements. In *paradigm 1* (Fig. 3) this perturbation was followed by an abrupt introduction of a perturbation. We used a 30° perturbation to match our visuomotor rotation task. In *paradigm 2* (Fig. 3), we built upon *paradigm 1* by adding a prolonged error clamp period that followed the perturbation, where error was completely eliminated. This type of intervention is a common way to isolate and measure retention of learned behavior. *Paradigm 2* concluded with a washout period that allowed the simulated learner to return to baseline behavior. *Paradigm 2* closely resembled our visuomotor task, considering that the simulated error clamp trials produced qualitatively similar behavior to that demonstrated by our subjects during no-feedback trials. In *paradigm 3*, we simulated a learner in a gradual perturbation environment, followed again by an error clamp period. In *paradigm 4*, we built upon the other paradigms by adding set breaks. In this simulated experiment, we followed a trial structure that is known to promote spontaneous recovery of behavior (Ethier et al. 2008; McDougle et al. 2015; Smith et al. 2006). After being exposed to a positive 30° perturbation, the sign of the perturbation is abruptly switched until the learner expresses approximately baseline behavior. Then, a block of error clamp trials is provided to test for spontaneous recovery (Fig. 3, *paradigm 4*).

For each paradigm, we simulated subject behavior 1,000 times using fixed model parameter values. On each run, we varied the seed for the random number generator, which resulted in different learning profiles due to motor and state noise. To simulate realistic data sets, we selected retention factors and error sensitivities that matched

Table 1. *Two-state model parameters*

Parameter	a_s	a_f	b_s	b_f	σ_x^2, °	σ_u^2, °^{2}	$x_s^{(1)}$, °	$x_f^{(1)}$, °	σ_1^2, °^{2}	d
Value	0.985	0.556	0.097	0.213	1.694	1.037	0	0	0	8

Model parameters used for the simulated experiments in our primary analyses. We selected these parameters specifically to match the dynamics of learning observed in our experimental data (Fig. 1). To do this, we fit epoch-by-epoch single subject behavior with expectation maximization (EM) and least mean square error (LMSE) estimation. For the retention factors and error sensitivities, we computed the average parameter value across all 20 subjects and both the EM and LMSE fits. We assumed that both the initial slow and fast states were equal to 0 to represent a naïve learner. For the initial state variance, we used a value of 0, indicating that each simulated fast and slow state truly began at the value zero for all simulations. For the state and motor noise, we used the mean variances predicted by the EM algorithm, as LMSE does not separately measure these noise terms. Finally, note that the parameter d only applies for simulation of *paradigm 4*, which included set breaks.

parameters estimated from our experimental data; we fit each subject's reaching behavior using EM and LMSE (described below) and used the mean parameter values across subjects and algorithms for the simulated retention factors and error sensitivities. For the state and motor noise variances, we selected the mean values obtained using the EM algorithm, as LMSE does not provide an estimate of these two noise sources. The complete parameter set is reported in Table 1. Three additional parameters appear in Table 1 that were not discussed in our derivation of the two-state model. These parameters are related to the initial state of the learner (see APPENDIX 1: MATHEMATICAL DESCRIPTION OF THE GENERALIZED EM ALGORITHM). We modeled the initial fast and slow states of the learner as normally distributed random variables with mean $x_s^{(1)}$ and $x_f^{(1)}$, respectively, and common variance σ_1^2. For our simulations, we considered a naïve learner who had an initial slow and fast state equal to zero.

Fitting EM and LMSE to data. We fit our measured data collected in the visuomotor rotation experiment and the simulated data in an identical manner. Both the EM and LMSE algorithms were given the observed motor actions y. Each algorithm then used its objective function to identify an estimate of the model parameters. This process is described in APPENDIX 1: MATHEMATICAL DESCRIPTION OF THE GENERALIZED EM ALGORITHM for EM and APPENDIX 2: OVERVIEW OF THE LMSE ALGORITHM for LMSE.

To obtain the model parameters, both algorithms were treated identically; they were numerically constrained to search an identical parameter space using the function *fmincon* in MATLAB R2016a. Our constrained parameter space was defined by upper and lower bounds as well as linear inequality constraints relating some of the parameters. The upper and lower bounds for each parameter are provided in Table 2. Linear inequality constraints were specified to enforce traditional two-state model dynamics according to

$$\begin{aligned} a_s &\geq a_f + 0.001 \\ b_f &\geq b_s + 0.001 \end{aligned} \quad (11)$$

The first of these inequalities requires that the slow state be retained more strongly trial by trial than the fast state. The second of these inequalities requires that the fast state learn more rapidly from error than the slow state.

EM is an iterative algorithm that attempts to increase the value of the likelihood function from one iteration to the next (see APPENDIX 1: MATHEMATICAL DESCRIPTION OF THE GENERALIZED EM ALGORITHM). For each EM fit, we performed 100 iterations of the algorithm. The EM algorithm is sensitive to the initial conditions used to initialize the first iteration (see APPENDIX 1: MATHEMATICAL DESCRIPTION OF THE GENERALIZED EM ALGORITHM). For each simulation, we started the EM algorithm from five different initial guesses. For our experimental data, we used 10 different initial guesses for each subject. These initial guesses were randomly sampled from the constrained parameter space. We selected the parameter set with the greatest likelihood at the conclusion of the one-hundredth iteration of the algorithm. Numerical implementation of the LMSE algorithm can also require different initial guesses for proper convergence of the *fmincon* algorithm. For LMSE, we seeded the *fmincon* search using 50 different starting parameter sets to better ensure the identification of minimal squared error within the constrained parameter space. As for the EM algorithm, these initial starting parameter sets were also sampled randomly from within the constrained parameter space.

Measuring the performance of the algorithms. We assessed how well EM and LMSE recovered the properties of the fast and slow states of learning in our simulated experiments, where the hidden states were explicitly known. After obtaining the parameter sets for EM and LMSE, we asked how well they predicted the time courses of the fast and slow states. To do this, we used each parameter set to simulate noise-free behavior. The noise-free version of our two-state model was obtained by removing the noise terms from *Eq. 10*:

$$\begin{aligned} \boldsymbol{x}^{(n+1)} &= A^{(n)}\boldsymbol{x}^{(n)} + \boldsymbol{b}^{(n)}e^{(n)} \\ y^{(n)} &= \boldsymbol{c}^T\boldsymbol{x}^{(n)} \end{aligned} \quad (12)$$

This noise-free system is equivalent to the expected value of the hidden states and observed behavior at any point in time. We compared the noise-free time courses of the slow state, fast state, and overall behavior, predicted by EM and LMSE, to the actual time courses for each simulation. To determine how well the EM and LMSE time courses matched the actual time courses, we computed the root mean squared error (RMSE) between the model fit and the actual data.

We also asked how well the EM and LMSE parameter sets matched the true parameter set. For this, we computed the absolute error between the fitted parameters and the underlying two-state model parameters used to simulate the data.

Sources of noise. In our model of learning, we considered two potential sources of noise, one in the generation of an action and the other in learning from error. Our EM algorithm identifies a parameter set that maximizes a likelihood function that attributes randomness in measured behavior to these two processes. In contrast, the likelihood function maximized by the LMSE parameter set attributes randomness in measured behavior to the measurement of the behavior itself and assumes that the underlying learning system behaves deterministically (see APPENDIX 2: OVERVIEW OF THE LMSE ALGORITHM). Therefore, the critical difference between EM and LMSE is the manner in which their corresponding likelihood functions account for variance in measured behavior.

To compare these likelihood models, we turned to our experimental data. We computed the corrected Akaike Information Criterion (AICc) for the likelihood models maximized by EM (*Eq. A1.25*) and LMSE (*Eq. A2.4*). AICc is a metric that can be used to compare the likelihoods associated with different models discounted by the number of parameters contained by these models. The corrected Akaike Information Criterion (AICc) differs from the conventional Akaike Information Criterion (AIC) by further penalizing the number of

Table 2. *Upper and lower bounds*

Parameter	a_s	a_f	b_s	b_f	σ_x^2, °^{2}	σ_u^2, °^{2}	$x_s^{(1)}$, °	$x_f^{(1)}$, °	σ_1^2, °^{2}	d
Value	0,1.1	0,1.1	0,1	0,1	10^{-7},10	10^{-7},10	−30,30	−30,30	10^{-7},10	10^{-6},30

When fitting our behavioral data and simulated data, EM and LMSE searched the same bounded parameter space. Here, we provide the upper and lower bounds used for each model fit.

parameters in the model. In this sense, it is a more conservative way to compare models that differ in parameter complexity. AICc is defined by the following equation:

$$AICc = 2k - 2\log_e\left(\{y\}_1^N \middle| \theta\right) + \frac{2k(k+1)}{n-k-1}$$

Here, k refers to the number of model parameters and n is the number of data points used to fit the model. The smaller the value of AICc, the greater evidence for the corresponding model. Our likelihood model for EM included a state noise and motor noise but omitted noise in the measurement of subject behavior. Our likelihood model for LMSE included noise in the measurement of subject behavior but omitted state and motor noise in the underlying learning process. Therefore, because EM has two sources of noise and LMSE only one, the EM model had one greater parameter. We fit these models to the experimental data by searching for each model's maximum log-likelihood and, along with the number of model parameters, computed the AICc for each subject. We compared the AICc for both models using a paired t-test across subjects.

To compute the maximum log-likelihood of the LMSE model, we searched its incomplete (marginal) log-likelihood function directly (*Eq. A2.4*). We performed this search from 50 different initial points using *fmincon* in MATLAB 2016a. We used the same search space and model constraints for EM and LMSE, as described previously. Note that the likelihood model maximized by LMSE also neglects noise in the initial state of learning (see APPENDIX 2: OVERVIEW OF THE LMSE ALGORITHM). Therefore, we excluded this initial state variance from both of our likelihood models. The total number of parameters for the EM likelihood model and LMSE likelihood model was 8 and 7, respectively.

Power analysis. One way to study the motor learning system is to compare how subjects perform in different experimental conditions or at different points in time. For example, we may perform a savings experiment where we adapt subjects to a perturbation, wash out the adapted behavior, and then readapt subjects to the same perturbation (Leow et al. 2016). Typically, we find that subjects adapt to the perturbation faster the second time. We can ask how this savings is expressed; did subjects experience an increase in retention, an increase in error sensitivity, or perhaps both? In other words, our analyses typically involve comparisons. To determine whether an intervention resulted in a change in behavior, we can ask how the model parameters that describe learning changed as the result of the intervention. This statistical comparison is dependent on the variance in our estimates of two-state model parameters; as this variance increases, more subjects are required to obtain a statistically significant result.

Here, we imagined that we performed an intervention that resulted in a change to the error sensitivity of a population and a separate intervention that resulted in a change to the forgetting rate of the population. For the former, this is what is observed in savings paradigms (Herzfeld et al. 2014b; Leow et al. 2016; Smith and Shadmehr 2004). The latter has been observed in experiments that provide feedback in rewarding and punishing environments (Galea et al. 2015). We performed power analyses to determine how well EM and LMSE could detect changes in these two-state model parameters at different effect sizes. Our power analysis considered two forms of experiments, within-subject experiments and between-subject experiments. For our within-subject experiment, we imagined that a set of subjects performed *paradigm 2* at two different time points: *point A* and *point B*. For our between-subject experiment, we imaged that different sets of subjects performed *paradigm 2* in contrasting experimental *conditions A* and *B*, like a randomized control trial.

To generate data for *condition A*, we simulated the behavior of 1,000 subjects for *paradigm 2*, where each subject's parameter vector was sampled from a multivariate normal distribution that we estimated from our experimental data (Table 1) by computing the mean and covariance matrix of the two-state model parameters estimated for our 20 individual subjects. To produce performance during *exposure B*, we imagined that one of these parameters (i.e., a single entry in the parameter vector) had changed to a different value and then resimulated a new set of 1,000 subjects. We fit each simulated behavior using EM and LMSE. From this subject pool, we selected a certain number of subjects. For our within-subject experiment, we sampled the same subjects from the *exposure A* and *exposure B* periods. For our between-subject experiment, we sampled subjects independently from the *A* and *B* distributions. We next identified the parameter values that EM and LMSE predicted for the subjects in each exposure. For our within-subject experiment, we performed a paired t-test across the *exposure A* and *exposure B* parameter values to determine whether either algorithm could be used to detect a statistically significant change in the parameter value at a confidence level of 95%. For our between-subject experiment, we used a two-sample t-test. We repeated this analysis for a given number of subjects a total of 10,000 times, each time resampling subjects from our large 1,000-subject pool. We tracked the percentage of times that EM and LMSE yielded a statistically significant difference in the parameter value across the 10,000 experiments. We used this percentage as a measure of how reliably each algorithm detected statistical differences for a given group size. We used these data to ask how many subjects would be required in order for EM and LMSE to detect a statistical difference for $\geq$85% of the simulated experiments.

The parameter values we selected for the *exposure B* period were motivated by previous studies. For the slow state retention factor, we considered differences of -3 to $+1.5$% of the *exposure A* retention factor, in agreement with the dynamic range seen across subjects adapting to visuomotor rotations with rewarding and punishing feedback (Galea et al. 2015). We speculate that the difference in retention factor for the single-state fits in this study is most reflective of the slow state of learning during the error clamp period. For the fast state error sensitivity, we selected $\pm$15, 25, and 50% to cover the large differences in error sensitivity observed for individuals adapting in punishing and rewarding environments (Galea et al. 2015). This range in error sensitivity is also similar in magnitude to differences in error sensitivities of the slow and fast processes observed across the control and reporting groups in a study examining implicit and explicit components of learning in force field adaptation and visuomotor rotation learning (McDougle et al. 2015).

Control studies. We performed a set of control studies where we tested EM and LMSE identification accuracy in situations of greater noise, different dynamics for the fast and slow processes, and different assumptions about the structure of our two-state model of learning. For each of these control studies, we followed the same general approach; we simulated our two-state model and fit-simulated behavior with EM and LMSE to determine how well each algorithm identified the slow and fast states of learning. For certain control analyses, we reanalyzed our subject data. Each of our control studies is discussed in each of the sections in RESULTS. In *Modeling higher levels of noise*, we tested EM and LMSE on simulated data sets with much greater levels of state and motor noise. In *Different noises in the fast and slow adaptive processes*, we discuss evidence for the existence of two different variances for the update of the fast and slow states and perform a sensitivity analysis where we test EM and LMSE on simulated data sets with different levels of fast and slow state update variances. In *Other sources of noise*, we consider a way to model internal noise in the sensory observation of an error. In *Changing the dynamics of the fast and slow adaptive processes*, we tested EM and LMSE on data sets where the dynamics of the fast and slow processes differed from the average processes measured in the subject behavior investigated in our primary analysis. In *Changing the bounds on the parameter space*, we tested the identification accuracy of EM and LMSE if we further restricted the parameter space searched by both algorithms. Finally, in *Trial-by-trial analysis of behavior*, we show that fitting our state-space model to the trial-by-trial subject behavior, as opposed to eight-trial epochs of subject

behavior, does not have any effect on our primary conclusions. For brevity, we relegated further details of each of these control studies to the appropriate section in RESULTS.

RESULTS

Our aim was to design an algorithm that could uncover hidden processes that contribute to learning from error. We considered two algorithms, LMSE and EM. To perform our comparison, we asked volunteers ($n = 20$) to participate in an adaptation experiment where they reached to eight targets (Fig. 1*A*). The task consisted of a baseline period followed by a 30° visuomotor rotation to the cursor representing the subject's hand position. After learning to compensate for the rotation, we removed visual feedback for an extended set of trials and then reinstated feedback in the absence of any visual perturbation.

The data, represented in epochs of eight trials, are shown in Fig. 1*B* for four subjects. At first, each subject reached to the target accurately with some noise (*top*, gray traces). Upon introduction of the perturbation, subjects rapidly learned to counter the imposed rotation, learning ~80% of the total rotation within 30 epochs. Upon removing visual feedback, the adapted behavior decayed gradually toward baseline and then rapidly washed out during the last 30 epochs of the experiment when feedback was reinstated.

We assumed that experience of error engaged two learning processes that differed in their sensitivity to error as well as retention properties (*Eq. 10*). We fit this two-state model to the measured data from each subject to determine the properties of the hypothesized fast and slow states. To fit our model, we used two different algorithms, LMSE and EM. Each algorithm was provided with the noisy, single-subject, epoch-by-epoch data (Fig. 1*B*, *top*, gray traces). From this behavior, the algorithms estimated model parameters. We used these estimates to predict time courses of behavior (Fig. 1*B*, *top*, blue and red lines) and the underlying slow and fast states (Fig. 1*B*, *bottom*). The EM and LMSE time courses corresponded to the expected value of the behavior and hidden states under each algorithm's estimates of the model parameters (*Eq. 12*).

We found that the two algorithms provided similar fits to the subject data (Fig. 1*B*, *top*). However, differences between the algorithms emerged at the level of the fast and slow states of learning. For some subjects (*subject S9*; Fig. 1*B*, *bottom*), EM and LMSE agreed quite well. However, for other subjects, the predictions made by LMSE and EM differed considerably (*subjects S2* and *S1*; Fig. 1*B*, *bottom*). In these cases, the LMSE predictions often appeared to be unreasonable. For example, in *subject S2*, LMSE predicted large initial biases in the hidden states (*subject S2*; Fig. 1*B*, *bottom*). These large initial states were accompanied by a near-zero error sensitivity in the slow state of learning. This near-zero error sensitivity allowed LMSE to use the slow state of learning as a bias, accounting for the dynamics of *subject S2*'s behavior with the fast state alone. LMSE predicted that five of the 20 subjects possessed such an error-insensitive slow state. In contrast, the EM algorithm did not yield any such aberrant estimates. In another example, for *subject S1*, LMSE estimated a slow-state retention factor that was >1 (*subject S1*; Fig. 1*B*, *bottom*). This led to a monotonically increasing slow state of learning (solid red lines) that never decayed. LMSE estimated that eight of the 20 subjects possessed an unstable slow-state retention factor. Again, the EM algorithm did not yield any such results.

Differences between EM and LMSE were further highlighted when the fits were averaged across subjects. That is, whereas the average fits to the behavior were nearly identical for EM and LMSE (Fig. 2*A*, *top*), the fast- and slow-state predictions differed considerably between the two algorithms. LMSE predicted larger contributions from the fast state than EM and smaller contributions from the slow state. Relative to LMSE, EM predicted a smaller slow-state retention factor [paired *t*-test, $t(19) = 3.6$, $P < 0.01$], a larger slow-state error sensitivity [paired *t*-test, $t(19) = 3.4$, $P < 0.01$], a smaller fast-state error sensitivity [paired *t*-test, $t(19) = 2.6$, $P < 0.05$], and a smaller fast-state retention factor [paired *t*-test, $t(19) = 2.7$, $P < 0.05$]. That is, the two algorithms made significantly different predictions regarding parameters of the system.

The different EM and LMSE model parameters led to contrasting levels of variance in the hidden state time courses; the LMSE time courses had considerably higher variability across subjects (compare sizes of error bars; Fig. 2*A*, *bottom*). This elevated variability was largely driven by the outlying, likely errant, single-subject fits we previously noted. The outlying fits also contributed to greater variance in the model parameters; the across-subject standard deviation of the LMSE estimates exceeded that of EM for five of the six model parameters (all except for the fast-state retention; Table 3).

These differences between the two algorithms produced a dilemma. Depending on the choice of algorithm, we obtained different descriptions of the fast and slow states of learning. Which estimate was closer to the truth?

