Flow-based Conformal Prediction for Multi-dimensional Time Series

Junghwan Lee, Chen Xu, Yao Xie
H. Milton Stewart School of Industrial and Systems Engineering
Georgia Institute of Technology
{jlee3541,cxu310}@gatech.edu, yao.xie@isye.gatech.edu

Abstract

Time series prediction is a crucial task in sequential decision-making. With the increasing use of black-box models for time series prediction, the need for uncertainty quantification has become more critical. Conformal prediction has gained attention as a reliable uncertainty quantification framework. However, conformal prediction for time series faces two key challenges: (1) effectively leveraging sequential correlations in features and non-conformity scores, and (2) handling multi-dimensional outcomes. To address these challenges, we propose a novel conformal prediction method for time series using flow with classifier-free guidance. We provide theoretical guarantees by establishing an exact non-asymptotic marginal coverage and a finite-sample bound on conditional coverage for our method. Evaluations on real-world multi-dimensional time series datasets demonstrate that our method constructs significantly smaller prediction sets while maintaining target coverage, outperforming existing baselines.

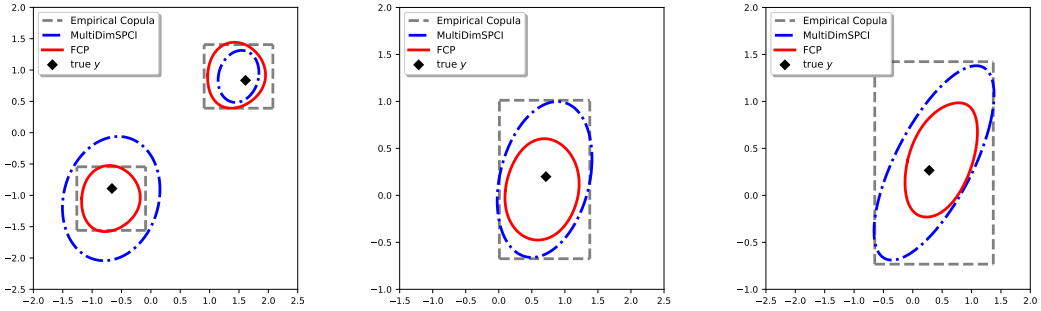
1 Introduction

Uncertainty quantification has become essential in scientific fields where black-box machine learning models are increasingly deployed [2]. Conformal prediction (CP) provides a distribution-free framework for uncertainty quantification by constructing prediction sets using three key components: a base prediction model, features, and observed outcomes [38, 45]. By computing non-conformity scores that quantify how atypical predicted values are relative to past observations, CP generates reliable prediction sets that satisfy a target confidence level.

Time series prediction aims to forecast future outcomes based on past sequential observations of features [8]. Recent advances in machine learning have led to the development of various foundation models designed for time series prediction [32, 46]. The widespread adoption of such models for time series prediction underscores the pressing need for reliable uncertainty quantification. Although CP has emerged as a powerful framework for uncertainty quantification, most existing CP methods fundamentally rely on the assumption of data exchangeability [4]. The exchangeability assumption is frequently violated in time series data, where observations exhibit complex temporal dependencies and stochastic variations that induce correlations in the non-conformity scores, thereby making the direct application of CP to time series prediction particularly challenging.

An additional challenge is that modern time series data often contain high-dimensional features and multi-dimensional outcomes. While CP methods for univariate outcomes are well-established, extending these methods to generate prediction sets for multi-dimensional outcomes is not straightforward and requires careful consideration in constructing prediction sets. Although some prior studies have proposed methods to generate prediction sets for multi-dimensional outcomes using copulas [30] or ellipsoidal uncertainty sets [31], these approaches still rely on the exchangeability assumption, thus

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: MLxOR: Mathematical Foundations and Operational Integration of Machine Learning for Uncertainty-Aware Decision-Making.



(a) Wind dataset (b) Traffic dataset (c) Solar dataset

Figure 1: Comparison of the prediction sets with a target coverage of 0.95, constructed by FCP (ours), MultiDimSPCI, and conformal prediction using empirical copulas on (a) the wind dataset, (b) the traffic dataset, and (c) the solar dataset. The prediction sets were manually selected from the test set for clear comparison.

limiting their applicability to time series data. Consquently, an effective CP method for time series prediction requires to address the two aformentioned challenges: leaveraging correlations in features and non-conformity scores and handling multi-dimensional outcomes. However, there remains a lack of studies that attempt to tackle both challenges simultaneously. For a comprehensive review of related work, we refer readers to Appendix A.

