A HIERARCHICAL BAYESIAN APPROACH FOR NON-MARKOVIAN EVOLVING DOMAIN GENERALIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Evolving Domain Generalization (EDG) considers learning from a sequence of domains whose underlying distributions change over time, reflecting the non-stationary nature of many real-world learning environments. Recent structure-aware methods model this evolution by separating static task factors from dynamic domain factors that evolve across domains. However, their autoregressive latents impose a first-order Markov assumption, tying the current domain state to its immediate predecessor, which is restrictive when the dynamics have persistent temporal structure or abrupt transitions. We propose a hierarchical state-space model (SSM) formulation that introduces regime latents governed by change-point variables. Moreover, the formulation naturally casts EDG as a sequential Bayesian inference problem, which we leverage Rao-Blackwellized particle filter (RBPF) to approximate the posterior over the latents. Experiments on synthetic benchmarks containing both gradual and abrupt distribution shifts and real-world evolving domain benchmarks show that the proposed approach improves mean target-domain accuracy by 17.06 points over the strongest baseline on the original streams and by 15.74 points on the abrupt variants. An ablation shows that the change-point latent contributes 12.47 points of abrupt-stream accuracy. These results highlight the value of hierarchical sequential Bayesian inference for modeling complex dynamics in evolving domains.

1 INTRODUCTION

Deep learning models often degrade when the data distribution changes (Gulrajani & Lopez-Paz, 2021; Koh et al., 2021). Domain generalization (DG) addresses shift across multiple sources (Blanchard et al., 2011; Zhou et al., 2023), but remains limited in dynamic environments (Qin et al., 2022; Nasery et al., 2021). Evolving domain generalization (EDG) uses a temporally ordered sequence of source domains to capture the pattern of change and to predict on near-future domains (Nasery et al., 2021; Qin et al., 2022; Wang et al., 2020).

Such evolving distributions naturally arise in real-world applications such as disease diagnosis along with multiple clinical visits and autonomous driving where environmental factors such as symptoms and weather can change continuously during deployment. For example, a vehicle may encounter a gradual transition from pre-dawn darkness to daylight, producing a sequence of domains with progressively changing illumination. This type of evolution can be characterized by a relatively smooth temporal dependency, where the current domain state can be reasonably predicted from its recent history Zeng et al. (2023); Sun et al. (2022).

However, real-world environments are not necessarily governed by a single homogeneous evolution mechanism throughout time. Consider the driving sequence illustrated in Figure 1. During the first regime, the vehicle experiences a gradual illumination transition from before sunrise to sunrise and subsequently to bright daylight. The resulting domain shift can be represented as a smooth evolution of an illumination-related latent domain factor. At a later time, however, the environment may undergo an abrupt transition: while the scene remains bright, weather conditions suddenly become foggy and subsequently evolve toward light and heavy rain. The source of domain variation has therefore changed from illumination to precipitation. Consequently, the temporal dynamics before and after this transition are governed by different mechanisms.

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

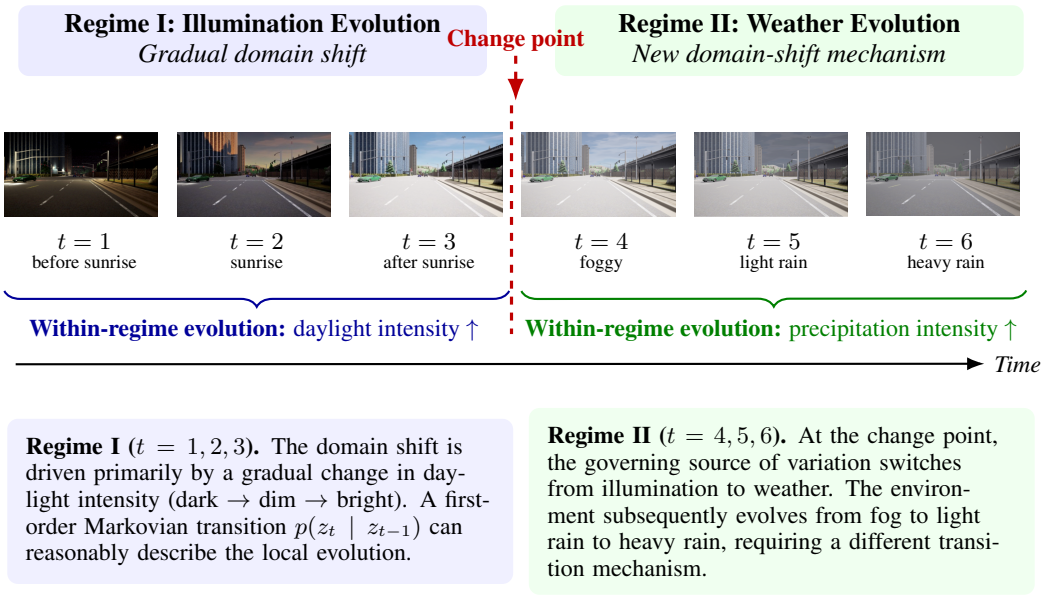

Figure 1: An illustrative EDG scenario in autonomous driving. In Regime I, the environment undergoes a gradual illumination-driven domain shift from before sunrise to daylight. At $t = 4$, an abrupt change in the environmental condition introduces a new weather-driven regime, followed by increasing precipitation intensity. Although local evolution within each regime may be modeled by a first-order Markov process, a single stationary transition model cannot adequately characterize the change in the underlying domain-shift mechanism. This motivates explicit Bayesian change-point inference for regime-aware EDG.

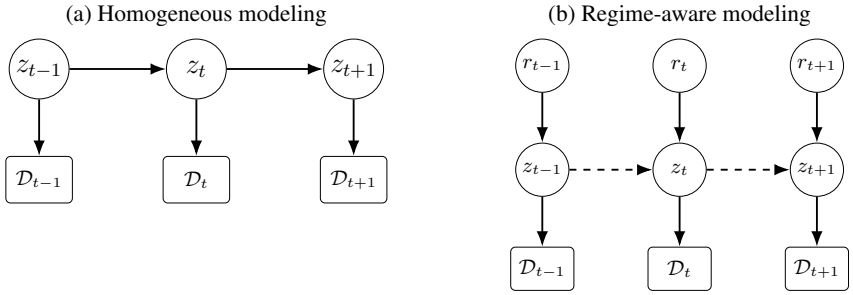

Figure 2: Comparison between homogeneous and regime-aware modeling of evolving domain dynamics. **(a)** Existing structure-aware EDG methods represent each observation using a dynamic domain latent z_t and commonly model its evolution through a homogeneous one-step transition, implicitly assuming that the same transition mechanism governs the entire sequence. **(b)** Our formulation introduces regime latents r_t to allow the transition mechanism itself to change over time. Domain evolution can therefore remain locally Markovian within each regime while exhibiting non-Markovian behavior marginally across regime changes.

Meanwhile, existing structure-aware EDG methods explain the evolving pattern through static task factors and dynamic domain factor (Qin et al., 2022; 2023; He et al., 2025), which corresponds to a latent dynamical system. However, they still assume a homogeneous transition $z_t = f(z_{t-1}) + \epsilon_t$ that treats domain dynamics statistically similar across time. From Figure 2, we can see that existing methods follow the Markovian assumption, while ours suggest a more general non-Markovian dynamics with regime changes governed by change-point variables.

In order to address the above challenges, our work combines within-regime drift with occasional changes of the transition itself that exists in many real-world sequential settings,

$$\underbrace{\mathcal{D}_1, \ldots, \mathcal{D}_{\tau-1}}_{\text{regime } k} \longrightarrow \underbrace{\mathcal{D}_\tau, \ldots}_{\text{regime } k+1}.$$

Inside a regime the latent may follow $z_t \approx f_k(z_{t-1})$. At a change point the map switches, $f_k \to f_{k+1}$. A single continuous random walk either oversmooths the jump or inflates process noise at every step, and the two effects are hard to separate from ordinary fluctuations. Our work models both levels of variation, which gradual evolution and abrupt regime transitions are inferred jointly, and the resulting change posterior is used for prediction after the change.

The overall model is a hierarchical switching state-space model which uses particle filtering for inference (SSSM-PF). A more complex stochastic process might be able to represent the same non-stationary dynamics, but its parameters would be numerous, difficult to interpret, and hard to estimate. The hierarchy instead factors the dynamics into a low-dimensional regime and a simple within-regime transition, which reduces the number of parameters and makes each factor interpretable, and also admits Bayesian inference with existing particle-filtering methods. A binary indicator $c_t \in \{0, 1\}$ records whether domain t continues the current regime or starts a new one. The regime r_t sets the transition statistics of a continuous domain state z_t, and the labeled batch $\mathcal{D}_t$ is generated from z_t. The generative structure is

$$c_t \;\to\; r_t \;\to\; p(z_t \mid z_{t-1}, r_t) \;\to\; \mathcal{D}_t.$$

The number of regimes is not fixed in advance. A hazard draw $c_t \sim \mathrm{Bernoulli}(H)$ starts a new regime, and the new parameters are sampled from a prior (Adams & MacKay, 2007; Fearnhead & Liu, 2007). The joint posterior $p(c_t, r_t, z_t \mid \mathcal{D}_{1:t})$ does not admit a closed form. We approximate it with a Rao-Blackwellized particle filter (RBPF) (Doucet et al., 2000; Schön et al., 2005). Particles carry the discrete change and regime history. A Kalman-like inner filter tracks $p(z_t \mid r_{1:t}^{(i)}, \mathcal{D}_{1:t})$. Classification likelihoods are not linear-Gaussian, so the inner measurement update uses importance-sampled moment matching. The same weights give the filtered change probability $P(c_t = 1 \mid \mathcal{D}_{1:t})$, which is the change-point score used in the experiments.

