

# HP-GSBS: Accelerating greedy state boundary search with reusable interval statistics

Yanzhe Zhao

[justinzhaoyz@outlook.com](mailto:justinzhaoyz@outlook.com)

Tianjin University

Yongzhang Ji

Southwest University

---

## Research Article

**Keywords:** neural state segmentation, GSBS, neuroinformatics, computational efficiency, EEG, fMRI

**Posted Date:** October 26th, 2026

**DOI:** <https://doi.org/10.21203/rs.3.rs-11477729/v1>

**License:**   This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

**Additional Declarations:** No competing interests reported.

---

# HP-GSBS: Accelerating greedy state boundary search with reusable interval statistics

Yanzhe Zhao<sup>1\*</sup> and Yongzhang Ji<sup>2</sup>

<sup>1</sup>School of Future Technology, Tianjin University, Tianjin, China.

<sup>2</sup>Faculty of Psychology, Southwest University, Chongqing, China.

\*Corresponding author(s). E-mail(s): [justinzhaoyz@outlook.com](mailto:justinzhaoyz@outlook.com);

## Abstract

Greedy State Boundary Search (GSBS) identifies transitions between relatively stable neural activity patterns in continuous recordings. Repeated template reconstruction and correlation evaluation make analyses across recordings and parameter settings computationally expensive. We present HP-GSBS, an open-source implementation that expresses the matching objective as additive interval contributions and computes t-distance from reusable correlation statistics. Streaming candidate reduction, sequential local refinement, and active-block reuse complement CPU, PyTorch CUDA, and Triton execution paths. Across seven controlled workloads with two input realizations each, median same-input speedups range from 154 to 1,730. All 168 optimized executions reproduce complete reference boundary histories and selected state counts, with a maximum absolute t-distance difference of  $1.88 \times 10^{-11}$ . Four publicly supplied BBBB EEG analysis matrices provide complementary cross-backend validation. A naturalistic fMRI application segments 4,230 angular-gyrus voxels in a group response and 16 participants; the 17 CPU fits total 213.10 seconds. Mean pairwise boundary alignment exceeds a circular-shift baseline, illustrating cohort-level use of the saved segmentations. Versioned results, recording-level parallelism, and optional low-memory execution support varied analysis scales. HP-GSBS connects an explicit mathematical reformulation of repeated computation to efficient and reproducible neural-state analysis.

**Keywords:** neural state segmentation, GSBS, neuroinformatics, computational efficiency, EEG, fMRI

Yanzhe Zhao: [ORCID 0009-0003-5342-6611](https://orcid.org/0009-0003-5342-6611)

## 1 Introduction

Continuous experience contains structure at multiple temporal scales. During a film, conversation, or familiar activity, sensory information evolves while the context of an ongoing event can remain relatively stable. Multivariate neural activity offers a way to examine this organization: a neural state corresponds to a temporally contiguous interval with a relatively stable spatial activity pattern, and a state boundary marks a transition between patterns. Naturalistic fMRI studies have linked such transitions to narrative structure and memory, motivating analyses that estimate event organization directly from recorded activity (Baldassano et al., 2017). EEG provides complementary temporal sampling during sustained experience, including educational video viewing (Madsen, Kuppaa, & Parra, 2026). The resulting

state sequence provides boundary locations, interval durations, and spatial patterns that can be related to stimulus timing or compared across participants. For naturalistic recordings, retaining this sequence at several state counts is particularly useful because a single stimulus contains transitions at different temporal scales. Computational access to the full search history therefore supports both an initial segmentation and later inspection of its temporal resolution. Together, these settings motivate computational tools that can characterize state sequences across recordings, participants, and spatial feature sets.

Greedy State Boundary Search (GSBS) estimates neural-state boundaries through iterative segmentation of a time-by-feature matrix (Geerligs, van Gerven, & Güçlü, 2021). Candidate boundaries are evaluated by the correspondence between observed spatial patterns and the mean pattern of their assigned state. A t-distance criterion contrasts correlations within

states with correlations between adjacent states and provides a basis for selecting the number of states. Statewise detection extends the search to pairs of boundaries, improving its ability to represent intervals embedded within longer states (Geerligs et al., 2022). GSBS has consequently supported analyses of cortical temporal organization, including a partially nested hierarchy of neural states across brain regions. Its data-driven boundary locations and explicit state-count curve are useful outputs for examining both common and participant-specific temporal structure.

The cost of these analyses grows with the number of recordings, feature sets, and parameter settings. A study that evaluates several brain regions in each participant requires many independent segmentations, while sensitivity analyses repeat them across state-count limits and refinement settings. Within each fit, the original implementation repeatedly reconstructs state templates and recalculates correlations for candidate partitions. A proposed boundary changes a small part of the partition, yet much of the corresponding score depends on intervals whose observations and state means remain fixed. Double-boundary search adds a quadratic set of candidate pairs within a state. The t-distance calculation introduces another source of repeated work by revisiting the same time-pair correlations after successive partition changes.

These dependencies suggest a computational formulation based on reusable statistics. The matching objective can be expressed as a sum of interval contributions derived from temporal prefixes. Each insertion then replaces the contribution of its parent interval with those of its child intervals. Similarly, t-distance depends on counts, sums, and squared sums of eligible correlations, which can be recovered from spatially indexed prefix tables. Streaming candidate reduction, sequential local refinement, and reuse of unchanged block winners complement these representations. This organization concentrates arithmetic in regular precomputation and makes CPU and GPU execution routes available through a common search controller.

Here we present HP-GSBS, an open-source implementation of this formulation (Zhao, 2026). We derive the interval and t-distance statistics, describe their integration with the original search decisions, and evaluate numerical agreement and complete-fit performance. Controlled workloads establish same-input runtime comparisons across three optimized backends. Four publicly supplied BBBD EEG analysis matrices connect saved boundary histories to real electrophysiological inputs. A naturalistic fMRI application extends the analysis to 4,230 angular-gyrus voxels in 16 participants and examines intersubject boundary alignment. These complementary evaluations connect the computational method to reproducible neural-state analysis at both recording and cohort levels.

## 2 Methods

### 2.1 GSBS formulation and computational structure

Let  $X \in \mathbb{R}^{T \times P}$  contain  $T$  observations of a spatial pattern with  $P$  features. A partition  $\mathcal{S}$  comprises contiguous half-open intervals  $[a, b)$ , each with mean pattern  $m_{ab} = (b - a)^{-1} \sum_{t=a}^{b-1} x_t$ . GSBS evaluates the mean spatial Pearson correlation between each observation and its assigned state mean:

$$J(\mathcal{S}) = \frac{1}{T} \sum_{[a,b) \in \mathcal{S}} \sum_{t=a}^{b-1} \text{corr}(x_t, m_{ab}). \quad (1)$$

The search iteratively proposes a single boundary or, with statewise detection, two boundaries within an existing state. The best matching-score proposal of each type is evaluated using t-distance; accepted boundaries undergo sequential refinement. Each accepted partition contributes a point to a state-count curve, making the quality of successive segmentations available for inspection. The selected state count maximizes the resulting t-distance curve (Geerligs et al., 2022, 2021).

This procedure has two distinct scoring tasks. Candidate search requires correlations with state templates. Partition comparison requires distributions of correlations between time points. Rebuilding templates for each candidate repeats operations over unchanged intervals; repeatedly constructing correlation distributions revisits unchanged pairs. HP-GSBS represents the first task with additive interval scores and the second with correlation sufficient statistics (Fig. 1). Both representations preserve the original candidate domain, proposal ordering, and sequential partition updates. Table 1 summarizes the principal computational costs.