In this work, we propose a novel conformal prediction method designed for time series prediction with multi-dimensional outcomes. Our method is designed to effectively address the aforementioned two challenges by using flow with classifier-free guidance. Specifically, we use a flow to model the distribution of prediction residuals and their transformations conditioned on historical context, which is encoded using Transformer. We define the non-conformity score as the Euclidean distance between the transformed prediction residual and the mean of a Gaussian source distribution of the flow. This allows us to construct prediction sets at a desired confidence level directly in the source distribution space of the flow. We provide theoretical guarantees by establishing an exact non-asymptotic marginal coverage and a finite-sample bound on conditional coverage for our method. Evaluations on three real-world multi-dimensional time series datasets demonstrate that our method constructs significantly smaller prediction sets while maintaining target coverage, outperforming existing baselines.

2 Conformal Prediction for Time Series using Guided Flow

Our problem setup and a comprehensive background on guided flow are provided in Appendix B and Appendix C, respectively. Our goal is to train a guided flow to model the conditional distribution of the prediction residual at time i, defined as $\hat{\epsilon}_i = \widehat{f}(x_i) - y_i$, conditioned on a guidance h that contains contextual information useful for predicting $\hat{\epsilon}_i$. We designed the Gaussian probability path as defined in equation (16), with interpolating scheduler $a_t = t$ and $\sigma_t = (1 - t)$. The source distribution was set to an isotropic Gaussian distribution with zero mean and covariance scale $\gamma > 0$.

Since explicit supervision is unavailable from the data, it is not feasible to directly learn a time-dependent classifier as required in equation (17) and (18). Instead, we extracted contextual information using an encoder applied to a context window k of the past features and residuals and modeled the conditional distribution using the classifier-free guidance as described in equation (20). The encoder was jointly trained with the guided vector field. In our method, we used Transformer [44] as the encoder, though alternative sequence models, such as recurrent neural networks (RNNs), are also applicable. The guided flow was trained using flow matching [28, 1, 29] with the loss:

$$\mathcal{L} = \mathbb{E}_{t,\eta,\hat{\epsilon}_t,(\hat{\epsilon}_i,z_i)} \left[\left\| u^{\theta}_{t|h}(\hat{\epsilon}_t \mid (1-\eta)h_i \cdot \eta h_{\varnothing}) - u^{\text{target}}_{t|h}(\hat{\epsilon}_t \mid \hat{\epsilon}_i) \right\|^2 \right], \tag{1}$$

where $t \sim \text{Unif}[0, 1]$, $\eta \sim \text{Bernoulli}(p_{\varnothing})$, $\hat{\epsilon}_t \sim p_{t|\hat{\epsilon}_1}(\cdot \mid \hat{\epsilon}_1)$, and $(\hat{\epsilon}_i, z_i) \sim q_{\text{data}}$. $h_{\varnothing} \in \mathbb{R}^{d_y}$ indicates the guidance representing the null condition and $||\cdot||$ denotes the 2-norm. $u^{\theta}_{t|h}$ denotes the guided vector field with trainable parameter set θ and $p_{\varnothing}$ is the probability of assigning the null condition $\varnothing$ instead of the guidance h during training. The context z_i is known and fixed for each $\hat{\epsilon}_i$.

Non-conformity score. The trained guided flow can model the conditional distribution $q(\hat{\epsilon}_i \mid h_i)$. Based on the guided flow ψ, we defined the non-conformity score $\hat{e}(y_i)$ for a given prediction residual $\hat{\epsilon}_i = y_i - \widehat{f}(x_i)$ as:

$$\hat{e}(y_i) = ||\psi_1^{-1}(\hat{\epsilon}_i \mid h_i)||, \tag{2}$$

where ψ_1^{-1} represents flow transformation in reverse time from $t = 1$ to $t = 0$, which transforms $\hat{\epsilon}_i$ to a sample from the source distribution conditioned on h_i. Intuitively, $\hat{e}(y_i)$ represents the Euclidean distance between the transformed residual and the origin, which corresponds to the mean of the source Gaussian distribution in our setting. This approach implicitly leverages sequential dependencies in the features and the resulting non-conformity scores, as the guidance vector h_i is constructed from the historical context and conditions both the distribution of $\hat{\epsilon}_i$ and its transformation into the non-conformity score.