Our main contributions are as follows:

- Proposing a Bayesian switching state-space model formulation of EDG, in which latent domain representations evolve continuously within regimes and switch between regimes.
- Implementing a Rao-Blackwellized particle filter that estimates the continuous domain state and the posterior over regime changes, without a preset number of regimes.
- Designing abrupt EDG benchmarks with controlled gradual and abrupt shifts on both synthetic and real-world datasets, with evaluation of post-change generalization and change-point inference.
- Demonstrating the effectiveness of the proposed method on both original EDG benchmarks and the abrupt variants, showing higher performance than the state-of-the-art EDG baselines.

2 Related Work

Domain generalization and evolving domain generalization. Domain generalization (DG) trains predictors for unseen domains without target data, typically by invariant features (Blanchard et al., 2011; Li et al., 2018), meta-learning (Li et al., 2017; Dou et al., 2019), or augmentation (Volpi et al., 2018; Zhou et al., 2021; Zhang et al., 2018). Source domains are usually treated as samples from a static environment (Zhou et al., 2023). Evolving domain generalization (EDG) instead assumes temporally ordered shift (Nasery et al., 2021; Qin et al., 2022; Wang et al., 2020; He et al., 2025). Existing EDG methods include time-sensitive predictors (Nasery et al., 2021), structure-aware sequential autoencoders (Qin et al., 2022; 2023), continuous domain trajectories (Wang et al., 2020; Zeng et al., 2024), and time-aware causal mechanisms (He et al., 2025). Most of these models use amortized variational inference on pooled domains. The present work updates a domain state sequentially and adds an explicit change-point hierarchy.

Switching state-space models and change-point inference. Switching linear dynamical systems and jump-Markov models describe discrete mode changes in continuous dynamics (Murphy, 1998; Ghahramani & Hinton, 2000). Bayesian online change-point detection computes $P(c_t = 1 \mid x_{1:t})$ under a hazard process, often with conjugate run-length recursions (Adams & MacKay, 2007; Fearnhead & Liu, 2007; Chopin, 2007). Particle methods apply the same idea to nonlinear, non-Gaussian models (Gordon et al., 1993; Arulampalam et al., 2002; Kitagawa, 1996; Doucet et al., 2001; Särkkä, 2013). Rao-Blackwellized particle filters sample the discrete or strongly nonlinear part of the state and integrate a conditionally Gaussian remainder (Doucet et al., 2000; Schön et al., 2005). In the EDG filter below, particles carry change and regime histories, and an inner Gaussian filter tracks the continuous domain factor.

3 METHODOLOGY

3.1 PRELIMINARIES

Evolving domain generalization. Let $x \in \mathcal{X}$ and $y \in \mathcal{Y}$ denote the input and label. An evolving process $P(x, y, t)$ generates sequential domains $\{\mathcal{D}_t\}_{t=0}^{T-1}$. A prefix $\mathcal{S} = \{\mathcal{D}_t\}_{t=0}^{T_s-1}$ is available for training (source). Intermediate and target domains follow in temporal order. Each domain is a labeled batch

$$\mathcal{D}_t = \{(x_{t,i}, y_{t,i})\}_{i=1}^{N_t}, \qquad (x_{t,i}, y_{t,i}) \sim P(x, y \mid t).$$

The predictor is fit from the source prefix and is evaluated as t moves beyond that prefix (Qin et al., 2022; Nasery et al., 2021).

State-space models. A standard SSM specifies a latent Markov chain z_t with

$$\begin{aligned} z_0 &\sim \pi(z_0), \\ z_t \mid z_{t-1} &\sim p(z_t \mid z_{t-1}), \qquad t \geq 1, \\ \mathcal{D}_t &\sim p(\mathcal{D}_t \mid z_t). \end{aligned}$$

The filtering distribution $p(z_t \mid \mathcal{D}_{0:t})$ is the posterior of interest (Särkkä, 2013). In this work z_t is the dynamic domain factor (decision-boundary parameters, an image transform, or a temporal index). A homogeneous filter on z_t alone, without discrete regime variables, is the continuous-only ablation in Section 4.3.

3.2 HIERARCHICAL SWITCHING SSM FOR EDG

The continuous domain state is paired with a discrete regime. Let $c_t \in \{0, 1\}$ be a change indicator. The value $c_t = 0$ keeps the current regime. The value $c_t = 1$ starts a new regime. The regime index r_t obeys

$$r_t = \begin{cases} r_{t-1}, & c_t = 0, \\ r_{\text{new}}, & c_t = 1, \end{cases} \tag{1}$$

so the number of regimes is determined by the data. Each regime r has parameters $\theta_r = (\mu_r, \Sigma_r, \kappa_r)$ that set the statistics of domain evolution (Ghahramani & Hinton, 2000). Here μ_r and Σ_r are the drift and process covariance of z_t, and κ_r is a discrete shift type (center, radius, or concept on Circle).

Hazard process. Following Bayesian online change-point models (Adams & MacKay, 2007; Fearnhead & Liu, 2007),

$$c_t \sim \text{Bernoulli}(H), \tag{2}$$

with a constant hazard H. When $c_t = 1$, new regime parameters are drawn from a prior,

$$\theta_{r_t} \sim p(\theta), \qquad \kappa_{r_t} \sim \text{Unif}\{1, \ldots, K_{\text{type}}\}, \tag{3}$$

with an optional state jump $\Delta_t \sim \mathcal{N}(\mu_{\text{jump}}(\kappa_{r_t}), \Sigma_{\text{jump}}(\kappa_{r_t}))$. When $c_t = 0$, $\theta_{r_t} = \theta_{r_{t-1}}$. New regimes are created as changes occur, without setting K to the number of synthetic regimes.

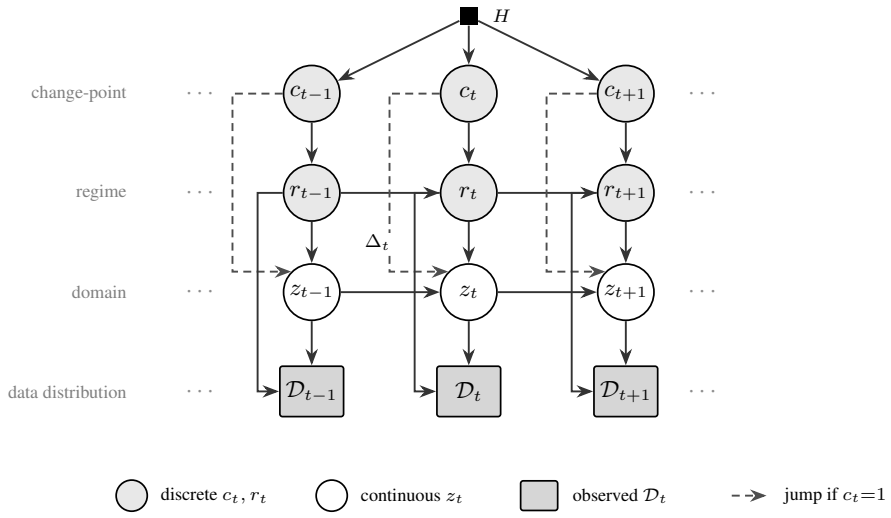

Figure 3: Graphical model of the hierarchical switching SSM. A constant hazard H draws independent $c_t \sim \text{Bernoulli}(H)$. Then c_t chooses whether r_t (carrying $\theta_r = (\mu_r, \Sigma_r, \kappa_r)$) is copied from r_{t-1} or replaced from the prior. The regime sets the drift of z_t; the dashed arrow is the jump Δ_t when $c_t = 1$. The batch $\mathcal{D}_t$ depends on (z_t, θ_{r_t}).

Continuous domain transition. Given the current regime, the domain state follows a drifted random walk

$$z_t = z_{t-1} + \mu_{r_t} + \epsilon_t, \qquad \epsilon_t \sim \mathcal{N}\big(0, \text{diag}(\sigma^2_{r_t})\big). \tag{4}$$

If $c_t = 1$, a jump Δ_t is applied before the predict step. The regime selects both the drift direction and the process scale. A slow regime uses a small σ_{r_t}. An abrupt shift uses a larger jump and process noise. On concept-reversal tasks, θ_r may also flip a binary invert bit that reverses the labeling rule.