### 2.2 Additive interval statistics

Define the spatial centering operator  $H = I - \mathbf{1}\mathbf{1}^\top/P$ , the centered observation  $c_t = Hx_t$ , and its sample-standardized pattern  $z_t = \sqrt{P-1} c_t / \|c_t\|_2$ . Temporal prefixes collect the centered and standardized observations:

$$A_b = \sum_{t < b} c_t, \quad B_b = \sum_{t < b} z_t. \quad (2)$$

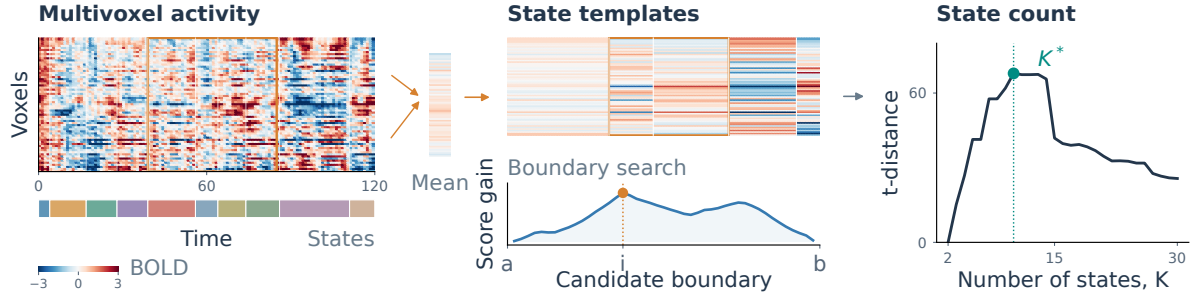
For  $\Delta A_{ab} = A_b - A_a$  and  $\Delta B_{ab} = B_b - B_a$ , centering the interval mean gives  $Hm_{ab} = \Delta A_{ab}/(b - a)$ . Its spatial normalization cancels the interval-length factor, yielding

$$J(\mathcal{S}) = \sum_{[a,b) \in \mathcal{S}} F(a, b), \quad (3)$$

$$F(a, b) = \frac{\Delta B_{ab}^\top \Delta A_{ab}}{T \sqrt{P-1} \|\Delta A_{ab}\|_2}.$$

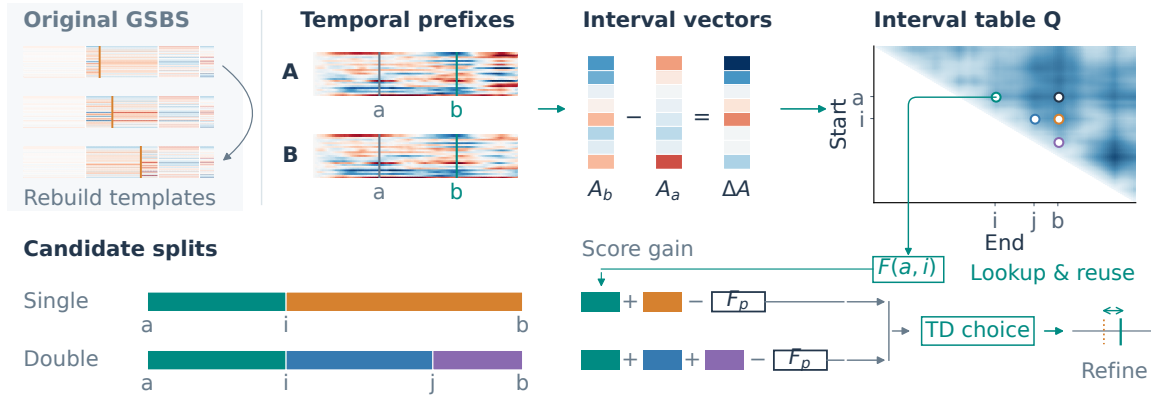
### a Neural state segmentation

GSBS



### b Interval score reuse

HP-GSBS · FULL-Q



### c Prefix-based t-distance

HP-GSBS · PREFIX2D

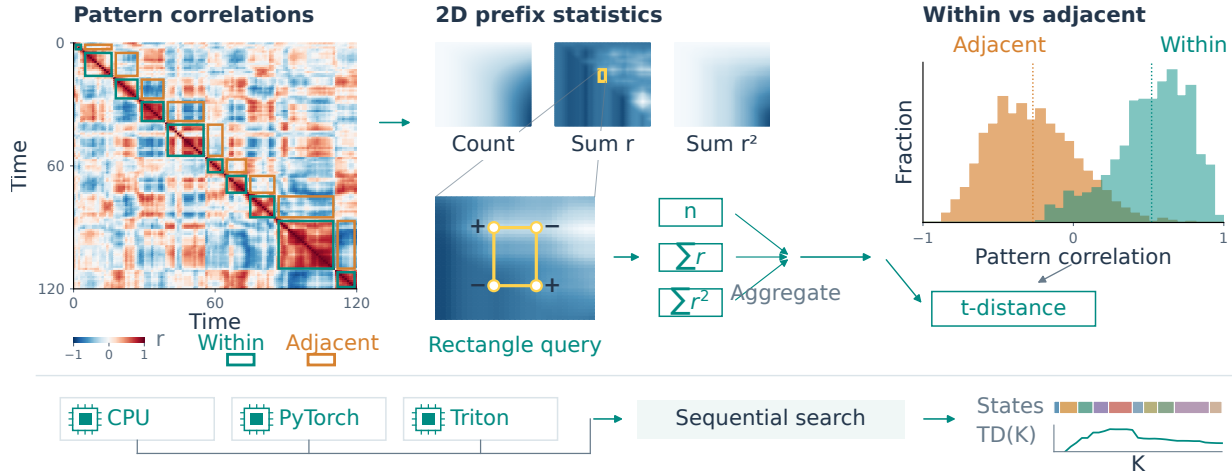


**Fig. 1** Neural-state segmentation and reusable computation. (a) GSBS constructs state templates, searches candidate boundaries and selects the state count using t-distance. (b) Temporal prefixes supply reusable interval scores,  $Q_{a,b} = F(a,b)$ ; single and double splits replace only their parent contribution. The best proposals undergo t-distance comparison and sequential refinement. (c) Two-dimensional prefixes of valid-pair counts, correlations and squared correlations provide within-state and adjacent-state statistics by four-corner queries. The full symmetric correlation matrix is shown; statistics use each eligible pair once. Three numerical backends feed a shared search controller. Data-derived elements use a 120-TR Sherlock fMRI excerpt (Baldassano, 2020; Chen et al., 2017); heatmaps show feature subsets, while fitting uses all 4,230 voxels. Detailed construction and provenance are in Online Resource 1, Section S4.

**Table 1** Principal computational costs.  $L$  is the searched block length,  $q$  the prefix chunk size, and  $K$  the current state count. Entries refer to the named operation; complete-fit cost also depends on the number of legal candidates and accepted updates.

| Operation                      | Reference-style calculation                   | HP-GSBS calculation                                           |
|--------------------------------|-----------------------------------------------|---------------------------------------------------------------|
| One candidate score            | Reconstruct and normalize an $O(TP)$ template | $O(1)$ Full-Q access; $O(P)$ raw-prefix score                 |
| Interval precomputation        | Repeated during candidate evaluation          | $O(T^2P)$ time; $O(T^2)$ Full-Q storage                       |
| Double-candidate score storage | $O(L^2)$ materialized scores                  | Running winner; bounded $q \times P$ workspace in prefix mode |
| Fixed-partition t-distance     | Scan $O(T^2)$ pair structure                  | $O(K)$ Prefix2D queries after endpoint extraction             |
| Low-memory t-distance          | Dense time-pair representation                | Feature moments: $O(TP^2 + KP^2 + TdP)$ arithmetic            |
| Complete boundary history      | $O(K_{\max}T)$ output                         | $O(K_{\max}T)$ output retained                                |

Thus  $F(a, b)$  depends only on the observations inside the interval. The derivation applies to spatially non-constant state means; the implementation follows the reference handling of numerical special cases (Online Resource 1, Section S1).

Inserting a boundary at  $i$  replaces one parent contribution with two child contributions. A double insertion at  $i < j$  introduces three children:

$$\Delta_1(i) = F(a, i) + F(i, b) - F(a, b), \quad (4)$$

$$\Delta_2(i, j) = F(a, i) + F(i, j) + F(j, b) - F(a, b). \quad (5)$$

All other terms remain available for reuse. In a local search block of length  $L_b$ , multiplication by  $T/L_b$  reproduces the original block normalization. The Full-Q route stores  $Q_{a,b} = F(a, b)$  for valid intervals, requiring  $O(T^2P)$  precomputation and  $O(T^2)$  storage. Thereafter, each candidate score requires a fixed number of table accesses and scalar operations. A raw-prefix alternative computes the same interval expression from  $A$  and  $B$  as needed, exchanging  $O(P)$  work per interval for reduced storage.

### 2.3 T-distance from sufficient statistics

T-distance evaluates a partition using correlations  $r_{ij}$  between spatial patterns at time points  $i$  and  $j$ . Let  $s(i)$  denote the state label and  $d$  the temporal exclusion parameter **dmin**. Eligible within-state and adjacent-state distributions are

$$W = \{r_{ij} : j - i \geq d, s(i) = s(j)\}, \quad (6)$$

$$A = \{r_{ij} : j - i \geq d, s(j) = s(i) + 1\}. \quad (7)$$

Each eligible pair enters once. For either distribution, count  $n$ , sum  $S$ , and squared sum  $R$  determine the mean and unbiased variance:

$$\mu = S/n, \quad v = \frac{R - S^2/n}{n - 1}. \quad (8)$$

The partition contrast is the Welch statistic (Welch, 1947),

$$TD = \frac{\mu_W - \mu_A}{\sqrt{v_W/n_W + v_A/n_A}}. \quad (9)$$

GSBS uses this statistic as a segmentation criterion and compares its values along the state-count curve.

