Diagnosing and Repairing Long-Horizon Failure in Transformer Queueing Simulators

Anonymous authors
Paper under double-blind review

Abstract

Neural event simulators are evaluated on short test histories but deployed over much longer rollouts, where accurate one-step predictions need not ensure faithful trajectories. Standard test loss leaves predictions beyond the training length unexamined. We investigate whether predictions on shared long reference histories can diagnose these failures and guide corrective supervision. In an M/M/1 queue, we evaluate 46 transformers trained on 400-event sequences using 256 rollouts of 40,000 events per model, and compute deep-prefix drift (DPD), the mean predicted queue increment beyond the training length, on 1,024 simulator-generated reference histories. Across 6 failing checkpoints and three fine-tuning seeds, we compare deep supervision with matched shallow supervision and continued training, and on three of them compare exact conditional targets with fresh simulator draws and recorded continuations. Despite near-oracle short-history test loss, models cross a fixed queue boundary on 0–255 of 256 rollouts, with 99.4% of first crossings occurring beyond the training length. On 36 evaluation models, DPD ranks crossing counts at Spearman +0.82 and flags 11 of 12 severe failures with 1 false positive. DPD requires neither candidate-model rollouts nor exact conditional probabilities, using model predictions on shared simulator-generated reference histories. Deep supervision produces fewer crossings than both controls in all 18 model-by-seed comparisons, by at least 37 of 256 rollouts, and fresh simulator targets outperform both controls in all 9 tested comparisons, while recorded continuations leave at least 199 more crossings than exact targets. These findings support shared long-history evaluation as a parallel diagnostic for learned event simulators and show that supervision beyond the training length can reduce failures hidden by short-history test loss.

1 Introduction

A practitioner choosing among neural event simulators faces a costly decision. Loss on short test histories is cheap to measure, whereas long-rollout behavior can only be judged by generating long sequences from every candidate, one event at a time. The two criteria can disagree sharply. In a single-server queue with exponential interarrival and service times (M/M/1), 46 transformers trained with the same recipe have event-type test cross-entropies (CE) within 1.53×10^{-4} nats of the exact queueing law's, yet the number of 40,000-event rollouts in which the queue exceeds 50 customers ranges from 0 to 255 out of 256 (Table 1 shows ten of them). Test loss, computed on sequences no longer than the training sequences, cannot score predictions at the longer contexts that rollouts reach. We ask whether such differences can be detected before rollout and can direct corrective supervision.

The gap between one-step prediction and trajectory quality is familiar from language generation, event forecasting and learned physical simulation (Arora et al., 2022; Karpukhin et al., 2026; Sanchez-Gonzalez et al., 2020). Deployment changes two things at once for a simulator trained on finite sequences. Its inputs become its own samples, the setting of exposure bias (Bengio et al., 2015; Arora et al., 2022), and its predictions are made at

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.

Seed	Test CE	Crossed	DPD	Seed	Test CE	Crossed	DPD
4	0.61243	0	−0.202	6	0.61235	194	+0.094
2	0.61238	2	−0.103	3	0.61240	215	+0.168
0	0.61239	27	−0.108	1	0.61241	240	+0.029
5	0.61247	41	−0.085	8	0.61238	245	+0.019
9	0.61240	160	+0.002	7	0.61243	255	+0.442
Oracle: test CE 0.61233, 2.86 crossings expected, drift −0.111							

Table 1: **Each of these ten models' test cross-entropy lies within 1.45×10^{-4} nats of the oracle's, while crossing counts span 0 to 255 of 256 rollouts.** Ten of the 46 models (seeds 0–9), ordered by crossing count. Test CE: event-type cross-entropy in nats on one shared test set of 400-event trajectories. Crossed: rollouts of 40,000 events in which the queue exceeds 50 (§3.1); the oracle's expected count comes from 10^5 trajectories. DPD: the mean queue increment predicted beyond the training length (§4.2); positive values predict growth.

positions that training never scored, the setting of length extrapolation (Press et al., 2022). Our diagnostic isolates the second change: it reads predictions beyond the training length on reference histories held fixed across models.

Queueing systems make this examination precise. The queue length, recovered from the event sequence, is the running sum of the generated arrivals and departures, so per-event errors accumulate in an observable state, and it determines the true next-event law, the *oracle*, exactly. The M/M/1 queue thus serves as a measurement instrument; learned simulators are fitted to systems without a closed-form law, so the diagnostic reads only model predictions on simulator-generated histories and the repair also works with simulator draws. We study the transformer event simulator of Mittal et al. (2025), trained on 400-event sequences, and record whether its queue crosses a fixed boundary during rollouts of 40,000 events, which the true process does with probability 0.011.

The experiments ask where the failures arise, whether they can be detected before rollout, and whether supervision at those contexts reduces their crossings. Our contributions are:

- **A failure that test loss cannot see, located beyond the training length.** Test CE separates severe failures from the rest at chance (AUROC 0.51 on the 36 evaluation models), because a probability error moves the queue's drift linearly and the loss quadratically (§3). Almost every crossing begins at a position no training sequence reaches, and 0 of 30 models with alternative position schemes fail severely, against 18 of 46 with the absolute sinusoidal encoding (§4.1).

- **A diagnostic read before any rollout.** *Deep-prefix drift* (DPD), the mean queue increment a model predicts beyond its training length on shared reference histories, ranks crossing counts at Spearman +0.82 on the 36 models trained after it was fixed, and its sign flags 11 of their 12 severe failures with no fitted threshold. Its AUROC exceeds that of KL divergence from the oracle by 0.04, a difference not resolved at 36 models. The diagnostic evaluates shared reference histories in parallel, without generating candidate-model rollouts or accessing exact conditional probabilities (§4.2).

- **A repair located by the diagnostic.** Supervising positions beyond the training length reduces crossings more than equal supervision within it and more than continued training in all 18 model-by-seed comparisons, by at least 37 of 256 rollouts, and brings all 9 severe failures it was applied to from 178–255 crossings to at most 28. Simulator draws serve as targets, and the histories' own recorded next events leave at least 199 more crossings with a test-loss change below its spread across models (§4.3).

2 Related work

Learned simulators must turn locally accurate predictions into reliable trajectories. This requirement connects our work to sequential prediction, neural event modeling and length generalization: a model trained on short reference histories must remain accurate both when its inputs are generated by itself and when its context extends beyond the positions encountered during training. These are distinct changes, and improvements in one-step prediction need not address both.

Our study connects diagnosis and intervention through a shared set of long reference histories. These histories allow us to compare models on identical inputs beyond the training length, summarize their predicted queue increments, and test whether supervision at those positions improves subsequent rollouts. The relevant gap is therefore how to connect a measurable prediction error outside training coverage to long-horizon stochastic behavior and an effective corrective intervention.

From one-step accuracy to reliable rollouts. Sequential prediction research establishes that a learner's decisions change the inputs it subsequently encounters, allowing local errors to accumulate. Imitation-learning reductions and dataset aggregation address this distribution shift through expert supervision on learner-visited states, while scheduled sampling and Professor Forcing bring training inputs or hidden-state dynamics closer to those encountered during generation (Ross & Bagnell, 2010; Ross et al., 2011; Bengio et al., 2015; Lamb et al., 2016). Related strategies in learned physical simulation use noisy inputs, predictions from model-generated states and multistep objectives to improve trajectory accuracy (Sanchez-Gonzalez et al., 2020; Brandstetter et al., 2022; Lam et al., 2023). Together, these approaches establish the value of training for the conditions encountered during rollout.

The severity and form of rollout error nevertheless depend on the task and measurement. Studies of language generation report both self-recovery and accumulated errors missed by next-token metrics, while deployment stress tests reveal different behavior among models with similar held-out accuracy (He et al., 2021; Arora et al., 2022; D'Amour et al., 2022). Mechanism-directed approaches such as PDE-Refiner further show how identifying a particular prediction deficiency can guide corrective training (Lippe et al., 2023). Our study examines an additional deployment change: predictions beyond the training length. Shared reference histories expose this change without differences in model-generated inputs, and controlled supervision tests whether those positions offer an effective intervention.