To answer this question, we computed AICc to determine the evidence for the likelihood models maximized by EM and LMSE. The critical difference between EM and LMSE is that the likelihood function maximized by EM contains noise in both the state update process and generation of a movement, whereas the LMSE likelihood function considers noise only in the measurement of behavior, not in the underlying learning or moving processes (see APPENDIX 2: OVERVIEW OF THE LMSE ALGORITHM).

We computed the differences in AICc across likelihood models for each subject, resulting in a within-subject comparison of EM vs. LMSE (Fig. 2*C*). We found that the AICc for the model including state and motor noise (EM) was lower (better) than that of the model excluding state and motor noise (LMSE): paired *t*-test across subjects, $t(19) = 3.4$, $P < 0.01$. This suggested that despite having an additional parameter, the learning process is better described by the stochastic system considered by EM rather than the deterministic one assumed by LMSE.

In summary, when we fit experimental data with each algorithm, we found that EM's estimates of the hidden states diverged from LMSE. Unlike LMSE, EM did not exhibit any aberrant fast- or slow-state predictions. EM also yielded states that were more consistent across subjects; they had smaller variances in the model parameters, leading to lower variance in the hidden-state time courses across subjects. LMSE uses a likelihood function that is blind to state and motor noise. In contrast, EM's likelihood function takes these noise sources into account. The likelihood model maximized by EM possessed a lower AICc than LMSE. This implied that a stochastic model of learning and moving was a better descriptor of the experimental data. Therefore,

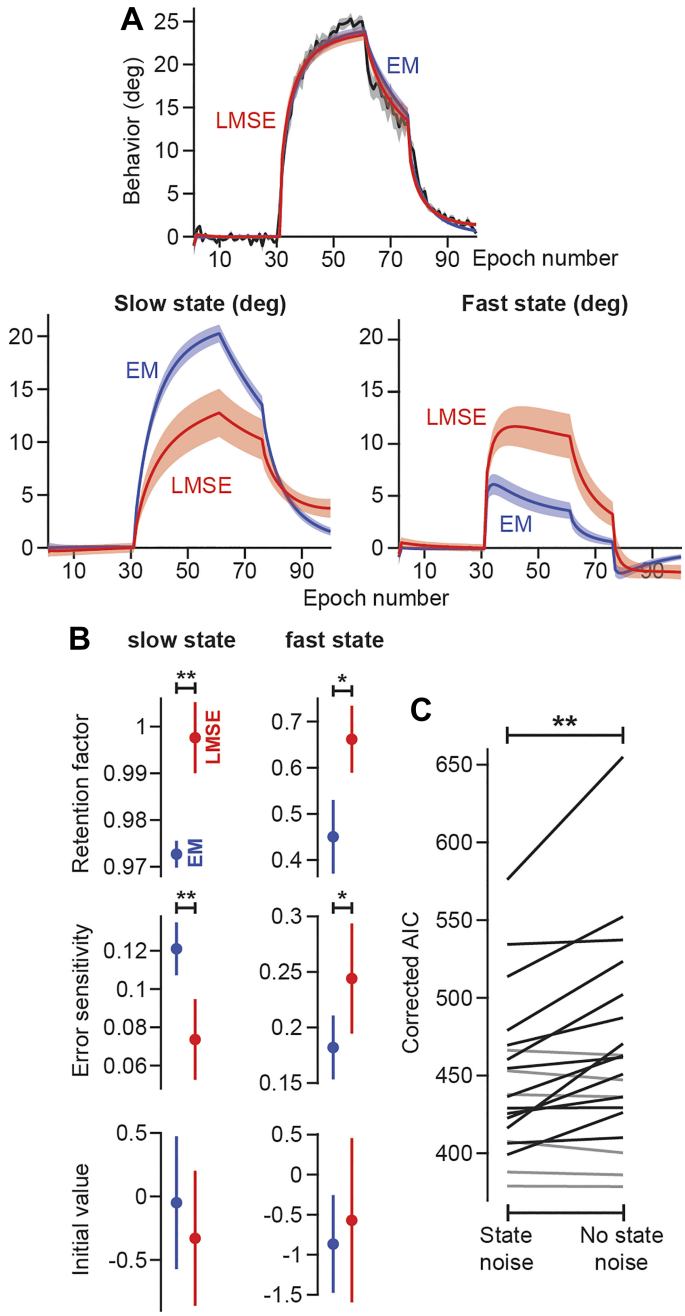

Fig. 2. Comparison of parameter values uncovered by EM and LMSE as fitted to experimental data. *A*: population behavior, represented by the average time course across all 20 subjects. *Top*: the average behavior (black) is shown overlaid with the average EM (blue) and average LMSE (red) fits. EM and LMSE had very similar fits to the behavior. However, the algorithms' predictions regarding the slow and fast states diverged. Error bars indicate ± 1 SE. *B*: model parameters. Bars indicate the mean value across the subjects. Error bars indicate ± 1 SE. *C*: we compared the corrected Akaike Information Criterion (AIC) of 2 competing likelihood models: one with state and motor noise, and one without state and motor noise. AICc was lower for a model with state and motor noise for 14 of the 20 subjects (black lines) and larger for 6 of the 20 subjects (gray lines). A paired *t*-test across subjects indicated that a model with motor and state noise possessed a lower AICc (that is, a better fit) than a model without these noise sources.

learning from error is likely a stochastic process, not a deterministic one.

Fitting the two-state model to simulated data. To better appreciate how well each algorithm could recover the hidden states when learning and moving are noisy processes, we performed simulations (*Eq. 10*) to produce realistic data sets using parameters obtained from our experiment. Unlike the behavior collected in the laboratory, the hidden states were known in these simulated data sets, providing the opportunity to assess how well EM and LMSE could uncover the hidden states. Four paradigms were considered: abrupt perturbations that included or excluded error clamp trials (*paradigms 1* and *2*; Fig. 3*A*), a paradigm with a gradual perturbation (*paradigm 3*; Fig. 3*A*), and a paradigm that demonstrated spontaneous recovery and decay of motor memory due to set breaks (*paradigm 4*; Fig. 3*A*). These simulated paradigms included only a single target, and the resulting data were analyzed trial by trial.

Figure 3*B* shows the noise-free time courses of the simulated behavior (generated via *Eq. 12*) and the corresponding fast and slow states. Presence of error produced rapid changes in the fast state but more gradual changes in the slow state when the perturbation was introduced abruptly (*paradigm 1*). Complete loss of the fast state occurred after set breaks (*paradigm 4*). Gradual introduction of the perturbation slowed the onset of the fast state of learning (*paradigm 3*). Error clamp trials produced rapid decay of the fast state but produced only small changes in the slow state (*paradigm 2*). Occasionally, the two states had opposite signs (washout trials; *paradigm 2*), which led to spontaneous recovery in error clamp trials (*paradigm 4*).

Figure 3, *C–E*, shows typical time courses of the simulated behavior and hidden states in the presence of noise (*Eq. 10*). Similar to our human data, we found that both EM and LMSE provided excellent fits to the simulated data (Fig. 3*C*). However, EM appeared to be more robust in uncovering the fast and slow states. In some conditions (*paradigms 1–3*; Fig. 3, *D* and *E*), LMSE produced estimates of the hidden states that diverged considerably from the truth. To compare each algorithm's accuracy in recovering the hidden states of learning, for each paradigm (1,000 simulations for each of the four paradigms), we computed the root mean squared error (RMSE) between three pairs of values, the true values of the simulated behavior, fast state, and slow state [y, x_f, x_s], and the predicted time courses obtained from EM and LMSE. We observed that, across paradigms, LMSE fit the simulated behavior better than EM [parameter y, paired t-test, $t(999) > 13.9$, $P < 10^{-5}$; Fig. 4, *top*). This was expected, as the objective function of LMSE minimizes the RMSE between the observed behavior and model predicted behavior (see APPENDIX 2: OVERVIEW OF THE LMSE ALGORITHM). With that said, LMSE offered only a modestly improved fit over EM [~10% over EM, parameter y, 1-sample t-test, $t(999) > 31.5$, $P < 10^{-5}$; Fig. 4, *bottom*].

Whereas LMSE more closely tracked the observed data, EM produced far more accurate estimates of the fast and slow states (x_f, x_s; Fig. 4). In other words, LMSE was more prone to predicting errant fast- and slow-state time courses (Fig. 3, *D* and *E*). LMSE performed particularly poorly in *paradigm 3*, producing fast- and slow-state estimates that had errors exceeding those of EM by >275% [*paradigm 3*, 1-sample t-test, $t(999) > 25.0$, $P < 10^{-5}$; Fig. 4, *bottom*]. LMSE also performed poorly in *paradigm 1*, where it produced state estimates that had errors exceeding those of EM by ~75–125% [*paradigm 1*, 1-sample t-test, $t(999) > 14.1$, $P < 10^{-5}$; Fig. 4, *bottom*). Including error clamp trials in *paradigm 2* improved LMSE performance marginally (Fig. 4). Finally, LMSE (and EM) performed best in *paradigm 4*, the paradigm that included both error clamp trials and set breaks (Fig. 4). In this case, EM

Table 3. *Parameter standard deviation*

Paradigm and Algorithm	Two-State Model Parameters						
	a_s	a_f	b_s	b_f	$x_s^{(1)}$	$x_f^{(1)}$	d
Paradigm 1 EM	0.0175	0.1878	0.0355	0.0669	1.5149	1.5345	
Paradigm 1 LMSE	0.0330	0.2679	0.0472	0.0780	5.9684	6.5494	
Paradigm 2 EM	0.0094	0.1667	0.0253	0.0549	1.5782	1.6078	
Paradigm 2 LMSE	0.0225	0.2513	0.0409	0.0715	6.6765	7.4575	
Paradigm 3 EM	0.0130	0.1966	0.0393	0.1042	1.5207	1.5892	
Paradigm 3 LMSE	0.0368	0.2965	0.0601	0.2902	12.1512	12.6409	
Paradigm 4 EM	0.0099	0.1057	0.0241	0.0300	1.3639	1.3469	6.2715
Paradigm 4 LMSE	0.0132	0.1521	0.0297	0.0354	3.8563	4.4528	10.0969
Human subjects EM	0.0127	0.3559	0.0616	0.1282	2.3391	2.7227	
Human subjects LMSE	0.0339	0.3236	0.0942	0.2212	2.3771	4.5877	

Here, we report the standard deviation of the two-state model parameter distributions shown in Fig. 5. For *paradigm 4*, we also provide the standard deviation for the set break decay factor, although this is not shown in Fig. 5. Note that for all parameters in our simulated paradigms, the standard deviation of the LMSE distribution exceeded that of EM in simulation. On the bottom 2 rows, we provide the standard deviation for each two-state model parameter across the 20 subjects that participated in our visuomotor rotation task.

was better than LMSE for the fast state [*paradigm 4*, 1-sample *t*-test, $t(999) = 11.8$, $P < 10^{-5}$; Fig. 4, *bottom*], with no difference in estimation of the slow state [*paradigm 4*, 1-sample *t*-test, $t(999) = 0.9$, $P = 0.39$; Fig. 4, *bottom*].

Overall, we found that LMSE performed substantially worse than EM in uncovering the hidden states. Why did LMSE fit the observed data well but was unable to robustly uncover the hidden states? We observed three modes of failure by LMSE.

In the first mode, LMSE identified retention factors for the slow state that were >1 (*paradigm 1*; Fig. 3*D*). Such retention factors resulted in unstable behavior of the slow state. Relative to EM, LMSE was significantly more prone to identifying unstable slow-state retention factors. For *paradigms 1* and *3*, LMSE identified unstable slow-state retention factors in 44.8 and 54% of the runs, respectively, whereas for EM this occurred in 15.2 and 6.4% of simulations. The inclusion of error clamp trials in paradigm 2 protected against this mode of failure, lowering the number of runs affected by unstable retention to 19.6% for LMSE and 1.3% for EM. Similarly, *paradigm 4* resulted in a very low frequency of error: 5.2 and 0.7% of simulations for LMSE and EM, respectively. This mode of failure was also demonstrated by LMSE fits to eight of the 20 subjects in our experiment (*subject S1*; Fig. 1*B*, *bottom*).

In the second mode of failure, LMSE converged on a slow state of learning that was insensitive to error and possessed near complete retention, causing it to function as a behavioral bias (*paradigm 2*; Fig. 3*D*). In this case, LMSE accounted for the dynamics of the observed data by relying solely on the fast state. These error-insensitive slow states of learning were also observed in LMSE fits to five of the 20 subjects in our experiment (*subject S2*; Fig. 1*B*, *bottom*).

In the final failure mode, LMSE predicted very large initial slow and fast states with similar magnitude and opposite signs (*paradigm 3*; Fig. 3, *D* and *E*). For *paradigms 1–4*, LMSE identified hidden fast or slow states that differed from the true value (threshold of 10° or greater absolute error) in 8.1, 12.2, 25.8, and 2.5% of simulations, respectively. For EM, this never occurred. This mode of failure appeared more prevalent in simulation than in our behavioral data, with EM and LMSE both predicting only one subject with an initial state that exceeded 10°, or one-third of the eventual perturbation.

Unsurprisingly, these three modes of failure were accompanied by larger error in the LMSE estimates of the model parameters. To quantify each algorithm's error in estimating model parameters, we computed the absolute error for each parameter across all paradigms (Fig. 5). We found that for all parameters and in all paradigms, LMSE possessed a greater absolute error than EM [bars labeled *P1–P4*, paired *t*-test, $t(999) > 6.0$, $P < 10^{-5}$; Fig. 5]. EM also better estimated the set break decay parameter in *paradigm 4* than LMSE [errors of 4.99 ± 0.25 and 7.67 ± 0.43 for EM and LMSE, respectively, paired *t*-test, $t(999) > 13.2$, $P < 10^{-5}$]. The inclusion of error clamp trials and set breaks improved the performance of both algorithms; with the exception of the initial fast and slow states, both algorithms had the lowest error in *paradigms 2* and *4*. Although error clamp trials were also present in *paradigm 3*, the gradual introduction of the perturbation hampered the response of the fast state of learning (*paradigm 3*; Fig. 3), which likely impaired the ability of each algorithm (LMSE more than EM) to differentiate properties of the two learning processes. The opposite was true for *paradigm 4*, where the two set breaks, opposite perturbations, and error clamp trials resulted in several "excitations" of the fast state (*paradigm 4*; Fig. 3*B*). These excitations significantly improved the ability to identify properties of the fast state, as evidenced by the markedly reduced error in fast state retention and error sensitivity (*paradigm 4*; Fig. 5) for both algorithms. Error clamp trials appeared to have a similarly dramatic effect on estimation of slow-state retention and error sensitivity (compare *paradigm 1* with *paradigm 2*; Fig. 5).

In summary, in all paradigms, EM was more accurate in uncovering the true parameters of the learning processes. Differences in hidden-state recovery were driven by three failure modes in the LMSE algorithm, two of which were also prominent in LMSE fits to our human behavior data set. Inclusion of set breaks and error clamp trials significantly improved the ability of both EM and LMSE to uncover the two-state model parameters.

A better tool for hypothesis testing. Sensorimotor tasks are occasionally designed to test the effectiveness of an intervention. Model fits provide a tool to ask whether the intervention significantly affected learning parameters such as error sensitivity or retention. Power analysis provides an estimate of how many subjects may be needed to detect a

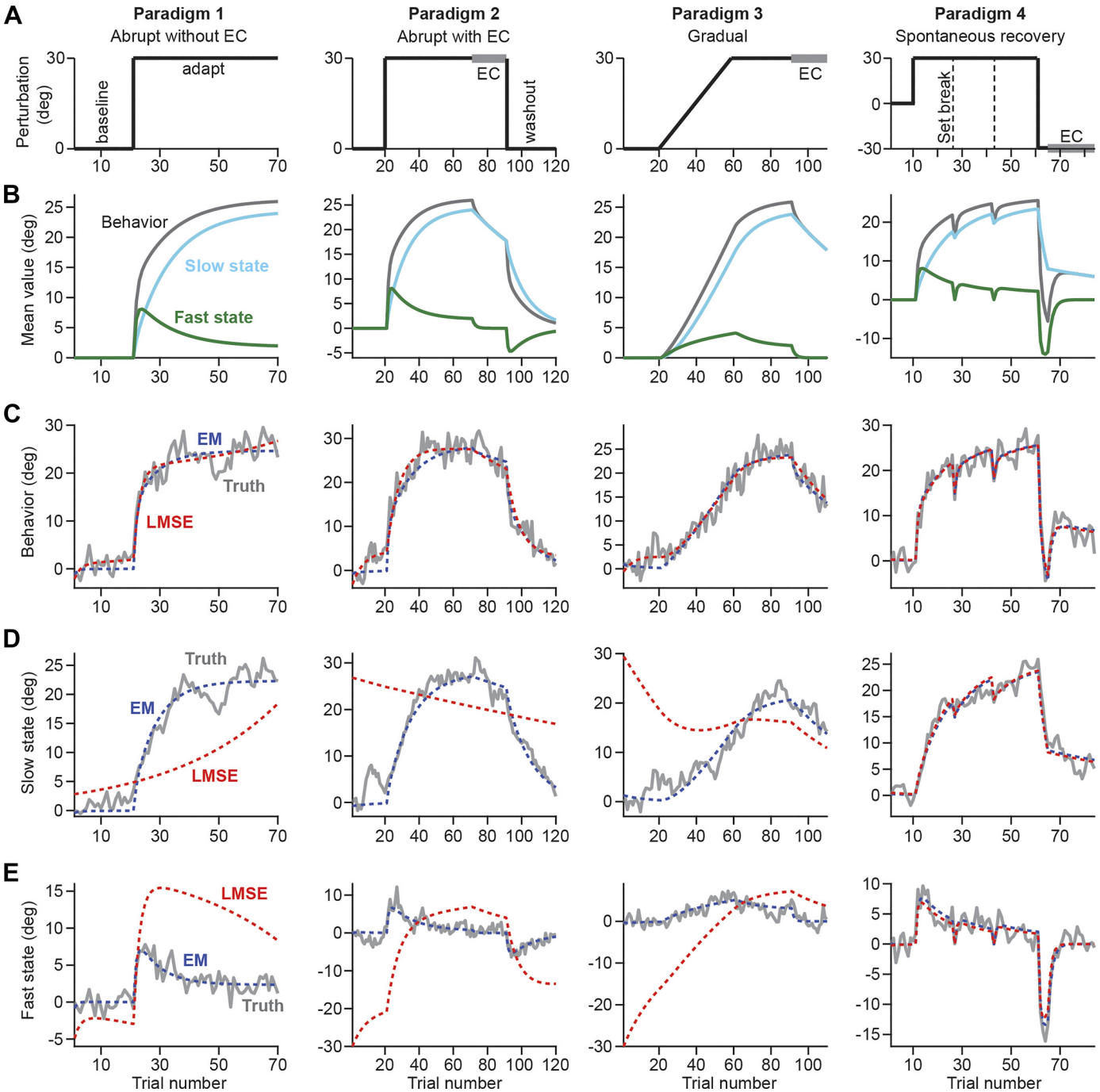

Fig. 3. Simulated paradigms and behavior. *A*: we simulated two-state models of learning in the context of 4 behavioral paradigms with abrupt and gradual introduction of perturbations and with error clamp (EC) trials and set breaks. *B*: the expected value of the measured behavior (black), fast state (green), and slow state (blue) of learning. These time courses correspond to two-state model parameters extracted from our subject population (Table 1). *C*: for each of the 4 paradigms, behavior was simulated according to a two-state model of learning (see *Eq. 10*); 1,000 simulations were performed for each paradigm. The two-state model parameters were fixed for each simulation, solely the seed for the random number generator varied from simulation to simulation. Here, we provide an example of a behavioral trajectory for each of the 4 paradigms. We fit each trajectory using EM and LMSE. *D*: the true slow state of learning along with EM and LMSE predictions. In the example simulations of *paradigms 1–3*, LMSE failed to capture the slow state of learning. In *paradigm 4*, both LMSE and EM closely tracked the true slow state. *E*: the true fast state of learning along with EM and LMSE predictions. For *paradigms 1–3*, LMSE predictions diverged from the true fast state trajectory. For *paradigm 4*, both EM and LMSE tracked the true fast-state time course.

significant difference. The number of subjects needed to test a hypothesis depends on the noise properties of the data. In our experiment, EM parameter estimates had a lower variance than LMSE (evident by visual inspection of Fig. 2*B*; see Table 3 for numerical details). This lower variability in the parameter estimates has a practical implication; it should improve the ability to test hypotheses.

