
MissPath-FM: Flow Matching with Structured Priors for Partially Observed Time Series

Genpei Zhang

Department of Computer Sciences
University of Wisconsin–Madison
genpei.zhang@wisc.edu

Abstract

Flow matching has emerged as a powerful alternative to diffusion models for generative modeling, but standard conditional flow matching is fundamentally misaligned with partially observed time series: starting from an uninformed Gaussian source forces the model to transport even already-observed entries from noise to the target, wasting capacity on trivially known regions. We propose **MissPath-FM**, a partial-observation-aware flow matching framework that redesigns the source prior and probability path for incomplete time series. MissPath-FM employs *structured source priors* (interpolation priors and Gaussian process posterior priors) to initialize the flow close to the observed signal, paired with a *missingness-aware smooth path* that allocates transport asymmetrically between observed and missing regions. We evaluate systematically across four datasets, four missingness mechanisms (MCAR, block, MNAR, irregular), two missing ratios, and three sequence lengths ($T \in \{24, 48, 96\}$), totaling 48 settings against eight baselines. MissPath-FM wins 29 of 32 settings at $T=48$ and 45 of 48 overall (93.8%). Our ablations reveal a regime-dependent design hierarchy in which no single component suffices: under random missingness the source prior yields a large initial gain, but under structured block missingness it can *fail entirely* and the learned flow becomes indispensable; the missingness-aware path contributes a further 12%–26% across all regimes. Mechanistically, structured priors shorten observed-region transport by up to $15\times$, concentrating the velocity field on genuinely uncertain entries. Project page: <https://zgp110.github.io/misspath-fm/>.

1 Introduction

Real-world time series are rarely fully observed. In healthcare, laboratory tests are sparse and asynchronous; in energy systems, sensors fail or report irregularly; missing values often occur in structured blocks rather than isolated dropouts. Models must therefore operate under *partial observability*, making predictions from incomplete and irregular evidence [3, 19].

Prior work addresses missingness through deterministic imputation, recurrent architectures, and probabilistic generative models. Diffusion-based methods [23, 1, 4] and transport-based approaches [5, 18] have further advanced generative imputation. While effective, these methods typically treat missingness as an input-side conditioning mechanism, overlooking how the *geometry of generative transport* interacts with partial observability.

Flow matching has emerged as a powerful alternative to diffusion models [13, 16], learning a continuous velocity field that transports a source distribution to the target via ODE integration [25]. Recent works apply flow matching to temporal data: TSFlow [12] integrates GP priors for forecasting, and TimeBridge [17] explores data-dependent priors for synthesis. Despite this progress, *none of these methods explicitly addresses the interaction between the flow-matching transport and partial*

observability. When $z_0 \sim \mathcal{N}(0, I)$, the model wastes capacity transporting already-observed entries from noise, complicating learning in genuinely uncertain regions.

We argue that the central bottleneck in applying flow matching to incomplete time series lies not in the velocity network architecture, but in the *geometry of transport under partial observability*. Two questions naturally arise: (1) what source distribution should the transport originate from when part of the sequence is already known? and (2) how should the probability path allocate transport effort between observed and missing regions? Addressing these questions yields a simple design principle: the source prior should respect the observed signal, and the transport path should concentrate generative effort on the uncertain entries.

Motivated by this perspective, we propose **MissPath-FM** (**Missingness-aware Path** design for **Flow Matching**), a partial-observation-aware flow matching framework for incomplete time series. MissPath-FM combines *structured source priors* (from interpolation priors to Gaussian-process posterior priors) with a *missingness-aware smooth path* and context conditioning based on missingness masks and time gaps. Instead of regenerating the entire sequence from pure noise, MissPath-FM learns a velocity field that transports uncertainty from observed evidence into the missing regions.

Empirically, our evaluation across 48 settings reveals that no single design choice is universally sufficient. Under random missingness (MCAR), the *source prior* yields a large initial gain. Under structured block missingness, however, the prior alone can *degrade* performance when contiguous gaps leave no flanking anchors; here the learned flow becomes indispensable. The missingness-aware path provides a further 12%–26% improvement across both regimes. Only the full pipeline—prior, flow, and path acting together—achieves robust performance across all missingness types. Mechanistically, structured priors shorten transport on observed regions by up to $15\times$, reallocating capacity toward genuinely uncertain entries.

Our primary contributions are:

- We introduce **MissPath-FM**, a flow matching framework for partial observability that combines structured source priors, mask/gap conditioning, and missingness-aware probability paths.
- We establish a **regime-dependent design hierarchy** in which no single component suffices: the source prior yields a large initial gain under random missingness but can fail under block patterns; the learned flow is indispensable precisely where the prior struggles; and the missingness-aware path contributes a consistent 12%–26% across all regimes. Only their combination achieves robust performance.
- We provide a **transport-geometric explanation**, showing how structured priors shorten observed-region trajectories and reallocate the velocity field’s capacity to genuinely uncertain entries.
- We demonstrate **state-of-the-art performance** across 32 settings at $T=48$ (4 datasets, 4 missingness mechanisms, 2 ratios), winning 29 against eight baselines (45/48 including $T \in \{24, 96\}$).

2 Related Work

Time-series modeling under missingness and irregular sampling. A large body of literature studies multivariate time series with missing values and irregular observations. Early deep learning approaches such as GRU-D and BRITS incorporate missingness masks and time intervals into recurrent architectures, showing that missingness patterns themselves carry informative signals [3, 2]. Beyond recurrence, mTAN introduces learned continuous-time embeddings with multi-head temporal attention for irregularly sampled data [20], Neural CDE treats the time series as a control path driving latent dynamics [11], and SAITS replaces recurrence with masked self-attention for efficient deterministic imputation [7]. While effective, these methods target point estimation or classification and do not model the full conditional distribution over missing entries. MissPath-FM addresses this limitation by formulating partial observation as a *conditional transport* problem, where the source prior, context encoding, and probability path are jointly designed around the missingness structure.

Probabilistic and generative models for incomplete time series. Another line of work tackles missingness through probabilistic and generative modeling. GP-VAE enforces a Gaussian-process prior in the latent space for smooth temporal structure [9], while Latent ODEs provide continuous-time latent dynamics that naturally accommodate non-uniform observations [19]. Within the diffusion framework, CSDI combines conditional score-based diffusion with a transformer backbone [23];

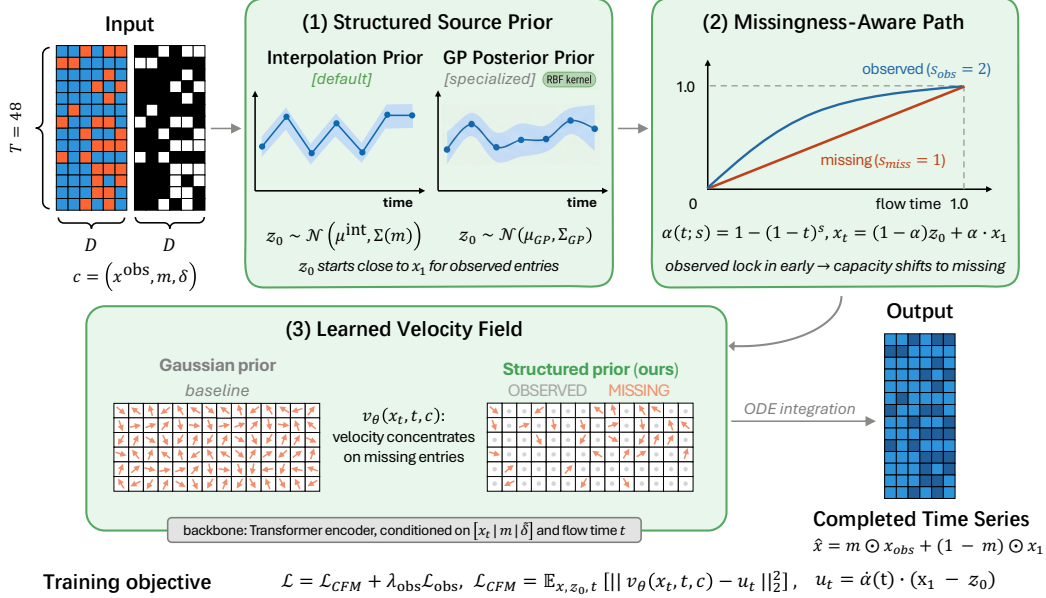


Figure 1: Overview of MissPath-FM. **(1)** A structured source prior initializes z_0 close to the observed signal. **(2)** A missingness-aware path ($s_{\text{obs}} > s_{\text{miss}}$) locks in observed entries early. **(3)** A context-conditioned Transformer predicts the velocity; at inference, z_0 is integrated with 20 Euler steps and observed entries are restored.

SSSD replaces the transformer with a structured state-space model, excelling in blackout-missing scenarios [1]; and PriSTI incorporates spatiotemporal dependencies as conditional priors [15]. Beyond standard diffusion, CSBI applies the Schrödinger bridge to time-series imputation via entropy-regularized optimal transport [4], and NewImp reinterprets diffusion-based imputation from a Wasserstein gradient flow perspective [5]. Despite strong empirical results, all these methods rely on predefined noise-corruption schedules or iterative bridge procedures, treating missingness primarily as an input-side conditioning signal. In contrast, MissPath-FM learns a continuous transport map whose source distribution is dynamically adapted to observed evidence, bypassing iterative denoising entirely.

Flow matching and transport-based generative modeling. Flow matching provides a simulation-free framework for training continuous normalizing flows by regressing velocity fields along predefined probability paths [13]. Rectified-flow formulations [16] and OT-CFM with minibatch couplings [25] further straighten trajectories for efficient few-step inference. These advances have catalyzed growing interest in transport-based models for temporal data: TFM extends flow matching to irregular clinical time series [29]; FM-TS adapts rectified flows for generation [10]; TSFlow integrates GP priors for probabilistic forecasting [12]; CLWF uses Lagrangian OT flows for imputation [18]; PSW-I applies optimal transport with posterior sampling for imputation [26]; and TimeBridge explores data-dependent priors within a diffusion bridge [17]. Three concurrent works are especially relevant. CFMI [22] applies conditional flow matching for general-purpose imputation but retains a Gaussian source without structured priors. CGFM [27] shares our motivation of informed priors by using auxiliary predictions as the source, but targets fully observed forecasting without mask conditioning. Impute-MACFM [14] introduces a mask-aware CFM objective for tabular data that penalizes velocity on observed entries, addressing partial observability through loss design rather than prior/path shaping. MissPath-FM is complementary: rather than relying on loss design, we reshape the source distribution so the velocity field *naturally* concentrates on missing entries, and pair it with a missingness-aware path that allocates transport asymmetrically.

3 Method

We present MissPath-FM, a generative framework for incomplete time series that reshapes the flow matching transport around partial observability (Figure 1). The method has three core components, all used in every reported result: (i) an observation-aware source prior (Section 3.2), (ii) a missingness-aware probability path (Section 3.3), and (iii) a context-conditioned velocity network (Section 3.4). Two further elements are not core: the GP posterior prior is an optional alternative to the default interpolation prior in (i) and appears only in ablations and analyses, and RandomRate is a training protocol rather than a part of the architecture (Section 4). Section 3.1 first identifies the geometric bottleneck that motivates all three; Table 1 collects the notation, and Figure 2 gives the complete training step in code.

Table 1: Notation, grouped by role. The two source priors are alternative choices for the same object $p_0(\cdot | c)$, not sequential steps.