The Prefix2D route precomputes zero-padded two-dimensional prefixes of the eligible-pair indicator, masked correlations, and masked squared correlations. A rectangle with row interval  $[a, b]$  and column interval  $[c, e]$  has sum  $\Pi_{b,e} - \Pi_{a,e} - \Pi_{b,c} + \Pi_{a,c}$ . Within-state rectangles and rectangles linking adjacent states therefore provide all six sufficient statistics in Eq. (9). Once partition endpoints are available, the calculation requires  $O(K)$  rectangle queries for  $K$  states. Endpoint extraction from the stored boundary vector costs  $O(T)$ . Full-Q serves interval-template scoring; Prefix2D serves time-pair aggregation. This distinction permits each representation to be optimized for its own access pattern.

Correlations are formed from row-standardized observations with residual row means removed, matching the reference covariance calculation. Small-sample handling follows the original implementation. Ambiguous comparisons and sensitive variance calculations invoke reference-expression reevaluation. The standard execution path uses FP64 throughout. Online Resource 1, Section S2 also derives an optional feature-moment formulation that aggregates correlations in feature space and avoids the dense time-pair tables.

### 2.4 Candidate search and sequential refinement

Double-boundary search visits legal pairs within a state. Materializing their scores requires a quadratic score array in the searched interval length. Streaming reduction visits these candidates in the original order and retains a running winner. With Full-Q, each visit combines interval-table entries; the raw-prefix route evaluates candidates in bounded chunks. Candidate enumeration, first-maximum tie handling, and

single-versus-double comparison follow the reference procedure. Streaming consequently reduces the storage associated with candidate scoring while retaining its search space.

Refinement removes and reinserts each boundary in sequence. HP-GSBS evaluates the requested local window and updates the current partition before moving to the next boundary. Here `finetune=1` specifies a one-sample search radius. Strength-based ordering follows the configured reference behavior. Sequential updates retain the dependency of later moves on earlier ones, while interval reuse limits recalculation to the affected states. An active-block cache stores local search winners and reuses them while their partition and legal candidate domain remain unchanged; affected entries are invalidated after an insertion or refinement move. These optimizations reduce redundant evaluations across successive iterations.

The result retains the complete boundary history and original state-count indexing. A double insertion can populate neighboring state-count slots with the same accepted partition, and the final accepted partition can contain  $K_{\max} + 1$  states. Preserving these arrays permits direct comparison with the original implementation and inspection of the complete t-distance curve.

## 2.5 CPU and GPU implementations

The CPU backend combines NumPy and SciPy array operations with Numba-compiled search kernels (Harris et al., 2020; Lam, Pitrou, & Seibert, 2015; Virtanen et al., 2020). Its Full-Q implementation uses bounded tiles during interval precomputation and fused running reductions during search. The PyTorch CUDA backend uses batched matrix products and prefix operations (Paszke et al., 2019). The Triton backend uses tiled kernels that combine interval differences, dot products, and norms (Tillet, Kung, & Cox, 2019). Both GPU routes transfer the resulting reusable statistics to the common CPU search controller.

This allocation places regular, high-volume arithmetic on the accelerator and retains partition-dependent control on the CPU. Candidate proposals, t-distance comparisons, and refinement moves proceed in the same sequence across backends. Performance therefore reflects the balance between precomputation, transfer, and sequential search for a particular workload. The stable configuration combines FP64 Full-Q, Prefix2D, and streaming search. The optional CPU low-memory route combines raw prefixes with feature-moment t-distance; FP32 and selective GPU-resident candidate reduction remain experimental options with separately documented numerical and resource scopes (Online Resource 1, Section S3).

## 2.6 Software interface and execution modes

HP-GSBS accepts prepared time-by-feature arrays. A configuration records the scientific parameters, including maximum state count, temporal exclusion, statewise detection, block size, and refinement. Execution options select the backend, resource limits, and latency or throughput scheduling. A minimal call is `fit(x, GSBSConfig(kmax=60, dmin=1), backend="cpu-fast")`. Results expose the selected state count, boundary arrays at every evaluated index, t-distance curve, state patterns, and boundary strengths. Versioned NPZ persistence retains these outputs for later comparison and visualization.

Recording collections can use independent CPU workers or compatible GPU batches. The manifest-based command-line interface records inputs and configuration, writes completed results atomically, and resumes tasks whose execution identities match. These features connect single-record segmentation to cohort workflows while retaining a separate result for every recording. A runnable example, installation information, and the result schema are supplied with the source materials.

ChatGPT and Codex (OpenAI) assisted software development, experiment and plotting code, manuscript drafting, and language revision. Figures were rendered programmatically from recorded numerical arrays using Matplotlib (Hunter, 2007). The authors directed the analyses, checked the numerical evidence, and take responsibility for the software, interpretation, and final manuscript.

## 2.7 Evaluation design and datasets

The evaluations address complementary questions: implementation agreement, paired complete-fit performance, real-data consistency, resource trade-offs, and cohort application (Table 2). HP-GSBS 0.2.0 is compared with unmodified statesegmentation 0.0.6. Boundary histories and selected state counts require exact equality. T-distance, state patterns, and strengths use  $\text{atol} = \text{rtol} = 10^{-8}$ , with array shapes and nonfinite masks checked separately.

### *Controlled benchmark.*

Seven configurations vary  $T \in \{256, 512, 768\}$ ,  $P \in \{8, 32, 64\}$ , or  $K_{\max} \in \{10, 20, 40\}$  around the shared center (512, 32, 20). Each uses seeds 20261017 and 20261018. Inputs comprise eight contiguous states with Gaussian spatial means and Gaussian noise of standard deviation 0.8. Nested feature subsets and reused inputs across state-count settings yield 14 configuration and seed combinations containing ten distinct matrices. Each case has one timed official fit and, for each optimized backend, one warm-up and three timed fits.

**Table 2** Datasets and evaluation roles.  $T \times P$  denotes time points by features. Numerical criteria and timing protocols are specified in Section 2.7 and Online Resource 1.

| Evaluation            | Input                                                                       | Configuration or scope                       | Principal evidence                                     |
|-----------------------|-----------------------------------------------------------------------------|----------------------------------------------|--------------------------------------------------------|
| Controlled comparison | Seven configurations; two seeds; $T = 256\text{--}768$ , $P = 8\text{--}64$ | $K_{\max} = 10\text{--}40$ ; 14 paired cases | Complete-fit speedup; 168 optimized-output comparisons |
| Illustrative signal   | $80 \times 16$ ; four generating states                                     | Seed 19; noise SD 0.05                       | Boundary recovery and decision traces                  |
| BBBD EEG              | Four matrices; $T = 1336\text{--}1857$ ; $P = 32$ or 64                     | 5.12 Hz; $K_{\max} = 121\text{--}172$        | Full-record saved-output consistency                   |
| Resource measurements | EEG ablation; long inputs; independent task collections                     | Fixed configurations per series              | Memory, component costs, throughput                    |
| Sherlock fMRI         | 16 participants; each $946 \times 4230$                                     | TR 1.5 s; $K_{\max} = 60$ ; 17 CPU fits      | Group and participant segmentation; boundary alignment |

All 168 optimized outputs, including warm-ups, enter numerical validation.

Measurements used a Ryzen 7 7840H CPU and RTX 4050 Laptop GPU (6 GB), Ubuntu 22.04 under WSL2, Python 3.10.12, NumPy 2.2.6, SciPy 1.15.3, Numba 0.68.0, PyTorch 2.11.0 with CUDA 12.8, and Triton 3.6.0. Numerical-library thread counts were one; TF32 was disabled. Backend order rotated across cases, and GPU calls were synchronized immediately before and after timing. Complete-fit times include construction, computation, and required transfers; validation and file operations occur outside the interval. For each input, the official time is divided by the median of three optimized times. Configuration summaries report the median of the two paired ratios and their observed range. Complementary ablation, memory, and throughput measurements retain their original release-development configurations and are detailed in Online Resource 1, Sections S4–S6.

### EEG.

Four frozen matrices derived from the Brain, Body, and Behavior Dataset (BBBD) (Madsen et al., 2026) contain 1,336–1,857 samples and 32 frontal or 64 channels. Recorded preprocessing comprises event cropping, artifact masking, nonoverlapping 25-sample block means from 128 to 5.12 Hz, and within-recording channel z-scoring using valid-bin sample standard deviations. Stored float32 inputs are converted to FP64 for fitting. All recordings use  $\text{dmin}=5$ , statewise detection, a one-sample refinement radius, strength-based refinement ordering, and block size 50, with recording-specific  $K_{\max}$  (Table 3). Recording 1 supplies a complete original-reference output; Recordings 2–4 supply the historical CPU reference chain previously validated against official results. Public provenance records identify the frozen arrays, processing history, and available upstream recording identifiers.

