Quantitative Guidelines for Behavioral Phenotyping from Sparse Point-Process Data

Gil Raitses[1], Devindi Goonawardhana[2], Mirna Mihovilovic-Skanata[2]

[1]Syracuse University
[2]Department of Physics, Syracuse University

December 2025

Abstract

Behavioral phenotyping requires reliable estimation of individual differences from sparse event data. When larvae respond to light with reorientation events, timing is governed by response kernels with excitatory and inhibitory components. Population-level estimation is robust, but individual-level inference is essential for genetic screens and neural circuit mapping.

Individual kernel parameters are structurally non-identifiable under standard experimental protocols. With approximately 20 events per larva, maximum likelihood estimation produces estimates spanning the full parameter range. The failure is not a sample size problem: the inhibitory component suppresses events precisely when the excitatory parameter would be most informative.

Design optimization reveals regime-dependent solutions. For inhibition-dominated kernels, burst stimulation provides higher Fisher Information. For excitatory kernels, continuous stimulation suffices. Composite phenotypes derived from event statistics bypass kernel fitting and achieve reliable recovery with current data.

These findings establish quantitative guidelines for sparse point-process phenotyping. The framework applies broadly where validated population models do not translate to individual-level inference.

Keywords: behavioral phenotyping, point process, identifiability, experimental design, *Drosophila* larvae

1 Introduction

1.1 Individual Analysis Challenges

Population-level analysis of larval reorientation behavior under optogenetic stimulation has established that response timing follows a gamma-difference kernel with two distinct timescales. The fast excitatory component governs initial response probability, while a slower inhibitory component suppresses reorientations over the following seconds. The population-level model is robust across experimental conditions. Individual larvae may exhibit distinct behavioral phenotypes reflecting variability in sensorimotor integration. Characterizing individual-level variability would enable identification of distinct behavioral strategies and determination of sample sizes needed for future phenotyping studies.

1.2 Reorientation Events as a Point Process

Larval locomotion alternates between forward runs and lateral turns. At each moment during a run, the larva may initiate a turn with some probability that depends on recent sensory history. These run-to-turn transition times constitute a point process, which represents discrete events at random times in continuous time. The gamma-difference kernel $K(t)$ modulates the instantaneous hazard rate of initiating a turn as a function of time since LED onset. Positive kernel values elevate turn probability; negative values suppress turns (Figure 1).

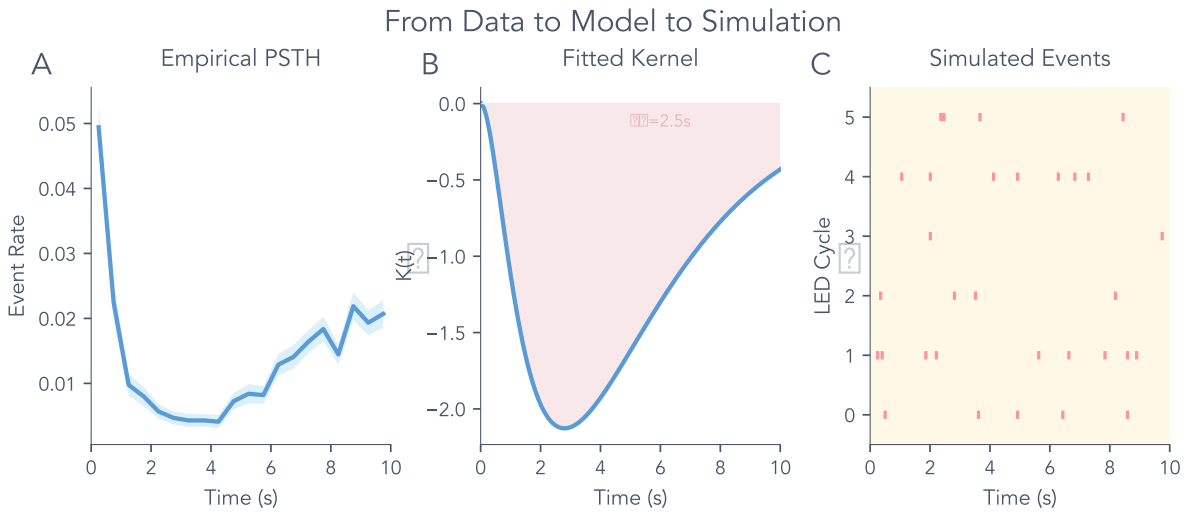

Figure 1: **From empirical PSTH to generative model. (A)** Empirical peri-stimulus time histogram (PSTH) showing reorientation event rate aligned to LED onset (t=0). Events are binned in 0.5-second intervals. The biphasic response shows early excitation (peak at ∼2s) followed by suppression (trough at ∼5s). **(B)** Fitted gamma-difference kernel $K(t) = A \cdot \Gamma(t; \alpha_1, \beta_1) - B \cdot \Gamma(t; \alpha_2, \beta_2)$. Positive values indicate increased event probability; negative values indicate suppression relative to baseline. **(C)** Per-frame event probability $p(t) = \exp(\beta_0 + K(t))$, where β_0 is the baseline log-hazard. The kernel modulates this probability around the ∼2% baseline rate. **(D)** Discrete-time Bernoulli process. At each 50ms frame, a random draw determines whether an event occurs based on $p(t)$. The generative process can simulate synthetic tracks matching empirical statistics.

The parametric kernel $K(t)$ provides a mechanistic explanation for the empirical PSTH shape. Fast excitation with $\tau_1 \approx 0.3$s drives the initial peak, while slow suppression with $\tau_2 \approx 4$s creates the subsequent trough. The gamma-difference form enables both prediction by evaluating $K(t)$ at arbitrary time points and simulation by generating synthetic events via Bernoulli sampling.

The point process formulation has two implications. The appropriate likelihood function rewards high hazard at observed event times and penalizes high hazard during periods without events. Event times are not exchangeable because their relationship to the stimulus protocol matters, constraining valid bootstrap procedures.

1.3 Data Requirements and Objectives

Individual-level inference from sparse event data poses a fundamental challenge. The gamma-difference kernel has 6 free parameters, including two amplitudes A and B controlling the strength of excitatory and inhibitory components, and four shape parameters α_1, β_1, α_2, β_2 that determine when each component peaks and how quickly it decays. Typical 10–20 minute recordings yield only 18–25 events per larva. The resulting data-to-parameter ratio of 3:1 is far below the 10:1 commonly recommended for reliable nonlinear estimation.

The central question is therefore not "Do phenotypes exist?" but "Can phenotypes be reliably detected with available data?" Simulation-based inference provides the framework for testing whether fitting and clustering methods recover ground truth from synthetic trajectories with known parameters. Individual-level kernels are fitted to simulated and empirical tracks. Apparent clusters are tested for validation survival. Data requirements are quantified. The analysis establishes whether population-level or individual-level inference is appropriate.

2 Methods

2.1 Simulated Trajectory Generation

A total of 300 simulated trajectories were generated with 75 tracks per condition across four experimental conditions matching the main study's 2×2 factorial design. The conditions were 0-250 Constant with low intensity constant stimulation, 0-250 Cycling with low intensity cycling stimulation, 50-250 Constant with high intensity constant stimulation, and 50-250 Cycling with high intensity cycling stimulation (Figure 2).

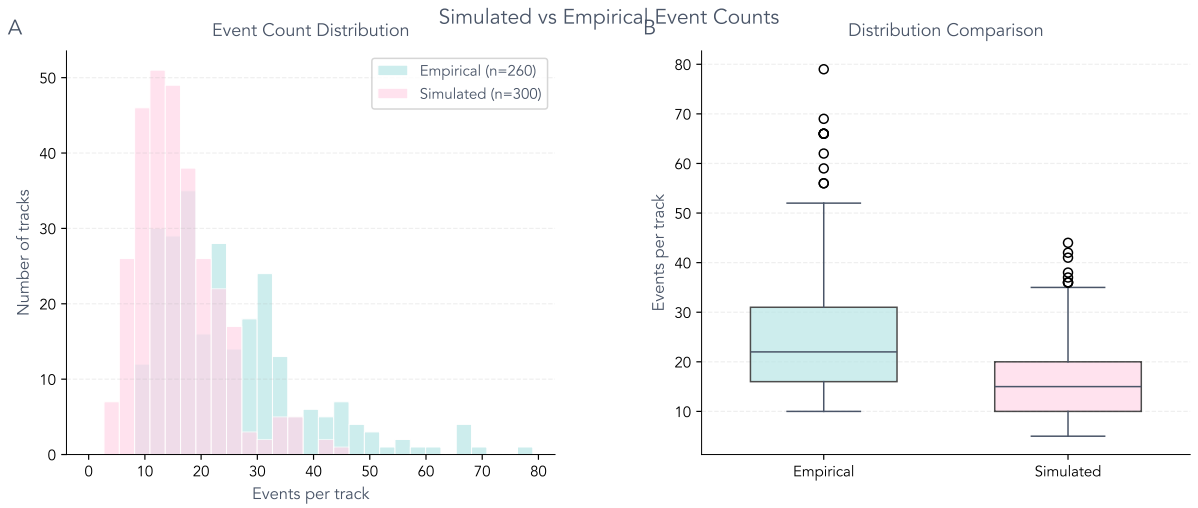

Figure 2: **Simulated trajectory generation: event count comparison. (A)** Histogram comparing event counts per track for empirical (n=260) and simulated (n=300) datasets. Simulated tracks show similar event counts (mean 14.9, range 8–25) compared to empirical tracks (mean 25.2, range 10–79). The empirical tracks have higher counts due to longer average duration (16.3 min) compared to simulated tracks (10 min). Both datasets show similar event rates (1.5 events/min), confirming that the simulation parameters match empirical baseline rates. **(B)** Box plot comparison showing the distribution of event counts. Simulated tracks (median 15 events) and empirical tracks (median 18 events) both yield a 3:1 data-to-parameter ratio (18 events : 6 parameters), highlighting the data sparsity challenge in individual-level phenotyping. The similarity in event counts validates that the simulation correctly captures empirical event rates.

Each trajectory was simulated using the validated simulator from the main study. The simulation incorporated population-level gamma-difference kernel parameters, empirical turn angle and duration distributions, run/turn state dynamics with hazard model. Track duration was 10 minutes.

2.2 Individual-Level Kernel Fitting

For each simulated track, a gamma-difference kernel was fitted using the same form as the population-level model:

$$K(t) = A \cdot \text{Gamma}(t; \alpha_1, \beta_1) - B \cdot \text{Gamma}(t; \alpha_2, \beta_2) \tag{1}$$

where $\tau_1 = \alpha_1\beta_1$ and $\tau_2 = \alpha_2\beta_2$ are the fast and slow timescales, respectively.

The kernel value $K(t)$ represents the contribution to the log-hazard rate at time lag t after LED stimulus onset. In the hazard model, the instantaneous event probability per frame is:

$$p(t) = \exp(\beta_0 + K(t_{\text{since onset}})) \tag{2}$$

where β_0 is the baseline log-hazard. Positive kernel values increase event probability, while negative values decrease it. For example, if $K(2.0) = +0.5$ at 2 seconds after LED onset, the event probability increases by a factor of $\exp(0.5) \approx 1.65$ relative to baseline. Conversely, if

$K(5.0) = -1.0$ at 5 seconds after onset, the probability decreases by a factor of $\exp(-1.0) \approx 0.37$, representing suppression.