Observation model. Given z_t, labels in a domain are conditionally independent,

$$p(\mathcal{D}_t \mid z_t, \theta_{r_t}) = \prod_{i=1}^{N_t} p(y_{t,i} \mid x_{t,i}, z_t, \theta_{r_t}). \tag{5}$$

The form of $p(y \mid x, z, \theta)$ depends on the task (Appendix B). Prediction marginalizes the filtering posterior,

$$p(y \mid x, \mathcal{D}_{0:t}) = \int p(y \mid x, z, \theta)\, p(z, \theta \mid \mathcal{D}_{0:t})\, dz\, d\theta. \tag{6}$$

Inference targets the joint posterior $p(c_t, r_t, z_t \mid \mathcal{D}_{0:t})$. The marginal

$$P(c_t = 1 \mid \mathcal{D}_{0:t}) \tag{7}$$

is reported at each domain as a Bayesian change-point score. Figure 3 shows the generative graph.

3.3 Rao-Blackwellized particle filtering

Joint sampling of (c_t, r_t, z_t) spends particles on the continuous state as well as on the discrete history. An RBPF (Doucet et al., 2000; Schön et al., 2005) represents the discrete part by particles and approximates the continuous remainder by a Gaussian. Particle i stores

$$\big(c_t^{(i)},\, \theta_{r_t}^{(i)},\, \kappa_t^{(i)},\, m_t^{(i)},\, P_t^{(i)}\big),$$

where $(m_t^{(i)}, P_t^{(i)})$ are the mean and diagonal covariance of $p(z_t \mid c_{1:t}^{(i)}, r_{1:t}^{(i)}, \mathcal{D}_{0:t}) \approx \mathcal{N}(m_t^{(i)}, \text{diag}(P_t^{(i)}))$. The joint filtering distribution is the mixture

$$p(z_t, r_t \mid \mathcal{D}_{0:t}) = \sum_{i=1}^{K} W_t^{(i)}\, \delta_{r_t^{(i)}}(r_t)\, \mathcal{N}\big(z_t; m_t^{(i)}, \text{diag}(P_t^{(i)})\big). \tag{8}$$

Predict. For $t \geq 1$ and each particle, draw $c_t^{(i)} \sim \text{Bernoulli}(H)$. If $c_t^{(i)} = 1$, sample a new regime as in (3) and add the corresponding jump to $m_{t-1}^{(i)}$. The Kalman predict implied by (4) is

$$m_{t|t-1}^{(i)} = m_{t-1}^{(i)} + \mu_{r_t}^{(i)}, \qquad P_{t|t-1}^{(i)} = P_{t-1}^{(i)} + \big(\sigma_{r_t}^{(i)}\big)^2, \tag{9}$$

after which a task-specific constraint is applied (minimum radius, wrapped angles, or a clipped temporal index).

Inner measurement update. The observation $p(\mathcal{D}_t \mid z_t)$ is not linear-Gaussian, so a Kalman measurement update does not apply. We draw L inner samples $z^{(i,\ell)} \sim \mathcal{N}(m_{t|t-1}^{(i)}, \text{diag}(P_{t|t-1}^{(i)}))$, evaluate $\log p(\mathcal{D}_t \mid z^{(i,\ell)})$, and moment-match a Gaussian to the importance-weighted inner cloud. The inner marginal likelihood

$$p(\mathcal{D}_t \mid r_{1:t}^{(i)}, \mathcal{D}_{0:t-1}) \approx \frac{1}{L} \sum_{\ell=1}^{L} p\big(\mathcal{D}_t \mid z^{(i,\ell)}\big) \tag{10}$$

is the incremental particle weight,

$$\log \tilde{w}_t^{(i)} = \log w_{t-1}^{(i)} + \log p(\mathcal{D}_t \mid r_{1:t}^{(i)}, \mathcal{D}_{0:t-1}). \tag{11}$$

Normalized weights $W_t^{(i)} \propto \tilde{w}_t^{(i)}$ approximate $p(r_{1:t}^{(i)} \mid \mathcal{D}_{0:t})$.

Change-point readout and resampling. The filtered change probability is the particle average of the current-step indicators,

$$P(c_t = 1 \mid \mathcal{D}_{0:t}) = \sum_{i=1}^{K} W_t^{(i)}\, c_t^{(i)}, \tag{12}$$

computed before resampling, so that ancestor duplication does not collapse the estimate. If the effective sample size $\text{ESS} = (\sum_i (W_t^{(i)})^2)^{-1}$ falls below ρK (with $\rho = 0.5$), systematic resampling is applied and weights are reset to $1/K$.

Prediction. Labels mix the observation model under the inner means,

$$p(y \mid x, \mathcal{D}_{0:t}) \approx \sum_{i=1}^{K} W_t^{(i)}\, p\big(y \mid x, m_t^{(i)}, \theta_{r_t}^{(i)}\big).$$

Algorithm 1 states the procedure. Dropping (c_t, θ_r) and running a bootstrap filter on z_t yields the continuous-only ablation in Section 4.3.

Algorithm 1 Rao-Blackwellized PF for switching EDG.

Input domain stream $\{\mathcal{D}_t\}$, particles K, inner samples L, hazard H
Output filtered means, $P(c_t = 1 \mid \mathcal{D}_{0:t})$, and predictions

Sample $m_0^{(i)}, P_0^{(i)}$ and $\theta_{r_0}^{(i)}$; set $w_0^{(i)} = 1/K$, $c_0^{(i)} = 0$
for $t = 0$ **to** $T - 1$ **do**
  if $t > 0$ **then**
    $c_t^{(i)} \sim \text{Bernoulli}(H)$; if $c_t^{(i)} = 1$, sample new $\theta_{r_t}^{(i)}$ and jump $m^{(i)}$
    Kalman predict $m^{(i)} \leftarrow m^{(i)} + \mu^{(i)}$, $P^{(i)} \leftarrow P^{(i)} + (\sigma^{(i)})^2$
  end if
  Draw inner samples $z^{(i,\ell)} \sim \mathcal{N}(m^{(i)}, \text{diag}(P^{(i)}))$, $\ell = 1, \ldots, L$
  Moment-match $(m^{(i)}, P^{(i)})$ to the likelihood-weighted inner cloud
  $w_t^{(i)} \propto w_{t-1}^{(i)} \cdot \frac{1}{L} \sum_\ell p(\mathcal{D}_t \mid z^{(i,\ell)})$
  Record $P(c_t = 1 \mid \mathcal{D}_{0:t}) = \sum_i W_t^{(i)} c_t^{(i)}$ before resampling
  Resample if $\text{ESS} < \rho K$; predict labels by mixing $p(y \mid x, m^{(i)}, \theta^{(i)})$
end for

Task-specific latent states, observation families, and the change-point scoring protocol are given in Appendix B.

4 EXPERIMENTS

4.1 ABRUPT EDG BENCHMARKS

Domain counts and source, intermediate, and target splits follow prior EDG work (Qin et al., 2022; He et al., 2025). Smooth evolution is replaced by controlled abrupt changes (Table 1). Synthetic Circle and Sine streams use the shift types of Appendix B. Jumps are placed after the source split so that labeled training does not observe the later regimes. RMNIST keeps source angles on $0°$-$90°$ and then jumps to $180°$ and $270°$. PowerSupply and CalTran permute the original domain order by a block-swap. If the natural timeline is $0, \ldots, n-1$ and cuts (a, b) are given, the presentation order is

$$0, \ldots, a-1, \quad b, \ldots, n-1, \quad a, \ldots, b-1.$$

The permutation inserts two discontinuities and leaves each block internally contiguous. Portraits keeps source and intermediate domains in original order and reverses the target block. Figure 4 shows the Circle stream. Geometry drifts slowly, radius dynamics change at $t{=}10$, and labels reverse at $t{=}20$.

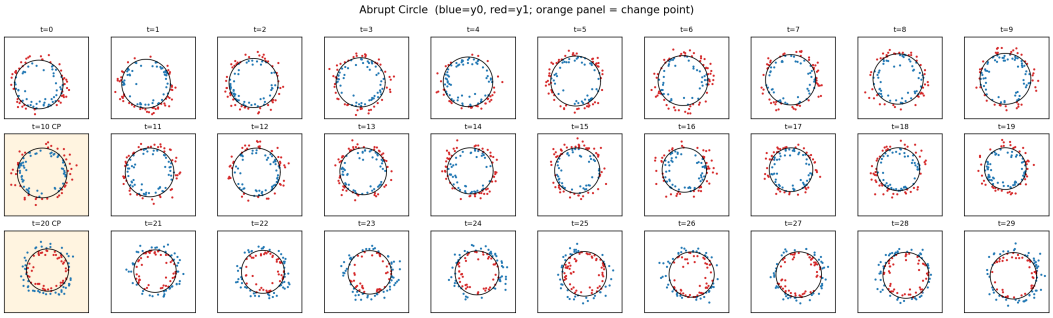

Figure 4: Abrupt Circle domains. Blue is $y{=}0$ (inside) and red is $y{=}1$ (outside). Orange panels mark ground-truth change points. After $t{=}20$ the labeling rule reverses while the circle continues to drift.

