# KUKULKAN: A STATE SPACE MODEL FOR TEMPORAL POINT PROCESSES WITH INTERPRETABILITY

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

## ABSTRACT

We write the temporal point process as a state space model and use the parallel scan of Mamba to model the excitation kernel explicitly: the Hawkes excitation is the state of a diagonal linear system, its compensator has a closed form and its branching ratio is the steady-state gain. Through structured state space duality the same computation is an attention over past events, which places the classical kernel, the attention models and the state space models in one form. The Mamba Hawkes process and S2P2 are built on that form but give the branching ratio up in exchange for expressivity. Kukulkan keeps it and relaxes it from a number to an interval. The context selects every channel's decay rate and the background rate freely, and the output map only through a gate between zero and one, so the likelihood stays exact and the offspring of every event lies below a number fixed at its birth. The branching ratio is partially identified, with a bound, a conditional estimate and a simulated estimate. Under an evaluation protocol that removes the quadrature bias of the standard held-out likelihood, Kukulkan matches the unconstrained neural output map on seven of nine public datasets and improves on the model with an exact branching ratio on all. On exchange order flow it improves on the unconstrained output map while bounding the offspring of every order, and its estimate of the market stays sub-critical.

## 1 INTRODUCTION

In the recent literature on neural point processes it is often pointed out that interpretability matters (Chang et al., 2025; Berghaus et al., 2026). A model that predicts the intensity $\lambda$ correctly is still of limited use if it cannot say where an event came from, which event caused the current one, or how many events self-excitation alone would produce. A Hawkes fit answers all three, which is why its branching ratio $\rho$, the expected number of events that one event causes, is the object of study in market microstructure (Bacry et al., 2015; Hardiman et al., 2013). The selective state space point processes that lead the recent evaluations inherit the Hawkes recurrence and compute it with the selective scan of Gu & Dao (2023). The Mamba Hawkes process (Gao et al., 2024) and S2P2 (Chang et al., 2025) read that recurrence through unconstrained nonlinear output maps, and the three answers are lost: according to Chang et al. (2025), the model "forgoes the interpretability of the parameters and latent states of linear Hawkes processes". Throughout, a model is interpretable when it reports such quantities, and a quantity is identified when the fitted model and the data determine it as a number, partially identified when they determine only an interval.

We introduce Kukulkan, named after the feathered serpent of Mesoamerican myth, a snake given the plumage of a bird: a Hawkes process given the selection of a state space model. Its context, a summary of the past events and of any covariates known at the event, freely selects how fast each component of the excitation decays and how large the background rate $\mu$ is, and selects how the excitation enters the intensity only through a gate between zero and one. The likelihood stays exact, and the expected number of offspring of every event stays below a number fixed when the event occurs, whatever the context does afterwards (Proposition 5). The branching ratio is thus relaxed to an interval rather than dropped, and it has three estimates which the paper reports together: the bound, a conditional estimate along the observed history that has a closed form, and a simulated estimate along paths the fitted model generates.



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.

Three findings carry the argument. The held-out likelihood by which recent evaluations compare these models carries a quadrature bias larger than the differences it reports, and every number here is computed under a protocol that removes it (Section 3). A correct likelihood still cannot see $\rho$: the value a selective model states disagrees with its own simulation, and when $\mu$ may move, fits of equal likelihood reach opposite conclusions about the market. Context and a checkable $\rho$ are nevertheless compatible: on the public datasets Kukulkan matches the unconstrained model and improves on the one with an exact branching ratio, and on exchange order flow it improves on both while bounding the offspring of every order, with a sub-critical estimate of the market (Section 6).

## 2 Related work

The Hawkes process is fitted for what its parameters say, above all in market microstructure, where the branching ratio measures how much of the order flow is generated by the flow itself (Bacry et al., 2015). That measurement has a history of disagreement. Hardiman et al. (2013) estimated the ratio close to one and read the market as critical, Filimonov & Sornette (2015) showed that a drifting $\mu$ or a kernel of the wrong shape is absorbed into the excitation term and pushes the estimate toward one, and Wheatley et al. (2019) found that once $\mu$ may move the evidence for criticality disappears. Regime models let the kernel change with a latent or observed state (Wang et al., 2012; Morariu-Patrichi & Pakkanen, 2022), and Jin & Huang (2026) give conditions under which hidden processes behind such changes are identified. Non-parametric estimators recover the kernel as a function (Lewis & Mohler, 2011; Bonnet & Sangnier, 2025; Kim & Iwata, 2026) and keep the branching ratio as its integral. Neural temporal point processes replaced the kernel by a network and gave that integral up. The Neural Hawkes Process reads a continuous-time recurrent state through a softplus (Mei & Eisner, 2017), intensity-free models parameterise the law of the next interval directly (Shchur et al., 2020), and the Transformer, Self-Attentive and Attentive Neural Hawkes Processes replace the recurrence by causal self-attention (Zuo et al., 2020; Zhang et al., 2020; Yang et al., 2022). None of these has a branching ratio.

Attention computes at every position a weighted sum over the past with weights normalised to one (Vaswani et al., 2017). Linear attention drops the normalisation, which turns the sum into a recurrence (Katharopoulos et al., 2020). Structured state space models reached the same recurrence from the other direction. S4 and SpaceTime showed that a linear recurrence with fixed dynamics competes with attention on long sequences (Gu et al., 2022; Zhang et al., 2023), Mamba let the dynamics depend on the input through a selection computed by a parallel scan (Gu & Dao, 2023), and Mamba-2 showed that with a scalar decay the unrolled recurrence is a structured matrix computable as an attention or by a chunked scan, the structured state space duality (Dao & Gu, 2024). The Hawkes process with exponential kernels is such a recurrence, known since Oakes (1975), and three recent models exploit it. The Mamba Hawkes process encodes events with Mamba blocks whose step is the interval between events (Gao et al., 2024), S2P2 stacks latent Hawkes layers with input-selected decay between nonlinearities (Chang et al., 2025), and the hyper Hawkes process lets a small recurrent network rewrite the latent dynamics after every event (Boyd et al., 2025). All three read the state through a nonlinearity, which is where the branching ratio is lost.

Since Zuo et al. (2020), neural temporal point processes have been compared by their held-out log-likelihood, with the integral of the intensity estimated from points drawn uniformly inside each interval, together with the accuracy of the next type and the error of the next time. Xue et al. (2024) standardised the datasets and the protocol in the EasyTPP library, which re-implements these models, hosts later ones such as S2P2, and replaced the random points by a fixed grid per interval. Bosser & Taieb (2023) documented that published numbers are inconsistent across papers, and Yang et al. (2022) footnote a defect in the reference code of Zuo et al. (2020) that scores each event from a state that has already seen it. Generative models of event sequences are instead evaluated on statistics of sampled sequences (Lüdke et al., 2026; Berghaus et al., 2026), and Riedlinger et al. (2025) check the calibration of an exact point-process likelihood outside the temporal setting. Section 3 measures the part of the inconsistency that comes from how the integral is discretised.

## 3 MOTIVATION: QUADRATURE BIAS

Temporal point process models output an intensity $\lambda$, a rate. Unlike the evaluation of time series models, where predicted expected values or quantiles are compared with the outcomes through statistics such as the mean squared error, the evaluation of a point process rests on sampling. A held-out sequence $S = \{(t_j, k_j)\}_{j=1}^{L}$ has $L$ events on $[0, T]$, event $j$ at time $t_j$ with type $k_j \in \{1, \dots, K\}$; $\mathcal{H}_t$ is the history before $t$, $\lambda_k(t \mid \mathcal{H}_t)$ the model's rate for type $k$, and $\lambda = \sum_k \lambda_k$ the total rate. The log-likelihood of the sequence is

$$\ell(S) = \sum_{j=1}^{L} \log \lambda_{k_j}(t_j \mid \mathcal{H}_{t_j}) \; - \; \Lambda, \qquad \Lambda = \int_0^T \lambda(t \mid \mathcal{H}_t)\, dt, \tag{1}$$

The second term $\Lambda$ is the compensator, the number of events the model expected over the window, subtracted so that the model pays for every event it predicted at times when nothing happened. The first term is exact for any model. The compensator is exact only when the rate is linear in a state, as in the Hawkes process; for a neural network it has no closed form and is replaced by a sum over a finite set of points inside each interval between consecutive events,

$$\widehat{\Lambda} = \sum_{j=2}^{L} (t_j - t_{j-1}) \, \frac{1}{M} \sum_{i=1}^{M} \lambda(u_{j,i} \mid \mathcal{H}_{u_{j,i}}), \qquad u_{j,i} \in (t_{j-1}, t_j), \tag{2}$$

This replacement is a discretisation, used in training and in the reported number alike. The $M$ points $u_{j,i}$ are drawn uniformly at random inside the interval, as in the Transformer Hawkes Process and its successors (Zuo et al., 2020), or placed on a fixed grid of equally spaced positions from the previous event to the current one, as in EasyTPP (Xue et al., 2024).

Any discretisation replaces the area under the rate by an average height times the width of the interval, and its error, the quadrature bias, depends on how the rate moves between the points. For a smooth rate it is small. For a Hawkes-type rate it is not: the rate jumps at each event and then decays, so it changes fastest right after the previous event, where a few points describe it worst. Random points give the right area on average, at the price of noise. Fixed points give an error with a sign: a point on the spike counts it as if it lasted a whole sub-interval, and the grid used in practice puts its first point on the previous event itself, so it over-estimates a decaying rate and under-estimates the likelihood. The error also depends on the model, because a model trained against a discretised objective is penalised only at the points. If the points are the same at every step it learns where they are, moves rate mass between them, and reports a likelihood that is over-estimated. The bias therefore belongs to the pair of a rate shape and a discretisation, and its sign depends on how the model was trained.

Figure 1 shows the fixed-grid case inside one interval of a simulated Hawkes process with a sharp spike after every event (Appendix B). Each panel asks how well the grid estimates the shaded area under one curve from its values at the ten red grid points; the dotted line is the height the grid reports. In A the curve is the true rate: the first grid point catches the spike and the grid reports more than twice the true area, while THP, which has smoothed the spike away, is read correctly but fits the data badly. In B the curve is S2P2 trained against the grid: it has learned a rate that is zero at the first grid point and very large just after it, where no grid point falls, so the grid reports a fraction of the model's own area and, over the test set, a likelihood above that of the true parameters, which no model can reach under exact integration. The same signs appear on the public datasets (Appendix D.1), by thousands of nats where the intervals are long, and a smaller bias of the same kind comes from tied time stamps.

Even when the Hawkes structure is built into a state space model, so that a branching ratio can be read from the parameters, that number is often misspecified and hard to separate from $\mu$, the background rate. We measured both inside a selective Hawkes model of our own, a one-layer member of the S2P2 class with a linear output map, the unconstrained model below. The branching ratio it states is the kernel integral with the context frozen at the event, while the offspring accumulate under the contexts that follow. Counting them in the model's own simulation puts the stated value off by a few percent on some datasets and by a factor of three on others (Table 13 in Appendix D.1). The count is also not identified when $\mu$ may move. A burst is explained equally well by events causing events or by $\mu$ rising for a minute, and on exchange order flow seeds of equal likelihood disagree on

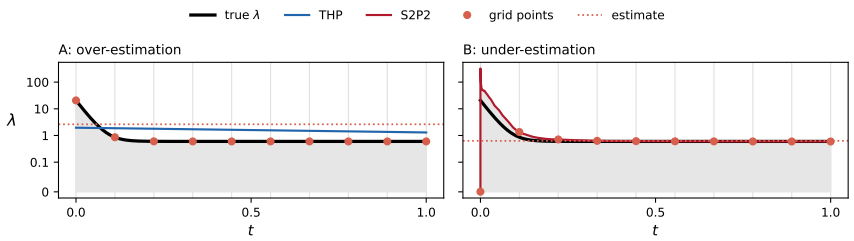



Figure 1: Quadrature bias of the fixed grid: A, area over-estimated. B, area under-estimated.

whether the market is sub-critical or super-critical when $\mu$ follows the context and agree once $\mu$ is fixed (Table 19 in Appendix D.2), the ridge that Filimonov & Sornette (2015) and Wheatley et al. (2019) describe for the classical model. Two questions follow: why does the reported likelihood depend on the discretisation, and why is the stated branching ratio not the count it names? Section 4 answers both, and Section 5 builds the model that avoids them.

## 4 Why quadrature bias exists in neural Hawkes processes

Quadrature bias is inevitable in current neural point process models because their intensity is a nonlinear function of a hidden state, whose integral has no closed form, so the compensator is evaluated at sampled times in training and in the reported number alike. A deterministic sample is the fixed grid of Section 3 and carries its bias. A random sample is unbiased but noisy, and it costs an intensity evaluation at every drawn point. A third problem is specific to selective models, whose dynamics depend on the input: they can learn a rate that is small at the fixed sample points and large between them, which raises the sampled objective without raising the true likelihood. Because of this bias we chose a linear output map for our model, whose integral has a closed form.

A Hawkes process with exponential kernels has the state space form of Oakes (1975),

$$\dot{h}(t) = -\beta \odot h(t) \text{ between events}, \qquad h(t_j^+) = h(t_j^-) + x_j, \qquad \lambda_k(t \mid \mathcal{H}_t) = \mu_k + \sum_{n=1}^{N} C_{kn}\, h_n(t), \tag{3}$$

with one state entry per channel $n$, decay rates $\beta_n$, inputs $x_j = B_{:,k_j}\, g(m_j)$ given by a non-negative matrix $B$ scaled by the mark, and a non-negative output map $C$ (Definition 2 in Appendix A.1). Channel $n$ is the kernel $e^{-\beta_n \tau}$, and when $N = K$ and $B = I$, equation 3 is the classical Hawkes process with exponential kernels; $k'$ denotes the type of a source event. Structured state space duality (Dao & Gu, 2024) states that the output of such a recurrence equals a masked matrix product over the inputs, which is a linear attention, and the Hawkes process inherits this.

**Proposition 1** (state space duality of the attention-based Hawkes process)**.** *Let $L_{js,n} = e^{-\beta_n (t_j - t_s)}$ for $s < j$. The state before event $j$ and the intensity that scores it are*

$$h(t_j^-) = \sum_{s<j} L_{js} \odot x_s, \qquad \lambda_k(t_j^-) = \mu_k + \sum_{s<j} M_{js}^{(k)}, \qquad M_{js}^{(k)} = \sum_n C_{kn}\, L_{js,n}\, x_{s,n}, \tag{4}$$

*so the intensity is a masked matrix product, or linear attention, in which the input $x_s$ is the value of past event $s$, the row $C_{k,:}$ is the query of the target type, and $M_{js}^{(k)}$ is a content score times the decay over the elapsed time. The same numbers follow from the recurrence $h(t_j^-) = e^{-\beta \Delta_j} \odot \big(h(t_{j-1}^-) + x_{j-1}\big)$ with $\Delta_j = t_j - t_{j-1}$, or from a chunked scan (Dao & Gu, 2024).*

The neural Hawkes processes keep this structure and change two things. The attention-based models replace the weights $M_{js}^{(k)}$ by softmax scores $a_{js}$ over learned keys and queries, so the state at event $j$ is $z_j = \sum_{s<j} a_{js} v_s$ with learned values $v_s$ (Zuo et al., 2020; Zhang et al., 2020; Yang et al., 2022). The state space models keep a recurrence like equation 3, select its decay from the input, and stack such layers with nonlinearities between them (Gao et al., 2024; Chang et al., 2025). In both families the intensity on the interval after event $j-1$ is a nonlinear function of the state and of the elapsed time,

$$\lambda_k(t_{j-1} + \tau \mid \mathcal{H}) = \sigma\big(f_k(z_{j-1}, \tau)\big), \qquad \sigma = \text{softplus}, \tag{5}$$

where $f_k$ is linear in $\tau$ for THP, an exponential decay toward a learned level for SAHP and NHP, and a stack of layers for S2P2. The first question of Section 3 arises at the softplus.

**Proposition 2** (the compensator of the two families)**.** *(i) Under the linear output map of equation 3, the compensator of type $k$ on the interval $(t_{j-1}, t_j]$ is*

$$\int_{t_{j-1}}^{t_j} \lambda_k \, dt = \mu_k \Delta_j + \sum_n C_{kn} \, \frac{h_n(t_{j-1}^+) - h_n(t_j^-)}{\beta_n}, \tag{6}$$

*the decrease of the state over the interval divided by the decay rate, channel by channel. (ii) Under equation 5 with $f_k$ linear in $\tau$, the compensator $\int_0^{\Delta_j} \sigma(f_k(z_{j-1}, \tau)) \, d\tau$ is a dilogarithm and has no elementary closed form.*

Part (i) follows from $\int_0^{\Delta} e^{-\beta\tau} d\tau = (1 - e^{-\beta\Delta})/\beta$ and part (ii) from $\int \log(1 + e^{a\tau+b}) \, d\tau = -\mathrm{Li}_2(-e^{a\tau+b})/a$ (Appendix A.2). The heads of the other neural models contain the same function of a decaying argument and are no simpler, so every softplus output is computed by the discretisation equation 2, while the linear output map needs no sample point. The discretisation then meets the shape of the rate: no input arrives between events, so on an interval the rate is the output of a decaying state, largest right after the previous event and falling at the decay rates, and the neural models learn the same shape because their $f_k$ falls in $\tau$. For a single channel the error of a fixed grid has a closed form.