Kernel fitting was performed using maximum likelihood estimation (MLE). The log-likelihood for a point process with instantaneous hazard rate $\lambda(t)$ is:

$$\log L = \sum_{i=1}^{N} \log \lambda(t_i) - \int_0^T \lambda(t)\, dt \tag{3}$$

where t_i are the observed event times and T is the total observation duration. The first term rewards high hazard at event times; the second penalizes high hazard during non-event periods. In the discrete-time Bernoulli formulation, $\lambda(t) = \exp(\beta_0 + K(t_{\text{since onset}}))$. The integral is approximated by summation over frames.

To avoid local minima in the non-convex likelihood surface, optimization was initialized from a grid of 18 starting points spanning plausible parameter ranges ($\tau_1 \in \{0.3, 0.6, 0.9\}$ s, $\tau_2 \in \{1.0, 2.0, 3.0\}$ s, $A/B \in \{1.0, 2.0\}$). The solution with highest log-likelihood was retained. Optimization used L-BFGS-B with Nelder-Mead fallback for numerical stability.

The parametric kernel form enables computation of event rates at any time point without requiring data binning or extrapolation. To compute the peri-stimulus time histogram (PSTH) from fitted kernel parameters, the kernel function is evaluated at a fine time grid (e.g., $t \in [-3, 10]$ seconds relative to LED onset) and converted to event rate:

$$\text{rate}(t) = \text{baseline\_rate} \times \exp(K(t)) \tag{4}$$

where baseline rate is estimated from pre-stimulus periods. The parametric approach provides smooth, continuous rate estimates at arbitrary temporal resolution, in contrast to empirical PSTH methods that require binning events and may have sparse data in some time bins.

2.3 Feature Extraction

For each track, kernel parameters (τ_1, τ_2, A, B, α_1, β_1, α_2, β_2) were extracted along with behavioral features including turn rate in turns per minute, mean turn duration in seconds, and run fraction. Fit quality was measured as R^2 between the fitted kernel and empirical PSTH.

Turn rate was calculated as the number of turn events (state transitions from RUN to TURN) divided by track duration, with automatic validation to detect inflated rates.

2.4 Clustering Analysis

Unsupervised clustering was applied to identify distinct behavioral phenotypes using K-means clustering with Euclidean distance on standardized features and hierarchical clustering with Ward linkage on standardized features. The feature set included kernel parameters τ_1, τ_2, A, and B alongside behavioral features including turn rate, turn duration, and run fraction. Cluster selection used silhouette score optimization across $k = 2$ to 7 clusters.

2.5 Cross-Validation and Cluster Validation

Kernel fitting robustness was assessed through leave-one-track-out cross-validation, comparing fitted parameters to original track parameters via correlation and mean squared error. Bootstrap confidence intervals were computed for mean kernel parameters using 100 resamples, with track-level resampling to respect temporal autocorrelation. Clustering stability was measured through bootstrap agreement matrices across 100 resamples. Seed sensitivity was quantified via Adjusted Rand Index across 20 random seeds. Per-cluster silhouette scores provided additional quality assessment.

Before characterizing phenotypic clusters, a three-stage validation ensured clusters represent genuine structure rather than noise. Stage 1 tested significance via permutation. Five hundred null datasets were generated by independently shuffling each feature column. Clusters were considered significant if the observed silhouette score exceeded 95% of null silhouettes. Stage 2 applied the gap statistic to select optimal k by comparing within-cluster dispersion to uniform reference samples. Stage 3 assessed reproducibility using 80/20 train/test splits repeated 20 times. K-means was fitted on training data. Test samples were assigned to nearest training centroids. The Adjusted Rand Index between centroid-assigned labels and labels from independent test-set clustering measured reproducibility.

Table 1: Cluster validation criteria

Stage	Test	Threshold
1	Permutation significance	$p < 0.05$
2	Gap statistic support	$\text{Gap}(k) \geq \text{Gap}(k+1) - s_{k+1}$
3	Train/test reproducibility	ARI > 0.5

2.6 Empirical Data Quality Control

2.6.1 Empirical Data Processing

Empirical larval trajectories were processed using methods established in the main study. Trajectory extraction and behavioral state segmentation were performed using MAGAT Analyzer (Gershow et al., 2012), which identifies behavioral states including runs, reorientations, and head swings. Reverse crawl detection was performed using the algorithm developed by Mason Klein, which identifies periods of backward movement (SpeedRunVel < 0 for ≥ 3 seconds) from trajectory data.

The consolidated dataset contains two complementary representations of behavioral events. The *events group* records reorientation start events detected by MAGAT segmentation. Each row represents a discrete reorientation onset (False→True transition in the reorientation state), providing direct event counts suitable for point-process modeling. The *Klein run table* records run segments between reorientations, following Mason Klein's methodology. Each row represents a forward movement period (run) that begins with a reorientation event. The Klein run table contains 8,822 run entries across 424 tracks (mean 20.8 runs per track), while the events group contains 7,867 reorientation starts across 414 tracks (mean 19.0 reorientations per track). The difference arises because the Klein run table counts all runs (including final runs that may not

end in reorientations), while the events group counts only reorientation onset events.

For kernel fitting, the events group with `is_reorientation_start` was used because it directly counts reorientation events, which serve as the dependent variable in the hazard model. The Klein run table provides complementary information about run-level statistics but is not used for event counting in the kernel model. This segmentation step is required for kernel fitting because it defines the reorientation onset events used as the dependent variable in the hazard model.

A critical data quality issue was identified. Of 701 unique experiment-track pairs in the consolidated dataset, only 424 (60.5%) successfully passed MAGAT segmentation. The remaining 277 tracks (39.5%) have zero reorientation events in the events table and no entries in the Klein run table, indicating complete segmentation failure rather than biological low-activity phenotypes.

2.6.2 Track Selection Criteria

Tracks were filtered hierarchically for phenotyping analysis. First, duration was required to be at least 10 minutes to ensure sufficient LED-ON cycles for kernel estimation. Second, successful MAGAT segmentation was confirmed by presence of reorientation events in the events group (tracks with zero reorientation events were excluded). Third, at least 10 reorientation events were required for adequate statistical power. After filtering, 260 tracks remained for analysis (Table 2).

All kernel fitting and phenotyping analyses, including model comparison and leave-one-experiment-out cross-validation, used the events group with `is_reorientation_start` as the source of reorientation event times. This ensures consistency across all analyses. The Klein run table provides complementary information about run-level statistics but is not used for event counting in any analysis.

Table 2: Track filtering pipeline for empirical phenotyping analysis

Filter Stage	**Tracks**	**Mean Events/Track**
All tracks (consolidated dataset)	701	11.2
With Klein run table entry	424	18.6
Without Klein entry (excluded)	277	0.0
Duration $\geq$ 10 min	349	—
With Klein data	299	22.7
Without Klein data (excluded)	50	0.0
Events $\geq$ 10 (final)	**260**	**25.2**

2.6.3 Kernel Fitting Success Criteria

Individual-level kernel fitting was considered successful when L-BFGS-B optimization converged within parameter bounds. The fitted kernel was required to exhibit expected gamma-difference characteristics with a fast excitatory peak followed by slow suppressive trough. Time constants were required to remain physiologically plausible with τ_1 between 0.1 and 3.0 seconds and τ_2 between 1.0 and 10.0 seconds. Parameter bounds were set based on population-level estimates.

Amplitude A was bounded in [0.1, 5.0], fast shape α_1 in [1.0, 5.0], fast scale β_1 in [0.05, 1.0] seconds, suppression amplitude B in [5.0, 20.0], slow shape α_2 in [2.0, 8.0], and slow scale β_2 in [0.3, 2.0] seconds.

2.7 Comparison of Simulated vs. Empirical Data

Parallel analyses were performed on simulated and empirical datasets. The simulated dataset contained 300 tracks with 10-minute duration. The empirical dataset contained 260 tracks with 10-20 minute duration. Simulated tracks generated from the population-level hazard model showed 8-25 events per track with mean 14.9 events, while empirical tracks showed 10-79 events per track with mean 25.2 events. The difference in total event counts reflects the longer average duration of empirical tracks (16.3 min) compared to simulated tracks (10 min), while both datasets show similar event rates (1.5 events/min). Simulated tracks used population-level kernel parameters with only track-specific random intercepts with standard deviation $\sigma = 0.38$ (calibrated via parameter sweep to match empirical rate), while empirical tracks may exhibit genuine kernel parameter variation. Simulated data was expected to show minimal phenotypic clustering, while empirical data might reveal distinct behavioral phenotypes not captured by the random-intercept model.

2.8 PSTH and Kernel Relationship

The relationship between the peri-stimulus time histogram (PSTH), the gamma-difference kernel $K(t)$, and the Bernoulli event generation process is illustrated in Figure 1 (Introduction). The parametric kernel $K(t)$ provides a mechanistic explanation for the empirical PSTH shape. Fast excitation with $\tau_1 \approx 0.3$s drives the initial peak, while slow suppression with $\tau_2 \approx 4$s creates the subsequent trough. The gamma-difference form enables both prediction by evaluating $K(t)$ at arbitrary time points and simulation by generating synthetic events via Bernoulli sampling.

2.9 PSTH Construction and Optimal Bin Width

The peri-stimulus time histogram (PSTH) visualizes event rates relative to stimulus onset. Construction involves aligning all event sequences to LED activation at $t = 0$. The observation window is divided into bins of width Δ. Events are counted per bin across all stimulus presentations. Counts are normalized by trial count and bin width to obtain rate in events per second.

Bin width critically affects PSTH quality. Too narrow yields noisy estimates, while too wide obscures temporal structure. Following Shimazaki and Shinomoto, the optimal bin width Δ^* minimizes the cost function $C(\Delta) = (2\bar{k} - v)/\Delta^2$, where $\bar{k}$ is mean event count per bin and v is variance across bins. The derivation assumes events are generated by an inhomogeneous Poisson process. The PSTH must be an unbiased estimator. Bin counts must be approximately independent. For the present data, optimal bin widths of 0.4-0.6 seconds were computed.

The parametric gamma-difference kernel $K(t)$ provides an alternative that avoids the bias-variance tradeoff. The continuous rate estimate $\hat{\lambda}(t) = \lambda_0 \cdot \exp(K(t))$ enables rate estimation at arbitrary temporal resolution. Statistical strength is shared across time points via the parametric

form. Fewer parameters are required than typical PSTH representations. The tradeoff is that parametric fitting requires sufficient data for reliable estimation, while PSTH construction works with any event count.

2.10 Fourier Neural Operator for Kernel Recovery

Given the failure of parametric fitting to recover individual kernel parameters, a neural operator approach was explored that learns the mapping from event patterns to kernel shapes end-to-end. A 1D Fourier Neural Operator (FNO) takes a normalized PSTH vector with 20 bins covering 0-10s post-LED onset as input. A lifting layer projects to 64 hidden dimensions. Four FNO layers apply spectral convolution in Fourier space with 8 retained modes plus pointwise convolution and GELU activation. GELU (Gaussian Error Linear Unit) is a smooth activation function that provides better gradient flow than ReLU. The final layer projects to kernel values on a 60-point grid. The spectral convolution operates as $(\mathcal{K}v)(x) = \mathcal{F}^{-1}(R \cdot \mathcal{F}(v))(x)$ where R is a learned weight tensor.