Prediction set. Using the non-conformity score defined in equation (2), we defined the prediction set at significance level α as:

$$\hat{C}_{i-1}(z_i, \alpha) = \{y : \hat{e}(y_i) \leq r_{\mathcal{B}_\alpha}\}, \tag{3}$$

where $r_{\mathcal{B}_\alpha}$ denotes the radius of the ball $\mathcal{B}_\alpha$, which contains $1 - \alpha$ probability mass. Since we used an isotropic Gaussian with zero mean and covariance matrix γI as the source distribution p, the radius corresponding to the ball containing probability mass $1 - \alpha$, $r_{\mathcal{B}_\alpha}$, can be computed by $r_{\mathcal{B}_\alpha} = \sqrt{\gamma}\chi_d^{-1}(1 - \alpha)$, where χ_d^{-1} is the inverse cumulative distribution function of the chi distribution with d degrees of freedom.

Size of the prediction set. The size of the prediction set defined in equation (3) is computed by:

$$\int_{\mathcal{B}_\alpha} \left|\det\left(J_{\psi_1}(x \mid h)\right)\right| \, dx, \tag{4}$$

where ψ_1 represents the flow transformation from $t = 0$ to $t = 1$, and $J_{\psi_1}(x \mid h)$ denotes the Jacobian of ψ_1 at $x \in \mathcal{B}_\alpha$ conditioned on h. This can be approximated by using Monte Carlo estimation:

$$\text{Size}(\mathcal{B}_\alpha)\frac{1}{N}\sum_{j=1}^{N}\left|\det(J_{\psi_1}(x_j \mid h))\right|, \tag{5}$$

where x_j are i.i.d. samples drawn from $\mathcal{B}_\alpha$, and N is the number of samples. However, direct computation of $\det\left(J_{\psi_1}(x \mid h)\right)$ is expensive as it requires solving the guided flow ODE and evaluating the full Jacobian matrix. Instead, we can compute the log-determinant of the Jacobian by solving the following ODE:

$$\begin{aligned} \frac{d}{dt}\log\left|\det J_{\psi_t}(x \mid h)\right| &= \text{div}\left(u_t(\psi_t(x \mid h) \mid h)\right), && \text{(log-determinant of the Jacobian ODE)} \\ \log\det\left(J_{\psi_0}(x \mid h)\right) &= 0, && \text{(initial condition)} \end{aligned} \tag{6}$$

where $\text{div}(\cdot)$ denotes the divergence operator. A detailed derivation is provided in Proposition E.3.

The accuracy of the prediction set size estimate depends on the Monte Carlo approximation. Purely random sampling from $\mathcal{B}_\alpha$ can lead to biased estimates due to uneven coverage of the sampling space, and a small sample size N can result in high variance. To reduce bias from random sampling, we employed quasi-Monte Carlo sampling based on Sobol sequences [39, 34], which ensures more uniform sampling from $\mathcal{B}_\alpha$. To control the variance from finite sampling, we monitor the relative error in terms of sample size N. Further implementation details are provided in the experiment section.

Properties of the prediction set. The prediction sets constructed by guided flow do not have a definite geometrical shape, such as spheres or ellipsoids. Nonetheless, useful topological properties of the prediction sets can be inferred. Specifically, the prediction sets are guaranteed to be closed and connected based on Theorem E.4. Figure 1 shows the prediction sets in a 2-dimensional space constructed by our method compared to two other methods based on copula and ellipsoid. The figure visually confirms that the prediction sets constructed by our method have flexible shapes.

Coverage guarantees. Our method provides exact non-asymptotic marginal coverage guarantees and finite-sample bounds on conditional coverage. Theoretical analysis is provided in Appendix D and detailed proofs are deferred in Appendix E.