Event forecasting and stochastic simulation. Neural temporal point processes learn event timing and event types through recurrent or attention-based history encoders. This literature has improved conditional likelihood and predictive accuracy, while comparative evaluations show that architecture, calibration and sequence diversity remain important challenges (Mei & Eisner, 2017; Zuo et al., 2020; Bosser & Ben Taieb, 2023; Karpukhin et al., 2026). Whole-sequence approaches, including diffusion-based generation and joint event prediction, offer alternatives to autoregressive forecasting and improve forecast quality or diversity on their evaluated tasks (Lüdke et al., 2023; Karpukhin & Savchenko, 2026). These advances motivate evaluating generated sequences beyond their one-step likelihood.

Queueing simulation makes that evaluation particularly concrete: generated events change an observable state whose conditional law and long-run behavior can be checked. Closest to our work, Mittal et al. (2025) learn event types and waiting times with autoregressive transformers for simulation, uncertainty quantification and counterfactual evaluation, while related queueing surrogates predict queue-length distributions (Baron et al., 2024; Sherzer et al., 2025) or learn arrival processes (Zheng et al., 2025). We investigate this architecture over much longer rollouts and connect its conditional predictions to queue-boundary crossing risk. Whereas forecasting evaluates future event sequences over a specified horizon, our question concerns whether repeated learned transitions preserve stochastic system behavior, and whether failures can be screened before generating those long trajectories.

Length generalization and corrective supervision. Length-generalization methods address the positional mismatch between short training sequences and longer deployment

contexts. Relative attention biases, removing explicit positional encodings and varying training positions improve extrapolation in language or algorithmic tasks (Press et al., 2022; Kazemnejad et al., 2023; Ruoss et al., 2023). Position Interpolation and YaRN extend usable language-model contexts through positional rescaling and fine-tuning (Chen et al., 2023; Peng et al., 2024). These methods establish positional treatment as an important design choice; our alternative-encoding experiments examine its consequences for queueing rollouts.

Our diagnostic and repair address the behavior of an already-trained simulator. DPD reads predicted queue increments directly from output probabilities on long reference histories, requiring neither a fitted representation probe nor oracle probabilities. Repair then supplies conditional targets at those deep positions while retaining the positional encoding. Comparing against matched shallow supervision and continued training tests the value of supervision placement, and simulator-sampled targets test whether the repair requires a closed-form conditional law. This connects length extrapolation to both a model-level diagnostic and an intervention.

3 Methodology

A neural event simulator with parameters ϕ replaces the event logic of a reference simulator with a learned transition. Given the history of previous event types and waiting times, it predicts a distribution for the next waiting time (here exponential, with a learned rate) and then, given that time, a distribution for the next event type. It is trained by maximum likelihood on reference trajectories and generates histories by sampling them autoregressively.

The queue and its exact reference. In M/M/1, customers arrive at rate λ and a single server completes service at rate ν; interarrival and service times are exponential. We use $\nu = 1$ and load $\rho = \lambda/\nu = 0.8$. The queue length q_n before event n is recovered from the event tokens: arrivals add one, departures subtract one, and a departure sampled at $q_n = 0$ is recorded as an invalid event and leaves the state at zero (at most 1×10^{-5} of generated events; Appendix A.1). At queue length q, write $r(q)$ for the total event rate, T for the next waiting time and $\mathbf{1}\{q > 0\}$ for the non-empty-state indicator. The oracle is the exact conditional law

$$r(q) = \lambda + \nu\,\mathbf{1}\{q > 0\}, \qquad p(\text{departure} \mid q) = \frac{\nu\,\mathbf{1}\{q > 0\}}{r(q)}, \qquad T \mid q \sim \text{Exp}(r(q)),$$

under which event type and waiting time are conditionally independent. It supplies both a reference for evaluating predictions and targets for corrective supervision. At every non-empty state, the departure probability is .5556, so the expected change in queue length per event, which we call the *drift*, is -0.111: the true queue is pulled back toward empty.

The queue length is the running sum of these increments, so per-event errors accumulate in it. A bias δ in the predicted departure probability at non-empty states shifts the drift by -2δ on every event of a busy period, and a downward bias of half the drift's magnitude cancels the pull toward empty. The same bias raises the expected event-type cross-entropy by only about $\delta^2/[2p(1-p)]$, with $p = .5556$: drift is first order in the error and loss second order. Since the probability of reaching a distant boundary decays exponentially at a rate set by the drift (the gambler's ruin), a bias that barely moves the loss can multiply the crossing rate.

Models and training. We reproduce the decoder-only transformer of Mittal et al. (2025): 8 layers, 4 heads, width 64 and 665k parameters. Each event enters as two tokens, a Time2Vec embedding (Kazemi et al., 2019) of its waiting time and a learned embedding of its type. Position is a sinusoidal function of the event index, computed in fp32 at every index, so positions at or beyond the training length receive encodings never seen in training. Each model is trained on 10^4 trajectories of $L = 400$ events from an empty queue with Adam for 300 epochs under fp16 autocast, and a training seed fixes both its initialization and its training set; all evaluation runs in fp32 (Appendix A.1, which also reports the waiting-time log-likelihood). We train 46 models with this recipe, identical except for the training seed

(Appendix Table 5 lists them). The diagnostic was developed on ten of them (seeds 0–9) and evaluated on the other 36, trained afterwards. Diagnostic comparisons use these 36 evaluation models; results pooled over all 46 are a labelled companion. A *checkpoint* is one trained model. Event positions are indexed from zero within a context. Positions below L, which training covers, are *shallow*; positions at or beyond L, which require length extrapolation, are *deep*. Event-type cross-entropy (CE), the mean negative log probability of the observed event type, scores every model on one shared test set of 1,000 oracle trajectories of 400 events.

3.1 Rollout evaluation

Regeneration-anchored rollouts. A *busy period* is a stretch of events between returns to an empty queue. The true process restarts independently of its past at each return, so clearing the model's context there and beginning a new sequence preserves the law being simulated; we call such rollouts *anchored*. A model therefore reaches a deep position only when a busy period outlasts the training length. To bound memory, queue and context also restart after 4,096 events without a return, a cap that ends at most 0.12% of any model's contexts and leaves the oracle's crossing probability essentially unchanged (Appendix A.2).

Each model generates 256 rollouts of 40,000 events from an empty queue. Models evaluated under the same protocol share their random numbers, which pairs rollouts across checkpoints. A rollout *crosses* if its queue length ever exceeds $q_{\text{esc}} = 50$, a fixed boundary; a crossing marks a large excursion, which may be followed by a return to empty. On 10^5 oracle trajectories the crossing probability is 0.011, or 2.86 expected crossings in 256 rollouts. We call a model a *severe failure* when at least 128 of its 256 rollouts cross.

Crossing counts saturate for the worst models, so we also use the *relative crossing rate* A^*, the per-event crossing rate $-\log[1 - F(h)]/h$ of a model's rollouts, with $F(h)$ the fraction crossed by event h, divided by the oracle's. The model's rate is read at the first horizon h^* by which half its rollouts have crossed (or at $h_O = 40{,}000$ if fewer do), the oracle's at h_O (Appendix A.3).

3.2 Deep-prefix drift

To compare predictions at deep positions on identical inputs, we evaluate every checkpoint on one bank $\mathcal{B}$ of 1,024 oracle busy periods, each lasting at least 800 events and truncated to that length, fed as inputs (teacher forcing); a model's own rollouts would confound its predictions with the histories it generated.

DPD reads the quantity the queue integrates. At a non-empty state, a model that predicts departure probability $\hat{p}$ predicts a mean queue increment of $1 - 2\hat{p}$, whose true value is the drift -0.111. Let i index a reference history in $\mathcal{B}$, t its event position and $q_t^{(i)}$ its queue length, and let $\hat{p}_{i,t}$ be the departure probability that the model with parameters ϕ predicts after reading that position's waiting time. *Deep-prefix drift* is the mean predicted increment over the set $\mathcal{I}$ of deep, non-empty positions:

$$\text{DPD}(\phi) = \frac{1}{|\mathcal{I}|} \sum_{(i,t)\in\mathcal{I}} (1 - 2\hat{p}_{i,t}), \qquad \mathcal{I} = \{(i,t) : i \in \mathcal{B},\ t \geq L,\ q_t^{(i)} > 0\}.$$

A positive DPD means that the model predicts growth on histories where the true queue is pulled toward empty. Computing DPD requires only the model's probabilities, the reference histories and the queue state recovered from their events; it uses no oracle targets.

