# CT-LFSSM: MULTISCALE CONTINUOUS-TIME STATE-SPACE LEARNING AND FORGETTING IN KNOWLEDGE TRACING

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Knowledge Tracing (KT) aims to model students' evolving knowledge from historical interactions and predict their future performance. However, existing methods often encode irregular temporal gaps as auxiliary inputs rather than explicitly evolving latent knowledge over elapsed time. They also tend to absorb forgetting, response evidence, and learning into a single state transition, making these effects difficult to distinguish. Independently parameterized items further limit statistical sharing across items associated with the same knowledge component. To address these limitations, we propose **CT-LFSSM**, a Continuous-Time Learning–Forgetting State-Space Model. CT-LFSSM represents each knowledge component with fast-, medium-, and slow-timescale latent states. It propagates these states over real elapsed time through personalized exponential decay, with half-lives constrained to increase from the fast to the slow timescale. Before each response, an event-conditioned convex gate aggregates the three states into a current mastery estimate. After observing the response, CT-LFSSM uses the prediction residual to revise the inferred pre-interaction state, while separately consolidating practice-induced learning gains across multiple timescales. This separation assigns distinct update paths to forgetting, response-driven belief revision, and learning. For response prediction, CT-LFSSM further decomposes item effects into shared knowledge-component-level parameters and regularized item-specific residuals. Experiments on four real-world datasets show that CT-LFSSM achieves state-of-the-art performance across all datasets, while remaining robust across varying temporal gaps and student interaction sequence lengths.

## 1 INTRODUCTION

Knowledge Tracing (KT) infers a student's evolving latent knowledge from historical interactions to predict future performance, which is central to intelligent tutoring systems for adaptive content selection and personalized practice (Piech et al., 2015; Zhou et al., 2024). In real educational settings, interactions are asynchronous and unevenly spaced, with consecutive attempts separated by minutes, days, or weeks. Observed responses thus provide only intermittent evidence about a knowledge state that continues to evolve between interactions. Consequently, the inferred state depends not only on what the student practiced but also on when those interactions occurred and how the learning history is distributed over time.

These characteristics have motivated KT research to move beyond interaction-indexed sequence modeling. Recurrent models such as DKT (Piech et al., 2015) learn latent states from interaction sequences, while attention-based models such as AKT (Ghosh et al., 2020) capture dependencies between historical and current interactions with temporal decay and psychometric structure. Subsequent work has modeled learning processes (Shen et al., 2021), continuous-time forgetting (Ma et al., 2022), question-centric representations (Chen et al., 2023), and generalization across sequence lengths (Li et al., 2024). More recent studies consider prerequisite relations and learner traits (Zhou et al., 2024), dynamic student–question–concept interactions (Cheng et al., 2024), and uncertainty in latent representations (Cheng et al., 2025).



Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

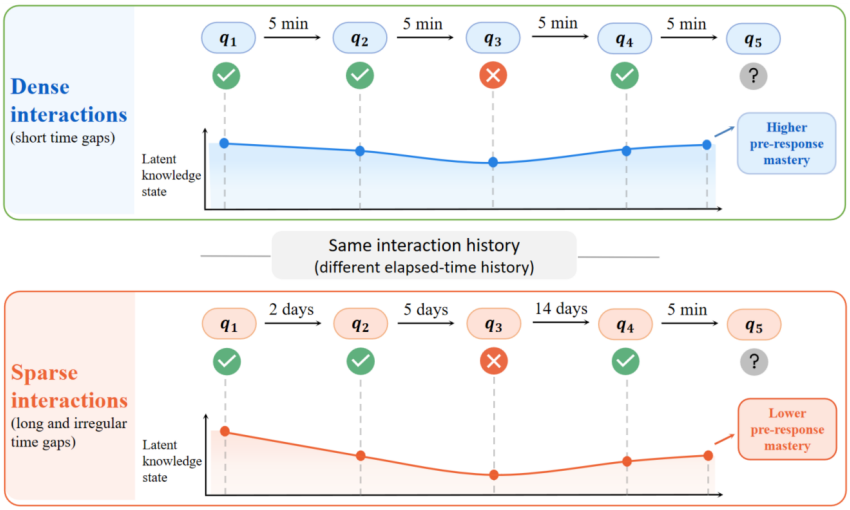



Figure 1: Illustration of irregular temporal gaps. Two students share the same item and response history but experience different elapsed-time patterns before the next interaction. An interaction-index view treats them as equivalent, yet their pre-response knowledge states can differ because forgetting continues between observed events. Trajectories are schematic.

Nevertheless, two fundamental issues remain unresolved. **First, the temporal evolution of latent knowledge is often insufficiently separated from event-driven state changes.** Educational interactions provide only discrete observations of a process that continues during interaction-free periods. Representing elapsed time as an auxiliary input or evolving states by interaction order fails to explicitly model how knowledge evolves during such intervals. Moreover, a single temporal trajectory obscures the coexistence of short-lived and persistent knowledge components, whose decay characteristics may differ across students and knowledge components.

**Second, distinct sources of state change and response variation can become entangled.** An observed response provides evidence about knowledge immediately before an interaction, whereas the practice opportunity itself may alter knowledge after that interaction. Absorbing these effects into a single unrestricted transition obscures whether a state change reflects revised inference or newly acquired knowledge. A related ambiguity arises at the observation level: independently parameterized items can be poorly estimated when infrequent and fail to exploit shared structure among items associated with the same knowledge component, while treating all such items identically ignores item-specific variations.

To address these limitations, we propose **CT-LFSSM**, a Continuous-Time Learning–Forgetting State-Space Model. For each knowledge component, CT-LFSSM maintains fast-, medium-, and slow-timescale latent states and propagates them over real elapsed time through personalized exponential decay, with half-lives constrained to increase from the fast to the slow timescale. Before each response, an event-conditioned convex gate aggregates the three states into a current mastery estimate. After observing the response, CT-LFSSM uses the prediction residual to revise the inferred pre-interaction state, while separately consolidating practice-induced learning gains across multiple timescales. This separation assigns distinct update paths to forgetting, response-driven belief revision, and learning, and the model follows a strict predict–observe–update order. For response prediction, CT-LFSSM further decomposes item effects into shared knowledge-component-level parameters and regularized item-specific residuals.

The key contributions of this paper are summarized as follows:

- We formulate KT as a continuous-time, multi-timescale state-space process where each knowledge component is represented by fast-, medium-, and slow-timescale latent states with personalized exponential decay, whose half-lives are constrained to increase from the fast to the slow timescale, and integrated via an event-conditioned convex gate.

- We develop a predict–observe–update mechanism that separates response-driven belief revision from practice-induced learning. Evidence revision and multi-timescale consolidation assign distinct roles while maintaining strict temporal ordering.
- We introduce a hierarchical 4PL-style item-response formulation combining knowledge-component-level sharing with regularized item-specific residuals. On four real-world datasets, CT-LFSSM achieves state-of-the-art performance, improving over the strongest baselines by $0.35$–$1.25$ percentage points in AUC and $0.30$–$1.14$ percentage points in accuracy, while remaining robust across temporal gaps and student interaction sequence lengths.

## 2 Related Work

### 2.1 Knowledge Tracing Models

Knowledge tracing has evolved from probabilistic mastery estimation to increasingly expressive neural representations. Bayesian Knowledge Tracing (BKT) models knowledge-component mastery as a latent probabilistic state (Corbett & Anderson, 1994), while Deep Knowledge Tracing (DKT) learns student states directly from interaction sequences using recurrent neural networks (Piech et al., 2015). Dynamic Key-Value Memory Networks (DKVMN) introduce concept-indexed memory (Zhang et al., 2017). Attention-based models subsequently improve the retrieval and representation of historical interactions: AKT combines context-aware attention with temporal decay and Rasch-based regularization (Ghosh et al., 2020), simpleKT demonstrates the effectiveness of standard attention with explicit question-specific variation (Liu et al., 2023b), and QIKT introduces question-centric cognitive representations with an item-response-theory-based prediction layer (Chen et al., 2023). More recent work incorporates learner traits and prerequisite relations (Zhou et al., 2024), dynamic student–question–concept interactions (Cheng et al., 2024), and uncertainty-aware latent states (Cheng et al., 2025).

### 2.2 Temporal and Continuous-Time Knowledge Modeling

A major line of KT research focuses on learning and forgetting over time. LPKT explicitly models learning gains and forgetting from interaction-dependent temporal information (Shen et al., 2021). HawkesKT captures temporally decaying cross-effects among knowledge components through a Hawkes-process formulation (Wang et al., 2021), while CT-NCM represents learning events together with continuous-time forgetting dynamics (Ma et al., 2022). Within attention-based KT, FoLiBi introduces a forgetting-aware linear bias (Im et al., 2023), and LefoKT uses relative forgetting attention to model heterogeneous forgetting patterns (Bai et al., 2025). Recent work has also explored temporal heterogeneity at multiple scales: DMKT jointly models long-term static and short-term dynamic cognitive states (Li et al., 2025), FlucKT separates long-term trends from short-term fluctuations (Hou et al., 2025), and TFKT decomposes knowledge states into multiple frequency components (Huang et al., 2026). Inspired by these studies, CT-LFSSM combines multi-timescale continuous-time states, personalized half-lives, and knowledge-component-level parameter sharing with regularized item-specific residuals.

## 3 Preliminaries

### 3.1 Problem Setting

Let $\mathcal{U}$, $\mathcal{Q}$, and $\mathcal{K}$ denote the sets of students, items, and knowledge components (KCs), with $K = |\mathcal{K}|$. Item–KC relations are represented by a binary matrix $\boldsymbol{A}$, where $A_{qk} = 1$ if item $q$ involves KC $k$.

For student $u$ with $T_u$ interactions, let $q_{u,n}$, $\tau_{u,n}$, and $y_{u,n} \in \{0, 1\}$ denote the item, timestamp, and response at interaction $n$. The interaction sequence and elapsed time are

$$\mathcal{S}_u = \left\{(q_{u,n}, \tau_{u,n}, y_{u,n})\right\}_{n=1}^{T_u}, \qquad \Delta\tau_{u,n} = \tau_{u,n} - \tau_{u,n-1} \geq 0. \tag{1}$$

Let $\mathcal{H}_{u,n-1} = \{(q_{u,j}, \tau_{u,j}, y_{u,j})\}_{j=1}^{n-1}$ denote the history available before interaction $n$. Knowledge tracing predicts

$$p_{u,n} = \Pr(y_{u,n} = 1 \mid \mathcal{H}_{u,n-1}, q_{u,n}, \tau_{u,n}), \qquad \epsilon_{u,n} = y_{u,n} - p_{u,n}, \tag{2}$$

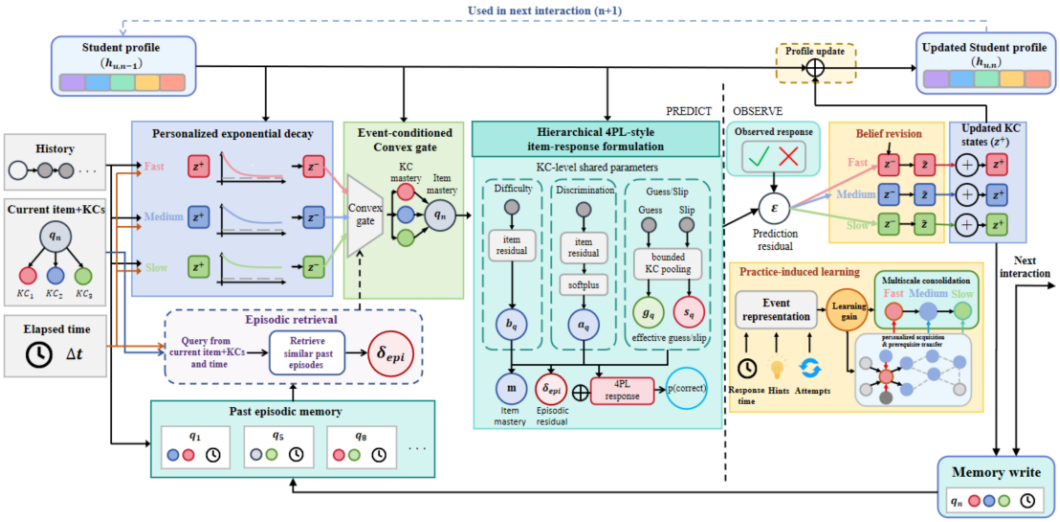



Figure 2: Overview of CT-LFSSM. Fast-, medium-, and slow-timescale latent states undergo personalized exponential decay and are combined by an event-conditioned convex gate. Past episodic information contributes to the observation logit, and a hierarchical 4PL-style formulation produces the prediction. After observation, belief revision and practice-induced learning update the KC states, followed by profile and memory updates for future interactions.

where $\epsilon_{u,n}$ is the prediction residual and becomes available only after $y_{u,n}$ is observed.

### 3.2 Background

For a latent state $z(\tau)$, let $\lambda > 0$ denote its exponential decay rate and $H = \log 2/\lambda > 0$ the corresponding half-life. Over an interaction-free interval $\Delta\tau$,

$$z(\tau + \Delta\tau) = z(\tau)\exp(-\lambda\Delta\tau) = z(\tau)2^{-\Delta\tau/H}. \tag{3}$$

Thus a smaller half-life corresponds to faster decay.

Let $\sigma(x) = 1/(1+\exp(-x))$ denote the logistic function. For the current item $q = q_{u,n}$, let $m_{u,n}$ denote item-relevant mastery, and let $b_q$, $a_q > 0$, $g_q$, and $s_q$ denote difficulty, discrimination, effective guessing probability, and effective slipping probability. A four-parameter logistic (4PL) response model is

$$p^{\mathrm{IRT}}_{u,n} = g_q + (1 - g_q - s_q)\sigma(a_q(m_{u,n} - b_q))\,. \tag{4}$$

CT-LFSSM retains this response form while sharing item parameters through their associated KCs.

## 4 Method

CT-LFSSM represents each KC with fast-, medium-, and slow-timescale latent states that evolve over real elapsed time. Figure 2 summarizes the predict–observe–update process. Before observing the current response, the model applies personalized exponential decay, combines the three timescales through an event-conditioned convex gate, retrieves relevant past episodic information, and predicts $p_{u,n}$. After observing $y_{u,n}$, the prediction residual drives response-driven belief revision followed by practice-induced learning and multi-timescale consolidation.

### 4.1 Multi-Timescale State Evolution

A single retention rate cannot simultaneously capture knowledge that decays within hours and knowledge that persists for weeks. CT-LFSSM therefore maintains three latent states per KC, each evolving at its own rate. Let $\mathcal{R} = \{\mathrm{f}, \mathrm{m}, \mathrm{s}\}$ and let $z^{-}_{u,n,k,r}$ and $z^{+}_{u,n,k,r}$ denote the pre- and post-response states for student $u$, KC $k$, and timescale $r$. Collecting all states gives

$$\boldsymbol{Z}^{\pm}_{u,n} = \left[z^{\pm}_{u,n,k,r}\right]_{k\in\mathcal{K},\, r\in\mathcal{R}} \in [0,1]^{K\times 3}, \qquad \mathcal{X}_{u,n-1} = \left(\boldsymbol{Z}^{+}_{u,n-1}, \boldsymbol{h}_{u,n-1}, \mathcal{M}_{u,n-1}\right), \tag{5}$$

where $\boldsymbol{h}_{u,n-1}$ is a student profile and $\mathcal{M}_{u,n-1}$ stores past episodic events. The profile is updated by a GRUCell only after each response, so $\boldsymbol{h}_{u,n-1}$ contains no information from interaction $n$.