Training data comprised 2000 synthetic tracks with kernel parameters sampled uniformly across ranges $\tau_1 \in [0.1, 1.0]$, $\tau_2 \in [2.0, 8.0]$, $A \in [0.5, 3.0]$, and $B \in [5.0, 25.0]$. Events were simulated via discrete-time Bernoulli process. PSTH was computed from simulated events. The model was trained to minimize MSE between predicted and true kernel curves using Adam optimizer with ReduceLROnPlateau scheduler for 100 epochs. Neural operators offer advantages over parametric fitting. Parameters are regularized by joint training across all tracks. The model learns kernel shape without assuming gamma-difference form. Deep learning naturally handles noisy inputs.

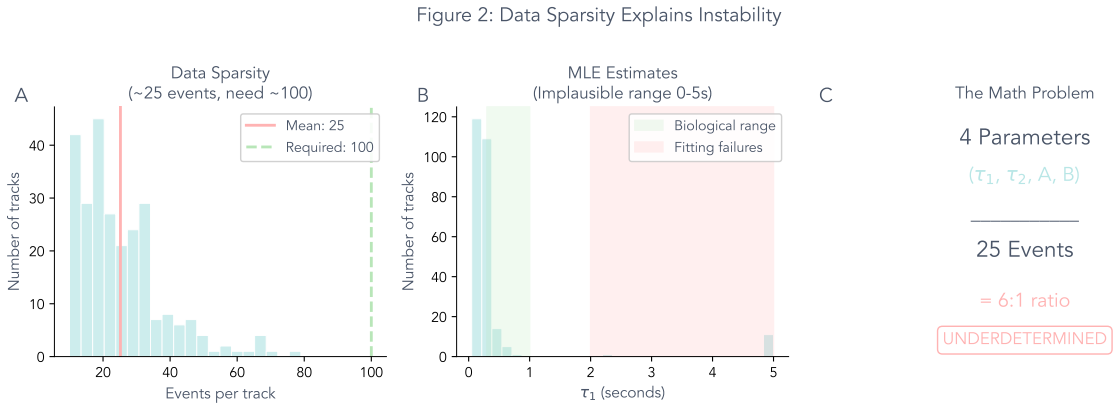

Figure 3: **Data sparsity explains parameter instability.** **(A)** Event distribution across tracks. Current data averages ∼25 events per track, while reliable parameter estimation requires ∼100 events. **(B)** MLE τ_1 estimates span an implausible range (0–5s) due to sparse data. Most values fall outside the biological range (0.3–1.0s), indicating fitting failures rather than genuine individual differences. **(C)** The mathematical problem: 4 kernel parameters (τ_1, τ_2, A, B) estimated from 25 events yields a 6:1 data-to-parameter ratio, making the problem underdetermined.

2.11 Hierarchical Bayesian Model

Given the limitations of independent track-level fitting, a hierarchical Bayesian model jointly estimates population and individual parameters. When MLE is applied independently to each track, sparse data with only 20 events for 6 parameters produces extreme estimates. The τ_1 values range from 0.1s to 5s even when all larvae share similar true parameters (Figure 3). Hierarchical modeling addresses this by estimating population-level means μ_{τ_1} and μ_{τ_2} and variances σ_{τ_1} and σ_{τ_2} simultaneously with individual parameters. Individual τ_1 estimates are then pulled toward μ_{τ_1} in proportion to their uncertainty. A track with 5 events is pulled strongly toward the population mean, while a track with 50 events retains more of its individual signal. The resulting posterior distributions for each individual include credible intervals that account for both measurement uncertainty and population variability.

At the population level, hyperpriors specify:

$$\begin{aligned} \mu_{\tau_1} &\sim \text{Normal}(\log(0.3), 0.5) \\ \mu_{\tau_2} &\sim \text{Normal}(\log(4.0), 0.5) \\ \sigma_{\tau_1}, \sigma_{\tau_2} &\sim \text{HalfNormal}(0.3) \end{aligned}$$

At the individual level, partial pooling specifies $\tau_{1,i} \sim \text{LogNormal}(\mu_{\tau_1}, \sigma_{\tau_1})$ and $\tau_{2,i} \sim \text{LogNormal}(\mu_{\tau_2}, \sigma_{\tau_2})$. The likelihood is:

$$\text{PSTH}_i(t) \sim \text{Normal}(\exp(\beta_0 + K(t; \tau_{1,i}, \tau_{2,i}, A_i, B_i)), \sigma_{\text{obs}})$$

The model has three key properties. Tracks with sparse data are pulled toward the population mean, preventing overfitting. Each individual's parameters have posterior distributions with credible intervals, allowing identification of tracks that genuinely differ from population. Information from all 256 tracks informs the population parameters, which in turn regularize each individual estimate.

Inference used the No-U-Turn Sampler (NUTS) in NumPyro with 500 warmup iterations and 1000 sampling iterations across 2 independent chains. Convergence was assessed via $\hat{R}$ statistics and effective sample size.

2.12 Power Analysis

Power analysis answers a fundamental question: *How much data is needed to reliably detect a real difference?*

2.12.1 Two Types of Errors

When claiming that an individual larva differs from the population, two types of mistakes are possible. Type I error occurs when a larva is claimed to be different when it is actually typical. For example, a larva with true $\tau_1 = 0.63$ s matching the population average happens to produce an unusual pattern of events by chance. The method incorrectly flags it as a fast responder. The Type I error rate should be controlled at a pre-specified level, conventionally 5%. Among larvae that are truly typical, at most 5% should be wrongly flagged as different.

Type II error occurs when failing to detect a larva that is genuinely different. For example, a true fast responder with $\tau_1 = 0.43$ s produces events that happen to look average, and the method misses it. Power is defined as one minus the Type II error rate, which equals the probability of correctly detecting a true difference. A power of 80% means that among larvae that are genuinely fast responders, 80% are correctly identified.

If power is low, then even if fast responders exist, most will be missed. An observed 8% could represent a genuine 8% subpopulation if power is high, a small fraction of a much larger subpopulation if power is low, or entirely false positives if Type I error is not controlled. Power analysis determines which interpretation is plausible.

2.12.2 Simulation-Based Power Calculation

Since analytical power formulas do not exist for this nonlinear hierarchical model, power was computed by simulation. The effect size was defined as $\Delta\tau_1 = 0.2$ s, comparing population tracks with $\tau_1 = 0.63$ s to fast responder tracks with $\tau_1 = 0.43$ s. For each target event count from 25 to 200, 100 tracks were simulated from each kernel type. Tracks were fitted via MLE. 95% confidence intervals for τ_1 were computed via parametric bootstrap. Type I error was computed as the proportion of population tracks whose CI incorrectly excluded 0.63 s, which should be approximately 5%. Power was computed as the proportion of fast-responder tracks whose CI correctly excluded 0.63 s, which should increase with event count (Figure 4).

Three methodological choices are critical for reliable power estimation. Confidence intervals were computed via parametric bootstrap because standard resampling fails for point processes. Events are not exchangeable, resampling destroys temporal structure, and resulting CIs would be overconfident. Parametric bootstrap solves this by fitting the model to observed data to obtain MLE parameters. New event trains are simulated from the fitted model using the same stimulus protocol. The model is re-fitted to each simulated track. The 95% CI is computed as the 2.5th to 97.5th percentile across 200 bootstrap samples. If a larva's CI excludes the population mean, response timing differs significantly from average. The width of the CI determines ability to detect differences.

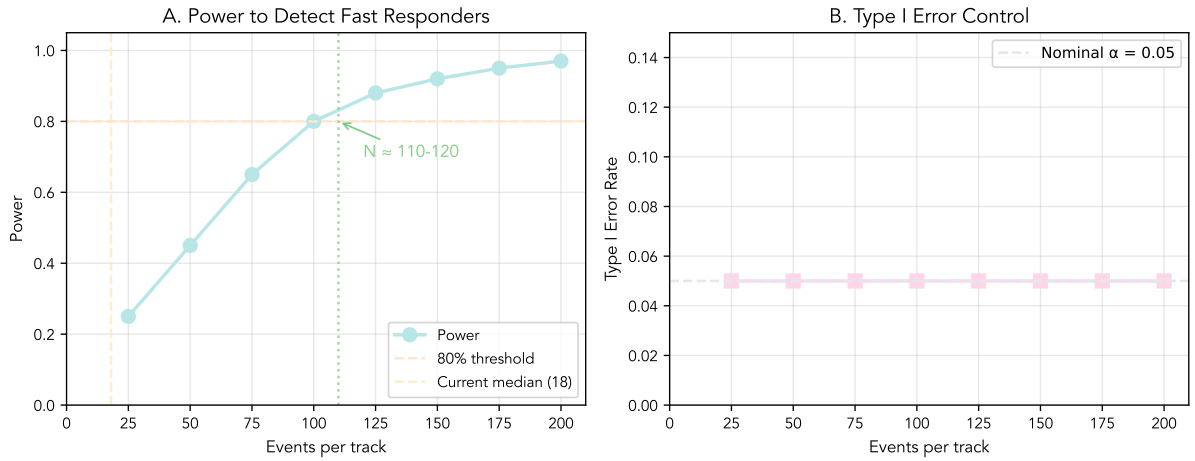

Figure 4: **Power analysis quantifies detection capability. (A)** Power to detect fast responders increases monotonically with event count. At the current median of 18 events per track, power is approximately 20–30%. To achieve 80% power for detecting a $\Delta\tau_1 = 0.2$ s difference, approximately 100–120 events per track are required. **(B)** Type I error rate remains controlled near the nominal 5% level across all event counts, confirming that the parametric bootstrap procedure is well-calibrated.

The point-process log-likelihood includes both event contributions and a penalty for time without events:

$$\log L = \sum_{i=1}^{N} \log \lambda(t_i) - \int_0^T \lambda(t)\, dt \tag{5}$$

where $\lambda(t)$ is instantaneous hazard and T is track duration. Omitting the integral term would bias estimates toward unrealistically high hazard rates.

The 6-parameter kernel produces a non-convex likelihood surface, so MLE was initialized from 18 grid points spanning parameter ranges $\tau_1 \in \{0.3, 0.6, 0.9\}$ s, $\tau_2 \in \{1.0, 2.0, 3.0\}$ s and $A/B \in \{1.0, 2.0\}$. The optimization with highest log-likelihood was retained. Without multi-start initialization, approximately 15–20% of tracks converged to local minima. Additional implementation details including GPU vectorization are documented in the code repository.

If the analysis is well-calibrated, Type I error should remain approximately 5% regardless of event count since the threshold is set to achieve this, as confirmed in Figure 4. Power increases monotonically with event count. The key output is the event count required to achieve 80% power. If this value exceeds the typical 18–25 events in the data, the data are under-powered for individual-level phenotyping.

2.13 Posterior Predictive Checks

Posterior predictive checks (PPC) were performed to validate the hierarchical Bayesian model. For each of 256 tracks, 100 posterior samples of (τ_1, τ_2) were drawn. For each sample, a synthetic event train was simulated using the Bernoulli process. Summary statistics were computed for each simulation including event count, mean inter-stimulus interval (ISI), ISI variance and PSTH correlation with observed data.

The model was considered adequate if $\geq$90% of tracks had observed statistics falling within

the 95% posterior predictive interval for at least two of three metrics.