4 Experiments

4.1 Failures beyond the training length

Table 1 shows ten of the 46 models. Across all 46, test CE lies 2.1×10^{-5} to 1.53×10^{-4} nats above the oracle's 0.61233 on the same test set: each model predicts training-length histories almost exactly as well as the true law. Yet they cross on 0 to 255 of 256 rollouts, against

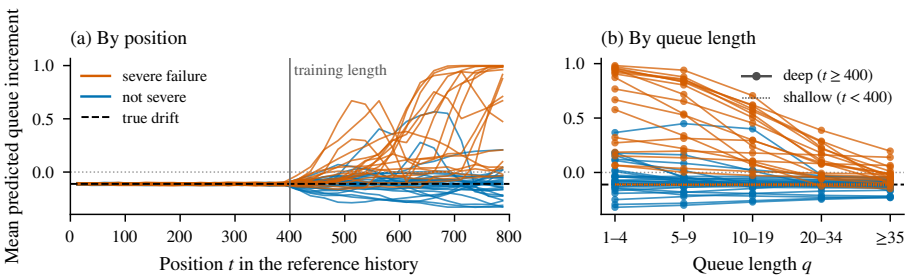


Figure 1: **Models track the true drift up to the training length and depart only beyond it, most strongly at short queues.** Mean predicted queue increment $1 - 2\hat{p}$ on the 1,024 shared oracle busy periods, one line per model (all 46), coloured by severe failure (at least 128 of 256 rollouts crossing). (a) By position t, in 25-event bins; the mix of queue lengths changes with t. (b) By queue length, over deep (solid) and shallow (dotted) positions. Dashed: true drift; grey dotted: zero.

2.86 expected under the oracle, and 18 of them are severe failures. Test CE separates the severe failures from the rest at chance (AUROC 0.51 on the 36 evaluation models, §4.2). It scores only shallow positions, and it is second order in a probability error (§3).

Where crossings begin. For each crossing rollout of the 46 models we record the position, within its current context, at which the queue first exceeds 50. Of the 4,343 first crossings, 99.4% occur at deep positions. The oracle's crossings are different: 63% of them begin within the training length, and a busy period outlasts 400 events with probability only 5×10^{-4} under the true process. The models' crossings are thus long excursions that begin in contexts no 400-event training sequence contains.

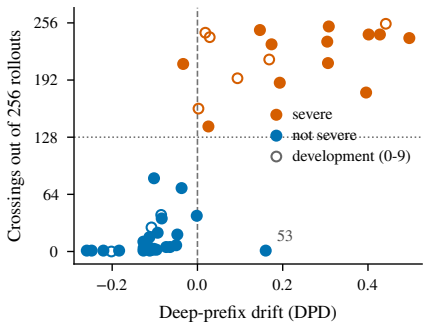
Removing long contexts brings the counts to the oracle's scale. Holding the weights of four models fixed (seeds 0, 3, 1 and 7), we restart queue and context after 400 events, so that no position reaches the training length. Their counts fall from 27–255 to 2–3 out of 256, against 1.8 expected for the oracle under the same restart (Appendix A.2). The restart also alters the process being simulated, lowering the oracle's crossing probability from 1.13% to 0.72%, so this experiment locates the excess in long contexts, and §4.2 examines what the models predict there.

Alternative position schemes reduce severe failures. The models above encode the absolute event index sinusoidally. We trained ten models each with three alternatives at the same training length: attention with linear biases (ALiBi; Press et al., 2022), no positional encoding (NoPE; Kazemnejad et al., 2023), and the sinusoidal encoding trained on a mix of consecutive and spread-out indices, so that indices beyond 400 occur in training. None of these 30 models fails severely, against 18 of the 46 sinusoidal models (two-sided Fisher exact $p = 4 \times 10^{-5}$), and none has positive DPD; 23 of them still cross more often than the oracle (at most 98 of 256), so a smaller long-horizon error remains. Training the sinusoidal recipe on 800-event sequences leaves 3 of 8 models severe (Appendix A.2).

4.2 Diagnosing failures with deep-prefix drift

Predictions beyond the training length. In Figure 1, the 46 models' mean predicted increments stay near the true -0.111 over shallow positions (-0.124 to -0.101) and spread from -0.260 to $+0.497$ over deep ones. Within each queue-state stratum, the departure begins at the training length (over positions 350–399 each model stays within 0.014 of its shallow mean), and for every severe failure the deep error is largest at queue lengths below 10 (Appendix A.3).

Predicting crossing risk. We evaluate DPD on the 36 models trained after it was fixed on the first 10 (Appendix Table 5 lists all 46). DPD ranks their crossing counts at Spearman



	Statistic	ρ	AUROC
Oracle-free	DPD (deep drift, bank)	+0.82	0.99
	Deep CE on bank events	+0.45	0.92
	Ordinary-history drift	+0.73	0.95
	Ordinary-history CE	+0.53	0.93
Oracle-informed	|DPD − true drift|	+0.60	0.97
	Deep event KL	+0.56	0.95
	Ordinary-history KL	+0.52	0.93
Within the training length	Shallow drift	+0.17	0.55
	Test CE (event type)	−0.07	0.51
	Test waiting-time NLL	+0.07	0.59
	Test joint NLL	+0.00	0.58

Figure 2: **On the 36 evaluation models, the sign of DPD detects 11 of the 12 severe failures with one false positive; statistics scored within the training length separate them near chance.** (a) DPD against crossings out of 256 rollouts for all 46 models (open markers: the development models); dashed: DPD = 0; dotted: 128 crossings. (b) Spearman correlation with the crossing count and AUROC for severe failure on the 36 evaluation models. Bank statistics use positions $t \geq 400$ with $q > 0$; ordinary-history statistics use positions 400–799 of unconditioned oracle trajectories fed continuously; higher values are read as higher risk.

+0.82 and separates their severe failures at AUROC 0.99 (Figure 2). Its sign gives a screen with no fitted parameter: positive DPD means the model predicts growth where the true queue is pulled toward empty, and zero drift is the boundary above which busy periods tend to grow (§3). The sign flags 11 of the 12 severe failures with 1 false positive, model 53 (§4.3); Appendix A.3 gives the pooled companion over all 46.

Baselines. Would other statistics computed on long reference histories identify the same failures? Figure 2b compares DPD with statistics from the same teacher-forced pass (the unsigned error |DPD − true drift|, the event-type KL divergence from the oracle, and the cross-entropy of the bank's own next events), with the same readings on ordinary long histories, and with losses scored within the training length. Within the training length, drift, test CE, waiting-time NLL and their joint NLL separate the severe failures near chance (AUROC 0.51 to 0.59). Every statistic read at deep positions of the bank detects them well above chance. DPD's AUROC exceeds deep KL's by 0.04, a difference not resolved at 36 models (95% paired-bootstrap interval 0.00 to 0.11 over 10,000 class-stratified resamples of models; Appendix A.3). Deep KL also requires the analytic transition law, whereas DPD uses only model predictions on simulator-generated reference histories. On ordinary long histories fed continuously (Appendix A.3), CE, KL and drift also detect the severe failures (AUROC 0.93 to 0.95) but rank them less well (Spearman at most +0.73), and the drift's sign flags all 12 with 9 false positives. Restricted to positions below 400, the drift statistic separates the failures near chance (AUROC 0.50 and 0.58), and computed on 32 of the 1,024 histories it keeps a median AUROC of 0.99 (Appendix Table 6). Scoring one model on the full bank took 10 seconds on one GeForce RTX 2080 Ti, after 38 seconds to build the shared bank once (Appendix A.3).