Three scales are useful only if their roles cannot be exchanged: otherwise they could collapse into three copies of the same scale. To prevent this, base half-lives are parameterized through positive log-space increments:

$$\begin{aligned} L_{k,\mathrm{f}} &= \theta_k^H, \qquad L_{k,\mathrm{m}} = L_{k,\mathrm{f}} + \mathrm{softplus}(\eta_{k,\mathrm{m}}^H), \\ L_{k,\mathrm{s}} &= L_{k,\mathrm{m}} + \mathrm{softplus}(\eta_{k,\mathrm{s}}^H), \qquad H_{k,r}^{(0)} = \exp(L_{k,r}). \end{aligned} \tag{6}$$

Because $\mathrm{softplus}(\cdot) > 0$, the fast-to-slow ordering $H_{k,\mathrm{f}}^{(0)} < H_{k,\mathrm{m}}^{(0)} < H_{k,\mathrm{s}}^{(0)}$ is a property of the parameterization rather than a soft preference. Details are in Appendix A.2.

Retention should also reflect who the student is and which KC is practiced. Let $f_H$ denote a learned student–KC interaction and $\kappa_H > 0$ bound its log-scale effect:

$$\phi_{u,n,k} = \exp[\kappa_H \tanh(f_H(\boldsymbol{h}_{u,n-1}, k))], \qquad H_{u,n,k,r} = H_{k,r}^{(0)} \phi_{u,n,k}. \tag{7}$$

Because the same positive factor multiplies all three base half-lives, personalization shifts retention while preserving the fast-to-slow ordering. Implementation is in Appendix A.3.

Between interactions, knowledge changes because time passes rather than because a new event occurs, so the pre-response state is obtained by personalized exponential decay:

$$z_{u,n,k,r}^{-} = z_{u,n-1,k,r}^{+} 2^{-\Delta\tau_{u,n}/H_{u,n,k,r}}, \qquad k \in \mathcal{K}, \quad r \in \mathcal{R}. \tag{8}$$

Elapsed time thus acts directly on the latent state instead of entering only as an auxiliary feature.

### 4.2 Predict–Observe–Update Process

Prediction must rely only on information available before the response, while learning must use the response itself. Merging them would obscure which part of a state change reflects revised inference and which reflects new knowledge. CT-LFSSM therefore follows *distinct update paths and strict execution ordering*.

#### 4.2.1 Pre-Response Readout

The relative informativeness of the three retention scales depends on the elapsed time: shortly after an interaction all three states are similar, whereas after a long gap the fast state has largely decayed while the slow state remains informative. A fixed mixing is therefore suboptimal. Let $\boldsymbol{e}_{q_{u,n}}$, $\boldsymbol{c}_{q_{u,n}}$, $\boldsymbol{t}_{u,n}$ denote the item, KC-context, and elapsed-time representations, and let $f_\omega$ denote the learned scale-gating function. The *event-conditioned convex gate* is

$$(\omega_{u,n,r})_{r\in\mathcal{R}} = \mathrm{softmax}\big(f_\omega(\boldsymbol{h}_{u,n-1}, \boldsymbol{e}_{q_{u,n}}, \boldsymbol{c}_{q_{u,n}}, \boldsymbol{t}_{u,n})\big), \qquad \sum_{r\in\mathcal{R}} \omega_{u,n,r} = 1. \tag{9}$$

The softmax constraint keeps the readout a convex combination, so KC-level mastery cannot exceed the range spanned by the three states. Representation construction is in Appendix A.3. With a KC-level bias $\nu_k$,

$$m_{u,n,k}^{\mathrm{KC}} = \sum_{r\in\mathcal{R}} \omega_{u,n,r} z_{u,n,k,r}^{-} + \nu_k. \tag{10}$$

#### 4.2.2 Hierarchical 4PL-Style Observation Model

Response probability depends on student mastery and on item characteristics. Estimating item parameters independently would give unreliable estimates for infrequent items, while treating all items identically would ignore item-level differences. CT-LFSSM therefore decomposes each response parameter into a KC-level shared component and a bounded item-specific residual; guessing and slipping, which are estimated from limited data, are pooled at the KC level only.

For item $q$, the normalized direct-KC weight is

$$\pi_{qk} = \frac{A_{qk}}{\sum_{j=1}^{K} A_{qj}}, \qquad \sum_{k=1}^{K} \pi_{qk} = 1. \tag{11}$$

Difficulty and discrimination are

$$b_q = \sum_{k=1}^{K} \pi_{qk} b_k + \delta_q^b, \qquad a_q = 0.1 + \mathrm{softplus}\left( \sum_{k=1}^{K} \pi_{qk} \alpha_k + \delta_q^a \right), \tag{12}$$

where the softplus transform ensures $a_q > 0.1$ and prevents discrimination from reversing the mastery–response relationship. Guessing and slipping follow the same pooling without item-specific residuals:

$$g_q = g_{\max} \sigma\left( \sum_{k=1}^{K} \pi_{qk} \theta_k^{\mathrm{guess}} \right), \qquad s_q = s_{\max} \sigma\left( \sum_{k=1}^{K} \pi_{qk} \theta_k^{\mathrm{slip}} \right), \tag{13}$$

which keeps these two channels from reverting to per-item memorization.

Let $\rho_{u,n,k}^{\mathrm{pred}}$ denote the normalized readout weights, $c_{u,n}^{\mathrm{cog}} > 0$ the cognition multiplier, $\beta_{u,n}$ the student bias, and $\delta_{u,n}^{\mathrm{epi}}$ the bounded episodic residual; parameterizations are in Appendix A.4. Item-level mastery, observation logit, and prediction are

$$\begin{aligned} m_{u,n} &= c_{u,n}^{\mathrm{cog}} \sum_{k=1}^{K} \rho_{u,n,k}^{\mathrm{pred}} m_{u,n,k}^{\mathrm{KC}} + \beta_{u,n}, \\ \ell_{u,n} &= a_{q_{u,n}} \left( m_{u,n} - b_{q_{u,n}} \right) + \delta_{u,n}^{\mathrm{epi}}, \\ p_{u,n} &= g_{q_{u,n}} + \left( 1 - g_{q_{u,n}} - s_{q_{u,n}} \right) \sigma(\ell_{u,n}). \end{aligned} \tag{14}$$

The episodic term enters only the observation logit and never the mastery state, so retrieval cannot silently alter the latent representation. All quantities are computed before observing $y_{u,n}$.

#### 4.2.3 Post-Response State Update

An observed response carries two distinct signals: it indicates that the pre-response state was misestimated, and it provides evidence that practice may have changed the student's knowledge. Conflating them would make the state hard to interpret, so CT-LFSSM applies two separate updates in strict order. After observing $y_{u,n}$, the residual $\epsilon_{u,n} = y_{u,n} - p_{u,n}$ is formed together with a post-response event representation $\boldsymbol{x}_{u,n}^{\mathrm{evt}}$; details are in Appendix A.4.

The first update, *response-driven belief revision*, corrects the pre-response state in logit space. Let $E_{k,r} \geq 0$ denote the KC- and timescale-specific evidence strength, parameterized so that $E_{k,\mathrm{f}} \geq E_{k,\mathrm{m}} \geq E_{k,\mathrm{s}} \geq 0$; let $f_E$ be the evidence-gating head and $\gamma_{u,n}^E = \sigma(f_E(\boldsymbol{x}_{u,n}^{\mathrm{evt}}))$. With $\mathrm{logit}(x) = \log(x/(1-x))$,

$$\tilde{z}_{u,n,k,r} = \sigma\Big( \mathrm{logit}(z_{u,n,k,r}^-) + A_{q_{u,n}k} E_{k,r} \gamma_{u,n}^E \epsilon_{u,n} \Big). \tag{15}$$

Because evidence strength decreases from fast to slow, short-lived components react more to individual responses than long-lived ones. Details are in Appendix A.5.

The second update, *practice-induced learning*, adds knowledge produced by the practice event. Let $f_G$ be the gain head, $G_{\max} > 0$ the maximum event-level gain, and $\chi_{\mathrm{err}} \in (0,1)$ the relative factor after an incorrect response:

$$G_{u,n} = G_{\max} \sigma\big( f_G(\boldsymbol{x}_{u,n}^{\mathrm{evt}}) \big) \left[ y_{u,n} + (1 - y_{u,n}) \chi_{\mathrm{err}} \right]. \tag{16}$$

An incorrect attempt therefore still induces learning, but its outcome factor is bounded below that of a correct response.

Let $c_{u,n}^{\mathrm{m}}, c_{u,n}^{\mathrm{s}} \in (0,1)$ denote the medium and slow consolidation factors, so that $G_{u,n,\mathrm{f}} = G_{u,n}$, $G_{u,n,\mathrm{m}} = G_{u,n} c_{u,n}^{\mathrm{m}}$, $G_{u,n,\mathrm{s}} = G_{u,n} c_{u,n}^{\mathrm{m}} c_{u,n}^{\mathrm{s}}$. Since both factors lie in $(0,1)$, a single event contributes most to the fast scale and least to the slow scale, matching the retention hierarchy. Let $\rho_{u,n,k}^{\mathrm{acq}} \in [0,1]$ denote the acquisition coefficient combining direct acquisition with prerequisite transfer. The post-response state is

$$z_{u,n,k,r}^+ = \tilde{z}_{u,n,k,r} + \rho_{u,n,k}^{\mathrm{acq}} G_{u,n,r} \left( 1 - \tilde{z}_{u,n,k,r} \right). \tag{17}$$

The factor $1 - \tilde{z}_{u,n,k,r}$ saturates the increment near the upper bound, keeping the state inside $[0,1]$. After the update, the profile is refreshed by a GRUCell and the event is appended to memory, so neither operation can influence the current prediction.

### 4.3 Training Objective

The prediction loss alone does not constrain how large post-response changes can be, and a single interaction could dominate the online state. CT-LFSSM therefore adds four light penalties to the binary cross-entropy. With $\mathcal{T}$ the supervised interactions,

$$\mathcal{L}_{\text{pred}} = -\frac{1}{|\mathcal{T}|} \sum_{(u,n)\in\mathcal{T}} \left[ y_{u,n} \log p_{u,n} + (1 - y_{u,n}) \log(1 - p_{u,n}) \right], \tag{18}$$
$$\mathcal{L} = \mathcal{L}_{\text{pred}} + \lambda_{\text{prof}}\mathcal{P}_{\text{prof}} + \lambda_{\text{evid}}\mathcal{P}_{\text{evid}} + \lambda_{\text{gain}}\mathcal{P}_{\text{gain}} + \lambda_{\text{item}}\mathcal{P}_{\text{item}},$$

where the four $\mathcal{P}$ terms penalize profile changes, belief revisions, learning updates, and item-specific difficulty residuals, respectively. The first three keep the online state from being dominated by any single interaction; the last shrinks the difficulty residual toward its KC-pooled component. Exact definitions and coefficients are in Appendix A.6. Theoretical properties, complexity, and the interaction algorithm are in Appendix A.7, and the complete configuration is in Appendix B.1.

## 5 Experiments

We evaluate next-response prediction on four public KT benchmarks (ASSIST2009, ASSIST2012, Algebra2005, and XES3G5M) with AUC as the primary metric and ACC as the secondary metric. Beyond overall predictive performance, our analyses are organized around the two modeling issues motivating CT-LFSSM. First, because the model explicitly evolves multi-timescale knowledge states over real elapsed time, we examine whether its advantage remains stable across different temporal regimes rather than concentrating at a particular horizon. Second, because prediction and post-response state evolution rely on several structured mechanisms, we assess whether the major state-update and prediction components provide complementary contributions to the full model.

For the first three datasets, we build a standardized re-benchmark under a unified pyKT-compatible protocol (Liu et al., 2022): a held-out 20% test set, five train/validation folds on the remaining 80%, minimum sequence length three, maximum input length 200 with question-level rolling-window evaluation, checkpoint selection by validation AUC only, and results reported as the five-fold mean $\pm$ population standard deviation. All baselines for these three datasets are rerun under the same split, model-selection, and test protocol as CT-LFSSM, and the original results of Zhang et al. (2023) are not mixed into our main comparison; for XES3G5M, we follow its official five-fold benchmark and compare against the published results of Liu et al. (2023c). Among the four datasets, ASSIST2012, Algebra2005, and XES3G5M provide real elapsed-time information and are used for time-aware analyses, whereas ASSIST2009 provides interaction order but no reliable wall-clock timestamps, so its temporal input is neutralized with $\Delta\tau = 0$. Dataset statistics, preprocessing, the complete baseline list, training configuration, and the full XES3G5M results are provided in Appendices B.1, B.2, and B.3.

### 5.1 Main Results

**Overall performance.** Table 1 first establishes the overall predictive effectiveness of CT-LFSSM before examining the conditions under which these gains arise. CT-LFSSM achieves the highest mean AUC and ACC on all three re-benchmarked datasets, with the largest margins observed on Algebra2005. The improvement on ASSIST2009 is also informative because its elapsed-time input is neutralized with $\Delta\tau = 0$, indicating that the overall gain is not reducible to direct access to wall-clock information alone. On XES3G5M, CT-LFSSM likewise achieves the best AUC and ACC, improving over the strongest reported baseline IEKT by $0.0035$ in AUC and $0.0064$ in ACC; detailed results are provided in Appendix B.3.

**Robustness to sequence length.** Sequence-length stratification provides a complementary test of whether CT-LFSSM depends on a particular amount of historical context. On ASSIST2012 (Table 2), CT-LFSSM achieves the best AUC in four of the five bins and ranks second in the shortest-history group. The advantage becomes clearer as more interaction history is available, indicating that the model can exploit longer histories without sacrificing performance on short sequences. Full AUC/ACC results on ASSIST2012 and the corresponding ASSIST2009 results are provided in Appendix B.4.