**Proposition 3** (fixed-grid bias of a decaying channel)**.** *Let a channel contribute $a\, e^{-\beta\tau}$ on an interval of length $\Delta$, and let the grid place $M$ points at $u_i = (i-1)\Delta/(M-1)$ for $i = 1, \ldots, M$. The exact area and the grid estimate are*

$$\Lambda = \frac{a\,(1 - e^{-\beta\Delta})}{\beta}, \qquad \widehat{\Lambda} = \frac{a\Delta}{M} \sum_{i=1}^{M} e^{-\beta u_i} = \frac{a\Delta}{M} \cdot \frac{1 - e^{-\beta\Delta M/(M-1)}}{1 - e^{-\beta\Delta/(M-1)}}, \tag{7}$$

*Hence $\widehat{\Lambda}/\Lambda \to 1$ as $\beta\Delta \to 0$ and $\widehat{\Lambda}/\Lambda \to \beta\Delta/M$ as $\beta\Delta \to \infty$. The grid over-estimates every channel whose decay rate exceeds $M/\Delta$, by a factor that grows with the interval. For $M$ points drawn uniformly at random in the interval, $\mathbb{E}\,\widehat{\Lambda} = \Lambda$ for every $\beta$ and $\Delta$.*

Panel A of Figure 1 shows this case. A fixed-grid objective penalises the rate only at the grid points and is indifferent to it everywhere else (Appendix A.2), so a selective model can hold a channel that decays within a fraction of the shortest interval and cancel it exactly at the first grid point, leaving a bump between two grid points that the exact compensator counts in full and the grid not at all. Panel B of Figure 1 shows this shape, found by gradient descent. Random sample points remove the incentive at the price of a noisy objective, and a linear output map removes the sample points altogether.

The second question arises from the same two changes. A branching ratio is the integral of what one event adds to the intensity, and it exists only when the intensity is additive over events. Under the linear output map of equation 3 it is, and the integral is $\sum_n C_{kn} B_{nk'} g(m)/\beta_n$ (Proposition 4). Under the softplus of equation 5 no part of the intensity can be attributed to one event, and under a softmax each event's share depends on its competitors, so the attention-based models have no branching ratio at all. The selective state space models keep a linear recurrence but select the decay and the output map from the context of later events, so the number they can state at an event, the integral of its contribution with the context frozen, is not what the event produces, because the output map that reads its remaining excitation is chosen afterwards. This is the stated value of Section 3, and when $\mu$ is also selected by the context, a burst can be explained by excitation or by a rise in $\mu$ with the same likelihood, which is the ridge. The model of the next section keeps a linear output map and restricts what later context may change about an earlier event.

## 5 KUKULKAN: AN INTERPRETABLE NEURAL HAWKES PROCESS

Table 1 places Kukulkan in the model class of Section 4. It inherits from both branches of the neural Hawkes processes: from the attention-based models the context, a learned summary of the past that enriches the intensity with what the events contained, and from the state space models the selection

Table 1: One form for the model class: the input each model writes at an event, the decay it applies to it, and the output that reads the state into the intensity, the output map.

| | input | decay | output |
|---|---|---|---|
| Hawkes (exponential) | fixed $\alpha\, g(m)$ | $e^{-\beta\Delta}$ | $\mu + Ch$ |
| Hawkes (non-parametric) | fixed, per type | $\phi(\Delta)$, a free function | $\mu + Ch$ |
| NHP | LSTM gates | per cell, toward a target | softplus of the state |
| THP / SAHP / SB-THP | query·key | none / in the head / in the logits | softplus head |
| MHP | $C_j \cdot B_s$ | $e^{\Delta A}$ | THP head |
| S2P2 / hyper Hawkes | latent linear Hawkes | $e^{-\beta_j \Delta}$, selected | softplus of the stack |
| Kukulkan | $B_{k_s}\, g(m_s)$ | $e^{-\beta_j \Delta}$, selected | $\mu(c_j) + W s(c_j) \beta_j h$ |

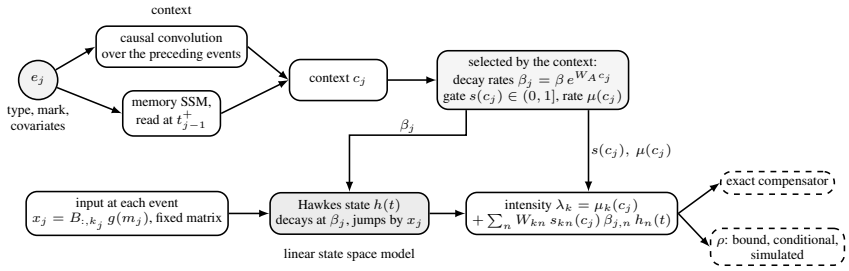



Figure 2: Kukulkan. The content of each event feeds a causal convolution and a memory whose summary is the context. The context selects the decay rates, the gate and $\mu$, while the input is a fixed matrix. The Hawkes state decays at the selected decay rates and jumps by the input, the output map is the fixed weight times the decay rate times the gate, the compensator is the decrease of the gated state, and the branching ratio has a bound, a conditional estimate and a simulated estimate.

of the dynamics from the input. It gives up the one thing both branches share, a nonlinear output map, and with it some expressivity, in exchange for an exact likelihood and a branching ratio that can be checked. The context selects the decay rates and the background rate freely, the input is fixed, and the output map is linear with a gate, so that an earlier event's excitation can be shortened or reduced but never amplified (Figure 2).

### 5.1 Context enrichment: the legacy of attention

In the matrix product of equation 4 each past event supplies a value, its input $x_s$, the target type supplies a query, the row $C_{k,:}$, and the weight is a content score times the decay over the elapsed time. Kukulkan keeps this reading with two differences from a transformer: the weights are not normalised, so a column sum is what one event gives to all later ones, whereas a softmax makes each event's share depend on its competitors and, on simulated Hawkes data, reports a branching ratio 2.6 times too large (Appendix C); and the output map is linear, so the intensity is additive over events. What Kukulkan takes from attention is the content. Each event has a content vector, an embedding of its type, of the logarithm of its mark and of any covariates known at the event. The context that applies on an interval combines a short causal convolution over the preceding events with a memory, a second diagonal state space model over the same content vectors with learned decay scales, read at the previous event rather than at the current time so that the length of the current interval cannot enter it (Appendix A.1).

### 5.2 Selective decay: the legacy of state space models

In the unconstrained model of Section 3 the context selects every decay rate, every input, $\mu$ and the output map, and the output map is the problem: selected later, it is applied to the remaining excitation of every earlier event and changes that event's contribution. Kukulkan keeps the selected decay rates and $\mu$, fixes the inputs, and constrains the output map. The output weight of a channel is a fixed weight $W_{kn}$, times the channel's current decay rate, times a gate in $(0, 1]$ that the context chooses for each target type and channel, so the context may suppress an event's influence but never amplify it. Because the output map follows the decay rate, the kernel an event leaves in a channel is a probability density over the time at which its influence arrives: the decay rates set its shape, its

mass is one on every path, and the gate alone decides how much of the influence arrives. With the gate fixed at one the branching ratio is a single number, exact on every path; we call this special case the exact-$\rho$ model, and every Kukulkan fit is trained further from it (Appendix C.1).

Proposition 5 (Appendix A.1) states the consequence. Whatever decay rates and gates the context selects, the compensator on each interval is the decrease of the gated state, so the likelihood is exact, and the expected number of direct offspring of every event lies between zero and its fixed weight, set at its birth by its type and mark, which we call the event's upper bound. The proof is three lines, because the integral of an event's contribution over the future is its fixed weight times the mean of the gate under a probability measure. The gate is one-sided because the data ask for one direction: on every dataset the context earns its likelihood by turning excitation down, with a median gated-to-ungated ratio of $0.58$ on the order-flow data and $0.18$ on Taxi, and a symmetric band around one reaches less of that improvement than a gate whose lower limit is zero (Appendix C.1). Two diagnostics complete the family, a latent regime entering through the decay rates and a softplus output of the same state without $\rho$ (Appendices A and C); the model fits the public datasets in seconds to minutes.

### 5.3 Interpretation

The linear output map also gives the branching ratio a definition.

**Proposition 4** (branching ratio of the linear output map)**.** *For the linear output map of Definition 2, the expected number of direct offspring of type $k$ from an event of type $k'$ with mark $m$ is*

$$\rho_{kk'}(m) = \int_0^\infty \sum_n C_{kn} B_{nk'} g(m)\, e^{-\beta_n \tau}\, d\tau = \sum_n \frac{C_{kn} B_{nk'} g(m)}{\beta_n}. \tag{8}$$

It follows from $\int_0^\infty e^{-\beta\tau} d\tau = 1/\beta$ (Appendix A.2). The branching ratio is the steady-state gain of the linear system and is additive over events because the output map is; a softplus of the sum has no such quantity. The proposition and the band give three estimates of $\rho$. The first is the bound. The fit does not pin the upper bound itself: fitted jointly with the gate, the fixed weights rise while the gate scales them down, so the bound is not an estimate, and it can be fixed at the value of the exact-$\rho$ model at a cost in likelihood (Appendix C.1). The second is conditional and exact. Because the output map is linear, the offspring of an observed event along the observed sequence has a closed form, the decrease of its gated excitation on each later interval plus the mass left at the end of the window (Appendix A.1); it is the branching ratio conditional on the history that occurred, the quantity a microstructure study asks for, and it exists for every model with a linear output map. The third is simulated, the same integral along paths that the fitted model generates (Section 6.2). For the exact-$\rho$ model the three coincide with $\rho$; for Kukulkan they are reported side by side in Section 6 and Appendix C.1.

## 6 Experiments and results

### 6.1 Setup

All models are scored with one convention: the negative log-likelihood (NLL) per event over events $2, \ldots, L$, with the compensator computed exactly where it has a closed form and otherwise on 256 log-spaced cells checked at 1024, and next-type accuracy as the type with the largest intensity. The neural baselines are re-implemented on one base with the defect of the THP reference code removed, and S2P2 is run from the released code, trained on random points and re-scored exactly (Appendix B). The nine public datasets are the EasyTPP releases (Xue et al., 2024). The order-flow dataset is the tick-by-tick flow of six liquid instruments on one stock exchange over ten trading days, seven for training, one for validation and two for testing: 2.9M aggressive orders in windows of 200, typed buy or sell, marked by their volume, timed uniformly inside the 200 ms batch in which the feed delivers them. Kukulkan's width is chosen on the validation set and its rows are means over three seeds; the seed spread, the difference between the best and the worst seed, is $0.001$–$0.03$ nats except $0.5$ on MIMIC-II. S2P2 is a mean over two seeds where two were run and the other baselines are one seed, so differences below $0.02$ nats are within the seed spread.

Table 2: Public datasets: NLL per event (lower is better). Rows above the line have no branching ratio. The exponential Hawkes process reports $\rho$ exactly and Kukulkan reports an upper bound and a conditional offspring per event. The variants of Kukulkan are in Appendix C.1, Table 5.

| model | Taxi | StackOv. | Retweet | Amazon | Taobao | Volcano | Earthq. | MIMIC-II | Financial |
|---|---|---|---|---|---|---|---|---|---|
| Hawkes, power law | 0.302 | 2.260 | 6.553 | 2.183 | $-1.254$ | 6.430 | 1.840 | 1.667 | $-6.723$ |
| Hawkes, nonparametric | 0.221 | 2.314 | 6.580 | 1.758 | $-1.285$ | 6.444 | 1.839 | 1.327 | $-6.822$ |
| THP | $-0.341$ | 2.323 | 6.694 | 2.095 | $-0.992$ | 6.868 | 1.904 | 1.268 | $-5.767$ |
| MHP | $-0.321$ | 2.423 | 6.628 | 2.139 | $-0.938$ | 6.854 | 1.869 | 1.538 | $-5.983$ |
| SAHP | $-0.486$ | 2.217 | 6.356 | 1.402 | $-1.239$ | 6.568 | 1.770 | 1.381 | $-6.665$ |
| S2P2 | $-0.522$ | 2.171 | $\approx 5.11$ | $-0.74$ | $-1.351$ | 6.052 | 1.712 | 1.393 | $-7.376$ |
| Hawkes, exponential | 0.308 | 2.291 | 6.548 | 2.109 | $-1.252$ | 6.464 | 1.830 | 1.587 | $-6.776$ |
| Kukulkan | $-0.305$ | 2.169 | 6.433 | 1.919 | $-1.277$ | 6.311 | 1.691 | 2.128 | $-6.780$ |

### 6.2 Results

Section 3 raised two problems, and Table 2 answers the first: Kukulkan's compensator is the decrease of the gated state (Proposition 2), so its likelihood is the same number under any quadrature rule, and every other model is re-scored exactly. Kukulkan improves on the exponential Hawkes process it contains by $0.1$–$0.6$ nats on every dataset and is within the seed spread of the unconstrained output map on seven of nine, so the bound of Proposition 5 costs nothing in likelihood there; the two exceptions, Retweet and MIMIC-II, and the other members of the model class are discussed in Appendix C.1. It matches S2P2 on StackOverflow and Taobao with a tenth of the parameters. S2P2 is better by $1.5$ nats on Amazon, where the best timing model is a hazard that rises through the interval, which no decaying intensity produces, by $1.2$ on Retweet and by $0.3$ on Volcano and MIMIC-II; where a type must be switched off entirely, a linear output map is $0.1$–$0.6$ nats behind a softplus output of the same state, since the gate can switch a channel off but not a type (Appendices D.1 and C.1).

The second problem is answered by Proposition 5 and by a simulation. For each fitted model we take the state after the last event of 300 test windows, inject one more event, simulate 200 continuations by thinning with the model's own recursion, and integrate the injected event's contribution along each path. For the exact-$\rho$ model the integral equals $\rho$ on every path to three decimals; for the unconstrained model the value with the context frozen overstates it by a factor of three on Taxi (Table 13). Figure 3 shows Kukulkan's three estimates inside the bound on every dataset. The conditional and simulated estimates agree on order flow, Financial, Taobao and StackOverflow, and differ by a third on Retweet and Volcano and by half on Taxi and Amazon, where the sequences are cascades or alternations that the model's own paths do not reproduce; that disagreement is itself a diagnostic, visible only because both estimates exist. Whether the number is identified from the data is a separate question, which the same figure answers on order flow: with $\mu$ fixed the seeds agree on a sub-critical market, with $\mu$ selected by the context they scatter along the ridge.

Order flow carries a context of its own, the recent flow, its imbalance and the time of day, and a Hawkes model of it is fitted for one number, the share of the activity that is self-generated. Table 3 shows three things. First, S2P2 is better than every other model by $0.13$ nats, with its advantage in the intervals below $0.15$ s, where its stack of nonlinearities can shape the spike after each event; we claim no likelihood advantage over it, and its training takes 71 minutes against one minute for Kukulkan. Second, the context improves on the classical kernel by $0.11$ nats in three steps that Appendix C.1 separates, the decay rates, the context selection of $\mu$ and the gate, and the gate's step is the whole advantage of the unconstrained output map, with a bound in place of a stated value. Third, the conditional offspring of the observed orders average $1.06$ and the simulated offspring $1.00$, while the fitted upper bound of $2.5$ to $4.1$ per event changes from seed to seed when the conditional value does not. Whether $\rho$ is identified depends on $\mu$. With $\mu$ fixed, the three seeds of the exact-$\rho$ model agree at $0.86$–$0.91$ for buy orders and $0.75$–$0.78$ for sell orders, sub-critical on windows of 200 orders and consistent with Hardiman et al. (2013) at this window length; with a context-selected $\mu$ they sit on the ridge of Section 3 (Table 19). Run forward for fifteen minutes with the state rebuilt from the last 200 orders at every step, Kukulkan diverges on none of sixteen paths and generates $7{,}100$ orders per hour against the data's $7{,}800$, although no fitted model reproduces the over-dispersion of the data at every time scale (Appendix D.2).

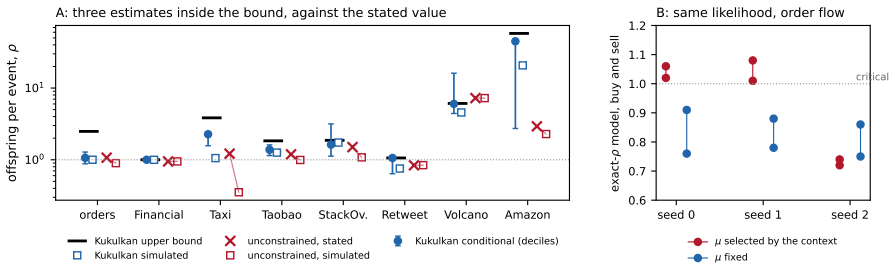



Figure 3: Interpretability in use. A: per dataset, Kukulkan's upper bound (bar), conditional offspring along the observed path (dot, deciles) and simulated offspring (open square), against the branching ratio the unconstrained model states (cross) and the offspring counted in its own simulation (open square). On Volcano and Amazon every fitted model is super-critical, since eruption and purchase sequences are cascades rather than stationary processes, and the estimates are dominated by the mass left after the window. They stay inside the bound. B: on exchange order flow, three seeds of the exact-$\rho$ model within $0.002$ nats of each other, with $\mu$ selected by the context and with $\mu$ fixed. The dotted line is the critical value.