4.3 Corrective supervision

Placement of supervision. Crossings begin at deep positions (§4.1), where the models' predictions diverge (§4.2). We therefore ask whether exact supervision at deep positions reduces crossings more than the same supervision placed within the training length. Six severe failures with positive DPD provide the starting checkpoints (seeds 7, 8, 1, 48, 49 and 51). Each is fine-tuned under three variants that share its initial weights, 3,120 optimization updates and one sequence of ordinary training minibatches; what varies is where extra supervision is placed. *Deep supervision* adds oracle targets at deep positions in queue states 1–19. The decisive control, *shallow* supervision, adds the same targets at shallow positions

Model	Original	Deep	Shallow	Continued
Severe failures, all three variants, three fine-tuning seeds				
Seed 7	255	1 (3,2)	209 (214,226)	254 (244,256)
Seed 8	245	28 (27,18)	256 (153,201)	256 (217,234)
Seed 1	240	1 (1,0)	192 (184,157)	178 (195,206)
Seed 48	189	3 (3,1)	82 (79,72)	154 (122,137)
Seed 49	178	4 (10,8)	41 (125,52)	117 (88,117)
Seed 51	235	1 (1,1)	169 (133,146)	137 (134,171)
Additional applications: severe failures, deep supervision only, one fine-tuning seed				
Seed 105	248	14	—	—
Seed 108	252	5	—	—
Seed 109	232	0	—	—
False positive: positive DPD, not a severe failure, three fine-tuning seeds				
Seed 53	1	12 (7,10)	1 (1,2)	1 (1,0)

Table 2: **On the six severe failures run under all three variants, supervision beyond the training length reduces crossings more than either control at every fine-tuning seed; it also reduces crossings in the three further severe failures to which it was applied alone.** Crossings of $q > 50$ out of 256 anchored rollouts of 40,000 events; the oracle expects 2.86. Within each row, variants share the starting checkpoint, 3,120 updates, the ordinary training minibatches and one rollout sampling seed. Each row gives the first fine-tuning seed, with the two further seeds in parentheses.

in the same states, reweighted to the deep variant's queue-state distribution, so the two differ in where along the history the targets fall. *Continued training* applies only the ordinary loss and so controls for extra optimization. The supervised variants add, with equal weight to the ordinary loss, cross-entropy and KL losses against the oracle's event-type and waiting-time laws on a separate bank of 2,048 oracle busy periods (Appendix A.4), at three fine-tuning seeds.

Deep supervision has fewer crossings than shallow supervision and than continued training in all 18 model-by-seed cells, and at least 37 crossings separate it from the nearer control in every cell (Table 2). Across these cells it leaves 0–28 crossings per cell, against 41–256 for shallow supervision and 88–256 for continued training. As additional applications without controls, deep supervision alone brought the three positive-DPD models among seeds 100–109, all severe failures, from 232–252 crossings to 0–14. Supervising every queue state instead of states 1–19 gave 1, 26 and 1 crossings on seeds 7, 8 and 1, against 1, 27 and 1 (one fine-tuning seed, one machine; Appendix Table 7), so the restriction showed no additional benefit. Likelihood fine-tuning on ordinary long trajectories with matched target counts left 5, 71 and 7 crossings on seeds 7, 8 and 1, against 1, 27 and 1 for deep supervision (one machine, one fine-tuning seed; Table 9): generic long-sequence training matched deep supervision on seed 7 and trailed it on the others; it beat both shallow supervision and continued training on all 3. Over all 9 severe failures and 21 runs, deep supervision reduced 178–255 crossings to at most 28.

Fidelity beyond the crossing count. Over the 18 cells, deep supervision leaves the total variation distance (TV) of time-weighted occupancy from the oracle at 0.005–0.055 (the six originals 0.020–0.41) and gives the smallest sojourn-time error of the three variants in 17 (Appendix A.4).

The one false positive. Model 53 was flagged (DPD +0.160) although it crosses on 1 rollout, and deep supervision raised its crossings to 7–12 across the three fine-tuning seeds (Table 2). An average drift over deep positions can miss how the predicted drift varies across queue states, and corrective training can worsen a model whose rollouts are already accurate (Appendix A.5).

Choice of targets. A practical simulator supplies draws from the law. On seeds 7, 8 and 1 at three fine-tuning seeds, we replaced the exact targets of deep supervision by one

transition drawn afresh from the law at each supervised position and presentation, whose negative log-likelihood equals the exact loss in expectation up to a constant. Sampled targets had fewer crossings than shallow supervision and continued training in all 9 cells (83, against 1,778 and 1,972 of 2,304 rollouts). Summed over fine-tuning seeds, they had more crossings than exact targets on seed 7 (30 against 7) and fewer on seed 8 (49 against 75), so the equal totals (83) are offsets that cancel (Appendix A.4).

The histories' own recorded next events are the obvious data-only targets, and they are biased because the repair bank's busy periods were selected on their future survival: at its supervised positions departures occur with frequency .492 against the oracle's .5556. Histories selected only on their past would give unbiased labels. Over the same 9 cells, own-label targets left 199–256 crossings per cell, at least 199 more than exact targets, while their test CE differed from the exact-target run of the same cell by at most 1.2×10^{-4} nats, less than the spread of test CE across the 46 models (1.3×10^{-4}).

5 Conclusion

We investigated whether long-horizon failures in learned event simulators can be diagnosed and corrected using predictions on shared reference histories. Short-history test loss concealed substantial differences in rollout behavior, whereas predicted drift beyond the training length identified most severe failures (11 of 12 among models it was not developed on). Supervision at those positions reduced crossings more than matched shallow supervision or continued training, including when conditional targets were supplied by simulator draws. These results support evaluating learned simulators on histories that probe deployment conditions and using the resulting discrepancies to guide training.

The study is limited to anchored M/M/1 queues and the tested transformer recipes. Against likelihood fine-tuning on ordinary long trajectories, the repair reduced crossings more on some models and not on others (three models, one fine-tuning seed). DPD's averaging can obscure state-dependent behavior, and corrective supervision does not improve every model. Extending the approach to queueing networks, developing state-dependent diagnostics, and preserving shallow-context accuracy during repair are natural next steps.

Reproducibility statement

We provide basic reproduction code for the core experiments. The manuscript and supplement describe the model architecture, training settings, reference-history construction, diagnostic, corrective supervision and rollout evaluation.

AI usage disclosure

In this work, we used generative AI tools to polish and improve the clarity of the writing (e.g., refining phrasing and grammar in drafted text), to assist with retrieval and discovery of related work (e.g., surfacing candidate papers during the literature search), and to help prepare the source code package for release (e.g., organizing and documenting code prior to submission). We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

References

Kushal Arora, Layla El Asri, Hareesh Bahuleyan, and Jackie Chi Kit Cheung. Why exposure bias matters: An imitation learning perspective of error accumulation in language generation. In *Findings of the Association for Computational Linguistics: ACL 2022*, pp. 700–710, 2022. doi: 10.18653/v1/2022.findings-acl.58. URL `https://aclanthology.org/2022.findings-acl.58/`. arXiv:2204.01171.

Opher Baron, Dmitry Krass, Arik Senderovich, and Eliran Sherzer. Supervised ML for solving the GI/GI/1 queue. *INFORMS Journal on Computing*, 36(3):766–786, 2024. doi: 10.1287/ijoc.2022.0263. URL https://doi.org/10.1287/ijoc.2022.0263.

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled sampling for sequence prediction with recurrent neural networks. In *Advances in Neural Information Processing Systems*, volume 28, 2015. URL https://arxiv.org/abs/1506.03099.

Tanguy Bosser and Souhaib Ben Taieb. On the predictive accuracy of neural temporal point process models for continuous-time event data. *Transactions on Machine Learning Research*, 2023. URL https://openreview.net/forum?id=3OSISBQPrM. arXiv:2306.17066.

Johannes Brandstetter, Daniel Worrall, and Max Welling. Message passing neural PDE solvers. In *International Conference on Learning Representations*, 2022. URL https://arxiv.org/abs/2202.03376.

Shouyuan Chen, Sherman Wong, Liangjian Chen, and Yuandong Tian. Extending context window of large language models via positional interpolation. *arXiv preprint arXiv:2306.15595*, 2023. URL https://arxiv.org/abs/2306.15595.

Alexander D'Amour, Katherine Heller, Dan Moldovan, Ben Adlam, Babak Alipanahi, Alex Beutel, Christina Chen, Jonathan Deaton, Jacob Eisenstein, Matthew D. Hoffman, Farhad Hormozdiari, Neil Houlsby, Shaobo Hou, Ghassen Jerfel, Alan Karthikesalingam, Mario Lucic, Yian Ma, Cory McLean, Diana Mincu, Akinori Mitani, Andrea Montanari, Zachary Nado, Vivek Natarajan, Christopher Nielson, Thomas F. Osborne, Rajiv Raman, Kim Ramasamy, Rory Sayres, Jessica Schrouff, Martin Seneviratne, Shannon Sequeira, Harini Suresh, Victor Veitch, Max Vladymyrov, Xuezhi Wang, Kellie Webster, Steve Yadlowsky, Taedong Yun, Xiaohua Zhai, and D. Sculley. Underspecification presents challenges for credibility in modern machine learning. *Journal of Machine Learning Research*, 23(226):1–61, 2022. URL https://jmlr.org/papers/v23/20-1335.html.