Table 1: Overall performance on ASSIST2012, ASSIST2009, and Algebra2005. Results are mean ± population standard deviation over five folds. Best results are in **bold** and second-best results are underlined.

| | ASSIST2012 | | ASSIST2009 | | Algebra2005 | |
|---|---|---|---|---|---|---|
| Model | AUC | ACC | AUC | ACC | AUC | ACC |
| PFA | 0.7052±0.0005 | 0.7179±0.0003 | 0.7299±0.0007 | 0.7097±0.0005 | 0.7470±0.0012 | 0.7415±0.0016 |
| KTM | 0.7623±0.0002 | 0.7251±0.0006 | 0.7571±0.0004 | 0.7255±0.0002 | 0.7991±0.0008 | 0.7738±0.0006 |
| DAS3H | 0.7645±0.0010 | 0.7279±0.0008 | 0.7583±0.0021 | 0.7269±0.0018 | 0.7999±0.0011 | 0.7831±0.0010 |
| DKT | 0.7316±0.0004 | 0.7344±0.0003 | 0.7516±0.0011 | 0.7232±0.0015 | 0.8030±0.0015 | 0.8012±0.0007 |
| DKVMN | 0.7305±0.0003 | 0.7345±0.0002 | 0.7448±0.0004 | 0.7191±0.0011 | 0.8003±0.0004 | 0.7994±0.0004 |
| SAKT | 0.7070±0.0006 | 0.7240±0.0003 | 0.7314±0.0012 | 0.7106±0.0004 | 0.7836±0.0017 | 0.7939±0.0009 |
| AKT | 0.7796±0.0003 | 0.7552±0.0003 | 0.7802±0.0015 | 0.7322±0.0012 | 0.8123±0.0016 | 0.8050±0.0007 |
| IEKT | 0.7705±0.0005 | 0.7514±0.0004 | 0.7659±0.0015 | 0.7295±0.0016 | 0.8015±0.0028 | 0.8005±0.0016 |
| DIMKT | 0.7681±0.0018 | 0.7506±0.0011 | 0.7637±0.0018 | 0.7286±0.0018 | 0.8069±0.0016 | 0.8012±0.0015 |
| QIKT | 0.7665±0.0003 | 0.7494±0.0001 | 0.7797±0.0019 | 0.7317±0.0004 | 0.8063±0.0012 | 0.8010±0.0012 |
| FoLiBiKT | 0.7768±0.0006 | 0.7545±0.0008 | 0.7789±0.0022 | 0.7326±0.0018 | 0.8171±0.0010 | 0.8020±0.0014 |
| LPKT | 0.7738±0.0005 | 0.7538±0.0012 | 0.7780±0.0022 | 0.7318±0.0014 | 0.8160±0.0005 | 0.8024±0.0005 |
| simpleKT | 0.7748±0.0002 | 0.7517±0.0007 | 0.7711±0.0012 | 0.7287±0.0018 | 0.8179±0.0010 | 0.8054±0.0009 |
| HawkesKT | 0.7481±0.0004 | 0.7409±0.0002 | 0.7266±0.0012 | 0.7044±0.0014 | 0.8080±0.0075 | 0.8013±0.0039 |
| **CT-LFSSM** | **0.7833±0.0005** | **0.7582±0.0003** | **0.7873±0.0019** | **0.7419±0.0014** | **0.8304±0.0005** | **0.8168±0.0016** |

Table 2: Sequence-length-stratified AUC performance on ASSIST2012. Bins are defined by complete student sequence length before windowing. Best results are in **bold** and second-best results are underlined.

| Model | $(0, 10]$ | $(10, 50]$ | $(50, 100]$ | $(100, 200]$ | $(200, +\infty)$ |
|---|---|---|---|---|---|
| PFA | 0.7165±0.0018 | 0.7008±0.0009 | 0.6983±0.0008 | 0.6956±0.0010 | 0.6839±0.0003 |
| KTM | 0.7697±0.0031 | 0.7618±0.0021 | 0.7627±0.0003 | 0.7696±0.0008 | 0.7326±0.0010 |
| DAS3H | 0.7701±0.0008 | 0.7631±0.0010 | 0.7603±0.0010 | 0.7735±0.0003 | 0.7229±0.0007 |
| DKT | 0.7584±0.0009 | 0.7332±0.0005 | 0.7331±0.0004 | 0.7426±0.0003 | 0.7262±0.0004 |
| DKVMN | 0.7533±0.0011 | 0.7295±0.0011 | 0.7304±0.0005 | 0.7418±0.0006 | 0.7257±0.0002 |
| SAKT | 0.7224±0.0019 | 0.7173±0.0005 | 0.7135±0.0010 | 0.7190±0.0011 | 0.6989±0.0004 |
| AKT | **0.7770±0.0019** | 0.7704±0.0004 | 0.7766±0.0004 | 0.7864±0.0005 | 0.7818±0.0005 |
| IEKT | 0.7701±0.0013 | 0.7689±0.0025 | 0.7698±0.0011 | 0.7801±0.0006 | 0.7808±0.0006 |
| DIMKT | 0.7762±0.0018 | 0.7665±0.0013 | 0.7673±0.0015 | 0.7736±0.0017 | 0.7662±0.0021 |
| QIKT | 0.7740±0.0017 | 0.7633±0.0005 | 0.7649±0.0006 | 0.7726±0.0004 | 0.7646±0.0004 |
| FoLiBiKT | 0.7732±0.0022 | 0.7707±0.0007 | 0.7752±0.0004 | 0.7836±0.0004 | 0.7806±0.0009 |
| LPKT | 0.7681±0.0018 | 0.7643±0.0008 | 0.7712±0.0007 | 0.7786±0.0007 | 0.7741±0.0005 |
| simpleKT | 0.7744±0.0007 | 0.7676±0.0004 | 0.7710±0.0004 | 0.7806±0.0004 | 0.7743±0.0004 |
| HawkesKT | 0.7601±0.0010 | 0.7471±0.0006 | 0.7484±0.0010 | 0.7570±0.0004 | 0.7450±0.0008 |
| **CT-LFSSM** | 0.7763±0.0020 | **0.7806±0.0004** | **0.7837±0.0006** | **0.7928±0.0005** | **0.7903±0.0005** |

**Robustness to elapsed time.** Time-gap stratification directly examines the temporal modeling motivation of CT-LFSSM. Because the model combines continuous-time states with different retention horizons, its predictive advantage should not be restricted to either short or long elapsed-time regimes. We therefore evaluate performance across four real elapsed-time intervals on Algebra2005 (Table 3). CT-LFSSM remains within the top two on both AUC and ACC in every interval and attains the best or tied-best result in five of the eight metric–interval comparisons. Individual competitors show stronger horizon-specific behavior, such as LPKT on the 1d–7d AUC and HawkesKT on the $>$ 7d AUC, whereas CT-LFSSM remains competitive across the full range. This pattern is consistent with the intended role of multi-timescale continuous-time states: different retention horizons can contribute across different gap regimes without forcing the model to specialize to a single temporal range.

### 5.2 Ablation Study

Table 4 examines whether the mechanisms used to represent, update, and read out student knowledge provide complementary contributions to the full model. For the state-evolution pathway, removing the cognition and acquisition gates, outcome conditioning, or the hierarchical state update consistently reduces mean performance. These variants respectively weaken personalized read-

Table 3: Time-gap-stratified performance on Algebra2005. Bins are defined by the real elapsed time between consecutive valid interactions. Best results are in **bold** and second-best results are underlined.

| | $\leq$ 1h | | 1h–1d | | 1d–7d | | $>$ 7d | |
|---|---|---|---|---|---|---|---|---|
| Model | AUC | ACC | AUC | ACC | AUC | ACC | AUC | ACC |
| PFA | 0.7699±0.0007 | 0.7565±0.0006 | 0.7496±0.0020 | 0.6872±0.0310 | 0.7475±0.0027 | 0.6732±0.0030 | 0.6519±0.0051 | 0.5997±0.0036 |
| KTM | 0.8086±0.0006 | 0.7953±0.0005 | 0.7799±0.0016 | 0.6996±0.0200 | 0.7710±0.0039 | 0.7000±0.0040 | 0.6832±0.0073 | 0.6221±0.0051 |
| DAS3H | 0.8107±0.0008 | 0.7997±0.0006 | 0.7811±0.0105 | 0.7056±0.0131 | 0.7713±0.0015 | 0.7096±0.0012 | 0.6905±0.0086 | 0.6293±0.0067 |
| DKT | 0.8133±0.0015 | 0.8117±0.0007 | 0.7839±0.0026 | 0.7147±0.0045 | 0.7754±0.0018 | 0.7115±0.0014 | 0.6973±0.0104 | 0.6325±0.0133 |
| DKVMN | 0.8002±0.0005 | 0.8040±0.0004 | 0.7895±0.0056 | 0.6916±0.0062 | 0.7768±0.0024 | 0.7012±0.0016 | 0.7168±0.0068 | 0.6331±0.0058 |
| SAKT | 0.7835±0.0018 | 0.7963±0.0009 | 0.7680±0.0049 | 0.6829±0.0086 | 0.7704±0.0026 | 0.6990±0.0074 | 0.7000±0.0190 | 0.6343±0.0073 |
| AKT | 0.8125±0.0016 | 0.8075±0.0008 | 0.7934±0.0033 | 0.6931±0.0019 | 0.7841±0.0025 | 0.7100±0.0022 | 0.7166±0.0050 | 0.6344±0.0102 |
| IEKT | 0.8114±0.0010 | 0.8102±0.0005 | 0.7827±0.0012 | 0.7075±0.0025 | 0.7729±0.0016 | 0.7109±0.0012 | 0.6893±0.0065 | 0.6300±0.0700 |
| DIMKT | 0.8257±0.0016 | 0.8135±0.0015 | 0.7925±0.0022 | 0.7030±0.0057 | 0.7832±0.0022 | 0.7185±0.0041 | 0.7599±0.0047 | 0.6667±0.0045 |
| QIKT | 0.8123±0.0011 | 0.8098±0.0046 | 0.7904±0.0035 | 0.6546±0.0265 | 0.7818±0.0078 | 0.6940±0.0091 | 0.7617±0.0066 | 0.6256±0.0162 |
| FoLiBiKT | 0.8199±0.0011 | 0.8090±0.0015 | 0.7883±0.0052 | 0.7053±0.0072 | 0.7852±0.0023 | 0.7224±0.0053 | 0.7541±0.0025 | **0.6939±0.0059** |
| LPKT | 0.8187±0.0005 | 0.8150±0.0004 | 0.7908±0.0057 | 0.6951±0.0100 | **0.8002±0.0013** | 0.7155±0.0057 | 0.7576±0.0038 | 0.6506±0.0062 |
| simpleKT | 0.8175±0.0009 | 0.8075±0.0009 | 0.7923±0.0023 | 0.7124±0.0047 | 0.7730±0.0014 | 0.7196±0.0030 | 0.7609±0.0033 | 0.6638±0.0097 |
| HawkesKT | 0.8170±0.0076 | 0.8141±0.0040 | **0.7940±0.0041** | 0.6688±0.0051 | 0.7858±0.0035 | 0.7030±0.0029 | **0.7953±0.0023** | 0.6669±0.0049 |
| **CT-LFSSM** | **0.8261±0.0005** | **0.8153±0.0016** | **0.7940±0.0034** | **0.7188±0.0072** | 0.7866±0.0029 | **0.7230±0.0023** | 0.7619±0.0049 | 0.6677±0.0180 |

Table 4: Ablation study on ASSIST2012, ASSIST2009, and Algebra2005. Results are mean ± population standard deviation over five folds. Best results are in **bold**.

| | ASSIST2012 | | ASSIST2009 | | Algebra2005 | |
|---|---|---|---|---|---|---|
| Model | AUC | ACC | AUC | ACC | AUC | ACC |
| Full CT-LFSSM | **0.7833±0.0005** | **0.7582±0.0003** | **0.7873±0.0019** | **0.7419±0.0014** | **0.8304±0.0005** | **0.8168±0.0016** |
| w/o hierarchical item-response | 0.7730±0.0006 | 0.7519±0.0012 | 0.7680±0.0022 | 0.7270±0.0007 | 0.8135±0.0014 | 0.8070±0.0016 |
| w/o cognition and acquisition gates | 0.7796±0.0015 | 0.7558±0.0009 | 0.7818±0.0016 | 0.7376±0.0012 | 0.8243±0.0005 | 0.8111±0.0019 |
| w/o outcome conditioning | 0.7822±0.0003 | 0.7574±0.0002 | 0.7857±0.0017 | 0.7408±0.0006 | 0.8267±0.0007 | 0.8138±0.0015 |
| w/o episodic memory | 0.7699±0.0008 | 0.7507±0.0007 | 0.7511±0.0006 | 0.7207±0.0009 | 0.8037±0.0017 | 0.7900±0.0068 |
| w/o hierarchical state update | 0.7816±0.0007 | 0.7569±0.0005 | 0.7863±0.0017 | 0.7412±0.0007 | 0.8261±0.0007 | 0.8117±0.0015 |

out/acquisition, response-dependent learning, and multi-timescale consolidation, suggesting that the post-response update benefits from combining these structured mechanisms rather than relying on a single unrestricted transition.

The remaining ablations examine prediction-side information. Removing the hierarchical item-response formulation causes a substantial reduction across all three datasets, supporting the value of sharing parameters through KCs while retaining item-specific variation. Removing episodic memory produces the largest degradation, indicating that retrieving relevant past events provides information that is complementary to the continuous latent state. Taken together, every ablated variant performs below the full model across all six metric–dataset combinations, showing that the overall performance is supported jointly by the structured state-update pathway and the complementary prediction components rather than by a single isolated module.

## 6 Conclusion

We presented CT-LFSSM, a continuous-time state-space model that represents each knowledge component with fast-, medium-, and slow-timescale latent states, propagates them through personalized exponential decay, and separates response-driven belief revision from practice-induced learning. Across four real-world benchmarks, CT-LFSSM achieves state-of-the-art performance under both standardized re-benchmarks and the official XES3G5M benchmark, and remains robust across student histories of different lengths and across elapsed-time intervals spanning minutes to weeks. These results indicate that jointly modeling continuous-time forgetting, multi-timescale knowledge states, and response-dependent state updates provides an effective foundation for knowledge tracing under irregular learning trajectories. Future work can further examine the statistical identifiability of the latent timescales and extend episodic modeling beyond the current finite memory.

## AI USE STATEMENT

In this work, we used generative AI tools for academic English expression, including language polishing and editing of the manuscript. We have not used generative AI tools for research ideation, literature review, model design, code generation, figure creation, or experimental result generation. All AI-assisted text was reviewed and revised by the authors to ensure factual accuracy, originality, and appropriate attribution. We take full responsibility for the final content of this work.

## ETHICS STATEMENT

This work uses only publicly available, de-identified educational datasets. No human subjects were directly involved, and no new data were collected. We do not foresee any ethical concerns regarding privacy, fairness, or potential harm. The datasets are used strictly for non-commercial academic research.

## REPRODUCIBILITY STATEMENT

To facilitate reproducibility, we provide a complete description of the model architecture, state evolution, and training objective in the Method section and the appendix. Dataset statistics, preprocessing steps, and baseline configurations are described in the appendix. The evaluation protocol, including data splits, model selection, and metrics, is summarized in the Experiments section. The code will be released upon acceptance. All random seeds, hyperparameters, and hardware configurations are reported in the appendix.

## REFERENCES

Ghodai Abdelrahman and Qing Wang. Knowledge tracing with sequential key-value memory networks. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 175–184, 2019. doi: 10.1145/3331184.3331195.

Youheng Bai, Xueyi Li, Zitao Liu, Yaying Huang, Mi Tian, and Weiqi Luo. Rethinking and improving student learning and forgetting processes for attention based knowledge tracing models. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2025.

Jiahao Chen, Zitao Liu, Shuyan Huang, Qiongqiong Liu, and Weiqi Luo. Improving interpretability of deep sequential knowledge tracing models with question-centric cognitive representations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 14196–14204, 2023. doi: 10.1609/aaai.v37i12.26661.

Ke Cheng, Linzhi Peng, Pengyang Wang, Junchen Ye, Leilei Sun, and Bowen Du. DyGKT: Dynamic graph learning for knowledge tracing. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, 2024.

Weihua Cheng, Hanwen Du, Chunxiao Li, Ersheng Ni, Liangdi Tan, Tianqi Xu, and Yongxin Ni. Uncertainty-aware knowledge tracing. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2025.

Benoît Choffin, Fabrice Popineau, Yolaine Bourda, and Jill-Jênn Vie. DAS3H: Modeling student learning and forgetting for optimally scheduling distributed practice of skills. In *Proceedings of the 12th International Conference on Educational Data Mining*, pp. 29–38, 2019.