### Naturalistic fMRI.

The Sherlock application uses the public angular-gyrus array distributed by Baldassano (2020), derived from Chen et al. (2017). The supplied array contains 16 participants, 946 time points at TR = 1.5 s, and 4,230 voxels, covering a 23.65-min movie segment. We retained the supplied preprocessing and converted each matrix to FP64. The arithmetic mean across participants was segmented as a group response, followed by each participant independently. All 17 fits used the CPU Full-Q/Prefix2D path,  $K_{\max} = 60$ ,  $\text{dmin}=1$ , statewise detection, a one-sample refinement radius, strength-based ordering, and block size 50. Complete-fit times were recorded once per matrix on the same CPU platform with single-thread numerical libraries.

Intersubject alignment was measured by one-to-one matching of selected boundaries within  $\pm 3$  TR ( $\pm 4.5$  s). For participants with  $n_i$  and  $n_j$  boundaries and  $M_{ij}$  matches,  $F_{1,ij} = 2M_{ij}/(n_i + n_j)$ . The cohort statistic is the mean across 120 participant pairs. A circular-shift null independently displaced each participant’s boundary train by a uniformly sampled nonzero offset, preserving its boundary count and circular spacing. Each of 1,000 draws produced one cohort-mean statistic; matching used absolute temporal distance on the displayed linear segment. The one-sided Monte Carlo probability was  $(1 + \#\{F_1^{\text{null}} \geq F_1^{\text{observed}}\})/1001$ , with seed 20261020. Online Resource 1, Section S8 gives complete outputs and implementation details.

## 3 Results

### 3.1 Numerical agreement with reference GSBS

The controlled experiment yielded exact agreement in complete boundary histories and selected state counts for all 168 optimized executions. State patterns and boundary strengths were identical, and the largest absolute t-distance difference was  $1.87583 \times 10^{-11}$ , within the prescribed tolerance. These comparisons cover all

three optimized backends, seven workload configurations, and both input realizations. They connect the interval and correlation-statistic formulations to the complete output of the original algorithm.

A four-state signal provides an interpretable illustration (Fig. 2). Its 80 observations span 16 features, with generating boundaries at samples 20, 40, and 60 and Gaussian noise of standard deviation 0.05. All implementations recover these boundaries and select four states. Maximum absolute t-distance differences are  $1.51 \times 10^{-11}$  for CPU and  $6.93 \times 10^{-12}$  for both GPU routes. The saved search and refinement traces also agree at every recorded decision. Additional seed and noise examples in Online Resource 1, Section S6 document how generating-boundary recovery varies while implementation agreement is maintained.

### 3.2 EEG segmentation and cross-backend agreement

The four BBBB EEG recordings extend validation to study-preprocessed electrophysiological matrices (Table 3). Complete boundary histories and selected state counts agree across CPU, PyTorch CUDA, and Triton outputs relative to their respective reference records. Selected counts range from 120 to 173, with maximum absolute t-distance differences of  $1.85 \times 10^{-13}$  to  $3.13 \times 10^{-13}$ . State patterns and strengths satisfy the same numerical criteria. The public collection supplies all four frozen inputs, time maps, configurations, FP64 outputs, and provenance records, enabling input-to-output verification for each recording.

Figure 3 follows Recording 1 through its full 64-channel analysis. All four implementations select 126 states and reproduce the complete boundary history; the maximum absolute t-distance difference is  $2.43 \times 10^{-13}$ . A fixed central window links the selected boundaries to the observed multichannel patterns and their temporal correlations. Boundaries and state intervals in this view are extracted from the full-record result. The complete state-count curve provides the corresponding selection context, connecting local pattern transitions to the recording-level segmentation.

### 3.3 Controlled runtime performance

Median same-input speedups span 154–1,730 across the controlled configurations and backends (Fig. 4). At the central configuration, official GSBS takes 64.7 s, compared with 0.142 s for CPU, 0.081 s for PyTorch CUDA, and 0.099 s for Triton. Median paired ratios are 454, 795, and 675, respectively. These ratios are calculated from unrounded per-input measurements before summarization across the two seeds.

The length sweep shows increasing gains as repeated temporal calculations become more expensive. At  $T = 768$ , CPU reaches a 770-fold speedup and PyTorch

reaches 1,730-fold. Across the feature-count and search-depth sweeps at  $T = 512$ , optimized times remain 0.065–0.165 s. The CPU results quantify the contribution of computational reuse on the same processor, while the GPU routes provide additional reductions where regular precomputation benefits from device execution. Complete numerical observations and separate historical extended workloads are retained in Online Resource 1, Sections S5–S6.

### 3.4 Computational reuse, memory, and throughput

The CPU ablation isolates complementary optimizations within an existing Full-Q implementation (Fig. 5a). On the  $1336 \times 64$  EEG input with  $K_{\max} = 131$ , streaming candidates reduces median complete-fit time from 3.364 to 1.803 s. Local-window refinement reduces it to 1.601 s, and active-block reuse reduces it further to 1.429 s. All configurations retain the same scientific settings and agree in their compared outputs. The progression shows that candidate storage, refinement work, and repeated block searches each contribute to runtime after interval scores have been precomputed.

Low-memory execution supports a different resource balance (Fig. 5b). On a  $5000 \times 8$  input with single-boundary search, refinement disabled, block size zero, and  $K_{\max} = 16$ , raw-prefix/feature-moment execution reduces peak process RSS from 1,598.0 to 190.6 MiB and complete-fit time from 9.754 to 0.110 s. A separate four-workload series reaches  $50000 \times 64$  with 469.5 MiB peak RSS and 1.444 s runtime. Both recording length and feature count increase in this series. Selected and final-partition t-distance values agree with an independent tiled calculation within  $1.15 \times 10^{-14}$ .

Recording-level parallelism and GPU batching show workload-dependent throughput (Fig. 5c,d). For short synthetic inputs, CPU throughput increases from 274 to 1,767 fits/minute with one versus eight workers. PyTorch throughput increases from 647 to 1,418 fits/minute at batch size eight. Repeated high- $K$  EEG tasks favor CPU workers: eight workers reach 155 fits/minute, compared with approximately 40–45 for the GPU routes. These measurements include scheduling and batch overhead and illustrate the practical balance between regular precomputation and each recording’s sequential search.

### 3.5 Naturalistic fMRI application and intersubject alignment

HP-GSBS segmented the group-mean angular-gyrus response and all 16 participant matrices using 4,230 voxels and 946 time points per fit (Fig. 6). The group response selected 60 states. Participant selections ranged from 19 to 61 states, with a median of 56.5; seven participants reached indices of 60 or 61.