Tianxing He, Jingzhao Zhang, Zhiming Zhou, and James Glass. Exposure bias versus self-recovery: Are distortions really incremental for autoregressive text generation? In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 5087–5102, 2021. doi: 10.18653/v1/2021.emnlp-main.415. URL https://aclanthology.org/2021.emnlp-main.415/. arXiv:1905.10617.

Ivan Karpukhin and Andrey Savchenko. Detecting the future: All-at-once event sequence forecasting with horizon matching. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(27):22536–22544, 2026. doi: 10.1609/aaai.v40i27.39413. URL https://ojs.aaai.org/index.php/AAAI/article/view/39413.

Ivan Karpukhin, Foma Shipilov, and Andrey Savchenko. HoTPP benchmark: Are we good at the long horizon events forecasting? *Neurocomputing*, 672:132771, 2026. ISSN 0925-2312. doi: 10.1016/j.neucom.2026.132771. URL https://www.sciencedirect.com/science/article/pii/S0925231226001682. arXiv:2406.14341.

Seyed Mehran Kazemi, Rishab Goel, Sepehr Eghbali, Janahan Ramanan, Jaspreet Sahota, Sanjay Thakur, Stella Wu, Cathal Smyth, Pascal Poupart, and Marcus Brubaker. Time2Vec: Learning a vector representation of time. *arXiv preprint arXiv:1907.05321*, 2019. URL https://arxiv.org/abs/1907.05321.

Amirhossein Kazemnejad, Inkit Padhi, Karthikeyan Natesan Ramamurthy, Payel Das, and Siva Reddy. The impact of positional encoding on length generalization in transformers. In *Advances in Neural Information Processing Systems*, volume 36, pp. 24892–24928, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/4e85362c02172c0c6567ce593122d31c-Abstract-Conference.html.

Remi Lam, Alvaro Sanchez-Gonzalez, Matthew Willson, Peter Wirnsberger, Meire Fortunato, Ferran Alet, Suman Ravuri, Timo Ewalds, Zach Eaton-Rosen, Weihua Hu, Alexander Merose, Stephan Hoyer, George Holland, Oriol Vinyals, Jacklynn Stott, Alexander Pritzel, Shakir Mohamed, and Peter Battaglia. Learning skillful medium-range global

weather forecasting. *Science*, 382(6677):1416–1421, 2023. doi: 10.1126/science.adi2336. URL https://doi.org/10.1126/science.adi2336.

Alex M. Lamb, Anirudh Goyal, Ying Zhang, Saizheng Zhang, Aaron C. Courville, and Yoshua Bengio. Professor forcing: A new algorithm for training recurrent networks. In *Advances in Neural Information Processing Systems*, volume 29, 2016. URL https://papers.neurips.cc/paper/2016/hash/16026d60ff9b54410b3435b403afd226-Abstract.html.

Phillip Lippe, Bastiaan S. Veeling, Paris Perdikaris, Richard E. Turner, and Johannes Brandstetter. PDE-Refiner: Achieving accurate long rollouts with neural PDE solvers. In *Advances in Neural Information Processing Systems*, volume 36, pp. 67398–67433, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/d529b943af3dba734f8a7d49efcb6d09-Abstract-Conference.html.

David Lüdke, Marin Biloš, Oleksandr Shchur, Marten Lienen, and Stephan Günnemann. Add and thin: Diffusion for temporal point processes. In *Advances in Neural Information Processing Systems*, volume 36, pp. 56784–56801, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/b1d9c7e7bd265d81aae8d74a7a6bd7f1-Abstract.html.

Hongyuan Mei and Jason Eisner. The neural Hawkes process: A neurally self-modulating multivariate point process. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL https://arxiv.org/abs/1612.09328.

Daksh Mittal, Shunri Zheng, Jing Dong, and Hongseok Namkoong. Data-driven stochastic modeling using autoregressive sequence models: Translating event tables to queueing dynamics. *arXiv preprint arXiv:2509.05839*, 2025. URL https://arxiv.org/abs/2509.05839.

Bowen Peng, Jeffrey Quesnelle, Honglu Fan, and Enrico Shippole. YaRN: Efficient context window extension of large language models. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=wHBfxhZu1u.

Ofir Press, Noah A. Smith, and Mike Lewis. Train short, test long: Attention with linear biases enables input length extrapolation. In *International Conference on Learning Representations*, 2022. URL https://arxiv.org/abs/2108.12409. arXiv:2108.12409.

Stéphane Ross and Drew Bagnell. Efficient reductions for imitation learning. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pp. 661–668, 2010. URL https://proceedings.mlr.press/v9/ross10a.html.

Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pp. 627–635, 2011. URL https://proceedings.mlr.press/v15/ross11a.html.

Anian Ruoss, Grégoire Delétang, Tim Genewein, Jordi Grau-Moya, Róbert Csordás, Mehdi Bennani, Shane Legg, and Joel Veness. Randomized positional encodings boost length generalization of transformers. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 1889–1903, 2023. doi: 10.18653/v1/2023.acl-short.161. URL https://aclanthology.org/2023.acl-short.161/.

Alvaro Sanchez-Gonzalez, Jonathan Godwin, Tobias Pfaff, Rex Ying, Jure Leskovec, and Peter Battaglia. Learning to simulate complex physics with graph networks. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 8459–8468, 2020. URL https://proceedings.mlr.press/v119/sanchez-gonzalez20a.html.

Eliran Sherzer, Opher Baron, Dmitry Krass, and Yehezkel Resheff. Approximating G(t)/GI/1 queues with deep learning. *European Journal of Operational Research*, 322 (3):889–907, 2025. doi: 10.1016/j.ejor.2024.12.030. URL https://doi.org/10.1016/j.ejor.2024.12.030.

Yufeng Zheng, Zeyu Zheng, and Tingyu Zhu. A doubly stochastic simulator with applications in arrivals modeling and simulation. *Operations Research*, 73(2):910–926, 2025. doi: 10.1287/opre.2021.0597. URL https://doi.org/10.1287/opre.2021.0597.

Simiao Zuo, Haoming Jiang, Zichong Li, Tuo Zhao, and Hongyuan Zha. Transformer Hawkes process. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 11692–11702, 2020. URL https://proceedings.mlr.press/v119/zuo20a.html.

A Supplementary material

This supplement supports the three main claims. Appendix A.1 gives the experimental setup and implementation, Appendix A.2 the localization of failures and the positional controls (§4.1), Appendix A.3 the evaluation of the diagnostic (§4.2), and Appendix A.4 the corrective supervision (§4.3); Appendix A.5 examines the two misclassified evaluation models.

A.1 Experimental setup and implementation

Architecture and training. A sequence starts with a token for the empty initial state, followed by alternating waiting-time and event-type tokens. Time2Vec maps each waiting time T to one linear feature $w_0T + b_0$ and 63 features $\sin(w_jT + b_j)$ with learned w_j, b_j; event types have learned embeddings. Two sinusoidal encodings are added to every token, one of the event index, which identifies the event a token belongs to, and one of period two, which marks waiting-time versus event-type tokens. Both are computed in fp32 from integer indices, with no maximum length. All transformers use Adam with a peak learning rate of 5×10^{-4}, a 30-epoch warm-up and cosine decay, batch size 32 and 300 epochs, under fp16 autocast with a gradient scaler, which skipped 31–38 optimizer steps per model over the 43 models with logged counts. Seed k fixes both the initialization and the training set. The 46 models of the main recipe use seeds 0–9, 20–35, 45–54 and 100–109. On the shared test set, their waiting-time negative log-likelihood lies within 8.5×10^{-5} nats of the oracle's 0.50352.

Reference banks. The diagnostic bank holds 1,024 oracle busy periods, each selected to last at least 800 events and truncated at that length. Every model receives the same event and time tokens and the same mask of deep, non-empty positions, so differences in DPD come from the models alone. The repair bank holds 2,048 oracle busy periods of 800 events, drawn independently of the diagnostic bank.

Rollout protocol. Rollouts are anchored (§3.1). The default cap of 4,096 events without a return to empty changes the oracle's crossing probability by less than 6×10^{-6} relative. Over the rollouts of the 46 models, invalid departures (a departure sampled at an empty queue) make up at most 1×10^{-5} of generated events, and the models place 5×10^{-6} to 1×10^{-4} of their probability on a departure at empty states.