To explore this question, we considered a within-subject and a between-subject experimental design. A within-subject experiment is typical for the study of savings and

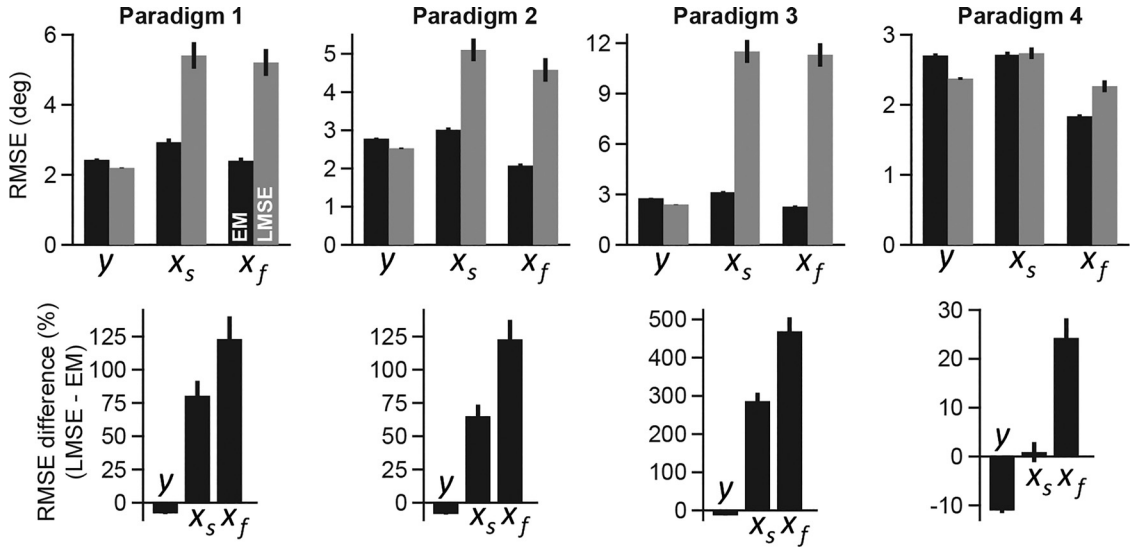

Fig. 4. Performance of EM and LMSE algorithms. For each paradigm, 1,000 simulations were performed with fixed two-state model parameters but a varying seed for the random number generator, altering noise. EM and LMSE were used to fit a two-state model to the simulated behavior. The EM and LMSE parameters were used to simulate noise-free time courses for behavior, slow state of learning, and fast state of learning. Next, we computed the root mean squared errors (RMSEs) describing how well EM and LMSE recovered the hidden fast and slow states of learning and the overall behavior. *Top*: the RMSE for the behavioral fit (y), slow-state fit (x_s), and fast-state fit (x_f) are shown. *Bottom*: a relative RMSE metric was computed to compare the RMSEs of EM and LMSE fits to the same simulated behavior; the RMSE for the LMSE algorithm was divided by that of EM and multiplied by 100, and then a factor of 100 was subtracted to compute a percent increase of LMSE RMSE over that of EM. All bars represent the mean RMSE across 1,000 simulations. Error bars represent 95% confidence intervals. LMSE improved upon the RMSE of the behavior fit by ~10% for all paradigms. However, EM was superior in uncovering the slow and fast states. The largest difference was observed for *paradigm 3*, followed by *paradigm 1*, then *paradigms 2* and *4*.

anterograde interference, where the same subjects are exposed to similar or contrasting perturbations at two different time points. Between-subject designs are common when testing the effects of some intervention against a control group, where the subjects in both conditions differ. Here, we imagined that these interventions may cause changes in learning rate, as is observed in savings paradigms (Herzfeld et al. 2014b), and retention, as is observed in differing reward environments (Galea et al. 2015).

We analyzed how many subjects would be required to achieve a particular level of confidence in the ability to detect differences in the learning parameter modified by the intervention. To do this, we created experiments by sampling subjects from a large pool. For our within-subject experiment, we sampled the same subjects in two different environments. For our between-subject experiment, we sampled subjects independently in the two different environments. We performed *t*-tests to see whether there was a difference in the EM and LMSE learning parameters across the two simulated environments. We used a paired *t*-test to test for within-subject changes and a two-sample *t*-test to test for between-subject changes. These simulated environments possessed the same trial structure (*paradigm 2*) but differed in that subjects were simulated with different sets of error sensitivities and retention factors. We analyzed changes in each error sensitivity and retention factor separately; i.e., in each of our simulated experiments, we varied the distribution of a single learning parameter at a time. We varied the number of subjects and determined the percentage of simulated experiments for which a statistically significant ($P < 0.05$) difference existed in the appropriate model parameter. Finally, we determined the number of subjects that would be required to detect a significant change in parameter value for ≥85% of the experiment repetitions.

Figure 6, *A* and *B*, provides the results for our within-subject and between-subject analyses, respectively. Unsurprisingly, we found that a within-subject comparison required fewer subjects than a between-subject comparison for both EM and LMSE. A within-subject test is more powerful, as it accounts for be-

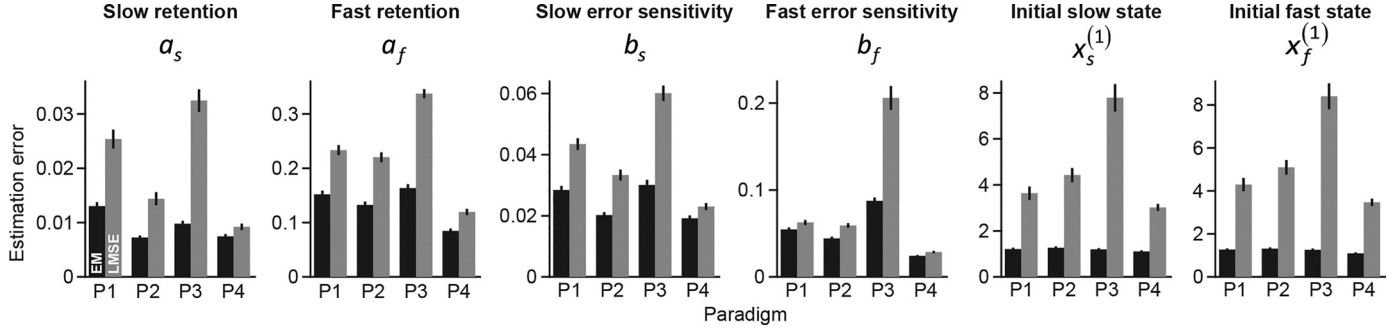

Fig. 5. Parameter estimation errors. For each simulation, we computed the absolute value of the difference between each true parameter and the parameter values predicted by EM and LMSE. All bars represent the mean absolute parameter error across all simulations. Error bars represent 95% confidence intervals. For all parameters and paradigms, EM had lower estimation error than LMSE. P1–P4, paradigms 1–4, respectively.

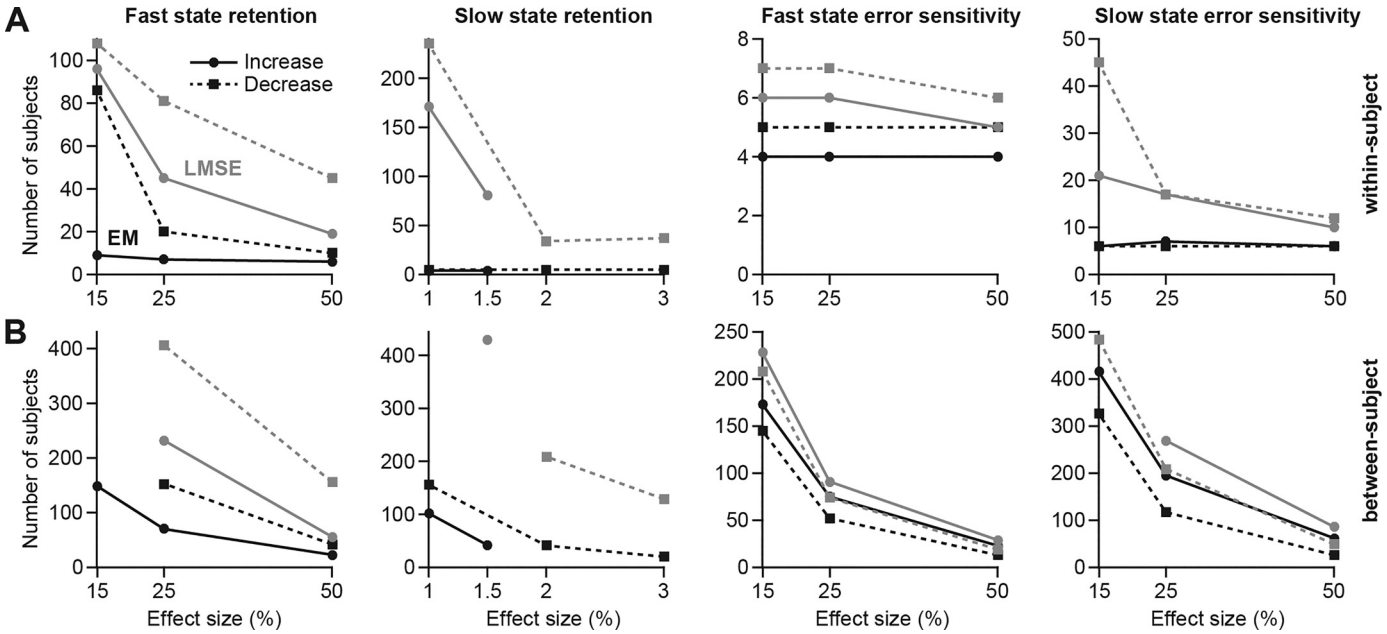

Fig. 6. Power analysis. We simulated within-subject and between-subject experiments to determine the number of subjects that would be required to detect a change in learning parameters. We created a pool of 1000 simulated subjects by sampling two-state model parameters from a multivariate normal distribution. We created different distributions by scaling a single learning parameter for each of the 1,000 subjects. We simulated behavior in *paradigm 2* and fit the data with EM and LMSE. We then sampled subjects to perform hypothesis testing. For within-subject tests, we sampled the same subjects from different parameter levels. For between-subject tests, we sampled subjects independently from different parameter levels. For each test, we performed a paired *t*-test (within-subject analysis) or two-sample *t*-test (between-subject analysis) to determine whether EM or LMSE detected a statistically significant difference in the learning parameter. We repeated this process for different random samples of our subject population (10,000 for each test). Finally, we determined the minimum number of subjects that would be required for each algorithm to detect a significant difference for 85% of our samples. Here, we show the number of subjects required to reach an 85% detection rate for both EM (black) and LMSE (gray) as a function of the magnitude of the true parameter difference for each test (the effect size). We performed tests for both increases (solid lines with filled circles) and decreases (dashed lines with filled squares) in two-state model parameters. The results for the within-subject analysis and between-subject analysis are shown in *A* and *B*, respectively. We report only results for which <500 subjects were required to reach the 85% detection rate. For LMSE, >500 subjects were required for 5 different parameter-effect size pairs in the between-subject analysis. For EM, this occurred only once.

tween-subject variability, increasing the power of the statistical comparison. For both the within-subject test and between-subject test, we found that the number of subjects required to achieve an 85% detection rate generally decreased as the effect size (the magnitude of the parameter difference across the 2 environments) increased. This applied to both increases (solid lines) and decreases (dashed lines) in each parameter. For all effect sizes and parameters, EM required fewer subjects to reach the threshold detection level. Specifically, EM required ~20–95 and 30–95% fewer subjects than LMSE for the within-subject comparison and across-subject comparison, respectively.

In summary, we found that, as compared with LMSE, EM reduced the number of subjects that would be needed to detect the effects of an intervention that altered error sensitivity or retention.

Modeling higher levels of noise. Subjects performing sensorimotor learning tasks exhibit different levels of noise in their motor behavior. Does the performance of either EM or LMSE worsen in the presence of higher levels of noise? To answer this question, we performed a sensitivity analysis by scaling the state and motor noise variances to different levels (0.5, 2, 4, 6, 8, and 10 times the values reported in Table 1). At each level, we simulated the performance of 1,000 subjects in *paradigm 2*. We fit the simulated behavior with EM and LMSE and computed the RMSEs for each algorithm's estimates of behavior and hidden states of learning.

As expected, LMSE always provided a closer fit to the behavioral data (Fig. 7*A*). This result was expected, because the objective function of LMSE minimizes the RMSE of the observed behavior. However, for all levels of noise, EM better isolated the hidden states of learning (Fig. 7, *B* and *C*). The differences in the hidden-state RMSE were approximately an order of magnitude larger than the difference in RMSE for the observed behavior. Therefore, irrespective of the noise level, it appeared that EM traded off small errors in fitting the observed behavior for larger improvements in uncovering the hidden fast and slow states. In summary, we expect that our conclusions about the relative performance of EM and LMSE would hold even at higher levels of state and motor noise.

Different noises in the fast and slow adaptive processes. We assumed that the fast and slow learning processes were affected by state noises with equal magnitude. However, faster processes that change more rapidly may also be accompanied by higher levels of noise. In fact, a relationship between learning rate and state variance would be expected from a Bayesian interpretation of learning (Kording et al. 2007). In light of these considerations, we reanalyzed our subject behavior with a two-state model of learning with two separate variances for fast and slow states of learning. In line with our intuition, we found that the variance of the fast state (mean ± SE, $2.74 \pm 0.58^{\circ 2}$) was greater [paired *t*-test, $t(19) = 4.39$, $P < 0.001$] than that of the slow state (mean ± SE, $0.63 \pm 0.34^{\circ 2}$).

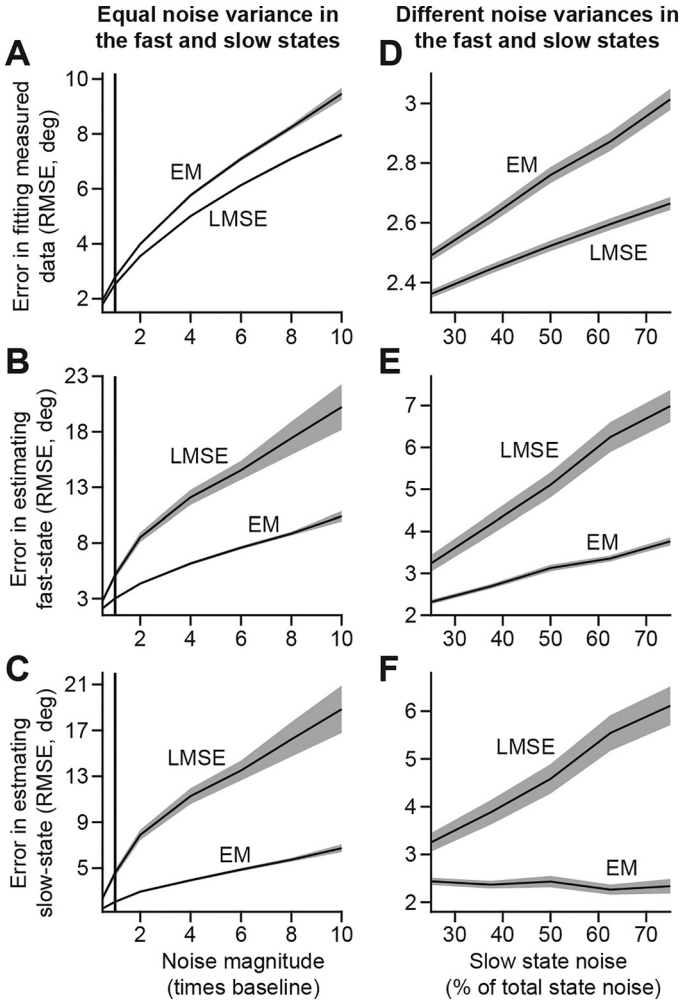

Fig. 7. Sensitivity analysis for state and motor noise. *A–C*: we scaled the state and motor noise variances by 0.5, 2, 4, 6, 8, and 10 times the values reported in Table 1 (measured from our subject population). At each noise level, we performed 1,000 simulations of *paradigm 2*. We fit the simulated reaching behavior with EM and LMSE, generated EM and LMSE estimates of the behavior, fast, and slow states of learning, and finally computed the RMSE between the true time courses and model fits. The RMSEs for the behavior, slow, and fast state are shown in *A*, *B*, and *C*, respectively. Solid lines indicate the mean RMSE across all 1,000 simulations at each noise level. The shaded error bars indicate 95% confidence intervals around the mean. *D–F*: we performed another analysis where we allowed the slow and fast processes to have different variances. We fixed the overall level of state noise (sum of the fast- and slow-state variances) and performed a sensitivity analysis where we assigned different fractions of the overall state noise differentially to the slow and fast states. We tested levels where the slow (or fast) state had 25, 37.5, 50, 62.5, and 75% of the overall variance. For each level, we simulated 1,000 simulations of *paradigm 2*. We fit the simulated behavior using EM and LMSE and computed RMSEs for the behavior, slow process, and fast process, as in *A*, *B*, and *C*, respectively. The RMSEs for the behavior, slow, and fast state are shown in *D*, *E*, and *F*, respectively. Solid lines indicate the mean RMSE across all 1,000 simulations at each noise level. The shaded error bars indicate 95% confidence intervals around the mean.

We next asked whether a model with separate variances for the slow and fast states was more likely to explain the data than a model with only one variance. To ask this question, we computed the AICc for each subject for these two models. A within-subject paired *t*-test across models did not reveal a significant difference in the AICc for a model with a single state noise or separate state noises for the slow and fast state [paired *t*-test, $t(19) = 0.39$, $P = 0.703$]. This result indicates that in our experimental data, a model with two separate variances for the fast and slow states was not justified.

Despite this, we performed a sensitivity analysis to determine how well EM and LMSE would identify the fast and slow states of learning if each process was affected by different levels of state noise. In this analysis, we fixed the cumulative level of slow- and fast-state variance (i.e., we fixed the sum of the slow- and fast-state variances) and changed the fraction of the total state noise variance attributed to either the fast and slow state. We analyzed levels of 25, 37.5, 50 (i.e., equal contribution from the fast and slow state), 62.5, and 75%. At each level, we simulated the behavior of 1,000 subjects performing paradigm 2. We fit the simulated behavior with EM and LMSE and computed the RMSEs for each algorithm's estimates of behavior and hidden states of learning, as in Fig. 4. Regardless of the noise level tested, EM had greater error in the fitting of the measured behavior (Fig. 7*D*), but it remained superior to LMSE in the identification of the slow and fast states (Fig. 7, *E* and *F*). Therefore, we expect that the superior performance of EM would generalize to systems with different levels of variance in the fast and slow states.