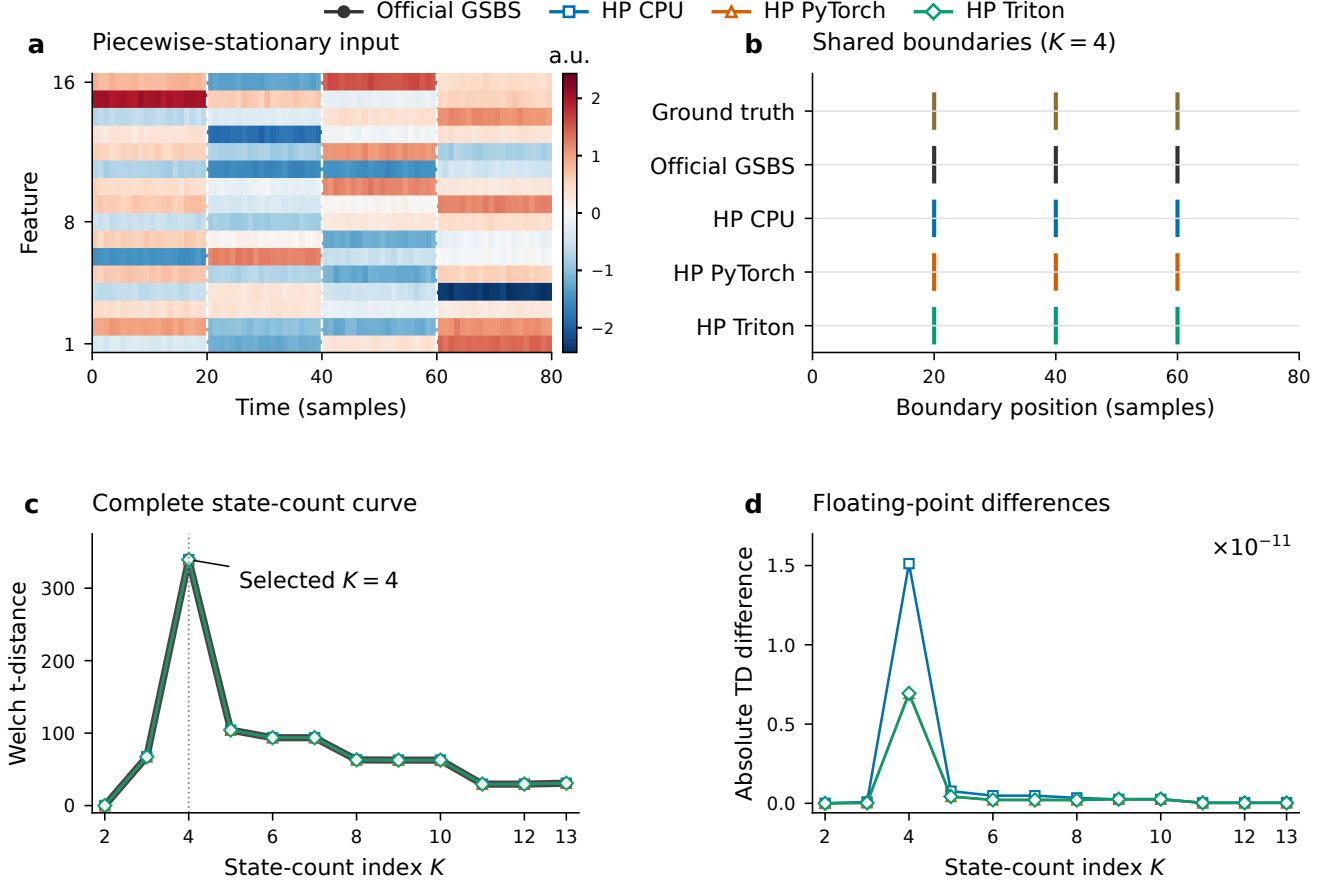


**Fig. 2** Segmentation agreement on a four-state signal. (a) Synthetic activity; dashed lines mark generating boundaries. (b) Ground truth and selected boundaries from the official implementation and three HP-GSBS backends. (c) Complete t-distance curves, retaining double-insertion indexing; all routes select  $K = 4$ . (d) Absolute differences from official values; GPU curves overlap. Seed 19; noise standard deviation 0.05;  $T = 80$ ,  $P = 16$ ,  $K_{\max} = 12$ .

**Table 3** Saved-output agreement on four BBBB EEG recordings, summarized over CPU, PyTorch CUDA, and Triton FP64 outputs. All recordings use  $\text{dmin}=5$ , statewise detection,  $\text{finetune}=1$ , strength-based refinement ordering, and block size 50. Recording 1 has a complete official reference; Recordings 2–4 use the historically official-validated CPU chain. Double insertion retains the original  $K_{\max} + 1$  indexing.

| Recording | $T \times P$     | $K_{\max}$ | Selected $K$ | Maximum $ \Delta\text{TD} $ | Equal histories |
|-----------|------------------|------------|--------------|-----------------------------|-----------------|
| 1         | $1336 \times 64$ | 131        | 126          | $2.43 \times 10^{-13}$      | Yes             |
| 2         | $1857 \times 32$ | 121        | 120          | $3.13 \times 10^{-13}$      | Yes             |
| 3         | $1754 \times 64$ | 172        | 173          | $2.77 \times 10^{-13}$      | Yes             |
| 4         | $1374 \times 64$ | 135        | 131          | $1.85 \times 10^{-13}$      | Yes             |

The group and these participant selections therefore lie at the terminal end of the evaluated range. Their curves describe segmentation within the specified search depth and motivate wider-range sensitivity analyses for detailed timescale estimation.

The group fit took 13.56 s. Individual fits took 11.88–13.15 s, with a median of 12.38 s, and the 17 complete fits totaled 213.10 s. These observations demonstrate the practical execution of a voxel-level cohort analysis using the CPU route. They are reported as application

fit times with their measured configuration; the controlled experiment provides the paired official-reference speedup estimates.

Selected boundaries showed greater intersubject alignment than the circular-shift baseline. Mean pairwise one-to-one  $F_1$  was 0.4141, compared with a null mean of 0.3249, a difference of 0.0892. The central 95% of the 1,000 null cohort means lay between 0.3148 and 0.3354, and the observed statistic exceeded all null draws ( $p = 0.001$ ). Figure 6 combines the group-response display, participant boundary raster, complete t-distance curves, and null distribution. The result

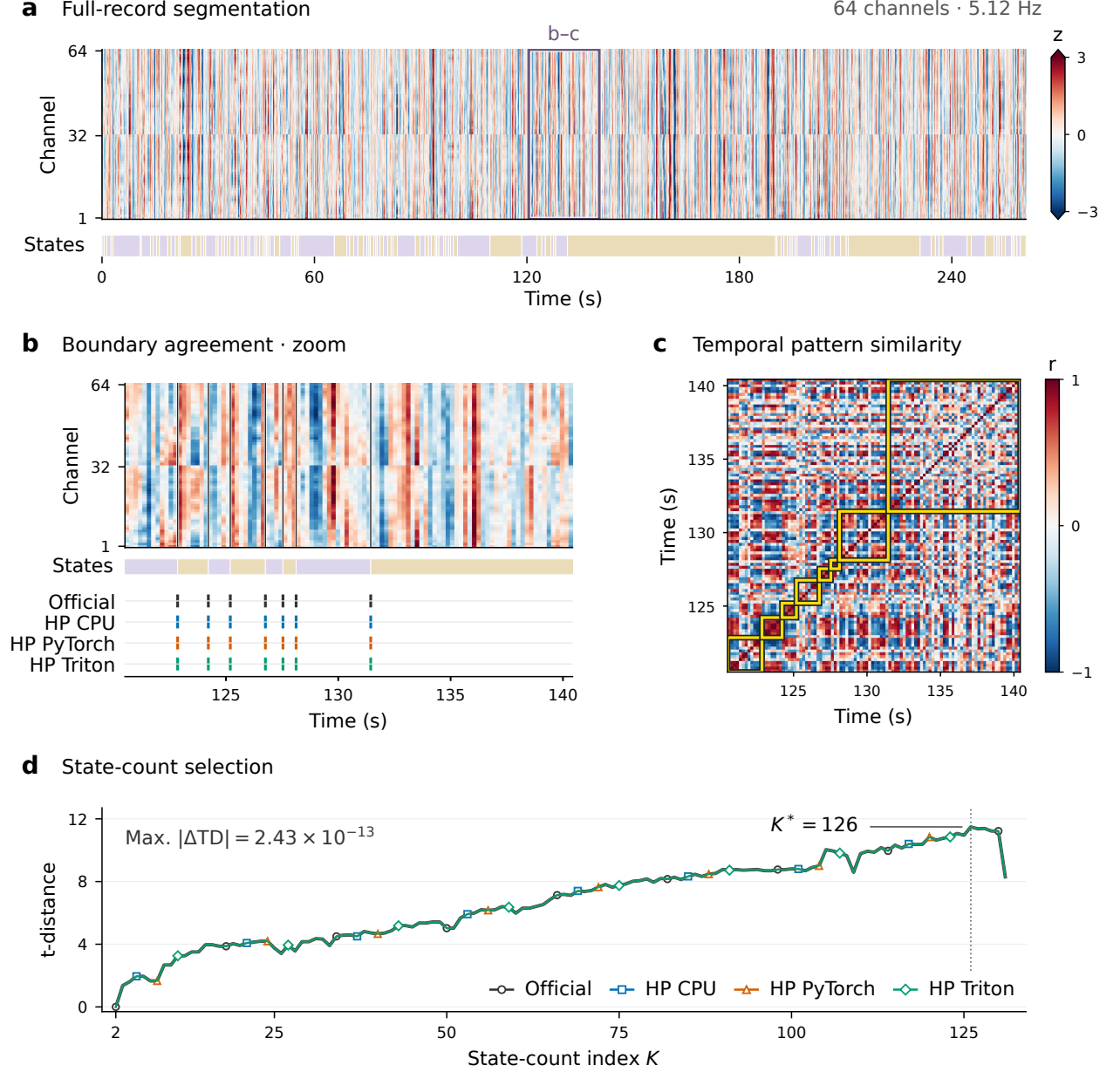


**Fig. 3** Real EEG segmentation from saved full-record fits. (a) Standardized 64-channel EEG and the selected state sequence; alternating colours distinguish adjacent intervals. (b) A fixed central window (120.51–140.43 s), with boundaries from four implementations aligned beneath the input. Both EEG heatmaps use a colour scale clipped at  $\pm 3$ . (c) Full symmetric spatial-pattern correlations for the same window; outlined blocks mark selected state intervals. (d) All evaluated t-distance indices, preserving double-insertion indexing; marker positions are staggered for visibility. All routes select 126 states with identical boundary histories and a maximum absolute TD difference of  $2.43 \times 10^{-13}$ . Parameters:  $K_{\max} = 131$ ,  $d_{\min}=5$ , statewise detection, sequential refinement, block size 50 and FP64.