Precision. Training sees only positions below 400, so fp16 arithmetic never reaches a deep position during training. Every evaluation (test CE, DPD and the other bank readings, and the rollouts, whose incremental decoder keeps an fp32 cache) runs in fp32. Retraining seeds 0 and 7 entirely in fp32 preserves their outcomes: seed 0 keeps a negative DPD (-0.105) and crosses on 33 of 256 rollouts, and seed 7 keeps a positive DPD ($+0.314$) and crosses on 256.

Shared random numbers. Every model's event-type CE uses one shared test set of 1,000 oracle trajectories, and every DPD reading one reference bank. Rollouts under the

same protocol share a sampling seed, and relative crossing rates use one 10^5-trajectory oracle reference.

Numerical checks. The cached incremental decoder used for rollouts reproduces the full model's predictions, including beyond the training length and at the beginning-of-sequence inputs introduced when a context restarts: on 8 rollouts of 1,000 events and 3,621 anchored segments with 3,605 natural and 154 forced resets, the largest relative discrepancy in predicted rates is 1×10^{-5} and 3×10^{-7}. An independent reconstruction of queue lengths from event tokens agrees with the rollout pipeline on all 203 of 203 stored rollouts.

A.2 Failure localization and positional controls

First-crossing position. For every crossing rollout, we record the position within the current context at which the queue first exceeds 50 (§4.1; 45 of the 46 models cross at least once). For the oracle, the closed-form first-passage law of the busy period gives a median crossing position of 348 and places 63% of crossings within the training length.

Resetting at the training length. We hold four checkpoints fixed and change only the reset rule: queue and context restart after 400 events, so every scored position lies in 0–399. The reset is applied before the queue is recorded on the last event of a capped context, the oracle calculation follows the same convention, and the capped rollouts use their own sampling seed. Table 3 gives the counts. Because the reset also changes the true process, the comparison is with the oracle's expected count under the identical reset.

Model	Default cap (4,096)	Cap at training length (400)
Seed 7	255	3
Seed 1	240	2
Seed 3	215	2
Seed 0	27	2
Oracle, expected	2.86	1.8

Table 3: **Resetting at the training length brings all four models to the oracle's scale.** Crossings of $q > 50$ out of 256 rollouts of 40,000 events with contexts capped at 4,096 or 400 events, and the oracle's expected count under each cap. Weights are fixed; the two protocols use separate sampling seeds.

Positional encoding and training length. The alternatives to the main recipe's sinusoidal absolute encoding are attention with linear biases (ALiBi), no positional encoding (NoPE), and the sinusoidal encoding trained with mixed positions, ten models each at training length 400. The mixed-position recipe uses consecutive indices for half the training sequences and spread-out indices for the rest, so indices beyond 400 occur in training and the consecutive evaluation pattern is part of it (event KL on ordinary 400-event histories at most 2.8×10^{-5} nats). A fourth recipe trains the main recipe on 800-event sequences, eight models, with half as many sequences and half the batch so that each update sees as many events. Everything else follows the main recipe, and every model is rolled out with the same sampling seed. Table 4 gives every count. Every DPD reading of the three alternatives is negative. For the 800-event models the bank's positions 400–799 lie inside the training length, so their DPD does not probe positions beyond it.

A.3 Diagnostic evaluation

All models. Table 5 lists every reading and outcome. The diagnostic's reference bank and zero threshold were chosen on seeds 0–9; the other 36 models were trained afterwards with the same recipe and form the evaluation set of §4.2. Pooled over all 46 as a companion, DPD ranks crossing counts at Spearman +0.84 with AUROC 0.98, and its sign flags 17 of 18 severe failures with 1 false positive. Seeds 100–109 were trained and rolled out on a second machine with the same code and sampling seed; their bank readings and test CE were computed on the shared bank and test set.

Recipe	Models	Severe	Crossings, sorted
Sinusoidal, $L = 400$	46	18	0, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 3, 4, 5, 5, 7, 10, 11, 16, 19, 21, 27, 37, 40, 41, 71, 82, 140, 160, 178, 189, 194, 210, 211, 215, 232, 235, 239, 240, 243, 243, 245, 248, 252, 255
ALiBi	10	0	1, 1, 1, 4, 6, 22, 23, 24, 30, 31
NoPE	10	0	1, 2, 2, 5, 6, 17, 19, 20, 49, 98
Mixed positions	10	0	1, 4, 6, 6, 8, 9, 15, 17, 69, 87
Sinusoidal, $L = 800$	8	3	0, 0, 1, 19, 113, 210, 250, 251

Table 4: **Positional encoding and training length (Appendix A.2).** Crossings of $q > 50$ in 256 anchored rollouts of 40,000 events, all at load 0.8 with one rollout sampling seed; severe means at least 128. Every recipe is the main one except for the named change; the $L = 800$ recipe also halves the number of training sequences and the batch. The oracle expects 2.86.

Seed	ΔCE	DPD	KL	A^*	Cross.	Seed	ΔCE	DPD	KL	A^*	Cross.
0	6.7	−0.108	0.60	9.9	27	33	8.4	−0.050	2.60	2.5	7
1	8.2	+0.029	273.08	200.2	240	34	11.8	−0.113	8.26	0.7	2
2	5.4	−0.103	3.17	0.7	2	35	2.1	+0.428	1573.64	262.2	243
3	7.8	+0.168	53.75	151.0	215	45	6.0	−0.119	4.51	3.5	10
4	10.2	−0.202	6.56	0.0	0	46	5.9	−0.002	18.30	15.1	40
5	14.5	−0.085	0.56	15.5	41	47	6.4	−0.126	0.78	1.4	4
6	2.8	+0.094	24.04	120.8	194	48	11.3	+0.193	780.82	116.9	189
7	10.9	+0.442	1949.14	390.2	255	49	7.4	+0.396	1930.33	105.8	178
8	5.5	+0.019	442.74	229.9	245	50	3.1	−0.096	0.36	0.7	2
9	7.5	+0.002	7.99	87.3	160	51	15.3	+0.304	1112.25	256.5	235
20	7.4	−0.127	9.93	3.9	11	52	9.5	−0.112	0.08	0.3	1
21	9.6	−0.037	3.38	28.9	71	53	10.4	+0.160	1007.81	0.3	1
22	13.5	−0.127	21.64	0.3	1	54	7.2	−0.109	2.51	0.7	2
23	6.0	+0.026	377.41	70.5	140	100	8.7	−0.113	12.29	5.7	16
24	6.8	−0.184	4.67	0.3	1	101	9.5	−0.102	3.11	34.4	82
25	6.2	−0.047	153.92	6.9	19	102	8.0	−0.073	1.23	1.8	5
26	2.3	−0.102	42.33	1.0	3	103	7.8	−0.093	2.30	7.6	21
27	12.6	+0.402	1534.20	237.3	243	104	10.8	−0.033	6.50	137.2	210
28	5.7	+0.306	1140.39	149.2	211	105	6.3	+0.147	414.91	346.3	248
29	8.2	+0.497	2072.09	211.2	239	106	13.9	−0.260	13.35	0.3	1
30	6.5	−0.221	8.92	0.3	1	107	9.7	−0.248	13.00	0.3	1
31	7.1	−0.064	1.68	1.8	5	108	9.0	+0.308	1023.34	365.0	252
32	7.8	−0.084	1.28	13.9	37	109	7.1	+0.174	549.77	196.0	232

Table 5: **All** 46 **models.** ΔCE: event-type test CE minus the oracle's on the shared test set, in units of 10^{-5} nats. DPD: deep-prefix drift on the shared reference bank (true drift −0.111). KL: mean deep event-type KL divergence from the oracle, in units of 10^{-3} nats. A^*: relative crossing rate. Cross.: crossings of $q > 50$ in 256 anchored rollouts of 40,000 events (oracle: 2.86 expected); severe failure is at least 128.

The relative crossing rate. With $F_\phi(h)$ and $F_{\text{oracle}}(h)$ the fractions of the model's and the oracle's rollouts that have crossed by event h,

$$A^* = \frac{-\log[1 - F_\phi(h^*)]/h^*}{-\log[1 - F_{\text{oracle}}(h_O)]/h_O}, \qquad h_O = 40{,}000,$$

the model's average first-passage hazard up to its horizon h^* divided by the oracle's over the full horizon. The model horizon is the first saved horizon at which at least half of the rollouts have crossed, or the full horizon if fewer do, so for an early crosser A^* compares hazards over different windows. The crossing count at the common horizon avoids this mismatch and ranks the models almost identically (Figure 2b).