<i>Data and context</i>		<i>Source prior (choose one)</i>	
$x \in \mathbb{R}^{T \times D}$	complete series	$\mu^{\text{int}}, \Sigma(m)$	Option A: interpolation (default)
$m \in \{0, 1\}^{T \times D}$	mask, 1 = observed	μ_*, Σ_*	Option B: GP posterior (optional)
δ	time since last observation	$\sigma_{\text{obs}}, \sigma_{\text{miss}}$	prior noise scales
$c = (x^{\text{obs}}, m, \delta)$			
<i>Flow matching</i>		<i>Path schedule</i>	
$z_0 \sim p_0(\cdot c)$	source endpoint ($t=0$)	$s_{\text{obs}}, s_{\text{miss}}$	transport speeds
$x_1 \sim p_{\text{data}}$	target endpoint ($t=1$)	$s_{\tau, d}$	entry-wise speed (Eq. 6)
x_t	interpolant at flow time t	$\alpha_{\tau, d}(t)$	entry-wise schedule (Eq. 7)
v_θ, u_t	learned / target velocity	$\mathcal{C}_{\text{obs}}, \mathcal{C}_{\text{miss}}$	transport cost (Prop. 1)

3.1 Problem Formulation and Geometric Bottleneck

Let $x \in \mathbb{R}^{T \times D}$ denote a multivariate time series of length T with D variables. A binary mask $m \in \{0, 1\}^{T \times D}$ indicates observed entries ($m_{\tau, d} = 1$), the observed input is $x^{\text{obs}} = m \odot x$, and a time-gap tensor $\delta \in \mathbb{R}_{\geq 0}^{T \times D}$ records the elapsed time since the last observation of each variable. Our goal is to model $p(x | c)$ given context $c = (x^{\text{obs}}, m, \delta)$.

Flow matching [13] learns a velocity field $v_\theta : \mathbb{R}^{T \times D} \times [0, 1] \rightarrow \mathbb{R}^{T \times D}$ transporting a source distribution p_0 to $p_1 \approx p_{\text{data}}$. Conditional flow matching (CFM) [13, 25] constructs conditional paths $p_t(x | z)$ anchored by pairs $z = (z_0, x_1)$ with $z_0 \sim p_0, x_1 \sim p_{\text{data}}$, yielding the training objective:

$$\mathcal{L}_{\text{CFM}} = \mathbb{E}_{t, x_1, z_0} \left[\|v_\theta(x_t, t) - u_t(x_t | z_0, x_1)\|_2^2 \right]. \quad (1)$$

Following Tong et al. [25], we write z_0 rather than x_0 for the source endpoint so that it is not confused with the data variable x : the transport runs from z_0 at $t=0$ to x_1 at $t=1$, and x_t is the point in between. The standard linear path $x_t = (1-t)z_0 + tx_1$ yields a constant conditional velocity $u_t = x_1 - z_0$. Under this path, the expected transport cost decomposes by observation status:

Proposition 1 (Transport cost decomposition). *Under the linear conditional path, the expected squared transport distance splits into an observed and a missing part:*

$$\mathbb{E}[\|x_1 - z_0\|^2] = \mathcal{C}_{\text{obs}} + \mathcal{C}_{\text{miss}}, \quad \mathcal{C}_{\text{obs}} = \sum_{m_{\tau, d}=1} \mathbb{E}[(x_{1, \tau, d} - z_{0, \tau, d})^2], \quad (2)$$

and $\mathcal{C}_{\text{miss}}$ is the same sum over missing entries ($m_{\tau, d} = 0$). Let $n_{\text{obs}} = \sum_{\tau, d} m_{\tau, d}$. Two choices of the source p_0 give:

- (a) Gaussian source $z_0 \sim \mathcal{N}(0, I)$, independent of x_1 : each observed entry contributes $\mu^2 + \sigma^2 + 1$, where μ, σ are its marginal mean and standard deviation. For z -scored data this equals 2, so $\mathcal{C}_{\text{obs}} = 2 n_{\text{obs}}$, growing linearly with the number of observed entries.
- (b) Interpolation source (Eq. 3): observed entries satisfy $z_0 = x_1 + \epsilon$ with $\epsilon \sim \mathcal{N}(0, \sigma_{\text{obs}}^2)$, so $\mathcal{C}_{\text{obs}} = \sigma_{\text{obs}}^2 n_{\text{obs}} \rightarrow 0$ as $\sigma_{\text{obs}} \rightarrow 0$.

The proof follows from linearity of expectation and standard Gaussian moments (Appendix D). Proposition 1 quantifies a fundamental inefficiency: for normalized data at 50% missingness, approximately half the network’s capacity is wasted on trivially known entries. MissPath-FM eliminates this waste through structured source priors (Section 3.2) and further concentrates capacity via a missingness-aware path (Section 3.3).

3.2 Observation-Aware Source Priors

We replace the uninformed Gaussian $z_0 \sim \mathcal{N}(0, I)$ with source distributions $z_0 \sim p_0(\cdot | c)$ conditioned on the observation context, guided by two principles: (i) *consistency*—observed entries in z_0 should be close to their known values, driving $\mathcal{C}_{\text{obs}} \rightarrow 0$; and (ii) *temporal coherence*—missing entries should be initialized with temporally plausible values rather than independent noise, reducing $\mathcal{C}_{\text{miss}}$ and easing the learning burden on the velocity network. We consider two instantiations of $p_0(\cdot | c)$, used *alternatively* rather than in sequence.

Option A: interpolation prior (default). For each variable d , we apply piecewise linear interpolation along the time axis: observed entries retain their values; missing entries are interpolated between flanking observations (with forward/backward fill at boundaries). Denoting this $\mu^{\text{int}} = \mathcal{I}(x^{\text{obs}}, m)$, the source distribution is:

$$z_0 \sim \mathcal{N}(\mu^{\text{int}}, \Sigma(m)), \quad \Sigma_{\tau,d}(m) = \begin{cases} \sigma_{\text{obs}}^2, & \text{if } m_{\tau,d} = 1, \\ \sigma_{\text{miss}}^2, & \text{if } m_{\tau,d} = 0. \end{cases} \quad (3)$$

Setting $\sigma_{\text{obs}} \ll \sigma_{\text{miss}}$ satisfies the consistency principle: $\mathcal{C}_{\text{obs}} = \sigma_{\text{obs}}^2 n_{\text{obs}}$ (Proposition 1). This prior has no fitted parameters and is robust, but provides only *pointwise* coherence—under structured missingness (e.g., contiguous block gaps), interpolation degenerates into flat extrapolation, violating temporal coherence.

Option B: Gaussian process posterior prior (optional). To provide *joint* temporal coherence, we introduce a GP posterior prior. For each variable d , given observed timestamps $\mathbf{t}_o$ and values $\mathbf{y}_o$, we compute the posterior predictive over the full grid $\mathbf{t}_* = (1, \dots, T)$:

$$\mu_* = K_{*o} (K_{oo} + \sigma_n^2 I)^{-1} \mathbf{y}_o, \quad (4)$$

$$\Sigma_* = K_{**} - K_{*o} (K_{oo} + \sigma_n^2 I)^{-1} K_{o*}, \quad (5)$$

where $K_{*o} = k_\psi(\mathbf{t}_*, \mathbf{t}_o)$ (RBF kernel by default). Sampling $z_{0,:d} \sim \mathcal{N}(\mu_*, \Sigma_*)$ injects temporal correlations unavailable to local interpolation: the GP posterior mean smoothly regresses toward the global trend within gaps, and the full covariance encodes structured uncertainty that grows with distance from observed anchors. The two priors are complementary—interpolation is robust but degrades in long gaps; GP exploits temporal structure at the cost of kernel sensitivity (Appendix H). *Choosing between them.* All main results use Option A. Option B is 23–39 $\times$ slower to construct (Appendix K.6), so we recommend it only for short, smooth series where its temporal structure pays for that cost. Both options drive $\mathcal{C}_{\text{obs}}$ toward zero; Option B additionally reduces $\mathcal{C}_{\text{miss}}$ (Section 5.4).

3.3 Missingness-Aware Probability Path

Even with a structured prior that reduces $\mathcal{C}_{\text{obs}}$, the standard linear path applies a uniform temporal schedule to all entries. This is suboptimal: observed entries start near their targets and need minimal refinement, while missing entries require the full transport capacity. We introduce a missingness-aware (MA) path with entry-wise speed:

$$s_{\tau,d} = s_{\text{miss}} + (s_{\text{obs}} - s_{\text{miss}}) m_{\tau,d}, \quad (6)$$

where $s_{\text{obs}} \geq s_{\text{miss}} \geq 1$. This defines $\alpha_{\tau,d}(t) = 1 - (1 - t)^{s_{\tau,d}}$ and the conditional path:

$$x_{t,\tau,d} = (1 - \alpha_{\tau,d}(t)) z_{0,\tau,d} + \alpha_{\tau,d}(t) x_{1,\tau,d}. \quad (7)$$

The conditional target velocity is:

$$u_{t,\tau,d} = \dot{\alpha}_{\tau,d}(t) (x_{1,\tau,d} - z_{0,\tau,d}), \quad \dot{\alpha}_{\tau,d}(t) = s_{\tau,d} (1 - t)^{s_{\tau,d}-1}. \quad (8)$$

Since $\alpha(0) = 0$, $\alpha(1) = 1$, and α is monotonically increasing and C^1 for all $s \geq 1$, the path is valid within the CFM framework [13].

Under the standard path, even when $z_0 \approx x_1$ on observed entries, the network must keep $v_\theta \approx 0$ there *across the entire interval* $t \in [0, 1]$. With $s_{\text{obs}} = 2$, observed entries complete 51% of their transport by $t = 0.3$ while missing entries ($s_{\text{miss}} = 1$) complete 30%; for $t > 0.5$ the observed residual is negligible, and the velocity field effectively reduces from $T \times D$ to $|\mathcal{M}|$ dimensions. Unlike the prior, the MA path leaves $\mathcal{C}_{\text{obs}}$ and $\mathcal{C}_{\text{miss}}$ essentially unchanged; what it shortens is the part of the trajectory during which the network must track observed entries, so its benefit shows up in accuracy rather than in trajectory length (Appendix I.1).

3.4 Velocity Network and Optimization

We parameterize $v_\theta(x_t, t, c)$ with a Transformer encoder along the temporal dimension. At each step τ , the input token is $[x_{t,\tau,:} \parallel m_{\tau,:} \parallel \tilde{\delta}_{\tau,:}] \in \mathbb{R}^{3D}$ (where $\tilde{\delta} = \delta/(\delta_{\text{max}} + 1)$), projected to d_{model} and augmented with learnable positional and sinusoidal flow-time embeddings. Full architectural details are in Appendix B.

Given $z_0 \sim p_0(\cdot | c)$ and $t \sim \mathcal{U}[0, 1]$, we construct x_t via Eq. (7) and minimize:

$$\mathcal{L} = \mathcal{L}_{\text{CFM}} + \lambda_{\text{obs}} \mathcal{L}_{\text{obs}}, \quad \mathcal{L}_{\text{CFM}} = \mathbb{E}_{x, z_0, t} \left[\|v_\theta(x_t, t, c) - u_t\|_2^2 \right], \quad (9)$$

where u_t is the conditional target velocity from Eq. (8), $\mathcal{L}_{\text{CFM}}$ is computed over all entries, and $\mathcal{L}_{\text{obs}}$ is the same squared error averaged over observed entries only, with $\lambda_{\text{obs}} = 0.1$ in all reported results. The auxiliary term is light: the structured prior and MA path already make the observed-entry targets small (Proposition 1), so it acts as a consistency safeguard rather than as the mechanism that concentrates the velocity field, unlike explicit velocity penalization [14].

At inference, we sample $z_0 \sim p_0(\cdot | c_{\text{test}})$ and integrate via fixed-step Euler: $x_{t+\Delta t} = x_t + \Delta t v_\theta(x_t, t, c_{\text{test}})$ with $\Delta t = 1/N_{\text{step}}$. Observed entries are deterministically restored after integration. For probabilistic imputation, K independent samples yield an ensemble for pointwise statistics. MissPath-FM extends to forecasting by constructing the prior exclusively from the history window (Appendix D). Figure 2 gives the complete training step.

```
# (i) Interpolation prior, Eq. 3
mu    = interp_1d(x_obs, mask)           # piecewise linear in time
sigma = where(mask, 0.1, 1.0)           # sigma_obs << sigma_miss
z0     = mu + sigma * randn_like(mu)

# (ii) Missingness-aware path, Eqs. 6-8
s      = where(mask, 2.0, 1.0)           # s_obs = 2, s_miss = 1
alpha  = 1 - (1 - t) ** s
x_t    = (1 - alpha) * z0 + alpha * x1
u_t    = s * (1 - t) ** (s - 1) * (x1 - z0)

# (iii) Loss, Eq. 9: CFM on all entries + light observed-entry term
err    = (v_theta(x_t, t, ctx) - u_t) ** 2
loss   = err.mean() + 0.1 * (mask * err).sum() / mask.sum()
```

Figure 2: The complete MissPath-FM training step. Variables follow Table 1; ctx is the context c .