Youngduck Choi, Youngnam Lee, Junghyun Cho, Jineon Baek, Byungsoo Kim, Yeongmin Cha, Dongmin Shin, Chan Bae, and Jaewe Heo. Towards an appropriate query, key, and value computation for knowledge tracing. In *Proceedings of the Seventh ACM Conference on Learning at Scale*, pp. 341–344, 2020. doi: 10.1145/3386527.3405945.

Albert T. Corbett and John R. Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction*, 1994.

Mingyu Feng, Neil T. Heffernan, and Kenneth R. Koedinger. Addressing the assessment challenge with an online system that tutors as it assesses. *User Modeling and User-Adapted Interaction*, 19 (3):243–266, 2009. doi: 10.1007/s11257-009-9063-7.

Aritra Ghosh, Neil T. Heffernan, and Andrew S. Lan. Context-aware attentive knowledge tracing. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 2330–2339, 2020. doi: 10.1145/3394486.3403282.

Xiaopeng Guo, Zhijie Huang, Jie Gao, Mingyu Shang, Maojing Shu, and Jun Sun. Enhancing knowledge tracing via adversarial training. In *Proceedings of the 29th ACM International Conference on Multimedia*, pp. 367–375, 2021. doi: 10.1145/3474085.3475554.

Mingliang Hou, Xueyi Li, Teng Guo, Zitao Liu, Mi Tian, Renqiang Luo, and Weiqi Luo. Cognitive fluctuations enhanced attention network for knowledge tracing. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2025.

Shuyan Huang, Zitao Liu, Xiangyu Zhao, Weiqi Luo, and Jian Weng. Towards robust knowledge tracing models via $k$-sparse attention. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 2441–2445, 2023. doi: 10.1145/3539618.3592073.

Tao Huang, Gang Xue, Junjie Hu, Shengze Hu, Huali Yang, Xinjia Ou, and Zhichao Zhang. Time-frequency domain coupling-oriented knowledge tracing via learning behavior decoupling. *Knowledge-Based Systems*, 2026.

Yoonjin Im, Eunseong Choi, Heejin Kook, and Jongwuk Lee. Forgetting-aware linear bias for attentive knowledge tracing. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pp. 3958–3962, 2023. doi: 10.1145/3583780.3615191.

Jinseok Lee and Dit-Yan Yeung. Knowledge query network for knowledge tracing: How knowledge interacts with skills. In *Proceedings of the 9th International Conference on Learning Analytics and Knowledge*, pp. 491–500, 2019. doi: 10.1145/3303772.3303786.

Qing Li, Xin Yuan, Jieyu Yue, Xiaoxuan Shen, Ruxia Liang, Sannyuya Liu, and Zhonghua Yan. Dual-view multi-scale cognitive representation for deep knowledge tracing. *Knowledge-Based Systems*, 2025.

Xueyi Li, Youheng Bai, Teng Guo, Zitao Liu, Yaying Huang, Xiangyu Zhao, Feng Xia, Weiqi Luo, and Jian Weng. Enhancing length generalization for attention based knowledge tracing models with linear biases. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2024.

Zitao Liu, Qiongqiong Liu, Jiahao Chen, Shuyan Huang, Jiliang Tang, and Weiqi Luo. pyKT: A python library to benchmark deep learning based knowledge tracing models. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Zitao Liu, Qiongqiong Liu, Jiahao Chen, Shuyan Huang, Boyu Gao, Weiqi Luo, and Jian Weng. Enhancing deep knowledge tracing with auxiliary tasks. In *Proceedings of the ACM Web Conference 2023*, pp. 4178–4187, 2023a. doi: 10.1145/3543507.3583866.

Zitao Liu, Qiongqiong Liu, Jiahao Chen, Shuyan Huang, and Weiqi Luo. simpleKT: A simple but tough-to-beat baseline for knowledge tracing. In *International Conference on Learning Representations*, 2023b.

Zitao Liu, Qiongqiong Liu, Teng Guo, Jiahao Chen, Shuyan Huang, Xiangyu Zhao, Jiliang Tang, Weiqi Luo, and Jian Weng. XES3G5M: A knowledge tracing benchmark dataset with auxiliary information. In *Advances in Neural Information Processing Systems*, volume 36, pp. 32958–32970, 2023c.

Ting Long, Yunfei Liu, Jian Shen, Weinan Zhang, and Yong Yu. Tracing knowledge state with individual cognition and acquisition estimation. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 173–182, 2021. doi: 10.1145/3404835.3462827.

Haiping Ma, Jingyuan Wang, Hengshu Zhu, Xin Xia, Haifeng Zhang, Xingyi Zhang, and Lei Zhang. Reconciling cognitive modeling with knowledge forgetting: A continuous time-aware neural network approach. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2022.

Koki Nagatani, Qian Zhang, Masahiro Sato, Yan-Ying Chen, Francine Chen, and Tomoko Ohkuma. Augmenting knowledge tracing by considering forgetting behavior. In *The World Wide Web Conference*, pp. 3101–3107. ACM, 2019. doi: 10.1145/3308558.3313565.

Hiromi Nakagawa, Yusuke Iwasawa, and Yutaka Matsuo. Graph-based knowledge tracing: Modeling student proficiency using graph neural network. In *IEEE/WIC/ACM International Conference on Web Intelligence*, pp. 156–163, 2019. doi: 10.1145/3350546.3352513.

Shalini Pandey and George Karypis. A self-attentive model for knowledge tracing. In *Proceedings of the 12th International Conference on Educational Data Mining*, pp. 384–389, 2019.

Philip I. Pavlik, Hao Cen, and Kenneth R. Koedinger. Performance factors analysis—a new alternative to knowledge tracing. In *Artificial Intelligence in Education*, pp. 531–538, 2009. doi: 10.3233/978-1-60750-028-5-531.

Chris Piech, Jonathan Bassen, Jonathan Huang, Surya Ganguli, Mehran Sahami, Leonidas J. Guibas, and Jascha Sohl-Dickstein. Deep knowledge tracing. In *Advances in Neural Information Processing Systems*, volume 28, 2015.

Shuanghong Shen, Qi Liu, Enhong Chen, Zhenya Huang, Wei Huang, Yu Yin, Yu Su, and Shijin Wang. Learning process-consistent knowledge tracing. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1452–1460, 2021. doi: 10.1145/3447548.3467237.

Shuanghong Shen, Zhenya Huang, Qi Liu, Yu Su, Shijin Wang, and Enhong Chen. Assessing student's dynamic knowledge state by exploring the question difficulty effect. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 427–437, 2022. doi: 10.1145/3477495.3531939.

Shashank Sonkar, Andrew E. Waters, Andrew S. Lan, Phillip J. Grimaldi, and Richard G. Baraniuk. qDKT: Question-centric deep knowledge tracing. In *Proceedings of the 13th International Conference on Educational Data Mining*, pp. 677–681, 2020.

John Stamper, Alexandru Niculescu-Mizil, Steve Ritter, Geoffrey Gordon, and Kenneth R. Koedinger. Algebra i 2005–2006 development data set from kdd cup 2010 educational data mining challenge. Carnegie Mellon University, Pittsburgh Science of Learning Center DataShop, 2010.

Jill-Jênn Vie and Hisashi Kashima. Knowledge tracing machines: Factorization machines for knowledge tracing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 750–757, 2019. doi: 10.1609/aaai.v33i01.3301750.

Chenyang Wang, Weizhi Ma, Min Zhang, Chuancheng Lv, Fengyuan Wan, Huijie Lin, Taoran Tang, Yiqun Liu, and Shaoping Ma. Temporal cross-effects in knowledge tracing. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, pp. 517–525, 2021. doi: 10.1145/3437963.3441802.

Chun-Kit Yeung. Deep-IRT: Make deep learning based knowledge tracing explainable using item response theory. In *Proceedings of the 12th International Conference on Educational Data Mining*, pp. 683–686, 2019.

Chun-Kit Yeung and Dit-Yan Yeung. Addressing two problems in deep knowledge tracing via prediction-consistent regularization. In *Proceedings of the 5th Annual ACM Conference on Learning at Scale*, pp. 5:1–5:10. ACM, 2018. doi: 10.1145/3231644.3231647.

Jiani Zhang, Xingjian Shi, Irwin King, and Dit-Yan Yeung. Dynamic key-value memory networks for knowledge tracing. In *Proceedings of the 26th International Conference on World Wide Web*, pp. 765–774, 2017. doi: 10.1145/3038912.3052580.

Moyu Zhang, Xinning Zhu, Chunhong Zhang, Feng Pan, Wenchen Qian, and Hui Zhao. No length left behind: Enhancing knowledge tracing for modeling sequences of excessive or insufficient lengths. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pp. 3226–3235, 2023. doi: 10.1145/3583780.3614988.

Hanqi Zhou, Robert Bamler, Charley M. Wu, and Álvaro Tejero-Cantero. Predictive, scalable and interpretable knowledge tracing on structured domains. In *International Conference on Learning Representations (ICLR)*, 2024.

# A Additional Method Details

This appendix expands the implementation details of CT-LFSSM while retaining the notation and computational ordering of the main text. We use the timescale index $r \in \mathcal{R} = \{\mathrm{f}, \mathrm{m}, \mathrm{s}\}$ throughout. All post-response quantities are computed only after the current prediction has been formed, so the current label and auxiliary behavior variables cannot leak into $p_{u,n}$.

## A.1 Notation

For compactness, Table 5 lists only the symbols that are reused across several parts of this appendix. Symbols that are local to one construction are defined at first use.

| Symbol | Meaning |
|---|---|
| $u, n, q, k, r$ | student, interaction index, item, KC, and timescale index, respectively |
| $\boldsymbol{A} = [A_{qk}]$ | binary item–KC incidence matrix; $A_{qk} = 1$ iff item $q$ directly requires KC $k$ |
| $\pi_{qk}$ | normalized *direct* item–KC weight used by the observation model |
| $z^{-}_{u,n,k,r}, z^{+}_{u,n,k,r}$ | pre-response and post-response latent mastery state for KC $k$ at timescale $r$ |
| $\boldsymbol{h}_{u,n}$ | student profile state after interaction $n$ |
| $\mathcal{M}_{u,n}$ | bounded episodic memory after interaction $n$ |
| $H^{(0)}_{k,r}, \phi_{u,n,k}, H_{u,n,k,r}$ | base, personalized multiplicative factor, and personalized half-life |
| $\omega_{u,n,r}$ | pre-response mixture weight over the three timescales |
| $m^{\mathrm{KC}}_{u,n,k}$ | pre-response KC mastery readout |
| $\rho^{\mathrm{pred}}_{u,n,k}$ | normalized KC weight used for prediction |
| $c^{\mathrm{cog}}_{u,n}, \beta_{u,n}, \delta^{\mathrm{epi}}_{u,n}$ | cognition scale, profile bias, and episodic logit residual |
| $E_{k,r}, \gamma^{E}_{u,n}$ | ordered evidence strength and event-dependent evidence gate |
| $G_{u,n}, G_{u,n,r}$ | overall practice gain and its timescale-specific consolidated gain |
| $\rho^{\mathrm{acq}}_{u,n,k}$ | KC-specific acquisition/transfer coefficient used only in the post-response learning update |

Table 5: Principal notation used in the additional method details.

## A.2 Half-Life Parameterization Details

**Ordered base half-lives.** The main text parameterizes the three log-half-lives for each KC as

$$L_{k,\mathrm{f}} = \theta^{H}_{k}, \tag{19}$$

$$L_{k,\mathrm{m}} = L_{k,\mathrm{f}} + \mathrm{softplus}\big(\eta^{H}_{k,\mathrm{m}}\big), \tag{20}$$

$$L_{k,\mathrm{s}} = L_{k,\mathrm{m}} + \mathrm{softplus}\big(\eta^{H}_{k,\mathrm{s}}\big). \tag{21}$$

Thus, before numerical boundary saturation, $L_{k,\mathrm{f}} \le L_{k,\mathrm{m}} \le L_{k,\mathrm{s}}$ and $H^{(0)}_{k,r} = \exp(L_{k,r})$ is ordered in the same way.

The implementation additionally limits log-half-lives to a numerically safe interval:

$$\bar{L}_{k,r} = \mathrm{clip}(L_{k,r}, \log H_{\min}, \log H_{\max}), \qquad H^{(0)}_{k,r} = \exp(\bar{L}_{k,r}). \tag{22}$$

Because clipping is monotone non-decreasing, it cannot reverse the ordering; it can only make neighboring scales equal when one or both values saturate at a boundary. The final V4 configurations use $H_{\min} = 10^{-3}$ days and $H_{\max} = 3650$ days.

**Initialization from interpretable half-lives.** For desired initial half-lives $0 < h_{\mathrm{f}} < h_{\mathrm{m}} < h_{\mathrm{s}}$, the raw parameters are initialized as

$$\theta^{H}_{k} \leftarrow \log h_{\mathrm{f}}, \tag{23}$$

$$\eta^{H}_{k,\mathrm{m}} \leftarrow \mathrm{softplus}^{-1}\left(\log \frac{h_{\mathrm{m}}}{h_{\mathrm{f}}}\right), \tag{24}$$

$$\eta^{H}_{k,\mathrm{s}} \leftarrow \mathrm{softplus}^{-1}\left(\log \frac{h_{\mathrm{s}}}{h_{\mathrm{m}}}\right). \tag{25}$$

The reported V4 initialization uses approximately $(h_{\mathrm{f}}, h_{\mathrm{m}}, h_{\mathrm{s}}) = (0.2481, 4.5753, 64.7496)$ days for every KC before training.

**Continuous-time decay.** Given the personalized half-life introduced below, the no-event evolution over a gap $\Delta\tau_{u,n}$ is

$$z^{-}_{u,n,k,r} = z^{+}_{u,n-1,k,r} \exp\left(-\ln 2 \, \frac{\Delta\tau_{u,n}}{H_{u,n,k,r}}\right) = z^{+}_{u,n-1,k,r} 2^{-\Delta\tau_{u,n}/H_{u,n,k,r}}. \tag{26}$$

This update is applied before reading the current response.

### A.3 Personalization and Gate Details

**Low-rank half-life personalization.** The main text writes

$$\phi_{u,n,k} = \exp\left[\kappa_H \tanh\big(f_H(\boldsymbol{h}_{u,n-1}, k)\big)\right], \qquad H_{u,n,k,r} = H^{(0)}_{k,r} \phi_{u,n,k}. \tag{27}$$

In the implementation, $f_H$ is a low-rank student–KC interaction rather than a separate state variable. Let $r_H$ be the personalization rank, $\boldsymbol{q}^H_{u,n} = \boldsymbol{W}_H \boldsymbol{h}_{u,n-1} \in \mathbb{R}^{r_H}$, and let $\boldsymbol{k}^H_k \in \mathbb{R}^{r_H}$ be a learned KC key. Then

$$f_H(\boldsymbol{h}_{u,n-1}, k) = b_H(\boldsymbol{h}_{u,n-1}) + \frac{\langle \boldsymbol{q}^H_{u,n}, \boldsymbol{k}^H_k \rangle}{\sqrt{r_H}}, \tag{28}$$

where $b_H(\cdot)$ is a scalar linear profile term. The final V4 configuration uses $r_H = 16$ and $\kappa_H = 0.7$. The profile term and student query projection are zero-initialized, so $\phi_{u,n,k} = 1$ initially. After multiplication by $\phi_{u,n,k}$, the implementation again applies the same half-life bounds $[H_{\min}, H_{\max}]$.