Baselines. All baselines share DPD's bank, positions and teacher-forced pass: the unsigned error |DPD − true drift|; the mean event-type KL divergence from the oracle, $\mathrm{KL}(p \,\|\, \hat{p})$; the same with the waiting-time KL added; and the cross-entropy of the bank's

own next events. Shallow drift is DPD's statistic at $t < 400$ of the same histories. Test CE, waiting-time NLL and their sum, the joint NLL of the model's factorization, are per-event means on the shared 400-event test set.

Ordinary long histories. A practical alternative to the bank evaluates each model on ordinary long oracle trajectories: 1,024 trajectories of 800 events from an empty queue, not selected for survival, fed continuously across returns to empty. This continuous-history evaluation differs from the anchored rollouts, which clear the context at every return, so a trajectory can contain several busy periods. At positions 400–799 we read the recorded-event CE and the event-type KL to the oracle over all 409,600 positions (the oracle law at empty states included), and the mean predicted increment over the 368,240 non-empty positions. Restricted to the non-empty positions, CE and KL give AUROC 0.93 and 0.93 and Spearman +0.57 and +0.57 on the 36 evaluation models, against 0.93 and 0.93 (+0.53 and +0.52) over all positions. KL counts every per-position error: some models have large per-position errors whose mean drift stays below zero (for example seeds 22, 25 and 26), errors that partly cancel in DPD's mean while their queues keep returning to empty. The paired bootstrap interval for the difference in AUROC between DPD and deep KL resamples models within each outcome class and is conditional on the fixed bank and the recorded rollout outcomes; it does not include uncertainty from bank construction or finite rollouts. Unsigned statistics score an error toward departures, which pulls the queue back toward empty, the same as an error toward arrivals, which pushes it out. The diagnostic bank is selected on its histories' future survival, so its deep continuations are arrival-heavy (departure frequency .512 over the diagnostic bank's deep non-empty positions, against the oracle's .5556), and a loss scored against them is biased toward predicted growth.

Sensitivity analyses. Table 6 gives the depth-window and bank-size sensitivity of DPD on the 36 evaluation models.

Variant	Scored events	AUROC	Spearman	Flagged / false pos.
Depth window on the full bank				
Positions 0–199	—	0.50	+0.07	0 / 0
Positions 200–399	—	0.58	+0.23	0 / 0
Positions 400–599	—	0.86	+0.64	4 / 1
Positions 600–799	—	0.98	+0.79	11 / 3
Positions 400–799 (DPD)	—	0.99	+0.82	11 / 1
Bank size, positions 400–799; median (5th–95th percentile) over 500 subsets				
32 histories	25,600	0.99 (0.98–0.99)	—	11 / 1
64 histories	51,200	0.99 (0.98–0.99)	—	11 / 1
128 histories	102,400	0.99 (0.98–0.99)	—	11 / 1
256 histories	204,800	0.99 (0.99–0.99)	—	11 / 1
512 histories	409,600	0.99 (0.99–0.99)	—	11 / 1
1,024 histories (DPD)	819,200	0.99	—	11 / 1

Table 6: **Depth-window and bank-size sensitivity of DPD on the 36 evaluation models.** Same bank, live-position mask, outcomes and zero threshold; a window's statistic weights its positions by their live counts. Bank subsets are whole histories drawn without replacement, shared across models. Fixed-bank sensitivity analyses; the DPD definition is unchanged.

Cost of DPD. The timing of §4.2 was measured for one model (seed 20) on one GeForce RTX 2080 Ti with nothing else running. DPD's teacher-forced fp32 pass over the 1,024 histories of the bank, at batch 16, took 10 seconds (median of 3 runs). Building the bank took 38 seconds; it is built once and shared by every model. Teacher forcing scores all positions of a history in one forward pass.

State composition. Long busy periods reach larger queue lengths than short ones, so deep and shallow positions differ in queue-state composition. Reweighting the deep positions to the shallow state mixture within five queue-length strata gives Spearman +0.85 with A^*

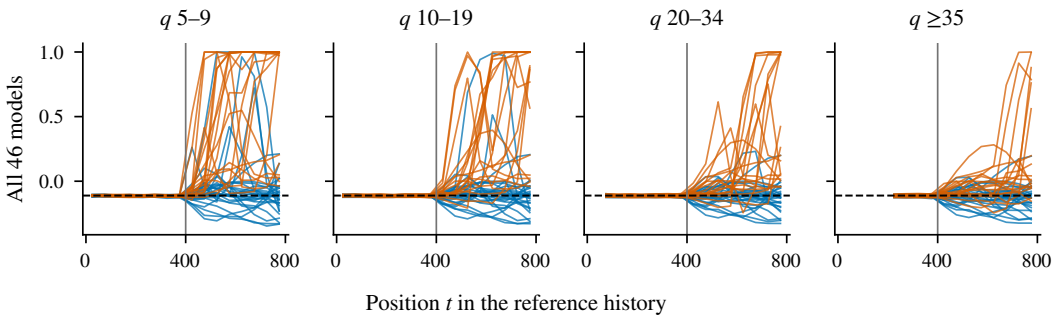


Figure 3: **Within each queue-length stratum shown, predictions stay on the true drift up to the training length and depart beyond it.** Mean predicted queue increment against position, in 50-event bins with at least 2,000 positions, on the shared 1,024-history bank, one line per model (all 46). Vermilion: severe failure (at least 128 of 256 rollouts crossing); blue: not severe. Dashed: the true drift; vertical line: the training length.

across the 46 models, against +0.85 unweighted. The check matches the state marginal over five strata and leaves other differences between shallow and deep histories in place.

State-conditioned profiles. The queue-state mix changes with position, because the bank keeps only busy periods that survive to 800 events. Figure 3 therefore plots the mean predicted increment against position within each queue-length stratum, in 50-event bins, drawing a bin only when it holds at least 2,000 positions. Stratum q 1–4 is omitted: beyond position 100 none of its bins reaches that support. Within every stratum shown, predictions stay on the true drift up to the training length and depart beyond it, often non-monotonically in position. Because every history starts at an empty queue, position index and context length coincide on the bank.

A.4 Corrective supervision

Training objective. Every checkpoint is fine-tuned with Adam at learning rate 5×10^{-5}, weight decay 10^{-5} and gradient clipping 1, using automatic mixed precision, for 20 epochs of 156 updates (3,120 in total). Ordinary minibatches contain fresh 400-event trajectories. Write $\mathcal{L}_{\text{ordinary}}$ for the original joint negative log-likelihood loss and $\mathcal{L}_{\text{oracle}}$ for the additional supervision loss. Let p_E, p_T be the oracle event-type and waiting-time laws at a reference state, and $\widehat{P}_{\phi,E}, \widehat{P}_{\phi,T}$ the corresponding model laws. The supervised variants minimize

$$\mathcal{L} = \tfrac{1}{2}\mathcal{L}_{\text{ordinary}} + \tfrac{1}{2}\mathcal{L}_{\text{oracle}},$$
$$\mathcal{L}_{\text{oracle}} = \text{mean}_{\text{selected positions}} \left[\text{CE}(p_E, \widehat{P}_{\phi,E}) + \text{KL}(p_T \,\|\, \widehat{P}_{\phi,T}) \right],$$

with positions selected on the repair bank.

Matched controls and fine-tuning seeds. Deep supervision selects positions $400 \le t < 800$ in queue states 1–19. Shallow supervision selects positions $t < 400$ in the same states, reweighted to the deep variant's queue-state marginal. Continued training uses only $\mathcal{L}_{\text{ordinary}}$. The three variants of a checkpoint share its initial weights, the ordinary minibatches and the schedule. The two further fine-tuning seeds change the order of ordinary and oracle minibatches and keep the training sequences, schedule and rollout sampling seed, so they measure optimization variation on a fixed checkpoint. All variants compared within a checkpoint and fine-tuning seed ran on one machine and share one rollout sampling seed. Deep supervision was applied to every model with positive DPD among seeds 45–54 and 100–109.