4 Experimental Setup

Datasets and evaluation protocol. We evaluate on four datasets: **Synthetic** ($D=8$; multivariate sinusoidal), **ETTm1** ($D=7$; electricity transformer temperatures) [30], **Electricity** ($D=20$; UCI Electricity, top-variance clients) [8], and **PhysioNet 2012** ($D=36$; ICU variables, $\sim 19\%$ natural observation rate) [21]. The default sequence length is $T=48$; we additionally test $T \in \{24, 96\}$ on ETTm1. We assess three tasks: imputation, forecasting ($h=24$), and unconditional generation. Missingness is imposed under four mechanisms at two ratios (30%, 50%): MCAR, Block (contiguous per-feature gaps), MNAR (value-dependent dropout), and Irregular (asynchronous per-feature patterns), yielding 32 settings at $T=48$. PhysioNet retains its natural sparsity with 10%–50% additional artificial holdout for evaluation. All features are z-score normalized per variable; data is split 80/20 with fixed seed 42. Full specifications are in Appendix A.

Table 2: Imputation results (MAE $\downarrow$, $T=48$). Best in **bold**, second best underlined. MissPath-FM wins **29/32** settings. All methods share identical splits and masks. Baselines use official code where available; FlowTS, TSFlow, and CLWF include re-implemented components (Appendix C).

Dataset	Mechanism	Rate	TRANSPORT-BASED				DETERMINISTIC / GENERATIVE				Ours
			FlowTS	CLWF	TSFlow	PSW-I	SAITS	BRITS	CSDI	GP-VAE	
ETTm1	MCAR	30%	0.393	0.190	0.121	0.144	<u>0.096</u>	0.103	0.159	0.454	0.075
	MCAR	50%	0.431	0.224	0.132	0.162	<u>0.109</u>	0.122	0.203	0.583	0.085
	Block	30%	0.396	0.197	0.245	0.265	<u>0.152</u>	0.160	0.248	0.511	0.106
	Block	50%	0.450	0.249	0.267	0.320	<u>0.199</u>	0.238	0.358	0.572	0.145
	MNAR	30%	0.439	0.366	0.186	0.243	<u>0.143</u>	0.186	0.219	0.746	0.117
	MNAR	50%	0.482	0.382	0.244	0.285	<u>0.216</u>	0.269	0.273	0.793	0.158
	Irregular	30%	0.444	0.382	0.144	0.188	<u>0.134</u>	0.141	0.246	0.735	0.101
	Irregular	50%	0.461	0.327	0.141	0.188	<u>0.132</u>	0.145	0.532	0.731	0.101
Synthetic	MCAR	30%	0.702	0.579	<u>0.112</u>	0.216	0.139	0.161	0.200	0.785	0.098
	MCAR	50%	0.794	0.729	<u>0.138</u>	0.289	0.211	0.200	1.052	0.799	0.110
	Block	30%	0.791	0.761	0.555	0.821	0.593	<u>0.547</u>	0.688	0.778	0.374
	Block	50%	0.823	0.784	0.645	0.893	<u>0.641</u>	0.794	0.726	0.783	0.470
	MNAR	30%	0.866	0.841	<u>0.119</u>	0.308	0.143	0.193	1.106	0.994	0.115
	MNAR	50%	0.918	0.873	<u>0.141</u>	0.401	0.256	0.246	1.038	0.924	0.126
	Irregular	30%	0.841	0.804	<u>0.248</u>	0.427	0.377	0.340	0.993	0.801	0.199
	Irregular	50%	0.843	0.811	<u>0.245</u>	0.425	0.379	0.335	1.031	0.800	0.196
Electricity	MCAR	30%	0.309	0.232	0.178	0.155	0.159	<u>0.145</u>	0.274	0.336	0.091
	MCAR	50%	0.324	0.231	0.193	0.183	0.170	<u>0.169</u>	0.330	0.359	0.100
	Block	30%	0.344	0.263	0.366	0.498	<u>0.184</u>	0.178	0.309	0.361	0.216
	Block	50%	0.361	0.258	0.355	0.531	0.195	<u>0.200</u>	0.375	0.417	0.252
	MNAR	30%	0.324	0.247	0.208	0.199	0.181	<u>0.170</u>	0.333	0.487	0.112
	MNAR	50%	0.422	0.310	0.255	0.289	<u>0.213</u>	0.216	0.369	0.598	0.130
	Irregular	30%	0.416	0.365	0.275	0.236	<u>0.189</u>	0.197	0.723	0.425	0.118
	Irregular	50%	0.409	0.306	0.239	0.237	<u>0.186</u>	0.201	1.070	0.409	0.120
PhysioNet	Art.	10%	0.524	0.334	0.399	0.379	<u>0.234</u>	0.257	0.363	0.594	0.227
	Art.	20%	0.558	0.346	0.407	0.391	<u>0.243</u>	0.279	0.394	0.616	0.238
	Art.	30%	0.523	0.364	0.413	0.400	<u>0.264</u>	0.299	0.443	0.638	0.250
	Art.	50%	0.545	0.412	0.440	0.427	<u>0.313</u>	0.347	0.571	0.687	0.308
	Block	30%	0.550	0.358	0.407	0.400	<u>0.264</u>	0.300	0.451	0.643	0.261
	Block	50%	0.536	0.417	0.435	0.427	<u>0.312</u>	0.345	0.558	0.687	0.299
	MNAR	30%	0.524	0.363	0.419	0.399	<u>0.266</u>	0.300	0.420	0.644	0.261
	MNAR	50%	0.540	0.410	0.438	0.425	0.314	0.346	0.514	0.687	<u>0.327</u>

Baselines and metrics. We compare against eight methods under a unified protocol (identical splits, masks, metrics): SAITS [7] and BRITS [2] (deterministic); CSDI [23] and GP-VAE [9] (diffusion/VAE); FlowTS [10], CLWF [18], TSFlow [12], and PSW-I [26] (transport-based). Baselines use official codebases where available; when official code does not cover a task, we provide faithful re-implementations (Appendix C). We report **MAE** on held-out missing positions for imputation and forecasting; for generation, **discriminative score**, **predictive score**, and **spectral distance** following [28] (definitions in Appendix C.2). Our model uses a 4-layer Transformer encoder ($d_{\text{model}}=256$, 4 heads, $\sim 3.9\text{M}$ parameters) with the interpolation prior and RandomRate training ($r \sim \mathcal{U}(0.1, 0.7)$ per sample), trained for 500 epochs with AdamW and Euler-20 inference. Full details are in Appendix B.

Hyperparameter protocol. A single global configuration is used for every dataset, missingness mechanism, rate, and sequence length: the architecture above, interpolation prior with $\sigma_{\text{obs}}=0.1$ and $\sigma_{\text{miss}}=1.0$, MA path with $s_{\text{obs}}=2$ and $s_{\text{miss}}=1$, $\lambda_{\text{obs}}=0.1$, RandomRate training, Euler-20, and $K=20$ samples. No hyperparameter is tuned per dataset, per mechanism, or per rate. Appendix H varies one axis at a time around this fixed configuration.

5 Results and Analysis

5.1 Main Imputation Results

Table 2 reports MAE across all 32 evaluation settings (4 datasets $\times$ 4 missingness mechanisms $\times$ 2 ratios) at $T=48$. MissPath-FM wins **29 of 32 settings** with an average rank of 1.16 across all

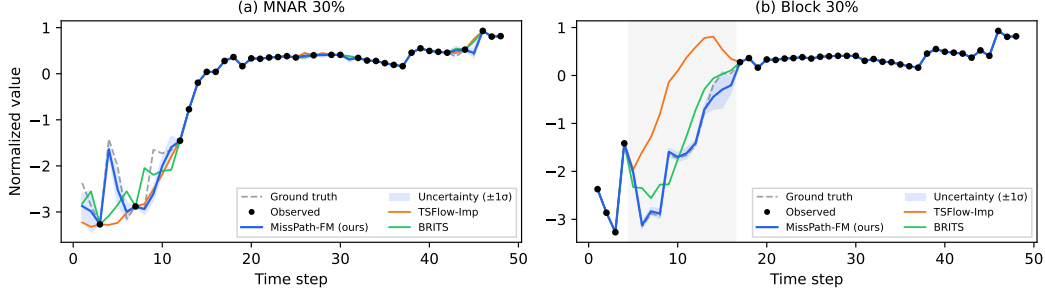


Figure 3: Qualitative imputation on ETTm1. **(a)** MNAR 30%: MissPath-FM recovers masked extreme values where baselines deviate. **(b)** Block 30%: within the contiguous gap (shaded), TSFlow diverges and BRITS flattens, while MissPath-FM tracks the ground truth with calibrated uncertainty.

Table 3: Incremental ablation (MAE $\downarrow$, 4 d.p.). Configurations A–F add one component at a time. Best per row in **bold**. Δ_{flow} : gain of (F) over best Prior-Only.

Dataset	Setting	PRIOR-ONLY			INCREMENTAL FLOW			Δ_{flow}
		(A) Gaussian +Flow	(B) Interpolation Prior	(C) GP Prior	(D) +Flow Matching	(E) +Context Encoding	(F) +Missingness- Aware Path	
ETTM1	MCAR 30%	0.7387	0.1091	0.1214	0.0918	0.0856	0.0751	−31%
	Block 30%	0.7248	0.2162	0.4150	0.1335	0.1253	0.1057	−51%
Synthetic	MCAR 30%	0.8165	0.1614	0.1089	0.1208	0.1182	0.0981	−10%
	Block 30%	0.8056	0.8574	0.6326	0.5578	0.4938	0.3737	−41%
Electricity	MCAR 30%	0.4028	0.1897	0.2230	0.1238	0.1137	0.0910	−52%
	Block 30%	0.4142	0.8417	0.7395	0.2789	0.2503	0.2158	−71%
PhysioNet	Art. 10%	0.6355	0.3632	0.4119	0.3305	0.3057	0.2271	−37%
	Block 30%	0.6307	0.3737	0.4404	0.3420	0.3248	0.2610	−30%

methods. The three losses occur in two specific regimes: Electricity Block 30% and 50% (BRITS and SAITS, whose recurrent and attention-based architectures handle moderate contiguous gaps on high-dimensional data effectively) and PhysioNet MNAR 50% (SAITS, by 4%).

MissPath-FM’s advantage is consistent across missingness mechanisms: it wins all 10 MCAR-type settings (including PhysioNet’s artificial holdout), 6/8 Block, 7/8 MNAR, and 6/6 Irregular, confirming that observation-aware transport design generalizes beyond any single missingness type. The margins are largest under structured missingness (ETTM1 Block 50%: −27%; Electricity MNAR 30%: −34%), where the geometric advantage of structured priors is most pronounced. On PhysioNet, MissPath-FM wins 7 of 8 settings with narrow but consistent margins over SAITS, the strongest baseline on clinical data. Figure 3 illustrates two representative cases: under MNAR, MissPath-FM recovers masked extreme values where baselines deviate; under Block, it tracks the ground truth within contiguous gaps with calibrated uncertainty. We additionally evaluate at $T \in \{24, 96\}$ on ETTm1 (Appendix E): MissPath-FM wins all 16 settings, with margins growing at longer sequences (19% at $T=24$, 26% at $T=96$). Beyond point estimates, **probabilistic calibration (Appendix F)** and **multi-seed evaluation (Appendix J)** confirm both well-calibrated uncertainty (29/32 wins, $\text{ACE} \leq 0.104$) and low variance ($\text{CV} < 2.4\%$). Appendix K further reports a 10–90% missing-rate sweep, an observation-noise study, an evaluation at $T=192$, and baselines retrained with the same RandomRate augmentation; the last shows that the augmentation does not help the baselines, so our margins do not stem from it.