The same positive factor $\phi_{u,n,k}$ is shared across all three timescales for a given $(u, n, k)$. Hence personalization changes the absolute forgetting speed while preserving the fast–medium–slow ordering before and after monotone boundary clipping.

**Item representation and direct KC context.** The scale gate uses an item representation $\boldsymbol{e}_q$ and a direct KC context $\boldsymbol{c}_q$. With $\pi_{qk} = A_{qk} / \sum_j A_{qj}$, the direct KC context is

$$\boldsymbol{c}_q = \sum_{k=1}^{K} \pi_{qk} \boldsymbol{e}^{\mathrm{KC}}_k, \tag{29}$$

where $\boldsymbol{e}^{\mathrm{KC}}_k$ is a learned KC embedding. The item representation combines a KC-factorized base with a low-dimensional item-specific semantic residual:

$$\boldsymbol{e}_q = \sum_{k=1}^{K} \pi_{qk} \boldsymbol{e}^{\mathrm{item}}_k + \lambda_{\mathrm{sem}} \tanh\big(\boldsymbol{W}_{\mathrm{sem}} \tanh(\mathbf{s}^{\mathrm{sem}}_q)\big). \tag{30}$$

The final V4 implementation uses a 64-dimensional item representation, an 8-dimensional item semantic code $\mathbf{s}^{\mathrm{sem}}_q$, and $\lambda_{\mathrm{sem}} = 0.1$. The semantic codes are initialized at zero.

**Multi-resolution time features.** Let the gap be measured in days. The implementation uses three normalized log-gap features

$$t^{(j)}_{u,n} = \frac{\min\{\log(1 + \Delta\tau_{u,n}/\delta_j), C_t\}}{C_t}, \qquad \delta_j \in \left\{\frac{10}{1440}, \frac{1}{24}, 1\right\}, \tag{31}$$

corresponding to 10-minute, one-hour, and one-day reference scales, with $C_t = 8$. We denote their concatenation by $\boldsymbol{t}_{u,n}$.

**Pre-response scale gate.** The abstract gate $f_\omega$ in the main text is instantiated as a global scale prior plus a residual MLP:

$$\mathbf{a}^{\omega}_{u,n} = \mathbf{b}^{\omega} + \mathrm{MLP}_{\omega}\big([\boldsymbol{h}_{u,n-1}; \boldsymbol{e}_{q_{u,n}}; \boldsymbol{c}_{q_{u,n}}; \boldsymbol{t}_{u,n}]\big), \tag{32}$$

$$\omega_{u,n,r} = \frac{\exp(a^{\omega}_{u,n,r})}{\sum_{r' \in \mathcal{R}} \exp(a^{\omega}_{u,n,r'})}. \tag{33}$$

The V4 implementation uses $\mathbf{b}^{\omega} = (0, 0.3, 0.7)$, a 128-unit hidden layer, and dropout $0.2$. The final output layer of the residual MLP is zero-initialized, so the initial gate is determined by the global scale prior. The resulting KC readout is

$$m^{\mathrm{KC}}_{u,n,k} = \sum_{r\in\mathcal{R}} \omega_{u,n,r} z^{-}_{u,n,k,r} + \nu_k, \tag{34}$$

where $\nu_k$ is a learned KC bias. All quantities in this gate are available before observing $y_{u,n}$.

### A.4 Readout and Update Details

#### A.4.1 KC prediction readout

The prediction weights $\rho^{\mathrm{pred}}_{u,n,k}$ are distinct from the direct pooling weights $\pi_{qk}$. The latter are used by the item-response parameterization, whereas $\rho^{\mathrm{pred}}$ may also incorporate prerequisite context.

Let $P^{(1)}_{qk} \geq 0$ and $P^{(2)}_{qk} \geq 0$ denote fixed one-hop and two-hop prerequisite-context buffers derived from the training-side KC graph. They are local implementation quantities and are not trainable item parameters. A structural candidate weight is

$$R_{qk} = A_{qk} + \lambda_1 P^{(1)}_{qk} + \lambda_2 P^{(2)}_{qk}, \tag{35}$$

with

$$\lambda_1 = \lambda_{1,\max}\sigma(\zeta_1), \qquad \lambda_2 = \lambda_1\lambda_{2,\max}\sigma(\zeta_2). \tag{36}$$

The final V4 implementation uses $\lambda_{1,\max} = 0.4$ and $\lambda_{2,\max} = 0.5$.

When question-aware attention is disabled, the prediction readout is simply

$$\rho^{\mathrm{pred}}_{u,n,k} = \frac{R_{q_{u,n},k}}{\sum_j R_{q_{u,n},j}}. \tag{37}$$

When it is enabled, attention is evaluated only on the sparse candidate set $\mathcal{C}_q = \{k : R_{qk} > 0\}$. For $k \in \mathcal{C}_q$,

$$a^{\mathrm{pred}}_{qk} = \frac{\langle \boldsymbol{W}_Q \boldsymbol{e}_q,\, \boldsymbol{W}_K \boldsymbol{e}^{\mathrm{KC}}_k \rangle}{\sqrt{d_{\mathrm{pred}}}\, T_{\mathrm{pred}}} + \lambda_{\mathrm{str}} \log(\max\{R_{qk}, \epsilon_{\mathrm{num}}\}), \tag{38}$$

and

$$\rho^{\mathrm{pred}}_{u,n,k} = \begin{cases} \dfrac{\exp(a^{\mathrm{pred}}_{q_{u,n},k})}{\sum_{j\in\mathcal{C}_{q_{u,n}}} \exp(a^{\mathrm{pred}}_{q_{u,n},j})}, & k \in \mathcal{C}_{q_{u,n}}, \\ 0, & \text{otherwise.} \end{cases} \tag{39}$$

Here $d_{\mathrm{pred}} = 64$, $T_{\mathrm{pred}} = 0.85$, and $\lambda_{\mathrm{str}} = 1$ in configurations that enable question-aware readout.

Define the prerequisite-aware latent item mastery before cognition scaling as

$$\bar{m}_{u,n} = \sum_{k=1}^{K} \rho^{\mathrm{pred}}_{u,n,k} m^{\mathrm{KC}}_{u,n,k}. \tag{40}$$

#### A.4.2 Cognition scale and profile bias

The positive cognition multiplier is

$$c^{\mathrm{cog}}_{u,n} = \exp\left[\kappa_c \tanh\big(f_c([\boldsymbol{h}_{u,n-1}; \boldsymbol{e}_{q_{u,n}}])\big)\right], \tag{41}$$

where $f_c$ is a two-layer MLP. With the V4 setting $\kappa_c = 0.5$, $c^{\mathrm{cog}}_{u,n} \in (e^{-0.5}, e^{0.5})$. The profile bias is a scalar affine readout

$$\beta_{u,n} = f_\beta(\boldsymbol{h}_{u,n-1}), \tag{42}$$

whose weight and bias are initialized at zero. The mastery entering the 4PL readout is therefore

$$m_{u,n} = c^{\mathrm{cog}}_{u,n} \bar{m}_{u,n} + \beta_{u,n}. \tag{43}$$

### A.4.3 PAST-ONLY EPISODIC RETRIEVAL

Each valid episodic slot $j$ stores a semantic key, KC-context key, value, and timestamp. Let its age before the current interaction be $\Delta\tau^{\mathrm{mem}}_{u,n,j}$. With normalized semantic and KC queries/keys, the retrieval score is

$$a^{\mathrm{epi}}_{u,n,j} = \left\langle \widehat{\boldsymbol{q}}^{\mathrm{sem}}_{u,n}, \widehat{\boldsymbol{k}}^{\mathrm{sem}}_{u,j} \right\rangle + \lambda_{\mathrm{KC}} \left\langle \widehat{\boldsymbol{q}}^{\mathrm{KC}}_{u,n}, \widehat{\boldsymbol{k}}^{\mathrm{KC}}_{u,j} \right\rangle - \ln 2 \, \frac{\Delta\tau^{\mathrm{mem}}_{u,n,j}}{H_{\mathrm{epi}}}. \tag{44}$$

The first two terms express semantic and structural relevance; the last term implements explicit temporal forgetting in retrieval. Masked softmax over valid slots gives $\alpha^{\mathrm{epi}}_{u,n,j}$. A second retention multiplier

$$r^{\mathrm{epi}}_{u,n,j} = \exp\left( - \ln 2 \, \frac{\Delta\tau^{\mathrm{mem}}_{u,n,j}}{H_{\mathrm{epi}}} \right) \tag{45}$$

modulates the retrieved value:

$$\boldsymbol{c}^{\mathrm{epi}}_{u,n} = \sum_j \alpha^{\mathrm{epi}}_{u,n,j} r^{\mathrm{epi}}_{u,n,j} \boldsymbol{v}^{\mathrm{epi}}_{u,j}, \tag{46}$$

$$s^{\mathrm{epi}}_{u,n} = \sum_j \alpha^{\mathrm{epi}}_{u,n,j} r^{\mathrm{epi}}_{u,n,j}. \tag{47}$$

The resulting logit residual is bounded as

$$\delta^{\mathrm{epi}}_{u,n} = \kappa_{\mathrm{epi}} s^{\mathrm{epi}}_{u,n} \tanh\left( f_{\mathrm{epi}}([\boldsymbol{c}^{\mathrm{epi}}_{u,n}; \boldsymbol{e}_{q_{u,n}}; \boldsymbol{h}_{u,n-1}]) \right). \tag{48}$$

The final V4 implementation uses capacity $M = 32$, 64-dimensional memory values, $\lambda_{\mathrm{KC}} = 1$, and $\kappa_{\mathrm{epi}} = 1.2$. Its episodic half-life is initialized at approximately $1.986$ days and constrained to $[10^{-3}, 365]$ days.

Crucially, $\mathcal{M}_{u,n-1}$ contains only past events when this retrieval is performed. The current interaction is appended only after prediction, evidence revision, practice learning, and profile updating. Moreover, $\delta^{\mathrm{epi}}_{u,n}$ is an observation-logit residual; it is never written into the latent mastery state.

### A.4.4 POST-RESPONSE EVENT REPRESENTATION AND PROFILE UPDATE

After $p_{u,n}$ is produced and $y_{u,n}$ becomes available, define $\epsilon_{u,n} = y_{u,n} - p_{u,n}$. The event encoder receives

$$\boldsymbol{x}^{\mathrm{raw}}_{u,n} = [\boldsymbol{h}_{u,n-1}; \boldsymbol{e}_{q_{u,n}}; \bar{m}_{u,n}; y_{u,n}; \epsilon_{u,n}; \boldsymbol{t}_{u,n}; \mathbf{b}_{u,n}; \mathbf{m}_{u,n}], \tag{49}$$

where $\mathbf{b}_{u,n}$ contains transformed response time, hint count, and attempt count, and $\mathbf{m}_{u,n}$ contains the corresponding missingness indicators. For a nonnegative behavior value $x$, the implementation uses

$$T_B(x; \lambda) = \frac{\log(1 + \max\{x, 0\})}{\lambda}, \tag{50}$$

with scales $8, 3, 3$ for response time, hint count, and attempt count, respectively. Missing values are replaced by zero before transformation and are explicitly marked by their missingness indicators.

The observed-event vector is

$$\boldsymbol{x}^{\mathrm{evt}}_{u,n} = f_{\mathrm{evt}}(\boldsymbol{x}^{\mathrm{raw}}_{u,n}), \tag{51}$$

where the V4 event encoder is a $140 \to 128 \to 128$ MLP with GELU activations and dropout $0.2$. The same event vector drives the evidence gate, gain head, consolidation head, and acquisition/transfer modules described below.

After the latent learning update, the persistent profile is updated by

$$\boldsymbol{h}_{u,n} = \mathrm{GRUCell}(\boldsymbol{x}^{\mathrm{evt}}_{u,n}, \boldsymbol{h}_{u,n-1}). \tag{52}$$

Only then is the current event encoded into the bounded FIFO episodic memory to form $\mathcal{M}_{u,n}$.

### A.5 Evidence and Learning Details

**Ordered evidence strengths.** To prevent the post-response evidence correction from being strongest on the slowest state, CT-LFSSM parameterizes

$$E_{k,\mathrm{f}} = E_{\max}\sigma(\zeta^E_{k,\mathrm{f}}), \tag{53}$$

$$E_{k,\mathrm{m}} = E_{k,\mathrm{f}}\sigma(\zeta^E_{k,\mathrm{m}}), \tag{54}$$

$$E_{k,\mathrm{s}} = E_{k,\mathrm{m}}\sigma(\zeta^E_{k,\mathrm{s}}). \tag{55}$$

Therefore $E_{k,\mathrm{f}} \geq E_{k,\mathrm{m}} \geq E_{k,\mathrm{s}} \geq 0$. The V4 implementation uses $E_{\max} = 1.25$, with the initial medium-to-fast and slow-to-medium ratios set to $0.7$.

The event-dependent evidence gate is

$$\gamma^E_{u,n} = \sigma\big(f_E(\boldsymbol{x}^{\mathrm{evt}}_{u,n})\big)\,. \tag{56}$$

For numerical stability, define $\bar{z}^-_{u,n,k,r} = \mathrm{clip}(z^-_{u,n,k,r}, \epsilon_{\mathrm{num}}, 1-\epsilon_{\mathrm{num}})$. Evidence revision is applied only to directly tagged KCs:

$$\widetilde{z}_{u,n,k,r} = \sigma\Big(\mathrm{logit}(\bar{z}^-_{u,n,k,r}) + A_{q_{u,n}k}E_{k,r}\gamma^E_{u,n}\epsilon_{u,n}\Big)\,. \tag{57}$$

Thus a positive residual shifts direct KC evidence upward and a negative residual shifts it downward, while untagged KCs satisfy $\widetilde{z}_{u,n,k,r} = z^-_{u,n,k,r}$ up to numerical clipping.

**Outcome-conditioned practice gain.** The overall practice gain is

$$\bar{G}_{u,n} = G_{\max}\sigma\big(f_G(\boldsymbol{x}^{\mathrm{evt}}_{u,n})\big)\,, \tag{58}$$

$$\chi_{\mathrm{err}} = \sigma(\zeta_{\mathrm{err}}), \tag{59}$$

$$G_{u,n} = \bar{G}_{u,n}\left[y_{u,n} + (1-y_{u,n})\chi_{\mathrm{err}}\right]. \tag{60}$$

The final V4 implementation uses $G_{\max} = 0.35$ and initializes $\chi_{\mathrm{err}} = 0.25$, allowing incorrect attempts to induce a smaller but nonzero practice gain.

A two-output consolidation head produces $c^{\mathrm{m}}_{u,n}, c^{\mathrm{s}}_{u,n} \in (0,1)$, and the three timescale-specific gains are constrained hierarchically:

$$G_{u,n,\mathrm{f}} = G_{u,n}, \qquad G_{u,n,\mathrm{m}} = G_{u,n}c^{\mathrm{m}}_{u,n}, \qquad G_{u,n,\mathrm{s}} = G_{u,n}c^{\mathrm{m}}_{u,n}c^{\mathrm{s}}_{u,n}. \tag{61}$$

This guarantees $G_{u,n,\mathrm{f}} \geq G_{u,n,\mathrm{m}} \geq G_{u,n,\mathrm{s}} \geq 0$. The consolidation output weights are zero-initialized, with initial factors approximately $0.6$ and $0.5$.