Table 3: Exchange orders: NLL per event, next-type accuracy and the reported branching ratio for (buy, sell); Kukulkan reports its upper bound and the conditional offspring, equation 15.

| model | params | NLL | acc | $\hat{\rho}$ row sums (buy, sell) |
|---|---|---|---|---|
| Hawkes, power law | 10 | 0.690 | 0.590 | |
| Hawkes, nonparametric | 34 | 0.657 | 0.654 | |
| THP / SB-THP / MHP | 67k | 0.702 / 0.653 / 0.631 | 0.618 / 0.627 / 0.629 | n/a |
| SAHP | 68k | 0.571 | 0.681 | n/a |
| S2P2 | 33k | 0.361 | 0.715 | n/a |
| Hawkes, exponential | 58 | 0.652 | 0.652 | $0.92,\ 0.74$ |
| Kukulkan | 2.5k | 0.543 | 0.668 | bound 2.5–4.1 per seed, conditional 1.06 |

## 7 Conclusion

A held-out likelihood is sensitive to how the intensity is integrated, by more than the differences between current models, and it cannot tell whether the fitted model is the process it claims to be. Kukulkan answers both by construction: it keeps the context from which neural models obtain their likelihood and grants it one selection fewer than Mamba does, the output map, so that the compensator is exact, the offspring of every event is bounded by a number fixed at its birth, and the branching ratio can be estimated three ways and checked. On the public datasets this costs nothing against the unconstrained output map; on exchange order flow Kukulkan improves on it and its estimate of the market stays sub-critical.

The bound is only as tight as the fitted upper bound, which changes from seed to seed while the conditional offspring does not; fixing the upper bound costs likelihood, and the value depends on the choice of $\mu$, the ridge that the model makes visible but does not remove. No model of this class is stable on MIMIC-II from a cold start, the neural baselines are single-seed re-implementations at one width, and the order-flow data cover ten days of one exchange.

Two extensions keep the closed-form compensator and the bound and add expressivity, a gate that varies inside the interval and paired channels with a rising kernel. In a preliminary run (Appendix C.1) they gain half a nat on Amazon, but fitted jointly they inflate the bound, and with the fixed weights frozen the gain is $0.16$ nats; making the larger gain hold with a tight bound is the next step. A variational latent $\mu$ and an offspring constraint inside training are the other natural steps.

### AI Use Statement

We used a generative AI coding assistant (Claude Code, Anthropic) under the authors' direction for tasks with required disclosure: implementing the model and the baselines, running the experiments, formulating and checking the mathematical statements of Sections 4 and 5, and drafting and editing

the manuscript. We also used it for literature search and code review. No dataset was generated, cleaned or relabelled by an AI system beyond the simulators described in the paper. All AI-assisted work was reviewed by the authors: every model was validated against a known-truth oracle, the derivations were checked by hand, and the defect in the THP reference code was verified by reading the original source. We take responsibility for the final content of this work.

### Ethics Statement

The public datasets are used under their EasyTPP release. The order-flow data are public market data, tick-by-tick trades of listed instruments, and contain no personal information.

### Reproducibility Statement

Sections 4 and 5 and Appendix A give the model, the closed-form likelihood and the chunked algorithm in full. Appendix B lists the datasets, the preprocessing, the hyperparameters, the deviations of our baseline re-implementations from their papers, and the patched lines of the THP reference likelihood. The code will be made available upon acceptance, including the model, the baselines, the S2P2 runner and exact re-scorer, the simulation test of Section 6.2, the simulators, the known-truth tests and the configuration files behind every number in the tables. The order-flow dataset cannot be redistributed; its construction script is included, and the public datasets reproduce all cross-model comparisons.

## References


Emmanuel Bacry, Iacopo Mastromatteo, and Jean-François Muzy. Hawkes processes in finance. *Market Microstructure and Liquidity*, 1(1):1550005, 2015. arXiv:1502.04592.

David Berghaus, Patrick Seifner, Kostadin Cvejoski, César Ojeda, and Ramsés J. Sánchez. In-context learning of temporal point processes with foundation inference models. In *International Conference on Learning Representations (ICLR)*, 2026. arXiv:2509.24762.

Anna Bonnet and Maxime Sangnier. Nonparametric estimation of Hawkes processes with RKHSs. In *Proceedings of the 28th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2025. arXiv:2411.00621.

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.

Alex Boyd, Andrew Warrington, Taha Kass-Hout, Parminder Bhatia, and Danica Xiao. Hyper Hawkes processes: Interpretable models of marked temporal point processes. *arXiv preprint arXiv:2511.01096*, 2025.

Yuxin Chang, Alex Boyd, Cao Xiao, Taha Kass-Hout, Parminder Bhatia, Padhraic Smyth, and Andrew Warrington. Deep continuous-time state-space models for marked event sequences. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. arXiv:2412.19634.

Tri Dao and Albert Gu. Transformers are SSMs: Generalized models and efficient algorithms through structured state space duality. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, 2024. arXiv:2405.21060.

Vladimir Filimonov and Didier Sornette. Apparent criticality and calibration issues in the Hawkes self-excited point process model: application to high-frequency financial data. *Quantitative Finance*, 15(8):1293–1314, 2015.

Anningzhe Gao, Shan Dai, and Yan Hu. Mamba Hawkes process. *arXiv preprint arXiv:2407.05302*, 2024.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. In *International Conference on Learning Representations (ICLR)*, 2022.

Stephen J. Hardiman, Nicolas Bercot, and Jean-Philippe Bouchaud. Critical reflexivity in financial markets: a Hawkes process analysis. *The European Physical Journal B*, 86(10):442, 2013.

Songyao Jin and Biwei Huang. Causal structure learning in Hawkes processes with complex latent confounder networks. In *International Conference on Learning Representations (ICLR)*, 2026. arXiv:2508.11727.

Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are RNNs: Fast autoregressive transformers with linear attention. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, 2020.

Hideaki Kim and Tomoharu Iwata. A representer theorem for Hawkes processes via penalized least squares minimization. In *International Conference on Learning Representations (ICLR)*, 2026. oral; arXiv:2510.08916.

Akitoshi Kimura. Extended state-dependent Hawkes process for limit order books: mathematical foundation and the reproduction of volatility signature plots. *arXiv preprint arXiv:2604.23961*, 2026.

Erik Lewis and George Mohler. A nonparametric EM algorithm for multiscale Hawkes processes. In *Journal of Nonparametric Statistics*, volume 1, pp. 1–20, 2011.

David Lüdke, Marten Lienen, Marcel Kollovieh, and Stephan Günnemann. Edit-based flow matching for temporal point processes. In *International Conference on Learning Representations (ICLR)*, 2026. arXiv:2510.06050.

Hongyuan Mei and Jason Eisner. The neural Hawkes process: A neurally self-modulating multivariate point process. In *Advances in Neural Information Processing Systems 30 (NeurIPS)*, 2017.

Maxime Morariu-Patrichi and Mikko S. Pakkanen. State-dependent Hawkes processes and their application to limit order book modelling. *Quantitative Finance*, 22(3):563–583, 2022. doi: 10.1080/14697688.2021.1983199.

David Oakes. The Markovian self-exciting process. *Journal of Applied Probability*, 12(1):69–77, 1975.

Tobias J. Riedlinger, Kira Maag, and Hanno Gottschalk. Towards reliable detection of empty space: Conditional marked point processes for object detection. *arXiv preprint arXiv:2506.21486*, 2025.

Oleksandr Shchur, Marin Biloš, and Stephan Günnemann. Intensity-free learning of temporal point processes. In *International Conference on Learning Representations (ICLR)*, 2020.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Ting Wang, Mark Bebbington, and David Harte. Markov-modulated Hawkes process with stepwise decay. *Annals of the Institute of Statistical Mathematics*, 64(3):521–544, 2012.

Spencer Wheatley, Alexander Wehrli, and Didier Sornette. The endo-exo problem in high frequency financial price fluctuations and rejecting criticality. *Quantitative Finance*, 19(7):1165–1178, 2019.

Siqiao Xue, Xiaoming Shi, Zhixuan Chu, Yan Wang, Fan Zhou, Hongyan Hao, Caigao Jiang, Chen Pan, Yi Xu, James Y. Zhang, Qingsong Wen, Jun Zhou, and Hongyuan Mei. EasyTPP: Towards open benchmarking temporal point processes. In *International Conference on Learning Representations (ICLR)*, 2024. arXiv:2307.08097.

Chenghao Yang, Hongyuan Mei, and Jason Eisner. Transformer embeddings of irregularly spaced events and their participants. In *International Conference on Learning Representations*, 2022.

Michael Zhang, Khaled K. Saab, Michael Poli, Tri Dao, Karan Goel, and Christopher Ré. Effectively modeling time series with simple discrete state spaces. In *International Conference on Learning Representations (ICLR)*, 2023. arXiv:2303.09489.

Qiang Zhang, Aldo Lipani, Omer Kirnap, and Emine Yilmaz. Self-attentive Hawkes process. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, 2020.

Simiao Zuo, Haoming Jiang, Zichong Li, Tuo Zhao, and Hongyuan Zha. Transformer Hawkes process. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, 2020. arXiv:2002.09291.

# A Mathematical analysis

## A.1 Model equations

**Definition 1** (marked multivariate Hawkes process)**.** *A marked temporal point process on $[0, T]$ with $K$ types is a sequence of events $(t_j, k_j, m_j)$, $j = 1, \dots, L$, with times $0 < t_1 < \cdots < t_L \le T$, types $k_j \in \{1, \dots, K\}$ and positive marks $m_j$. Its history at $t$ is $\mathcal{H}_t = \{(t_j, k_j, m_j) : t_j < t\}$, and the process is specified by the conditional intensities $\lambda_k(t \mid \mathcal{H}_t)$, $k = 1, \dots, K$, the rate of type-$k$ events at $t$ given the history. It is a Hawkes process when*

$$\lambda_k(t \mid \mathcal{H}_t) = \mu_k + \sum_{t_s < t} \phi_{kk_s}(t - t_s)\, g(m_s), \tag{9}$$

*with a background rate $\mu_k \ge 0$, non-negative kernels $\phi_{kk'}(\tau)$ that give the influence of a source event of type $k'$ on the intensity of the target type $k$ at lag $\tau$, and a non-negative mark function $g$. The compensator of type $k$ is $\Lambda_k(t) = \int_0^t \lambda_k(u \mid \mathcal{H}_u)\, du$, the log-likelihood of a sequence is equation 1, and the branching matrix is*

$$\rho_{kk'} = \mathbb{E}[g(m)] \int_0^\infty \phi_{kk'}(\tau)\, d\tau, \tag{10}$$

*the expected number of direct offspring of type $k$ that one event of type $k'$ produces. Its column sum $\sum_k \rho_{kk'}$ is the expected number of direct offspring of one event of type $k'$, and the process is sub-critical, and has a stationary version, when the spectral radius of $\rho$ is below one.*

With intervals $\Delta_j = t_j - t_{j-1}$, a Hawkes process with exponential kernels $\phi_{kk'}(\tau) = \sum_{n=1}^N C_{kn} B_{nk'} e^{-\beta_n \tau}$ and $N$ channels is

$$\lambda_k(t) = \mu_k + \sum_{t_s < t} \sum_{n=1}^N C_{kn}\, B_{nk_s}\, g_n(m_s)\, e^{-\beta_n (t - t_s)}, \tag{11}$$

When $N = K$ and $B = I$, equation 11 is the classical Hawkes process with one exponential kernel per source type.

**Definition 2** (state space form)**.** *The state $h(t) \in \mathbb{R}^N_{\ge 0}$ decays between events at the decay rates $\beta \in \mathbb{R}^N_{>0}$, jumps at each event $j$ by the input $x_j = B_{:,k_j}\, g(m_j)$, a column of a non-negative table $B \in \mathbb{R}^{N \times K}$ scaled by the mark, and is read into the intensity by a non-negative output map $C \in \mathbb{R}^{K \times N}$, as in equation 3.*

Solving the decay between events gives back the kernel form $\lambda_k(t) = \mu_k + \sum_{t_s < t} \sum_n C_{kn} B_{nk_s} g(m_s) e^{-\beta_n (t - t_s)}$: channel $n$ is the exponential $e^{-\beta_n \tau}$, and a kernel with several time scales is several channels. The state $h(t) \in \mathbb{R}^N$ is what each past event still contributes at $t$. Between events it decays, at events it jumps, and the intensity reads it:

$$h_j^- = e^{-\beta \Delta_j} \odot h_{j-1}^+, \qquad h_j^+ = h_j^- + x_j, \quad x_j = B_{:,k_j}\, g(m_j), \qquad \lambda(t_j^-) = \mu + C\, h_j^-, \tag{12}$$

a diagonal linear state space model with transition $A_j = \mathrm{diag}(e^{-\beta \Delta_j})$, input $x_j$ and a point-process observation model. Unrolling gives $h_j^- = \sum_{s<j} L_{js} \odot x_s$ with $L_{js} = e^{-\beta (S_j - S_s)}$ and $S_j = \sum_{r \le j} \Delta_r$. The history-weight matrix $L$ is 1-semiseparable. The compensator on an interval is $\int_{t_{j-1}}^{t_j} \lambda_k = \mu_k \Delta_j + \sum_n C_{kn} h_{j-1,n}^+ (1 - e^{-\beta_n \Delta_j})/\beta_n$, and the branching-ratio matrix is the steady-state gain of the linear system,

$$\rho_{kk'} = \int_0^\infty \phi_{kk'} = \sum_n C_{kn}\, \mathbb{E}[g_n(m)]\, B_{nk'}/\beta_n = -C\, \mathbb{E}[G]\, A_c^{-1} B, \qquad A_c = -\mathrm{diag}(\beta), \tag{13}$$

the expected number of direct offspring of type $k$ from an event of type $k'$. In the attention view of the masked product, the value is $v_s = g(m_s)$, the key $\kappa_s = B_{:,k_s}$, the query $q_j = C_{k,:}$ for target type $k$ and the weight $w_{js} = \sum_n q_{j,n}\kappa_{s,n}L_{js,n}$.

Kukulkan's context is $c_j = \tanh(W[\mathrm{conv}_j;\, m(t^+_{j-1})])$, a depthwise causal convolution over the four events before $j$ and a memory $m$, a second diagonal state space model over the event embeddings $e_j = \mathrm{Emb}(k_j) + w \log m_j \,(+W_z z_j)$ with learned decay scales, read at $t^+_{j-1}$ and rate-normalised. The unconstrained model selects $\beta_j = \beta \odot e^{W_A c_j}$, $x_j = B_{:,k_j} g(m_j) \odot e^{W_B c^+_j}$ and $(\mu(c_j), C(c_j))$. Kukulkan selects

$$\beta_{j,n} = \beta_n\, e^{(W_A c_j)_n}, \qquad x_j = B_{:,k_j}\, g(m_j), \qquad C_{kn}(c_j) = W_{kn}\, s_{kn}(c_j)\, \beta_{j,n}, \qquad \mu = \mu(c_j), \tag{14}$$

with a matrix $W \geq 0$ and a scale $s_{kn}(c) \in [s_{\mathrm{lo}}, s_{\mathrm{hi}}]$ read from $c_j$ for each target type and channel: the gate $s = e^{-\mathrm{softplus}(Vc)} \in (0, 1]$, or the band $s = \exp(\log(1+\epsilon)\tanh(Vc)) \in [\frac{1}{1+\epsilon}, 1+\epsilon]$. With $s \equiv 1$ the model is the exact-$\rho$ special case, in which $\rho$ is a single number. On interval $k$ a past event $s$ contributes $W_{kn} s_{kn}(c_j)\, x_{s,n}\, \beta_{j,n} e^{-\int_{t_s}^{t} \beta_n}$ to $\lambda_k$, and $\beta_n(t) e^{-\int_{t_s}^{t} \beta_n}$ is a probability density in $t$ on every path.

**Proposition 5** (offspring band)**.** *Under equation 14, whatever the sequence of selected decay rates and scales, (i) the compensator on each interval is the decrease of the scaled state, $\int_{t_{j-1}}^{t_j} \lambda_k = \mu_k(c_j)\Delta_j + \sum_n W_{kn} s_{kn}(c_j)\, h^+_{j-1,n}\big(1 - e^{-\beta_{j,n}\Delta_j}\big)$, so the likelihood is exact; (ii) with marks independent of the history, the expected direct offspring of type $i$ from an event $s$ of type $k'$ lies in $\rho_{kk'}(s)\,[s_{\mathrm{lo}}, s_{\mathrm{hi}}]$ with $\rho_{kk'}(s) = \sum_n W_{kn} g_n(m_s) B_{nk'}$, a number fixed at the event, provided $\int_{t_s}^{\infty} \beta_n = \infty$ on every path. If a decay rate can fall to zero for good the count is smaller still, never larger. Averaged over marks, $\rho_{kk'} = \sum_n W_{kn}\mathbb{E}[g_n] B_{nk'}$ bounds the population offspring in the same way.*

The conditional branching ratio of an observed event $s$ is its offspring integrated along the observed sequence,

$$\rho_{\mathrm{obs}}(s) = \sum_n W_{kn} x_{s,n} \sum_{j>s} s_{kn}(c_j)\Big(e^{-\int_{t_s}^{t_{j-1}} \beta_n} - e^{-\int_{t_s}^{t_j} \beta_n}\Big) + \text{tail}, \tag{15}$$

the decrease of the scaled residual on each later interval, with the tail the mass left at the end of the window read under the last context.