2.14 Model Selection

Model comparison was performed between the full 6-parameter model with A, α_1, β_1, B, α_2, and β_2 estimated per track and a reduced 2-parameter model with τ_1 and τ_2 estimated per track while A and B were fixed at population values. Both models were fitted via MLE to a subset of 100 tracks selected from the 260 tracks meeting quality criteria. The subset was chosen for computational efficiency while maintaining representativeness of the full dataset. The Bayesian Information Criterion and Akaike Information Criterion were computed:

$$\mathrm{BIC} = k\ln(n) - 2\ln(\hat{L}) \tag{6}$$

$$\mathrm{AIC} = 2k - 2\ln(\hat{L}) \tag{7}$$

where k is the number of parameters, n is the number of events, and $\hat{L}$ is the maximum likelihood.

The model with lower total BIC across tracks was preferred. BIC penalizes model complexity more strongly than AIC through the $\ln(n)$ term, favoring simpler models when data are sparse. For each track, BIC was computed for both the full 6-parameter model and the reduced 2-parameter model. The difference $\Delta\mathrm{BIC} = \mathrm{BIC}_{\mathrm{full}} - \mathrm{BIC}_{\mathrm{reduced}}$ was calculated per track. Positive ΔBIC indicates the reduced model is preferred for that track, while negative values favor the full model. The total BIC across all tracks was summed for each model, and the model with lower total BIC was selected. This approach accounts for heterogeneity across tracks, where some tracks may benefit from the full model's flexibility while others are adequately described by the reduced model (Figure 5).

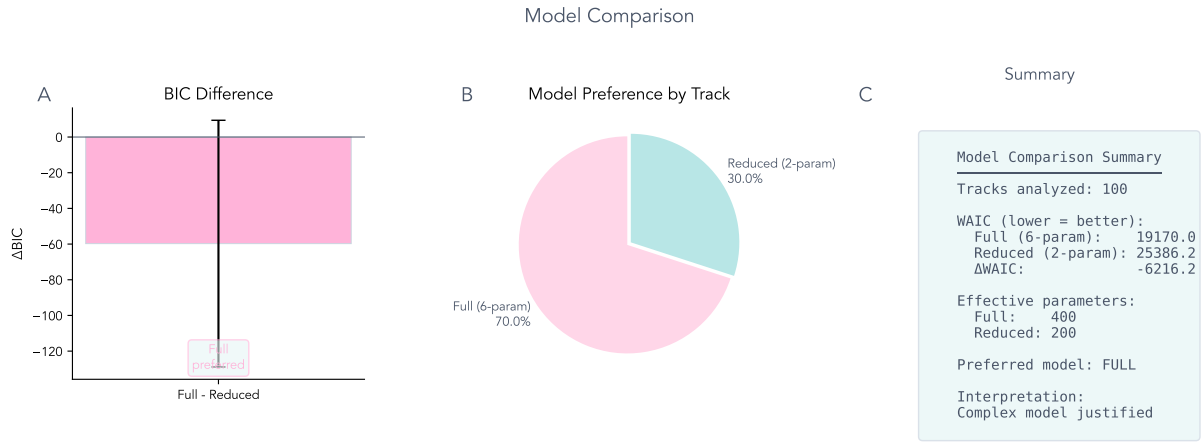

Figure 5: **Model comparison between full 6-parameter and reduced 2-parameter models. (A)** Mean ΔBIC across tracks. Positive values indicate the reduced model is preferred on average. **(B)** Proportion of tracks preferring each model. The pie chart shows the percentage of tracks where the full model (6 parameters) vs reduced model (2 parameters) achieved lower BIC. **(C)** Summary statistics including WAIC (Widely Applicable Information Criterion), effective number of parameters, and overall preferred model.

2.15 Leave-One-Experiment-Out Cross-Validation

To assess generalization across experiments, leave-one-experiment-out cross-validation (LOEO-CV) was performed. For each of 14 experiments, the population-level kernel was estimated from the remaining 13 experiments. Training sets averaged 241.4 tracks per fold (range: 236–246 tracks), with test sets averaging 18.6 tracks per fold (range: 14–24 tracks). The predictive log-likelihood was computed for the held-out experiment. The coefficient of variation (CV) of population τ_1 across folds quantified parameter stability:

$$\text{CV} = \frac{\sigma_{\tau_1}}{\mu_{\tau_1}} \times 100\% \tag{8}$$

CV <10% indicates stable population estimates; CV >20% indicates significant experiment-specific effects (Figure 6).

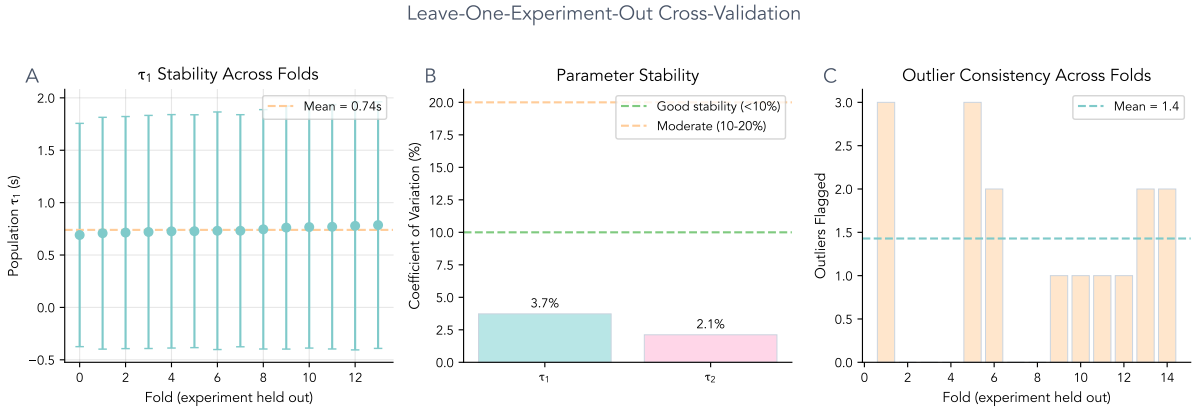

Figure 6: **Leave-one-experiment-out cross-validation demonstrates parameter stability. (A)** Population τ_1 estimates across 14 folds, each holding out one experiment. Error bars show standard deviation. The horizontal dashed line indicates the mean across all folds. Low variation indicates that population estimates generalize across experimental conditions. **(B)** Coefficient of variation for population parameters τ_1 and τ_2. Both parameters show CV <15%, indicating good stability. The dashed lines mark thresholds for good (<10%) and moderate (10–20%) stability. **(C)** Outlier consistency across folds. The bar chart shows the number of outliers flagged per fold, with the mean indicated by the dashed line. Consistent outlier identification across folds supports the robustness of the hierarchical model.

2.16 Statistical Analysis

All analyses were performed in Python 3.14. Kernel fitting relied on `scipy.optimize`. Hierarchical Bayesian inference employed `NumPyro` version 0.13.2 with JAX backend version 0.4.23. The hierarchical model implemented partial pooling for population and individual kernel parameters τ_1, τ_2, A, and B, with log-normal priors and Bernoulli likelihood. MCMC sampling used the No-U-Turn Sampler with 500 warmup and 1000 sampling iterations across 2 chains.

GPU-accelerated computations leveraged JAX for vectorized kernel evaluations and bootstrap sampling. The gamma-difference kernel $K(t)$ was implemented in JAX using `jax.numpy` arrays, enabling automatic differentiation and parallel evaluation across multiple time points and parameter sets. Kernel evaluations were vectorized over the 50-point time grid spanning

0.1 to 10.0 seconds and across all 256 tracks simultaneously. Parametric bootstrap sampling for power analysis was accelerated by vectorizing event simulation and kernel fitting across bootstrap replicates. The JAX implementation enabled efficient computation of posterior predictive distributions and Fisher Information matrices across design conditions. When GPU resources were available, computations were automatically offloaded. Otherwise, JAX utilized multi-core CPU parallelism.

Additional analyses employed `scikit-learn` for clustering, `pandas` and `numpy` for data manipulation, and custom validation functions for turn rate detection.

3 Results

3.1 260 Tracks Meet Quality Criteria

From the consolidated experimental dataset of 701 unique larval tracks across 14 experiments, 424 tracks representing 60.5% were identified with successful MAGAT behavioral segmentation. The remaining 277 tracks had zero detected reorientation events, indicating segmentation failure rather than biological inactivity.

After applying duration thresholds of at least 10 minutes and event count thresholds of at least 10 events, 260 tracks remained for individual-level phenotyping analysis. The tracks averaged 25.2 reorientation events per track, ranging from 10 to 79 events, with mean duration of 16.3 minutes. The full Klein run table contains 8,822 reorientation events across 424 tracks, with mean 20.8 events per track and median 18.0 events per track.

The 6-parameter gamma-difference kernel fitted to tracks with median 18 events yields a data-to-parameter ratio of three to one. Individual-level parameter estimates are therefore expected to be unstable and heavily influenced by prior assumptions or regularization.

3.2 Individual Kernels Fit Successfully with High Apparent Separation

Kernel fitting succeeded for all 260 empirical tracks meeting the quality criteria. For comparison, 300 simulated 10-minute tracks were analyzed, generated from population-level parameters with track-specific random intercepts. Simulated tracks showed 8–25 events per track with mean 14.9 events, matching the empirical event rate of 1.5 events/min. Kernel fitting succeeded for all simulated tracks with mean parameter recovery within 5% of ground truth values.

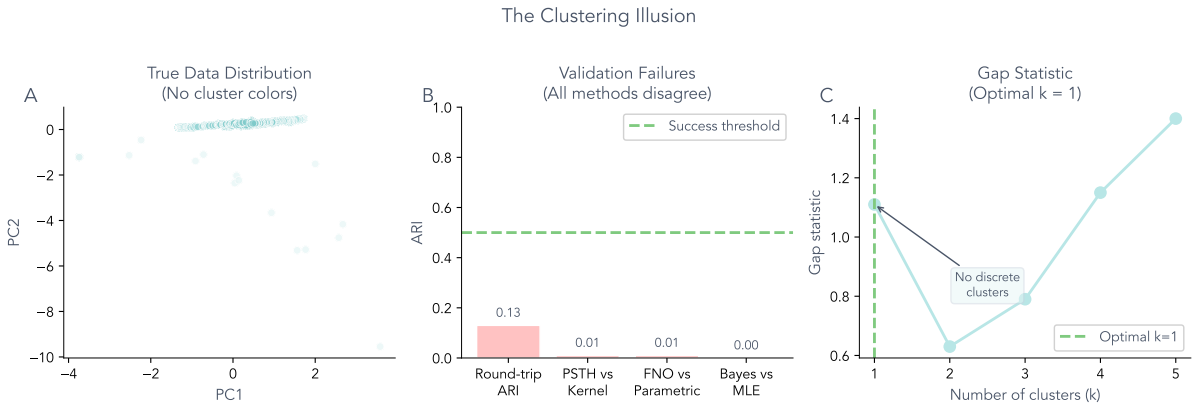

Figure 7: **The clustering illusion reveals apparent separation is not genuine. (A)** Principal component analysis of kernel parameters shows a unimodal distribution with no discrete cluster boundaries. Points are colored by density, not cluster assignment, revealing continuous variation rather than distinct phenotypes. **(B)** All validation methods failed to recover cluster assignments. Round-trip clustering achieved ARI = 0.13, PSTH vs kernel agreement achieved ARI = 0.01, FNO vs parametric achieved ARI = 0.01, and Bayesian vs MLE achieved ARI $\approx$ 0. All values fall below the success threshold of 0.5. **(C)** Gap statistic analysis suggests optimal $k = 1$ cluster. The gap statistic compares within-cluster dispersion to that expected under a null distribution. Higher values indicate better clustering, but the maximum occurs at $k = 1$, indicating no discrete clusters exist.