Proposition 2.1 (Marginal coverage)**.** *Let $\alpha \in (0,1)$ be a pre-specified significance level. Under Assumptions D.1 and D.3, if the ball $\mathcal{B}_\alpha$ defining the prediction set in equation (3) has probability mass $1-\alpha$, then the prediction set achieves exact marginal coverage of $1-\alpha$.*

Theorem 2.2 (Conditional coverage bound under i.i.d. non-conformity scores)**.** *Under Assumption D.1, D.6, D.7, D.9, and D.10, with probability $1-\delta$, we have:*

$$\begin{aligned} &\left|\mathbb{P}(Y_{T+1} \in \widehat{C}^{\alpha}_{T+1} \mid Z_{T+1} = z_{T+1}) - (1-\alpha)\right| \\ &\leq 12\sqrt{\frac{\log(16T)}{T}} + 4(L_{T+1} + \frac{1}{2})(2C + \delta_T). \end{aligned} \tag{7}$$

Corollary 2.3 (Conditional coverage bound under stationary and strongly mixing non-conformity scores)**.** *Under Assumption D.1, D.7, D.9, D.10, and D.16, with probability $1-\delta$, we have:*

$$\begin{aligned} &\left|\mathbb{P}(Y_{T+1} \in \widehat{C}^{\alpha}_{T+1} \mid Z_{T+1} = z_{T+1}) - (1-\alpha)\right| \\ &\leq 12\frac{(\frac{M}{2})^{1/3}(\log T)^{2/3}}{T^{1/3}} + 4(L_{T+1} + \frac{1}{2})(2C_+\delta_T). \end{aligned} \tag{8}$$

3 Experiments

Setup For notational convenience, we refer to our method as FCP. We evaluated FCP using two different architectures to model the guided vector field. The first one is MLP with Softplus activation, and the second one is iResNet [5] with Softplus activation. Both architectures are smooth and continuously differentiable, satisfying the conditions needed to ensure the existence and uniqueness of the guided flow (Assumption D.3). Moreover, the iResNet architecture satisfies the bi-Lipschitz condition of the guided flow, which is required to derive the conditional coverage bound of FCP (Assumption D.7). Details of the experiment setup and implementation details are provided in Appendix F and Appendix G, respectively.

Datasets. We evaluated FCP and baselines using three real-world time series datasets: wind, traffic, and solar datasets. For the wind and traffic datasets, we randomly selected $d_y \in \{2, 4, 8\}$ locations to construct five sequences of d_y-dimensional time series. For the solar dataset, we randomly selected $d_y \in \{2, 4\}$ locations to construct five sequences of d_y-dimensional time series. Base predictor $\hat{f}$ is required to provide a point prediction $\hat{y}$. We used two types of base predictors for each dataset: (1) leave-one-out (LOO) bootstrap multivariate linear regression, and (2) recurrent neural network (RNN) with long short-term memory (LSTM) units [21].

Baselines. We evaluated our method against several CP methods designed for multi-dimensional time series or i.i.d. data: MultiDimSPCI [51], conformal prediction using local ellipsoid [31], CopulaCPTS [42], and conformal prediction using empirical and Gaussian copulas [30]. We also included two probabilistic time series forecasting methods as baselines: TFT [27] and DeepAR [37].

Evaluation metrics. *Efficient* prediction sets are those that are as small as possible while satisfying the desired coverage. Therefore, we used two evaluation metrics: empirical coverage and the average size of the prediction sets.

Results Table 1 (see Appendix) presents the results of experiments on three real-world datasets. FCP consistently obtained smaller prediction sets than all baselines while maintaining the target coverage. The performance gains of FCP were especially notable for higher outcome dimensions. While FCP maintained stable coverage across varying d_y, baseline methods often suffered from undercoverage, or overcoverage accompanied by inflated prediction set sizes.

4 Conclusion and Future Works

In this study, we proposed a novel conformal prediction method for multi-dimensional time series using flow with classifier-free guidance. Our method provides theoretical guarantees, including exact non-asymptotic marginal coverage and finite-sample bounds on conditional coverage. Evaluations on real-world datasets demonstrated the effectiveness of the proposed approach, achieving superior performance over existing state-of-the-art methods. Future research directions include: (1) exploring alternative flow-based architectures for prediction set construction; (2) extending conformal prediction to discrete outcomes such as image labels; and (3) developing conformal prediction methods in the latent space.