Other sources of noise. We assumed that noise from all processes that contribute to learning from error could be combined into a cumulative state noise by adding the variances of each noise source together. However, there exist sources of noise that violate this assumption. Consider, for example, noise involved in the learner's observation of error. To correctly incorporate this source of noise, we could modify *Eq. 6* by adding an observation noise that adds to the true error. In this model, the observation noise would be multiplied by error sensitivity, and therefore the state update noise in *Eq. 6* would also depend on error sensitivity. We considered this more complete model by simulating *Eq. 6* with an added observation noise term. We performed 1,000 simulations of this model for *paradigm 2* and attempted to recover the variances of the three different noise sources using EM. Unfortunately, we found that our algorithm was unable to estimate the variance of observation noise. For ~80% of simulations, EM's estimate of the observation noise variance converged to either an upper or lower bound of the parameter space, far from the true value we used in simulation. That is, our algorithm had no power to estimate this variance separately from the independent state and motor noises.

This limitation was caused by the multiplication of observation noise by error sensitivity. Because of this multiplication, the variance contributed by the observation noise is multiplied by the square of error sensitivity, and was therefore exceedingly small relative to the independent state noise variance. Therefore, while a more accurate model of learning might include this observation noise, we currently cannot estimate its magnitude using our algorithm.

Changing the dynamics of the fast and slow adaptive processes. Motor learning in different effectors (e.g., eye, arm, etc.) and paradigms (e.g., force field adaptation, visuomotor adaptation, etc.) occurs at different rates. Would the accuracy of EM or LMSE estimates of the hidden processes vary with the underlying learning and forgetting rates of the subject population? To investigate this question, we performed sensitivity analyses for four parameters: slow retention, fast reten-

tion, slow error sensitivity, and fast error sensitivity. We varied each of these parameters in turn across the ranges used for our power analysis. For each parameter level, we performed 1,000 simulations of our two-state model (*Eq. 10*) for *paradigm 2*. We fit the simulated behavior with EM and LMSE and computed the RMSEs for each algorithm's estimates of behavior and hidden states of learning, as in Fig. 4. As expected, LMSE continued to better fit the observed data in all simulations (Fig. 8, *top*). However, in every case, EM was more accurate in uncovering the fast and slow states (Fig. 8, *middle* and *bottom*). This difference in hidden-state RMSE was an order of magnitude larger than that of the observed behavior RMSE. Therefore, we expect that our conclusions about EM and LMSE would generalize to environments where learning exhibited dynamics different from those explored within our primary analysis.

Changing the bounds on the parameter space. Both algorithms were constrained to search an identical parameter space (Table 2). Despite this, we found that LMSE frequently exhibited modes of failure in the identification of two-state model parameters. One of these modes of failure involved the identification of slow-state retention factors whose magnitude exceeded 1. Retention factors in this range can lead to unstable behavior of the slow state of learning. We doubt that a biological system would exhibit this unstable behavior. Could the LMSE algorithm be rescued by modifying the parameter space to prevent the identification of these unstable retention factors? To answer this question, we reanalyzed our primary simulations for *paradigms 1–4* (Figs. 3, 4, and 5) by refitting the EM and LMSE algorithm in a parameter space whose upper bound for the slow- and fast-state retention factor was equal to 1. We fit each simulated behavior with EM and LMSE and

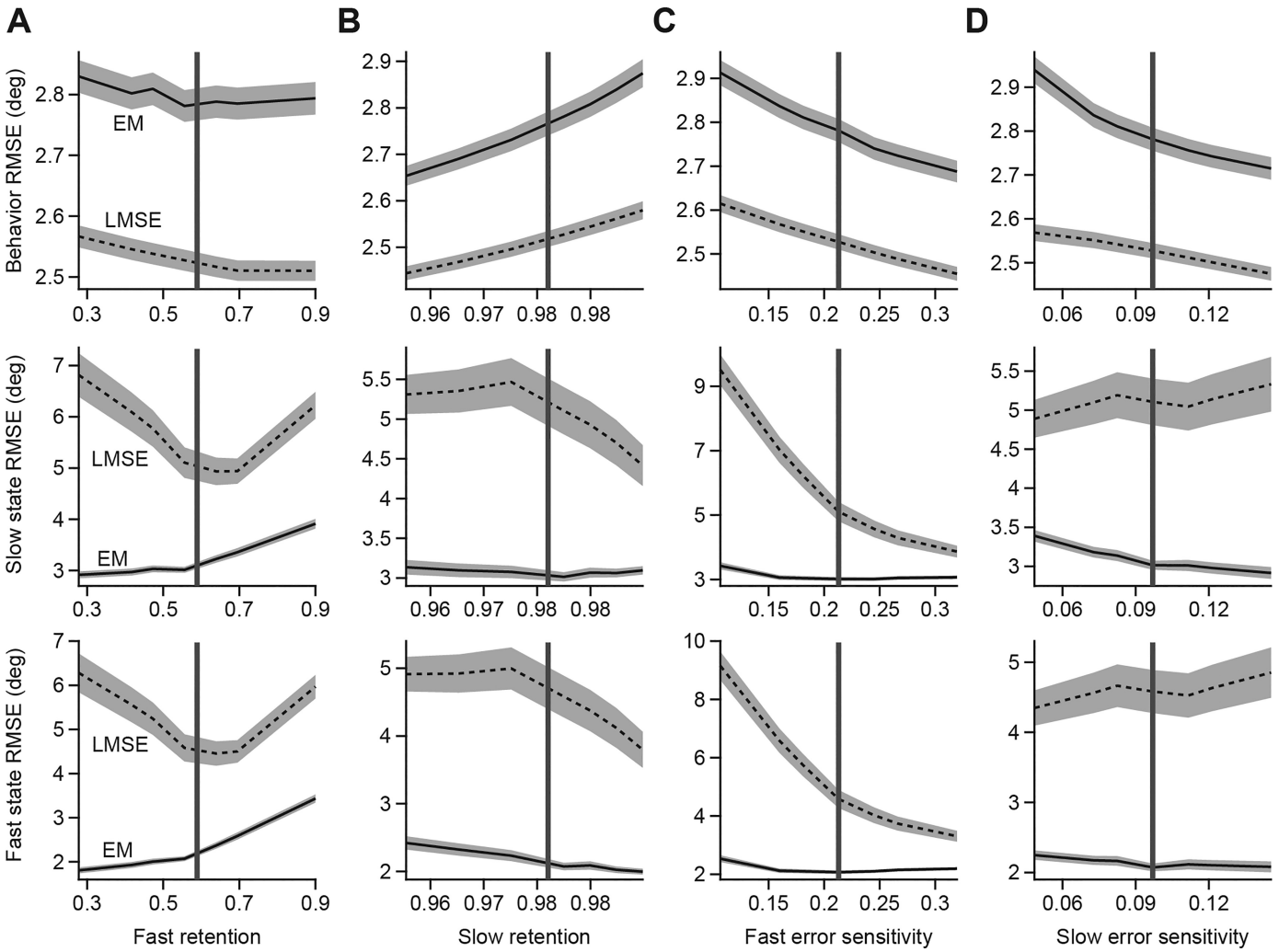

Fig. 8. Sensitivity analysis for the dynamics of the fast and slow states. We performed sensitivity analyses to determine how well EM and LMSE could isolate the fast and slow states of learning for two-state model parameters that differed from those observed for our visuomotor rotation subject population (Table 1). We analyzed one parameter at a time, fixing the remaining two-state model parameters to the values reported in Table 1. For each analysis, we scaled the two-state model parameter to several different values, corresponding to the effect sizes used in our power analysis in Fig. 6. At each parameter level, we performed 1,000 simulations of p*aradigm 2*. We fit the simulated reaching behavior with EM and LMSE, generated EM and LMSE estimates of the behavior, fast, and slow states of learning, and finally computed the RMSE between the true time courses and model fits. The *top*, *middle*, and *bottom* rows, show the RMSE for the behavior, slow state, and fast state fits, respectively. The shaded error bars indicate 95% confidence intervals. The parameters investigated are as follows: the fast-state retention factor (*A*), the slow-state retention factor (*B*), the fast-state error sensitivity (*C*), and the slow-state error sensitivity (*D*). For each analysis, EM identified slow and fast states of learning with lower RMSE than LMSE. These results indicate that the relative difference between EM and LMSE performance would generalize to other dynamics of learning.

computed the RMSEs for each algorithm's estimates of behavior and hidden states of learning (Fig. 9).

We found that restricting the parameter space had very little effect on the performance of EM (compare RMSE for the EM algorithm in Figs. 4 and 9). This is to be expected, considering that EM did not identify many solutions with unstable retention factors for any of the four paradigms. In contrast, the performance of LMSE improved in terms of identifying the fast and slow states, specifically for *paradigms 1*, *2*, and *3* (compare RMSE for the LMSE algorithm in Figs. 4 and 9). This was also expected, considering that LMSE predicted many behaviors with a slow-state retention factor that exceeded 1. However, despite the improvement in LMSE identification of the hidden processes in *paradigms 1–3*, LMSE error exceeded that of the EM algorithm (*paradigms 1*, *2*, and *3*; x_s, x_f; Fig. 9). Therefore, restrictions on the parameter space that eliminated unstable behavior of the slow state of learning improved but did not rescue the performance of LMSE relative to the EM algorithm.

Trial-by-trial analysis of behavior. Prior to fitting our state-space model to our behavioral data, we averaged single-subject behavior across eight-trial epochs, where each of the eight targets in our rotation task was visited once. We chose an epoch-based time scale for the analysis of subject behavior to minimize the effects of generalization on trial-based learning (see *Epoch vs. trial-by-trial analysis of behavior*). However, an epoch-based time scale of motor behavior exhibits different dynamics from a trial-biased timescale; more is learned and forgotten in an epoch of eight trials, and the variance in trial-by-trial behavior differs from that of epoch-by-epoch behavior. Therefore, the retention factors, error sensitivities, and noise variances that describe single-subject data for our epoch-based analysis will differ from those of a trial-based analysis.

We performed a set of control analyses to confirm that these differences in the trial-by-trial data would not lead to changes in the identification accuracy of EM or LMSE. We reanalyzed our experimental data on a trial-by-trial basis with two different models of generalization. Because we did not measure the generalization function of each subject, we considered two extreme cases, one where subjects fully generalized learning from one target to all other targets (full-generalization model) and another where subjects had no generalization of learning from one target to other targets (no-generalization model). For the full-generalization model, we applied *Eq. 10* to our trial-by-trial subject behavior as if the same target was visited on each trial. For the no-generalization model, we extended the dimensionality of our state to include a fast and slow state for each target (a total of 16 states, 2 hidden states for each of the 8 targets). For a given trial, only the fast and slow states for that target experienced error-based learning consistent with the case of no generalization, but all states experienced trial-by-trial forgetting. The full details of this model are described in APPENDIX 3: MULTIPLE TARGET STATE-SPACE MODEL OF LEARNING.

The trial-by-trial analysis yielded strikingly similar results to our epoch-by-epoch analysis. As before (Fig. 2*A*), EM and LMSE provided similar fits of the measured data for both the full- and no-generalization models (Fig. 10*A*, *top* and *bottom*, respectively). As in the epoch-by-epoch predictions in Fig. 2*A*, for both trial-by-trial models, EM estimated larger contributions from the slow state, and LMSE estimated larger contributions from the fast state (Fig. 10*B*). The differences in these trajectories were driven by different estimates of the two-state model parameters (not shown in figure). As with our epoch-by-epoch model parameters (Fig. 2*B*), in our full-generaliza-

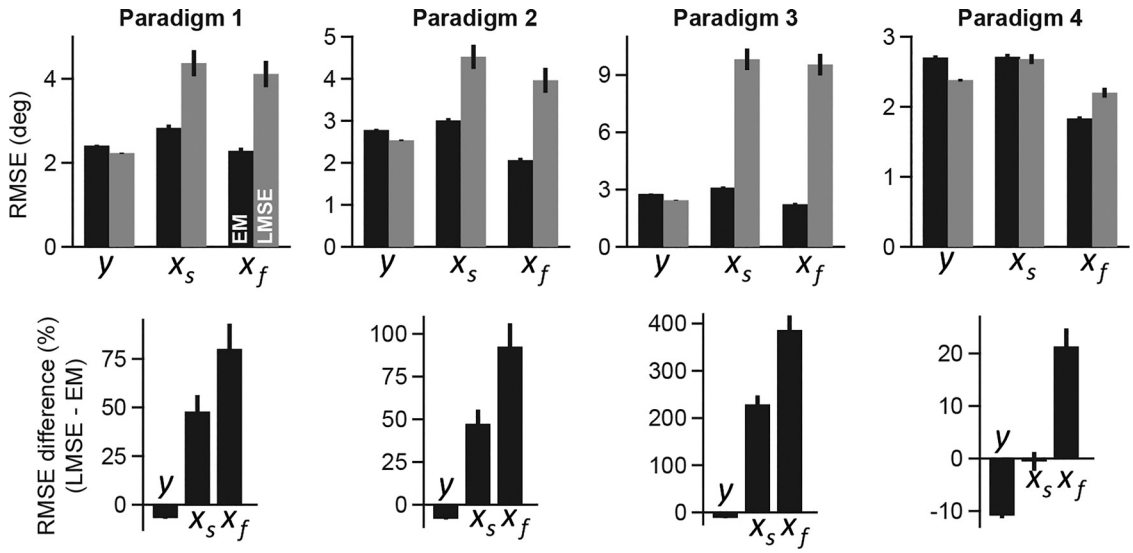

Fig. 9. Comparison of EM and LMSE in a restricted parameter space. In our primary analysis, we found a preference for LMSE to assign slow retention factors that exceed 1, which led to unstable behavior of the predicted slow process. We asked whether LMSE could be rescued by modifying the parameter search space to prevent the identification of these unstable retention factors. To answer this question, we reanalyzed our simulations for *paradigms 1–4* (see Figs. 3, 4, and 5) by refitting the EM and LMSE algorithm in a parameter space whose upper bounds for the slow and fast state retention factors were equal to 1. We used the EM and LMSE parameters to simulate noise-free time courses for behavior, slow state of learning, and fast state of learning. Next, we computed the RMSEs describing how well EM and LMSE recovered the hidden fast and slow states of learning and the overall behavior for the same simulations depicted in Fig. 4. *Top*, the RMSE for the behavioral fit (y), slow-state fit (x_s), and fast-state fit (x_f) are shown. *Bottom*: we computed the % difference between the RMSEs for EM and LMSE. Positive values indicate a larger RMSE for the LMSE algorithm. Error bars represent 95% confidence intervals. We found that restricting the upper bound on the slow- and fast-state retention factors improved the RMSE of the LMSE fits to the hidden states (compare Fig. 4 with Fig. 9) but did not completely rescue LMSE predictions; still, EM more accurately identified the true slow and fast states of learning.

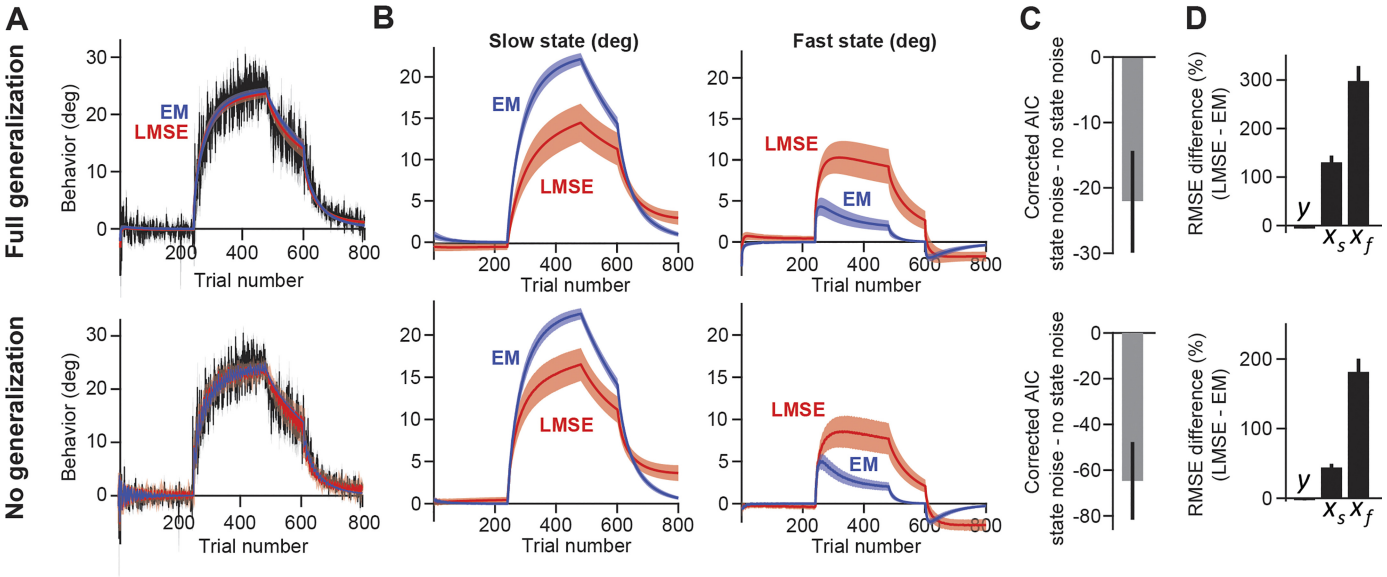

Fig. 10. Comparison of EM and LMSE on a trial-by-trial analysis of the data. We collected the behavior of $n = 20$ subjects in a visuomotor rotation task. We fit the two-state model to the trial-by-trial data recorded for individual subjects using EM and LMSE. We considered two trial-by-trial models that differed in terms of generalization. *Top*: the full-generalization model consisted of a single fast and slow state whose learning was completely generalized across targets. *Bottom* the no-generalization model consisted of separate fast and slow states for each of the 8 targets, whose learning did not generalize across targets. *A*: population behavior. We computed the average trial-to-trial behavior of the subject population. The average behavior (black) is shown overlaid with the average EM (blue) and average LMSE (red) fits. EM and LMSE had very similar fits to the behavior. *B*: predicted fast and slow states. For both the full and no-generalization models, EM estimated larger contributions from the slow state of learning and smaller contributions from the fast state of learning. Error bars indicate ± 1 SE. Here, the average time courses across the 8 fast states and 8 slow states are shown for the no-generalization model. *C*: we compared the corrected AIC of 2 competing likelihood models: one with state and motor noise and one without these noise sources. AICc was lower (better) for a model with state and motor noise. Here, we provide the mean difference in AICc for both models (state and motor noise likelihood – no state and motor noise likelihood). *D*: we used the trial-by-trial parameters to perform a set of control simulations. We simulated *paradigm 2* a total of 1,000 times and fit each simulated data set with EM and LMSE. Whereas LMSE fit the observed reaching behavior more closely (y), EM vastly outperformed LMSE in the identification of the hidden slow and fast states (x_s and x_f).

tion model, EM estimated a smaller slow-state retention factor than LMSE [paired t-test, $t(19) = 3.20$, $P = 0.0047$] and a larger slow-state error sensitivity [paired t-test, $t(19) = 3.65$, $P = 0.0017$]. For the no-generalization model, these trends were not statistically significant. As before, the variance of the LMSE parameter estimates exceeded those of EM for all retention factors and error sensitivities of the no-generalization model and all parameters except for the fast-state retention factor of the full-generalization model. These increased variances were caused by errant LMSE slow- and fast-state predictions that resembled the failure modes noted in Fig. 1*B* for the epoch-by-epoch analysis. For eight and seven of the 20 subjects (no-generalization and full-generalization, respectively), LMSE estimated unstable retention factors that exceeded 1. For three and six of the 20 subjects (no-generalization and full-generalization, respectively), LMSE estimates of slow-state error sensitivity tended toward zero, yielding slow states of learning that were largely insensitive to error.