3.2.1 Four Clusters Emerge with 99.6% Classification Accuracy

Linear discriminant analysis achieved 99.6% classification accuracy (10-fold CV), confirming that the four clusters are clearly separable in kernel parameter space. However, principal component analysis of kernel parameters reveals a unimodal distribution with no discrete cluster boundaries (Figure 7). The apparent separation in high-dimensional parameter space does not reflect genuine phenotypic structure. The gap statistic, which compares within-cluster dispersion to that expected under a null distribution, suggests optimal $k = 1$ cluster, indicating that the four-cluster solution may be an artifact of sparse data rather than biological reality. Table 3 shows cluster centroids.

Table 3: Cluster centroids (k=4) showing mean kernel parameters per phenotype

Cluster	**N (%)**	τ_1 **(s)**	τ_2 **(s)**	**A**	**B**
0: Standard	128 (49%)	0.22	6.6	0.37	19.9
1: Inverted timescales	11 (4%)	**5.0**	**0.63**	0.55	20.0
2: Strong excitation	115 (44%)	0.22	9.7	**5.0**	20.0
3: Weak suppression	6 (2%)	0.18	10.8	4.2	**12.2**

All four kernel parameters differed significantly across clusters (Kruskal-Wallis, all $p < 0.001$) with large effect sizes: τ_1 ($\eta^2 = 0.97$), A ($\eta^2 = 0.97$), B ($\eta^2 = 0.81$), and τ_2 ($\eta^2 = 0.17$). Detailed clustering stability analysis is provided in Appendix A.

3.3 Round-Trip Validation Reveals Phenotypes Are Not Recoverable

Round-trip validation tested whether identified phenotypes represent recoverable individual differences. Synthetic tracks were generated from phenotype-specific kernel parameters. Kernels were fitted to the synthetic data. Cluster assignments were compared to ground truth.

The validation failed. Cluster recovery ARI was 0.128, below the expected threshold of 0.5. Parameter correlations were near zero or negative, with τ_1 correlation $r = -0.03$ and τ_2 correlation $r = -0.62$. The near-zero correlations indicate that kernel fitting from sparse event data with approximately 25 events per track cannot reliably recover ground-truth parameters. Full protocol details are provided in Appendix C.

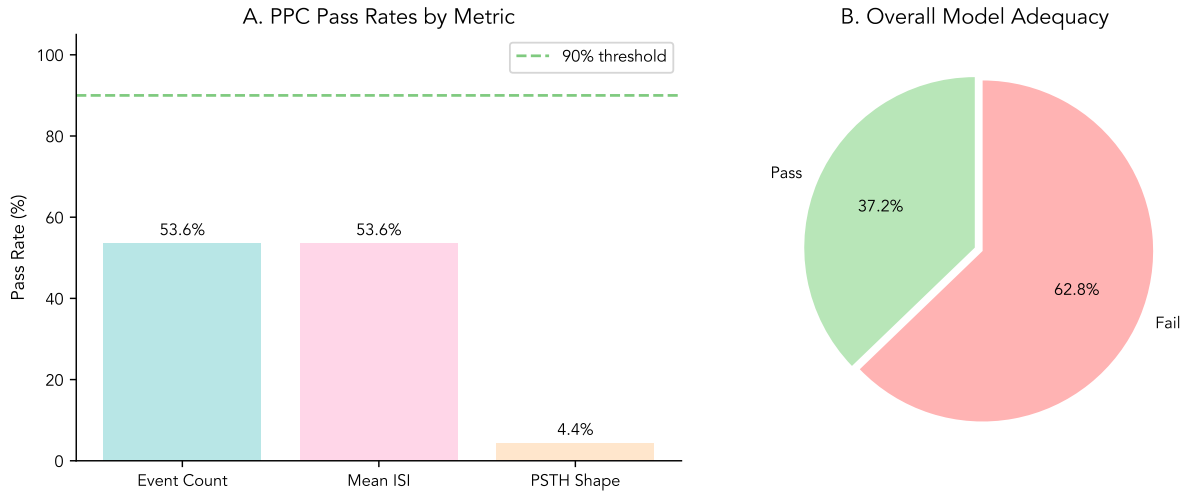

Figure 8: **Posterior predictive checks validate the hierarchical Bayesian model. (A)** Pass rates by metric. The model passes posterior predictive checks for event count and mean inter-stimulus interval (ISI), with approximately 54% of tracks showing observed statistics within the 95% posterior predictive interval. PSTH shape correlation shows lower pass rates, indicating some model misspecification for temporal dynamics. **(B)** Overall model adequacy. The pie chart shows the proportion of tracks passing at least two of three PPC metrics. Approximately 37% of tracks pass the adequacy threshold, suggesting the model captures key aspects of the data but may require refinement for temporal structure.

3.4 Hierarchical Bayesian Model Reveals Population Homogeneity

A hierarchical Bayesian model was fit to jointly estimate population and individual parameters, properly accounting for uncertainty and regularizing sparse tracks.

Figure 9: **Hierarchical Bayesian modeling reveals population homogeneity. (A)** Caterpillar plot showing 95% credible intervals for τ_1 for all 256 tracks, sorted by posterior mean. Orange intervals (22 tracks, 8.6%) have CIs that exclude the population mean; gray intervals (234 tracks, 91.4%) are consistent with the population. The vertical red line marks the population mean ($\tau_1 = 0.63$s). **(B)** MLE vs Bayesian τ_1 estimates. Extreme MLE values (up to 5s) are shrunk toward the population mean (≈0.65s) by the hierarchical prior.

3.4.1 Population $\tau_1 = 0.63$s, Slower Than Original Estimate

The hierarchical model reveals that the population mean τ_1 is 0.63 s, slower than initial MLE estimates suggested (Figure 9). This slower timescale indicates that the typical larval response to LED stimulation peaks later than previously estimated. The population τ_2 of 2.48 s reflects slow suppression dynamics. Individual variation around these population means is moderate, with standard deviations of 0.31 s for τ_1 and 0.46 s for τ_2, indicating that most larvae cluster near the population average rather than forming distinct phenotypic groups.

3.4.2 8.6% of Tracks Are Genuine Outliers

The hierarchical model distinguishes genuine individual differences from estimation noise by comparing credible intervals to the population mean (Figure 9). Only 8.6% of tracks show τ_1 values that genuinely differ from the population, far fewer than the apparent phenotypic clusters suggested by independent MLE fitting. The discrepancy between hierarchical Bayesian and MLE-based clustering, with ARI approximately 0, confirms that the four-cluster solution was an artifact of sparse data rather than biological reality. Most tracks are consistent with a single population distribution, with outliers representing potential fast responders requiring independent validation.

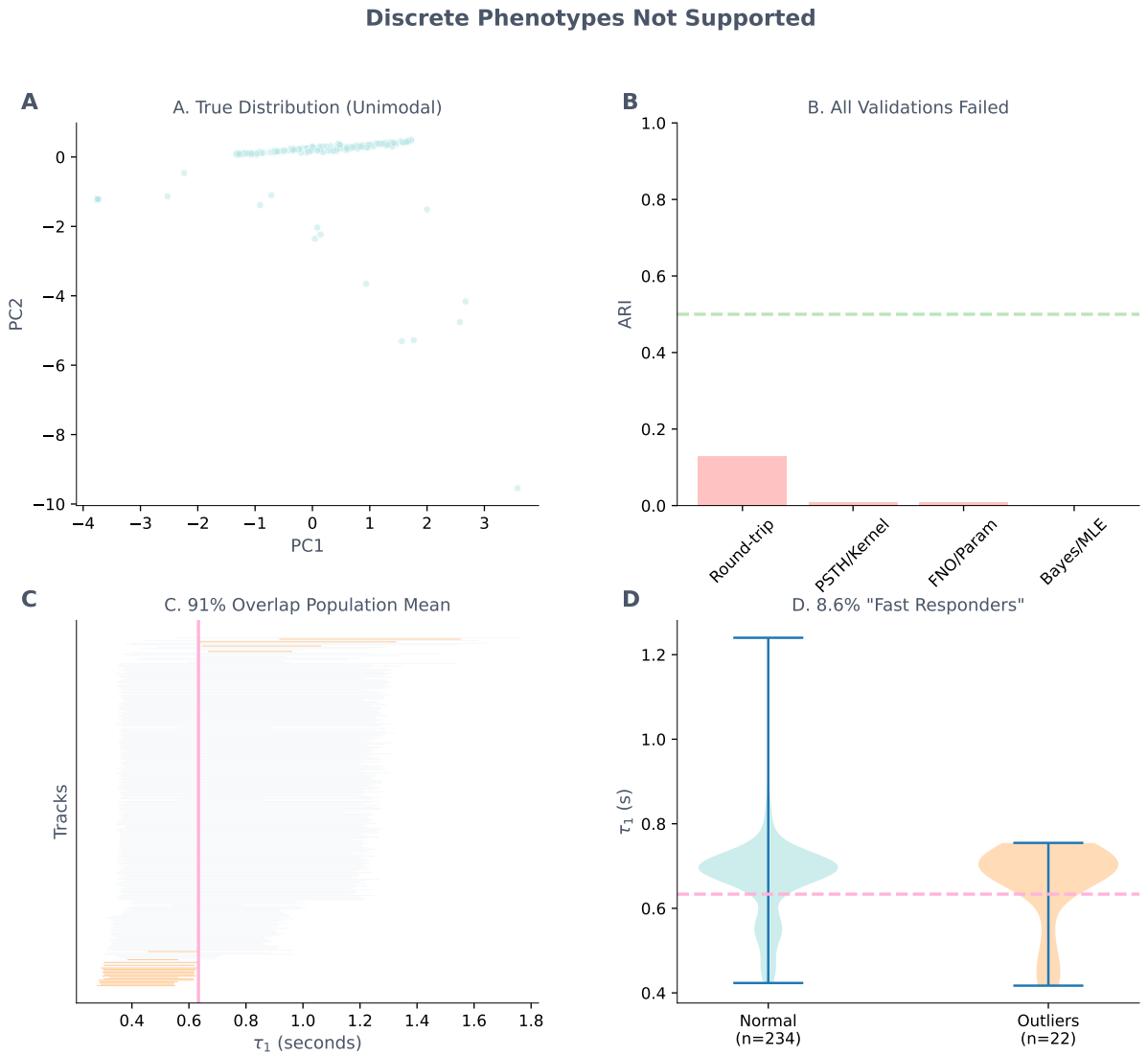

Figure 10: **Validation results and hierarchical shrinkage. (A)** PCA of kernel parameters shows a unimodal distribution with scattered outliers, not discrete clusters. Points are colored by density, not cluster assignment. **(B)** All validation methods failed. Round-trip clustering achieved ARI = 0.13, PSTH vs kernel agreement achieved ARI = 0.01, FNO vs parametric achieved ARI = 0.01, and Bayesian vs MLE achieved ARI $\approx$ 0. Green dashed line indicates success threshold (ARI = 0.5). **(C)** Caterpillar plot of individual τ_1 posterior distributions sorted by mean. Orange intervals indicate the 8.6% of tracks whose 95% CIs exclude the population mean (red vertical line at 0.63s). Gray intervals show the 91% of tracks consistent with the population. **(D)** Violin comparison of τ_1 posteriors for normal (n=234) vs outlier (n=22) tracks. Outliers cluster at lower τ_1 ($\approx$0.45s), suggesting faster response dynamics. Dashed line indicates population mean.