Dataset	#Dom.	Change points	Shift sequence	Split (src/int/tgt)
Circle	30	10, 20	center → radius → concept	15 / 5 / 10
Sine	24	12, 16	phase → amplitude → frequency	12 / 4 / 8
RMNIST	19	10, 13	$0°$-$90°$ → $180°$ jump → $270°$ jump	10 / 3 / 6
PowerSupply	30	10, 20	block-swap 0-9, 20-29, 10-19	15 / 5 / 10
Portraits	34	24, 29	source 0-18; reverse target 33. . .24	19 / 5 / 10
CalTran	34	11, 22	block-swap 0-10, 23-33, 11-22	19 / 5 / 10

Table 1: Abrupt EDG benchmarks. Indices are 0-based. Image models subsample a mini-batch per domain for the filter update.

Implementation. Particle counts, inner-sample sizes, hazard, polynomial degree, and source-classifier training details are listed in Appendix C. Source classifiers, when used, are trained only on source domains. Target labels are not used to set SSM hyperparameters. Comparison methods are LSSAE (Qin et al., 2022), MMD-LSAE (Qin et al., 2023), SYNC (He et al., 2025), and SDE-EDG (Zeng et al., 2024), trained on the same original and abrupt pickles. Mean target-domain accuracy is the generalization metric. Change-point scores are reported for SSSM-PF.

4.2 MAIN RESULTS

Table 2 reports mean target-domain accuracy. Avg is the mean over the six datasets in each block; Total averages those twelve scores. Per-domain numbers are in Appendix F, and domain-generalization and transformer baselines on the original datasets are in Appendix E.

Table 2: Mean target-domain accuracy (%). Left: original EDG datasets and their average. Right: abrupt EDG datasets and their average. Total is the mean of the twelve dataset scores. Best entry in each column is in bold; second best is underlined. Our method is SSSM-PF which is on the last row.

	Original							Abrupt							
Method	Circle	Sine	Power	RMNIST	Portr.	CalTr.	Avg	Circle	Sine	Power	RMNIST	Portr.	CalTr.	Avg	Total
LSSAE	70.20	57.33	71.53	35.72	**87.94**	58.21	63.49	38.50	81.50	71.84	45.18	87.89	52.59	62.92	63.20
MMD-LSAE	64.70	59.63	70.99	36.75	68.80	45.75	57.77	39.90	82.12	71.44	46.25	68.77	43.75	58.71	58.24
SYNC	83.40	73.50	71.64	46.52	87.70	66.63	71.57	37.60	81.38	72.39	52.64	87.79	67.99	66.63	69.10
SDE	63.80	59.37	71.03	44.19	87.51	61.87	64.63	44.00	86.00	69.52	44.50	87.46	69.58	66.84	65.74
SSSM-PF	**99.90**	**97.82**	**76.88**	**97.13**	87.20	**72.86**	**88.63**	**95.90**	**94.88**	**77.85**	**58.85**	**88.99**	**78.99**	**82.58**	**85.60**

On original streams SSSM-PF averages 88.63%, against 71.57% for SYNC. The gaps are largest on Circle (99.90 vs. 83.40) and RMNIST (97.13 vs. 46.52), where the filter tracks a low-dimensional state that pooled models miss. PowerSupply is closer (76.88 vs. 71.64). On Portraits, LSSAE (87.94) slightly exceeds SSSM-PF (87.20).

On abrupt streams SSSM-PF averages 82.58%, against 66.84% for SDE-EDG. After the Circle concept reversal, SSSM-PF stays at 95.90% while every baseline falls below 45%: the invert bit flips $P(Y \mid X)$ and z_t keeps the geometry. Smaller leads remain on Sine (94.88 vs. 86.00), RMNIST (58.85 vs. 52.64), CalTran (78.99 vs. 69.58), and PowerSupply (77.85 vs. 72.39). Portraits is close (88.99 vs. 87.89 for LSSAE). Source accuracy stays high (Appendix D), so the hazard does not destabilize tracking when no jump is present.

4.3 Change-point detection

Table 3 drops (c_t, θ_r) and filters z_t with a bootstrap particle filter, using the same observation family and particle count.

Table 3: Ablation of the change-point latent. Mean target-domain accuracy (%). The continuous-only row drops (c_t, θ_r) and filters z_t with a bootstrap particle filter. SSSM-PF adds the discrete change and regime variables. Δ is SSSM-PF minus the continuous-only control. Best entry in each column is in bold; second best is underlined.

	Original							Abrupt						
Method	Circle	Sine	Power	RMNIST	Portr.	CalTr.	Avg	Circle	Sine	Power	RMNIST	Portr.	CalTr.	Avg
w/o change-point	99.30	93.79	74.13	77.73	**88.18**	**73.27**	84.40	43.70	94.00	77.28	48.77	82.82	74.07	70.11
with change-point	**99.90**	**97.82**	**76.88**	**97.13**	87.20	72.86	**88.63**	**95.90**	**94.88**	**77.85**	**58.85**	**88.99**	**78.99**	**82.58**
Δ	+0.60	+4.03	+2.75	+19.40	-0.98	-0.41	+4.23	+52.20	+0.88	+0.57	+10.08	+6.17	+4.92	+12.47

The change bit helps when z_t cannot absorb the jump. On original streams the mean gain is $+4.23$ (88.63 vs. 84.40), mostly from RMNIST ($+19.40$); Portraits and CalTran do not improve, because a continuous temporal index already matches that timeline. On abrupt streams the gain is $+12.47$ (82.58 vs. 70.11), driven by Circle ($+52.20$), where the continuous filter cannot invert the label rule. Sine and PowerSupply stay within one point; RMNIST, Portraits, and CalTran gain $+10.08$, $+6.17$, and $+4.92$.

Figure 5 shows $P(c_t = 1 \mid \mathcal{D}_{1:t})$. Sine and polynomial Circle place that mass on the true times with no delay. RMNIST detects both jumps and adds later spikes in the reverse segment.

5 Limitations and future work

The experiments distinguish two operating regimes. When the observation model isolates the coordinate that jumps, as on polynomial Circle and parametric Sine, the filter recovers post-change accuracy and places mass of $P(c_t = 1 \mid \mathcal{D}_{0:t})$ at the true change times. When z_t is a flexible quadratic coefficient vector or a temporal index, as on PowerSupply, Portraits, and CalTran, the continuous state can absorb the jump without a regime switch. Target accuracy can still exceed that of a pooled EDG baseline (Table 2). In this case c_t indicates that the emission is too flexible to identify the change. In deployment, predictions may therefore be taken from the filter by default,

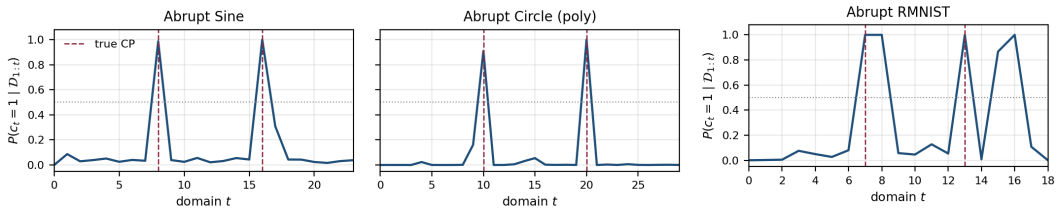

Figure 5: Filtered change probability $P(c_t = 1 \mid \mathcal{D}_{1:t})$ (solid) and ground-truth change points (dashed). Sine and polynomial Circle recover the true times with no delay. RMNIST detects both jumps and later false positives.

while $P(c_t = 1 \mid \mathcal{D}_{0:t})$ should be used as a change alarm only after the observation model has been validated on a stream with known jumps.

The emission is specified per task, where monomial features on Circle and Sine, an inverse similarity warp on RMNIST, and a calibrated temporal index on Portraits and CalTran. A direct extension is a learned emission, trained on source domains only and regularized so that z_t remains a low-dimensional domain factor. The switching backbone and the RBPF would remain unchanged.

The abrupt streams are constructed manually. For the real-world tasks, the abrupt presentation order is obtained by reordering the original domains: a block swap on PowerSupply and CalTran, and a reversed target block on Portraits. The change times are therefore known by construction. Applying the same filter to streams whose regime changes occur in the recorded timeline is a natural next step.

Two further components of the generative model can be replaced independently. The constant hazard H assigns the same switch probability to every domain. A run-length or time-varying hazard can replace (2) without modifying the inner Gaussian filter. Tighter priors on θ_r would move residual jump mass onto c_t for the block-swapped streams, where a flexible z_t currently accounts for the discontinuity. Neither change requires pooling target domains or retraining an amortized sequence model.

The RBPF scales linearly in the number of domains and in the particle count. Image-domain updates already use a subsampled batch. Longer streams, or a modestly higher-dimensional z_t, therefore remain standard filtering problems. This matches the EDG protocol, in which domains arrive in order and a prediction is required before the next domain is observed.