5.2 Ablation: What Drives the Gains?

Table 3 reports an incremental ablation across all four datasets (MCAR 30% + Block 30%), **adding one component at a time**. Full per-component details are in Appendix G. A notable property is that *every* incremental component improves MAE in *every* setting: (B)→(D), (D)→(E), and (E)→(F) are all strictly negative across all 8 rows, confirming that each design choice contributes independently.

MCAR: prior provides the dominant initial gain. Under random missingness, replacing the Gaussian with interpolation reduces MAE by 43%–85%, the single largest step. The learned flow

Table 4: Forecasting results at $h=24$ (MAE $\downarrow$). Best in **bold**, second underlined.

Dataset	History	SAITS	BRITS	CSDI	GP-VAE	FlowTS	CLWF	TSFlow	Ours
ETTM1	Clean	0.489	0.692	0.763	0.694	0.702	0.483	<u>0.355</u>	0.266
	30% MCAR	0.528	0.692	0.593	0.704	0.679	0.480	<u>0.365</u>	0.264
Synthetic	Clean	0.791	0.781	1.084	0.787	0.850	0.825	<u>0.566</u>	0.511
	30% MCAR	0.806	0.778	1.071	0.794	0.850	0.837	<u>0.573</u>	0.572
Electricity	Clean	1.040	0.873	0.861	0.937	0.935	0.947	<u>0.226</u>	0.149
	30% MCAR	0.871	0.821	1.056	0.907	0.918	0.953	<u>0.237</u>	0.153

Table 5: Generation quality (all metrics $\downarrow$). Best in **bold**, second underlined.

Method	Synthetic			ETTM1			Electricity		
	Disc.	Pred.	Spec.	Disc.	Pred.	Spec.	Disc.	Pred.	Spec.
CSDI	0.500	0.925	1.540	0.474	0.745	1.313	0.500	0.631	1.294
GP-VAE	0.500	0.809	1.657	0.500	0.887	1.062	0.477	0.581	1.341
FlowTS	<u>0.275</u>	0.266	<u>0.126</u>	<u>0.184</u>	<u>0.214</u>	<u>0.045</u>	0.250	0.279	<u>0.112</u>
TSFlow	0.500	0.929	1.538	0.500	0.906	1.699	0.500	0.734	1.655
Gauss.+Std	0.300	0.240	0.155	0.198	0.286	0.256	<u>0.182</u>	<u>0.201</u>	0.113
MissPath-FM	0.169	<u>0.261</u>	0.092	0.064	0.186	0.008	0.155	0.141	0.104

adds a further 9%–35% over the prior alone ((B) $\rightarrow$ (D)). The MA path contributes a consistent 12%–26% ((E) $\rightarrow$ (F)), indicating that missingness-aware scheduling is beneficial even under random patterns, where the obs/miss asymmetry is sufficient for differential transport to improve accuracy.

Block: flow provides the dominant recovery. The pattern reverses under block missingness: interpolation-only *fails* on Electricity (MAE = 0.842, worse than Gaussian+Flow at 0.414) and is near-random on Synthetic (MAE = 0.857), because contiguous gaps leave no flanking observations for interpolation. Here the learned flow provides the dominant recovery (−67% on Electricity Block, −35% on Synthetic Block via (B) $\rightarrow$ (D)). The MA path contributes 14%–24% in these hard regimes, consistent with its design purpose of concentrating capacity on uncertain entries.

Summary. Under MCAR the prior dominates (−65% on average) and the flow is secondary (−21%); under block missingness the prior can be harmful and the flow becomes the primary driver (−37%); the MA path contributes a consistent 18%–19% in both regimes. A complementary leave-one-out ablation, removing one component from the full model at a time (Appendix K.1), confirms that every component remains necessary: removing the prior raises MAE by 373% on average, context encoding by 23%, the MA path by 20%, and RandomRate by 18%.

5.3 Forecasting and Generation

Forecasting. Table 4 reports forecasting at $h=24$ across three datasets with clean and corrupted histories. MissPath-FM wins all 6 settings, with margins of 25%–35% on ETTm1 and Electricity; the Synthetic MCAR margin is narrow (0.572 vs 0.573), consistent with TSFlow’s strength on smooth signals. On ETTm1 and Electricity the margin widens under history corruption, consistent with observation-aware transport being most useful when histories are partial.

Generation. Table 5 evaluates unconditional generation across three datasets. MissPath-FM wins 8 of 9 metric–dataset combinations, including the best *discriminative score* on all three datasets (0.064–0.169), indicating that generated samples are nearly indistinguishable from real data. The single loss is the predictive score on Synthetic (0.261 vs Gauss+Std 0.240), where the uninformed Gaussian flow slightly better preserves next-step dynamics on smooth sinusoidal signals. Since all entries are missing in the unconditional setting, the generation comparison isolates the benefit of training-time transport efficiency (Appendix C.2).

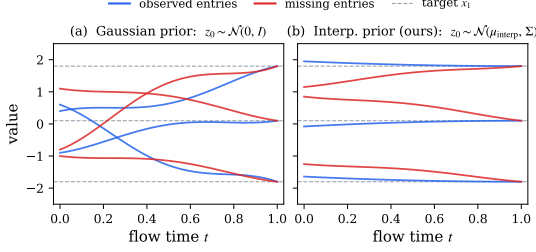


Figure 4: Transport trajectories under Gaussian prior (a) and interpolation prior (b).

Config	MAE	ℓ_{obs}	ℓ_{miss}	Ratio
Gauss.+Std	0.882	0.716	0.757	1.1×
Interp+Std	0.651	0.047	0.906	19.2×
Interp+MA	0.634	0.048	0.895	18.9×
GP+Std	0.558	0.043	0.557	13.0×
GP+MA	0.534	0.042	0.552	13.1×

Table 6: Trajectory-length analysis (Synth. Block 50%). $\ell_{\text{obs}}/\ell_{\text{miss}}$: mean arc length on observed/missing entries.

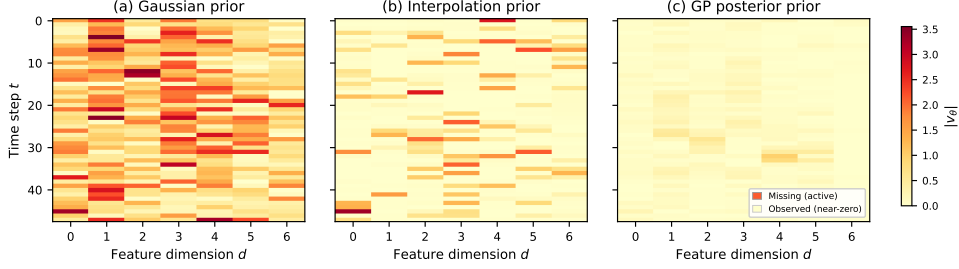


Figure 5: Velocity field magnitude $\|v_\theta\|$ at $t=0.5$ on ETTm1 MCAR 30%. (a) Gaussian prior: velocity is spread across all entries. (b) Interpolation prior: observed entries are silenced and velocity concentrates on missing entries. (c) GP posterior prior: missing-entry velocity is further reduced, since the posterior mean starts closer to the target.

5.4 Transport Geometry Analysis

Figure 4 and Table 6 directly quantify the transport reallocation on Synthetic Block 50%. Under the Gaussian prior, all entries travel long distances from noise to target; under the interpolation prior, observed entries stay near targets ($\sim 15\times$ shorter transport). The miss-to-obs ratio increases from $1.1\times$ to $19.2\times$, confirming that the velocity field concentrates on genuinely uncertain entries. The GP prior additionally shortens missing-region transport by 39%, explaining its advantage under block and MNAR settings. Figure 5 visualizes this progression at the entry level: from uniform velocity under the Gaussian prior (a), to concentrated missing-entry velocity under interpolation (b), to further reduced velocity under the GP prior (c). The same effect is visible in the final samples: measured *before* the deterministic restoration step, observed-entry MAE is 0.023–0.025 for MissPath-FM against 0.40–0.77 for a Gaussian-source flow (Appendix K.2), so observed values are held in place by the transport itself rather than by the final projection. This perspective explains why many alternatives fail: once the prior removes the dominant inefficiency, regularizers, cost-aware paths, and larger backbones have little residual room to contribute (Appendix L, H).

6 Conclusion

We presented **MissPath-FM**, a flow matching framework that treats partial observability as a transport design problem rather than an input-side conditioning mechanism. By combining structured source priors, a missingness-aware probability path, and context-conditioned velocity estimation, MissPath-FM reshapes the generative transport to concentrate on genuinely uncertain entries. MissPath-FM wins 45 of 48 evaluation settings (93.8%) across four datasets, four missingness mechanisms, and three sequence lengths against eight baselines.

Limitations and future work. Our largest tested setting is $D=36$ and $T=192$; behaviour at larger scale is not established. The advantage shrinks when observations are very sparse or unreliable: against SAITS, MissPath-FM falls behind on two of four rate-sweep settings at 90% missingness, and under observation noise beyond about 0.2 (ETTm1) to 0.3 (Electricity) z-scored units (Appendix K). The GP prior is per-variable and ignores cross-variable dependencies; a multi-output GP could exploit them, but its cost grows cubically in D , already the bottleneck at $D=36$. Downstream decision tasks and continuous-time irregular settings remain open. More broadly, we hope this work encourages treating partial observability as a *transport design problem* for generative time-series modeling.

Acknowledgments and Disclosure of Funding

The author thanks the anonymous reviewers and the area chair for their constructive feedback, which substantially improved the clarity of this paper. The author received no specific funding for this work and declares no competing interests.

References

- [1] Juan Miguel Lopez Alcaraz and Nils Strodthoff. Diffusion-based time series imputation and forecasting with structured state space models. *arXiv preprint arXiv:2208.09399*, 2022.
- [2] Wei Cao, Dong Wang, Jian Li, Hao Zhou, Lei Li, and Yitan Li. BRITS: Bidirectional recurrent imputation for time series. In *Advances in Neural Information Processing Systems*, 2018.
- [3] Zhengping Che, Sanjay Purushotham, Kyunghyun Cho, David Sontag, and Yan Liu. Recurrent neural networks for multivariate time series with missing values. *Scientific Reports*, 8(1):6085, 2018.
- [4] Yu Chen, Wei Deng, Shikai Fang, Fengpei Li, Ni Yang, Yikai Zhang, Kashif Rasul, Shandian Zhe, Anderson Schneider, and Yuriy Nevmyvaka. Provably convergent Schrödinger bridge with applications to probabilistic time series imputation. In *International Conference on Machine Learning*, 2023.
- [5] Zhichao Chen, Haoxuan Li, Fangyikang Wang, Odin Zhang, Hu Xu, Xiaoyu Jiang, Zhihuan Song, and Hao Wang. Rethinking the diffusion models for missing data imputation: A gradient flow perspective. In *Advances in Neural Information Processing Systems*, 2024.
- [6] Wenjie Du, David Cote, and Yan Liu. PyPOTS: A Python toolbox for data mining on partially-observed time series. *arXiv preprint arXiv:2305.18811*, 2023.
- [7] Wenjie Du, David Cote, and Yan Liu. SAITS: Self-attention-based imputation for time series. *Expert Systems with Applications*, 219:119619, 2023.
- [8] Dheeru Dua and Casey Graff. UCI machine learning repository. <https://archive.ics.uci.edu/ml>, 2017.
- [9] Vincent Fortuin, Dmitry Baranchuk, Gunnar Rätsch, and Stephan Mandt. GP-VAE: Deep probabilistic time series imputation. In *International Conference on Artificial Intelligence and Statistics*, 2020.
- [10] Yang Hu, Xiao Wang, Lirong Wu, Huatian Zhang, Stan Z. Li, Sheng Wang, and Tianlong Chen. FM-TS: Flow matching for time series generation. *arXiv preprint arXiv:2411.07506*, 2024.
- [11] Patrick Kidger, James Morrill, James Foster, and Terry Lyons. Neural controlled differential equations for irregular time series. In *Advances in Neural Information Processing Systems*, 2020.
- [12] Marcel Kollovich, Marten Lienen, David Lüdke, Leo Schwinn, and Stephan Günnemann. Flow matching with Gaussian process priors for probabilistic time series forecasting. In *International Conference on Learning Representations*, 2025.
- [13] Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023.
- [14] Dengyi Liu, Honggang Wang, and Hua Fang. Impute-MACFM: Imputation based on mask-aware flow matching. *arXiv preprint arXiv:2509.23126*, 2025.
- [15] Mingzhe Liu, Han Huang, Hao Feng, Leilei Sun, Bowen Du, and Yanjie Fu. PriSTI: A conditional diffusion framework for spatiotemporal imputation. In *IEEE 39th International Conference on Data Engineering (ICDE)*, pages 1927–1939, 2023.