**Recurrence and its unrolled form.** With $h(t) = \sum_{t_s<t} x_s \odot e^{-\beta(t-t_s)}$ and $x_s = B_{:,k_s} g(m_s)$, the excitation vector satisfies $h(t) = e^{-\beta(t-t_{j-1})} \odot h^+_{j-1}$ for $t \in (t_{j-1}, t_j]$ and $h^+_j = h^-_j + x_j$, which is equation 12. Iterating gives the unrolled sum of Section 4 with $S_j = \sum_{r\leq j} \beta\Delta_r$. The matrix $L_{js} = e^{-(S_j - S_s)}\mathbb{1}[s < k]$ is 1-semiseparable in the sense of Dao & Gu (2024), i.e. every submatrix below the diagonal has rank one, because $L_{js} = e^{-S_j} e^{S_s}$. Selected decay ($\beta_j$ depending on the context $c_j$ of events before $k$) keeps this form since it only changes the increments of $S$.

**Compensator.** For the linear output map, $\int_{t_{j-1}}^{t_j} \lambda_k(t)\, dt = \mu_k\Delta_j + \sum_n C_{kn} h^+_{j-1,n} \int_0^{\Delta_j} e^{-\beta_n u}\, du = \mu_k\Delta_j + \sum_n C_{kn} h^+_{j-1,n} \frac{1-e^{-\beta_n\Delta_j}}{\beta_n}$, with $(\mu, C) = (\mu(c_j), C(c_j))$ fixed on the interval. The tail after the last event to a horizon $T$ has the same form with $\Delta = T - t_L$. For the softplus output we integrate $f(\mu_2 + W(h^+_{j-1} \odot e^{-\beta u}))$ with a midpoint rule on cells whose edges are $\Delta_j \cdot \{0, 10^{-6}, \ldots, 1\}$, log-spaced, so that the resolution near $u = 0$ is $10^{-6}\Delta_j$; $M = 64$ cells for training, 256 for reporting, 1024 as a check.

**Branching ratio as DC gain.** The kernel from source $k'$ to target $k$ is $\phi_{kk'}(\tau) = \sum_n C_{kn} B_{nk'}\, \mathbb{E}[g_n(m)]\, e^{-\beta_n\tau}$ (marks independent of the past), so $\int_0^{\infty} \phi_{kk'} = \sum_n C_{kn}\mathbb{E}[g_n] B_{nk'}/\beta_n$. In continuous time the state obeys $\dot h = A_c h + Bx$ with $A_c = -\mathrm{diag}(\beta)$, whose steady-state gain from a constant input is $-CA_c^{-1}B$. With the mark expectation folded into $B$ this is equation 13. $\mathbb{E}[g_n]$ is taken over the training marks. Omitting it reads $\rho$ off by the factor by which the scale migrated into $g$ (1.66× in our recovery experiment). With context selection the

gains enter elementwise, $\rho_j = C(c_j)\,\mathrm{diag}(\mathbb{E}[g]\,e^{W_B c_j^+}/\beta_j)\,B$, and we report the distribution of its row sums over events. With regimes, $\rho_r = (C \odot e^{G_r})\,\mathrm{diag}(\mathbb{E}[g]/\beta)\,B$ at zero context.

**Proof of Proposition 5 (offspring band).** Fix a channel $n$ and an event $s$ with jump $x_{s,n} = B_{nk_s} g_n(m_s)(e^{W_B c_s^+})_n$. Let $\beta_n(t)$ be the decay rate and $s_{kn}(t)$ the scale that applies at $t$, both piecewise constant between events and selected by contexts that do not see the event they score. The residual of event $s$ at $t > t_s$ is $x_{s,n} e^{-\int_{t_s}^t \beta_n}$ and the output map that applies is $W_{kn} s_{kn}(t) \beta_n(t)$, so its contribution to $\lambda_k(t)$ is $W_{kn} x_{s,n}\, s_{kn}(t)\, \beta_n(t) e^{-\int_{t_s}^t \beta_n} = W_{kn} x_{s,n}\, s_{kn}(t)\, dF_n(t)$ with $F_n(t) = 1 - e^{-\int_{t_s}^t \beta_n}$, a distribution function on $(t_s, \infty)$ whenever $\int_{t_s}^\infty \beta_n = \infty$. (i) On an interval $(t_{j-1}, t_j]$ the scale and the decay rate are constant, so integrating and summing over past events gives $W_{kn} s_{kn}(c_j) h^+_{j-1,n}(1 - e^{-\beta_{j,n}\Delta_j})$, the decrease of the scaled state, plus $\mu_k(c_j)\Delta_j$ from the background rate. (ii) Over $(t_s, \infty)$ the integral is $W_{kn} x_{s,n} \int s_{kn}\, dF_n$, the mean of the scale under the probability measure $dF_n$, which lies in $W_{kn} x_{s,n}[s_{\mathrm{lo}}, s_{\mathrm{hi}}]$ on every path. Summing over channels and reading $k_s = k'$ gives $\rho_{kk'}(s)[s_{\mathrm{lo}}, s_{\mathrm{hi}}]$, and the expectation over marks and birth contexts gives the population statement. If $\int_{t_s}^\infty \beta_n < \infty$ with positive probability the measure has mass $F_n(\infty) < 1$ and the integral is smaller, never larger. With $s \equiv 1$ the interval collapses to the point $\rho_{kk'}(s)$, the exact-$\rho$ case. Nothing in the argument uses what the decay rates or scales are or how they were selected, only that they are predictable and constant within an interval, so it holds for context-selected decay rates, regime banks and their combination. The unconstrained model differs at one place: with $C_{kn}(c_j)$ unbounded the mean of the scale is unbounded, and the integral over the future depends on every later context, which is why the value at the birth context is only a stated value, not the offspring (Table 13).

**Proof of the exact-$\rho$ case (Proposition 5 with $s \equiv 1$).** Fix a channel $n$ and an event $s$ with jump $x_{s,n} = B_{nk_s} g_n(m_s)$. Let $\beta_n(t)$ be the decay rate that applies at $t$, piecewise constant between events and selected by contexts that do not see the event they score. The residual of event $s$ at $t > t_s$ is $x_{s,n} e^{-\int_{t_s}^t \beta_n}$ and the output map that applies is $W_{kn}\beta_n(t)$, so the contribution to $\lambda_k(t)$ is $W_{kn} x_{s,n}\, \beta_n(t) e^{-\int_{t_s}^t \beta_n} = -W_{kn} x_{s,n} \frac{d}{dt} e^{-\int_{t_s}^t \beta_n}$. (i) Integrating over an interval $(t_{j-1}, t_j]$ on which $\beta_n \equiv \beta_{j,n}$ and summing over past events gives $W_{kn} h^+_{j-1,n}(1 - e^{-\beta_{j,n}\Delta_j})$, the decrease of the state, plus $\mu_k \Delta_j$ from the background rate. (ii) Integrating over $(t_s, \infty)$ on a path gives $W_{kn} x_{s,n}\big(1 - e^{-\int_{t_s}^\infty \beta_n}\big)$. Taking the expectation over marks, summing over channels and reading $k_s = k'$ gives $\rho_{kk'}\big(1 - \mathbb{E}[e^{-\int_{t_s}^\infty \beta_n}]\big)$ channel by channel, which is $\rho_{kk'} = \sum_n W_{kn}\mathbb{E}[g_n] B_{nk'}$ when $\int_{t_s}^\infty \beta_n = \infty$ almost surely (the rate keeps running, which holds whenever $\beta_n$ is bounded below by a positive constant, as it is with a bounded selector $W_A c_j$), and at most that otherwise. Nothing in the argument uses what the decay rates are or how they were selected, only that they are predictable and fixed within an interval, so the invariance holds for context-selected decay rates, regime decay rates and their combination. The unconstrained model differs at one place: with $C_{kn}(c_j)$ free, the contribution of event $s$ on interval $k$ is $C_{kn}(c_j) x_{s,n} e^{-\int \beta_n}$ and its integral over the future depends on every later context, which is why the DC gain at the birth context is not the offspring (Section 6.2).

**Chunked algorithm.** With chunk length $Q$, write $a_j = -\beta\Delta_j$ (log-decay) and split the sequence into chunks. Inside chunk $c$ the contribution of earlier events in the same chunk is the $Q \times Q$ masked product $\exp(\mathrm{segsum}(a))\, x$. The state at the end of the chunk is $\sum_{s \in c} e^{(A^{\mathrm{cum}}_{\mathrm{end}} - A^{\mathrm{cum}}_s)} x_s$. The states are passed across chunks by a recurrence at chunk granularity, and the incoming state contributes $h_{\mathrm{in}}\, e^{A^{\mathrm{cum}}_j}$ inside the chunk. This is Listing 1 of Dao & Gu (2024) specialised to scalar decay and $B = C = 1$. Cost $O(LQ)$ time and memory. On an RTX 5090 with $N = 8$ states, forward and backward take 1.4 ms for $L = 16{,}384$ with 412 MiB, against ≈100 GiB the quadratic form would need.

**Why the output map is not selected by the current event.** If $C$ depended on the token at position $j$ as Mamba's $C_t$ does, the intensity $\lambda(t_j^-)$ used to score event $j$ would be a function of event $j$'s type and mark, which is the leak of Section 3 by construction. Selection of the decay and output map for interval $j$ from events before $j$, and of the jump at event $j$ from events $\le j$, affects only intensities after the information is available and is therefore admissible.

**The regime filter.** The regime is a continuous-time Markov chain with generator $Q$ between events, allowed to jump at events with row-stochastic probabilities $T_j$ selected by the context. Let $\alpha_j \in \mathbb{R}^K_{\geq 0}$ be the unnormalised belief after event $j$. Over the interval the chain and the survival probability evolve together, at the event Bayes' rule multiplies by the intensity of the observed type in each regime, then the jump applies:

$$\alpha_j = \alpha_{j-1}\,\Phi_j\,\mathrm{diag}\big(\lambda_{k_j}(t_j^- \mid \cdot)\big)\,T_j, \qquad \Phi_j = \prod_{i=1}^{m} \exp\Big(\big(Q - \mathrm{diag}\,\bar{\lambda}_i\big)\,\ell_i\Big), \tag{16}$$

where the interval is cut into $m$ log-spaced sub-intervals of lengths $\ell_i$ on which $\bar{\lambda}_i$ is the exact integrated total rate per regime divided by $\ell_i$. The predictive intensity is the belief average and the marginal likelihood is $\log \alpha_L \mathbf{1}$ plus the survival to the end of the window.

**The regime filter is exact.** Let $r(t)$ be a continuous-time Markov chain with generator $Q$ ($Q_{rr'} \geq 0$ for $r \neq r'$, rows summing to zero) and let events arrive with intensity $\lambda_k(t \mid r(t))$ given the shared state $h(t)$, which is a deterministic function of the past events. Define $\alpha_j(r) = p\big(r(t_j^+) = r,\ (t_i, k_i)_{i \leq j}\big)$. Between events, the joint probability of being in regime $r$ at time $t$ and having seen no event since $t_{j-1}$, $u_r(t)$, satisfies the Kolmogorov forward equation with killing,

$$\dot{u} = u\,\big(Q - \mathrm{diag}\,\bar{\lambda}(t)\big), \qquad \bar{\lambda}_r(t) = \sum_k \lambda_k(t \mid r),$$

whose solution over a sub-interval on which $\bar{\lambda}$ is replaced by its exact average $\bar{\lambda}_i = \frac{1}{\ell_i}\int \bar{\lambda}$ is $u(t+\ell_i) = u(t)\exp\big((Q - \mathrm{diag}\,\bar{\lambda}_i)\ell_i\big)$. The average is exact because the state between events is $h^+_{j-1} \odot e^{-\beta u}$ in closed form, so $\int \bar{\lambda}$ is the compensator above per regime. The only approximation is that the rate is frozen at its sub-interval mean while the chain moves, which vanishes as the sub-intervals shrink (we use 16 log-spaced ones for training and 64 for reported numbers). Composing the sub-intervals gives $\Phi_j$ of equation 16. At the event the density of an arrival of type $k_j$ in regime $r$ is $\lambda_{k_j}(t_j^- \mid r)$, and the selected jump $T_j$ moves the regime after the event. Hence $\alpha_j = \alpha_{j-1}\Phi_j\,\mathrm{diag}(\lambda_{k_j})\,T_j$, $p(\text{events}) = \alpha_L\mathbf{1}$ times the survival from $t_L$ to the horizon, and the predictive belief entering event $k$ is $\pi_{j-1}\Phi_j$ normalised: everything the model reports conditions on events before $k$ only.

The hidden-Markov-model (HMM) style step $\alpha_{j-1}\,e^{Q\Delta_j}\,\mathrm{diag}\big(e^{-\int \bar{\lambda}_r}\big)$, transition first and then survival under the regime of arrival, is not the solution of the forward equation: it lets a sequence switch into a low-rate regime at the start of a long interval and pay that regime's survival for the whole interval, and the resulting next-interval density integrates to more than one. Our first implementation used it and beat the oracle on simulated data, which is how we noticed. The check that catches this, and that we run as a standing test, is to integrate the model's next-interval density $p(\Delta \mid \text{past}) = \sum_r \big[\pi\,\Phi(\Delta)\big]_r \bar{\lambda}_r(\Delta)$ numerically on a fine log grid and verify that it and the survival sum to one for fast switching and distinct regimes (mass $1.00004$ on our checks, above one for the HMM step).

**Log-scale scan.** The step matrices $M_j = \Phi_j\,\mathrm{diag}(\lambda_{k_j})\,T_j$ have entries that range over $e^{\pm 100}$ on real data (a lunch break in a window of exchange orders is an interval with $\int \bar{\lambda}$ in the hundreds), so their products under- and overflow float32 and the normalised belief loses its gradient. We factor a scalar out of every step, $M_j = s_j \tilde{M}_j$ with $\log s_j = -\min_r \int \bar{\lambda}_r + \log\max_r \lambda_{k_j}(\cdot \mid r)$, take the matrix exponential of the shifted generator in float64, and scan the $\tilde{M}_j$ by a log-depth prefix product inside chunks of 64 in which every partial product is divided by its largest entry with the logarithm of that entry carried alongside, the belief being renormalised at chunk boundaries. The per-event term is $\log(\alpha_{j-1}\tilde{P}_j\mathbf{1}) - \log(\alpha_{j-1}\tilde{P}_{j-1}\mathbf{1}) + \log s_j + \log c_j - \log c_{j-1}$ with $\tilde{P}$ the rescaled products and $c$ their scales, and it is $O(1)$ in every factor. The rescaling matters in the backward pass more than in the forward: a fit that has drifted to a regime with an intensity of $10^{10}$ makes $\lambda/\lambda_{\max}$ of order $10^{-9}$ for the regimes that carry the belief, sixty-four of those underflow even float64, and the gradient $1/(\alpha\mathbf{1})$ of the unscaled product overflows float32 on its way back into the parameters (we found this from a saved batch whose loss was finite and whose gradient was not). The unit test fits a two-regime model to a sequence with an interval worth 700 nats and checks the likelihood against the $K = 1$ closed form and the gradient for finiteness.

**Regimes, implicit and explicit.** An attention model has no regimes as objects: whatever it uses is a direction in its hidden representation, learned jointly with the MLP, with no number to choose and nothing to read out. The unconstrained model has both versions. The continuous one is the context output map $C(c_j), \mu(c_j)$: the weighting of the past changes with the recent events, no $K$ is specified, and it carries most of the likelihood. It is an implicit regime, as unreadable as attention's, and Section 6.2 shows that its branching ratio is a proxy. The discrete one is the belief of Section 5.2: $K$ explicit regimes, each with its own kernel and branching matrix, and a posterior computed by Bayes; $K$ must be chosen, as the number of components of a mixture must, and in exchange the model returns $\rho_r$, a regime path and the confounding result. The no-switch control of Table 9 shows that surplus regimes are invented by cold fits and exposed only by a held-out set, so $K$ is a claim to be gated, not a free dial. A nonparametric prior on $K$ exists (infinite hidden Markov models) but not with an exact parallel filter, and we leave it there.