6 Conclusion

Generalization problems on evolving domains don't always follow a one-step Markovian assumption, and they can be formulated as hierarchical switching state-space processes. Our work uses a hazard driven change-point indicator to govern a new regime, followed by the regime which sets the drift of a continuous domain state, and at last generates the observations from that latent state. We utilize Rao-Blackwellized particle filtering to update the joint posterior on the discrete change, regime history and the continuous domain factor. On the original gradual EDG tasks the switching model remains the strongest method we compare, while on the abrupt streams we designed, it detects the turning points and improves post-change prediction therefore improves the generalization performance significantly. Our work establishes a valuable connection between transfer learning and Bayesian learning, while providing a robust baseline for model generalization in evolving environments.

AI use statement

In this work, we used generative AI tools for revising and polishing manuscript text based on our non-AI initial draft in order to provide a better presentation of the work. We have not used generative AI tools to invent any theoretical claim or experimental results. The research ideas including the complete design of hierarchical switching SSM, the usage of RBPF inference, and design of

abrupt EDG benchmarks all originate from the authors own knowledge and understanding. We take responsibility for the final content of this work.

Ethics statement

This work studies on publicly released benchmarks and their varients. No new human-subject data were collected.

Reproducibility statement

Code for dataset generation, switching SSMs, the RBPF, and evaluation metrics accompanies the submission. Sections 3.2 and 3.3 specify the generative model and filter. Appendix B lists latent parameterizations. Appendix C lists numerical hyperparameters. Table 1 and Appendix A describe abrupt dataset construction. Appendix D reports per-split accuracies that underlie Table 2. Appendix F lists intermediate-domain and per-domain accuracy tables.

References

Ryan Prescott Adams and David J. C. MacKay. Bayesian online changepoint detection. *arXiv preprint arXiv:0710.3742*, 2007.

M. Sanjeev Arulampalam, Simon Maskell, Neil Gordon, and Tim Clapp. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Transactions on Signal Processing*, 50(2):174–188, 2002.

Gilles Blanchard, Gyemin Lee, and Clayton Scott. Generalizing from several related classification tasks to a new unlabeled sample. In *Advances in Neural Information Processing Systems*, 2011.

Nicolas Chopin. Dynamic detection of change points in long time series. *Annals of the Institute of Statistical Mathematics*, 59(2):349–366, 2007.

Hoang Anh Dau, A. Bagnall, Kaveh Kamgar, Chin-Chia Michael Yeh, Yan Zhu, Shaghayegh Gharghabi, Chotirat Ann Ratanamahatana, and Eamonn J. Keogh. The ucr time series archive. *IEEE/CAA Journal of Automatica Sinica*, 6:1293–1305, 2018.

Qi Dou, Daniel Coelho de Castro, Konstantinos Kamnitsas, and Ben Glocker. Domain generalization via model-agnostic learning of semantic features. *Advances in neural information processing systems*, 32, 2019.

Arnaud Doucet, Nando de Freitas, Kevin Murphy, and Stuart Russell. Rao-Blackwellised particle filtering for dynamic Bayesian networks. In *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence*, pp. 176–183, 2000.

Arnaud Doucet, Nando de Freitas, and Neil Gordon. Sequential Monte Carlo methods in practice. *Springer*, 2001.

Paul Fearnhead and Zhen Liu. On-line inference for multiple changepoint problems. *Journal of the Royal Statistical Society: Series B*, 69(4):589–605, 2007.

Zoubin Ghahramani and Geoffrey E. Hinton. Variational learning for switching state-space models. *Neural Computation*, 12(4):831–864, 2000.

Muhammad Ghifary, W. Bastiaan Kleijn, Mengjie Zhang, and David Balduzzi. Domain Generalization for Object Recognition with Multi-task Autoencoders . In *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 2551–2559, Los Alamitos, CA, USA, December 2015. IEEE Computer Society.

Shiry Ginosar, Kate Rakelly, Sarah Sachs, Brian Yin, and Alexei A. Efros. A century of portraits: A visual historical record of american high school yearbooks. In *2015 IEEE International Conference on Computer Vision Workshop (ICCVW)*, pp. 652–658, 2015.

Neil J. Gordon, David J. Salmond, and Adrian F. M. Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proceedings F – Radar and Signal Processing*, 140(2): 107–113, 1993.

Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. In *International Conference on Learning Representations*, 2021.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *International Conference on Machine Learning*, 2017.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

Zhuo He, Shuang Li, Wenze Song, Longhui Yuan, Jian Liang, Han Li, and Kun Gai. Learning time-aware causal representation for model generalization in evolving domains. In *International Conference on Machine Learning*, pp. 22523–22547. PMLR, 2025.

Judy Hoffman, Trevor Darrell, and Kate Saenko. Continuous Manifold Based Adaptation for Evolving Visual Domains . In *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 867–874, Los Alamitos, CA, USA, June 2014. IEEE Computer Society.

Maximilian Ilse, Jakub M. Tomczak, Christos Louizos, and Max Welling. DIVA: Domain invariant variational autoencoders. In *Conference on Medical Imaging with Deep Learning*, 2020.

Genshiro Kitagawa. Monte Carlo filter and smoother for non-Gaussian nonlinear state space models. *Journal of Computational and Graphical Statistics*, 5(1):1–25, 1996.

Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, Tony Lee, Etienne David, Ian Stavness, Wei Guo, Berton Earnshaw, Imran Haque, Sara Beery, Jure Leskovec, Anshul Kundaje, Emma Pierson, Sergey Levine, Chelsea Finn, and Percy Liang. WILDS: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning*, 2021.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. In *Proceedings of the IEEE*, volume 86, pp. 2278–2324, 1998.

Bo Li, Yifei Shen, Jingkang Yang, Yezhen Wang, Jiawei Ren, Tong Che, Jun Zhang, and Ziwei Liu. Sparse mixture-of-experts are domain generalizable learners. In *International Conference on Learning Representations*, 2023.

Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M. Hospedales. Learning to generalize: Meta-learning for domain generalization, 2017.

Haoliang Li, Sinno Jialin Pan, Shiqi Wang, and Alex C Kot. Domain generalization with adversarial feature learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5400–5409, 2018.

Kevin P. Murphy. Switching Kalman filters. *Technical report, UC Berkeley*, 1998.

Anshul Nasery, Harsh Thakur, and Moritz Hardt. Training for the future: A simple gradient interpolation loss to generalize along temporal domains. In *Advances in Neural Information Processing Systems*, 2021.

Tiexin Qin, Shiqi Wang, and Haoyu Li. Generalizing to evolving domains with latent structure-aware sequential autoencoder. In *International Conference on Machine Learning*, 2022.

Tiexin Qin, Shiqi Wang, and Haoliang Li. Evolving domain generalization via latent structure-aware sequential autoencoder. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–14, 2023.

Simo Särkkä. Bayesian filtering and smoothing. *Cambridge University Press*, 2013.

Thomas Schön, Fredrik Gustafsson, and Per-Johan Nordlund. Marginalized particle filters for mixed linear/nonlinear state-space models. *IEEE Transactions on Signal Processing*, 53(7):2279–2289, 2005.

Tao Sun, Mattia Segu, Janis Postels, Yuxuan Wang, Luc Van Gool, Bernt Schiele, Federico Tombari, and Fisher Yu. Shift: a synthetic driving dataset for continuous multi-task domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21371–21382, 2022.

Vladimir N. Vapnik. An overview of statistical learning theory. *IEEE Transactions on Neural Networks*, 10(5):988–999, 1999.

Riccardo Volpi, Hongseok Namkoong, Ozan Sener, John C Duchi, Vittorio Murino, and Silvio Savarese. Generalizing to unseen domains via adversarial data augmentation. *Advances in neural information processing systems*, 31, 2018.

Hao Wang, Hao He, and Dina Katabi. Continuously indexed domain adaptation. In *International Conference on Machine Learning*, 2020.

Qiuhao Zeng, Wei Wang, Fan Zhou, Charles Ling, and Boyu Wang. Foresee what you will learn: data augmentation for domain generalization in non-stationary environment. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pp. 11147–11155, 2023.

Qiuhao Zeng, Changjian Shui, Long-Kai Huang, Peng Liu, Xi Chen, Charles Ling, and Boyu Wang. Latent trajectory learning for limited timestamps under distribution shift over time. In *The Twelfth International Conference on Learning Representations*, 2024.

Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018.

Zangwei Zheng, Xiangyu Yue, Kai Wang, and Yang You. Prompt vision transformer for domain generalization. *arXiv preprint arXiv:2208.08914*, 2022.

Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. Domain generalization with mixstyle. In *International Conference on Learning Representations*, 2021.

Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. Domain generalization: A survey. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 45, pp. 4396–4415, 2023.

APPENDIX

The following sections are supplementary material. The appendix documents dataset construction, task-specific instantiations, hyperparameters, and additional accuracy tables.

A DATASET CONSTRUCTION

Abrupt Circle. Each of T=30 domains contains 100 points sampled near the current circle. The latent $s_t = (c_x, c_y, r)$ starts at $(0, 0.5, 3.5)$. Regime 1 (domains 0-9) applies a slow center drift $\mu = (0.10, 0.10, 0.02)$. At t=10 an abrupt jump $(-0.50, 0.40, -0.08)$ occurs and drift becomes radius-dominated. At t=20 a further jump is combined with a concept reversal, and labels switch from outside to inside. The sequence is covariate shift, then boundary shift, then concept shift.