- [16] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations*, 2023.
- [17] Jinseong Park, Seungyun Lee, Woojin Jeong, Yujin Choi, and Jaewook Lee. TimeBridge: Better diffusion prior design with bridge models for time series generation. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 1135–1146, 2026.
- [18] Weizhu Qian, Dalin Zhang, and Yan Zhao. Conditional Lagrangian Wasserstein flow for time series imputation. *arXiv preprint arXiv:2410.07550*, 2024.
- [19] Yulia Rubanova, Ricky T. Q. Chen, and David Duvenaud. Latent ordinary differential equations for irregularly-sampled time series. In *Advances in Neural Information Processing Systems*, 2019.
- [20] Satya Narayan Shukla and Benjamin M. Marlin. Multi-time attention networks for irregularly sampled time series. In *International Conference on Learning Representations*, 2021.
- [21] Ikaro Silva, George B. Moody, Daniel J. Scott, Leo Anthony Celi, and Roger G. Mark. Predicting in-hospital mortality of ICU patients: The PhysioNet/Computing in Cardiology challenge 2012. In *Computing in Cardiology*, pages 245–248, 2012.
- [22] Vaidotas Simkus and Michael U. Gutmann. CFMI: Flow matching for missing data imputation. *arXiv preprint arXiv:2506.09258*, 2025.
- [23] Yusuke Tashiro, Jiaming Song, Yang Song, and Stefano Ermon. CSDI: Conditional score-based diffusion models for probabilistic time series imputation. In *Advances in Neural Information Processing Systems*, 2021.
- [24] Michalis K. Titsias. Variational learning of inducing variables in sparse Gaussian processes. In *International Conference on Artificial Intelligence and Statistics*, 2009.
- [25] Alexander Tong, Nikolay Malkin, Guillaume Huguette, Yanlei Zhang, Jarrod Rector-Brooks, Kilian Fatras, Guy Wolf, and Yoshua Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport. *Transactions on Machine Learning Research*, 2024.
- [26] Hao Wang, Zhengnan Li, Haoxuan Li, Xu Chen, Mingming Gong, Bin Chen, and Zhichao Chen. Optimal transport for time series imputation. In *International Conference on Learning Representations*, 2025.
- [27] Huibo Xu, Runlong Yu, Likang Wu, Xianquan Wang, and Qi Liu. Bridging the last mile of prediction: Enhancing time series forecasting with conditional guided flow matching. *arXiv preprint arXiv:2507.07192*, 2025.
- [28] Jinsung Yoon, Daniel Jarrett, and Mihaela van der Schaar. Time-series generative adversarial networks. In *Advances in Neural Information Processing Systems*, 2019.
- [29] Xi Zhang, Yuan Pu, Yuki Kawamura, Andrew Loza, Yoshua Bengio, Dennis L. Shung, and Alexander Tong. Trajectory flow matching with applications to clinical time series modeling. *arXiv preprint arXiv:2410.21154*, 2024.
- [30] Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2021.

Appendix Overview

- A. Dataset and Missingness Details
- B. Implementation Details
- C. Baselines and Evaluation Metrics
- D. Proofs and Additional Method Details
- E. Sequence Length Generalization
- F. Uncertainty Calibration
- G. Ablation Details
- H. Hyperparameter Sensitivity
 - I. Extended Analysis (Transport Geometry, Gap-Bucketed)
 - J. Multi-Seed Evaluation
- K. Robustness and Additional Analyses (Leave-One-Out, Consistency, Fairness, Rate, Noise, Scale)
- L. Negative Results
- M. Reproducibility

A Dataset and Missingness Details

Table 7 summarizes the four evaluation datasets, which span three application domains (synthetic, energy, clinical) and a wide range of dimensionalities ($D \in \{7, 8, 20, 36\}$) and natural sparsity levels. All datasets use non-overlapping windows, an 80/20 train/validation split with random seed 42, and per-variable z-score normalization. PhysioNet retains its natural clinical missingness ($\sim 19\%$ observation rate); for the remaining three datasets, missingness is synthetically imposed to enable controlled evaluation.

Table 7: Dataset summary.

Dataset	Samples	T	D	Description
Synthetic	2,000	48	8	Multivariate sinusoidal (freq. 0.5–2.0, amp. 0.5–2.0, noise $\sigma=0.1$)
ETTm1	1,451	48	7	Electricity transformer temperatures
Electricity	3,000	48	20	UCI Electricity; 20 highest-variance clients
PhysioNet	4,000	48	36	ICU clinical variables (set-a); 48h hourly bins

We evaluate under four missingness mechanisms, each designed to test a different aspect of partial observability. **MCAR** (Missing Completely At Random) applies independent Bernoulli masking at rate r , serving as the simplest baseline mechanism. **Block** creates per-feature contiguous gaps of length 5–20, iterating until the target rate is reached; this tests extrapolation across temporal gaps where no local interpolation anchor exists. **MNAR** (Missing Not At Random) draws 50% of missing entries from values exceeding the 80th percentile, with the remaining 50% drawn as MCAR; this creates a systematic bias where extreme values are preferentially unobserved. **Irregular** generates per-feature random base intervals of 1–8 with jitter ± 2 , producing asynchronous observation patterns across variables. PhysioNet uses its natural clinical observation pattern with additional artificial holdout (10%–50%) for evaluation.

B Implementation Details

Velocity network. The velocity network is a 4-layer Transformer encoder with $H=4$ attention heads, $d_{\text{model}}=256$, feedforward dimension 1024, GELU activation, and dropout 0.1 ($\sim 3.9\text{M}$ parameters total). At each time step τ , the input token is formed by concatenating $[x_t \| m \| \tilde{\delta}] \in \mathbb{R}^{3D}$ and projecting to d_{model} via a linear layer. Temporal position is encoded via a learnable positional embedding. The flow time t is encoded through a sinusoidal embedding followed by an MLP

($d_{\text{model}} \rightarrow 4d_{\text{model}} \rightarrow d_{\text{model}}$) and broadcast-added to all tokens before the Transformer layers. The output head applies Linear(256 $\rightarrow$ 256) $\rightarrow$ GELU $\rightarrow$ Linear(256 $\rightarrow$ D).

Source priors. The interpolation prior uses $\sigma_{\text{obs}}=0.1$ and $\sigma_{\text{miss}}=1.0$. The GP prior uses a radial basis function (RBF) kernel with $\ell=5.0$, $\sigma_f^2=1.0$, $\sigma_n^2=0.01$; GPs are fit independently per variable with fixed hyperparameters (length scale selected via grid search, Section H).

Training. Missingness-aware (MA) path: $s_{\text{obs}}=2.0$, $s_{\text{miss}}=1.0$. Loss: CFM regression plus the observed-entry term of Eq. (9) with $\lambda_{\text{obs}}=0.1$. Optimizer: AdamW with lr 10^{-3} , weight decay 10^{-4} , gradient clipping at max norm 1.0, batch size 64. Schedule: 5-epoch linear warmup followed by cosine decay. We use *RandomRate training*: instead of training on a fixed missing rate, each sample independently draws a random MCAR rate from $\mathcal{U}(0.1, 0.7)$ at every epoch. This exposes the model to a continuum of missingness levels during training and improves generalization to unseen rates and mechanisms at test time. Default training: 500 epochs for final results.

Inference. Fixed-step Euler solver with $N_{\text{step}}=20$, $K=20$ independent samples. After integration, observed entries are deterministically restored: $\hat{x} = m \odot x^{\text{obs}} + (1 - m) \odot x_1$.

C Baselines and Evaluation Metrics

C.1 Baselines

We compare against eight external methods spanning deterministic, generative, and transport-based families. For each baseline we specify the reproduction method: PyPOTS denotes evaluation via the PyPOTS library [6], Official denotes the authors’ released codebase, and Re-impl. denotes our faithful re-implementation following the published algorithm.

- **SAITS** [7] introduces a joint-optimization training strategy with a diagonal-masked self-attention mechanism for time-series imputation. It learns to reconstruct missing values by attending to observed entries across all time steps simultaneously, producing deterministic point estimates. Evaluated via PyPOTS for both imputation and forecasting (impute-then-forecast pipeline with a shared downstream LSTM forecaster).
- **BRITS** [2] is a bidirectional recurrent model that incorporates missingness masks and time-gap information directly into the hidden-state dynamics. Forward and backward RNNs produce complementary estimates that are combined via a consistency loss. Evaluated via PyPOTS for imputation and forecasting.
- **CSDI** [23] applies conditional score-based diffusion with a Transformer backbone. It conditions on observed entries and iteratively denoises missing positions, producing calibrated uncertainty estimates via multiple samples. Evaluated via PyPOTS for imputation, forecasting, and unconditional generation.
- **GP-VAE** [9] is a variational autoencoder that enforces a Gaussian-process prior in the latent space to encourage temporally smooth representations. The GP structure provides principled uncertainty quantification over missing entries. Evaluated via PyPOTS for imputation, forecasting, and unconditional generation.
- **FlowTS** [10] adapts rectified flows for time-series generation, learning a velocity field that transports Gaussian noise to the data distribution. The Official repository supports generation only; we Re-impl. its imputation pipeline (conditioning on observed entries during ODE integration) and forecasting pipeline (impute-then-forecast) following the paper’s description.
- **TSFlow** [12] integrates Gaussian-process priors within conditional flow matching for probabilistic forecasting, using the GP posterior as the source distribution. The Official repository supports forecasting only; we Re-impl. its imputation pipeline by adapting the GP-conditioned flow to reconstruct missing entries. On Electricity ($D=20$), TSFlow’s per-sample GP Cholesky loop has $O(B \times D \times T^3)$ complexity, requiring ~ 60 minutes per epoch. We implemented a batched variant that groups samples sharing identical mask patterns and amortizes the Cholesky factorization, yielding a $\sim 600\times$ speedup while preserving identical GP posteriors. All tuning configurations converged smoothly.

- **PSW-I** [26] formulates imputation as an optimal-transport problem and uses importance-weighted samples to minimize the Wasserstein distance between imputed and true distributions. Evaluated using the Official codebase, which supports imputation only.
- **CLWF** [18] applies Lagrangian optimal-transport flows that learn a continuous-time transport map from a source distribution to the data manifold, with a Lagrangian cost function that encourages straight trajectories. No official code is available; we Re-impl. both imputation and forecasting pipelines following the published algorithm.

Why baselines are rerun rather than quoted. No prior work reports results on this exact grid (four mechanisms, two rates, and three sequence lengths under a single protocol), and published imputation numbers differ in masking convention, normalization, and window length, so quoting them would not give a like-for-like comparison. We therefore rerun every baseline under identical splits, masks, and metrics.

Fairness of comparison. Three baselines include re-implemented components: FlowTS (imputation/forecasting), TSFlow (imputation), and CLWF (all tasks). To verify that these re-implementations are competitive rather than artificially weakened, and that the comparison is fair, we note three facts. First, the three settings where MissPath-FM *loses* are all to official implementations (SAITS and BRITS via PyPOTS), not to any re-implemented baseline. Second, our re-implemented TSFlow achieves second-best performance on 6 of 32 imputation settings (all on Synthetic), indicating that the re-implementation faithfully captures the method’s strengths. Third, retraining the two strongest baselines with our RandomRate augmentation does not improve them (Appendix K.3). All re-implementations follow published algorithmic descriptions.