**Personalized acquisition and prerequisite transfer.** For a direct KC, the event-conditioned acquisition score is

$$\alpha^{\mathrm{acq}}_{u,n,k} = \sigma\big(f_A([\boldsymbol{x}^{\mathrm{evt}}_{u,n}; \boldsymbol{e}^{\mathrm{KC}}_k])\big)\,. \tag{62}$$

The transfer coefficients are bounded hierarchically:

$$\eta_1 = \eta_{1,\max}\sigma(\zeta^A_1), \qquad \eta_2 = \eta_1\eta_{2,\max}\sigma(\zeta^A_2), \tag{63}$$

with $\eta_{1,\max} = 0.25$ and $\eta_{2,\max} = 0.5$ in V4. Using the same fixed one-hop and two-hop prerequisite buffers introduced above, the final acquisition coefficient is

$$\rho^{\mathrm{acq}}_{u,n,k} = \mathrm{clip}\Big(A_{q_{u,n}k}\alpha^{\mathrm{acq}}_{u,n,k} + \eta_1 P^{(1)}_{q_{u,n}k} + \eta_2 P^{(2)}_{q_{u,n}k}, 0, 1\Big)\,. \tag{64}$$

This quantity is deliberately separate from $\rho^{\mathrm{pred}}_{u,n,k}$: the former controls post-response state acquisition, while the latter controls the pre-response prediction readout.

Finally,

$$z^+_{u,n,k,r} = \widetilde{z}_{u,n,k,r} + \rho^{\mathrm{acq}}_{u,n,k}G_{u,n,r}\left(1-\widetilde{z}_{u,n,k,r}\right). \tag{65}$$

Because $\rho^{\mathrm{acq}}_{u,n,k} \in [0,1]$, $G_{u,n,r} \in [0, G_{\max}]$, and $G_{\max} < 1$, this update is non-decreasing relative to $\widetilde{z}$ and remains in $[0,1]$ up to floating-point safeguards.

### A.6 Observation Model and Training

**Direct KC pooling for item parameters.** For each item with at least one tagged KC,

$$\pi_{qk} = \frac{A_{qk}}{\sum_{j=1}^{K} A_{qj}}. \tag{66}$$

These direct normalized weights parameterize item difficulty and discrimination:

$$b_q = \sum_{k=1}^{K} \pi_{qk} b_k + \delta_q^b, \tag{67}$$

$$a_q = 0.1 + \mathrm{softplus}\left( \sum_{k=1}^{K} \pi_{qk} \alpha_k + \delta_q^a \right). \tag{68}$$

The item residuals are bounded in the implementation as

$$\delta_q^b = \kappa_b \tanh(\bar{\delta}_q^b), \qquad \delta_q^a = \kappa_a \tanh(\bar{\delta}_q^a), \tag{69}$$

with $\kappa_b = 1.0$ and $\kappa_a = 0.5$ in V4. Only the difficulty residual receives the explicit squared shrinkage term in the reported loss; the discrimination residual is controlled by its bounded parameterization.

Guessing and slipping use only KC-factorized raw parameters:

$$g_q = g_{\max} \sigma\left( \sum_{k=1}^{K} \pi_{qk} \theta_k^{\mathrm{guess}} \right), \tag{70}$$

$$s_q = s_{\max} \sigma\left( \sum_{k=1}^{K} \pi_{qk} \theta_k^{\mathrm{slip}} \right). \tag{71}$$

There are no item-specific residuals for $g_q$ or $s_q$. The reported configuration uses $g_{\max} = s_{\max} = 0.2$.

**4PL probability.** Combining the readout components gives

$$\ell_{u,n} = a_{q_{u,n}} \left( m_{u,n} - b_{q_{u,n}} \right) + \delta_{u,n}^{\mathrm{epi}}, \tag{72}$$

$$p_{u,n} = g_{q_{u,n}} + \left( 1 - g_{q_{u,n}} - s_{q_{u,n}} \right) \sigma(\ell_{u,n}). \tag{73}$$

Thus episodic memory perturbs the observation logit but does not alter $\boldsymbol{Z}^{\pm}$, while $c_{u,n}^{\mathrm{cog}}$ and $\beta_{u,n}$ act through the current mastery readout.

**Training objective.** Let $\mathcal{T}$ be the set of supervised target interactions in a mini-batch and $\mathcal{V}$ the set of valid recurrent events. The prediction term is the empirical binary cross-entropy

$$\mathcal{L}_{\mathrm{pred}} = -\frac{1}{|\mathcal{T}|} \sum_{(u,n)\in\mathcal{T}} \left[ y_{u,n} \log p_{u,n} + (1 - y_{u,n}) \log(1 - p_{u,n}) \right], \tag{74}$$

implemented with a numerically stable logit-based BCE routine.

The four regularizers appearing in the main text correspond to

$$\mathcal{P}_{\mathrm{prof}} = \frac{1}{|\mathcal{V}|} \sum_{(u,n)\in\mathcal{V}} \frac{\| \boldsymbol{h}_{u,n} - \boldsymbol{h}_{u,n-1} \|_2^2}{d_h}, \tag{75}$$

$$\mathcal{P}_{\mathrm{evid}} = \frac{1}{|\mathcal{V}|} \sum_{(u,n)\in\mathcal{V}} \frac{1}{3K} \sum_{k=1}^{K} \sum_{r\in\mathcal{R}} \left| \widetilde{z}_{u,n,k,r} - z^{-}_{u,n,k,r} \right|, \tag{76}$$

$$\mathcal{P}_{\mathrm{gain}} = \frac{1}{|\mathcal{V}|} \sum_{(u,n)\in\mathcal{V}} \frac{1}{3K} \sum_{k=1}^{K} \sum_{r\in\mathcal{R}} \left( z^{+}_{u,n,k,r} - \widetilde{z}_{u,n,k,r} \right), \tag{77}$$

$$\mathcal{P}_{\mathrm{item}} = \frac{1}{|\mathcal{T}|} \sum_{(u,n)\in\mathcal{T}} \left( \delta^b_{q_{u,n}} \right)^2. \tag{78}$$

In particular, the evidence penalty is an $L_1$ revision magnitude and the gain penalty is the mean nonnegative acquisition increment; neither is a squared penalty in the implementation.

The total objective is

$$\mathcal{L} = \mathcal{L}_{\text{pred}} + \lambda_{\text{prof}}\mathcal{P}_{\text{prof}} + \lambda_{\text{evid}}\mathcal{P}_{\text{evid}} + \lambda_{\text{gain}}\mathcal{P}_{\text{gain}} + \lambda_{\text{item}}\mathcal{P}_{\text{item}}. \tag{79}$$

The final V4 runs use $\lambda_{\text{prof}} = 10^{-4}$ and $\lambda_{\text{evid}} = \lambda_{\text{gain}} = \lambda_{\text{item}} = 5 \times 10^{-4}$. Although the codebase supports an optional gap-balanced prediction loss, the reported final configuration sets its coefficient to zero and therefore uses the standard empirical BCE above. Optimizer, batching, stopping, and dataset-specific hyperparameters are reported in Appendix B.1.

### A.7 Theory, Complexity, and Online Algorithm

#### A.7.1 Structural properties

**Ordered personalized half-lives.** Before clipping, $H^{(0)}_{k,\text{f}} \le H^{(0)}_{k,\text{m}} \le H^{(0)}_{k,\text{s}}$. Because $\phi_{u,n,k} > 0$ is common to all three scales,

$$H_{u,n,k,\text{f}} \le H_{u,n,k,\text{m}} \le H_{u,n,k,\text{s}}. \tag{80}$$

The implementation's scalar clipping map is monotone, so clipping can create equality at a boundary but cannot reverse this order.

**Semigroup property of no-event decay.** For a fixed half-life $H > 0$, define $D_H(\Delta) = 2^{-\Delta/H}$. Then for $\Delta_1, \Delta_2 \ge 0$,

$$D_H(\Delta_1 + \Delta_2) = D_H(\Delta_1)D_H(\Delta_2). \tag{81}$$

Consequently, when $H_{u,n,k,r}$ is held fixed over an interval containing no events, splitting that interval into arbitrary subintervals yields the same decayed state as a single update over the total elapsed time. This statement does not assert that the personalized half-life itself is constant across observed events.

**Convex multiscale state readout.** Since $\omega_{u,n,r} \ge 0$ and $\sum_r \omega_{u,n,r} = 1$,

$$\min_r z^-_{u,n,k,r} \le \sum_r \omega_{u,n,r} z^-_{u,n,k,r} \le \max_r z^-_{u,n,k,r}. \tag{82}$$

The state-dependent part of $m^{\text{KC}}_{u,n,k}$ is therefore a convex combination of the three decayed states. The additive bias $\nu_k$ is applied after this convex combination.

**Bounded 4PL probability.** If $0 \le g_q < 1$, $0 \le s_q < 1$, and $g_q + s_q < 1$, then

$$g_q < p_{u,n} < 1 - s_q \tag{83}$$

for every finite $\ell_{u,n}$. With $g_{\max} = s_{\max} = 0.2$, these conditions hold automatically in the reported configuration.

**Scope of the structural constraints.** The ordering and boundedness properties above are architectural constraints; they do not by themselves establish statistical identifiability of the latent states, nor do they guarantee that learned fast, medium, and slow components will exhibit a particular empirical specialization. Those behaviors are evaluated experimentally rather than assumed from the parameterization.

#### A.7.2 Per-event complexity

Let $d_h$ be the profile dimension, $r_H$ the forgetting-personalization rank, $M$ the episodic capacity, and $d_e$ the episodic value dimension. Ignoring fixed-width MLP constants and exploiting sparse item–KC candidate sets where available, the dominant per-event costs are

$$O(Kr_H) \quad \text{for personalized forgetting,} \tag{84}$$

$$O(3K) \quad \text{for multiscale state decay/update,} \tag{85}$$

$$O(Md_e) \quad \text{for episodic retrieval.} \tag{86}$$

Thus the primary per-event complexity is

$$O(Kr_H + 3K + Md_e), \tag{87}$$

with an additional sparse question-aware readout cost when that option is enabled. The persistent online state requires

$$O(3K + Md_e + d_h) \tag{88}$$

memory per student.

Model parameters separate naturally into: (i) KC-level parameters and embeddings, including the three half-life/evidence hierarchies; (ii) item-dependent low-dimensional semantic codes and bounded difficulty/discrimination residuals; and (iii) fixed-width neural modules. The binary item–KC matrix and prerequisite buffers are treated as fixed non-trainable structure. In the final V4 implementation, $r_H = 16$, $M = 32$, $d_e = 64$, and $d_h = 64$.

### A.7.3 Online event order

For completeness, one online interaction $(q_{u,n}, \tau_{u,n}, y_{u,n})$ is processed in the following order.

1. Compute the elapsed time $\Delta\tau_{u,n} = \tau_{u,n} - \tau_{u,n-1}$ and the multi-resolution time features $\boldsymbol{t}_{u,n}$.
2. From the *previous* profile $\boldsymbol{h}_{u,n-1}$, compute $\phi_{u,n,k}$, personalized half-lives $H_{u,n,k,r}$, and decay every latent state to obtain $z^-_{u,n,k,r}$.
3. Using only pre-response information, compute the scale gate $\omega_{u,n,r}$ and KC masteries $m^{\mathrm{KC}}_{u,n,k}$.
4. Retrieve from $\mathcal{M}_{u,n-1}$, which contains only past events, to obtain $\delta^{\mathrm{epi}}_{u,n}$.
5. Compute $\rho^{\mathrm{pred}}_{u,n,k}$, $\bar{m}_{u,n}$, $c^{\mathrm{cog}}_{u,n}$, $\beta_{u,n}$, the 4PL parameters, and finally $p_{u,n}$.
6. Observe $y_{u,n}$ and form $\epsilon_{u,n} = y_{u,n} - p_{u,n}$. Encode the current observed event as $\boldsymbol{x}^{\mathrm{evt}}_{u,n}$.
7. Apply direct-KC evidence revision to obtain $\widetilde{z}_{u,n,k,r}$.
8. Compute the practice gain $G_{u,n,r}$ and the acquisition coefficient $\rho^{\mathrm{acq}}_{u,n,k}$, then update to $z^+_{u,n,k,r}$.
9. Update the persistent profile with $\mathrm{GRUCell}(\boldsymbol{x}^{\mathrm{evt}}_{u,n}, \boldsymbol{h}_{u,n-1})$.
10. Append the current event to the bounded episodic memory, yielding $\mathcal{M}_{u,n}$, and carry $(\boldsymbol{Z}^+_{u,n}, \boldsymbol{h}_{u,n}, \mathcal{M}_{u,n})$ to the next interaction.

This ordering makes the prediction causally pre-response while allowing the observed event to revise evidence, drive practice-induced learning, update the longer-lived profile, and become available for retrieval only on future interactions.

## B Additional Experimental Results

### B.1 Training Configuration and Hyperparameters

This section summarizes the training and implementation configuration used for CT-LFSSM. Dataset construction, evaluation, and baseline comparison procedures are described elsewhere; here we report only the settings required to reproduce model training. The same configuration is used across all four datasets. CT-LFSSM uses fast-, medium-, and slow-timescale latent states, personalized exponential decay, and an event-conditioned convex gate. The response head follows the hierarchical 4PL-style item-response formulation with knowledge-component-level parameters and regularized item-specific residuals.

**Common hyperparameters.** Table 6 lists the shared CT-LFSSM settings.

Table 6: Common CT-LFSSM training hyperparameters.

| Hyperparameter | Value |
|---|---|
| Number of timescales $S$ | 3 |
| Profile dimension $d_h$ | 64 |
| Item / KC representation dimension | $64/64$ |
| Half-life personalization rank $r_H$ | 16 |
| Episodic memory capacity $M$ | 32 |
| Episodic value width $d_e$ | 64 |
| Gate / event-encoder hidden dimension | 128 |
| Dropout probability | 0.2 |
| Optimizer | Adam |
| Learning rate | $1.0 \times 10^{-3}$ |
| Weight decay | 0 |
| Gradient clipping norm | 5.0 |
| Batch size | 128 |
| Maximum epochs | 200 |
| Early stopping | validation AUC, patience 10 |
| Minimum improvement | $10^{-3}$ |
| Random seed | 3407 |
| Half-life bounds $(H_{\min}, H_{\max})$ | $(10^{-3}, 3650)$ days |
| Personalization log-scale bound $\kappa_H$ | 0.7 |
| Consolidation initialization $c^{\mathrm{m}}$ | 0.6 |
| Consolidation initialization $c^{\mathrm{s}}$ | 0.5 |

The half-life personalization rank $r_H = 16$ controls the low-rank student–KC retention interaction. The event-conditioned convex gate uses a hidden width of 128 to combine the fast-, medium-, and slow-timescale states. The hierarchical 4PL-style item-response formulation uses knowledge-component-level parameters as its shared component and regularized item-specific residuals for item-level deviations.

**Regularization coefficients.** The training objective uses four auxiliary coefficients: $\lambda_{\mathrm{prof}}$ for profile smoothness, $\lambda_{\mathrm{evid}}$ for evidence-state changes, $\lambda_{\mathrm{gain}}$ for learning-gain changes, and $\lambda_{\mathrm{item}}$ for the item-specific residual penalty. Their values are shared across all datasets and are given in Table 7.