We repeated our AICc analysis on the likelihood models fit by EM and LMSE. Recall that the LMSE model excludes state and motor noise and, therefore, possesses fewer parameters than EM. We found the same result as before (Fig. 2*C*); the likelihood used by EM possessed a lower AICc than the likelihood model used by LMSE [paired t-test for full generalization, $t(19) = 2.83$, $P = 0.011$, paired t-test for no generalization, $t(19) = 3.81$, $P = 0.001$; Fig. 10*C*]. Therefore, trial-by-trial analyses of our data assuming either complete generalization or no generalization of learning both suggested that a model that includes state and motor noise in the process of error-based learning was more likely to explain measured behavior.

Finally, although in our primary analysis we did not perform any averaging of our simulated data sets, these simulations were indirectly affected by our epoch-based analysis of subject behavior because we used these parameter sets as a basis for the simulated data. For this reason, we used the trial-by-trial parameters (Fig. 10*B*) to perform a set of control simulations. We replicated our epoch-by-epoch analysis by simulating *paradigm 2* (a total of 1,000 times with our no-generalization and full-generalization state-space models). We found that although LMSE fit the observed reaching behavior more closely (y; Fig. 10*D*), improving on EM by ~5%, EM vastly outperformed LMSE in the identification of the hidden slow and fast states by ~50–300% (x_s and x_f; Fig. 10*D*).

In summary, in both the epoch-by-epoch and trial-by-trial data, EM and LMSE identified different slow- and fast-state trajectories. The likelihood model maximized by EM was more likely to explain both the epoch-by-epoch and trial-by-trial behavior. That is, in all cases the evidence pointed to a learning model that was stochastic in its adaptation to error. In simulation, EM was superior to LMSE in the identification of the slow and fast states when the parameter set was taken from trial-by-trial subject behavior as well as from epoch-by-epoch subject behavior.

DISCUSSION

State-space models were first applied to data in adaptation experiments following the observation that experience of a single error produced robust trial-by-trial changes in behavior (Scheidt et al. 2001; Thoroughman and Shadmehr 2000). This

provided the possibility to assay learning not only in typical scenarios where perturbations were sustained but also in scenarios where the perturbations were random (Donchin et al. 2003; Srimal et al. 2008). The initial models assumed a single state; however, the observation of spontaneous recovery during saccade adaptation in monkeys (Kojima et al. 2004) suggested that experience of error engaged multiple learning processes. Smith et al. (2006) modified the state-space equations by proposing that the putative learning processes included a fast process that learned strongly from error but forgot rapidly and a slow process that learned weakly from error but exhibited robust retention. Unfortunately, the task of identifying these processes was difficult because there was typically no direct way to measure them. Rather, their state had to be inferred from their collective influence on behavior. Here, we approached this estimation problem in the context of data measured in typical motor learning studies and designed an algorithm to uncover the hidden processes.

Design of a new algorithm. Previous attempts to estimate fast and slow processes of learning had predominantly relied on least-squares techniques (Colagiorgio et al. 2015; Galea et al. 2015; McDougle et al. 2015; Trewartha et al. 2014). However, to our knowledge, robustness of the least-squares technique was not tested and compared with alternative algorithms. Here, we used LMSE to fit a two-state model to behavioral data collected in a reach-adaptation task. We found that although LMSE fit the observed behavior well, for a subset of subjects it appeared to misidentify properties of the hidden processes; for these subjects, LMSE estimated parameters that produced physically unrealistic trajectories of the fast and slow states. We hypothesized that LMSE occasionally yielded aberrant results because it incorrectly attributed any noise in the measured behavior to the measurement itself rather than the underlying learning process. In other words, LMSE was ignorant of any randomness in the state update process, which is affected by both state noise and motor noise (due to the process of error-based learning). To rectify this problem with LMSE, we developed a new algorithm based on EM.

Unlike LMSE, the EM algorithm is compatible with systems where both learning from error and generating a movement have independent noise sources. We developed a two-state model that represented the random processes involved in learning from error and production of movements. For such a system, our EM algorithm used a different and in some sense more complete likelihood model than LMSE.

EM has received limited application in the sensorimotor literature (Cheng and Sabes 2006). We speculate that this is because of two reasons. *1*) Previous applications of the algorithm were restricted to experiments that could be described by time-invariant state-space models. That is, in previous descriptions of EM, one could not use the algorithm with modern sensorimotor experiments that include behavioral probes such as error clamp trials (Scheidt et al. 2000) and set breaks, the second of which causes time-dependent changes to the state-space equations. Rather than using a closed-form solution (Cheng and Sabes 2006), here we used a numerical approach, allowing us to fit models to data that included error clamp trials and set breaks. *2*) Previous implementations could not specify bounds on the model parameters and constrain relationships between model parameters. Here, we solved this problem by implementing a generalized EM algorithm that maximized the expected complete log-likelihood function numerically within a constrained parameter space.

Evaluating the new algorithm. We performed a visuomotor rotation experiment and fit the measured data with EM and found that parameters estimated by the algorithm differed from those of LMSE. EM appeared to eliminate the aberrant single-subject fits observed for the LMSE algorithm. Additionally, EM parameters had lower variability across subjects, leading to reduced variance in the corresponding fast and slow state time courses. We computed the AICc for models that included (EM) or ignored (LMSE) noise in the process of motor learning. We found that the model with motor and state noise was more likely to explain the measured data in our experiment than one where noise was attributed externally to the measurement of subject behavior. Therefore, we were able to make two conclusions: *1*) experimental data suggested that equations that include noise in the learning process and moving process are a better descriptor of behavioral data than those that omit these noise sources; and *2*) EM, but not LMSE, is the appropriate algorithm to uncover parameters of a stochastic learning system.

To determine how well EM would perform over LMSE on a data set with state and motor noise, we performed simulations where the true trajectories of the hidden states were known. We tested EM and LMSE on simulated data generated in four learning paradigms across a variety of noise conditions and parameter values. Although in all cases EM performed slightly worse than LMSE in fitting the observed data, it consistently outperformed LMSE in the identification of the hidden states. Specifically, EM predicted hidden-state time courses that were more closely matched to the true states and identified model parameters that were more tightly distributed about the true values. For these reasons, we expect the EM toolbox to provide a more robust method of fitting state-space models to single-subject behavioral data.

In comparison with LMSE, EM identified learning parameters that had significantly reduced variance. For this reason, in simulated power analyses of typical experiments, we found that the use of EM significantly reduced the number of subjects needed to statistically make within-subject and between-subject comparisons. Therefore, it appears that the new algorithm might allow for a more robust method of hypothesis testing.

Limitations of the algorithm. Our model assumed that two sources of noise, a state noise and a motor noise, affected the processes of learning and movement production, respectively. Other authors have also considered state noise in models of learning (Cheng and Sabes 2006; Tanaka et al. 2012). However, at present, we do not have a complete understanding of the properties of such noise. With that said, here, we found evidence that a model with state noise was more likely to explain human behavior during a visuomotor rotation task. To show this, we repeated our analysis in Fig. 2*C* for the comparison of a model with both state and motor noise and a separate model with only motor noise. For our subject population, the corrected AIC for the model with both noise sources was lower (better) than that with only a motor noise source [paired t-test, $t(19) = 2.549$, $P < 0.05$]. Furthermore, the differentiation between state and motor noise is more harmonious with a Bayesian interpretation of motor learning (Kording et al. 2007; Wei and Körding 2010). Additionally, the existence of state noise is also consistent with autocorrelations that arise between successive movements that can accumulate due to

variability in planning a movement that is independent of variability in performing the movement itself (van Beers et al. 2013).

The manner in which we described state and motor noise was not entirely accurate. For example, we assumed that the variance of motor noise was signal independent. Although this assumption seems reasonable for visuomotor rotation learning where each movement has the same amplitude, for motor effectors like the eye, a better model of learning might account for scaling of noise with the amplitude of movement. Accounting for this signal dependency would require fundamental modifications to the E- and M-steps of our EM toolbox.

A useful modification to our model would be the inclusion of time-varying error sensitivity (Herzfeld et al. 2014b). Such a modification could be incorporated by adding an additional parameter that determines the rate at which error sensitivity changes over time. Along these lines, the processes of learning and retention may possess nonlinearities not accounted for by our model. These modifications to the model would require derivation of a different expected complete log-likelihood function and an extended Kalman filter (for nonlinear systems), and therefore, fundamental modifications to our EM toolbox would be required to fit such nonlinear behavior.

Another extension could be the inclusion of more than two states of learning. Preliminary evidence suggests that slow learning states can be subdivided into two component processes (yielding a total of 3 hidden states) with differing levels of susceptibility to temporal decay (Brennan and Smith 2016; Inoue et al. 2015). Such a model of learning would be directly compatible with our EM approach with the addition of a retention factor, error sensitivity, and initial state.

Different model fits could be obtained by changing the restrictions on the parameter space we searched for both algorithms during the fitting process. In some cases, the experimenter can fix the initial states to zero to improve fits. However, the goal of our work was to identify an algorithm that could perform robustly for any perturbation sequence independent of the subject's initial states and independent of the modeler's knowledge of the subject's initial states. For this reason we did not attempt to improve EM and LMSE performance by fixing the initial states, as this constraint cannot be applied in general circumstances.

The form of the model can lead to correlations within the estimated parameters. For example, consider washout of learning. The rate of washout is determined by both forgetting and learning from error. Fast washout of learning can be explained by high error sensitivity and low retention factor. For this reason and others, there is a tendency to predict that these parameters are correlated. We quantified correlations within parameters estimated by EM and LMSE. For EM, there were two pairs of model variables with appreciable correlation (absolute value of correlation coefficient > 0.4), and for LMSE there were three. The two pairs shared by both algorithms were slow-state retention/slow-state error sensitivity (EM, $r = -0.455$; LMSE, $r = -0.628$) and fast-state retention/slow-state error sensitivity (EM, $r = -0.412$; LMSE, $r = -0.682$). The third pair exclusive to LMSE was fast-state retention/slow-state retention (LMSE, $r = 0.536$). Therefore, both algorithms were affected by correlated parameter estimates, and the magnitude of these correlations appeared smaller for the EM algorithm.

There are alternative approaches to uncovering hidden behavioral states. EM can be thought of as a frequentist's approach to mathematical estimation. It identifies the parameter set with the most likely solution to the problem. However, EM assumes a flat prior distribution over the parameters; in other words, the algorithm currently does not allow the modeler to use prior information regarding the probability distributions of the parameters of the learning system. These considerations can be accounted for within the context of Bayesian approaches to estimation. To our knowledge, this Bayesian framework has not been applied to two-state models of adaptation; Bayesian techniques represent an exciting avenue that may further improve upon the robustness of the EM approach we pursued here.

Relationship between mathematical hidden states and neural substrates of learning. Why might it be useful to mathematically identify the hidden processes that underlie learning? In terms of behavior, various studies have posited that the fast and slow time scales of learning map onto dissociable components of a movement: during saccades, the early component of the movement exhibits properties that resemble influence of the slow process, whereas the later component of the same movement exhibits influence of the fast process (Chen-Harris et al. 2008). During reaching, fast processes appear to mirror more temporally labile components of memory, and slower processes appear more temporally stable (Ethier et al. 2008; Hadjiosif and Smith 2015; Kording et al. 2007; Smith et al. 2006). Fast processes may relate to explicit or cognitive types of motor learning, where slower processes are supported by implicit, unconscious motor learning mechanisms (McDougle et al. 2015; Taylor et al. 2014). Fast states of learning may require larger amounts of preparation time to be expressed than slow states of learning, which are present in behavior executed at low reaction times (Haith et al. 2015).

In terms of neural substrates of learning, some authors have found that the neural basis of the fast process may depend on the cerebellum, as evidenced by the observation that noninvasive cerebellar stimulation can modulate learning from error (Galea et al. 2011; Herzfeld et al. 2014a) and damage to the cerebellum can spare slow processes (Xu-Wilson et al. 2009). For example, people with cerebellar damage maintain the ability to modulate error sensitivity of the slow process (Hanajima et al. 2015). Imaging results suggest that for arm movements, both fast and slow adaptive processes may depend on the cerebellum (Kim et al. 2015) as well as regions of the cerebral cortex.

At the neuronal level, the existence of different time scales of memory may be present within the architecture of the cerebellum. A recent study found that the basic computational unit in the cerebellum may be microclusters of P cells that share a common preference for error (Herzfeld et al. 2015). Anatomic studies show that a given error is transmitted to the cerebellum via complex spikes that engage different microclusters of P cells placed in disparate regions of the cerebellum (Fujita and Sugihara 2013). This raises the possibility that a single error produces plasticity in multiple regions of the cerebellum, engaging distinct neural elements that can combine their outputs in service of adaptation (Shadmehr 2017). Indeed, P cells in the flocculus exhibit a preference for error direction (Yang and Lisberger 2014). When a visual error is in the preferred direction of a P cell, that cell produces complex spikes, which in turn results in depression of simple spikes on the subsequent trial. If the temporal distance between the two trials is large, these changes fade away, akin to a process of

forgetting. In contrast, that same visual error is in the anti-preferred direction of another group of P cells, resulting in reduction of complex spikes below baseline, which produces small potentiation of the simple spikes on the subsequent trial. These two groups of P cells exhibit different sensitivities to the same error and exhibit forgetting with passage of time, two elements that appear quite similar to mathematical two-state models inferred from behavior. That is, the neural basis of multiple time scales of memory may be in part associated with the diversity of error preferences in the P cells of the cerebellum.

Comparison of mathematical estimation of the hidden states of learning with these probes provides an opportunity to identify the neural substrates that mediate the multiple time scales of motor memory.

APPENDICES

This article contains four appendices. The first appendix contains a mathematical description of our EM algorithm. Our form of the EM algorithm applies a different maximization step from previous descriptions in the sensorimotor literature (Cheng and Sabes 2006). Our numerical implementation of the M-step would best classify our algorithm as a generalized EM algorithm. First, we provide a qualitative overview of EM and generalized EM. We then derive the equations required to apply EM to our two-state model. The toolbox that implements the generalized EM algorithm, along with supporting documentation, is available at http://shadmehrlab.org/tools. The second appendix provides a description of the least-squares (LMSE) algorithm. We discuss the general structure of LMSE and provide the equations we used for the algorithm. The third and final appendix discusses an extended model of learning where multiple targets are presented to the subject.

APPENDIX 1: MATHEMATICAL DESCRIPTION OF THE GENERALIZED EM ALGORITHM

Given a state-space model of behavior (*Eq. 10*), we ask how the parameters of the model can be estimated from a set of measured behavioral data. Suppose that our paradigm consists of N trials. The experimental design of our paradigm specifies the sequence of targets $[\{g\}_1^N = g^{(1)},g^{(2)},\cdots,g^{(N)}]$, error clamp trials, and perturbations $[\{r\}_1^N = r^{(1)},r^{(2)},\cdots,r^{(N)}]$. During the experiment, we record the subject's motor outputs $[\{u\}_1^N = u^{(1)},u^{(2)},\cdots,u^{(N)}]$, which, along with $\{g\}_1^N$, allows us to compute the subject's motor action on each trial relative to the target $\{y\}_1^N = y^{(1)},y^{(2)},\cdots,y^{(N)}$. We provide a description of these variables in the context of three common sensorimotor learning paradigms (force-field adaptation, visuomotor adaptation, and saccade adaptation) in Table A1.

Our goal is to determine the parameter set ($\theta = \{a_s,a_f,b_s,b_f,\sigma_x^2,\sigma_u^2,d\}$) that best explains the measured data. Note that the d parameter is relevant only for paradigms that include set breaks. A standard approach to parameter estimation is maximum likelihood estimation (MLE). MLE identifies the parameter set that maximizes the likelihood of observing the measured data, given a model parameter set. We will refer to this likelihood as the incomplete likelihood function. It is incomplete, as it does not include all the random variables of our system; i.e., it omits the hidden states of learning. Please note that other sources may refer to the incomplete likelihood function as the marginal likelihood function. Stated mathematically, MLE identifies the parameter set $\hat{\theta}$ according to $\hat{\theta} = \underset{\theta}{\text{argmax}}\, L(\{y\}_1^N|\theta)$. We provide a derivation of the incomplete likelihood for our two-state model in *The incomplete likelihood function*. Although many MLE problems can be solved by maximizing this function directly, for systems described by our two-state model (*Eq. 10*), this maximization has no closed-form solution and can also be difficult to solve numerically.

Another approach to the MLE problem is an algorithm known as EM. Instead of finding the maximum likelihood estimator in one step by maximizing the incomplete likelihood function, EM iteratively increases the incomplete likelihood function by maximizing a different objective function known as the expected complete log-likelihood function. We will derive this function shortly. Central to EM is the complete likelihood function described by $L_c = L(\{\boldsymbol{x}\}_1^N,\{y\}_1^N|\theta)$. We will later show that this complete likelihood is the product of several exponential terms. Therefore, it is simpler to work with the natural logarithm of the complete likelihood l_c, where $l_c = \log_e[L(\{\boldsymbol{x}\}_1^N,\{y\}_1^N|\theta)]$.

As its name suggests, EM is composed of an expectation (E-step) and a maximization (M-step) step. The algorithm begins by guessing an initial parameter set (θ_0) and then performs the E-step and M-step in order in an iterative fashion. During the E-step, we consider the conditional expectation of the complete log-likelihood function, $E[l_c|\{y\}_1^N,\theta_t]$, where θ_t is the estimate of the parameter set obtained from the M-step of the previous EM iteration. We will refer to this expectation as the expected complete log-likelihood function. We will later show that a functional form of the expected complete log-

Table A1. *Interpretation of model variables for common learning paradigms*

Model Variable	Sensorimotor Learning Paradigm		
	Force-field adaptation	Visuomotor adaptation	Saccade adaptation
$y^{(n)}$	The adaptation index describing the force profile	Heading angle of the reach relative to the target	The saccadic end point relative to the cued target
$r^{(n)}$	A value (typically 0 or 1) encoding the presence or absence of the force field. It can be standardized to some force field magnitude level and then take fractional values.	An external rotation to the cursor around the starting position of the reaching movement	A displacement of the target position presented after the subject executes her primary saccade
$e_c^{(n)}$	Error clamp trials in force-field tasks apply a stiff spring to eliminate error between subject forces and robot force; should take the value 0	The value of the clamped angular error between feedback of the subject's hand position and the target position, which, during no feedback trials, can take the value 0 (Kitago et al. 2013).	The value of the clamped angular error between the final target position and the end point of the subject's primary saccade.

Our derivation of the two-state model used general language that could apply across different sensorimotor learning modalities. Here, we provide a description of some key parameters in the context of force-field adaptation, visuomotor rotation, and saccade adaptation to assist the general reader.

likelihood function can be derived using the Kalman filter, concluding the E-step of the EM algorithm.

In the M-step, we compute an updated parameter set that maximizes the expected complete log-likelihood function according to $\theta_{t+1} = \underset{\theta}{\text{argmax}} E[l_c|\{y\}_1^N,\theta_t]$. Critically (see Wu 1983), iteration of the E- and M-steps guarantees that the incomplete likelihood function increases with each update to the model parameters: $L(\{y\}_1^N|\theta_{t+1}) > L(\{y\}_1^N|\theta_t)$. The E-step and M-step are iterated until the incomplete likelihood function converges to a stationary point.