**The regime filter as a second selective scan.** Equation equation 16 is itself a linear recurrence, $\alpha_j = \alpha_{j-1} M_j$, whose $K \times K$ transition $M_k$ is selected by the data: by the elapsed time and the state through $\Phi_j$, by the observed type through $\lambda_{k_j}$, and by the context through $T_j$. It is computed by the same chunked scan as the excitation state, with matrix products in place of scalar ones (prefix products inside chunks of 64 events by a log-depth scan, the belief carried across chunks with its log-scale), so training cost is $O(LK^3)$ in parallel and inference is $O(K^2)$ per event; $K \leq 4$ in all our experiments. The normalised belief is the one place where a softmax appears in Kukulkan: with no switching, $\pi_{j,r} \propto \pi_{0,r} \exp \sum_{j \leq k} \ell_i(r)$ is a softmax over the accumulated log-likelihood of each hypothesis, an attention over regimes whose weights are set by Bayes rather than by a learned score, and $Q$ and $T$ are its forgetting. The regime posterior, the jump probabilities and $\rho_r$ are what the model explains with.

**Confounding.** A latent state that raises $\mu$ for all types is a common cause of clustered events. A single-regime Hawkes fit has only one way to explain the clustering, excitation, and reports a branching ratio near one: the apparent criticality of Filimonov & Sornette (2015), and the same confounder that makes Granger-type estimates of a Hawkes kernel unsafe (Jin & Huang, 2026). The belief layer gives the clustering a second explanation and lets the likelihood choose. Appendix C shows the two fits on data with a known switching $\mu$.

### A.2 Proofs for Section 4

*Proof of Proposition 1.* Between events $\dot{h} = -\beta \odot h$, so $h(t_j^-) = e^{-\beta \Delta_j} \odot h(t_{j-1}^+) = e^{-\beta \Delta_j} \odot \big(h(t_{j-1}^-) + x_{j-1}\big)$, which is the recurrence. Unrolling it from $h(t_1^-) = 0$ gives $h(t_j^-) = \sum_{s<j} \big(\prod_{r=s+1}^{j} e^{-\beta \Delta_r}\big) \odot x_s = \sum_{s<j} L_{js} \odot x_s$, and the intensity follows by applying $C$. Because $L_{js,n} = e^{-\beta_n t_j}\, e^{\beta_n t_s}$ is a product of a factor of $j$ and a factor of $s$ on $s < j$, the lower-triangular matrix $L_{\cdot\cdot,n}$ is 1-semiseparable, and the product of a 1-semiseparable mask with a rank-one content matrix is the structure whose chunked evaluation is the state space duality algorithm of Dao & Gu (2024). The costs quoted are theirs. □

*Proof of Proposition 2.* (i) On the interval the rate is $\lambda_k(t_{j-1} + \tau) = \mu_k + \sum_n C_{kn} h_n(t_{j-1}^+) e^{-\beta_n \tau}$, and $\int_0^{\Delta_j} e^{-\beta_n \tau} d\tau = (1 - e^{-\beta_n \Delta_j})/\beta_n$. Since $h_n(t_j^-) = h_n(t_{j-1}^+) e^{-\beta_n \Delta_j}$, the integral of the $n$-th term is $C_{kn}\big(h_n(t_{j-1}^+) - h_n(t_j^-)\big)/\beta_n$. (ii) With $f_k(z, \tau) = a\tau + b$ and $\sigma(x) = \log(1 + e^x)$, substituting $v = -e^{a\tau + b}$ gives $\int \log(1 + e^{a\tau+b})\, d\tau = -\frac{1}{a} \int \frac{\log(1-v)}{v}\, dv = -\frac{1}{a} \mathrm{Li}_2(-e^{a\tau+b})$, the dilogarithm, which is not an elementary function. □

*Proof of Proposition 4.* An event of type $k'$ with mark $m$ adds $B_{nk'} g(m) e^{-\beta_n \tau}$ to channel $n$ at lag $\tau$, read with weight $C_{kn}$, and $\int_0^\infty e^{-\beta_n \tau} d\tau = 1/\beta_n$. □

*Proof of Proposition 3.* With $r = e^{-\beta \Delta/(M-1)}$ the grid values are $a\, r^{\,i-1}$, so $\sum_{i=1}^{M} e^{-\beta u_i} = \sum_{i=0}^{M-1} r^i = (1 - r^M)/(1 - r)$, which is the stated form. As $\beta\Delta \to 0$, $r \to 1$ and the sum tends to $M$, so $\widehat{\Lambda} \to a\Delta$, while $\Lambda = a(1 - e^{-\beta\Delta})/\beta \to a\Delta$. As $\beta\Delta \to \infty$, $r \to 0$ and the sum

tends to 1, so $\widehat{\Lambda} \to a\Delta/M$, while $\Lambda \to a/\beta$. The ratio tends to $\beta\Delta/M$. For $u_i$ independent and uniform on $(0, \Delta)$, $\mathbb{E}\, e^{-\beta u_i} = (1 - e^{-\beta\Delta})/(\beta\Delta)$, so $\mathbb{E}\,\widehat{\Lambda} = a\Delta \cdot (1 - e^{-\beta\Delta})/(\beta\Delta) = \Lambda$. □

**The fixed-grid objective.** Under a fixed grid the training objective is $\widehat{\ell} = \sum_j \log \lambda(t_j^-) - \sum_j \frac{\Delta_j}{M} \sum_{i=1}^{M} \lambda(u_{j,i})$. It depends on the rate only at the event times and at the grid points. Its derivative with respect to $\lambda(t_j^-)$ is $1/\lambda(t_j^-)$, its derivative with respect to $\lambda(u_{j,i})$ is $-\Delta_j/M$, and its derivative with respect to the rate at any other time is zero. So the objective pushes the rate down at the grid points and up at the events and is indifferent to the rate everywhere else, whereas the exact log-likelihood counts the area under the rate everywhere. For the construction in Section 4, let two channels with decay rates $\beta_2 > \beta_1$ be read through a softplus as $\lambda(\tau) = \mathrm{softplus}\big(a_1 e^{-\beta_1 \tau} - a_2 e^{-\beta_2 \tau} + b\big)$ on an interval. At $\tau = 0$ the argument is $a_1 - a_2 + b$, which is as negative as desired for $a_2$ large, so $\lambda(u_1)$ is as small as desired. For $\tau \gg 1/\beta_2$ the second term has vanished and $\lambda(\tau) \approx \mathrm{softplus}(a_1 e^{-\beta_1 \tau} + b)$, a bump of height about $a_1 + b$ that decays at the rate $\beta_1$. Choosing $\beta_2 > \beta_1 \gg (M-1)/\Delta$ places the whole bump inside $(u_1, u_2)$, where the grid has no point, so it adds to $\Lambda$ and not to $\widehat{\Lambda}$. The selective model chooses $a_1, a_2, b, \beta_1, \beta_2$ from its input.

## B EXPERIMENTAL DETAILS

**Datasets.** Table 4 lists the ten sets. The nine public ones are the EasyTPP JSON releases (Xue et al., 2024) with their standard splits. Sequences with fewer than two events are dropped. Times are rescaled so that the mean inter-event interval is 1 for optimisation and likelihoods are reported in the original unit by adding $\log(\text{scale})$ per event. Volcano has a single type, so its accuracy is 1 by construction and only its likelihood is informative. Financial (EasyTPP) is 200 sequences of 3,319 events with a median interval of $5 \times 10^{-5}$ and no exact ties. The order-flow set is built from the tick-by-tick trade feed of one stock exchange: regular session only, type from the aggressor flag, volume from the difference of the cumulative volume field, prints with the same stamp and side merged into one order with the summed volume, and the orders of a stamp placed uniformly at random inside the feed batch that precedes it (200 ms or the interval to the previous stamp, whichever is shorter), in the recorded order (Section 3). Six instruments were selected by trade count on the first day. Ten consecutive trading days. Days 1–7 train, day 8 dev, days 9–10 test. Windows of 200 orders.

Table 4: Datasets. Sequences per split, events in all splits, mean and maximum length, mean interval in the set's unit, and the share of exactly tied consecutive stamps in the training split.

| set | $D$ | train | dev | test | events | len (mean) | len (max) | mean interval | ties |
|---|---|---|---|---|---|---|---|---|---|
| Taxi | 10 | 1,400 | 200 | 400 | 74k | 37.0 | 38 | 0.225 | 0% |
| StackOverflow | 22 | 1,401 | 401 | 401 | 143k | 64.6 | 101 | 0.889 | 0.5% |
| Retweet | 3 | 20,000 | 2,000 | 2,000 | 2.61M | 108.8 | 264 | 2,596 | 4.0% |
| Amazon | 16 | 6,454 | 922 | 1,851 | 413k | 44.7 | 94 | 0.521 | 0% |
| Taobao | 17 | 1,300 | 200 | 500 | 113k | 56.5 | 64 | 0.046 | 0% |
| Volcano | 1 | 307 | 37 | 144 | 9.0k | 18.9 | 244 | 817 | 4.9% |
| Earthquake | 7 | 3,000 | 400 | 896 | 70.7k | 16.5 | 18 | 1.30 | 0% |
| MIMIC-II | 75 | 454 | 66 | 130 | 2.4k | 3.7 | 33 | 0.648 | 0% |
| Financial | 2 | 90 | 10 | 100 | 415k | 3,319 | 3,319 | 0.0013 | 0% |
| orders (200) | 2 | 10,263 | 1,451 | 2,660 | 2.87M | 200 | 200 | 0.465 s | 1.3% |

**Simulated spike process.** The process behind Figure 1 is a two-type exponential Hawkes process with $\mu_k = 0.3$, $\beta_k = 40$ for both types and offspring matrix $\rho$ with $0.4$ on the diagonal and $0.1$ off it (row sums $0.5$), so that every event raises the rate by $a_{kk'} = \rho_{kk'}\beta_{k'}$, 16 for its own type and 4 for the other, and the excess decays to a tenth within $0.06$ time units, the median interval. Sequences are simulated by Ogata's thinning on $[0, 85]$, about 103 events each; 1,500 train, 200 development and 400 test sequences. The true parameters give a test negative log-likelihood of $0.409$ nats per event, the floor that no fitted model can beat under exact integration. An exponential-kernel fit (the exact-$\rho$ model with the matrix output, $N = 8$) reaches it. The re-implemented THP, SAHP and MHP are fitted with the shared-base recipe below. S2P2 with its paper configuration for 150 epochs at batch

64, once against the library's default 10-point grid and once against 10 uniformly random points per interval (Appendix D.1, Table 14).

**Kukulkan.** Width $N \in \{K, 2K, 4K\}$ chosen on dev NLL ($4D$ only on Retweet, Financial and the trade set; $\{D, 2D\}$ on MIMIC-II). Decay rates initialised log-uniformly over $[0.1, 10]$ mean intervals; $\mu$ at half the empirical rate per type; $C$ and $B$ at $0.3/\sqrt{N}$ so the initial $\rho$ is subcritical. Mark scale $g(m) = \exp(w \log m + c)$ with $w = 1, c = 0$ (marks only on the trade set). Context: embedding width 16, depthwise causal convolution of width 4, memory with four decay scales log-spaced over $[10^{-3}, 1]$ mean intervals, all gains initialised to 1 (zero weights), log-gains clipped to $[-4, 4]$. Exact-$\rho$ rows (Kukulkan, $s \equiv 1$): the decay-rate selector $W_A$ reads the memory context per channel, the output matrix $W$ is fixed (initialised as $C$ above) and multiplied by the decay rate that applies on the interval, the jump is the table $B_{:,y}g(m)$ (on the public sets with the same context gain at birth as the unconstrained model, which Proposition 5 permits), and $\mu$ is either fixed ("decay rate only") or $\mu(c_j)$ from the same context. Everything else, including width selection and training, is as for the unconstrained model. Unconstrained rows (Kukulkan, unconstrained): $\beta_j$, $x_j$ and $(\mu, C)(c_j)$ all selected. Kukulkan (gate) and Kukulkan (band $1 + \epsilon$): the output map of equation 14, $C = Ws(c_j)\beta_j$, with the scale read from the same memory context by one linear map per target type and channel ($16DN$ weights), initialised at $s = 1$ (band. Zero weights) or $s = 0.95$ (gate. Bias $-3$). Every band and gate fit starts from the fitted exact-$\rho$ model of the same seed and width (its decay rates, $\mu$, matrix and context copied, the scale at its initial value) and is trained with the same recipe, because a cold start of the same model lands $0.01$ nats behind the exact-$\rho$ model's optimum on the orders at the narrowest band (Appendix C). The exact-$\rho$ rows (Kukulkan, $s \equiv 1$) are the cold fits. Regimes: cold fits use $K = 3$ on the matrix output and $K = 2$ on top of the context output map, log-gains $G_r, g_r$ initialised at $\mathcal{N}(0, 0.1^2)$ with the $\mu$ gains at $1\times$ to $2\times$ and the dwell at 100 events. Warm fits use $K = 3$ in both cases, start from the fitted $K = 1$ model of the same seed and width (its output map as the base), draw $G_r$ at $\mathcal{N}(0, 0.3^2)$, $g_r = 0$, and initialise the dwell at 300 events. Jump probabilities $T$ initialised to the identity (bias 4 on the diagonal). 16 log-spaced sub-intervals per interval for training and 64 for every reported number (checked at 256). Adam, learning rate 0.03, batch 64 (128 on the trade set, 8 on Financial), gradient clipping 5, up to 200 epochs with patience 15 on dev NLL (60/8 on Retweet, Financial and the trade set). Chunk length 64 for both scans. Softplus compensators: log-spaced quadrature, 64 cells in training, 256 at evaluation, 1024 as a check (also for the baselines). Kukulkan rows are three seeds. Baselines are one seed.

**Baselines (shared base).** Type embedding plus THP's sinusoidal encoding of absolute time. Two post-LN causal transformer layers (4 heads, width 64, feed-forward width 128, dropout 0.1) or two Mamba event blocks (expansion 2, 16 states, depthwise convolution 4). Adam $10^{-3}$, batch 64, clipping 5, up to 100 epochs, patience 10. THP head: $\mathrm{softplus}(\alpha_i \tau + w_i \cdot h + b_i)$ ***without*** the division of $\tau$ by the absolute timestamp used in the official code. SAHP head: $\mathrm{softplus}(\mu + (\eta - \mu)e^{-\gamma\tau})$ with $\mu, \eta$ ***linear*** in $h$ and $\gamma = \mathrm{softplus}(\cdot)$. The paper's GELU on $\mu, \eta$ bounds the intensity below by $\mathrm{softplus}(-0.17) \approx 0.61$ in units where the mean interval is 1 and could not fit Taxi (NLL 4.8). SB-THP: per-head learnable $\alpha, \gamma$, $g(m) = \mathrm{softplus}(\mathrm{MLP}(m))$ with one hidden layer of 16, sigmoid gate from the query. MHP: $\Delta_i$ equal to the inter-event interval, zero-order hold, S4D-real initialisation of $A$, THP head, no marks. Its scan is computed with our chunked algorithm on the flattened (channel, state) axis, verified against the sequential loop to $10^{-7}$. Official THP: repository defaults (width 64, 4 heads, 4 layers), 30 epochs, best test epoch as in its protocol. The leak-free variant patches only the likelihood function. FIM-PP: released checkpoint, 400 training sequences (truncated to 100 events) as context, encoded once. Intensities requested on a stratified grid and at the event times. The model's causality (history strictly before each evaluation time) makes it leak-free. It supports at most 22 types, so it has no MIMIC-II number.

**The grid rule.** EasyTPP estimates the compensator on each interval by the mean of the intensity at $G$ equally spaced points from the previous event to the current one, both ends included, with $G = 10$ for S2P2 and 20 for the other models it hosts. The same rule is used in training, in model selection and in the reported number. The library's paper describes this term as the Monte Carlo approximation of Mei & Eisner (2017) with sampled times, which is unbiased (Xue et al., 2024). In the released code (version 0.3.0, `make_dtime_loss_samples`) the points are placed by `linspace`, and the option named after Monte Carlo samples only replaces the trapezoid weights

by equal weights, so the estimator is deterministic in either setting. Our re-implementation of the rule reproduces the library's reported numbers to the fourth decimal.

**S2P2.** Run from the authors' code inside EasyTPP 0.3.0 with the paper's configuration (four layers, hidden size 128, $P = 16$, dropout 0.1, GELU, learning rate $10^{-2}$, batch 256, 300 epochs with the library's early stopping. 33–37k parameters), seeds 2019 and 1, and the library's two training options for the compensator: its default fixed grid of $G = 10$ equally spaced points per interval ("grid") and $G$ uniformly random points per interval ("random"). For memory we used batch 64 on StackOverflow and Amazon and batch 128 on the exchange orders (100 epochs there, 150 on StackOverflow and Amazon). No other setting was changed and nothing was tuned. Every reported S2P2 number is the same checkpoint re-scored under our convention: the event term unchanged, the compensator by 1024 cells log-spaced toward the previous event, checked against 256 cells and against 4000 uniform midpoint cells (which agree to $10^{-4}$ except on Amazon, Section 3). Our re-implementation of the library's grid rule reproduces its reported number to the fourth decimal, and a second generic scorer (the model's own loss with a patched sampler) gives the same values. The exchange orders were exported to EasyTPP's format with relative times and the same windows, splits and tie handling as for every other model. Wall-clock training on one RTX 5090: S2P2 4,278 and 4,359 s on the exchange orders (100 epochs, batch 128) and 536 s on Taxi, against 55 s (33 epochs) and 11 s for the exact-$\rho$ model, 59 s and 10 s for Kukulkan, unconstrained and 53 s and 12 s for the softplus Kukulkan with early stopping. The code paths differ (EasyTPP's training loop against ours), so this is an order of magnitude, not a benchmark.