GP scalability. The GP posterior prior used by MissPath-FM requires a Cholesky decomposition of the $n_o \times n_o$ kernel matrix per variable, with $O(n_o^3)$ time and $O(n_o^2)$ memory, giving total per-sample cost $O(D \cdot n_o^3)$. On ETTm1 ($D=7$) and Synthetic ($D=8$), GP prior computation adds negligible overhead ($< 1\%$ of training time). On PhysioNet ($D=36$, $n_o \approx 9$ per variable on average), the total overhead accumulates to ~ 2 hours compared to ~ 5 minutes for the interpolation prior. For larger-scale settings ($T > 200$ or $D > 100$), sparse GP approximations such as inducing-point methods [24] could reduce cost to $O(n_o \cdot k^2)$ per variable; we leave exploration of these approximations to future work. Measured construction times are reported in Appendix K.6.

C.2 Evaluation Metrics

Imputation and forecasting. We report the mean absolute error (MAE) computed over missing entries only:

$$\text{MAE} = \frac{1}{|\mathcal{M}|} \sum_{(i,j) \in \mathcal{M}} |x_{ij} - \hat{x}_{ij}|, \quad (10)$$

where $\mathcal{M}$ is the set of missing positions, x_{ij} is the ground-truth value, and $\hat{x}_{ij}$ is the imputed (or forecasted) value. For probabilistic models that produce K samples, we report the MAE of the sample mean: $\hat{x}_{ij} = \frac{1}{K} \sum_{k=1}^K \hat{x}_{ij}^{(k)}$.

Generation quality. We evaluate unconditional generation following the protocol of [28] using three complementary metrics.

Discriminative score. A 2-layer LSTM classifier (hidden dim 64, dropout 0.2) is trained on a balanced 50/50 mix of real and generated sequences with an 80/20 train/test split. The score is:

$$\text{Disc} = |\text{Accuracy} - 0.5|, \quad (11)$$

where Accuracy is the test-set classification accuracy. A score near zero indicates that generated samples are indistinguishable from real data.

Predictive score. A separate 2-layer LSTM (hidden dim 64) is trained on generated data to predict the next time step. The score is the MAE evaluated on held-out real data:

$$\text{Pred} = \frac{1}{N \cdot D} \sum_{n=1}^N \sum_{d=1}^D |x_{n,T,d} - \hat{x}_{n,T,d}|, \quad (12)$$

where $\hat{x}_{n,T,d}$ is the LSTM’s prediction for the last time step. Lower values indicate that generated data preserves the temporal dynamics of the real distribution.

Spectral distance. For each variable d , we compute the power spectral density (PSD) via the fast Fourier transform (FFT) and report the mean ℓ_1 distance between the real and generated PSDs:

$$\text{Spec} = \frac{1}{D} \sum_{d=1}^D \sum_f |\text{PSD}_{\text{real}}(d, f) - \text{PSD}_{\text{gen}}(d, f)|, \quad (13)$$

where the sum over f runs over all frequency bins. Lower values indicate better preservation of the frequency structure.

Unconditional generation protocol. For unconditional generation, all entries are treated as missing ($m = 0$, $\delta = 0$), so the interpolation prior reduces to zero-fill with σ_{miss} noise and the MA path reduces to the standard linear schedule ($s_{\text{miss}} = 1$ everywhere). The source distribution thus approaches $\mathcal{N}(0, \sigma_{\text{miss}}^2 I)$. The generation comparison therefore isolates the effect of *training-time* transport efficiency: MissPath-FM’s velocity network, trained with structured priors that concentrate capacity on uncertain entries, learns a higher-quality velocity field that transfers to the unconditional setting. The Gauss+Std baseline in Table 5 uses the identical 256d/4L architecture but is trained with standard Gaussian CFM, confirming that the gains stem from the training regime rather than architectural differences.

D Proofs and Additional Method Details

Proof of Proposition 1. The decomposition follows from linearity of expectation, conditioning on m : $\mathbb{E}[\|x_1 - z_0\|^2] = \sum_{\tau,d} \mathbb{E}[(x_{1,\tau,d} - z_{0,\tau,d})^2]$, which splits into observed and missing sums by partitioning on $m_{\tau,d}$. For $z_0 \sim \mathcal{N}(0, I)$ independent of x_1 , each entry contributes $\mathbb{E}[(x_1 - z_0)^2] = \mathbb{E}[x_1^2] + \mathbb{E}[z_0^2] = \mu^2 + \sigma^2 + 1$, where μ, σ are the marginal mean and standard deviation of that entry; for per-variable z-scored data $\mu = 0$ and $\sigma = 1$, so each observed entry contributes 2 and $\mathcal{C}_{\text{obs}} = 2 n_{\text{obs}}$. For the interpolation prior, observed entries satisfy $z_0 = x_1 + \epsilon$ with $\epsilon \sim \mathcal{N}(0, \sigma_{\text{obs}}^2)$, giving $\mathbb{E}[(x_1 - z_0)^2] = \sigma_{\text{obs}}^2$ per observed entry and $\mathcal{C}_{\text{obs}} = \sigma_{\text{obs}}^2 n_{\text{obs}}$.

Prior bias–variance tradeoff. The interpolation and GP priors represent complementary points on a bias–variance spectrum. The interpolation prior has no fitted parameters and is robust to any temporal structure, but produces biased (flat) initializations in long contiguous gaps where no flanking observations exist. The GP prior can exploit temporal correlations when the kernel length scale ℓ matches the data’s characteristic frequency, reducing missing-entry transport $\mathcal{C}_{\text{miss}}$ substantially (our trajectory analysis shows -39% on missing-entry arc length); however, mismatched ℓ degrades performance severely ($+41\%$ at $\ell=8$ vs the optimal $\ell=5$; see Table 11). This sensitivity motivates our recommendation of interpolation as the default prior, with GP as an optional enhancement when domain knowledge about temporal structure is available.

Forecasting mode. When used for forecasting with horizon h , the source prior is constructed exclusively from the history window. The interpolation operator $\mathcal{I}$ uses only observed entries within the history; the future segment receives the forward-fill value with σ_{miss}^2 variance. For the GP prior, the conditioning set $(\mathbf{t}_o, \mathbf{y}_o)$ contains only history observations, and the posterior naturally extrapolates into the future with increasing uncertainty. No future information enters the prior construction.

E Sequence Length Generalization

Table 8 reports the full evaluation at $T=24$ and $T=96$ on ETTm1 with all eight baselines and all eight missingness settings per length. MissPath-FM wins all 16 settings across both lengths, and the advantage *grows* with longer sequences: the average margin over the second-best method is 19% at $T=24$ and 26% at $T=96$. This trend is expected: longer sequences provide richer interpolation anchors for the structured prior, yielding a more informative z_0 that starts closer to the target.

At $T=24$, the reduced sequence length limits the number of available observations, making interpolation less precise. Nevertheless, MissPath-FM still outperforms all baselines, suggesting that even

Table 8: Sequence length generalization on ETTm1 (MAE ↓). MissPath-FM wins all 16 settings.

T	Mechanism	Rate	TRANSPORT-BASED				DETERMINISTIC / GENERATIVE				Ours
			FlowTS	CLWF	TSFlow	PSW-I	SAITS	BRITS	CSDI	GP-VAE	
$T = 24$	MCAR	30%	0.338	0.155	0.127	0.175	<u>0.096</u>	0.117	0.162	0.427	0.079
	MCAR	50%	0.361	0.181	0.140	0.195	<u>0.106</u>	0.141	0.186	0.582	0.090
	Block	30%	0.396	0.203	0.237	0.339	<u>0.181</u>	0.196	0.325	0.560	0.122
	Block	50%	0.400	<u>0.207</u>	0.254	0.364	0.235	0.264	0.365	0.609	0.145
	MNAR	30%	0.319	0.253	0.176	0.262	<u>0.127</u>	0.200	0.215	0.632	0.118
	MNAR	50%	0.348	0.264	0.191	0.262	<u>0.158</u>	0.241	0.229	0.752	0.134
	Irregular	30%	0.372	0.302	0.152	0.231	<u>0.131</u>	0.159	0.279	0.749	0.107
	Irregular	50%	0.365	0.246	0.148	0.228	<u>0.132</u>	0.157	0.272	0.748	0.110
$T = 96$	MCAR	30%	0.427	0.224	0.125	0.129	0.112	<u>0.108</u>	0.459	0.474	0.079
	MCAR	50%	0.487	0.282	0.136	0.147	0.132	<u>0.127</u>	0.571	0.601	0.090
	Block	30%	0.472	0.259	0.282	0.278	<u>0.176</u>	0.185	0.271	0.533	0.128
	Block	50%	0.533	0.331	0.296	0.311	<u>0.212</u>	0.251	0.447	0.583	0.169
	MNAR	30%	0.492	0.431	0.198	0.212	<u>0.181</u>	0.185	0.371	0.845	0.121
	MNAR	50%	0.539	0.459	0.292	0.261	0.253	<u>0.250</u>	2.012	0.840	0.179
	Irregular	30%	0.498	0.465	0.150	0.168	0.159	<u>0.148</u>	4.912	0.728	0.109
	Irregular	50%	0.501	0.414	<u>0.146</u>	0.167	0.152	0.156	0.617	0.733	0.121

coarse interpolation is sufficient to provide a meaningful geometric advantage over an uninformed Gaussian source. At $T=96$, the advantage is particularly pronounced under MNAR missingness (MNAR 30%: -33% vs SAITS, the second-best), because longer sequences amplify the benefit of structured priors in recovering systematically missing extreme values. The strongest competitor shifts from SAITS at $T=24$ (which benefits from attention-based point estimates on short sequences) to BRITS at $T=96$ (which benefits from longer recurrent context).

F Uncertainty Calibration

Our main evaluation uses MAE of the sample mean, which assesses point-estimate quality but not distributional calibration. To address this, we evaluate four complementary probabilistic metrics on 8 representative settings ($4 \text{ datasets} \times \{\text{MCAR } 30\%, \text{Block } 30\%\}$) against the three probabilistic baselines that produce sample ensembles: CSDI, TSFlow, and GP-VAE.

CRPS (Continuous Ranked Probability Score) jointly measures calibration and sharpness. **ACE** (Average Calibration Error) computes the mean absolute gap between nominal and empirical coverage across 19 quantile levels (5%–95%); a perfectly calibrated model achieves $\text{ACE} = 0$. **IS₉₀** (Interval Score at 90%) combines interval width with a penalty for non-coverage. **Energy Score** is the multivariate generalization of CRPS, capturing cross-feature dependencies. All metrics are ↓ (lower is better).

Table 9 shows that MissPath-FM’s advantage extends beyond point estimates to full distributional quality. Across $4 \text{ metrics} \times 8 \text{ settings} = 32 \text{ comparisons}$, MissPath-FM wins **29** (90.6%), with a clean sweep on the multivariate Energy Score (8/8). Two of the three losses are to CSDI on Electricity Block 30% (CRPS and IS₉₀), the same regime where deterministic baselines also compete strongly (Table 2), suggesting that this regime benefits from CSDI’s iterative refinement; the third is ACE on ETTm1 Block 30%, where TSFlow is better calibrated. Notably, ACE values are consistently below 0.10 for MissPath-FM (except Synth Block: 0.104), indicating well-calibrated uncertainty across quantile levels. GP-VAE achieves $\text{ACE} = 0.50$ everywhere, confirming that its posterior collapses to a point estimate and provides no meaningful uncertainty.

G Ablation Details

Table 3 in the main text reports the incremental ablation across 8 settings. Table 10 summarizes the two key transitions: the gain from introducing a structured prior over the Gaussian baseline (A→B), and the gain from adding the full flow pipeline over the best prior-only baseline (B→F).

Table 9: Uncertainty calibration (all metrics ↓). MissPath-FM wins 29/32 metric–setting combinations, including a clean sweep on the multivariate Energy Score.