Previous descriptions of the EM algorithm (Ghahramani and Hinton 1996; Cheng and Sabes 2006) have outlined the E-step and M-step for linear time-invariant dynamic systems similarly, albeit not identically, to the form of *Eq. 10*. The previous implementations of EM assumed state equations that were time invariant, representing experiments that had neither set breaks nor error clamp (Scheidt et al. 2000) trials. Under these assumptions, there existed closed-form solutions for the M-step. However, the introduction of set breaks in *Eq. 10* introduces time-varying nonlinearities to the two-state model that make the closed-form specification of the M-step difficult, if not impossible. Furthermore, our restrictions to the fast- and slow-state dynamics (i.e., the constraints relating the fast and slow retention factors and error sensitivities) complicate our ability to identify a closed-form expression for a parameter set that globally solves the M-step. In such cases, a more general form of the EM algorithm known as generalized EM is useful (Dempster et al. 1977). In generalized EM, rather than maximizing the expected complete log-likelihood function during the M-step, one selects θ_{t+1} such that

$$E\left[\log_e\left[L\left(\{\boldsymbol{x}\}_1^N,\{y\}_1^N\middle|\,\theta_{t+1}\right)\right]\middle|\{y\}_1^N,\theta_t\right] > E\left[\log_e\left[L\left(\{\boldsymbol{x}\}_1^N,\{y\}_1^N\middle|\,\theta_t\right)\right]\middle|\{y\}_1^N,\theta_t\right]$$

In words, to increase the incomplete likelihood function using EM, it is sufficient to identify a parameter set that simply increases the expected complete log-likelihood over the value associated with the parameter set attained on the previous EM iteration.

In our generalized EM algorithm, we select an invariant parameter space. We numerically search this parameter space during each M-step of the algorithm to maximize the expected complete log-likelihood function. This maximization ensures that we satisfy the condition of the generalized EM algorithm above. That is, by identifying the maximal value of the expected complete log-likelihood function in a parameter space that does not change from one iteration to the next, we guarantee that the updated parameter set is better than (or at least as good as) the previous parameter set, which is also contained in the same parameter space. This is a generalized EM algorithm in that it does not globally maximize the expected complete log-likelihood function. Although more computationally intensive than the standard EM algorithm, our generalized EM algorithm has the benefits of allowing the modeler to specify hard parameter bounds as well as functional constraints on the relationship between model parameters. Therefore, when using our algorithm, the modeler can restrict the parameter space to obtain only physically relevant solutions and appropriate two-state model dynamics. In the following section, we describe the mathematics that define our generalized EM algorithm.

Expectation step. The E-step requires derivation of the expected complete log-likelihood function. Note that the complete likelihood function can be factored given the Markov form of *Eq. 10* (Shadmehr and Mussa-Ivaldi 2012). This factorization allows the complete likelihood function to be expressed as the following product:

$$L_c = \left[\prod_{n=1}^{N} L\left(y^{(n)}\middle|\boldsymbol{x}^{(n)},\theta\right)\right]\left[\prod_{n=1}^{N-1} L\left(\boldsymbol{x}^{(n+1)}\middle|\boldsymbol{x}^{(n)},y^{(n)},\theta\right)\right]L\left(\boldsymbol{x}^{(1)}\right). \tag{A1.1}$$

Eq. A1.1 expresses the complete likelihood function in terms of three types of probability density functions. Our goal is to find general expressions for the likelihood functions on the right-hand side of *Eq. A1.1*. We can obtain the first likelihood $L(y^{(n)}|\boldsymbol{x}^{(n)},\theta)$ directly from the observation equation of *Eq. 10*. This likelihood is the probability density function of a normal, random variable, which is provided below:

$$L\left(y^{(n)}\middle|\boldsymbol{x}^{(n)},\theta\right) = N\left(\boldsymbol{c}^T\boldsymbol{x}^{(n)},\sigma_u^2\right). \tag{A1.2}$$

The second likelihood on the right-hand side of *Eq. A1.1* can be obtained from the state update equation of *Eq. 10* and is the probability density function for a multivariate normal random variable described by

$$L\left(\boldsymbol{x}^{(n+1)}\middle|\boldsymbol{x}^{(n)},y^{(n)},\theta\right) = N\left(A^{(n)}\boldsymbol{x}^{(n)} + \boldsymbol{b}^{(n)}e^{(n)},Q^{(n)}\right) \tag{A1.3}$$

To fully specify the complete likelihood of *Eq. A1.1*, we must also obtain an expression for $L(\boldsymbol{x}^{(1)})$, the probability density function for the initial state. We will assume that the initial state of the learner is itself a normal random variable:

$$L\left(\boldsymbol{x}^{(1)}\right) = N\left(\overline{\boldsymbol{x}}_1,\overline{V}_1\right). \tag{A1.4}$$

The mean of the normal random variable $\overline{\boldsymbol{x}}_1$ can be represented as $\overline{\boldsymbol{x}}_1 = [x_s^{(1)} \;\; x_f^{(1)}]^T$, which introduces two additional parameters to our state-space model, $x_s^{(1)}$ and $x_f^{(1)}$, the mean initial values of the slow and fast states, respectively. We will assume that we can represent the variance-covariance matrix $\overline{V}_1$ in the diagonal form $\overline{V}_1 = \begin{bmatrix}\sigma_1^2 & 0\\ 0 & \sigma_1^2\end{bmatrix}$, which introduces the parameter σ_1^2, the variance of the initial states. Our full parameter set that we seek to identify now consists of 10 variables, i.e., $\theta = \{a_s,a_f,b_s,b_f,\sigma_x^2,\sigma_u^2,d,x_s^{(1)},x_f^{(1)},\sigma_1^2\}$. Substitution of *Eqs. A1.2 to A1.4* into *Eq. A1.1* yields the following expression for the complete likelihood function:

$$\begin{aligned} L_C = &\prod_{n=1}^{N}\left(2\pi\sigma_u^2\right)^{-1/2}\exp\left(-\frac{1}{2\sigma_u^2}\left(y^{(n)}-c^T\boldsymbol{x}^{(n)}\right)^2\right)\\ &\times\prod_{n=1}^{N-1}\left(2\pi\right)^{-1}\left|Q^{(n)}\right|^{-1/2}\exp\left(-\frac{1}{2}\left(\boldsymbol{x}^{(n+1)}-A^{(n)}\boldsymbol{x}^{(n)}-e^{(n)}f^{(n)}\right)^T{Q^{(n)}}^{-1}\left(\boldsymbol{x}^{(n+1)}-A^{(n)}\boldsymbol{x}^{(n)}-e^{(n)}f^{(n)}\right)\right)\\ &\times\left(2\pi\right)^{-1}\left|\overline{V}_1\right|^{-1/2}\exp\left(-\frac{1}{2}\left(\boldsymbol{x}^{(1)}-\overline{\boldsymbol{x}}_1\right)^T{\overline{V}_1}^{-1}\left(\boldsymbol{x}^{(1)}-\overline{\boldsymbol{x}}_1\right)\right)\end{aligned} \tag{A1.5}$$

Due the various products of exponential functions in *Eq. A1.5*, it is simpler to consider the natural logarithm of the likelihood function, which we will refer to as l_c. Taking the natural logarithm of both sides of *Eq. A1.5* yields the following form for l_c.

$$
\begin{aligned}
l_c = &-\frac{1}{2\sigma_u^2}\sum_{n=1}^{N}\left(y^{(n)} - \boldsymbol{c}^T\boldsymbol{x}^{(n)}\right)^2 \\
&-\frac{1}{2}\sum_{n=1}^{N-1}\left(\boldsymbol{x}^{(n+1)} - A^{(n)}\boldsymbol{x}^{(n)} - \boldsymbol{b}^{(n)}e^{(n)}\right)^T Q^{(n)-1}\left(\boldsymbol{x}^{(n+1)} - A^{(n)}\boldsymbol{x}^{(n)} - \boldsymbol{b}^{(n)}e^{(n)}\right) \\
&-\frac{1}{2}\left(\boldsymbol{x}^{(1)} - \bar{\boldsymbol{x}}_1\right)^T \bar{V}_1^{-1}\left(\boldsymbol{x}^{(1)} - \bar{\boldsymbol{x}}_1\right) - \frac{1}{2}\log_e\left(\left|\bar{V}_1\right|\right) - \frac{N}{2}\log_e\left(\sigma_u^2\right) - \frac{3N}{2}\log_e(2\pi) - \frac{1}{2}\sum_{n=1}^{N-1}\log_e\left(\left|Q^{(n)}\right|\right).
\end{aligned}
\tag{A1.6}
$$

We proceed with the E-step by deriving an expression for $E[l_c|\{y\}_1^N,\theta_t]$. This conditional expectation yields the expected complete log-likelihood function that we analyze in the M-step. To conserve space, we will represent the conditioned terms in the expectation using a $\cdots$ symbol. We can easily obtain $E[l_c|\cdots]$ by expanding the quadratic terms in *Eq. A1.6* and then taking the conditional expectation. Doing so yields the following intermediate form of the expected complete log-likelihood function:

$$
\begin{aligned}
E\left[l_c\middle|\cdots\right] = &-\frac{1}{2\sigma_u^2}\sum_{n=1}^{N} y^{(n)2} + \boldsymbol{c}^T E\left[\boldsymbol{x}^{(n)}\boldsymbol{x}^{(n)T}\middle|\cdots\right]\boldsymbol{c} - 2y^{(n)}\boldsymbol{c}^T E\left[\boldsymbol{x}^{(n)}\middle|\cdots\right] \\
&-\frac{1}{2}\sum_{n=1}^{N-1}\begin{bmatrix}
E\left[\boldsymbol{x}^{(n+1)T}Q^{(n)-1}\boldsymbol{x}^{(n+1)}\middle|\cdots\right] - E\left[\boldsymbol{x}^{(n+1)T}Q^{(n)-1}A^{(n)}\boldsymbol{x}^{(n)}\middle|\cdots\right] - E\left[\boldsymbol{x}^{(n+1)T}\middle|\cdots\right]Q^{(n)-1}\boldsymbol{b}^{(n)}e^{(n)} \\
-E\left[\boldsymbol{x}^{(n)T}A^{(n)T}Q^{(n)-1}\boldsymbol{x}^{(n+1)}\middle|\cdots\right] + E\left[\boldsymbol{x}^{(n)T}A^{(n)T}Q^{(n)-1}A^{(n)}\boldsymbol{x}^{(n)}\middle|\cdots\right] \\
+E\left[\boldsymbol{x}^{(n)T}\middle|\cdots\right]A^{(n)T}Q^{(n)-1}\boldsymbol{b}^{(n)}e^{(n)} - e^{(n)}\boldsymbol{b}^{(n)T}Q^{(n)-1}E\left[\boldsymbol{x}^{(n+1)}\middle|\cdots\right] \\
+e^{(n)}\boldsymbol{b}^{(n)T}Q^{(n)-1}A^{(n)}E\left[\boldsymbol{x}^{(n)}\middle|\cdots\right] + e^{(n)}\boldsymbol{b}^{(n)T}Q^{(n)-1}\boldsymbol{b}^{(n)}e^{(n)}
\end{bmatrix} \\
&-\frac{1}{2}\left[E\left[\boldsymbol{x}^{(1)T}\bar{V}_1^{-1}\boldsymbol{x}^{(1)}\middle|\cdots\right] - E\left[\boldsymbol{x}^{(1)T}\middle|\cdots\right]\bar{V}_1^{-1}\bar{\boldsymbol{x}}_1 - \bar{\boldsymbol{x}}_1^T\bar{V}_1^{-1}E\left[\boldsymbol{x}^{(1)}\middle|\cdots\right] + \bar{\boldsymbol{x}}_1^T\bar{V}_1^{-1}\bar{\boldsymbol{x}}_1\right] \\
&-\frac{1}{2}\log_e\left(\bar{V}_1\right) - \frac{N}{2}\log_e\left(\sigma_u^2\right) - \frac{3N}{2}\log_e(2\pi) - \frac{1}{2}\sum_{n=1}^{N-1}\log_e\left(\left|Q^{(n)}\right|\right).
\end{aligned}
\tag{A1.7}
$$

As we can see from the above equation, the expected value operator affects only terms within *Eq. A1.7* that are functions of the hidden states. Our final step is to derive an alternative form for the expectation terms in *Eq. A1.7* that are quadratic functions of the unknown states. We note the following identity, which applies to any pair of multivariate random variables $\boldsymbol{x}$ and $\boldsymbol{y}$ and some matrix A of appropriate dimension.

$$E\left[\boldsymbol{x}^T A\boldsymbol{y}\right] = E\left[\boldsymbol{x}\right]^T AE\left[\boldsymbol{y}\right] + tr\left[A\mathrm{cov}\left(\boldsymbol{y},\boldsymbol{x}\right)\right].$$

Here $tr[\cdots]$ is the trace operator. This identity allows us to express the quadratic terms of *Eq. A1.7* as a function of linear state expectations and covariances, which we can compute using the Kalman filter. Applying this identity to *Eq. A1.7* yields our final expression for the expected complete log-likelihood function:

$$
\begin{aligned}
E\left[l_c\right] = &-\frac{1}{2\sigma_u^2}\sum_{n=1}^{N} y^{(n)2} + \boldsymbol{c}^T\left(V^{n|N} + \hat{\boldsymbol{x}}^{n|N}\hat{\boldsymbol{x}}^{n|N^T}\right)\boldsymbol{c} - 2y^{(n)}\boldsymbol{c}^T\hat{\boldsymbol{x}}^{n|N} \\
&-\frac{1}{2}\sum_{n=1}^{N-1}\begin{bmatrix}
\hat{\boldsymbol{x}}^{n+1|N^T}Q^{(n)-1}\hat{\boldsymbol{x}}^{n+1|N} + tr\left[Q^{(n)-1}V^{n+1|N}\right] - \hat{\boldsymbol{x}}^{n+1|N^T}Q^{(n)-1}A^{(n)}\hat{\boldsymbol{x}}^{n|N} - tr\left[Q^{(n)-1}A^{(n)}V^{n+1,n|N^T}\right] \\
-\hat{\boldsymbol{x}}^{n+1|N^T}Q^{(n)-1}\boldsymbol{b}^{(n)}e^{(n)} - \hat{\boldsymbol{x}}^{n|N^T}A^{(n)T}Q^{(n)-1}\hat{\boldsymbol{x}}^{n+1|N} - tr\left[A^{(n)T}Q^{(n)-1}V^{n+1,n|N}\right] \\
+\hat{\boldsymbol{x}}^{n|N^T}A^{(n)T}Q^{(n)-1}A^{(n)}\hat{\boldsymbol{x}}^{n|N} + tr\left[A^{(n)T}Q^{(n)-1}A^{(n)}V^{n|N}\right] + \hat{\boldsymbol{x}}^{n|N^T}A^{(n)T}Q^{(n)-1}\boldsymbol{b}^{(n)}e^{(n)} \\
-e^{(n)}\boldsymbol{b}^{(n)T}Q^{(n)-1}\hat{\boldsymbol{x}}^{n+1|N} + e^{(n)}\boldsymbol{b}^{(n)T}Q^{(n)-1}A^{(n)}\hat{\boldsymbol{x}}^{n|N} + e^{(n)}\boldsymbol{b}^{(n)T}Q^{(n)-1}\boldsymbol{b}^{(n)}e^{(n)}
\end{bmatrix} \\
&-\frac{1}{2}\left[\hat{\boldsymbol{x}}^{1|N^T}\bar{V}_1^{-1}\hat{\boldsymbol{x}}^{1|N} + tr\left[\bar{V}_1^{-1}V^{1|N}\right] - \hat{\boldsymbol{x}}^{1|N^T}\bar{V}_1^{-1}\bar{\boldsymbol{x}}_1 - \bar{\boldsymbol{x}}_1^T\bar{V}_1^{-1}\hat{\boldsymbol{x}}^{1|N} + \bar{\boldsymbol{x}}_1^T\bar{V}_1^{-1}\bar{\boldsymbol{x}}_1\right] \\
&-\frac{1}{2}\log_e\left(\left|\bar{V}_1\right|\right) - \frac{N}{2}\log_e\left(\sigma_u^2\right) - \frac{3N}{2}\log_e(2\pi) - \frac{1}{2}\sum_{n=1}^{N-1}\log_e\left(\left|Q^{(n)}\right|\right).
\end{aligned}
\tag{A1.8}
$$

Eq. A1.8 is the culminating result of the E-step. Note that the following shorthand notations have been applied in *Eq. A1.8*.

$$
\begin{aligned}
\hat{\boldsymbol{x}}^{n|N} &= E\left[\boldsymbol{x}^{(n)}\middle|\left\{y\right\}_1^N,\theta_t\right] \\
V^{n|N} &= \mathrm{var}\left(\boldsymbol{x}^{(n)}\middle|\left\{y\right\}_1^N,\theta_t\right) \\
V^{n+1,n|N} &= \mathrm{cov}\left(\boldsymbol{x}^{(n+1)},\boldsymbol{x}^{(n)}\middle|\left\{y\right\}_1^N,\theta_t\right)
\end{aligned}
\quad . \tag{A1.9}
$$

The shorthand quantities $\hat{\boldsymbol{x}}^{n|N}$, $V^{n|N}$, and $V^{n+1,n|N}$ can be computed using a smoothed Kalman filter (Ghahramani and Hinton 1996). To summarize the Kalman smoother equations, we begin by evaluating the following posteriors that are computed using a forward pass of the standard Kalman filter:

$$
\begin{aligned}
\hat{\boldsymbol{x}}^{n|n} &= E\left[\boldsymbol{x}^{(n)}\middle|\left\{y\right\}_1^n,\theta_t\right] \\
V^{n|n} &= \mathrm{var}\left(\boldsymbol{x}^{(n)}\middle|\left\{y\right\}_1^n,\theta_t\right)
\end{aligned}
\quad . \tag{A1.10}
$$

The terms $\hat{\boldsymbol{x}}^{n|n}$ and $V^{n|n}$ are our posterior state estimate and variance covariance matrix on the nth trial, given our current parameter estimate and all of our observations up to the nth trial. Note that this expectation and variance are similar but not equivalent to the desired $\hat{\boldsymbol{x}}^{n|N}$ and $V^{n|N}$, which we will refer to as our smoothed Kalman estimates. These smoothed Kalman estimates are the expectation and variance of the state on the nth trial, given all the observations we have made. To first compute $\hat{\boldsymbol{x}}^{n|n}$ and $V^{n|n}$, we must calculate the Kalman gain $\boldsymbol{k}^{(n)}$ according to

$$\boldsymbol{k}^{(n)} = \left(\boldsymbol{c}^T V^{n|n-1}\boldsymbol{c} + \sigma_u^2\right)^{-1} V^{n|n-1}\boldsymbol{c}. \tag{1.11}$$

To compute the Kalman gain, we require the prior estimate for the variance-covariance matrix, denoted as $V^{n|n-1}$. Using the Kalman gain, we can compute $\hat{\boldsymbol{x}}^{n|n}$ and $V^{n|n}$ according to the following equations:

$$\hat{\boldsymbol{x}}^{n|n} = \hat{\boldsymbol{x}}^{n|n-1} + \boldsymbol{k}^{(n)}\left(y^{(n)} - \boldsymbol{c}^T\hat{\boldsymbol{x}}^{n|n-1}\right) \tag{A1.12}$$

and

$$V^{n|n} = \left(\mathrm{I} - \boldsymbol{k}^{(n)}\boldsymbol{c}^T\right)V^{n|n-1}. \tag{A1.13}$$

Next, we forward project these posteriors to obtain the prior estimates for the next trial:

$$\hat{\boldsymbol{x}}^{n+1|n} = A^{(n)}\hat{\boldsymbol{x}}^{n|n} + \boldsymbol{b}^{(n)}e^{(n)} \tag{A1.14}$$

and

$$V^{n+1|n} = A^{(n)}V^{n|n}A^{(n)T} + Q^{(n)}. \tag{A1.15}$$

The forward Kalman filter proceeds by recursively iterating *Eqs. A1.11* to *A1.15* a total of N times to compute $\hat{\boldsymbol{x}}^{1|1},\hat{\boldsymbol{x}}^{2|2},\cdots,\hat{\boldsymbol{x}}^{N|N}$ as well as $V^{1|1},V^{2|2},\cdots,V^{N|N}$. The prior states and covariances will also be required for smoothing. We initialize this recursion with the priors $\hat{\boldsymbol{x}}^{1|0}$ and $V^{1|0}$. Here, these priors are taken as the parameter estimates for $\overline{\boldsymbol{x}}_1$ and $\overline{V}_1$ obtained on the previous iteration of the algorithm. That is, the prior expectation and variance are computed from the values of $x_s^{(1)}$, $x_f^{(1)}$, and σ_1^2 that were obtained from the M-step of the previous EM iteration.