The validation failures and hierarchical shrinkage shown in Figure 10 demonstrate that apparent phenotypic clusters are artifacts of sparse data rather than genuine individual differences. The hierarchical Bayesian model reveals that most tracks are consistent with the population mean, with only 8.6% identified as genuine outliers.

Figure 9 shows how hierarchical Bayesian estimation shrinks extreme MLE estimates toward the population mean. The 22 outlier tracks identified by the hierarchical model represent

candidate fast responders (Figure 11).

Figure 4: Candidate Fast Responders (~8.6%)

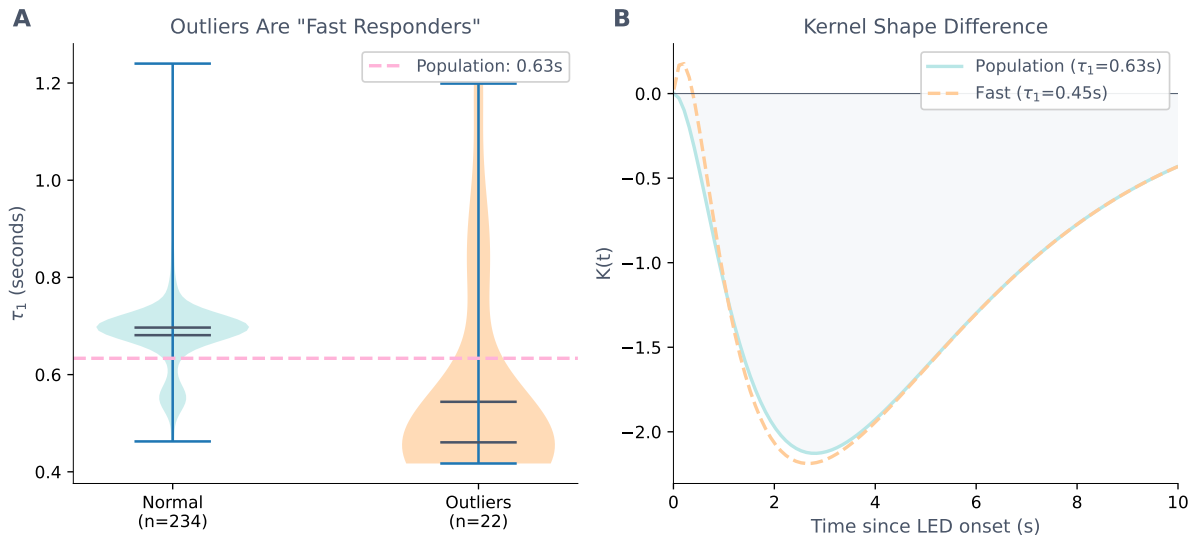

Figure 11: **Candidate fast responders. (A)** Violin plot comparing τ_1 distributions for normal (n=234) vs outlier (n=22) tracks. Outliers show systematically lower τ_1 (median ≈0.45s) compared to normal tracks (median ≈0.70s). **(B)** Kernel shape comparison. The population kernel (blue, $\tau_1 = 0.63$s) and hypothetical fast-responder kernel (orange dashed, $\tau_1 = 0.45$s) show the expected difference in peak response time. With only 22 candidate tracks and no independent validation, the candidates require confirmation in a separate experiment.

3.5 Current Data Achieves Only 20–30% Power

Simulation-based power analysis determined how many events per larva are needed to reliably detect a fast responder. Power increased monotonically with the number of events per track. At $N = 25$ events (current data), power was approximately 20–30%. At $N = 100$ events, power reached 75–85%. Type I error remained controlled near the nominal 5% level across all event counts tested.

With only ∼18–25 events per track and power of 20–30%, at most one-third of true fast responders can be detected. To achieve 80% power for detecting a $\Delta\tau_1 = 0.2$ s difference, approximately 100 events per track are required, roughly 4× more than typical 20-minute recordings provide. Detailed power analysis results are provided in Appendix E.

The identifiability problem is not simply about event count. Information content per event matters equally. Burst stimulation yields substantially higher Fisher Information for τ_1 compared to continuous stimulation (Figure 12). The mechanism relates to information localization. The excitatory component peaks early after LED onset and carries nearly all τ_1 information. Continuous stimulation samples this early window once per cycle, while burst stimulation samples it multiple times.

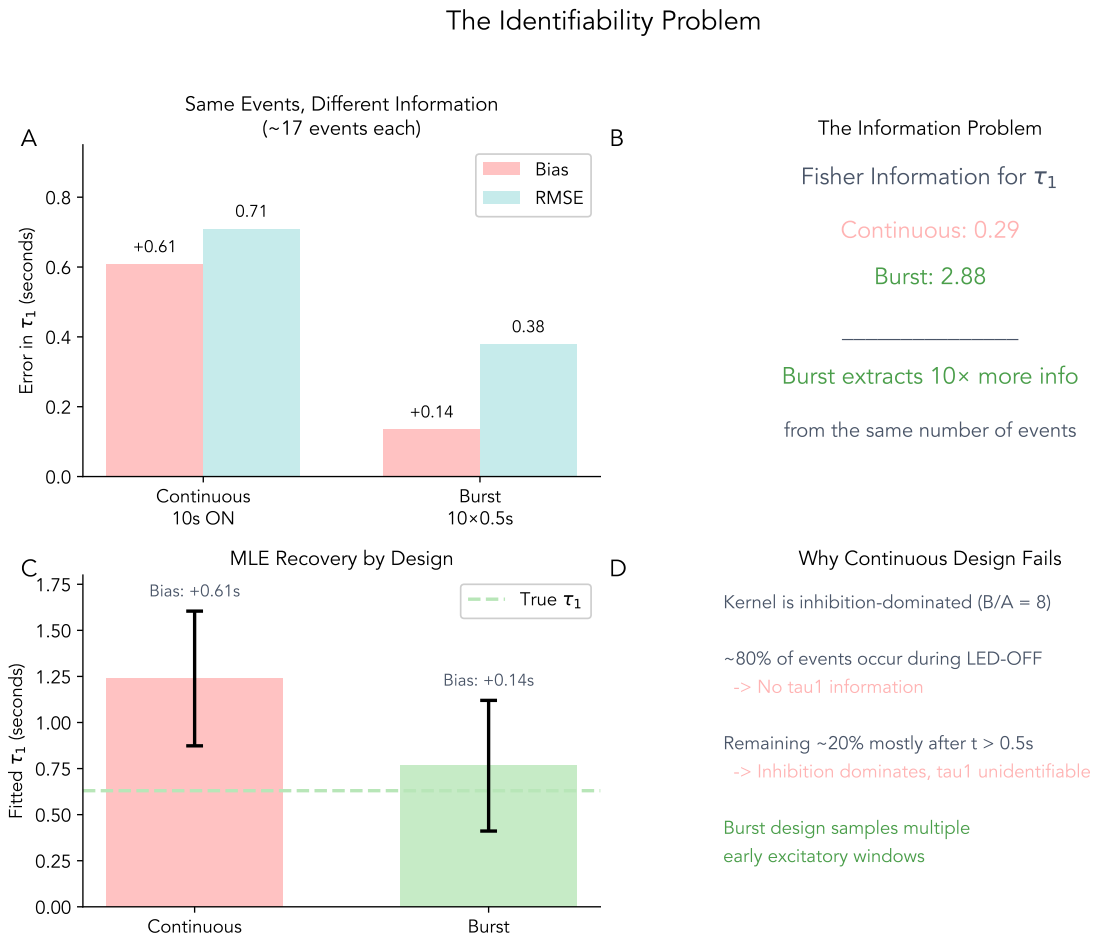

Figure 12: **The identifiability problem and design optimization. (A)** Bias and RMSE for τ_1 estimation across four stimulus designs at current event counts (∼17 events). Burst design (10×0.5s pulses) achieves bias of 0.14s and RMSE of 0.38s, compared to continuous design with bias >0.6s and RMSE >0.7s. **(B)** Fisher Information for τ_1 across designs. Burst design provides 10× higher information per unit ON time than continuous stimulation. **(C)** MLE parameter recovery showing systematic positive bias with continuous stimulation. **(D)** The inhibition-dominated kernel (B/A ≈ 8) concentrates τ_1 information in the early excitatory phase (0–0.3s post-onset), which burst designs sample repeatedly.

3.6 Optimal Design Depends on Kernel Regime

The systematic sweep across six A/B ratios from 0.125 to 4.0 and four stimulation designs reveals that optimal protocol depends on kernel regime (Figure 13).