**Classical kernels.** Power law $\phi_{kk'}(\tau) = a_{ij}(\tau + c_j)^{-(1+p_j)}$. Histogram kernel with 8 log-spaced bins on $[10^{-2}, 20]$ mean intervals. Both fitted by exact MLE in the quadratic form (closed-form compensators), with source-type indexing so that the memory is $O(L^2 D)$ rather than $O(L^2 D^2)$. The exponential kernel is Kukulkan with the matrix output, $K = 1$ and $N = K$.

**The THP likelihood leak, and how we fixed it.** While re-implementing the baselines we found that the reference implementation of THP scores each event from a state that has already seen it. The issue was raised against the code in 2021 and is footnoted by Yang et al. (2022). Patching the one function moves THP's test log-likelihood by $0.65$, $1.85$ and $2.59$ nats on Taxi, StackOverflow and Retweet, and every THP-family number in this paper is from the patched likelihood. In `Utils.log_likelihood` of the reference implementation, `all_hid = model.linear(data)` is evaluated at every position $j$ and multiplied by the one-hot type of event $j$. `get_subsequent_mask` is `triu(..., diagonal=1)`, so position $j$ attends to itself. In `compute_integral_unbiased` the hidden state used for the interval $(t_j, t_{j+1}]$ is `model.linear(data)[:, 1:]`, i.e. the state at $j{+}1$. The leak-free version uses the state at $j{-}1$ for the event term of event $j$ and for the integral over $(t_{j-1}, t_j]$, and divides by the number of scored events.

# C Ablation study

## C.1 Kukulkan variants

Kukulkan in the main text is the gated model. This appendix compares it with the other members of the same family, all trained from the same starting point: the exact-$\rho$ model ($s \equiv 1$), bands that confine the scale to a symmetric interval around one, the gate with its bound fixed, the unconstrained output map of Section 3, and a softplus output of the same state, which has no branching ratio.

**The family.** The output weight of a channel is the fixed weight times the decay rate times a scale $s(c)$ read from the context. The gate confines the scale to $(0, 1]$, so the context may suppress an event's influence but never amplify it. The band confines it to $[1/(1+\epsilon), 1+\epsilon]$, a symmetric interval around one. With the scale fixed at one the model is the exact-$\rho$ model. If the upper bound must equal the exact-$\rho$ model's $\rho$, it can be made to. The exact-$\rho$ model is fitted first, its fixed weights are frozen, and the gate, the decay rates, $\mu$ and the context are trained further from that optimum, so that the upper bound is the exact-$\rho$ model's $\rho$ by construction. The cost is likelihood, a fraction of the gate's gain on order flow (Figure 4), and where a context-selected $\mu$ has absorbed the excitation, as on Taxi with $0.07$ offspring per event, the gate has nothing left to scale. The upper bound is only as

meaningful as the first stage. Section 6.2 fixes it with a fixed $\mu$, where the fixed weight is identified and sub-critical.

**What the context gains, and how much of the upper bound it uses.** Figure 4 traces the held-out likelihood along the model class, each member trained further from the exact-$\rho$ model's optimum with its training procedure (Appendix B). On exchange order flow the exact-$\rho$ model scores $0.564$ nats per event and the unconstrained model $0.548$. Bands of width $1.25$, $2$ and $3$ recover $0.001$, $0.009$ and $0.013$ of that difference, the gate recovers $0.021$, and the gate with frozen fixed weights $0.003$. On Taxi, $-0.269$ against $-0.327$, the bands recover $0.03$ and the gate $0.05$. On Amazon and Earthquake the gate closes the difference and the bands recover a third to a half of it. What the fitted models do with their freedom explains the pattern. For each scored interval we measure the excitation read with the scale divided by the excitation the unscaled output map would give. On exchange order flow the gate's median is $0.58$ and its fifth percentile $0.25$, while the bands stay near one, with medians $0.89$, $1.04$ and $1.32$. On Taxi the gate's median is $0.18$: the context all but switches the excitation off, which a model of alternating pickups and drop-offs requires. The context improves the likelihood by turning excitation down, and a band whose lower limit stays above zero reaches less of that improvement than a gate whose lower limit is zero. The one direction the data ask for is the direction a one-sided bound permits, which is why the gate is the model written Kukulkan in the main text.

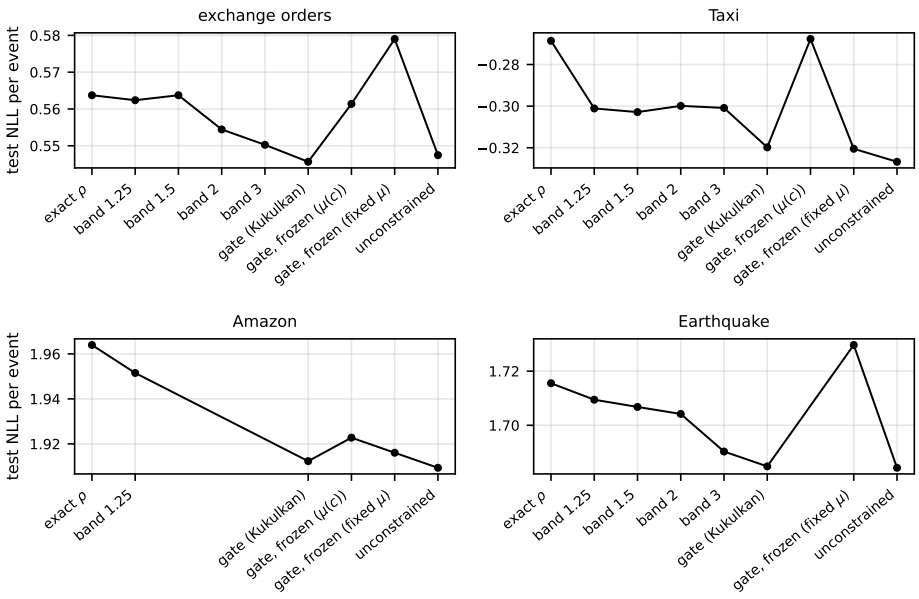



Figure 4: Held-out negative log-likelihood (NLL) per event along the model class, seed 0, every fit warm-started from the exact-$\rho$ optimum: the exact-$\rho$ model ($s \equiv 1$, left), bands of width $1.25$ to $3$, the gate (Kukulkan), the gate with the fixed weight frozen at the exact-$\rho$ model's value or at the fixed-$\mu$ value, and the unconstrained model (right).

**Public datasets.** Table 5 is the full version of Table 2. Under exact scoring the gated model is within $0.02$ nats of the unconstrained model, the seed spread, on seven of the nine public datasets. It is better than the unconstrained model on StackOverflow, Taobao and Volcano, worse by $0.06$ on Retweet, where the unconstrained model's context-dependent input carries what the output map cannot, and worse by $0.2$ on MIMIC-II, whose 2,400 events no model in this class learns reliably from a cold start. It is better than the exact-$\rho$ model on every dataset, by $0.004$ on Financial and by $0.04$ on Taxi and Amazon, so the bound of Proposition 5 costs nothing in likelihood against the unconstrained model. The fixed-bound row, trained further from the exact-$\rho$ model with a fixed $\mu$, is within $0.01$ of the unconstrained model on Taxi, Amazon and Volcano, better than the gate on Retweet ($6.385$), and worse than the exact-$\rho$ model on StackOverflow and Earthquake, where the

fixed weights it inherits from the fixed-$\mu$ fit are far from the optimum with a context-selected $\mu$. The rows with a linear output map are $0.1$–$0.6$ nats behind the softplus row where a type must be switched off entirely, on Taxi and Amazon: the gate can switch a channel off but not a type.

Table 5: Public datasets, all variants: NLL per event (lower is better) under the protocol of Section 6. Rows above the line have no branching ratio; below it the exact-$\rho$ model reports $\rho$ exactly, the gate (Kukulkan) an upper bound and a conditional offspring per event, the fixed-bound row inherits its upper bound from the fixed-$\mu$ fit, and the unconstrained row states a value that is not the offspring. — = not run.

| model | Taxi | StackOv. | Retweet | Amazon | Taobao | Volcano | Earthq. | MIMIC-II | Financial |
|---|---|---|---|---|---|---|---|---|---|
| Hawkes, power law | 0.302 | 2.260 | 6.553 | 2.183 | $-1.254$ | 6.430 | 1.840 | 1.667 | $-6.723$ |
| Hawkes, nonparametric | 0.221 | 2.314 | 6.580 | 1.758 | $-1.285$ | 6.444 | 1.839 | 1.327 | $-6.822$ |
| THP | $-0.341$ | 2.323 | 6.694 | 2.095 | $-0.992$ | 6.868 | 1.904 | 1.268 | $-5.767$ |
| MHP | $-0.321$ | 2.423 | 6.628 | 2.139 | $-0.938$ | 6.854 | 1.869 | 1.538 | $-5.983$ |
| SAHP | $-0.486$ | 2.217 | 6.356 | 1.402 | $-1.239$ | 6.568 | 1.770 | 1.381 | $-6.665$ |
| S2P2 | $-0.522$ | 2.171 | $\approx 5.11$ | $-0.74$ | $-1.351$ | 6.052 | 1.712 | 1.393 | $-7.376$ |
| Kukulkan, softplus output | $-0.509$ | 2.191 | 6.354 | 0.828 | $-1.350$ | 6.459 | 1.680 | 1.725 | $-6.771$ |
| Hawkes, exponential | 0.308 | 2.291 | 6.548 | 2.109 | $-1.252$ | 6.464 | 1.830 | 1.587 | $-6.776$ |
| Kukulkan, exact $\rho$ | $-0.268$ | 2.176 | 6.442 | 1.964 | $-1.269$ | 6.321 | 1.709 | 2.173 | $-6.783$ |
| Kukulkan (gate) | $-0.305$ | 2.169 | 6.433 | 1.919 | $-1.277$ | 6.311 | 1.691 | 2.128 | $-6.780$ |
| Kukulkan, bound fixed | $-0.320$ | 2.235 | 6.385 | 1.916 | $-1.259$ | 6.313 | 1.730 | 1.859 | $-6.785$ |
| Kukulkan, unconstrained | $-0.327$ | 2.184 | 6.371 | 1.906 | $-1.266$ | 6.317 | 1.688 | 1.923 | $-6.787$ |

**Exchange order flow.** Table 6 completes Table 3. The context improves on the classical kernel by $0.11$ nats in three steps, the decay rates $0.045$, the context selection of $\mu$ $0.042$ and the gate $0.021$, and the gate's step is the whole of the unconstrained model's advantage, whose score is $0.547$, with a bound in place of a stated value. Whether $\rho$ is identified depends on $\mu$, and the upper bound can be fixed to the identified value at a price. With a fixed $\mu$ the three seeds of the exact-$\rho$ model agree, at $0.86$–$0.91$ for buy orders and $0.75$–$0.78$ for sell orders, sub-critical on windows of 200 orders, consistent with Hardiman et al. (2013) at this window length. Training the gate, the decay rates and the context-selected $\mu$ further from that fit with its fixed weights frozen gives a model in which every order has at most $0.91$ or $0.76$ direct offspring on every path, with context, at $0.579$ nats: $0.027$ better than the fixed-$\mu$ model from which it inherits the bound and $0.036$ worse than the jointly fitted gate. Its conditional offspring average $0.92$, its simulated offspring $0.93$, and it uses $0.97$ of its upper bound on average, so under a fixed sub-critical bound the context seldom gates and mostly moves $\mu$ and the decay rates. With a context-selected $\mu$ the seeds of the exact-$\rho$ model sit on the ridge of Section 3 (Table 19), and a bound fixed there inherits it. In the free run of Section 6.2 the unconstrained model diverged on seven of sixteen paths, and the gate with its bound fixed below one generates $3{,}300$ orders per hour against the data's $7{,}800$ and is under-dispersed.

Table 6: Exchange orders (windows of 200), all variants: NLL per event, direction accuracy, and the branching-ratio row sums the fit reports for (buy, sell), seed by seed where they differ (three seeds; the fixed-bound rows one seed). "stated": the stated value is not the offspring (Table 13); for the gate rows the stated value is the upper bound and "conditional" the mean offspring of the observed orders along the observed path, equation 15.

| model | params | NLL | acc | $\hat{\rho}$ row sums (buy, sell) |
|---|---|---|---|---|
| Hawkes, exponential | 58 | 0.652 | 0.652 | 0.92, 0.74 |
| Kukulkan, exact $\rho$, fixed $\mu$ | 322 | 0.606 | 0.663 | 0.91, 0.76 \| 0.88, 0.78 \| 0.86, 0.75 |
| Kukulkan, exact $\rho$, $\mu(c)$ | 2.2k | 0.564 | 0.665 | 1.02, 1.06 \| 1.08, 1.01 \| 0.72, 0.74 |
| Kukulkan (gate) | 2.5k | 0.543 | 0.668 | bound 2.5–4.1 per seed; conditional 1.06 |
| Kukulkan, bound fixed (fixed $\mu$) | 2.5k | 0.579 | 0.666 | $\leq 0.91,\ 0.76$; conditional 0.92 |
| Kukulkan, bound fixed ($\mu(c)$) | 2.5k | 0.561 | 0.662 | $\leq 1.02,\ 1.06$; conditional 0.99 |
| Kukulkan, unconstrained | 2.5k | 0.547 | 0.671 | stated |
| Kukulkan, softplus output | 2.3k | 0.495 | 0.686 | n/a |

**The three estimates of $\rho$ and the band usage.** Table 7 gives, per fitted model, the upper bound (mean over test events of $\sum_n W_{kn} x_{s,n}$), the conditional offspring equation 15 (mean and deciles over test events with at least twenty later events in their window), the frozen-context value and the

simulated offspring of Section 6.2 (300 windows, 200 paths), and the median and fifth percentile of $s_{\text{eff}}$ over test intervals. For the exact-$\rho$ model the four numbers coincide, which is the check of the tools. Seed 0 throughout.

Table 7: Three estimates of $\rho$: upper bound, conditional offspring along the observed path (mean; deciles), the frozen-context value and the simulated offspring, and the share of the upper bound the context reads ($s_{\text{eff}}$ median / 5th percentile).

| set | model | NLL | upper bound | conditional | frozen | simulated | $s_{\text{eff}}$ |
|---|---|---|---|---|---|---|---|
| exchange orders | exact $\rho$ | 0.564 | 1.09 | 1.09 (0.94, 1.20) | 1.10 | 1.10 | 1 / 1 |
| exchange orders | gate (Kukulkan) | 0.546 | 2.49 | 1.06 (0.88, 1.28) | 1.09 | 1.00 | 0.58 / 0.25 |
| exchange orders | gate, bound fixed | 0.579 | 0.95 | 0.92 (0.74, 1.06) | 0.92 | 0.93 | 1.00 / 0.85 |
| exchange orders | band 2 | 0.554 | 1.07 | 1.07 (0.90, 1.23) | 1.11 | 1.05 | 1.04 / 0.80 |
| exchange orders | unconstrained | 0.547 | none | 1.11 (0.53, 1.36) | 1.07 | 0.90 | — |
| Taxi | exact $\rho$ | $-0.269$ | 0.03 | 0.03 (0.03, 0.04) | 0.03 | 0.03 | 1 / 1 |
| Taxi | gate (Kukulkan) | $-0.320$ | 3.84 | 2.27 (1.57, 2.28) | 1.87 | 1.05 | 0.18 / 0.01 |
| Taxi | gate, bound fixed | $-0.320$ | 4.34 | 1.24 (0.87, 1.85) | 2.80 | 2.88 | 0.78 / 0.53 |
| Taxi | band 2 | $-0.300$ | 0.16 | 0.30 (0.02, 0.64) | 0.27 | 0.26 | 0.58 / 0.50 |
| Taxi | unconstrained | $-0.327$ | none | 0.42 (0.09, 0.99) | 1.22 | 0.35 | — |
| Taobao | exact $\rho$ | $-1.270$ | 1.35 | 1.35 (1.18, 1.44) | 1.34 | 1.34 | 1 / 1 |
| Taobao | gate (Kukulkan) | $-1.279$ | 1.84 | 1.38 (1.14, 1.61) | 1.41 | 1.26 | 0.81 / 0.63 |
| Taobao | band 2 | $-1.283$ | 1.61 | 1.81 (1.21, 3.30) | 1.97 | 1.67 | 1.15 / 0.76 |
| Taobao | unconstrained | $-1.264$ | none | 1.16 (0.75, 1.63) | 1.19 | 0.99 | — |

**Why the fixed weight is frozen.** Fitted jointly with the gate, the fixed weight rises and the gate spends it down, and the bound of Proposition 5 is true and empty. On exchange orders (seed 0, three-seed upper bound of the exact-$\rho$ model $1.02$–$1.08$ per row with a context $\mu$): a gate fitted jointly with a free fixed weight reports an upper bound of $2.2$ and $2.6$ for the two directions and realises $1.0$ in the simulation test. A gate over a fixed $\mu$ reports $4.6$ and $4.5$. Freezing the output matrix alone leaves the input matrix and the mark scale free and the upper bound moves to $1.8$ and $42$. Freezing the three together fixes it at the exact-$\rho$ model's value. On Taxi the free-fixed-weight gate over a fixed $\mu$ reports row sums of $210$. Warm-started with the fixed weight frozen, the gate moves the orders from $0.564$ to $0.561$ under the exact-$\rho$ model's recipe, to $0.559$ with the longer recipe and to $0.553$ at a learning rate of $0.01$, against $0.546$ for the jointly fitted gate and $0.548$ for the unconstrained model. On Taxi, where the exact-$\rho$ model's amount under a context $\mu$ is $0.07$ per row, the warm-started fit does not move ($-0.268$). The upper bound of the frozen fit is the exact-$\rho$ model's $\rho$ by construction, and its conditional offspring on the orders is $0.99$ of $1.09$.