To obtain the expectations and covariances required for EM (*Eq. A1.9*), we will now perform Kalman smoothing. The Kalman smoother uses backward recursions to compute the means and variances of the probability distributions described in *Eq. A1.9*. In other words, after obtaining the posterior state and variance-covariance matrix for all N time steps, we can recursively smooth our previous estimates. Our current implementation has been described previously (Cheng and Sabes 2006; Ghahramani and Hinton 1996). We note that the smoothed Kalman estimates on the final time step, N, were already computed in the final step of the forward Kalman filter. Therefore, computation of the quantities in *Eq. A1.9* begins with time step $N-1$. First we compute the helper variable $J^{(n)}$, which functions similarly to a Kalman gain:

$$J^{(n)} = V^{n|n}A^{(n)T}\left(V^{n+1|n}\right)^{-1}. \tag{A1.16}$$

With the computation of $J^{(n)}$, we can now compute $V^{n|N}$, our smoothed variance-covariance matrix:

$$V^{n|N} = V^{n|n} + J^{(n)}\left(V^{n+1|N} - V^{n+1|n}\right)J^{(n)T}. \tag{A1.17}$$

We also need to compute our smoothed state estimates:

$$\hat{\boldsymbol{x}}^{n|N} = \hat{\boldsymbol{x}}^{n|n} + J^{(n)}\left(\hat{\boldsymbol{x}}^{n+1|N} - \hat{\boldsymbol{x}}^{n+1|n}\right). \tag{A1.18}$$

Recursion of *Eqs. A1.16* to *A1.18* computes $\hat{\boldsymbol{x}}^{1|N},\hat{\boldsymbol{x}}^{2|N},\cdots,\hat{\boldsymbol{x}}^{N-1|N}$ and $V^{1|N},V^{2|N},\cdots,V^{N-1|N}$. To complete the E-step, we also require a smoothed estimate for the covariance of consecutive states denoted by $V^{n+1,n|N}$. We can obtain this covariance using the following equation:

$$V^{n+1,n|N} = V^{n+1|N}J^{(n)}. \tag{A1.19}$$

Note that $V^{2,1|N},V^{3,2|N},\cdots,V^{N,N-1|N}$ does not need to be computed in a recursive process and can be calculated after recursion of *Eqs. A1.16* to *A1.18*.

Maximization step. In the M-step of the EM algorithm, the goal is to maximize the expected complete log-likelihood function that is derived in the E-step. For our two-state model, there exists no closed-form expression that globally maximizes the expected complete log-likelihood function. Therefore, we used a generalized M-step that numerically maximizes the expected complete log-likelihood function (*Eqn A1.8*) in a constrained parameter space. We maintained the same parameter space for each iteration of our generalized EM algorithm. As we described in the introduction to this appendix, maintaining this invariant parameter space is sufficient to guarantee convergence of the EM algorithm.

To perform our numerical maximization, we used *fmincon* in MATLAB R2016a. Because *fmincon* performs constrained minimization, we converted our maximization problem to a minimization problem by minimizing the negated expected complete log-likelihood in *Eq. A1.8*, with respect to the two-state model parameter set $\theta = \{a_s,a_f,b_s,b_f,\sigma_{\boldsymbol{x}}^2,\sigma_u^2,d,x_s^{(1)},x_f^{(1)},\sigma_1^2\}$. We constrained the parameter space for this numerical optimization in two ways. First, we specified lower and upper bounds for all the model parameters according to *Eq. A1.20*:

$$\begin{aligned}
&a_{s,\min} \le a_s \le a_{s,\max} && a_{f,\min} \le a_f \le a_{f,\max}\\
&b_{s,\min} \le b_s \le b_{s,\max} && b_{f,\min} \le b_f \le b_{f,\max}\\
&\sigma_{\boldsymbol{x},\min}^2 \le \sigma_{\boldsymbol{x}}^2 \le \sigma_{\boldsymbol{x},\max}^2 && \sigma_{u,\min}^2 \le \sigma_u^2 \le \sigma_{u,\max}^2\\
&x_{s,\min}^{(1)} \le x_s^{(1)} \le x_{s,\min}^{(1)} && x_{f,\min}^{(1)} \le x_f^{(1)} \le x_{f,\max}^{(1)}\\
&\sigma_{1,\min}^2 \le \sigma_1^2 \le \sigma_{1,\max}^2 && d_{min} \le d \le d_{max}.
\end{aligned} \tag{A1.20}$$

The numerical values for the upper and lower bounds that specify *Eq. A1.20* are provided in Table 2. We used identical bounds for our least-squares algorithm, as described in APPENDIX 2: OVERVIEW OF THE LMSE ALGORITHM. The second way we constrained our parameter space is by enforcing conventional two-state model dynamics. Recall that the fast and slow states have the following properties; the fast state learns rapidly but also forgets rapidly. The slow state learns slowly but forgets slowly. To enforce these state dynamics, one can specify the following parameter constraints:

$$\begin{aligned}
&a_s \ge a_f + \Delta a\\
&b_f \ge b_s + \Delta b.\\
&\Delta a, \Delta b > 0
\end{aligned} \tag{A1.21}$$

As defined in *Eq. A1.21*, the slow state will have a greater retention factor than the fast state, and the fast state will have greater error sensitivity than the slow state. For *Eq. A1.21*, we used the value 0.001 for Δa and Δb. To summarize the generalized M-step, we numerically maximized *Eq. A1.8* using *fmincon* in a parameter space that is constrained by *Eqs. A1.20* and *A1.21*.

Note that the selection of the upper and lower bounds in Table 2 will be specific to modeler preferences and the features of the behavioral data. In the current work, we selected upper bounds on the initial states that were equal in magnitude to the size of the perturbation. We felt that this was a logical bound, as it represents the maximum value that could be attained by the slow or fast state at any point during the adaptation time course in the absence of noise. For our retention factors, we specified an upper bound of 1.1. We selected this bound to be greater than 1 to demonstrate that the LMSE algorithm tended to identify unstable properties in the slow state of learning. In *Changing the bounds on the parameter space*, we detail a control analysis where we changed this upper bound to 1 to prevent the identification of unstable fast and slow retention factors. For our error sensitivities, we specified a lower and upper bound of 0 and 1 to prevent "negative" learning or unstable learning, respectively. Finally, for all the variances of our noise terms (state, motor, and initial state),

we specified an upper bound of $10^{\circ 2}$. This parameter will be specific to the range and units of the behavioral data. Here, we selected this value to greatly exceed the variance of the residuals for the state-space model fit to any of our subject behaviors.

Algorithm summary. Here, we offer a practical summary of the algorithm. The algorithm begins by specifying an initial guess for θ_0, which is the initial value for the model parameters that will seed the algorithm.

1. Use the current parameter estimate θ_t to compute the posteriors $\hat{\boldsymbol{x}}^{1|1},\hat{\boldsymbol{x}}^{2|2},\cdots,\hat{\boldsymbol{x}}^{N|N}$ and $V^{1|1},V^{2|2},\cdots,V^{N|N}$ as well as the priors $\hat{\boldsymbol{x}}^{2|1},\hat{\boldsymbol{x}}^{3|2},\cdots,\hat{\boldsymbol{x}}^{N|N-1}$ and $V^{2|1},V^{3|2},\cdots,V^{N|N-1}$ by recursively applying *Eqs. A1.11* to *A1.15*. These forward recursions are seeded using the current parameter estimates $\hat{\boldsymbol{x}}^{1|0}=\bar{x}_1$ and $V^{1|0}=\bar{V}_1$.
2. Use the posterior and prior estimates from *step 1* to compute the smoothed conditional expectations and variances $\hat{\boldsymbol{x}}^{1|N},\hat{\boldsymbol{x}}^{2|N},\cdots,\hat{\boldsymbol{x}}^{N-1|N}$ and $V^{1|N},V^{2|N},\cdots,V^{N-1|N}$ by recursively applying *Eqs. A1.16* to *A1.18* backward in time. Compute the conditional covariances $V^{2,1|N},V^{3,2|N},\cdots,V^{N,N-1|N}$ by applying *Eq. A1.19*.
3. Numerically maximize (e.g., *fmincon* in MATLAB) the expected complete log-likelihood function (*Eq. A1.8*) with respect to the model parameters subject to desired bounds (*Eq. A1.20*) and linear constraints (*Eq. A1.21*). The maximizing model parameters now become the parameter estimates for the current EM iteration.
4. Return to *step 1* and start the next EM iteration using the updated model parameters computed in *step 3* to perform the state estimation. Stop when the incomplete likelihood function has converged.

The incomplete likelihood function. The EM algorithm iteratively locates local extrema of the incomplete (marginal) likelihood function. This function can be evaluated at the conclusion of each iteration of the algorithm to track convergence. Here, we provide a brief derivation of the form of the incomplete likelihood function $L(\{y\}_1^N|\theta)$. First, by successive application of the definition of conditional likelihood, we can factor the incomplete likelihood function as follows:

$$L\left(y^{(1)},y^{(2)},\cdots,y^{(N)}\,\middle|\,\theta\right)=L\left(y^{(1)}\,\middle|\,\theta\right)\prod_{n=2}^{N}L\left(y^{(n)}\,\middle|\,\theta,\{y\}_1^{n-1}\right). \tag{A1.22}$$

This factored incomplete likelihood reveals a direct relationship between the complete likelihood and the Kalman filter; $L(y^{(n)}|\theta,\{y\}_1^{n-1})$ is a normal, random variable with a mean and variance that can be computed from the priors obtained using the forward Kalman filter:

$$\begin{aligned}E\left[y^{(n)}\,\middle|\,\theta,\{y\}_1^{n-1}\right]&=E\left[\boldsymbol{c}^T\boldsymbol{x}^{(n)}+\varepsilon_u^{(n)}\,\middle|\,\theta,\{y\}_1^{n-1}\right]=\boldsymbol{c}^T\hat{\boldsymbol{x}}^{n|n-1}\\ \mathrm{var}\left(y^{(n)}\,\middle|\,\theta,\{y\}_1^{n-1}\right)&=\mathrm{var}\left(\boldsymbol{c}^T\boldsymbol{x}^{(n)}+\varepsilon_u^{(n)}\,\middle|\,\theta,\{y\}_1^{n-1}\right)\\ &=\boldsymbol{c}^TV^{n|n-1}\boldsymbol{c}+\sigma_u^2.\end{aligned} \tag{A1.23}$$

Therefore, the incomplete likelihood function can be expressed as follows:

$$L\left(y^{(1)},y^{(2)},\cdots,y^{(N)}\,\middle|\,\theta\right)=\prod_{n=1}^{N}N\left(\boldsymbol{c}^T\hat{\boldsymbol{x}}^{n|n-1},\boldsymbol{c}^TV^{n|n-1}\boldsymbol{c}+\sigma_u^2\right). \tag{1.24}$$

Given that *Eq. A1.24* is a product of exponentials, we consider the natural logarithm:

$$\begin{aligned}\log_e\left(L\left(\{y\}_1^N\,\middle|\,\theta\right)\right)=&-\frac{1}{2}\sum_{n=1}^{N}\frac{1}{\Sigma^{(n)}}\left(y^{(n)}-\mu^{(n)}\right)^2\\ &-\frac{1}{2}\sum_{n=1}^{N}\log_e\left(\Sigma^{(n)}\right)-\frac{N}{2}\log_e\left(2\pi\right)\\ \text{where}\quad \mu^{(n)}&=\boldsymbol{c}^T\hat{\boldsymbol{x}}^{n|n-1}\\ \Sigma^{(n)}&=\boldsymbol{c}^TV^{n|n-1}\boldsymbol{c}+\sigma_u^2.\end{aligned} \tag{A1.25}$$

In summary, to compute the incomplete log-likelihood associated with a given set of model parameters, we use *Eqs. A1.11* to *A1.15* to compute the prior hidden state expectations and variances and subsequently apply *Eq. A1.25*.

Convergence. Here, we discuss issues related to the convergence of the generalized EM algorithm. In a constrained parameter space, we can expect our implementation of the generalized EM algorithm to converge to either a stationary point or a boundary of the constrained parameter space (Nettleton 1999). As with any EM algorithm, we are not guaranteed that this stationary point is the desired global maximum of the incomplete log-likelihood function. The stationary point reached by an EM algorithm is determined by its initial conditions (i.e., the starting parameter guess). Therefore, it is imperative to perform the EM algorithm using different initial conditions. Here, for each set of data, we used 5 or 10 initial conditions (for simulated and behavioral data, respectively) in an attempt to identify the parameter set that resulted in the greatest incomplete log-likelihood. We found that using 50 different initial conditions for the algorithm did not meaningfully affect our results, and therefore, we chose a smaller number of initial conditions to make the computation time for our study more tractable.

For each initial condition, we performed a fixed number of EM iterations. We found that the number of iterations required to achieve convergence scaled with the size of the data set. In virtually all cases, we found that 100 iterations of the EM algorithm were more than sufficient to achieve convergence of the incomplete log-likelihood function. The only exception was for our trial-by-trial control analysis, where we used 200 iterations, due to the greater number of trials and slower convergence rate. Although we used a fixed number of iterations in this study, the modeler could set a convergence criterion that terminates the algorithm once the change in log-likelihood from one EM iteration to the next fell below some threshold.

APPENDIX 2: OVERVIEW OF THE LMSE ALGORITHM

The current standard technique used for fitting state-space models to motor learning data is one that selects the model parameters that minimize the squared error between measured variables and model predictions. Here, we offer a brief description of one form of this algorithm, the least mean squared error (LMSE) technique, which generalizes to other least-squares techniques implemented within the literature. To use LMSE, we imagine a noise-free state-space analog of *Eq. 10*. We previously described this system in *Eq. 18* but reproduce it again below:

$$\begin{aligned}\boldsymbol{x}^{(n+1)}&=A^{(n)}\boldsymbol{x}^{(n)}+\boldsymbol{b}^{(n)}e^{(n)}\\ y^{(n)}&=\boldsymbol{c}^T\boldsymbol{x}^{(n)}\end{aligned}. \tag{A2.1}$$

This noise-free system is equivalent to the expected value of the states and observed behaviors predicted by our general two-state model of *Eq. 10*. To be clear, $e^{(n)}$ refers to the model prediction for the error in our noise-free system (i.e., the expected value of the error in *Eq. 10* given our model parameters), not the errors actually measured during the experiment (with the exception of error-clamp trials where the expected value of the error is equal to the error imposed on that trial).

Note that *Eq. A2.1* describes a deterministic system; specification of the parameter set $\theta_{LMSE}=\{a_s,a_f,b_s,b_f,x_s^{(1)},x_f^{(1)},d\}$ determines entirely the progression of the slow and fast states for a given sequence of perturbations, error clamp trials, and set breaks. The LMSE algorithm simply searches this seven-dimensional parameter space to identify the parameter set that satisfies the following optimization:

$$\theta_{LMSE}=\arg\min_{\theta}\left\{\frac{1}{N}\sum_{n=1}^{N}\left(y^{(n)}-\hat{y}^{(n)}\right)^2\right\}. \tag{A2.2}$$

Here, $y^{(n)}$ is the measurement on trial n and $\hat{y}^{(n)}$ the model prediction, which is computed from the output of *Eq. A2.1*. The argument of the *argmin* function of *Eq. A2.2* is the mean squared error between the observation and model prediction for a given parameter set. For all LMSE fits in this study, we used the MATLAB function *fmincon* to identify the least-squares solutions. For each LMSE fit, the algorithm was seeded at 50 points scattered across the parameter space to better ensure the identification of the minimum mean squared error within the search space. The parameter space used for the LMSE algorithm was identical to that of the EM algorithm and is reported in Table 2.

The least-squares algorithm can also be thought of as a maximum likelihood estimator under certain conditions. For our two-state model, the least-squares algorithm maximizes the likelihood of a system where any randomness in the measured behavior is attributed to the measurement process itself (i.e., it is external to the underlying motor learning process). For this system, we assume that the actual motor action generated by the subject $y^{(n)}_{true}$ differs from the observed behavior $y^{(n)}$ due to some non-zero Gaussian measurement noise:

$$\begin{gathered}\boldsymbol{x}^{(n+1)} = A^{(n)}\boldsymbol{x}^{(n)} + \boldsymbol{b}^{(n)}e^{(n)} \\ y^{(n)}_{true} = \boldsymbol{c}^T\boldsymbol{x}^{(n)} \\ y^{(n)} = y^{(n)}_{true} + \varepsilon^{(n)}_{measure} \qquad \varepsilon^{(n)}_{measure} \sim N(0, \sigma^2_m) \\ \text{where} \\ e^{(n)} = \begin{cases} r^{(n)} - y^{(n)}_{true}, \text{ not an error-clamp trial} \\ e^{(n)}_c, \text{ error-clamp trial} \end{cases}\end{gathered} \tag{A2.3}$$

Here, σ^2_m represents the variance of our measurement. Note for the EM algorithm, we did not include this term and assumed that there was no measurement noise in our system [i.e., $y^{(n)} = y^{(n)}_{true}$]. To reiterate, *Eq. A2.3* describes a system where the underlying learning process is deterministic. The only noise in our model is that in the experimental measurement of the behavior, which does not affect the true behavior of the learner.

To prove that the MLE for this system satisfies the least-squares optimization in *Eq. A2.2*, we must compute its incomplete likelihood function. For this, we consider the factored form of the incomplete likelihood function of *Eq. A1.22*. To specify this form, we must compute the conditional means and variances described in *Eq. A1.23*. Critically, for our system in *Eq. A2.3*, the measured behavior of the subject is corrupted by a measurement noise that is independent on every trial. Therefore, all of the measured behaviors are independently distributed and *Eq. A1.23* simplifies to $E[y^{(n)}|\theta,\{y\}_1^{n-1}] = E[y^{(n)}|\theta]$ and $\text{var}[y^{(n)}|\theta,\{y\}_1^{n-1}] = \text{var}[y^{(n)}|\theta] = \sigma^2_m$. Applying these conditional means and variances to *Eq. A1.25* yields the following form of the incomplete log-likelihood:

$$\begin{aligned}\log_e\left(L\left(\{y\}_1^N \middle| \theta\right)\right) = &-\frac{1}{2\sigma^2_m}\sum_{n=1}^{N}\left(y^{(n)} - E\left[y^{(n)}|\theta\right]\right)^2 - \frac{N}{2}\log_e\left(\sigma^2_m\right) \\ &- \frac{N}{2}\log_e(2\pi)\end{aligned} \tag{A2.4}$$

Note that $E[y^{(n)}|\theta]$ in *Eq. A2.4* is also equal to $\hat{y}^{(n)}$ in the objective function for LMSE (*Eq. A2.2*). Furthermore, note that maximization of *Eq. A2.4* implies minimization of the quantity $\sum_{n=1}^{N}(y^{(n)} - E[y^{(n)}|\theta])^2$. This is the same quantity minimized by the LMSE algorithm in *Eq. A2.2*. Therefore, the parameter set that solves our least-squares algorithm also maximizes the likelihood of a system without state and motor noise.