Abrupt Sine. Domains t=0, . . . , 23 are generated from $y = A\sin(\omega x + \phi) + b$ with a slow phase drift of 0.10 rad/domain. Source domains 0-11 keep A=1, ω=1. At t=12, amplitude jumps from 1.0 to 1.5. At t=16, frequency jumps from 1.0 to 1.5. Phase continues to evolve in every regime. Figure 6 shows the corresponding decision boundaries.

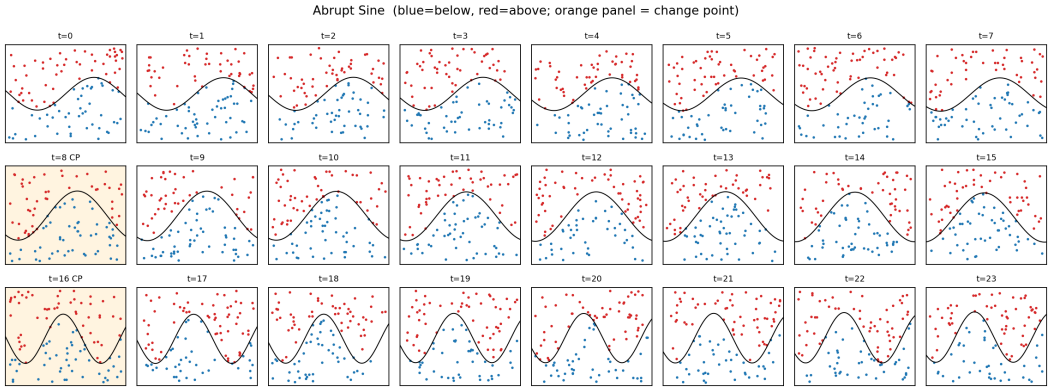

Figure 6: Abrupt Sine domains. Blue is y=0 (below the curve) and red is y=1 (above). Orange panels mark ground-truth change points.

Abrupt RMNIST. MNIST digits are split across 19 domains. Source domains 0-9 use $0°, 10°, \ldots, 90°$. At t=10 the angle jumps to 180° and then 195°, 210°. At t=13 it jumps to 270° and continues $285°, \ldots, 345°$. A similarity variant adds translation and scale jumps at the same change points.

Block-swap and reverse-tail streams. For PowerSupply (n=30, cuts 10, 20) the presentation order is original domains 0-9, 20-29, 10-19. For CalTran (n=34, cuts 11, 22) it is 0-10, 23-33, 11-22. For Portraits, source domains keep original ids 0-18, intermediate 19-23, and the target block is reversed original $33, \ldots, 24$, with change points at t=24 and t=29.

B TASK-SPECIFIC INSTANTIATIONS

The switching backbone is shared. Only the meaning of z_t and $p(y \mid x, z, \theta)$ changes. Table 4 lists the instantiations. Shift-type priors $p(\theta \mid \kappa)$ place mass on a particular kind of domain change.

Polynomial boundaries. Without assuming any function families (e.g., circle or sine family), z_t is a coefficient vector on source-standardized monomials of total degree D. The map $\phi_D(x)$ enumerates $x_1^a x_2^b$ with $1 \le a + b \le D$, and $p(y{=}1 \mid x, z) = \sigma(\beta(w^\top \phi_D(x) + b))$. Regimes jump degree-1 coefficients, higher-degree coefficients and bias, or (on Circle) invert the label rule. The main tables use D=4 on both Circle and Sine.

Dataset	Latent z_t	Shift types κ	Observation $p(y \mid x, z)$
Poly. Circle/Sine	(w, b) on $\phi_D(x)$	linear / higher / concept	$\sigma(\beta(w^\top \phi_D(x) + b))$
PowerSupply	quadratic (w, b)	linear / quadratic / bias	$\sigma(\beta(w^\top \phi(x) + b))$
RMNIST	$(\theta, t_x, t_y, \log s, \eta)$	rot. / trans. / scale	CNN after gated inverse warp
Portraits/CalTran	τ (time index)	continue / skip-fwd / skip-back	$\mathrm{softmax}((f_\omega(x) + b(\tau))/T(\tau))$

Table 4: Switching SSM instantiations. Dynamics follow (4). ϕ_D denotes source-normalized monomials. Image classifiers f_ω are trained on source domains only. Reported Circle/Sine numbers use the polynomial family with $D{=}4$.

PowerSupply. Hourly electricity records form 30 day-domains with a.m./p.m. labels (Dau et al., 2018). A fixed quadratic map $\phi(x) = \mathrm{standardize}([x_1, x_2, x_1^2, x_2^2, x_1 x_2])$ yields $z_t \in \mathbb{R}^6$. Regimes jump linear, quadratic, or bias coordinates.

Rotated-MNIST. A source-trained digit CNN f_ω is a static predictor (LeCun et al., 1998). The latent z_t is the warp inverted before classification. The similarity SSM expands z_t to $(\theta, t_x, t_y, \log s)$ plus mixture logits that gate inverse rotation, translation, and scale (Ghifary et al., 2015).

Temporal vision (Portraits, CalTran). When the domain factor is time rather than an explicit geometric parameter, $z_t = \tau_t \in \mathbb{R}$ is a continuous temporal index (Ginosar et al., 2015; Hoffman et al., 2014). A ResNet18 classifier (He et al., 2016) is trained on source domains and calibrated with piecewise-linear temperature and class-bias knots (Guo et al., 2017),

$$p(y \mid x, \tau) = \mathrm{softmax}\left(\frac{f_\omega(x) + b(\tau)}{\max(T(\tau), \varepsilon)}\right)_y .$$

Skip-forward and skip-back regimes place large positive or negative jumps on τ. A continue regime keeps a unit drift.

C Hyperparameters

All SSM hyperparameters are set from source domains and then frozen. The hazard is $H{=}0.08$ for every dataset.

Particle filter. Reported Circle and Sine runs (SSSM-PF and the continuous-only ablation) use the polynomial observation with degree $D{=}4$, $K{=}600$ particles, and $L{=}8$ inner samples. PowerSupply uses $K{=}400$ and $L{=}8$. RMNIST, Portraits, and CalTran use $K{=}80$ and $L{=}4$ with mini-batch subsampling of 256 examples per domain (RMNIST, Portraits) or 128 (CalTran). The ESS resampling threshold is $\rho{=}0.5$ with systematic resampling. Inner diagonal covariances are clipped at 4, and states are constrained after both predict and update.

Observation and classifiers. Default process scales are $\sigma = (0.05, 0.05, 0.02)$ for Circle geometry, with regime-specific overrides at changes; $\sigma_\theta{=}3^\circ$ for RMNIST; $\sigma_\tau{=}0.35$ for Portraits and CalTran; and a small diagonal σ on polynomial and PowerSupply coefficients (0.08 when source IRLS estimation is skipped). Logistic sharpness is $\beta{=}16$. Image CNNs use 5 epochs (RMNIST, Portraits) or 8 epochs (CalTran) of source-only training. CalTran inputs are $84{\times}84$. Portraits uses a pretrained ResNet18 backbone.

D Per-split accuracy

Table 5 reports source, intermediate, and target accuracy together. Source accuracy stays high on every stream, including abrupt Circle and RMNIST, so the hazard process does not collapse tracking before a jump. The CalTran source number (73.64% original, 69.25% abrupt) is limited by the static ResNet and the $84{\times}84$ resolution. Later domains on the abrupt stream reach 78.99% target accuracy after the filter tracks τ.

	Original			Abrupt		
Dataset	Source	Inter.	Target	Source	Inter.	Target
Circle	99.60	99.80	99.90	95.13	98.60	95.90
Sine	96.80	99.18	97.82	98.33	99.75	94.88
PowerSupply	77.31	78.34	76.88	77.46	75.52	77.85
RMNIST	98.03	97.50	97.13	98.26	77.47	58.85
Portraits	99.19	73.10	87.20	98.65	74.30	88.99
CalTran	73.64	77.82	72.86	69.25	70.12	78.99

Table 5: Mean domain accuracy (%) of SSSM-PF on source, intermediate, and target splits. These are the same runs as Table 2.

E Domain-generalization and transformer baselines

Table 6 compares domain-generalization and transformer-based methods with the EDG methods on the original datasets. The EDG rows, including SSSM-PF, are the original-dataset columns of Table 2.

Table 6: Mean target-domain accuracy (%) on the original EDG datasets. The first block is domain generalization, the second is transformer-based, and the third is the EDG methods from Table 2. Best entry in each column is in bold; second best is underlined.