**Cold against warm.** The narrowest band fitted from a cold start lands $0.008$ nats behind the exact-$\rho$ model it contains on the orders ($0.572$ against $0.564$, dev $0.595$ against $0.586$), and the same band warm-started from the exact-$\rho$ optimum lands $0.003$ ahead. With the warm start as the first early-stopping candidate a warm-started fit cannot end behind its start on the dev set. Every band and gate row in the paper is a warm-started fit.

**Two extensions that keep the closed form.** Both properties of Section 5 survive any change that keeps the intensity linear in a state with a known integral, non-negative and additive over events. We tried two such changes in a small run, single seed, warm-started from the exact-$\rho$ fit, Taxi capped at 60 epochs and Amazon at 25. The first is a gate that varies inside the interval, $s(\tau) = s_0 + (s_1 - s_0)e^{-\gamma\tau}$ with $s_0, s_1 \in (0, 1]$ and $\gamma$ selected by the context; its product with a decaying channel is still a sum of exponentials, so the compensator stays closed-form (checked against a 2,000-cell quadrature to $5 \times 10^{-5}$ per interval) and $s \leq 1$ keeps the bound of Proposition 5. The second pairs channels into $a(e^{-\beta_1\tau} - e^{-\beta_2\tau})$ with $\beta_2 > \beta_1$, a kernel that is zero at the event, rises and then decays, normalised to mass one; it is the only way a linear model can produce the rising hazard that Amazon rewards. Table 8 gives the result. The time-varying gate gains $0.14$ nats on Taxi and $0.55$ on Amazon, $0.65$ with the paired channels, at unchanged next-type accuracy, but fitted jointly it buys the gain by raising the fixed weights while the gate stays near zero most of the time, so the upper bound rises by two orders of magnitude and becomes empty, the same effect as in the paragraph on freezing above. With the fixed weights frozen at the exact-$\rho$ values, so that the bound stays at two to three offspring per event, Amazon still gains $0.16$ nats over the gate, and the paired

channels add nothing. On Taxi the frozen fits stopped at the warm start within the patience budget, so that cell is inconclusive.

Table 8: Two extensions that keep the closed-form compensator and the bound, single seed, epochs capped at 60 (Taxi) and 25 (Amazon): test NLL per event and the mean row sum of the fitted upper bound. The gate row is the paper's model refitted under the same caps.

| model | Taxi NLL | Amazon NLL | Amazon bound | Taxi bound |
|---|---|---|---|---|
| gate (Kukulkan), same caps | $-0.269$ | 1.910 | 76 | 0.08 |
| time-varying gate | $-0.458$ | 1.355 | 2,235 | 243 |
| paired channels + gate | $-0.267$ | 1.781 | 55 | 0.08 |
| paired channels + time-varying gate | $-0.432$ | 1.258 | 6,652 | 64 |
| time-varying gate, fixed weights frozen | $-0.269$ | 1.751 | 2.4 | 0.08 |
| paired channels + time-varying gate, frozen | $-0.267$ | 1.922 | 1.9 | 0.08 |

### C.2 Model ablations

**State computations.** Forward and backward on an RTX 5090, batch 4, $N = 8$. Error is the maximum absolute difference of $h^+$ from the float64 recurrence.

| $L$ | recurrent | quadratic | chunked ($Q$=64) | Mamba-2 Triton kernel |
|---|---|---|---|---|
| 256 | 23 ms, $5\times10^{-7}$ | 87 ms, 33 MiB, $2\times10^{-5}$ | 35 ms, 23 MiB, $2\times10^{-6}$ | 275 MiB, $4\times10^{-3}$ |
| 2048 | 95 ms, $8\times10^{-7}$ | 7 ms, 1.6 GiB, $2\times10^{-4}$ | 0.9 ms, 67 MiB, $4\times10^{-6}$ | 1.2 s, 29 MiB, $5\times10^{-3}$ |
| 16384 | — | — | 1.4 ms, 412 MiB | 0.6 s, 100 MiB |

The Triton kernel's first call includes $\approx 40$ s of autotuning. Its error is tf32 arithmetic inside its matrix products. It becomes attractive only if the per-state channel is widened. The regime filter's scan has the same structure with $K \times K$ products. A training step of the two-regime model on the context output map costs 21 ms forward and 60 ms backward on a batch of 128 windows of 200 orders.

**Softmax ablation.** On the known-truth data of Section C: oracle 1.7778, Kukulkan 1.7779, row-normalised Kukulkan 1.8147 nats per event. Implied $\hat{\rho}$ row sums 0.47 (truth), 0.47 (Kukulkan), 1.26 (normalised).

**The input side.** Making the jump a function of the event's content, $x_s = \mathrm{softplus}(We_s)\, g(m_s)$ in place of the type matrix, changed nothing on StackOverflow and Retweet and collapsed on Taxi (0.537 against 0.313 for the table). A context gain on the table, $x_s = B_{:,k_s} g(m_s) e^{W_B c_s^+}$, is worth 0.00–0.03 nats across the four sets of the design sweep. In the duality view the input of an event is its type and size. What the past means is decided at read time.

**The time change.** One time-scale factor per interval, $s_j = e^{w \cdot c_j}$, scaling the decay and the intensity together (Section 5.2), is exact (we check it against the plain model evaluated in operational time plus $\sum_j \log s_j$, to $10^{-6}$) and leaves $\rho$ unconditional. Its row sums stay near the plain model's (trades 0.88, 0.68 against 0.92, 0.74. StackOverflow 2.1, 1.5 against 2.4, 1.3), where the unconstrained model's per-channel decay selection reports row sums of 4 to 90 that no longer mean anything. The gain-at-birth model of Section 5.2 is the per-channel version of this time change with the output map following each decay rate and $\mu$ not scaled, and it keeps $\rho$ exact. It moves the likelihood by $+0.06$ (Taxi), $+0.01$ (StackOverflow), $+0.006$ (trades) and $+0.002$ (Retweet) nats over the plain Hawkes process, against $+0.63$, $+0.05$, $+0.08$ and $+0.15$ for the context output map and $+0.16$, $+0.04$, $+0.03$ and $+0.11$ for the per-channel selection. With three regimes on top it reaches $-0.30$, $2.243$, $0.604$ and $6.400$ against $-0.28$, $2.254$, $0.577$ and $6.379$ for the regimes with per-channel selection. Adding the context gain on the jump to it collapses on Taxi (0.45) as the content write did, and is worth $0.05$ on trades (0.598) and $0.10$ on Retweet (6.444), where the jump gain carries what the rate alone does not. On Taxi two of three widths report $\rho$ row sums above 8: a strongly varying decay rate on 37-event windows does not enforce stationarity in operational time, so the "unconditional" $\rho$ of a time-changed fit still has to be read with the window length in mind.

**Depth.** With $L$ selective event-SSM blocks under the embedding (Section 5.2), single seed, width by dev, same training recipe as the tables: on MIMIC-II the linear model goes $1.90 \to 1.57 \to 1.46$ for $L = 0, 1, 2$ (accuracy $0.73 \to 0.79$) and the softplus model $1.84 \to 1.76 \to 1.67$, so the set every flat Kukulkan row lost is a matter of mixing depth, although THP (1.27) stays ahead. On Amazon the softplus model goes $0.70 \to 0.25 \to 0.21$, where the flat model already led SAHP by 0.6 nats. On Taxi nothing changes ($-0.33$, $-0.33$, $-0.31$ linear; $-0.51$, $-0.50$, $-0.50$ softplus). On the trade windows the deeper fits stop early at the shared learning rate and land 0.02–0.08 nats behind the flat model. Depth gains likelihood where the type structure is rich (75 types, 16 types) and nothing where there are two or ten. Its cost is that the block's output at $k$ contains event $k$, so nothing downstream of it may be read at $t_j^-$, which is the constraint Section 5.2 builds in.

**Regime count and jump selection.** On Taxi, three regimes on the matrix output reach $-0.28$ against $-0.26$ for two. Four are no better. Selecting the jump from the four-event context instead of the current event changes nothing ($-0.260$ vs $-0.261$ at $K = 2$; $-0.412$ vs $-0.407$ on the context output map). Selected decay adds 0.02–0.04 nats to the context model on trades and StackOverflow and nothing to the regime models.

**Known-truth regime checks.** Table 10 is the single-seed table behind the apparent-criticality result of Section C (gain regimes). Table 9 is the three-seed suite that compares gain regimes with decay-rate regimes (Section 5.2) on a separate dev set: scenarios A ($\mu$ switches 0.1/0.6, $\rho = 0.33$), B (kernel magnitude switches, $\rho$ 0.33/0.80), D (kernel time scale switches, $\rho$ fixed) and C (no switch). "Reaches the oracle" means within 0.005 nats of the generating model on the test set. On C every cold fit of either family improves its training likelihood by inventing a regime, and only the separate dev set exposes it, which is why a regime claim in this paper is gated on the held-out likelihood and not on the fit's own report.

Table 9: Gain regimes against decay-rate regimes, $K = 2$, three cold seeds each: seeds that reach the oracle (test NLL within 0.005), with the best margin where none does; warm starts from the single-kernel fit reach the oracle in every scenario for both families. Right: real data, $K = 3$, test NLL per seed.

| family | A $\mu$ | B magnitude | D time scale | C: regime invented |
|---|---|---|---|---|
| gain regimes | 2/3 | 1/3 | 0/3 (best $+0.014$) | 4/4 |
| decay-rate regimes | 3/3 | 2/3 (one partial) | 2/3 ($+0.0001$) | 4/4 |

| set | gain regimes | decay-rate regimes |
|---|---|---|
| Taxi | $-0.450$ / $-0.456$ / $-0.446$ | $-0.443$ / $-0.442$ / $-0.438$ |
| exchange orders | 0.555 / 0.557 / 0.556 | 0.553 / 0.506 / 0.566 |
| Taxi, exact-$\rho$ output map | — | $-0.431$ / $-0.409$ / $-0.341$ |
| exchange orders, exact-$\rho$ output map | — | 0.560 / 0.560 / 0.567 |

Table 10: Known-truth regime checks: test NLL per event and $\rho$ row sums (mean over the two types). Oracle = the generating model; "warm" = initialised from the fitted single kernel. Regime accuracy = share of test events whose most likely filtered regime is the true one.

| scenario | truth ($\rho_1$, $\rho_2$) | oracle | Hawkes $N{=}D$ | Hawkes $N{=}4D$ | $K{=}2$, cold | $K{=}2$, warm |
|---|---|---|---|---|---|---|
| A: $\mu$ switch | 0.33, 0.33 | 1.429 | 1.477 ($\hat{\rho}$ 0.85) | 1.446 ($\hat{\rho}$ 0.91) | 1.432 ($\hat{\rho}_r$ 0.34, 0.35; acc. 0.95) | — |
| B: kernel switch | 0.33, 0.80 | 0.487 | 0.497 ($\hat{\rho}$ 0.74) | 0.499 ($\hat{\rho}$ 0.82) | 0.501 (dwell 2–6; acc. 0.6) | 0.488 ($\hat{\rho}_r$ 0.79, 0.35; acc. 0.94) |
| C: no switch | 0.33, 0.33 | 1.584 | 1.584 ($\hat{\rho}$ 0.33) | 1.586 ($\hat{\rho}$ 0.52) | 1.586 ($\hat{\rho}$ 0.33, 0.38) | — |

**Decay rate against $\mu$, set by set.** Table 11 separates what the exact-$\rho$ model's two selections buy: the decay rates alone (fixed $\mu$, output matrix times the decay rate. 300–900 parameters) and the decay rates with a context $\mu(c_j)$, next to the classical kernel and the unconstrained model. The decay rates alone carry 0.03–0.12 nats on StackOverflow, Retweet, Amazon, Volcano, Earthquake and the exchange orders and 0.53 on Taxi, and nothing on Taobao and Financial. The context $\mu$ adds 0.04–0.10 on Taxi, StackOverflow, Amazon, Volcano, Earthquake and the orders and costs 0.01 on Retweet. MIMIC-II (2,400 events) is not fit reliably by any row from cold starts.

Table 11: Exact-$\rho$ output map ablation, test NLL per event: classical kernel (nothing selected), decay rates only, decay rates with a context $\mu$ (three-seed mean), and the unconstrained model. Seed 0 unless stated.

| set | classical | decay rates only | decay rates + $\mu(c)$ | free |
|---|---|---|---|---|
| Taxi | 0.313 | $-0.214$ | $-0.268$ | $-0.327$ |
| StackOverflow | 2.291 | 2.257 | 2.176 | 2.188 |
| Retweet | 6.548 | 6.431 | 6.442 | 6.370 |
| Amazon | 2.101 | 2.015 | 1.964 | 1.909 |
| Taobao | $-1.257$ | $-1.247$ | $-1.269$ | $-1.264$ |
| Volcano | 6.465 | 6.364 | 6.321 | 6.317 |
| Earthquake | 1.836 | 1.790 | 1.709 | 1.684 |
| MIMIC-II | 1.589 | 2.782 | 2.173 | 1.897 |
| Financial | $-6.775$ | $-6.769$ | $-6.783$ | $-6.790$ |
| exchange orders | 0.655 | 0.606 | 0.565 | 0.547 |

# D ADDITIONAL RESULTS

## D.1 THE EVALUATION

**The bias on the public datasets.** Table 12 measures on the public datasets the bias that Figure 1 shows inside one interval: the same S2P2 checkpoint scored with the fixed grid and with exact integration. The biases of THP, SAHP and MHP are our re-implementations scored under the 10-point grid and exact integration, and the NHP value is the EasyTPP checkpoint on Taxi, grid-trained ($0.557$ under the grid against $0.509$ exact). Table 13 is the stated-value measurement of Section 3: the branching ratio the unconstrained model states at each event against the offspring counted in its own simulation, 300 test windows and 200 paths each.

Table 12: Quadrature bias of the fixed-grid compensator on the S2P2 checkpoints. Same checkpoint, same event term, only the integration rule changes: log-likelihood per event (higher is better) under EasyTPP's 10- and 20-point grids and under exact integration. "grid" and "random" are S2P2's two training samplers; the last column is the bias of the 10-point rule.

| model (set) | trained on | 10-point | 20-point | exact | bias |
|---|---|---|---|---|---|
| S2P2 (Taxi) | grid | 0.588 | 0.540 | 0.489 | $+0.10$ |
| S2P2 (Taxi) | random | 0.578 | 0.545 | 0.520 | $+0.06$ |
| S2P2 (Taobao) | grid / random | 1.567 / 1.530 | 1.453 / 1.447 | 1.334 / 1.358 | $+0.23$ / $+0.17$ |
| S2P2 (StackOverflow) | grid / random | $-1.988$ / $-2.711$ | $-2.090$ / $-2.434$ | $-2.475$ / $-2.171$ | $+0.49$ / $-0.54$ |
| S2P2 (exchange orders) | grid / random | $-0.226$ / $-0.733$ | $-0.308$ / $-0.542$ | $-0.853$ / $-0.362$ | $+0.63$ / $-0.37$ |
| S2P2 (Amazon) | grid / random | 0.695 / $-0.781$ | $-0.280$ / 0.190 | $-2.485$ / $\approx 0.74$ | $+3.18$ / $\approx -1.5$ |
| S2P2 (Volcano / Retweet) | random | $-7873$ / $-7 \times 10^5$ | $-3940$ / $-4 \times 10^5$ | $-6.05$ / $-5.11$ | $-7867$ / $-7 \times 10^5$ |

Table 13: The branching ratio the unconstrained model states at the event against the offspring counted in its own simulation (Section 6.2).

| set | stated $\rho$ | simulated | set | stated $\rho$ | simulated |
|---|---|---|---|---|---|
| Retweet | 0.83 | 0.84 | StackOverflow | 1.49 | 1.07 |
| Amazon | 2.88 | 2.26 | Taxi | 1.24 | 0.36 |
| Taobao | 1.18 | 0.99 | | | |
| exchange orders | 1.08 | 0.90 | | | |

**De-quantised public sets.** Retweet stamps are whole seconds and StackOverflow stamps lie on a grid of $1.2 \times 10^{-4}$ days. Placing every event uniformly at random inside its cell (order preserved) removes all exact ties. NLL / accuracy, standard split $\rightarrow$ de-quantised: Kukulkan softplus + context 6.404 / 0.602 $\rightarrow$ 6.405 / 0.603 on Retweet and 2.195 / 0.470 $\rightarrow$ 2.197 / 0.471 on StackOverflow. SAHP 6.356 / 0.610 $\rightarrow$ 6.355 / 0.610 and 2.217 / 0.475 $\rightarrow$ 2.217 / 0.475. The tie mass is shared by all models and the ranking is unchanged.