APPENDIX 3: MULTIPLE TARGET STATE-SPACE MODEL OF LEARNING

In the state-space model outlined in METHODS, we assume that the learner possesses a single slow and fast state that are consistently engaged in the learning process on each trial. This model is most compatible with data sets where the same movement target is provided to the subject on each trial. Here, we consider a more complicated paradigm where the learner is presented with a sequence of trials where she is instructed to make a movement toward a target that can change from one trial to the next. We assume that the number of targets is finite, and equal to G. Note that for our visuomotor rotation paradigm, $G = 8$. We will now provide the necessary modifications to the equations described in METHODS for this multiple target model.

As for the single target case, the learner adjusts her movement toward each target according to her estimate of the perturbation. As before, this estimate depends on the state of a slow and fast adaptive process. The learner has a separate fast and slow state for each target, all of which are included in the state vector $\boldsymbol{x}^{(n)} \in \mathbb{R}^{2Gx1} = [s^{(n)}_1 \ \cdots \ s^{(n)}_G \ f^{(n)}_1 \ \cdots \ f^{(n)}_G]^T$. Here, s_k is the slow state for target k and f_k is the fast state for target k. The learner's estimate of the perturbation depends on this state vector according to the following equation:

$$\begin{gathered}\hat{r}^{(n)} = \boldsymbol{c}^T\boldsymbol{x}^{(n)} \\ \text{where} \quad \boldsymbol{c} \in \mathbb{R}^{2Gx1} = \left[c^{(n)}_1 \ \cdots \ c^{(n)}_G \ c^{(n)}_1 \ \cdots \ c^{(n)}_G\right]^T \\ \text{and} \quad c^{(n)}_k = \begin{cases} 1, & \text{target k is presented on trial n} \\ 0, & \text{otherwise.} \end{cases}\end{gathered} \tag{A3.1}$$

Eq. A3.1 formalizes the selection of the appropriate fast and slow states for the current target. On each trial it will contain two entries that are equal to one (all others are zero). In this way, the estimate of the perturbation is the sum of the fast and slow states that correspond to the target presented on trial n.

The evolution of the fast and slow states from one trial to the next depends both on forgetting and error-based learning according to *Eq. A3.2*:

$$\boldsymbol{x}^{(n+1)} = A\boldsymbol{x}^{(n)} + C^{(n)}be^{(n)} + C^{(n)}\varepsilon^{(n)}_{learn} + \varepsilon^{(n)}_{base}. \tag{A3.2}$$

In the no-generalization model analysis described in RESULTS, we constrained the parameters of our modified state update equation, with the following four assumptions. Of course, any of these restrictions could be relaxed to allow for a more general model. First, the fast states of learning all exhibit the same forgetting properties, as do the slow states (i.e., they have the same slow and fast retention factors). Second, all fast states of learning learn at the same rate, as do the slow states (i.e., they have the same slow and fast error sensitivities). Third, there is no generalization of learning across targets. Therefore, the error experienced on trial n only engages the fast and slow processes that correspond to the target presented on trial n. Finally, all states experience a baseline level of state noise on each trial. The fast and slow states engaged in the learning process experience amplified noise on that trial.

To enforce all these rules, we made the following constraints on the parameters in *Eq. A3.2*. To enforce that each state forgets from one trial to the next, with no difference in forgetting across targets, we represented A as a 2G by 2G diagonal matrix of the form $A \in \mathbb{R}^{2Gx2G} = diag(a_s,\cdots,a_s,a_f,\cdots,a_f)$. To enforce that all slow states and all fast states learn at the same rate, we used a common error sensitivity vector $\boldsymbol{b} \in \mathbb{R}^{2x1} = [b_s \ \ b_f]^T$.

To account for generalization, we introduced the selector matrix $C^{(n)}$. Suppose that target k is visited on trial n. For our no-generalization model, the selector matrix causes only the slow and fast states corresponding to target k to learn on trial n. In this case, the selector matrix is a $2G \times 2$ matrix of the following form:

$$C^{(n)} \in \mathbb{R}^{2Gx2} = \begin{bmatrix} c^{(n)}_1 & \cdots & c^{(n)}_G & 0 & \cdots & 0 \\ 0 & \cdots & 0 & c^{(n)}_1 & \cdots & c^{(n)}_G \end{bmatrix}^T.$$

It is critical to note that this selector matrix could be used to encode generalization of learning across targets. In the case of generalization, the modeler could replace the zero elements in matrix C to cause a single error to differentially affect the update of each state.

Finally, to enforce our rule concerning the variance of the state update process, we separated state noise into two terms in *Eq. A3.2*: a baseline state noise that affects the evolution of each state on every trial $\varepsilon_{base}^{(n)}$ and an additional noise source that affects only the states that experienced learning on a particular trial $\varepsilon_{learn}^{(n)}$ (i.e., the states for the target presented on trial n). To enforce the latter property, we multiplied the learning noise by the selector matrix. Given the dimensions of our system, our baseline and learning noises had the following form; the baseline noise $\varepsilon_{base}^{(n)} \in \mathbb{R}^{2Gx1}$ was unbiased with mean $[0 \ \cdots \ 0]^T$ and covariance matrix $B \in \mathbb{R}^{2Gx2G} = \sigma_{base}^2 \mathrm{I}_{2Gx2G}$, where σ_{base}^2 represents a common baseline variance for all states. The learning noise $\varepsilon_{learn}^{(n)}$ was unbiased with mean $[0\ 0]^T$ and variance-covariance matrix $L = \sigma_{learn}^2 \mathrm{I}_{2x2}$, where σ_{learn}^2 represents the state update variance associated with learning.

We can account for set breaks with the decay factor introduced in our model described in *Eq. 10*. With the introduction of set breaks, our two-state model can now be represented as the following system of state-space equations that account for both error clamp trials, set breaks, and multiple targets:

$$
\begin{aligned}
&\boldsymbol{x}^{(n+1)} = A^{(n)}\boldsymbol{x}^{(n)} + \boldsymbol{b}^{(n)}e^{(n)} + \varepsilon_x^{(n)} \qquad \varepsilon_x^{(n)} \sim N\left(\begin{bmatrix}0 & \cdots & 0\end{bmatrix}^T, Q^{(n)}\right)\\
&y^{(n)} = \boldsymbol{c}^T\boldsymbol{x}^{(n)} + \varepsilon_u^{(n)} \qquad \varepsilon_u^{(n)} \sim N\left(0, \sigma_u^2\right)\\
&A^{(n)} = \begin{cases} A & \text{no set break} \\ A^{d+1} & \text{set break} \end{cases} \qquad Q^{(n)} = \begin{cases} C^{(n)}L\,C^{(n)T} + \mathrm{B} & \text{no set break} \\ A^d\left(C^{(n)}LC^{(n)T} + \mathrm{B}\right)A^{dT} & \text{set break} \end{cases}\\
&\boldsymbol{b}^{(n)} = \begin{cases} C^{(n)}\boldsymbol{b} & \text{no set break} \\ A^dC^{(n)}\boldsymbol{b} & \text{set break} \end{cases} \qquad e^{(n)} = \begin{cases} r^{(n)} - y^{(n)} & \text{not an error clamp trial} \\ e_c^{(n)} & \text{error clamp trial} \end{cases}
\end{aligned}
\tag{A3.3}
$$

GRANTS

This work was supported Grant Nos. NS-095706 from the National Institute of Neurological Disorders and Stroke (NINDS) to S. T. Albert, NS078311 from the NINDS to R. Shadmehr, and N00014-15-1-2312 from the Office of Naval Research to R. Shadmehr.

DISCLOSURES

No conflicts of interest, financial or otherwise, are declared by the authors.

AUTHOR CONTRIBUTIONS

S.T.A. and R.S. conceived and designed research; S.T.A. performed experiments; S.T.A. analyzed data; S.T.A. and R.S. interpreted results of experiments; S.T.A. and R.S. prepared figures; S.T.A. and R.S. drafted manuscript; S.T.A. and R.S. edited and revised manuscript; S.T.A. and R.S. approved final version of manuscript.

REFERENCES

Berniker M, Kording KP. Estimating the relevance of world disturbances to explain savings, interference and long-term motor adaptation effects. *PLOS Comput Biol* 7: e1002210, 2011. doi:10.1371/journal.pcbi.1002210.

Brennan AE, Smith MA. The decay of motor memories is independent of context change detection. *PLOS Comput Biol* 11: e1004278, 2015. doi:10.1371/journal.pcbi.1004278.

Brennan AE, Smith MA. Motor memories are confined to distinct channels with differing stability across time and experience [Poster]. Society for Neuroscience Annual Meeting, San Diego, CA, November 12–16, 2016.

Chen-Harris H, Joiner WM, Ethier V, Zee DS, Shadmehr R. Adaptive control of saccades via internal feedback. *J Neurosci* 28: 2804–2813, 2008. doi:10.1523/JNEUROSCI.5300-07.2008.

Cheng S, Sabes PN. Modeling sensorimotor learning with linear dynamical systems. *Neural Comput* 18: 760–793, 2006. doi:10.1162/neco.2006.18.4.760.

Colagiorgio P, Bertolini G, Bockisch CJ, Straumann D, Ramat S. Multiple timescales in the adaptation of the rotational VOR. *J Neurophysiol* 113: 3130–3142, 2015. doi:10.1152/jn.00688.2014.

Criscimagna-Hemminger SE, Shadmehr R. Consolidation patterns of human motor memory. *J Neurosci* 28: 9610–9618, 2008. doi:10.1523/JNEUROSCI.3071-08.2008.

Dempster AP, Laird NM, Rubin DB. Maximum likelihood from incomplete data via the EM algorithm. *J R Stat Soc B* 39: 1–38, 1977.

Donchin O, Francis JT, Shadmehr R. Quantifying generalization from trial-by-trial behavior of adaptive systems that learn with basis functions: theory and experiments in human motor control. *J Neurosci* 23: 9032–9045, 2003.

Ethier V, Zee DS, Shadmehr R. Spontaneous recovery of motor memory during saccade adaptation. *J Neurophysiol* 99: 2577–2583, 2008. doi:10.1152/jn.00015.2008.

Fujita H, Sugihara I. Branching patterns of olivocerebellar axons in relation to the compartmental organization of the cerebellum. *Front Neural Circuits* 7: 3, 2013. doi:10.3389/fncir.2013.00003.

Galea JM, Mallia E, Rothwell J, Diedrichsen J. The dissociable effects of punishment and reward on motor learning. *Nat Neurosci* 18: 597–602, 2015. doi:10.1038/nn.3956.

Galea JM, Vazquez A, Pasricha N, de Xivry JJ, Celnik P. Dissociating the roles of the cerebellum and motor cortex during adaptive learning: the motor cortex retains what the cerebellum learns. *Cereb Cortex* 21: 1761–1770, 2011. doi:10.1093/cercor/bhq246.

Ghahramani Z, Hinton GE. *Parameter Estimation For Linear Dynamical Systems. Technical Report CRG-TR-96-2*. http://mlg.eng.cam.ac.uk/zoubin/papers/tr-96-2.pdf. Department of Computer Science, University of Toronto [22 Feb 1996].

Hadjiosif AM, Smith MA. Savings is restricted to the temorally labile component of motor adaptation. Translational and Computational Motor Control Conference, San Diego, CA, November 2013.

Haith AM, Huberdeau DM, Krakauer JW. The influence of movement preparation time on the expression of visuomotor learning and savings. *J Neurosci* 35: 5109–5117, 2015. doi:10.1523/JNEUROSCI.3869-14.2015.

Hanajima R, Shadmehr R, Ohminami S, Tsutsumi R, Shirota Y, Shimizu T, Tanaka N, Terao Y, Tsuji S, Ugawa Y, Uchimura M, Inoue M, Kitazawa S. Modulation of error-sensitivity during a prism adaptation task in people with cerebellar degeneration. *J Neurophysiol* 114: 2460–2471, 2015. doi:10.1152/jn.00145.2015.

Herzfeld DJ, Kojima Y, Soetedjo R, Shadmehr R. Encoding of action by the Purkinje cells of the cerebellum. *Nature* 526: 439–442, 2015. doi:10.1038/nature15693.

Herzfeld DJ, Pastor D, Haith AM, Rossetti Y, Shadmehr R, O'Shea J. Contributions of the cerebellum and the motor cortex to acquisition and retention of motor memories. *Neuroimage* 98: 147–158, 2014a. doi:10.1016/j.neuroimage.2014.04.076.

Herzfeld DJ, Vaswani PA, Marko MK, Shadmehr R. A memory of errors in sensorimotor learning. *Science* 345: 1349–1353, 2014b. doi:10.1126/science.1253138.

Inoue M, Uchimura M, Karibe A, O'Shea J, Rossetti Y, Kitazawa S. Three timescales in prism adaptation. *J Neurophysiol* 113: 328–338, 2015. doi:10.1152/jn.00803.2013.

Joiner WM, Smith MA. Long-term retention explained by a model of short-term learning in the adaptive control of reaching. *J Neurophysiol* 100: 2948–2955, 2008. doi:10.1152/jn.90706.2008.

Kim S, Ogawa K, Lv J, Schweighofer N, Imamizu H. Neural substrates related to motor memory with multiple timescales in sensorimotor adaptation. *PLoS Biol* 13: e1002312, 2015. doi:10.1371/journal.pbio.1002312.

Kitago T, Ryan SL, Mazzoni P, Krakauer JW, Haith AM. Unlearning versus savings in visuomotor adaptation: comparing effects of washout, passage of time, and removal of errors on motor memory. *Front Hum Neurosci* 7: 307, 2013. doi:10.3389/fnhum.2013.00307.

Kojima Y, Iwamoto Y, Yoshida K. Memory of learning facilitates saccadic adaptation in the monkey. *J Neurosci* 24: 7531–7539, 2004. doi:10.1523/JNEUROSCI.1741-04.2004.

Kojima Y, Soetedjo R, Fuchs AF. Changes in simple spike activity of some Purkinje cells in the oculomotor vermis during saccade adaptation are appropriate to participate in motor learning. *J Neurosci* 30: 3715–3727, 2010. doi:10.1523/JNEUROSCI.4953-09.2010.

Kording KP, Tenenbaum JB, Shadmehr R. The dynamics of memory as a consequence of optimal adaptation to a changing body. *Nat Neurosci* 10: 779–786, 2007. doi:10.1038/nn1901.

Krakauer JW, Pine ZM, Ghilardi MF, Ghez C. Learning of visuomotor transformations for vectorial planning of reaching trajectories. *J Neurosci* 20: 8916–8924, 2000.

Leow LA, de Rugy A, Marinovic W, Riek S, Carroll TJ. Savings for visuomotor adaptation require prior history of error, not prior repetition of successful actions. *J Neurophysiol* 116: 1603–1614, 2016. doi:10.1152/jn.01055.2015.

McDougle SD, Bond KM, Taylor JA. Explicit and implicit processes constitute the fast and slow processes of sensorimotor learning. *J Neurosci* 35: 9568–9579, 2015. doi:10.1523/JNEUROSCI.5061-14.2015.

Nettleton D. Convergence properties of the EM algorithm in constrained parameter spaces. *Can J Stat* 27: 639–648, 1999. doi:10.2307/3316118.

Pekny SE, Criscimagna-Hemminger SE, Shadmehr R. Protection and expression of human motor memories. *J Neurosci* 31: 13829–13839, 2011. doi:10.1523/JNEUROSCI.1704-11.2011.

Pekny SE, Shadmehr R. Optimizing effort: increased efficiency of motor memory with time away from practice. *J Neurophysiol* 113: 445–454, 2015. doi:10.1152/jn.00638.2014.

Scheidt RA, Dingwell JB, Mussa-Ivaldi FA. Learning to move amid uncertainty. *J Neurophysiol* 86: 971–985, 2001. doi:10.1152/jn.2001.86.2.971.

Scheidt RA, Reinkensmeyer DJ, Conditt MA, Rymer WZ, Mussa-Ivaldi FA. Persistence of motor adaptation during constrained, multi-joint, arm movements. *J Neurophysiol* 84: 853–862, 2000. doi:10.1152/jn.2000.84.2.853.

Shadmehr R. Learning to predict and control the physics of our movements. *J Neurosci* 37: 1663–1671, 2017. doi:10.1523/JNEUROSCI.1675-16.2016.

Shadmehr R, Mussa-Ivaldi S. *Biological Learning and Control: How the Brain Builds Representations, Predicts Events, and Makes Decisions*. Cambridge, MA: MIT Press, 2012. doi:10.7551/mitpress/9780262016964.001.0001

Smith MA, Ghazizadeh A, Shadmehr R. Interacting adaptive processes with different timescales underlie short-term motor learning. *PLoS Biol* 4: e179, 2006. doi:10.1371/journal.pbio.0040179.

Smith MA, Shadmehr R. Modulation of the rate of error-dependent learning by statistical properties of the task. Advances in Computational Motor Control, Washington, DC, November 14, 2004.

Srimal R, Diedrichsen J, Ryklin EB, Curtis CE. Obligatory adaptation of saccade gains. *J Neurophysiol* 99: 1554–1558, 2008. doi:10.1152/jn.01024.2007.

Stollhoff N, Menzel R, Eisenhardt D. Spontaneous recovery from extinction depends on the reconsolidation of the acquisition memory in an appetitive learning paradigm in the honeybee (*Apis mellifera*). *J Neurosci* 25: 4485–4492, 2005. doi:10.1523/JNEUROSCI.0117-05.2005.

Tanaka H, Krakauer JW, Sejnowski TJ. Generalization and multirate models of motor adaptation. *Neural Comput* 24: 939–966, 2012. doi:10.1162/NECO_a_00262.

Taylor JA, Krakauer JW, Ivry RB. Explicit and implicit contributions to learning in a sensorimotor adaptation task. *J Neurosci* 34: 3023–3032, 2014. doi:10.1523/JNEUROSCI.3619-13.2014.

Thoroughman KA, Shadmehr R. Learning of action through adaptive combination of motor primitives. *Nature* 407: 742–747, 2000. doi:10.1038/35037588.

Trewartha KM, Garcia A, Wolpert DM, Flanagan JR. Fast but fleeting: adaptive motor learning processes associated with aging and cognitive decline. *J Neurosci* 34: 13411–13421, 2014. doi:10.1523/JNEUROSCI.1489-14.2014.

van Beers RJ, Brenner E, Smeets JBJ. Random walk of motor planning in task-irrelevant dimensions. *J Neurophysiol* 109: 969–977, 2013. doi:10.1152/jn.00706.2012.

Vaswani PA, Shadmehr R. Decay of motor memories in the absence of error. *J Neurosci* 33: 7700–7709, 2013. doi:10.1523/JNEUROSCI.0124-13.2013.

Vaswani PA, Shmuelof L, Haith AM, Delnicki RJ, Huang VS, Mazzoni P, Shadmehr R, Krakauer JW. Persistent residual errors in motor adaptation tasks: reversion to baseline and exploratory escape. *J Neurosci* 35: 6969–6977, 2015. doi:10.1523/JNEUROSCI.2656-14.2015.

Wei K, Körding K. Uncertainty of feedback and state estimation determines the speed of motor adaptation. *Front Comput Neurosci* 4: 11, 2010. doi:10.3389/fncom.2010.00011.

Wu CFJ. On the convergence properties of the EM algorithm. *Ann Stat* 11: 95–103, 1983. doi:10.1214/aos/1176346060.

Xu-Wilson M, Chen-Harris H, Zee DS, Shadmehr R. Cerebellar contributions to adaptive control of saccades in humans. *J Neurosci* 29: 12930–12939, 2009. doi:10.1523/JNEUROSCI.3115-09.2009.

Yang Y, Lisberger SG. Role of plasticity at different sites across the time course of cerebellar motor learning. *J Neurosci* 34: 7077–7090, 2014. doi:10.1523/JNEUROSCI.0017-14.2014.