	Circle	Sine	RMNIST	Portraits	CalTran	PowerSup.	Avg
ERM (Vapnik, 1999)	49.9	63.0	43.6	87.8	66.3	62.1	62.1
Mixup (Zhang et al., 2018)	48.4	62.9	44.9	87.8	66.0	62.0	62.0
MMD (Li et al., 2018)	50.7	55.8	44.8	87.3	57.1	59.1	59.1
DIVA (Ilse et al., 2020)	67.9	52.9	42.7	**88.2**	69.2	64.2	64.2
DoPrompt (Zheng et al., 2022)	52.3	63.3	50.2	**88.2**	<u>70.4</u>	70.6	65.8
GMoE (Li et al., 2023)	57.9	62.4	<u>52.5</u>	<u>87.9</u>	70.1	71.0	67.0
LSSAE (Qin et al., 2022)	70.2	57.3	35.7	<u>87.9</u>	58.2	71.5	63.5
MMD-LSAE (Qin et al., 2023)	64.7	59.6	36.8	68.8	45.8	71.0	57.8
SDE-EDG (Zeng et al., 2024)	63.8	59.4	44.2	87.5	61.9	71.0	64.6
SYNC (He et al., 2025)	<u>83.4</u>	<u>73.5</u>	46.5	87.7	66.6	<u>71.6</u>	<u>71.6</u>
SSSM-PF	**99.9**	**97.8**	**97.1**	87.2	**72.9**	**76.9**	**88.6**

F Additional accuracy tables

All accuracies are in percent. Avg is the arithmetic mean of the domains in that table. These tables complement the target-domain summary in Table 2. The best entry in each column is in bold and the second best is underlined.

F.1 Intermediate-domain averages

Table 7: Average intermediate-domain accuracy (%) on original datasets.

Method	Circle	Sine	PowerSupply	RMNIST	Portraits	CalTran	Avg
LSSAE	93.60	83.89	72.98	73.65	**73.63**	65.49	77.21
MMD-LSAE	<u>97.40</u>	<u>86.87</u>	<u>72.98</u>	73.44	62.54	40.02	72.21
SYNC	91.60	75.77	72.24	86.34	71.62	<u>72.24</u>	<u>78.30</u>
SDE	84.00	61.64	72.86	<u>90.70</u>	72.25	68.14	74.93
SSSM-PF	**99.80**	**99.17**	**78.34**	**97.50**	<u>73.10</u>	**77.82**	**87.62**

Table 8: Average intermediate-domain accuracy (%) on abrupt datasets.

Method	Circle	Sine	PowerSupply	RMNIST	Portraits	CalTran	Avg
LSSAE	76.60	89.50	71.53	43.55	73.63	50.95	67.63
MMD-LSAE	69.40	87.25	71.53	39.92	62.54	52.94	63.93
SYNC	73.60	86.75	72.02	45.37	71.59	56.44	67.63
SDE	88.60	88.50	73.20	44.26	72.25	68.82	72.60
SSSM-PF	**98.60**	**99.75**	**75.52**	**77.47**	**74.30**	**70.12**	**82.63**

Table 9: Per-target-domain accuracy (%) on the original Circle dataset.

Method	T20	T21	T22	T23	T24	T25	T26	T27	T28	T29	Avg
LSSAE	92.00	94.00	90.00	83.00	67.00	58.00	61.00	56.00	45.00	56.00	70.20
MMD-LSAE	89.00	86.00	79.00	74.00	71.00	57.00	47.00	45.00	55.00	44.00	64.70
SYNC	84.00	92.00	83.00	93.00	83.00	81.00	71.00	77.00	82.00	88.00	83.40
SDE	66.00	72.00	80.00	84.00	71.00	51.00	58.00	55.00	45.00	56.00	63.80
SSSM-PF	**99.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**99.90**

F.2 Per-target-domain accuracy

F.3 Per-intermediate-domain accuracy

Table 10: Per-target-domain accuracy (%) on the original Sine dataset.

Method	T16	T17	T18	T19	T20	T21	T22	T23	Avg
LSSAE	92.00	59.34	47.62	50.00	55.10	54.44	49.47	50.67	57.33
MMD-LSAE	61.00	72.53	86.90	78.57	44.90	44.44	47.37	41.33	59.63
SYNC	67.00	82.42	72.62	85.71	75.51	67.78	66.32	70.67	73.50
SDE	49.00	75.82	88.10	52.38	55.10	54.44	49.47	50.67	59.37
SSSM-PF	**97.00**	**97.80**	**97.60**	**100.00**	**99.00**	**100.00**	**97.90**	**93.30**	**97.82**

Table 11: Per-target-domain accuracy (%) on the original PowerSupply dataset.

Method	T20	T21	T22	T23	T24	T25	T26	T27	T28	T29	Avg
LSSAE	70.21	72.82	72.02	66.40	79.44	73.22	70.51	72.32	71.31	67.10	71.53
MMD-LSAE	69.61	70.11	68.61	65.30	85.16	76.33	69.91	69.71	69.21	66.00	70.99
SYNC	71.11	71.72	70.51	68.51	78.74	75.53	**70.91**	71.31	69.91	68.20	71.64
SDE	69.81	69.61	68.51	64.19	**86.96**	76.33	70.01	69.81	69.31	65.80	71.03
SSSM-PF	**72.80**	**79.00**	**77.50**	**75.50**	86.80	**76.40**	70.70	**78.20**	**77.60**	**74.30**	**76.88**

Table 12: Per-target-domain accuracy (%) on the original RMNIST dataset.

Method	T13	T14	T15	T16	T17	T18	Avg
LSSAE	45.63	34.66	30.70	30.48	35.91	36.94	35.72
MMD-LSAE	46.72	37.43	33.22	31.11	32.71	39.31	36.75
SYNC	61.29	48.21	41.53	40.07	42.62	45.41	46.52
SDE	68.92	53.07	40.88	33.77	32.46	36.02	44.19
SSSM-PF	**98.80**	**97.30**	**94.10**	**98.80**	**97.30**	**96.50**	**97.13**

Table 13: Per-target-domain accuracy (%) on the original Portraits dataset.

Method	T24	T25	T26	T27	T28	T29	T30	T31	T32	T33	Avg
LSSAE	**76.26**	84.48	**89.42**	**92.89**	92.55	91.10	90.83	86.09	89.26	86.54	**87.94**
MMD-LSAE	61.46	64.94	65.31	71.48	73.30	72.95	69.74	64.50	75.15	69.12	68.80
SYNC	72.85	83.08	88.71	92.17	92.88	**93.39**	**91.32**	**87.68**	89.26	85.69	87.70
SDE	73.56	83.08	89.07	90.73	92.81	92.19	90.71	86.09	**89.86**	86.97	87.51
SSSM-PF	75.80	**85.90**	85.50	92.20	**93.40**	89.10	87.10	84.40	89.50	**89.10**	87.20

Table 14: Per-target-domain accuracy (%) on the original CalTran dataset.

Method	T24	T25	T26	T27	T28	T29	T30	T31	T32	T33	Avg
LSSAE	44.38	90.68	59.75	42.50	87.58	48.43	45.62	81.37	45.22	36.59	58.21
MMD-LSAE	63.75	20.50	44.03	58.75	23.60	48.43	53.12	19.88	57.96	67.48	45.75
SYNC	56.25	**92.55**	**62.26**	54.38	90.68	61.01	47.50	88.20	54.14	59.35	66.63
SDE	67.50	76.40	61.64	**63.12**	54.66	61.64	53.12	42.86	**66.24**	71.54	61.87
SSSM-PF	**70.30**	91.40	61.70	61.70	**93.80**	**62.50**	**57.00**	**94.50**	62.50	**73.20**	**72.86**

Table 15: Per-target-domain accuracy (%) on the abrupt Circle dataset.

Method	T20	T21	T22	T23	T24	T25	T26	T27	T28	T29	Avg
LSSAE	30.00	29.00	41.00	40.00	32.00	41.00	46.00	40.00	45.00	41.00	38.50
MMD-LSAE	30.00	28.00	37.00	41.00	38.00	43.00	48.00	49.00	47.00	38.00	39.90
SYNC	26.00	28.00	40.00	38.00	34.00	41.00	43.00	44.00	45.00	37.00	37.60
SDE	40.00	37.00	38.00	43.00	44.00	43.00	54.00	46.00	53.00	42.00	44.00
SSSM-PF	**86.00**	**93.00**	**94.00**	**96.00**	**98.00**	**99.00**	**98.00**	**99.00**	**98.00**	**98.00**	**95.90**

Table 16: Per-target-domain accuracy (%) on the abrupt Sine dataset.

Method	T16	T17	T18	T19	T20	T21	T22	T23	Avg
LSSAE	87.00	78.00	84.00	81.00	86.00	81.00	76.00	79.00	81.50
MMD-LSAE	88.00	80.00	84.00	82.00	85.00	83.00	76.00	79.00	82.12
SYNC	86.00	78.00	86.00	84.00	84.00	81.00	75.00	77.00	81.38
SDE	88.00	80.00	87.00	85.00	85.00	91.00	84.00	88.00	86.00
SSSM-PF	**92.00**	**87.00**	**95.00**	**98.00**	**96.00**	**93.00**	**98.00**	**100.00**	**94.88**

Table 17: Per-target-domain accuracy (%) on the abrupt PowerSupply dataset.