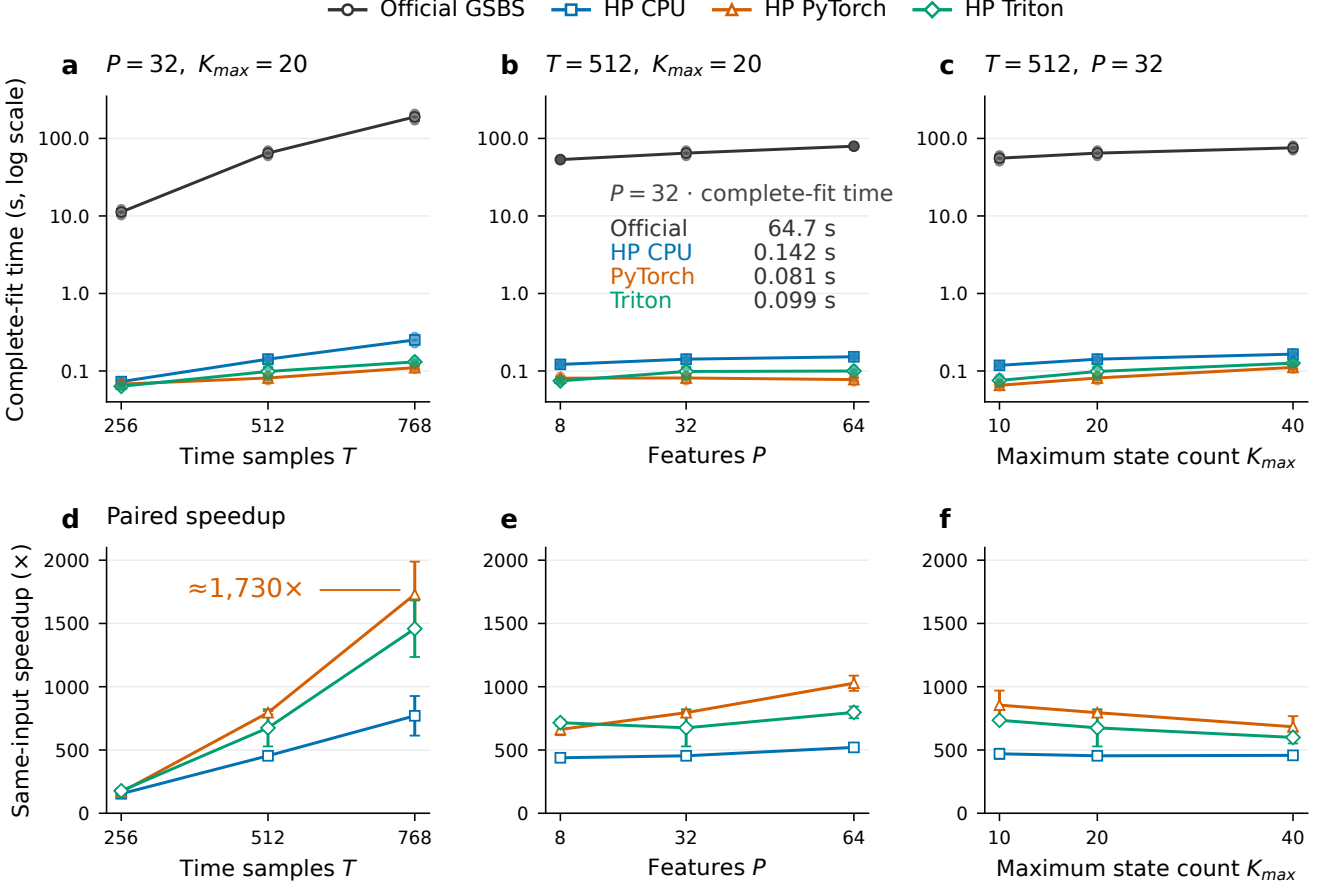


**Fig. 4** Controlled complete-fit performance. (a–c) Runtime versus  $T$ ,  $P$  and  $K_{max}$ , with the other two fixed. Small filled points show two per-input observations; open markers show their median, and whiskers their range. HP observations summarize three timed fits after warm-up; official observations are single fits. The annotation in b gives central-configuration runtimes. (d–f) Same-input official/HP speedups across seeds; d highlights the median PyTorch speedup at  $T = 768$ . The central configuration is reused across panels. All fits use statewise detection and sequential refinement.

illustrates how a collection of saved segmentations supports a cohort-level analysis of shared temporal structure while retaining individual boundary locations and state-count curves.

## 4 Discussion

### 4.1 Computational contributions

HP-GSBS changes the organization of repeated work in neural-state segmentation. The interval decomposition makes the locality of a boundary update explicit: its matching-score effect is determined by the parent interval and its proposed children. Temporal prefixes supply these contributions, and Full-Q converts repeated candidate evaluation into table access. Correlation sufficient statistics provide an analogous representation for partition comparison. Their combination links two computational tasks that recur throughout GSBS to data structures that can be built once and queried repeatedly.

This formulation also clarifies the roles of the subsequent optimizations. Streaming reduction lowers the

storage cost of candidate scoring, sequential window refinement focuses work on the configured neighborhood, and active-block reuse carries forward winners whose local search problem remains unchanged. The ablation shows complementary runtime gains from these components. Exact boundary histories across the controlled experiment establish that these changes retain the measured segmentation decisions, while small t-distance differences characterize floating-point accumulation across the numerical routes. The four EEG cases provide an additional check on longer, study-preprocessed matrices with substantially higher selected state counts. Together, the evaluations establish a reproducible comparison at several levels: complete partition histories, selected outputs, continuous score curves, and interpretable real-data displays. Maintaining these levels in a common result object makes it possible to inspect a numerical difference in the context of the boundary decisions it accompanies.