Table 7: Regularization coefficients used in CT-LFSSM training.

| | $\lambda_{\mathrm{prof}}$ | $\lambda_{\mathrm{evid}}$ | $\lambda_{\mathrm{gain}}$ | $\lambda_{\mathrm{item}}$ |
|---|---|---|---|---|
| All datasets | $1 \times 10^{-4}$ | $5 \times 10^{-4}$ | $5 \times 10^{-4}$ | $5 \times 10^{-4}$ |

**Dataset-specific configurations.** Only the input adapters required by the available question–KC and temporal information differ across datasets. These differences are summarized in Table 8; they do not alter the optimizer or training schedule above. This design allows CT-LFSSM to adapt to heterogeneous dataset characteristics while preserving a unified modeling framework. The dataset-specific configurations only affect information encoding and do not introduce additional modules or optimization procedures.

Table 8: Dataset-specific CT-LFSSM configuration.

| Dataset | Question-aware readout | Question–KC handling | Time input |
|---|---|---|---|
| ASSIST2009 | Enabled | normalized multi-KC question representation | $\Delta\tau = 0$; no wall-clock interval is inferred |
| ASSIST2012 | Disabled | one-to-one question–KC mapping | observed timestamps converted to elapsed days |
| Algebra2005 | Enabled | normalized multi-KC representation from native KC assignments | observed *First Transaction Time* converted to elapsed days |
| XES3G5M | Enabled | official question-level multi-KC representation | official temporal input retained |

For ASSIST2009, elapsed time is deliberately neutralized because the adopted preprocessing does not provide a reliable wall-clock timestamp; interaction order is therefore not converted into artificial time gaps. ASSIST2012 uses a one-to-one question–KC mapping, so the question-aware readout is disabled. Algebra2005 and XES3G5M retain the question-aware readout for multi-KC questions.

**Hardware.** All experiments were executed on a single NVIDIA GeForce RTX 4090 GPU with 48 GiB of device memory. No multi-GPU or heterogeneous-hardware training was used. Automatic mixed-precision training was not used.

**Reproducibility notes.** All CT-LFSSM training runs use a fixed random seed of 3407, which is passed to the common seed-initialization utility before model and training execution. We do not perform independent multi-seed repetitions of each fold; the reported experiments use the fixed seed throughout the corresponding fold runs. The resolved configuration used for each formal run is saved together with the checkpoint, allowing the effective training settings to be audited after training. The code will be released upon acceptance, and the supplementary materials include the preprocessing scripts, data-construction scripts, and the configuration files used for all reported runs.

### B.2 Dataset and Baseline Details

This section provides the dataset statistics, preprocessing conventions, and baseline coverage omitted from the main text for space.

**Dataset statistics.** Table 9 summarizes the four benchmarks under the data representations used in our experiments. For the three rerun datasets, the reported student counts refer to distinct real students rather than prepared windows or sequence segments.

Table 9: Dataset statistics. Statistics for ASSIST2009, ASSIST2012, and Algebra2005 are reported after removing sequences shorter than three interactions, with average sequence length computed before rolling-window segmentation; XES3G5M statistics follow the official benchmark. "Time" indicates usable wall-clock timestamps, and "Multi-KC" indicates whether at least one item is associated with multiple KCs.

| Dataset | Students | Items | KCs | Interactions | Avg. length | Time | Multi-KC |
|---|---|---|---|---|---|---|---|
| ASSIST2009 | 3,852 | 17,737 | 123 | 282,607 | 73.37 | No | Yes |
| ASSIST2012 | 27,485 | 53,070 | 265 | 2,709,587 | 98.58 | Yes | No |
| Algebra2005 | 574 | 173,113 | 112 | 607,014 | 1,057.52 | Yes | Yes |
| XES3G5M | 18,066 | 7,652 | 865 | 5,549,635 | 307.19 | Yes | Yes |

The adopted ASSIST2009 representation contains 3,017 multi-KC items, with up to four KCs per item. ASSIST2012 contains no multi-KC items after preprocessing, whereas Algebra2005 contains 59,532 multi-KC items and supports up to seven KCs per item. XES3G5M natively supports multi-KC questions (Liu et al., 2023c).

**Preprocessing.** For **ASSIST2009**, we use the corrected Skill Builder data from the ASSISTments ecosystem (Feng et al., 2009). The field `order_id` is used only to recover within-student interaction order and is never interpreted as elapsed time. Because the adopted data do not provide reliable wall-clock timestamps, the temporal input is neutralized as $\Delta\tau = 0$ rather than inferred from index differences. Response time, hint count, and attempt count are likewise treated as unavailable in the standardized benchmark. ASSIST2009 therefore evaluates sequential knowledge tracking but is not used as evidence for real elapsed-time forgetting.

For **ASSIST2012**, the available timestamps are retained and consecutive valid interactions determine the elapsed intervals used for elapsed-time state evolution. Each retained item is associated with a single KC under the adopted representation, so the multi-KC question-aware readout is not activated for this dataset.

**Algebra2005** is based on the Algebra I 2005–2006 data from the KDD Cup 2010 Educational Data Mining Challenge (Stamper et al., 2010). Interactions are placed in chronological order, and real elapsed intervals are derived from the recorded *First Transaction Time*. The resulting representation supports up to seven KCs per item.

**XES3G5M** is a large-scale mathematics KT benchmark containing student response timestamps and questions associated with one or multiple KCs (Liu et al., 2023c). These properties allow evaluation with both real elapsed-time information and multi-KC question representations. XES3G5M statistics are reported as in the official benchmark, as we adopt its released splits without further filtering.

For the standardized preparation of ASSIST2009, ASSIST2012, and Algebra2005, sequences shorter than three interactions are discarded, the maximum input length is 200, and prediction uses question-level rolling-window evaluation. Students are divided into a held-out 20% test set and an 80% development pool, from which five train/validation folds are formed under the pyKT-compatible protocol (Liu et al., 2022). Splits are student-disjoint, so interactions belonging to the same student are not shared across partitions. Checkpoint selection uses validation AUC only, and test data are reserved for final evaluation.

**Baselines.** The complete comparison covers 25 established KT models from complementary methodological families. **Factor-analysis / cognitive-diagnosis methods** include PFA (Pavlik et al., 2009), KTM (Vie & Kashima, 2019), and DAS3H (Choffin et al., 2019). **Recurrent methods** include DKT (Piech et al., 2015), DKT+ (Yeung & Yeung, 2018), DKT-F (Nagatani et al., 2019), and qDKT (Sonkar et al., 2020). **Memory-augmented methods** include DKVMN (Zhang et al., 2017) and SKVMN (Abdelrahman & Wang, 2019). **Graph-based methods** include GKT (Nakagawa et al., 2019) and KQN (Lee & Yeung, 2019), while ATKT (Guo et al., 2021) represents the **adversarial** family.

The **attention-based and recent deep-KT methods** comprise SAKT (Pandey & Karypis, 2019), SAINT (Choi et al., 2020), AKT (Ghosh et al., 2020), simpleKT (Liu et al., 2023b), sparseKT (Huang et al., 2023), QIKT (Chen et al., 2023), DIMKT (Shen et al., 2022), IEKT (Long et al., 2021), and AT-DKT (Liu et al., 2023a). The **learning/forgetting-aware group** contains LPKT (Shen et al., 2021), FoLiBiKT (Im et al., 2023), HawkesKT (Wang et al., 2021), and DeepIRT (Yeung, 2019).

For ASSIST2009, ASSIST2012, and Algebra2005, we rerun the 14 models appearing in the main comparison—PFA, KTM, DAS3H, DKT, DKVMN, SAKT, AKT, IEKT, DIMKT, QIKT, FoLiBiKT, LPKT, simpleKT, and HawkesKT—under the same standardized preprocessing, student splits, five-fold validation, checkpoint-selection, and test protocol as CT-LFSSM.

For XES3G5M, we use the 19 baseline results reported by the official benchmark (Liu et al., 2023c): DKT, DKT+, DKT-F, KQN, qDKT, IEKT, DIMKT, AT-DKT, QIKT, DKVMN, SKVMN, DeepIRT, ATKT, GKT, SAKT, SAINT, AKT, simpleKT, and sparseKT. These values are published benchmark results rather than reruns on our hardware. The XES3G5M comparison and the three-dataset rerun comparison are therefore treated as separate evaluation settings; officially reported baseline values are not mixed with rerun values within the same apples-to-apples comparison.

### B.3 Results on XES3G5M

Table 10 reports the performance of CT-LFSSM and the 19 baseline results published in the official XES3G5M benchmark (Liu et al., 2023c). CT-LFSSM achieves the highest AUC and ACC, improving over the strongest baseline IEKT by $0.0035$ in AUC and $0.0064$ in ACC. This result is consistent with the broader findings on the three re-benchmarked datasets and shows that the gains of CT-LFSSM extend to a large-scale benchmark under its official evaluation protocol.

Table 10: Performance on XES3G5M under its official five-fold benchmark. Best results are in **bold** and second-best results are underlined.

| Model | Ques. ID | KC ID | Timestamp | AUC | Accuracy |
|---|---|---|---|---|---|
| DKT | No | Yes | No | 0.7852±0.0006 | 0.8173±0.0002 |
| DKT+ | No | Yes | No | 0.7861±0.0002 | 0.8178±0.0001 |
| DKT-F | No | Yes | Yes | 0.7940±0.0006 | 0.8209±0.0003 |
| KQN | No | Yes | No | 0.7793±0.0006 | 0.8152±0.0002 |
| qDKT | Yes | No | No | 0.8225±0.0002 | 0.8301±0.0000 |
| IEKT | Yes | Yes | No | 0.8280±0.0002 | 0.8316±0.0002 |
| DIMKT | Yes | Yes | No | 0.8220±0.0002 | 0.8291±0.0006 |
| AT-DKT | Yes | Yes | No | 0.7932±0.0004 | 0.8198±0.0004 |
| QIKT | Yes | Yes | No | 0.8222±0.0006 | 0.8300±0.0005 |
| DKVMN | No | Yes | No | 0.7792±0.0004 | 0.8155±0.0001 |
| SKVMN | No | Yes | No | 0.7514±0.0005 | 0.8075±0.0003 |
| DeepIRT | No | Yes | No | 0.7785±0.0005 | 0.8150±0.0002 |
| ATKT | No | Yes | No | 0.7783±0.0004 | 0.8145±0.0002 |
| GKT | No | Yes | No | 0.7727±0.0006 | 0.8135±0.0004 |
| SAKT | No | Yes | No | 0.7693±0.0008 | 0.8124±0.0002 |
| SAINT | Yes | Yes | Yes | 0.8074±0.0007 | 0.8177±0.0006 |
| AKT | Yes | Yes | No | 0.8207±0.0008 | 0.8273±0.0007 |
| simpleKT | Yes | Yes | No | 0.8163±0.0006 | 0.8246±0.0005 |
| sparseKT | Yes | Yes | No | 0.8165±0.0015 | 0.8234±0.0009 |
| **CT-LFSSM** | Yes | Yes | Yes | **0.8315±0.003** | **0.8380±0.002** |

### B.4 Full Sequence-Length-Stratified Results

For completeness, we provide the full sequence-length-stratified results with both AUC and ACC on ASSIST2012 and ASSIST2009 in Tables 11 and 12, respectively. Sequence lengths are measured on the original student histories before rolling-window segmentation, and results are reported as mean $\pm$ population standard deviation over five folds. The results complement the AUC-only analysis presented in the main text by providing a complete view of model performance across different history lengths. CT-LFSSM consistently achieves competitive performance across sequence-length groups on both benchmarks, demonstrating its robustness to varying amounts of student interaction history.

Table 11: Full sequence-length-stratified performance on ASSIST2012. Sequence length is measured on the original student history before rolling-window segmentation. Results are reported as mean $\pm$ population standard deviation over five folds. Best results are in **bold** and second-best results are underlined.

| | $(0, 10]$ | | $(10, 50]$ | | $(50, 100]$ | | $(100, 200]$ | | $(200, +\infty)$ | |
|---|---|---|---|---|---|---|---|---|---|---|
| Model | AUC | ACC | AUC | ACC | AUC | ACC | AUC | ACC | AUC | ACC |
| PFA | 0.7165±0.0018 | 0.7184±0.0020 | 0.7008±0.0009 | 0.7195±0.0012 | 0.6983±0.0008 | 0.7111±0.0004 | 0.6956±0.0010 | 0.6997±0.00015 | 0.6839±0.0003 | 0.6917±0.00005 |
| KTM | 0.7697±0.0031 | 0.7308±0.0033 | 0.7618±0.0021 | 0.7398±0.0019 | 0.7627±0.0003 | 0.7501±0.0003 | 0.7696±0.0008 | 0.7413±0.0007 | 0.7326±0.0010 | 0.7400±0.0010 |
| DAS3H | 0.7701±0.0008 | 0.7355±0.0010 | 0.7631±0.0010 | 0.7427±0.0006 | 0.7603±0.0010 | 0.7494±0.0015 | 0.7735±0.0003 | 0.7459±0.0004 | 0.7229±0.0007 | 0.7388±0.0010 |
| DKT | 0.7584±0.0009 | 0.7231±0.0011 | 0.7332±0.0005 | 0.7286±0.0005 | 0.7331±0.0004 | 0.7430±0.0008 | 0.7426±0.0003 | 0.7294±0.0005 | 0.7262±0.0004 | 0.7352±0.0003 |
| DKVMN | 0.7533±0.0011 | 0.7176±0.0022 | 0.7295±0.0011 | 0.7279±0.0004 | 0.7304±0.0005 | 0.7427±0.0005 | 0.7418±0.0006 | 0.7299±0.0006 | 0.7257±0.0002 | 0.7355±0.0002 |
| SAKT | 0.7224±0.0019 | 0.7138±0.0019 | 0.7173±0.0005 | 0.7204±0.0013 | 0.7135±0.0010 | 0.7336±0.0010 | 0.7190±0.0011 | 0.7181±0.0010 | 0.6989±0.0004 | 0.7244±0.0002 |
| AKT | **0.7770±0.0019** | 0.7396±0.0021 | 0.7704±0.0004 | 0.7466±0.0007 | 0.7766±0.0004 | **0.7621±0.0005** | 0.7864±0.0005 | 0.7529±0.0004 | 0.7818±0.0005 | 0.7613±0.0004 |
| IEKT | 0.7701±0.0013 | 0.7365±0.0015 | 0.7689±0.0025 | 0.7503±0.0028 | 0.7698±0.0011 | 0.7561±0.0006 | 0.7801±0.0006 | 0.7512±0.0010 | 0.7808±0.0006 | 0.7609±0.0008 |
| DIMKT | 0.7762±0.0018 | 0.7432±0.0022 | 0.7665±0.0013 | 0.7441±0.0010 | 0.7673±0.0015 | 0.7579±0.0008 | 0.7736±0.0017 | 0.7452±0.0011 | 0.7662±0.0021 | 0.7519±0.0012 |
| QIKT | 0.7740±0.0017 | 0.7378±0.0016 | 0.7633±0.0005 | 0.7444±0.0005 | 0.7649±0.0006 | 0.7552±0.0005 | 0.7726±0.0004 | 0.7437±0.0007 | 0.7646±0.0004 | 0.7510±0.0002 |
| FoLiBiKT | 0.7732±0.0022 | 0.7395±0.0030 | 0.7707±0.0007 | 0.7484±0.0014 | 0.7752±0.0004 | 0.7610±0.0006 | 0.7836±0.0004 | 0.7517±0.0007 | 0.7806±0.0009 | 0.7604±0.0010 |
| LPKT | 0.7681±0.0018 | 0.7322±0.0021 | 0.7643±0.0008 | 0.7449±0.0014 | 0.7712±0.0007 | 0.7593±0.0010 | 0.7786±0.0007 | 0.7477±0.0006 | 0.7741±0.0005 | 0.7563±0.0015 |
| simpleKT | 0.7744±0.0007 | 0.7389±0.0019 | 0.7676±0.0004 | 0.7428±0.0009 | 0.7710±0.0004 | 0.7561±0.0010 | 0.7806±0.0004 | 0.7461±0.0010 | 0.7743±0.0004 | 0.7541±0.0006 |
| HawkesKT | 0.7601±0.0010 | 0.7139±0.0028 | 0.7471±0.0006 | 0.7340±0.0009 | 0.7484±0.0010 | 0.7464±0.0009 | 0.7570±0.0004 | 0.7349±0.0007 | 0.7450±0.0008 | 0.7431±0.0002 |
| **CT-LFSSM** | 0.7763±0.0020 | **0.7433±0.0030** | **0.7806±0.0004** | **0.7550±0.0008** | **0.7837±0.0006** | 0.7607±0.0006 | **0.7928±0.0005** | **0.7605±0.0004** | **0.7903±0.0005** | **0.7661±0.0003** |