Target construction. The target comparison fine-tunes seeds 7, 8 and 1 at fine-tuning seeds 0–2 with deep supervision, changing only the targets, and reruns exact targets, shallow

Checkpoint	Original	Deep, states 1–19	Deep, all states
Seed 7	255	1	1
Seed 8	245	27	26
Seed 1	240	1	1

Table 7: **Restricting deep supervision to queue states 1–19 showed no additional benefit over supervising all states.** Crossings of $q > 50$ out of 256 rollouts at one fine-tuning seed. Both variants supervise positions 400–799 with the same number of targets per batch and ran on the same machine with one rollout sampling seed.

supervision and continued training beside them on the same machine. Sampled targets replace (p_E, p_T) at each supervised position by one transition drawn afresh from the oracle law at every presentation, $T' \sim \mathrm{Exp}(\lambda+\nu)$ and a departure with probability $\nu/(\lambda+\nu)$, and train on its negative log-likelihood. Own-label targets train on the selected busy periods' recorded next events, which are fixed across presentations and inherit the arrival-heavy bias of the survival-selected rows. Table 8 gives every count. On seed 8, own labels raised the mean absolute error of the predicted drift at positions below 400 by 0.041–0.049 across the three fine-tuning seeds, against at most 0.016 for sampled or exact targets in any cell.

Seed	FT	Sampled	Exact	Own labels	Shallow	Continued	Sampled−exact
7	0	12	1	229	210	241	+11
7	1	6	2	224	219	238	+4
7	2	12	4	235	223	256	+8
8	0	16	27	256	233	256	−11
8	1	24	26	256	169	203	−2
8	2	9	22	256	183	184	−13
1	0	0	1	239	189	181	−1
1	1	4	0	242	194	215	+4
1	2	0	0	199	158	198	0

Table 8: **Simulator-sampled and own-label targets (Appendix A.4).** Crossings of $q > 50$ in 256 rollouts for five fine-tuning variants of seeds 7, 8 and 1, all run on one machine and gated on the rollout stream of Table 2. FT is the fine-tuning seed. Sampled: one oracle transition drawn per supervised position and presentation; exact: deep supervision with the oracle laws; own labels: the selected busy periods' recorded next events. The oracle expects 2.86 crossings.

Checkpoint	Ordinary long	Deep (exact)	Shallow	Continued	Own labels
Seed 7	5	1	210	241	229
Seed 8	71	27	233	256	256
Seed 1	7	1	189	181	239

Table 9: **Ordinary long-sequence fine-tuning against the matched variants, fine-tuning seed 0, one machine.** Crossings of $q > 50$ out of 256 rollouts on one rollout sampling seed. Ordinary long: joint likelihood of the recorded next events on unconditioned 800-event oracle trajectories fed continuously, with as many targets per update as deep supervision (50% of them beyond position 400); the other variants as in Table 8. Three checkpoints at one fine-tuning seed are an initial comparison.

Distributional fidelity. Crossings record rare excursions, so we also compare the queue's distribution with the oracle's: occupancy TV (§4.3); the event-averaged queue length; and sojourn error, the Wasserstein-1 distance W_1 between the model's and the oracle's distributions of customer sojourn time (arrival to departure). All are computed on the same rollouts as the crossing counts. Over the 18 cells with both controls, occupancy TV is 0.011–0.476 after shallow supervision and 0.020–0.567 after continued training, and sojourn W_1 is 0.35–227.74 and 0.46–249.54 (deep supervision in §4.3). Along the models' own rollouts, the

mean per-event KL divergence between the oracle's and the model's waiting-time laws is at most 0.0002 nats for the deep-supervision runs of seeds 7, 8 and 1, down from at least 0.028 on the original checkpoints. Tables 10 and 11 give the detailed values.

Measure	Original	Nine deep-supervision runs	Oracle / MC floor
Occupancy TV	0.24–0.41	0.009–0.019	0.0017
Mean queue (event-averaged)	18–91	4.2–4.8	4.5
Sojourn W_1	28–159	0.07–0.33	—

Table 10: **On seeds 7, 8 and 1, deep supervision moves occupancy, mean queue length and sojourn times toward the oracle's.** Ranges over the three original checkpoints and over the nine deep-supervision runs (three checkpoints by three fine-tuning seeds), from anchored rollouts of 40,000 events. The occupancy reference is the matched oracle-versus-oracle Monte Carlo floor; the mean-queue reference is the oracle mean.

		Occupancy TV, fine-tuning seed			Sojourn W_1, fine-tuning seed		
Model	Variant	1	2	3	1	2	3
48	Original		0.06076			4.2182	
	Deep	0.00539	0.00601	0.00493	0.0363	0.0396	0.0694
	Shallow	0.01334	0.01113	0.01052	0.9986	0.8051	0.7143
	Continued	0.05223	0.02369	0.04392	2.7163	1.5902	2.1052
49	Original		0.01967			0.8126	
	Deep	0.01128	0.05526	0.03335	0.2102	0.9035	0.5515
	Shallow	0.01535	0.03696	0.01086	0.3544	1.1499	0.4527
	Continued	0.01979	0.05229	0.02072	0.8457	1.4719	0.4559
51	Original		0.06663			4.8364	
	Deep	0.01087	0.02327	0.01857	0.0737	0.3722	0.3128
	Shallow	0.02635	0.02098	0.01807	1.5722	0.9673	1.1959
	Continued	0.02153	0.02596	0.02428	0.9661	1.0125	1.6795
53	Original		0.01295			0.3323	
	Deep	0.05013	0.01925	0.03108	0.8000	0.3061	0.5044
	Shallow	0.01487	0.01349	0.01361	0.3531	0.1802	0.2958
	Continued	0.02602	0.01173	0.01668	0.5470	0.3038	0.2198

Table 11: **Occupancy and sojourn-time fidelity of seeds 48, 49, 51 and 53 after fine-tuning.** Every variant at three fine-tuning seeds; the original checkpoint has none. Distances use the same 256 anchored rollouts of 40,000 events as the crossing counts in Table 2. Seeds 2 and 3 were run on a second machine. Matched oracle-versus-oracle Monte Carlo floors are 0.00173 for occupancy TV and 0.02251 for sojourn Wasserstein-1 distance. Smaller values indicate greater fidelity.

A.5 Two misclassified models

Among the 36 evaluation models, the zero threshold misclassifies two: model 53, flagged although it rarely crosses, and model 104, a severe failure that it misses. We examined both after their outcomes were known; each points to a practical safeguard.

Model 53: an upward drift that turns back before the boundary. DPD (+0.160) reads a real upward drift at short queues: at deep positions, model 53 predicts a mean increment of +0.405 below queue length 20 and −0.170 at 35 and above, a stronger pull toward empty than the true −0.111. Its excursions grow and then turn back: its rollouts exceed 36 on 153 of 256 paths, 40 on 52 and the boundary 50 on 1. The sign of an average over deep positions cannot register where along the queue the drift reverses, so whether such a model counts as a failure depends on the boundary. Deep supervision corrected both parts of the profile, which removed the return, and also lowered the departure probability within the training length from 0.555 to 0.546, a shift of 1.5×10^{-4} nats per position that the likelihood anchor does not resolve. Gambler's ruin on these within-length predictions gives 15.4 expected crossings of 256 after repair against 3.4 before, and the repaired model's crossings begin at a median window depth of 388, inside the training length (original: 753). Across three fine-tuning seeds it crossed 7–12 times, against 1 before.

Model 104: a failure diluted by averaging and beginning beyond the bank. Model 104's DPD is −0.033, just below zero, yet it crosses on 210 of 256 rollouts. Its upward drift is confined to short queues and to part of the deep window: the mean increment is +0.066 at queue lengths 1–4 and +0.002 at 5–9, negative at longer queues, and across positions it rises to +0.031 at 600–649 before falling to −0.041 at 750–799. Averaging over all deep positions and states dilutes this to a small negative value. Its failures also begin later than the bank reaches: the median first-crossing window depth is 1,049 events, against 756 for the other severe failures and the bank's 800. Its deep KL (6.5×10^{-3}) lies between the medians of the other severe (1.11) and non-severe (4.5×10^{-3}) models, and on ordinary long histories fed continuously its drift is positive (+0.126).

Safeguards. A state-resolved sign, which flags a model when any queue stratum drifts upward at deep positions, addresses the miss. On the evaluation models, checked with these outcomes known, it catches model 104 and all 12 severe failures but adds 8 false positives, so it suits settings where a missed failure costs more than a false alarm. Model 104 also suggests extending the bank beyond position 800, since its failures begin later; we have not tested such a bank. For the repair, model 53 suggests monitoring the drift within the training length during fine-tuning, which the likelihood anchor does not constrain at this scale. The side effect decides the outcome only when the original model is already safe, as for DPD's false positive: a run on seed 1 with a comparable within-length shift (+0.0171 in drift error) still went from 240 crossings to 0.