References

[1] Michael S Albergo and Eric Vanden-Eijnden. Building normalizing flows with stochastic interpolants. In *11th International Conference on Learning Representations, ICLR 2023*, 2023.

[2] Anastasios N Angelopoulos and Stephen Bates. A gentle introduction to conformal prediction and distribution-free uncertainty quantification. *arXiv preprint arXiv:2107.07511*, 2021.

[3] Andreas Auer, Martin Gauch, Daniel Klotz, and Sepp Hochreiter. Conformal prediction for time series with modern hopfield networks. *Advances in Neural Information Processing Systems*, 36, 2024.

[4] Rina Foygel Barber, Emmanuel J Candes, Aaditya Ramdas, and Ryan J Tibshirani. Conformal prediction beyond exchangeability. *The Annals of Statistics*, 51(2):816–845, 2023.

[5] Jens Behrmann, Will Grathwohl, Ricky TQ Chen, David Duvenaud, and Jörn-Henrik Jacobsen. Invertible residual networks. In *International conference on machine learning*, pages 573–582. PMLR, 2019.

[6] Peter J Bickel, Ya'acov Ritov, and Alexandre B Tsybakov. Simultaneous analysis of lasso and dantzig selector. 2009.

[7] Stéphane Boucheron, Gábor Lugosi, and Olivier Bousquet. Concentration inequalities. In *Summer school on machine learning*, pages 208–240. Springer, 2003.

[8] George EP Box, Gwilym M Jenkins, Gregory C Reinsel, and Greta M Ljung. *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.

[9] Ricky TQ Chen, Jens Behrmann, David K Duvenaud, and Jörn-Henrik Jacobsen. Residual flows for invertible generative modeling. *Advances in neural information processing systems*, 32, 2019.

[10] Xiaohong Chen and Halbert White. Improved rates and asymptotic normality for nonparametric neural network estimators. *IEEE Transactions on Information Theory*, 45(2):682–691, 1999.

[11] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

[12] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016.

[13] Jacopo Diquigiovanni, Matteo Fontana, and Simone Vantini. Distribution-free prediction bands for multivariate functional time series: an application to the italian gas market. *arXiv preprint arXiv:2107.00527*, 2021.

[14] Jacopo Diquigiovanni, Matteo Fontana, and Simone Vantini. Conformal prediction bands for multivariate functional data. *Journal of Multivariate Analysis*, 189:104879, 2022.

[15] John R Dormand and Peter J Prince. A family of embedded runge-kutta formulae. *Journal of computational and applied mathematics*, 6(1):19–26, 1980.

[16] Aryeh Dvoretzky, Jack Kiefer, and Jacob Wolfowitz. Asymptotic minimax character of the sample distribution function and of the classical multinomial estimator. *The Annals of Mathematical Statistics*, pages 642–669, 1956.

[17] Zhenhan Fang, Aixin Tan, and Jian Huang. CONTRA: Conformal prediction region via normalizing flow transformation. In *The Thirteenth International Conference on Learning Representations*, 2025. URL `https://openreview.net/forum?id=p0O9cqLq7Q`.

[18] Thomas Hakon Gronwall. Note on the derivatives with respect to a parameter of the solutions of a system of differential equations. *Annals of Mathematics*, 20(4):292–296, 1919.

[19] Morris W Hirsch, Stephen Smale, and Robert L Devaney. *Differential equations, dynamical systems, and an introduction to chaos*. Academic press, 2013.

[20] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

[21] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

[22] Chancellor Johnstone and Eugene Ndiaye. Exact and approximate conformal inference in multiple dimensions. *arXiv preprint arXiv:2210.17405*, 2022.

[23] Diederik P Kingma. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[24] Michael R Kosorok. *Introduction to empirical processes and semiparametric inference*, volume 61. Springer, 2008.

[25] Jonghyeok Lee, Chen Xu, and Yao Xie. Kernel-based optimally weighted conformal prediction intervals. *arXiv preprint arXiv:2405.16828*, 2024.

[26] Junghwan Lee, Chen Xu, and Yao Xie. Transformer conformal prediction for time series. *arXiv preprint arXiv:2406.05332*, 2024.