Table 14: The simulated spike process of Figure 1, whole test set (40,422 events): log-likelihood per event (higher is better) of the true parameters and of each fitted model under EasyTPP's fixed grids and under exact integration, and the bias of the 10-point grid (grid minus exact). The true rate spikes on the first grid position, so the grid under-states the true parameters and every model that reproduces the spike (SAHP); THP and MHP smooth the spike away and are scored correctly but fit badly; S2P2 trained against the grid reports, under the grid, a likelihood above that of the true parameters, which no model can reach under exact integration; the same model trained on random points learns the true rate ($-0.413$ exact) and is under-estimated by the grid like it.

| model | compensator in training | 10-point | 20-point | exact | bias |
|---|---|---|---|---|---|
| true parameters | — | $-2.252$ | $-1.244$ | $-0.409$ | $-1.84$ |
| THP (re-implemented) | log-spaced quadrature | $-1.297$ | $-1.290$ | $-1.283$ | $-0.01$ |
| SAHP (re-implemented) | log-spaced quadrature | $-2.245$ | $-1.242$ | $-0.422$ | $-1.82$ |
| MHP (re-implemented) | log-spaced quadrature | $-1.293$ | $-1.288$ | $-1.283$ | $-0.01$ |
| S2P2 | 10-point grid | $+0.132$ | $-0.076$ | $-0.875$ | $+1.01$ |
| S2P2 | 10 random points | $-2.303$ | $-1.269$ | $-0.413$ | $-1.89$ |

**Where the held-out likelihood can still move.** Two information floors, estimated on the training split and scored on the test split under the protocol of Section 3, say how much of each public set's likelihood is history at all. Type part: order-$k$ Markov chains over types against the best neural model's type part (nats per event). Timing part: a histogram of the interval (60 log-spaced bins, tie-aware) conditioned on nothing, on the last type, or on the last type and the tercile of the last interval, against the best neural model's timing part.

Table 15: Information floors. Left: type part, Markov chains over types. Right: timing part, conditional interval histograms. "best" is the best model of Table 2 on that part.

| set | marginal | Markov-1 | Markov-2 | Markov-3 | best neural |
|---|---|---|---|---|---|
| Taxi | 1.134 | 0.303 | 0.300 | 0.299 | 0.212 |
| StackOverflow | 1.897 | 1.740 | 1.729 | 1.794 | 1.521 |
| Retweet | 0.845 | 0.829 | 0.819 | 0.812 | 0.771 |
| Taobao | 2.177 | 1.565 | 1.478 | 1.514 | 1.367 |
| Amazon | 2.251 | 2.193 | 2.151 | 2.157 | 1.881 |
| exchange orders | 0.688 | 0.688 | 0.674 | 0.668 | 0.543 |

| set | none | last type | + last interval | best neural |
|---|---|---|---|---|
| Taxi | $-0.576$ | $-0.710$ | $-0.711$ | $-0.732$ (S2P2) |
| StackOverflow | 0.787 | 0.721 | 0.721 | 0.649 (S2P2) |
| Retweet | 6.117 | 6.108 | 5.774 | 5.579 (K-sp) |
| Taobao | $-2.744$ | $-2.742$ | $-2.744$ | $-2.741$ (S2P2) |
| Amazon | $-2.518$ | $-2.522$ | $-2.521$ | $-2.633$ (S2P2) |
| exchange orders | 0.125 | 0.112 | 0.063 | $-0.181$ (S2P2) |

Neural models beat the third-order chain on the type by 0.04–0.27 nats, which is what timing, marks and longer history add to the type prediction, small against the 0.2–1.9 nats of type entropy that no model reaches. On the timing, a memoryless histogram is within 0.02–0.11 nats of the best neural model on Taxi, Taobao and Amazon. On Taobao it ties S2P2, and on Amazon it beats every Hawkes-family model by more than a nat, so S2P2's Amazon lead is the marginal shape of the interval (a hazard that rises toward three quarters of the interval), not history. History matters for timing on Retweet (0.2 nats beyond last type and last interval) and above all on the exchange orders (0.24 nats beyond the conditional histogram, 0.31 beyond the memoryless one): order flow is the one set here on which the process is Hawkes-like enough for an excitation state to earn its keep, which is why the paper's process tests are run there. Two more measurements point the same way: doubling the training data moves the softplus Kukulkan by 0.01 nats on Taxi (an eighth of the data gives $-0.477$, all of it $-0.510$) and by 0.1 on StackOverflow. An equal-weight mixture of two models' densities gains 0.004–0.04 nats over the better one, with per-event log-likelihoods correlated at 0.95–0.97 between families (S2P2 and the softplus Kukulkan included), so the current models make the same errors on the same events.

**Accuracy.** Table 16 gives next-type accuracy for the runs of Table 2.

Table 16: Public benchmarks: next-type accuracy (higher is better; Volcano has one type). Same runs as Table 2.

| model | Taxi | StackOv. | Retweet | Amazon | Taobao | Volcano | Earthq. | MIMIC-II | Financial |
|---|---|---|---|---|---|---|---|---|---|
| Hawkes, power law | 0.817 | 0.466 | 0.594 | 0.347 | 0.616 | 1 | 0.472 | 0.850 | 0.623 |
| Hawkes, nonparametric | 0.823 | 0.471 | 0.593 | 0.370 | 0.616 | 1 | 0.471 | 0.819 | 0.624 |
| THP | 0.913 | 0.466 | 0.606 | 0.345 | 0.602 | 1 | 0.473 | 0.850 | 0.617 |
| MHP | 0.915 | 0.456 | 0.599 | 0.355 | 0.603 | 1 | 0.472 | 0.844 | 0.617 |
| SAHP | 0.928 | 0.475 | 0.610 | 0.335 | 0.609 | 1 | 0.468 | 0.833 | 0.616 |
| FIM-PP, zero-shot | 0.430 | 0.441 | 0.488 | 0.252 | 0.518 | 1 | 0.460 | — | 0.518 |
| Hawkes, exponential | 0.827 | 0.467 | 0.594 | 0.342 | 0.616 | 1 | 0.472 | 0.777 | 0.619 |
| Kukulkan, unconstrained | 0.922 | 0.476 | 0.605 | 0.403 | 0.619 | 1 | 0.469 | 0.766 | 0.622 |
| Kukulkan, softplus output | 0.928 | 0.473 | 0.609 | 0.366 | 0.616 | 1 | 0.472 | 0.792 | 0.622 |

**Next-time error.** Table 17 gives the root-mean-square error (RMSE) of the expected next interval and the scale-free error $|\log\hat{\tau}-\log\tau|$ (Retweet's RMSE is above $10^4$ s for every model and omitted). The linear Kukulkan is within 0.01 of SAHP on Taxi, StackOverflow, Amazon and Taobao, better on Earthquake and trades, worse on MIMIC-II, where the exponential Hawkes process has the best next-time error of all models. The softplus rows carry a larger seed spread on the time metrics (RMSE $0.92 \pm 0.14$ on MIMIC-II, $0.99 \pm 0.03$ on trades) than the linear rows, a stability cost of the nonlinear output map that the likelihood does not show.

Table 17: Next-time error: RMSE of the expected interval (top) and mean $|\log\hat{\tau} - \log\tau|$ (bottom), in each set's unit. Kukulkan rows are means over three seeds.

| model | Taxi | StackOv. | Amazon | Taobao | Earthq. | MIMIC-II | trades |
|---|---|---|---|---|---|---|---|
| SAHP | 0.280 | 1.016 | 0.333 | 0.126 | 1.268 | 0.947 | 0.968 |
| THP | 0.283 | 1.022 | 0.332 | 0.126 | 1.249 | 0.813 | 1.042 |
| Hawkes, nonparametric | 0.295 | 1.033 | 0.367 | 0.127 | 1.273 | 0.766 | 1.044 |
| Hawkes, exponential | 0.300 | 1.030 | 0.347 | 0.126 | 1.253 | 0.743 | 1.051 |
| Kukulkan | 0.284 | 1.024 | 0.334 | 0.126 | 1.246 | 0.803 | 0.940 |
| Kukulkan, softplus output | 0.287 | 1.028 | 0.348 | 0.126 | 1.233 | 0.917 | 0.992 |
| SAHP | 0.814 | 1.306 | 1.404 | 1.755 | 1.530 | 1.422 | 1.136 |
| THP | 0.815 | 1.323 | 1.379 | 1.571 | 1.605 | 1.321 | 1.206 |
| Hawkes, nonparametric | — | — | 1.392 | 1.678 | 1.736 | 1.207 | 1.174 |
| Hawkes, exponential | 0.938 | 1.313 | 1.401 | 1.636 | 1.744 | 1.099 | 1.166 |
| Kukulkan | 0.825 | 1.299 | 1.369 | 1.705 | 1.433 | 1.286 | 1.135 |
| Kukulkan, softplus output | 0.827 | 1.301 | 1.386 | 1.690 | 1.390 | 1.342 | 1.121 |

### D.2 FINANCIAL DATA

**What the three-regime fits on the orders contain.** The rate-regime fits of Table 9 on the exchange orders (context $\mu(c_j)$ from the four-event convolution, $K = 3$ banks of four channels) put the mass in $\mu$: seed 0 has regime occupancies 0.86 / 0.13 / 0.005 with $\rho_r$ row sums (0.06, 0.05), (0.68, 0.21) and (0.19, 0.15) (proxies, unconstrained output map). Seed 1 has one regime at 0.99 occupancy with row sums (0.21, 0.11). The exact-$\rho$ output map version (exact $\rho_r$. Test NLL 0.560 / 0.560 / 0.567 over three seeds) has one regime at 0.96–0.99 occupancy and dwell 18,000–22,000 events with row sums (0.18, 0.09), (0.23, 0.17) and (0.29, 0.20), a rare regime of a few events with row sums 0.16–0.32, and a third regime with 1–4% occupancy and row sums below 0.15. On Taxi the exact-$\rho$ output map regime fits use the regimes as a fast jump machine (dwell 3–4 events, $\rho_r$ row sums 0.06–0.13, seed spread 0.09 nats), a degenerate use. A regime layer over a context $\mu$ finds a slow modulation of $\mu$ and little excitation on the orders, the same absorption as in Section 6.2. The calm/active/burst description of Table 18 comes from the fixed-$\mu$ matrix output, where the excitation has nowhere else to go.

**The three-regime diagnostic on the orders.** Regime 1 holds 95% of the stationary mass and 63% of test events: a calm regime in which a buy excites sells ($\rho_{\text{sell}\leftarrow\text{buy}} = 0.74$) more than anything excites its own side (self share 0.23), the mean-reverting order flow of a two-sided book. Regime 2 (36% of events by posterior, entered through the event-selected jumps) has a $\mu$ twenty times higher

and a cross-excitation of 4.2: an active, oscillating market. Regime 3 (0.3% of events, intervals of a few milliseconds) is a burst in which every kernel is large and self-excitation has half the mass. Its branching ratios (30 to 36) are not stationary quantities, since the regime lasts a few events, and it lives at the batch scale of the feed, so we read it as "several orders in one batch" rather than as a microsecond mechanism. A single kernel fitted to the same data reports $\rho$ row sums of 0.92 and 0.74 for the two directions and cannot say which of these it is describing. The jump probabilities depend weakly on the event (larger orders and sells switch more), and the belief moves mainly through the likelihood: a regime is inferred from the evidence of the last tens of events, as in Section 5.2.

Table 18: Trades, three-regime Kukulkan: stationary occupancy, dwell (events), $\mu$ per type and $\rho_r$ (row = target, column = source, buy first).

| regime | occupancy | dwell | $\mu$ (buy, sell) | $\rho_r$ |
|---|---|---|---|---|
| 1 calm | 0.95 | $> 10^4$ | 0.032, 0.016 | $[[0.07, 0.03], [0.74, 0.16]]$ |
| 2 active | 0.05 | $5 \times 10^3$ | 0.65, 0.50 | $[[0.33, 0.14], [4.20, 0.38]]$ |
| 3 burst | $< 0.01$ | 43 | 0.037, 0.018 | $[[30, 23], [36, 17]]$ |

Table 19: The ridge of Section 3 on exchange order flow: three seeds of the exact-$\rho$ model, within $0.002$ nats of each other in test likelihood, and the branching ratio each reports (buy, sell row sums) with $\mu$ selected by the context and with $\mu$ fixed.

| seed | $\rho$ with $\mu(c)$ | $\rho$ with $\mu$ fixed |
|---|---|---|
| 0 | 1.02, 1.06 | 0.91, 0.76 |
| 1 | 1.08, 1.01 | 0.88, 0.78 |
| 2 | 0.72, 0.74 | 0.86, 0.75 |

**Branching ratios.** For the unconstrained model the DC gain at a context is a proxy (Section 6.2). Its median row sums over test events are 0.6–0.9 on Taxi, StackOverflow and Earthquake and 0.9 and 0.7 for the two directions on exchange orders, against 0.92 and 0.74 for the exponential kernel, and on Retweet the two frequent types exceed 1 (cascades conditioned on a root are not stationary). For the exact-$\rho$ model $\rho$ is the offspring by Proposition 5, and the offspring test of Section 6.2 returns the same integral on every simulated path, on exchange orders (fixed $\mu$ 0.95, context $\mu$ 1.10 on seed 0) and on Taxi. On Taxi the two choices of $\mu$ bracket the answer: with a fixed $\mu$ the fit reports row sums up to 36, which no stationary process has, and with a context $\mu$ 0.07, i.e. $\mu$ absorbs the excitation. Taxi's alternation of pickups and drop-offs is not a Hawkes process, and an exact $\rho$ makes that visible where the proxy ($1.24$ stated, $0.36$ simulated) did not. The regime-conditional $\rho_r$ of the trades diagnostic in Table 18 are proxies of the same kind.

**The free run against the data's multi-scale variance.** Order flow is over-dispersed at every scale, and the over-dispersion grows: deseasoned by the intraday profile, the Fano factor of the order count over sixty sessions rises as roughly the bin width to the power $0.7$ from $4.5$ at ten seconds to $137$ at twenty minutes, and order signs have long memory. A generator that has learned the process should reproduce this curve from a real warm-up. Figure 5 runs each fitted model forward for fifteen minutes under the protocol it was trained under, with the state recomputed from the last 200 orders at every step. No model reproduces the curve: all over-disperse by $2.5$ to $5$ times from eight seconds up, and the two S2P2 fits, which are within $0.002$ nats of each other in held-out likelihood, generate order rates six times apart. Within our family the best-likelihood linear model is the least stable generator (seven of sixteen paths run away) and the softplus model runs away in three. The exact-$\rho$ model runs away in none, generates the data's order rate within 15% and is the closest of the model class to the data's curve at every width, while still over-dispersing by two to three times from eight seconds up. The window protocol forgives a supercritical fit, since the state is rebuilt from 200 orders at every step. In a one-hour free run with the state carried, the exact-$\rho$ fit whose own $\rho$ is $1.10$ runs away on every path, as its $\rho$ says it must, and the fixed-$\mu$ fit with $\rho = 0.95$ does not (15k orders per hour, Fano 15 at eight seconds). An exact $\rho$ predicts the generator. The unconstrained model's proxy ($1.08$ stated, $0.90$ simulated, runaway under both protocols) does not. This is the test the finance literature runs on every Hawkes model of order flow (Kimura, 2026).

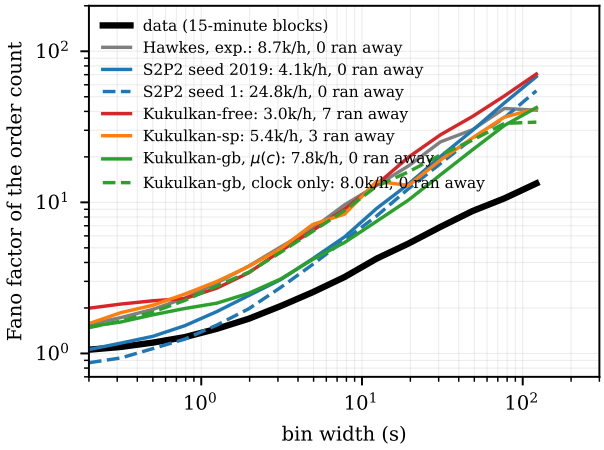



Figure 5: The free run from a real warm-up on exchange order flow, state recomputed from the last 200 orders at every step as in training, 15 minutes, 16 paths: the Fano factor of the order count against the bin width, with the median order rate per hour and the number of paths that ran away in the legend. The data are 15-minute blocks of the test sessions.