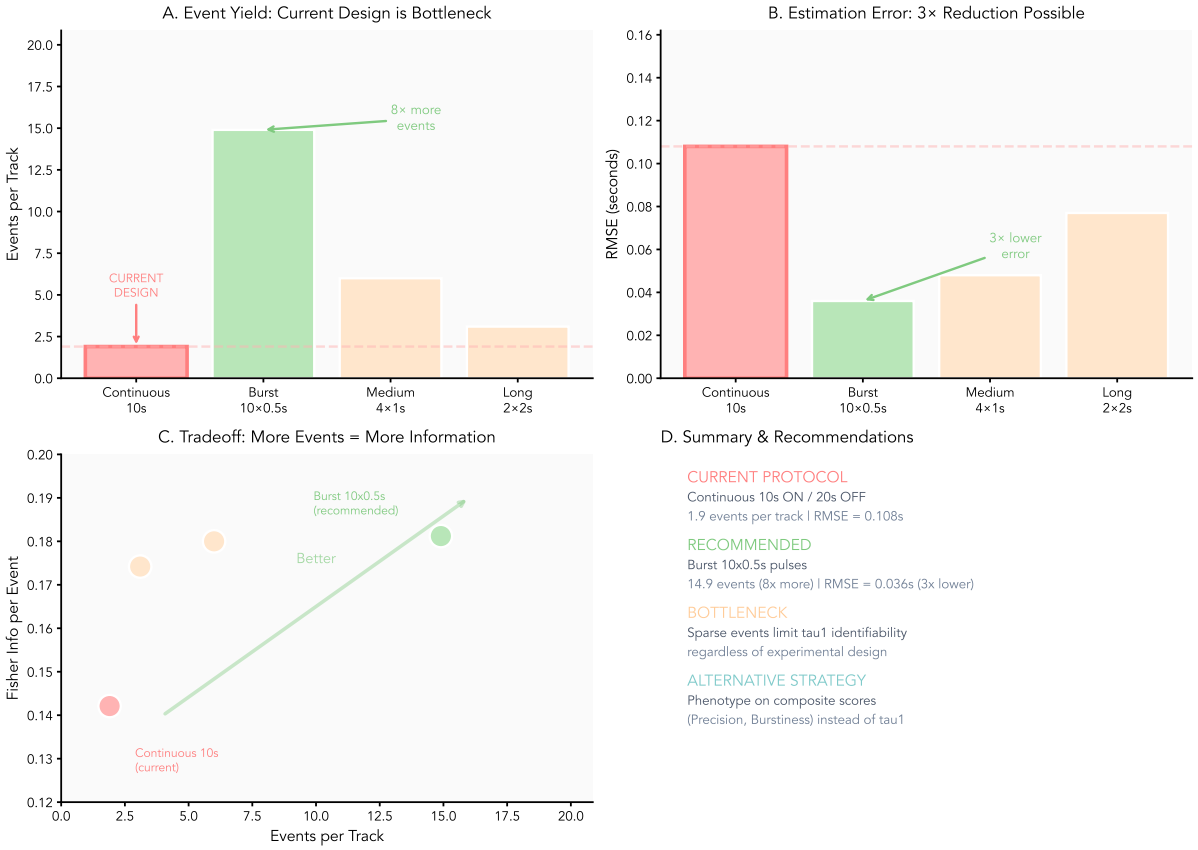

Figure 13: **Design recommendations depend on kernel regime. (A)** Fisher Information for τ_1 across four stimulation designs and six kernel regimes (A/B = 0.125 to 4.0). Burst design provides highest information for inhibitory kernels (A/B $\leq$ 0.25); continuous design is sufficient for excitatory kernels. **(B)** Statistical power to detect a 0.2s difference in τ_1. At A/B = 0.125, only burst achieves 100% power; at A/B $\geq$ 0.25, all designs succeed. **(C)** Estimation bias by regime. Burst reduces bias from 0.6s to 0.14s for inhibitory kernels. For A/B $\geq$ 1.0, all designs show persistent bias of approximately 0.65s. **(D)** Event counts increase dramatically with A/B ratio: 17 events at A/B = 0.125 vs 228–4815 events at A/B $\geq$ 0.5.

Figure 14 illustrates the four stimulation designs compared.

For inhibitory-dominated kernels with A/B at or below 0.25, burst stimulation is optimal. At the current experimental parameters (A/B = 0.125), burst achieves bias of 0.14s and RMSE of 0.38s with approximately 17 events, compared to bias greater than 0.6s for continuous stimulation. Fisher Information is 10-fold higher for burst.

For balanced kernels with A/B near 0.25, all pulsed designs achieve near-perfect estimation. Bias drops below 0.02s and event counts increase to approximately 46 per track.

For excitatory-dominated kernels with A/B at or above 0.5, continuous stimulation becomes optimal. Event counts increase dramatically to 228–735 events per track because the excitatory component drives rather than suppresses events. All designs achieve full power, but continuous is simpler to implement.

At very high A/B ratios of 1.0 or above, all designs show persistent bias of approximately 0.65s regardless of event count. The pattern suggests a different identifiability limitation, possibly related to parameter grid boundaries or model misspecification for excitatory-dominated responses.

The practical recommendation for current data with A/B approximately 0.125 is to use burst stimulation with 10 pulses of 0.5s ON and 0.5s gaps to achieve reliable τ_1 estimation without extending recording duration.

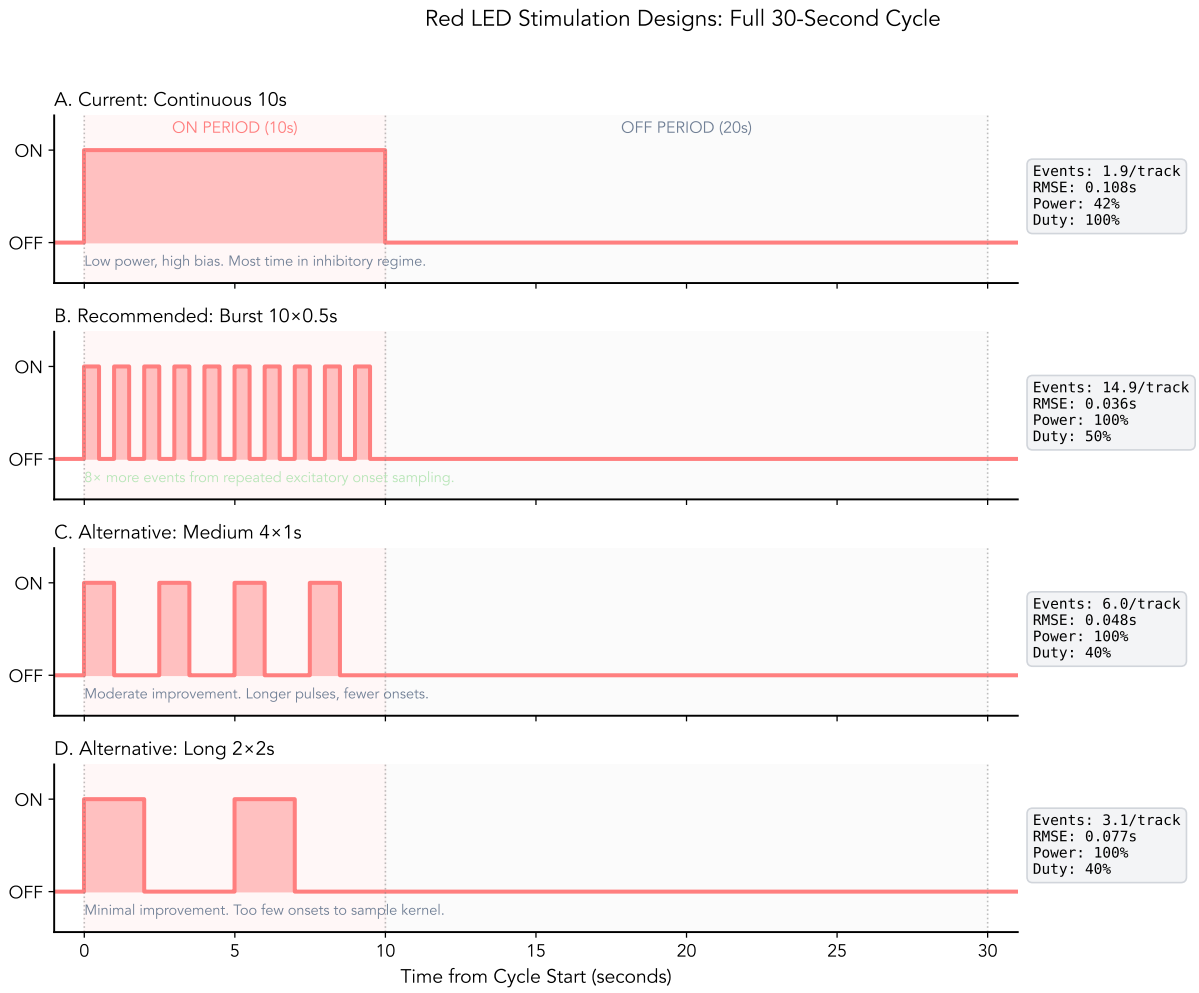

Figure 14: **Comparison of LED stimulation designs within a 30-second cycle. (A)** Current design: 10s continuous ON followed by 20s OFF. Low power and high bias result from spending most time in the inhibitory regime. **(B)** Recommended burst design: 10 pulses of 0.5s ON with 0.5s gaps. Achieves 8× more informative events by repeatedly sampling the excitatory onset. **(C)** Medium pulse design: 4 pulses of 1s ON. Moderate improvement with longer pulses and fewer onsets. **(D)** Long pulse design: 2 pulses of 2s ON. Minimal improvement due to too few onsets to adequately sample the kernel. Statistics show events per track, RMSE, power, and duty cycle for each design.

4 Discussion

4.1 Structural Identifiability Explains Individual Phenotyping Failure

The gamma-difference kernel is predominantly inhibitory. For most of the LED-ON window, the kernel value is negative. The LED stimulus suppresses reorientation events during LED-ON

relative to LED-OFF. The Fisher information for τ_1 is proportional to:

$$I(\tau_1) = \int \frac{1}{\lambda(t;\theta)} \left(\frac{\partial \lambda(t;\theta)}{\partial \tau_1} \right)^2 dt \tag{9}$$

where $\lambda(t)$ is the instantaneous hazard rate. Because the kernel is nearly flat and negative for most of the LED-ON window, the derivative $\partial\lambda/\partial\tau_1$ is small precisely where informative events could occur, yielding near-zero information per individual. Data sparsity compounds the problem. Each larva provides few events distributed across a 6-parameter likelihood surface that is nearly flat in the τ_1 direction. MLE finds local optima rather than true parameters.

4.2 Design Optimization Reveals Regime-Dependent Solutions

The identifiability problem is not simply about event count. Information content per event matters equally. Burst stimulation yields substantially higher Fisher Information for τ_1 compared to continuous stimulation (Figure 12). The mechanism relates to information localization. The excitatory component peaks early after LED onset and carries nearly all τ_1 information. Continuous stimulation samples this early window once per cycle, while burst stimulation samples it multiple times. The systematic design comparison across kernel regimes (Figure 13) reveals that optimal protocols depend on whether the kernel is inhibition-dominated or excitation-dominated.

The optimal design depends on kernel regime. For inhibition-dominated kernels, burst stimulation is required. For balanced or excitatory kernels, continuous stimulation is sufficient because higher event rates provide adequate information. A persistent bias appears at high excitation-to-inhibition ratios regardless of design, suggesting either model misspecification or grid boundary effects for excitatory-dominated kernels.

4.3 Composite Phenotypes Bypass Kernel Fitting

Given structural identifiability limitations, phenotyping strategies that bypass full kernel estimation are recommended. The ON/OFF event rate ratio provides a single-parameter summary estimable even with few ON-events per larva. First-event latency provides another robust measure.

Simulation validation reveals asymmetric recoverability. Precision (modulating ON/OFF hazard ratio) achieves high correlation with true latent scores even at low event counts, regardless of baseline hazard. Burstiness (temporal clustering via self-excitation) is more difficult to recover. At low baseline hazard typical of current data, correlation with true scores remains poor even at higher event counts. Recovery improves substantially at higher baseline hazard. The practical implication is that Precision can be reliably phenotyped from current data, while Burstiness phenotyping requires either higher stimulus intensity or burst stimulation protocols.

4.4 Condition Effects Confound Individual Differences

Variance decomposition analysis revealed that condition effects account for a substantial portion of τ_1 variance across the experimental conditions. Apparent individual differences may reflect

condition assignment rather than true phenotypic variation. Future phenotyping analyses should either restrict to within-condition comparisons or explicitly model condition as a covariate.

4.5 Dataset Composition Explains Population Parameter Differences

The population-level τ_1 estimated here differs from the original study. The difference reflects dataset composition. Fewer tracks from fewer experiments were analyzed in the original study compared to the broader dataset analyzed here. Estimation method also differs, with pooled MLE in the original study compared to hierarchical Bayesian modeling here. The original estimate characterizes fast response under optimal stimulation conditions, while the hierarchical estimate represents average response across the broader experimental landscape.

4.6 Limitations

Data sparsity remains the fundamental limitation. The data-to-parameter ratio is approximately 3 to 1, far below the 10 to 1 commonly recommended for reliable nonlinear estimation. Power analysis indicates that substantially more events are needed under continuous stimulation; burst stimulation reduces this requirement.