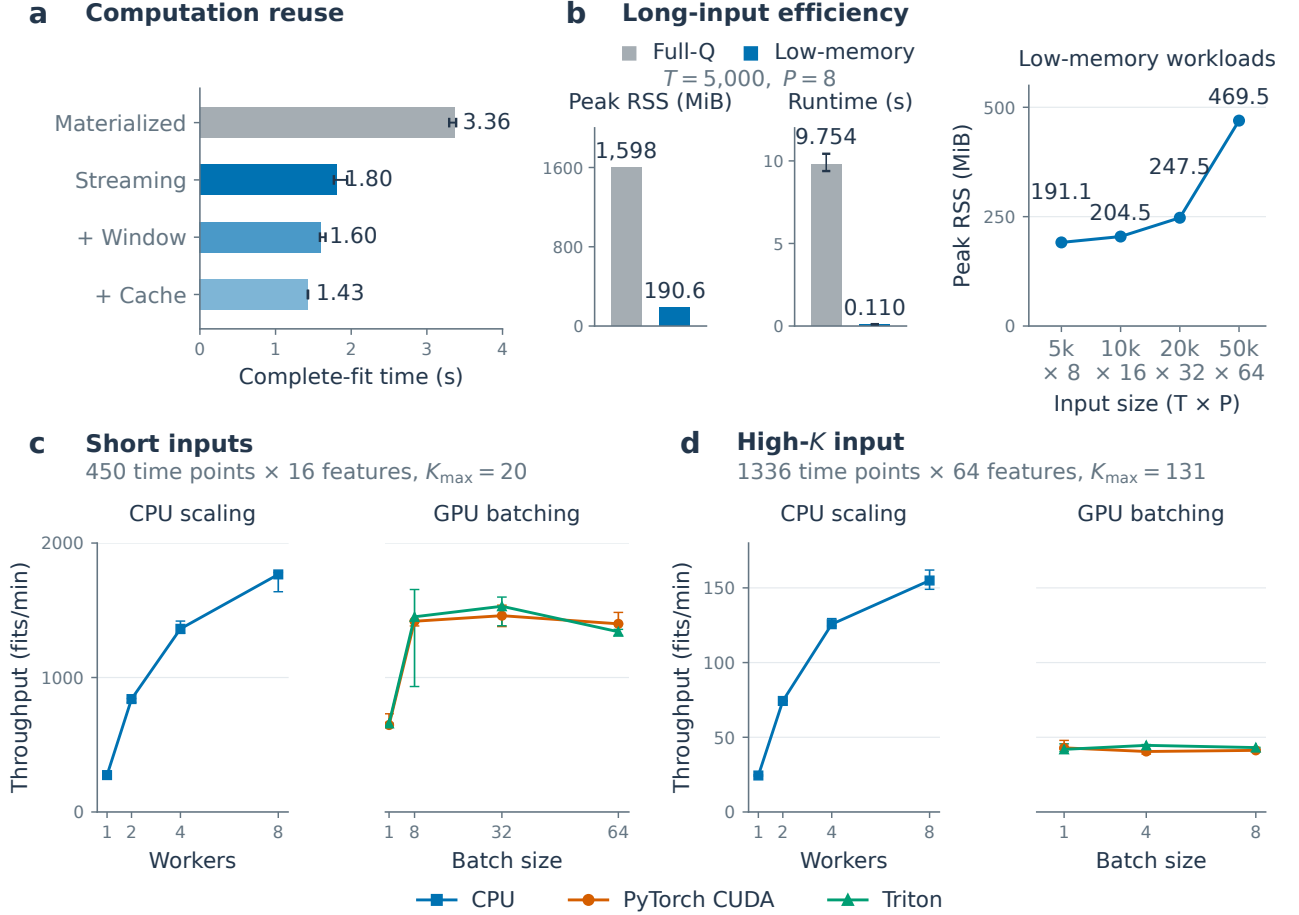


**Fig. 5** Computation reuse, memory and parallel execution. (a) CPU ablation on a  $1336 \times 64$  EEG input,  $K_{\max} = 131$ . (b) Left: peak process RSS and complete-fit time for a  $5000 \times 8$  input. Right: four increasing low-memory workloads; both  $T$  and  $P$  change. Separate measurements give 190.6 and 191.1 MiB at  $5000 \times 8$ . All b measurements use single-boundary search, no refinement, block size zero and  $K_{\max} = 16$ . (c,d) Throughput versus CPU workers and GPU batch size for short synthetic and high- $K$  EEG inputs. CPU batches contain 32 and eight tasks, respectively; high- $K$  tasks repeat one EEG input. a, b runtimes and c/d show three-run medians and observed-range whiskers; memory values are measured peaks. These are existing release-development measurements.

## 4.2 Implications for scalable neuroscience analysis

Neural-state analysis often expands across participants, regions, stimuli, and parameter settings. Reducing complete-fit latency changes how readily these dimensions can be explored. The controlled experiment turns minute-scale fits into subsecond calculations on laptop-class hardware. Recording-level scheduling extends that benefit to collections, and saved boundary histories permit subsequent state-count inspection and comparison. Publicly supplied analysis matrices, configuration records, and numerical outputs further support reuse of the method, supporting independent inspection and reuse of computational neuroscience methods.

The EEG and fMRI applications address complementary forms of reuse. Four BBBD inputs provide a traceable connection between prepared electrophysiological matrices and cross-backend outputs. The Sherlock application applies the same interface to high-dimensional fMRI and produces both group and

participant segmentations within a short cohort computation. Its boundary-alignment analysis shows how these outputs can be combined into a statistic of shared temporal structure. This direction is consistent with naturalistic studies of common neural organization during narrative processing (Baldassano et al., 2017; Chen et al., 2017) and with GSBS analyses of regional neural-state structure (Geerligs et al., 2022).

The interpretation follows the sampling and search choices. The fMRI analysis uses one angular-gyrus ROI and a 23.65-min segment; its matching tolerance is 4.5 s, and its null preserves each participant’s boundary count and circular spacing. The observed excess alignment quantifies agreement under those choices. Several selected counts reach the search-range limit, making a wider  $K_{\max}$  range a useful next step for state-timescale analyses. The saved curves and boundaries provide the necessary basis for such sensitivity work. This application demonstrates an end-to-end computational use

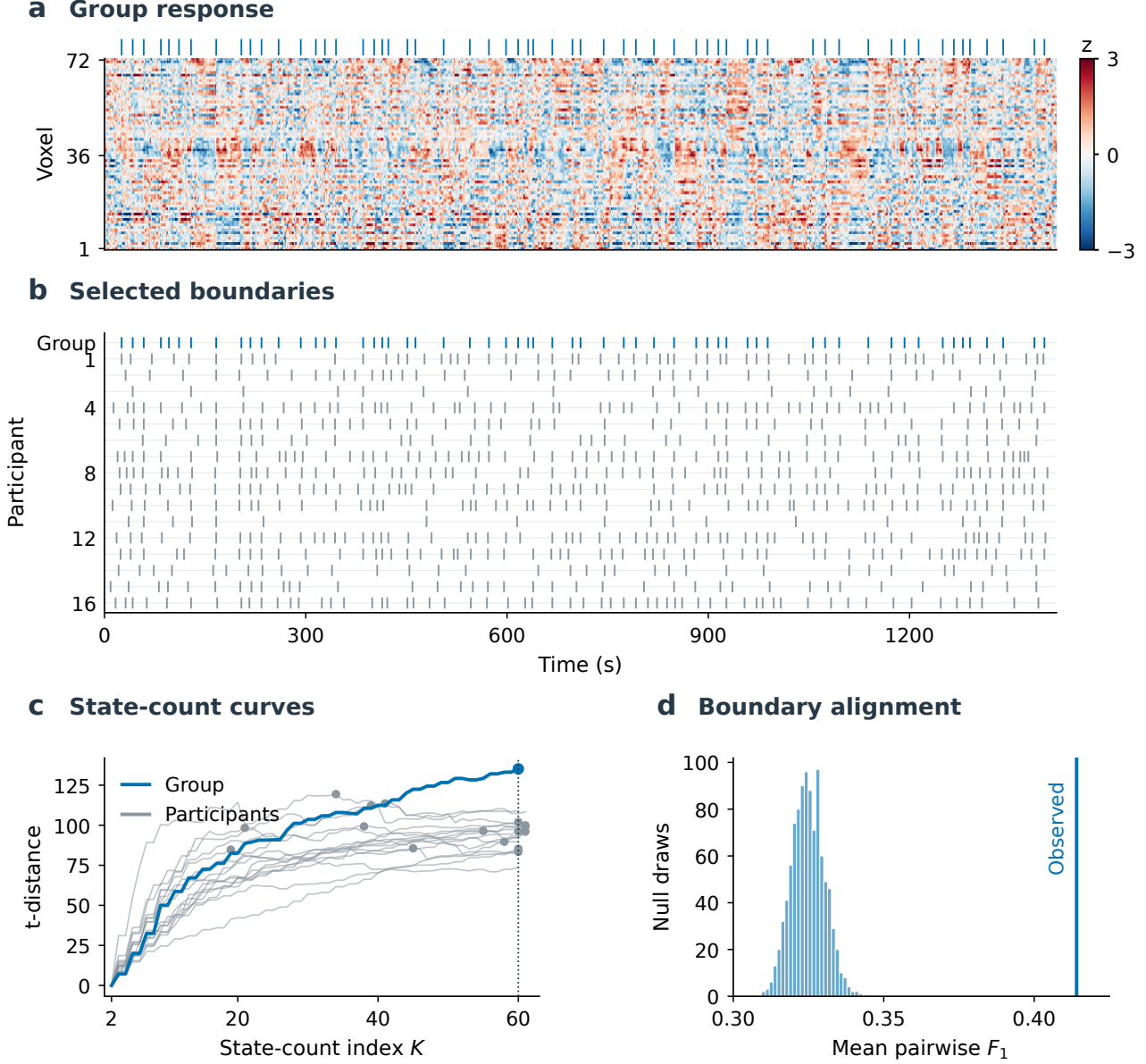


**Fig. 6** Naturalistic fMRI segmentation and intersubject boundary alignment. (a) Group-mean angular-gyrus activity, with group boundaries above the heatmap. The display uses 72 evenly sampled voxels and spatial z-scoring at each time point, clipped to  $\pm 3$ ; all fits use 4,230 voxels. (b) Selected boundaries for the group and 16 participants; participant labels correspond to the supplied array order. (c) Complete evaluated t-distance curves and selected state-count markers. The vertical line marks  $K_{\max} = 60$ ; the original double-insertion convention permits index 61. (d) Circular-shift distribution of the cohort-mean pairwise boundary-matching  $F_1$ , with the observed value marked. Tolerance is  $\pm 3$  TR ( $\pm 4.5$  s); 1,000 null draws yield  $p = 0.001$ . Data cover a 23.65-min movie segment at TR = 1.5 s.

of neural-state segmentation, from voxel matrices to individual outputs and intersubject comparison.