Method	T20	T21	T22	T23	T24	T25	T26	T27	T28	T29	Avg
LSSAE	71.54	71.14	72.55	74.65	72.04	71.04	70.74	69.34	72.12	73.22	71.84
MMD-LSAE	71.44	71.04	71.14	73.15	**72.34**	70.84	70.64	69.54	71.31	72.92	71.44
SYNC	72.75	69.44	75.75	72.24	71.14	73.65	72.04	70.24	74.32	72.32	72.39
SDE	77.15	74.45	61.22	65.33	70.14	**77.25**	**76.65**	73.65	57.37	61.99	69.52
SSSM-PF	**77.70**	**75.90**	**87.40**	**75.90**	70.70	76.90	76.60	**74.50**	**86.30**	**76.60**	**77.85**

Table 18: Per-target-domain accuracy (%) on the abrupt RMNIST dataset.

Method	T13	T14	T15	T16	T17	T18	Avg
LSSAE	39.77	36.70	32.38	33.63	50.54	78.07	45.18
MMD-LSAE	36.56	35.31	36.86	40.72	52.36	75.71	46.25
SYNC	44.57	**41.45**	38.06	42.40	61.32	**88.06**	52.64
SDE	**44.82**	41.10	37.46	36.75	44.68	62.21	44.50
SSSM-PF	37.90	36.70	**48.80**	**68.80**	**78.10**	82.80	**58.85**

Table 19: Per-target-domain accuracy (%) on the abrupt Portraits dataset.

Method	T24	T25	T26	T27	T28	T29	T30	T31	T32	T33	Avg
LSSAE	86.26	89.26	85.96	90.59	91.21	92.49	92.74	**89.12**	84.65	76.56	87.89
MMD-LSAE	69.97	75.53	64.24	70.11	72.89	72.98	71.26	64.37	65.23	61.11	68.77
SYNC	86.26	89.41	**87.02**	91.08	**93.80**	93.32	92.46	88.01	83.26	73.26	87.79
SDE	86.40	89.56	85.17	91.01	92.82	93.07	90.73	88.83	83.43	73.61	87.46
SSSM-PF	**87.90**	**89.80**	85.20	**91.80**	91.00	**93.40**	**95.70**	88.70	**85.50**	**80.90**	**88.99**

Table 20: Per-target-domain accuracy (%) on the abrupt CalTran dataset.

Method	T24	T25	T26	T27	T28	T29	T30	T31	T32	T33	Avg
LSSAE	63.98	40.25	50.00	61.70	56.60	52.50	54.66	43.67	49.35	53.16	52.59
MMD-LSAE	14.91	59.75	71.88	21.99	52.20	69.38	18.63	45.57	60.39	22.78	43.75
SYNC	87.58	58.49	36.25	**85.82**	65.41	48.75	**92.55**	62.66	50.00	**92.41**	67.99
SDE	72.05	72.96	60.00	79.43	72.96	59.38	78.26	65.82	55.84	79.11	69.58
SSSM-PF	**88.30**	**76.60**	**75.80**	85.20	**79.70**	**73.40**	89.80	**68.80**	**61.70**	90.60	**78.99**

Table 21: Per-intermediate-domain accuracy (%) on the original Circle dataset.

Method	I15	I16	I17	I18	I19	Avg
LSSAE	94.00	92.00	92.00	95.00	95.00	93.60
MMD-LSAE	98.00	**100.00**	98.00	98.00	93.00	97.40
SYNC	92.00	92.00	88.00	92.00	94.00	91.60
SDE	90.00	88.00	83.00	83.00	76.00	84.00
SSSM-PF	**99.00**	100.00	**100.00**	**100.00**	**100.00**	**99.80**

Table 22: Per-intermediate-domain accuracy (%) on the original Sine dataset.

Method	I12	I13	I14	I15	Avg
LSSAE	98.84	77.17	79.78	79.78	83.89
MMD-LSAE	**100.00**	91.30	83.15	73.03	86.87
SYNC	91.86	67.39	77.53	66.29	75.77
SDE	69.77	55.43	66.29	55.06	61.64
SSSM-PF	100.00	**96.70**	**100.00**	**100.00**	**99.17**

Table 23: Per-intermediate-domain accuracy (%) on the original PowerSupply dataset.

Method	I15	I16	I17	I18	I19	Avg
LSSAE	71.94	71.64	67.33	80.64	73.32	72.98
MMD-LSAE	69.74	68.24	65.83	85.06	76.03	72.98
SYNC	69.94	69.24	68.24	78.13	75.63	72.24
SDE	69.24	68.14	64.13	**86.56**	76.23	72.86
SSSM-PF	**77.40**	**77.20**	**74.60**	86.00	**76.50**	**78.34**

Table 24: Per-intermediate-domain accuracy (%) on the original RMNIST dataset.

Method	I10	I11	I12	Avg
LSSAE	85.80	74.81	60.34	73.65
MMD-LSAE	84.64	75.52	60.18	73.44
SYNC	94.54	88.68	75.81	86.34
SDE	**96.17**	92.18	83.74	90.70
SSSM-PF	96.10	**98.40**	**98.00**	**97.50**

Table 25: Per-intermediate-domain accuracy (%) on the original Portraits dataset.

Method	I19	I20	I21	I22	I23	Avg
LSSAE	93.30	**73.33**	64.11	68.36	**69.04**	**73.63**
MMD-LSAE	74.62	62.44	59.34	59.13	57.16	62.54
SYNC	93.30	70.83	63.47	65.38	65.13	71.62
SDE	93.43	71.99	63.06	65.65	67.13	72.25
SSSM-PF	**94.10**	71.50	**64.80**	**69.50**	65.60	73.10

Table 26: Per-intermediate-domain accuracy (%) on the original CalTran dataset.

Method	I19	I20	I21	I22	I23	Avg
LSSAE	88.82	55.06	38.96	89.87	54.72	65.49
MMD-LSAE	24.22	45.57	59.74	17.72	52.83	40.02
SYNC	88.20	65.19	51.30	87.34	69.18	72.24
SDE	78.88	63.92	57.79	75.95	64.15	68.14
SSSM-PF	**94.50**	**68.00**	**63.30**	**93.80**	**69.50**	**77.82**

Table 27: Per-intermediate-domain accuracy (%) on the abrupt Circle dataset.

Method	I15	I16	I17	I18	I19	Avg
LSSAE	76.00	81.00	79.00	71.00	76.00	76.60
MMD-LSAE	70.00	74.00	70.00	65.00	68.00	69.40
SYNC	75.00	77.00	76.00	67.00	73.00	73.60
SDE	92.00	92.00	85.00	83.00	91.00	88.60
SSSM-PF	**99.00**	**99.00**	**98.00**	**97.00**	**100.00**	**98.60**

Table 28: Per-intermediate-domain accuracy (%) on the abrupt Sine dataset.

Method	I12	I13	I14	I15	Avg
LSSAE	92.00	94.00	87.00	85.00	89.50
MMD-LSAE	87.00	93.00	83.00	86.00	87.25
SYNC	89.00	93.00	83.00	82.00	86.75
SDE	89.00	89.00	87.00	89.00	88.50
SSSM-PF	**100.00**	**100.00**	**100.00**	**99.00**	**99.75**

Table 29: Per-intermediate-domain accuracy (%) on the abrupt PowerSupply dataset.

Method	I15	I16	I17	I18	I19	Avg
LSSAE	73.82	72.22	71.41	70.61	69.61	71.53
MMD-LSAE	73.62	72.22	71.51	70.91	69.41	71.53
SYNC	72.62	72.02	73.82	72.52	69.11	72.02
SDE	68.71	**73.22**	76.33	75.43	72.32	73.20
SSSM-PF	**77.30**	71.90	**77.60**	**76.90**	**73.90**	**75.52**

Table 30: Per-intermediate-domain accuracy (%) on the abrupt RMNIST dataset.

Method	I10	I11	I12	Avg
LSSAE	42.35	44.57	43.73	43.55
MMD-LSAE	37.46	39.88	42.43	39.92
SYNC	43.02	45.66	47.42	45.37
SDE	40.93	45.06	46.80	44.26
SSSM-PF	**71.10**	**77.70**	**83.60**	**77.47**

Table 31: Per-intermediate-domain accuracy (%) on the abrupt Portraits dataset.

Method	I19	I20	I21	I22	I23	Avg
LSSAE	93.30	73.33	64.11	68.36	**69.04**	73.63
MMD-LSAE	74.62	62.44	59.34	59.13	57.16	62.54
SYNC	93.30	70.71	63.47	65.51	64.96	71.59
SDE	**93.43**	71.99	63.06	65.65	67.13	72.25
SSSM-PF	93.40	**74.60**	**66.40**	**71.90**	65.20	**74.30**

Table 32: Per-intermediate-domain accuracy (%) on the abrupt CalTran dataset.

Method	I19	I20	I21	I22	I23	Avg
LSSAE	51.55	51.59	56.91	47.80	46.88	50.95
MMD-LSAE	20.50	56.69	71.54	57.23	58.75	52.94
SYNC	92.55	55.41	49.59	45.91	38.75	56.44
SDE	84.47	**71.34**	60.98	**66.67**	**60.62**	68.82
SSSM-PF	**93.00**	71.10	**72.40**	53.90	60.20	**70.12**