Setting	CRPS ↓				ACE ↓			
	Ours	CSDI	TSFlow	GP-VAE	Ours	CSDI	TSFlow	GP-VAE
ETTM1 MCAR30	0.053	0.075	0.081	0.524	0.018	0.073	0.123	0.500
ETTM1 Block30	0.076	0.121	0.173	0.559	0.050	0.167	0.025	0.500
Synth MCAR30	0.069	0.079	0.089	0.798	0.046	0.064	0.249	0.500
Synth Block30	0.245	0.324	0.475	0.799	0.104	0.111	0.203	0.500
Electr MCAR30	0.065	0.106	0.110	0.349	0.052	0.091	0.288	0.500
Electr Block30	0.171	0.127	0.358	0.368	0.098	0.110	0.302	0.500
Physio Art10	0.165	0.170	0.330	0.579	0.028	0.037	0.309	0.500
Physio Block30	0.194	0.197	0.345	0.600	0.040	0.050	0.300	0.500
Wins	7	1	0	0	7	0	1	0

Setting	Interval Score ₉₀ ↓				Energy Score ↓			
	Ours	CSDI	TSFlow	GP-VAE	Ours	CSDI	TSFlow	GP-VAE
ETTM1 MCAR30	0.475	0.689	0.945	10.48	0.784	1.124	1.220	6.873
ETTM1 Block30	0.718	1.274	1.544	11.18	1.093	1.732	2.395	7.331
Synth MCAR30	0.575	0.670	1.138	15.96	0.932	1.078	1.250	10.03
Synth Block30	2.268	2.909	6.057	15.99	3.652	5.078	7.015	10.25
Electr MCAR30	0.608	0.983	1.591	6.988	1.679	2.645	2.746	7.914
Electr Block30	1.365	1.254	4.817	7.361	3.097	3.143	7.865	8.258
Physio Art10	1.681	1.701	5.374	11.59	1.779	1.795	2.919	4.557
Physio Block30	1.825	1.988	5.469	12.00	3.405	3.611	5.563	8.307
Wins	7	1	0	0	8	0	0	0

Table 10: Prior and flow contributions (%). A→B: best prior-only vs Gaussian+Flow; B→F: Full MissPath-FM vs best prior-only.

Dataset	Setting	A→B (Prior)	B→F (Flow)
ETTM1	MCAR 30%	−85%	−31%
ETTM1	Block 30%	−70%	−51%
Synthetic	MCAR 30%	−87%	−10%
Synthetic	Block 30%	−21%	−41%
Electricity	MCAR 30%	−53%	−52%
Electricity	Block 30%	+79%	−71%
PhysioNet	Art. 10%	−43%	−37%
PhysioNet	Block 30%	−41%	−30%
Average		−40%	−40%

Two patterns stand out. First, the prior gain (A→B) is large under MCAR (−43% to −87%), where scattered missing entries provide abundant flanking observations for interpolation. However, under block missingness the prior can be *harmful*: on Electricity Block 30%, the best prior-only MAE (0.740, GP) is 79% worse than the Gaussian+Flow baseline (0.414), because contiguous gaps spanning 5–20 time steps leave no nearby anchors for interpolation or GP extrapolation. Second, the flow gain (B→F) is uniformly negative across all 8 settings (−10% to −71%), confirming that the learned velocity field *always* improves over the best prior-only baseline. The flow gain is largest precisely where the prior struggles most: Electricity Block (−71%) and ETTM1 Block (−51%), demonstrating the complementary roles of prior and flow. Notably, the average contributions of prior and flow are now equal (−40% each), underscoring that both components are indispensable.

Context encoding. On Synthetic Block 50%, we ablated four context encoding modes: no_mask (0.687, +10.4% vs full), mask_only (0.662, +6.4%), gap_only (0.641, +3.1%), and full mask+gap (**0.622**, baseline). Both the missingness mask and normalized time gaps contribute independently, with the mask providing the larger single contribution. Notably, adding raw observed values on top of the full context (full_obs: 0.645, +3.7%) slightly *degrades* performance, confirming that the structured source prior already encodes observation information into z_0 , making the raw signal redundant as an additional network input. This finding simplifies the architecture: a 3-channel input (x_t , mask, gaps) is sufficient.

Path family and loss. With the final configuration, the MA path improves over the standard linear path by 12%–26% across all eight ablation settings (Table 3, (E)→(F)), and removing it from the full model raises MAE by 20% on average (Appendix K.1). An OT-like path, which replaces the smooth power reparameterization with a cost-aware coupling, did not improve over the MA path (Appendix L), suggesting that the interpolation prior already minimizes the local transport cost that an OT path would otherwise address. For the loss, an early study with a smaller model on Synthetic Block 50% found that the observed-entry term ($\lambda_{\text{obs}}=0.1$) changed MAE by +2.1% and gap-aware weighting by +1.9%, i.e., loss-side terms matter little once the prior anchors observed entries. The final configuration keeps the observed-entry term at $\lambda_{\text{obs}}=0.1$ (Eq. 9), the default of our implementation, and every reported result uses it; this weight was not re-tuned at the final model size. Gap-aware weighting is not used.

Table 11: Hyperparameter sensitivity. Each block varies one axis; defaults in **bold**.

Axis	Value	MAE	Dataset / Setting
Solver	Euler-10	0.550	Synthetic Block 50%
	Euler-20	0.470	
	Euler-50	0.476	
	Euler-100	0.482	
	Adaptive RK23	0.481	
GP ℓ	2.0	0.517	Synthetic Block 50%
	3.0	0.525	
	5.0 (RBF)	0.470	
	5.0 (Mat. 5/2)	0.528	
	8.0	0.663	
s_{obs}	1.0	0.0861	ETTM1 MCAR 30%
	1.5	0.0812	
	2.0	0.0754	
Capacity	128d / 3L / 500ep	0.105	ETTM1 MCAR 30%
	256d / 4L / 500ep	0.081	
	384d / 4L / 500ep	0.095	
	256d / 4L / 500ep + RR	0.075	

H Hyperparameter Sensitivity

Table 11 consolidates sensitivity analyses across four hyperparameter dimensions: ODE solver, GP kernel length scale, MA path speed parameter, and model capacity. The s_{obs} rows are separately trained models with the final configuration (the noise-free column of Table 18); $s_{\text{obs}}=1.0$ is the standard linear path. Each block varies one axis while keeping all others at their default values.

I Extended Analysis

I.1 Transport Geometry

Table 6 (main text) quantifies the arc-length analysis visualized in Figure 4. Arc lengths are computed by numerically integrating $\|v_\theta(x_t, t)\|$ along the ODE trajectory for each entry. Under the Gaussian

prior, observed and missing entries traverse nearly identical distances ($\ell_{\text{obs}} = 0.716$, $\ell_{\text{miss}} = 0.757$, ratio $1.1\times$), confirming that the velocity field wastes half its capacity on already-known values. The interpolation prior reduces ℓ_{obs} by $15\times$ (from 0.716 to 0.047) while slightly increasing ℓ_{miss} (from 0.757 to 0.906), yielding a miss-to-obs ratio of $19.2\times$. This confirms the core mechanism: the structured prior concentrates the velocity field on genuinely uncertain entries.

The GP prior further reduces ℓ_{miss} by 39% (from 0.906 to 0.557) compared to interpolation, because the GP posterior mean provides a better starting estimate for missing entries, reducing the distance the flow must traverse. Interestingly, the MA path has minimal effect on arc lengths ($19.2\times$ vs $18.9\times$), suggesting that its benefit operates through the temporal *scheduling* of transport rather than through reducing total transport distance.

I.2 Gap-Bucketed Analysis

Table 12 stratifies reconstruction error by gap length on Synthetic Block 50%, distinguishing medium gaps (5–12 contiguous missing steps) from long gaps (> 12 steps). GP priors yield the largest gains on medium gaps (-34.8% over interpolation, from 0.456 to 0.297), where the GP posterior mean can leverage observations flanking the gap at moderate distance to produce an informative estimate. Long gaps remain challenging for all methods: even GP+MA achieves 0.607, far from the medium-gap performance of 0.296. This highlights a fundamental limitation: when no nearby observations exist within the kernel’s effective range, neither the prior nor the velocity network can fully resolve the uncertainty, and the model must rely on global distributional statistics rather than local temporal structure.

Table 12: Gap-bucketed MAE (Synth. Block 50%). GP priors benefit medium gaps most; long gaps remain challenging for all configurations.

Prior+Path	Medium gaps	Long gaps
Gaussian+Std	0.890	0.877
Interp+Std	0.456	0.684
GP+Std	0.297	0.618
GP+MA	0.296	0.607

J Multi-Seed Evaluation

Table 13 reports MissPath-FM results across three random seeds (42, 123, 456) on three representative settings that span different datasets and missingness types. All settings show low variance (std/mean $< 2.4\%$), and the worst individual seed still outperforms all baselines reported in Table 2. For example, on ETTm1 MCAR 30%, the worst seed achieves 0.078, still 19% below SAITS (0.096), the strongest baseline on this setting. This confirms that MissPath-FM’s performance gains are robust to random initialization and not an artifact of seed selection.

Table 13: Multi-seed evaluation (MAE $\downarrow$, 3 seeds). Low variance confirms reproducibility.

Setting	Seed 42	Seed 123	Seed 456	Mean $\pm$ Std
ETTm1 MCAR 30%	0.075	0.078	0.077	0.077 ± 0.001
Synth. Block 50%	0.470	0.453	0.473	0.465 ± 0.009
Electr. MCAR 30%	0.091	0.092	0.089	0.091 ± 0.001

K Robustness and Additional Analyses

This section reports six additional studies: a leave-one-out ablation, observed-entry consistency before restoration, baselines retrained with our training augmentation, a missing-rate sweep, an observation-noise study, and an evaluation at longer sequence length together with the cost of the GP prior. All runs use the global configuration of Section 4 and a shared evaluation harness; absolute

numbers can differ slightly from Tables 2 and 3, which were produced in separate runs, but all comparisons within a table are paired (same split, masks, and seed).

K.1 Leave-one-out ablation

The ablation in Table 3 is incremental. Table 14 gives the subtractive view: starting from the full model, one component is removed and everything else is held fixed (seed 42, 500 epochs). Every component remains necessary. The structured prior dominates (+373% on average), while context encoding (+23%), the MA path (+20%), and RandomRate (+18%) contribute comparable amounts. RandomRate has no effect on PhysioNet, whose missingness is natural rather than imposed, consistent with its role as robustness to the synthetic missing rate.

Table 14: Leave-one-out ablation (MAE ↓; in parentheses, change relative to Full). [†]PhysioNet uses a fresh draw of the artificial holdout mask in this harness, so its absolute MAE differs by 4–6% from Tables 2 and 3.

Setting	Full	–Prior	–MA path	–Context	–RandomRate
ETTm1 MCAR30	0.0754	0.7437 (+886%)	0.0862 (+14%)	0.0895 (+19%)	0.0860 (+14%)
ETTm1 Block30	0.1072	0.6953 (+548%)	0.1292 (+21%)	0.1332 (+24%)	0.1293 (+21%)
Synth. MCAR30	0.0999	0.8056 (+706%)	0.1191 (+19%)	0.1088 (+9%)	0.1262 (+26%)
Synth. Block30	0.3586	0.8090 (+126%)	0.4869 (+36%)	0.4911 (+37%)	0.5659 (+58%)
Electr. MCAR30	0.0920	0.4177 (+354%)	0.1112 (+21%)	0.1101 (+20%)	0.1184 (+29%)
Electr. Block30	0.2241	0.4027 (+80%)	0.2454 (+10%)	0.2751 (+23%)	0.2181 (−3%)
Physio Art10 [†]	0.2371	0.6178 (+161%)	0.2968 (+25%)	0.3075 (+30%)	0.2370 (−0%)
Physio Block30 [†]	0.2771	0.6204 (+124%)	0.3257 (+18%)	0.3377 (+22%)	0.2766 (−0%)
Average change	–	+373%	+20%	+23%	+18%

K.2 Observed-entry consistency before restoration

At inference, observed entries are restored deterministically after ODE integration (Section 3.4). To check whether the transport itself keeps observed values in place, or whether the restoration step does all the work, Table 15 measures observed-entry MAE at the end of integration, *before* restoration, together with the mean velocity magnitude on observed entries along the trajectory. The structured source reduces the pre-restoration error on observed entries by 16–32× and the observed-entry velocity by 22–28× relative to a Gaussian source, so restoration only removes a small residual. This is the pre-entry counterpart of Proposition 1.