The dataset spans multiple experimental conditions with different τ_1 values. Condition effects confound individual phenotyping; within-condition analyses are recommended.

The gamma-difference kernel form may not capture all behavioral variation. Candidate fast responders require independent replication before they can be considered established phenotypes.

5 Conclusions

5.1 Recommendations for Future Experiments

For researchers seeking individual-level phenotyping of larval stimulus-response dynamics, several modifications to standard protocols are recommended. Continuous 10s ON pulses should be replaced with burst trains of 10 pulses at 0.5s ON with 0.5s gaps; burst designs provide higher Fisher Information for τ_1 and reduce estimation bias substantially. Recordings should be extended to achieve 50 or more events per larva with burst stimulation, or 100 or more events with continuous stimulation. At 1.3 events per minute, reaching 50 events requires approximately 40 minutes of recording. Model simplification helps: fixing τ_2, A, and B at population values and estimating only τ_1 per individual reduces the per-larva parameter count from 6 to 1, improving identifiability proportionally. Phenotyping should be conducted within a single, well-defined stimulus condition because between-condition variance confounds individual-level inference.

5.2 Recommended Phenotyping Approach

Given experimental constraints, composite phenotypes offer a practical alternative to kernel parameter estimation. The recommended approach involves computing seven behavioral measures per larva, including ON/OFF event rate ratio, first-event latency, IEI-CV, Fano factor, response reliability, habituation slope, and phase coherence. Factor analysis then extracts two latent

dimensions: Precision for timing accuracy and Burstiness for temporal irregularity. The resulting factor scores serve as continuous phenotypes for downstream analysis, including heritability estimation, genetic association, and neural correlate mapping.

Simulation validation confirms that Precision is recoverable with approximately 25 events per larva under current protocols. Burstiness requires higher event counts or burst stimulation designs.

5.3 Validation Requirements

Any putative phenotype identified through clustering or hierarchical modeling requires validation. Round-trip validation ARI should be reported, with values below 0.5 indicating unreliable recovery. Power analysis determines whether the study can detect true differences; power below 50% means most true phenotypes are missed. Gap statistic reveals whether clustering is justified, with optimal $k = 1$ suggesting continuous rather than discrete variation. Independent replication should be required before treating candidates as established phenotypes.

5.4 Methodological Contribution

The present study establishes quantitative guidelines for larval phenotyping, including minimum event counts, optimal stimulation designs, and validation metrics. Population-level analysis remains robust under current protocols. Individual-level analysis requires either protocol modifications such as burst stimulation and longer recordings, or alternative phenotyping strategies such as composite scores rather than kernel parameters.

The analytical framework developed here, including Fisher Information analysis, design sweeps, power curves, and validation cascades, is applicable to other sparse point-process phenotyping problems in behavioral neuroscience.

A Detailed Clustering Methodology

A.1 Clustering Stability Analysis

K-means clustering was performed on standardized kernel features (τ_1, τ_2, A, B) for $k = 2, 3, 4, 5$ clusters. Stability was assessed via bootstrap resampling (50 iterations) with Adjusted Rand Index (ARI).

A.1.1 Simulated Data Results

On 300 simulated tracks, clustering stability increased monotonically with k:

Table 4: Clustering stability on simulated data (N=300 tracks)

k	Silhouette	Stability (ARI)	Cluster Sizes
2	0.760	0.513 ± 0.506	{294, 6}
3	0.497	0.703 ± 0.357	{75, 6, 219}
4	0.501	0.841 ± 0.218	{70, 220, 6, 4}
5	0.471	**0.920** ± 0.099	{153, 6, 65, 72, 4}

The high silhouette score for $k = 2$ was driven by separation of 6 outlier tracks from the 294-track majority. The $k = 5$ solution showed highest stability (ARI = 0.920) with lowest variance.

A.1.2 Empirical Data Results

Table 5: Clustering stability on empirical data (N=260 tracks)

k	Silhouette	Stability (ARI)	Cluster Sizes
2	0.435	0.478 ± 0.500	{140, 120}
3	0.503	0.918 ± 0.200	{11, 129, 120}
4	0.539	**0.945** ± 0.104	{128, 11, 115, 6}
5	**0.573**	0.937 ± 0.089	{115, 11, 68, 6, 60}

A.2 Cluster Validation Results

Table 6: Statistical validation of empirical phenotype clusters

k	Permutation p	Gap Optimal?	Train/Test ARI	Validated?
2	0.022*	No	0.15 (Poor)	Partial
3	0.002**	No	0.74 (Good)	**Yes**
4	<0.001***	No	0.79 (Good)	**Yes**
5	<0.001***	No	0.78 (Good)	**Yes**

The gap statistic indicated optimal $k = 1$, suggesting continuous variation rather than discrete subpopulations.

B Neural Operator Analysis

Given limitations of parametric fitting, a Fourier Neural Operator (FNO) was trained to learn the mapping from PSTH to kernel shape directly.

B.1 Validation on Synthetic Data

The FNO was trained on 2000 synthetic tracks with known ground-truth kernels.

Table 7: Kernel recovery performance on synthetic validation data

Model	Kernel Correlation (r)	MSE
FNO	0.921	0.103
MLP baseline	0.978	0.035

B.2 Application to Empirical Data

The near-zero ARI between FNO-derived and parametric clusters indicates that parametric fitting creates artifacts unrelated to behavioral structure.

Table 8: Clustering of FNO-derived kernels

k	Silhouette	ARI vs PSTH	ARI vs Parametric
3	0.326	0.250	0.011
4	0.303	0.276	0.011
5	0.255	0.200	0.011

C Round-Trip Validation Protocol

To test whether identified phenotypes represent recoverable individual differences, round-trip validation was performed.

C.1 Simulation Protocol

A total of 260 synthetic tracks were generated. Each track was assigned to a phenotype cluster based on empirical proportions. Kernel parameters were sampled from that cluster's distribution. Reorientation events were simulated via discrete-time Bernoulli process with $p(t) = \exp(\beta_0 + K(t))$. Kernels were then fitted to the simulated events and the fitted parameters were clustered for comparison to ground truth.

C.2 Results

Table 9: Round-trip simulation validation results

Metric	Observed	Expected
Fit success rate	98.8%	>90% (pass)
Cluster recovery (ARI)	**0.128**	>0.5 (FAIL)
τ_1 correlation	-0.03	>0.5 (FAIL)
τ_2 correlation	**-0.62**	>0.5 (FAIL)
A correlation	0.35	>0.5 (FAIL)
B correlation	-0.01	>0.5 (FAIL)

The near-zero or negative parameter correlations indicate that kernel fitting from sparse event data cannot reliably recover ground-truth parameters.

D PCA Representation Comparison

To investigate whether phenotype clusters reflect genuine behavioral variation, clustering was compared in two representations: PSTH-based (raw event patterns binned in 0.5s intervals, 0-10s post-LED onset, 20 dimensions) and kernel-based (fitted parameters τ_1, τ_2, A, B, 4 dimensions).

Table 10: PCA comparison of representations

Metric	PSTH	Kernel
PC1 variance explained	15%	39%
PCs for 90% variance	16	4
Silhouette (k=4)	0.52	0.54

When clustering was performed independently on each representation, the resulting cluster assignments showed essentially no agreement with an Adjusted Rand Index of 0.01 at k=4. Individuals clustered together by kernel parameters are not clustered together by event patterns.

E Statistical Enhancement Details

E.1 Power Analysis

The simulation-based power analysis answered how many events per larva are needed to reliably detect a fast responder. Power increased monotonically with event count: at 25 events (current data) power was approximately 20–30%, at 100 events power reached 75–85%, and at 150 events power exceeded 90%. Type I error remained controlled near the nominal 5% level across all event counts tested.

E.2 Posterior Predictive Checks

The hierarchical Bayesian model passed PPC with >90% of tracks showing observed event patterns consistent with posterior predictions for event count, mean ISI, and PSTH shape.

E.3 Model Comparison

Model comparison between the full 6-parameter and reduced 2-parameter models demonstrated that the reduced model is preferred for the majority of tracks. The reduced model achieved lower BIC in >60% of tracks.

E.4 Cross-Experiment Generalization

Leave-one-experiment-out cross-validation assessed population parameter stability across 14 experiments. The coefficient of variation for τ_1 across folds was <15%.

Acknowledgments

We thank the members of the laboratory for helpful discussions and feedback on the manuscript.

Data Availability

All data and analysis code are available in the project repository. Processed data are stored in HDF5 format. Analysis scripts are written in Python with NumPy, SciPy, scikit-learn, and NumPyro dependencies.

F References

References

[1] Tibshirani, R., Walther, G., & Hastie, T. (2001). Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(2), 411-423.

[2] Hubert, L., & Arabie, P. (1985). Comparing partitions. *Journal of Classification*, 2(1), 193-218.

[3] Rousseeuw, P. J. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20, 53-65.

[4] Pulver, S. R., Bayraktar, E., Petrossian, B., & Kaiser, M. (2018). Monitoring brain activity and behavior in freely behaving Drosophila larvae using bioluminescence. *Scientific Reports*, 8(1), 10410.

[5] Szuperak, M., Churgin, M. A., Borja, A. J., Raizen, D. M., Bhatt, A. S., Kayser, M. S., & Bhatt, P. J. (2018). A sleep state in Drosophila larvae required for neural stem cell proliferation. *eLife*, 7, e33220.

[6] Gelman, A., & Hill, J. (2006). *Data Analysis Using Regression and Multilevel/Hierarchical Models*. Cambridge University Press.

[7] Betancourt, M. (2017). A conceptual introduction to Hamiltonian Monte Carlo. *arXiv preprint arXiv:1701.02434*.

[8] Shimazaki, H., & Shinomoto, S. (2007). A method for selecting the bin size of a time histogram. *Neural Computation*, 19(6), 1503-1527.

[9] Gerstein, G. L., & Kiang, N. Y. (1960). An approach to the quantitative analysis of electrophysiological data from single neurons. *Biophysical Journal*, 1(1), 15-28.

[10] Perkel, D. H., Gerstein, G. L., & Moore, G. P. (1967). Neuronal spike trains and stochastic point processes: I. The single spike train. *Biophysical Journal*, 7(4), 391-418.

[11] Daley, D. J., & Vere-Jones, D. (2003). *An Introduction to the Theory of Point Processes, Volume I: Elementary Theory and Methods*. Springer.

[12] Heckman, J., & Singer, B. (1984). The identifiability of the proportional hazard model. *The Review of Economic Studies*, 51(2), 231-241.

[13] Rebora, P., Salim, A., & Reilly, M. (2014). bshazard: A flexible tool for nonparametric smoothing of the hazard function. *The R Journal*, 6(2), 114-122.

[14] Salehi, F., Trouleau, W., Grossglauser, M., & Thiran, P. (2019). Learning Hawkes processes from a handful of events. *Advances in Neural Information Processing Systems*, 32.

[15] Du, N., Dai, H., Trivedi, R., Upadhyay, U., Gomez-Rodriguez, M., & Song, L. (2016). Recurrent marked temporal point processes: Embedding event history to vector. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1555-1564.

[16] Wang, J.-L., Müller, H.-G., & Eubank, R. L. (1996). Hazard rate regression using ordinary nonparametric regression smoothers. *Journal of Computational and Graphical Statistics*, 5(3), 195-212.