### 4.3 Implementation and resource trade-offs

Backend selection depends on the relative costs of pre-computation, transfer, search, and scheduling. FP64 Full-Q/Prefix2D is the standard route for the evaluated statewise analyses: it pays a quadratic storage cost to make repeated scoring inexpensive. CPU execution is attractive when sequential search dominates

or independent recordings can occupy several workers. GPU execution is useful when dense precomputation is a substantial fraction of the fit and compatible inputs permit batching. The short-input and high- $K$  throughput measurements illustrate these different balances. Device characteristics, particularly FP64 performance and available memory, will influence the same trade-off on other systems.

The low-memory route offers a complementary option for long inputs with modest feature counts and a suitable search configuration. Its feature-moment

statistics replace dense time-pair storage with feature-space summaries, while raw prefixes provide interval scores on demand. The reported memory series uses single-boundary search and disabled refinement; its workload definition is therefore central to interpreting the observed efficiency. Statewise double-candidate enumeration still grows quadratically within long intervals, and the full boundary history requires  $O(K_{\max}T)$  storage in every route. These costs provide concrete guidance when choosing analysis depth and execution resources.

#### 4.4 Future development

Further work can extend the present formulation across broader hardware and analysis settings. Incremental updates to partition statistics may reduce repeated t-distance queries, and selective device-resident reduction may benefit large candidate domains. Mixed-precision work can investigate the relation between throughput, near-tied scores, and boundary stability. Such extensions can build on the complete-history comparisons and reproducible inputs supplied here.

At the application level, multi-region naturalistic analyses and wider state-count sweeps would test how computational gains support richer questions about temporal organization. Subject-level parallelism is especially relevant to resampling, parameter sensitivity, and larger cohorts. Retaining frozen inputs, explicit configurations, and intermediate selection curves will help connect these expanded analyses to their numerical evidence. The present CPU and GPU routes provide a shared starting point for that development. Beyond GSBS, the interval formulation illustrates a useful design principle for sequential analysis software: identify which quantities depend only on immutable observations, compute them in a regular form, and express model updates through their changing local contributions. This separation makes the mathematical source of computational savings visible and helps connect optimized kernels to an interpretable search procedure.

## 5 Conclusions

HP-GSBS accelerates neural-state segmentation through additive interval statistics, efficient t-distance aggregation, and reuse of unchanged search work. Three FP64 backends reproduce the measured reference decisions, with 154–1,730-fold same-input speedups in controlled workloads and consistent saved outputs across four EEG recordings. A 16-participant naturalistic fMRI application demonstrates voxel-level segmentation and intersubject boundary analysis. The mathematical formulation, common result interface, and reusable numerical materials support efficient and traceable GSBS analyses across recordings and cohorts.

## Information Sharing Statement

HP-GSBS 0.2.0 is available under the MIT license at <https://github.com/JustinZHAO-05/HP-GSBS/tree/v0.2.0> and through its versioned Zenodo record (Zhao, 2026). Reproducibility materials version 1.1.0 are archived at <https://doi.org/10.5281/zenodo.23573202> (Zhao & Ji, 2026a) and correspond to tag v1.1.0 in <https://github.com/JustinZHAO-05/HP-GSBS-SoftwareX-Reproducibility>. This archive supplies all four EEG analysis inputs, time maps, scientific configurations and saved FP64 outputs, the complete Sherlock application with 17 segmentations and boundary-alignment statistics, and the data and scripts for six main figures. The earlier version 1.0.1 archive (Zhao & Ji, 2026b) preserves the original benchmark and figure materials; the all-versions identifier is <https://doi.org/10.5281/zenodo.23457341>. Online Resource 1 (ESM\_1.pdf) provides the mathematical derivations, complete benchmark and resource tables, EEG provenance and file mapping, and Sherlock protocol with reproduction commands. Original BBBB data are described by Madsen et al. (2026); the Sherlock angular-gyrus input is available from Baldassano (2020). Source attribution and preprocessing records accompany the derived inputs. Reproduction code is MIT licensed; distributed data, derived figures, and documentation retain CC BY 4.0 attribution.

## Statements and Declarations

### Funding

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

### Competing interests

The authors declare that they have no competing interests.

### Author contributions

**Yanzhe Zhao:** Conceptualization, Methodology, Software, Formal analysis, Investigation, Validation, Data curation, Visualization, Writing—original draft, Writing—review and editing, Project administration.

**Yongzhang Ji:** Methodology, Formal analysis, Writing—review and editing.

### Ethics approval and consent

The study comprises secondary computational analyses of existing publicly shared human EEG and fMRI data. Participant recruitment, informed consent, and ethical oversight are described in the original dataset publications (Chen et al., 2017; Madsen et al., 2026). The

present analyses use the shared numerical recordings and their supplied identifiers.

## Data and code availability

Software, analysis data, numerical outputs, and reproduction materials are identified in the Information Sharing Statement.

## References

- Baldassano, C. (2020). *Sherlock data for OHBM*. figshare. Retrieved from <https://doi.org/10.6084/m9.figshare.12436955.v1> (Dataset. CC BY 4.0)
- Baldassano, C., Chen, J., Zadbood, A., Pillow, J.W., Hasson, U., Norman, K.A. (2017). Discovering event structure in continuous narrative perception and memory. *Neuron*, 95(3), 709–721.e5, <https://doi.org/10.1016/j.neuron.2017.06.041>
- Chen, J., Leong, Y.C., Honey, C.J., Yong, C.H., Norman, K.A., Hasson, U. (2017). Shared memories reveal shared structure in neural activity across individuals. *Nature Neuroscience*, 20, 115–125, <https://doi.org/10.1038/nn.4450>
- Geerligs, L., Gözükar, D., Oetinger, D., Campbell, K.L., van Gerven, M., Güçlü, U. (2022). A partially nested cortical hierarchy of neural states underlies event segmentation in the human brain. *eLife*, 11, e77430, <https://doi.org/10.7554/eLife.77430>
- Geerligs, L., van Gerven, M., Güçlü, U. (2021). Detecting neural state transitions underlying event segmentation. *NeuroImage*, 236, 118085, <https://doi.org/10.1016/j.neuroimage.2021.118085>
- Harris, C.R., Millman, K.J., van der Walt, S.J., Gommers, R., Virtanen, P., Cournapeau, D., ... Oliphant, T.E. (2020). Array programming with NumPy. *Nature*, 585, 357–362, <https://doi.org/10.1038/s41586-020-2649-2>
- Hunter, J.D. (2007). Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, 9(3), 90–95, <https://doi.org/10.1109/MCSE.2007.55>
- Lam, S.K., Pitrou, A., Seibert, S. (2015). Numba: A LLVM-based Python JIT compiler. *Proceedings of the second workshop on the llvm compiler infrastructure in hpc* (pp. 1–6). ACM.
- Madsen, J., Kuppa, N., Parra, L.C. (2026). The Brain, Body, and Behavior Dataset (BBBD): Multimodal recordings during educational videos. *Scientific Data*, 13, 920, <https://doi.org/10.1038/s41597-026-07215-1>
- Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., ... Chintala, S. (2019). PyTorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems* (Vol. 32). Retrieved from <https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html>
- Tillet, P., Kung, H.T., Cox, D. (2019). Triton: An intermediate language and compiler for tiled neural network computations. *Proceedings of the 3rd acm sigplan international workshop on machine learning and programming languages* (pp. 10–19). ACM.
- Virtanen, P., Gommers, R., Oliphant, T.E., Haberland, M., Reddy, T., Cournapeau, D., ... Vázquez-Baeza, Y. (2020). SciPy 1.0: Fundamental algorithms for scientific computing in Python. *Nature Methods*, 17, 261–272, <https://doi.org/10.1038/s41592-019-0686-2>
- Welch, B.L. (1947). The generalization of ‘Student’s’ problem when several different population variances are involved. *Biometrika*, 34(1–2), 28–35, <https://doi.org/10.1093/biomet/34.1-2.28>
- Zhao, Y. (2026). *HP-GSBS: High-performance greedy state boundary search*. Zenodo, version 0.2.0. Retrieved from <https://doi.org/10.5281/zenodo.23422148>
- Zhao, Y., & Ji, Y. (2026a). *Hp-gsbs: Reproducibility materials for neural-state segmentation and performance evaluation*. Zenodo. Retrieved from <https://doi.org/10.5281/zenodo.23573202> (Version 1.1.0)
- Zhao, Y., & Ji, Y. (2026b). *Hp-gsbs: Software reproducibility materials*. Zenodo. Retrieved from <https://doi.org/10.5281/zenodo.23457342> (Version 1.0.1)

## Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- [ESM1.pdf](#)