Table 15: Observed entries before deterministic restoration. Gaussian source: Gaussian prior with standard path. Ours: interpolation prior with MA path.

Setting	Observed-entry MAE ↓		Mean $\ v_\theta\ $ on observed entries	
	Gaussian source	Ours	Gaussian source	Ours
ETTm1 MCAR30	0.7418	0.0229	0.9536	0.0394
ETTm1 Block30	0.7663	0.0236	0.9661	0.0398
Electr. MCAR30	0.3993	0.0247	1.0313	0.0371
Physio Art10	0.6153	0.0253	0.8068	0.0362

K.3 Baselines with the same training augmentation

MissPath-FM is trained with RandomRate, drawing a new MCAR rate from $\mathcal{U}(0.1, 0.7)$ per sample and epoch. The baselines, as run, do not see this distribution: PyPOTS SAITS and BRITS train with a fixed artificial missing rate matched to the test rate (plus PyPOTS’s internal fixed-rate masking), and our TSflow imputation re-implementation trains at the fixed test rate. To check that our margins do not come from this difference, we retrained the two strongest baselines with the same RandomRate augmentation (Table 16). The augmentation does not help them consistently: SAITS is flat or slightly worse in all six settings, and TSflow is mixed and often worse. RandomRate therefore

helps in combination with the observation-aware prior and path rather than as a generic trick. The Electricity Block loss is unchanged by the augmentation. TSFlow was not run on PhysioNet, where its per-sample GP prior is prohibitively slow at $D=36$ (Appendix K.6).

Table 16: Baselines retrained with RandomRate (+RR). MAE ↓; best per row in **bold**.

Setting	SAITS	SAITS+RR	TSFlow	TSFlow+RR	Ours
ETTM1 MCAR30	0.0942	0.0950	0.1190	0.1173	0.0751
ETTM1 Block30	0.1432	0.1468	0.2401	0.3092	0.1067
Electr. MCAR30	0.1583	0.1599	0.1584	0.1465	0.0920
Electr. Block30	0.1836	0.1905	0.3412	0.5414	0.2241
Physio Art10	0.2336	0.2346	–	–	0.2271
Physio MNAR50	0.3138	0.3162	–	–	0.3283

K.4 Missing-rate sweep

Table 17 evaluates missing rates from 10% to 90%. MissPath-FM and Gaussian+Flow are RandomRate-trained, so a single checkpoint of each is evaluated at every rate; SAITS is trained separately per rate. Comparing against Gaussian+Flow isolates how much the structured prior contributes. Its advantage narrows as observations become scarcer but does not vanish: from +91% to +74% on ETTm1 MCAR, +87% to +47% on ETTm1 Block, +79% to +32% on Electricity MCAR, and +53% to +16% on Electricity Block. Gaussian+Flow is essentially flat across rates, since it cannot exploit observations. Against SAITS, MissPath-FM leads through 70% on ETTm1 (both mechanisms) and Electricity MCAR, and falls behind at 90% on ETTm1 Block and Electricity MCAR: with so few observed anchors, interpolation is no longer informative. Electricity Block favours SAITS at every rate, matching Table 2.

Table 17: Missing-rate sweep (MAE ↓). Gauss.: Gaussian prior with flow matching. Best per row and mechanism in **bold**.

Dataset	Rate	MCAR			Block		
		Ours	Gauss.	SAITS	Ours	Gauss.	SAITS
ETTM1	10%	0.069	0.752	0.085	0.084	0.671	0.120
	30%	0.076	0.751	0.098	0.106	0.709	0.149
	50%	0.086	0.742	0.110	0.144	0.712	0.210
	70%	0.103	0.738	0.132	0.219	0.714	0.252
	90%	0.189	0.740	0.212	0.389	0.728	0.302
Electricity	10%	0.086	0.409	0.145	0.194	0.417	0.178
	30%	0.092	0.412	0.158	0.220	0.411	0.187
	50%	0.101	0.413	0.169	0.254	0.424	0.195
	70%	0.123	0.413	0.185	0.303	0.437	0.204
	90%	0.279	0.413	0.221	0.391	0.465	0.252

K.5 Noisy observations

The MA path locks in observed entries early, which could over-anchor the flow if observations are unreliable. We add Gaussian noise of standard deviation σ_{noise} to the observed test entries (targets stay clean; MAE is computed on missing positions) and compare three separately trained models with observed-entry speed $s_{\text{obs}} \in \{2.0, 1.5, 1.0\}$; $s_{\text{obs}}=1.0$ is the standard path. Aggressive locking ($s_{\text{obs}}=2.0$) is best at every noise level on both datasets, and the order $2.0 < 1.5 < 1.0$ never reverses (Table 18). The gap between $s_{\text{obs}}=2.0$ and 1.0 stays roughly constant (0.008–0.013 on ETTm1, 0.017–0.020 on Electricity), so the MA path does not become a liability under noise; its relative benefit shrinks as observations degrade (on Electricity, 21% at $\sigma_{\text{noise}}=0$ and 9% at 0.5). MissPath-FM leads SAITS up to $\sigma_{\text{noise}} \approx 0.2$ on ETTm1 and 0.3 on Electricity; beyond that SAITS is more robust. Raising σ_{obs} at test time to loosen the prior does not help (ETTM1 at $\sigma_{\text{noise}}=0.2$: 0.1256, 0.1308, 0.1395 for $\sigma_{\text{obs}} = 0.1, 0.2, 0.3$; Electricity shows the same trend), because it departs from the training-time value.

Table 18: Observation-noise sensitivity (MCAR 30%, MAE $\downarrow$). Best per row in **bold**.

Dataset	σ_{noise}	$s_{\text{obs}}=2.0$	$s_{\text{obs}}=1.5$	$s_{\text{obs}}=1.0$	SAITS
ETTM1	0.0	0.0754	0.0812	0.0861	0.1055
	0.1	0.0902	0.0961	0.0990	0.1109
	0.2	0.1256	0.1325	0.1363	0.1323
	0.3	0.1731	0.1794	0.1862	0.1712
	0.5	0.2888	0.2944	0.2963	0.2557
Electricity	0.0	0.0921	0.1033	0.1114	0.1557
	0.1	0.1002	0.1097	0.1193	0.1543
	0.2	0.1186	0.1303	0.1385	0.1583
	0.3	0.1399	0.1518	0.1589	0.1651
	0.5	0.1854	0.1969	0.2028	0.1768

K.6 Longer sequences and cost of the GP prior

Table 19 evaluates ETTm1 at $T=192$, four times the default length. MissPath-FM (interpolation prior) is trained once with RandomRate and evaluated on all three mechanisms; baselines are trained per mechanism. MissPath-FM wins all three mechanisms with no degradation relative to $T=48$, trains $3.3\times$ faster than BRITS, and peaks at about 1 GB of GPU memory. Table 20 isolates the cost of constructing the source prior. The GP posterior prior is $23\text{--}39\times$ slower than interpolation, which is why interpolation is the default and the only practical choice at scale.

Table 19: ETTm1 at $T=192$, rate 30% (MAE $\downarrow$), with training time and peak GPU memory.

Method	MCAR	Block	MNAR	Train (s)	Peak memory
SAITS	0.1222	0.1846	0.2032	95	565 MB
BRITS	0.1180	0.2294	0.2020	1541	242 MB
TSFlow	0.1149	0.2727	0.1967	696	306 MB
Ours	0.0775	0.1181	0.1235	466	1037 MB

Table 20: Source-prior construction time per batch ($B=64$, $D=7$).

Length T	GP posterior prior	Interpolation prior	Ratio
48	601 ms	15.5 ms	$39\times$
192	1316 ms	56.5 ms	$23\times$

L Negative Results

Table 21 summarizes design alternatives that we explored but that failed to improve over our default configuration. We report these explicitly because they collectively reinforce the paper’s central thesis: when the source prior is well-designed, most “obvious” improvements to the velocity network, loss function, solver, or path become redundant. The failure of OT-like paths confirms that the interpolation prior already achieves near-optimal transport allocation, making explicit cost-aware couplings unnecessary. Gap-aware loss weighting likewise failed to help, indicating that once the prior anchors observed entries, reweighting the loss has little left to contribute; the only loss-side term we keep is the light observed-entry term of Eq. (9) (Appendix G).

M Reproducibility

All experiments are implemented in PyTorch and executed on NVIDIA A5000 GPUs. We use a unified 80/20 train/validation split and identical missingness masks across all methods. All eight external baselines are evaluated using their official codebases or faithful re-implementations under the same data processing and metric pipeline (Appendix C). Typical wall-clock training time per model on a single GPU: ~ 6 minutes for ETTm1, ~ 3 minutes for Electricity, ~ 8 minutes for Synthetic, and

Table 21: Negative or neutral design choices.

Design choice	Outcome
OT-like path	No improvement over MA path
Gap-aware loss	+1.9%; disrupts velocity field
Adaptive solver (RK23)	$5\times$ higher NFE, no improvement
Matérn kernels	Worse than RBF at matched ℓ
EMA (exp. moving avg.)	Neutral or negative
Hybrid prior (Interp+GP)	Worse than either pure prior
Spline prior	Comparable to interpolation

~ 32 minutes for PhysioNet ($D=36$, 4,000 samples). The project evaluates over 200 configurations across all experiments; all reported numbers are frozen in the final results registry with no post-hoc selection. Multi-seed evaluation (Table 13) confirms low variance (std/mean $< 2.4\%$). Code and scripts to train and evaluate MissPath-FM, and to reproduce the main imputation results and the component ablations, are available at <https://github.com/zgp110/misspath-fm>.

NeurIPS Paper Checklist

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper’s contributions and scope?

Answer: [Yes]

Justification: The main claims in the abstract and introduction accurately reflect the contributions and scope of MissPath-FM, including the design of structured priors, missingness-aware path, and empirical evaluation across 48 experimental settings.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [Yes]

Justification: Section 6 states the tested scale ($D \leq 36$, $T \leq 192$), the missing-rate and observation-noise ranges beyond which a deterministic baseline is stronger, the per-variable GP prior and its cost, and the evaluation scope; Appendix K gives the supporting experiments.

3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: The paper includes Proposition 1 with its assumptions stated in Section 3.1, and the complete proof is provided in Appendix D.

4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: All algorithmic details, dataset splits, hyperparameters, and evaluation protocols are fully described in the main paper and appendix, enabling reproducibility of the main experimental results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: Code and instructions to train and evaluate MissPath-FM and to reproduce the main imputation results and ablations are publicly available at <https://github.com/zgp110/misspath-fm> (Appendix M).

6. Experimental setting/details

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: The paper specifies training/testing details including dataset splits, missingness mechanisms, sequence lengths, hyperparameters, and optimizer settings in both the main paper and supplemental material.

7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: We report three-seed means and standard deviations for representative settings (Appendix J; coefficient of variation below 2.4%), and all robustness comparisons use paired splits, masks, and seeds (Appendix K). Error bars are not reported for every cell of the main table because of the number of settings.

8. **Experiments compute resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: The type of compute (GPUs), memory, and approximate runtime for all experiments are reported in the main paper and supplemental material.

9. **Code of ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [Yes]

Justification: The research fully conforms to the NeurIPS Code of Ethics, with no ethical violations in data, methodology, or reporting.

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [N/A]

Justification: The work focuses on methodological improvements for partially observed time series and does not have direct positive or negative societal impacts.

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The paper does not release high-risk models, pre-trained generative models, or scraped datasets, so no special safeguards are needed.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All datasets and code baselines used in the paper are properly cited and licenses are respected.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: The code for MissPath-FM is released publicly with documentation and scripts to reproduce the main imputation results and ablations.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The research does not involve human subjects or crowdsourced experiments.

15. Institutional review board (IRB) approvals or equivalent for research with human subjects

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The paper does not involve human subjects, so no IRB approvals are required.

16. Declaration of LLM usage

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: LLMs were not used as any important, original, or non-standard component of the core research methodology.