Table 12: Sequence-length-stratified performance on ASSIST2009. Sequence length is measured on the original student history before rolling-window segmentation. Results are reported as mean $\pm$ population standard deviation over five folds. Best results are in **bold** and second-best results are underlined.

| | $(0, 10]$ | | $(10, 50]$ | | $(50, 100]$ | | $(100, 200]$ | | $(200, +\infty)$ | |
|---|---|---|---|---|---|---|---|---|---|---|
| Model | AUC | ACC | AUC | ACC | AUC | ACC | AUC | ACC | AUC | ACC |
| PFA | 0.7225±0.0016 | 0.7247±0.0010 | 0.7201±0.0023 | 0.7173±0.0015 | 0.7139±0.0053 | 0.6993±0.0044 | 0.7170±0.0021 | 0.7009±0.0013 | 0.6993±0.0041 | 0.6981±0.0055 |
| KTM | 0.7881±0.0025 | 0.7488±0.0023 | 0.7812±0.0010 | 0.7159±0.0019 | 0.7793±0.0015 | 0.7217±0.0030 | 0.7399±0.0018 | 0.7080±0.0020 | 0.7217±0.0012 | 0.7230±0.0029 |
| DAS3H | 0.7891±0.0008 | 0.7468±0.0010 | 0.7752±0.0024 | 0.7099±0.0038 | 0.7815±0.0005 | 0.7235±0.0006 | 0.7425±0.0010 | 0.7102±0.0015 | 0.7269±0.0028 | 0.7249±0.0040 |
| DKT | 0.7874±0.0029 | 0.7433±0.0035 | 0.7641±0.0015 | 0.7122±0.0034 | 0.7817±0.0010 | 0.7277±0.0020 | 0.7473±0.0017 | 0.7106±0.0017 | 0.7311±0.0010 | 0.7298±0.0015 |
| DKVMN | 0.7699±0.0028 | 0.7494±0.0036 | 0.7582±0.0019 | 0.7111±0.0029 | 0.7734±0.0016 | 0.7212±0.0037 | 0.7439±0.0010 | 0.7061±0.0008 | 0.7235±0.0006 | 0.7256±0.0026 |
| SAKT | 0.7633±0.0054 | 0.7417±0.0079 | 0.7522±0.0015 | 0.7075±0.0023 | 0.7606±0.0045 | 0.7083±0.0053 | 0.7260±0.0036 | 0.6926±0.0025 | 0.7077±0.0010 | 0.7190±0.0010 |
| AKT | 0.7893±0.0045 | **0.7603±0.0050** | 0.7902±0.0027 | 0.7362±0.0026 | 0.8036±0.0037 | 0.7411±0.0030 | **0.7724±0.0034** | 0.7284±0.0015 | 0.7750±0.0054 | 0.7445±0.0017 |
| IEKT | 0.7835±0.0012 | 0.7501±0.0021 | 0.7785±0.0008 | 0.7293±0.0012 | 0.7900±0.0017 | 0.7301±0.0018 | 0.7598±0.0010 | 0.7178±0.0012 | 0.7501±0.0019 | 0.7347±0.0011 |
| DIMKT | 0.7829±0.0042 | 0.7494±0.0056 | 0.7780±0.0039 | 0.7244±0.0043 | 0.7873±0.0012 | 0.7293±0.0012 | 0.7555±0.0011 | 0.7179±0.0030 | 0.7482±0.0022 | 0.7336±0.0015 |
| QIKT | **0.8006±0.0059** | 0.7508±0.0039 | 0.7833±0.0024 | 0.7229±0.0011 | 0.7973±0.0008 | 0.7337±0.0018 | 0.7590±0.0012 | 0.7187±0.0007 | 0.7767±0.0037 | 0.7430±0.0006 |
| FoLiBiKT | 0.7731±0.0062 | 0.7510±0.0048 | 0.7817±0.0012 | 0.7303±0.0021 | 0.7982±0.0025 | 0.7380±0.0025 | 0.7602±0.0020 | 0.7189±0.0026 | 0.7783±0.0055 | 0.7441±0.0026 |
| LPKT | 0.7844±0.0051 | 0.7424±0.0067 | 0.7734±0.0010 | 0.7175±0.0018 | 0.7928±0.0027 | 0.7275±0.0014 | 0.7577±0.0023 | 0.7172±0.0018 | 0.7802±0.0060 | 0.7459±0.0029 |
| simpleKT | 0.7879±0.0054 | 0.7489±0.0030 | 0.7778±0.0014 | 0.7251±0.0028 | 0.7932±0.0007 | 0.7297±0.0014 | 0.7583±0.0011 | 0.7164±0.0024 | 0.7620±0.0032 | 0.7332±0.0018 |
| HawkesKT | 0.7478±0.0048 | 0.7087±0.0042 | 0.7385±0.0014 | 0.6886±0.0012 | 0.7579±0.0025 | 0.7016±0.0024 | 0.7247±0.0029 | 0.6952±0.0006 | 0.7142±0.0015 | 0.7142±0.0027 |
| **CT-LFSSM** | 0.7905±0.0040 | 0.7521±0.0027 | **0.7905±0.0013** | **0.7366±0.0032** | **0.8081±0.0013** | **0.7439±0.0014** | 0.7704±0.0007 | **0.7294±0.0021** | **0.7909±0.0044** | **0.7553±0.0026** |

## C Additional Visualizations

### C.1 Learned Half-Lives

Figure 3 shows the distribution of the half-lives learned by CT-LFSSM after training. Panel (a) reports the base half-lives $H_{k,s}^{(0)}$ for every knowledge component at the fast, medium, and slow timescales on a logarithmic scale. The three groups are clearly separated by several orders of magnitude, and no learned value touches the numerical bounds $H_{\min}$ or $H_{\max}$, so the positive-increment parameterization is not saturated at either boundary. Panel (b) reports the personalized half-lives $H_{u,n,k,s}$ for a random sample of student–KC pairs, with each line connecting the three timescales of the same pair. All curves increase monotonically from fast to slow, confirming that the shared positive factor $\phi_{u,n,k}$ preserves the relative ordering established by the base half-lives. The spread of each box across student–KC pairs also confirms that personalization produces meaningful variation instead of collapsing to a single constant.

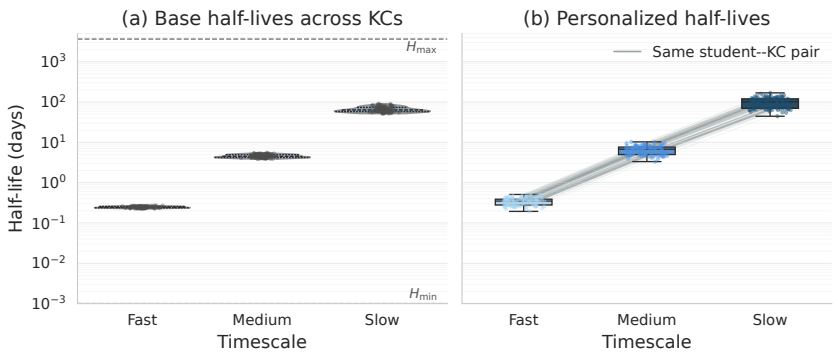



Figure 3: Learned half-lives across knowledge components and timescales. (a) Base half-lives of each KC at the fast, medium, and slow timescales, shown on a logarithmic scale. The dashed lines mark the numerical bounds $H_{\min}$ and $H_{\max}$. (b) Personalized half-lives for a random sample of student–KC pairs, with lines connecting the three timescales of the same pair. The fast-to-slow ordering is preserved after personalization, and all personalized values remain within the configured bounds.

### C.2 Gate Weights over Elapsed Time

Figure 4 shows how the event-conditioned convex gate distributes weight across the three timescales as a function of the observed elapsed time on the ASSIST2012 validation set. At short intervals, the gate assigns comparable weight to all three timescales. This is consistent with the fact that the three latent states have undergone little decay and are therefore similarly informative. As the elapsed time grows, the gate systematically shifts weight away from the fast timescale and toward the slow timescale. By the scale of weeks, the slow timescale dominates, which is consistent with the intended behavior that long interaction-free periods are handled primarily by states with long retention. The medium timescale remains between the two extremes throughout, and the three weights sum to one at every elapsed-time bin.

The observed pattern provides direct empirical evidence that the gate is not a fixed mixing layer. Instead, it adapts to the observed elapsed time and progressively delegates the pre-response mastery estimate to states that remain informative at the corresponding temporal scale.

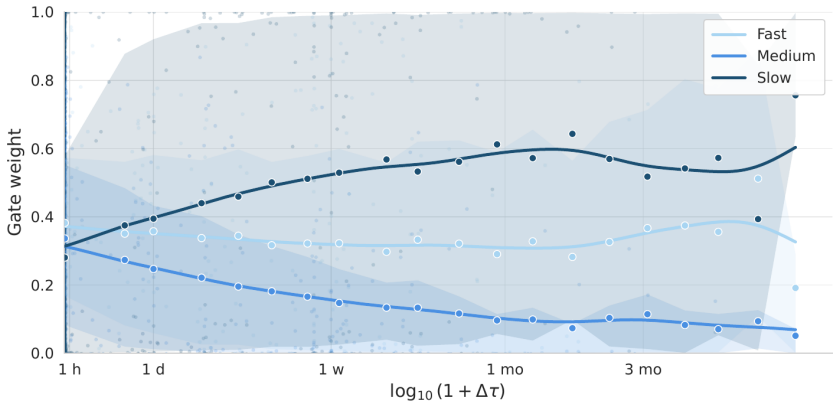



Figure 4: Gate weights as a function of elapsed time on ASSIST2012. Each curve shows the mean gate weight assigned to one timescale within each elapsed-time bin, with shaded interquartile ranges. The gate shifts weight from the fast timescale to the slow timescale as the elapsed time grows, while the medium timescale remains between them.

### C.3 KNOWLEDGE-STATE TRAJECTORY CASE STUDY

Figure 5 shows the fast-, medium-, and slow-timescale latent states for a single student across a real interaction sequence from ASSIST2012, together with the predicted probabilities and the observed responses.

The three timescales exhibit clearly different dynamics. The fast timescale reacts strongly to individual responses, rising after a correct response and dropping after an incorrect one, and it also decays quickly during long interaction-free periods. The slow timescale changes much more gradually and remains close to its previous level even after extended intervals, which reflects its role in retaining long-term knowledge. The medium timescale lies between the two extremes and shows both localized reactions and slower drift. The predicted probabilities track the combined contribution of all three states, and the residual between each prediction and the observed response is the quantity that drives the post-response update described in Section 4.2.3.

The three representative elapsed intervals marked at the top of the figure ($\Delta\tau = 0.3$ min, 2.9 days, and 40 days) span the within-session, within-day, and multi-day regimes. Across these regimes, the fast state increasingly decays while the slow state remains informative, which is the behavior targeted by the personalized exponential decay and the event-conditioned convex gate.

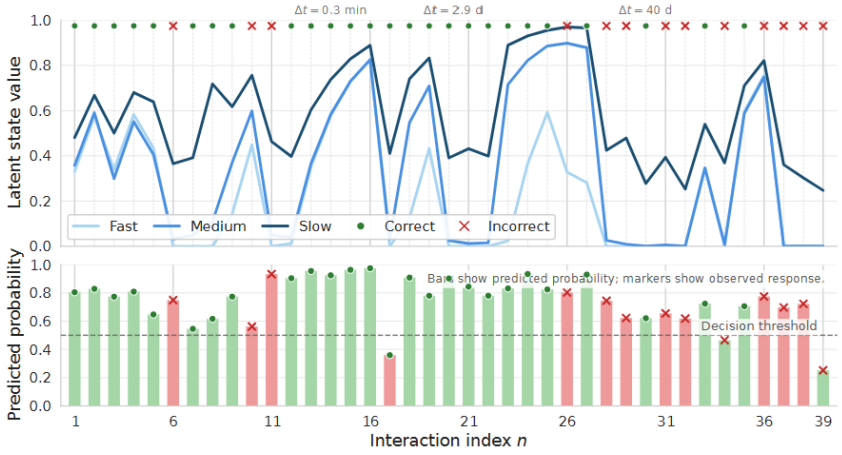



Figure 5: Knowledge-state trajectory for a single student on ASSIST2012. Top: fast-, medium-, and slow-timescale latent states across a real interaction sequence, with vertical markers indicating correct and incorrect responses. Bottom: predicted correctness probabilities and observed responses, with a dashed line at the 0.5 decision threshold. The three timescales evolve at different rates, and the fast state decays most rapidly during extended interaction-free periods.

### C.4 EVIDENCE STRENGTH ACROSS TIMESCALES

Figure 6 shows the learned KC-level evidence strengths used in response-driven belief revision, evaluated after training on ASSIST2012. For each knowledge component, the fast-timescale evidence strength is consistently the largest and the slow-timescale evidence strength the smallest, so the constraint $e_{k,\mathrm{f}} \geq e_{k,\mathrm{m}} \geq e_{k,\mathrm{s}} \geq 0$ is satisfied across all KCs. The gray lines connecting the three timescales of the same KC show that this ordering holds per KC rather than only on average. No value reaches the configured upper bound $E_{\max} = 1.25$, which indicates that the evidence strengths are not saturated and remain sensitive to subsequent gradient updates.

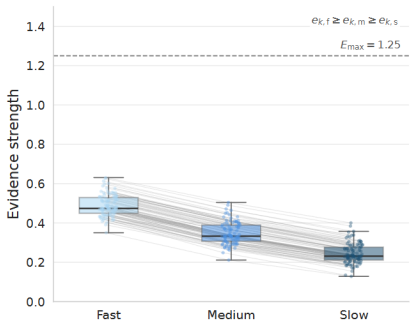



Figure 6: KC-level evidence strength across timescales on the ASSIST2012 validation set. Each point corresponds to one knowledge component, and lines connect the three timescales of the same KC. The ordering fast $\geq$ medium $\geq$ slow is satisfied for every KC, and no value reaches the configured upper bound $E_{\max} = 1.25$.