[27] Bryan Lim, Sercan Ö Arık, Nicolas Loeff, and Tomas Pfister. Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International Journal of Forecasting*, 37(4): 1748–1764, 2021.

[28] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

[29] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *The Eleventh International Conference on Learning Representations (ICLR)*, 2023.

[30] Soundouss Messoudi, Sébastien Destercke, and Sylvain Rousseau. Copula-based conformal prediction for multi-target regression. *Pattern Recognition*, 120:108101, 2021.

[31] Soundouss Messoudi, Sébastien Destercke, and Sylvain Rousseau. Ellipsoidal conformal inference for multi-target regression. In *Conformal and Probabilistic Prediction with Applications*, pages 294–306. PMLR, 2022.

[32] John A Miller, Mohammed Aldosari, Farah Saeed, Nasid Habib Barna, Subas Rana, I Budak Arpinar, and Ninghao Liu. A survey of deep learning and foundation models for time series forecasting. *arXiv preprint arXiv:2401.13912*, 2024.

[33] J.R. Munkres. *Topology*. Featured Titles for Topology. Prentice Hall, Incorporated, 2000. ISBN 9780131816299. URL `https://books.google.com/books?id=XjoZAQAAIAAJ`.

[34] Art B. Owen. *Practical Quasi-Monte Carlo Integration*. `https://artowen.su.domains/mc/practicalqmc.pdf`, 2023.

[35] Kashif Rasul, Abdul-Saboor Sheikh, Ingmar Schuster, Urs Bergmann, and Roland Vollgraf. Multivariate probabilistic time series forecasting via conditioned normalizing flows. *arXiv preprint arXiv:2002.06103*, 2020.

[36] Emmanuel Rio et al. *Asymptotic theory of weakly dependent random processes*, volume 80. Springer, 2017.

[37] David Salinas, Valentin Flunkert, Jan Gasthaus, and Tim Januschowski. Deepar: Probabilistic forecasting with autoregressive recurrent networks. *International journal of forecasting*, 36(3): 1181–1191, 2020.

[38] Glenn Shafer and Vladimir Vovk. A tutorial on conformal prediction. *Journal of Machine Learning Research*, 9(3), 2008.

[39] Ilya M Sobol. The distribution of points in a cube and the approximate evaluation of integrals. *USSR Computational mathematics and mathematical physics*, 7:86–112, 1967.

[40] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

[41] Kamile Stankeviciute, Ahmed M Alaa, and Mihaela van der Schaar. Conformal time-series forecasting. *Advances in neural information processing systems*, 34:6216–6228, 2021.

[42] Sophia Sun and Rose Yu. Copula conformal prediction for multi-step time series forecasting. *arXiv preprint arXiv:2212.03281*, 2022.

[43] Ryan J Tibshirani, Rina Foygel Barber, Emmanuel Candes, and Aaditya Ramdas. Conformal prediction under covariate shift. *Advances in neural information processing systems*, 32, 2019.

[44] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[45] Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. *Algorithmic learning in a random world*, volume 29. Springer, 2005.

[46] Qingsong Wen, Tian Zhou, Chaoli Zhang, Weiqi Chen, Ziqing Ma, Junchi Yan, and Liang Sun. Transformers in time series: a survey. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, pages 6778–6786, 2023.

[47] Chen Xu and Yao Xie. Conformal prediction interval for dynamic time-series. In *International Conference on Machine Learning*, pages 11559–11569. PMLR, 2021.

[48] Chen Xu and Yao Xie. Conformal anomaly detection on spatio-temporal observations with missing data. *arXiv preprint arXiv:2105.11886*, 2021.

[49] Chen Xu and Yao Xie. Conformal prediction for time series. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[50] Chen Xu and Yao Xie. Sequential predictive conformal inference for time series. In *International Conference on Machine Learning*, pages 38707–38727. PMLR, 2023.

[51] Chen Xu, Hanyang Jiang, and Yao Xie. Conformal prediction for multi-dimensional time series by ellipsoidal sets. *arXiv preprint arXiv:2403.03850*, 2024.

[52] Minghe Zhang, Chen Xu, Andy Sun, Feng Qiu, and Yao Xie. Solar radiation ramping events modeling using spatio-temporal point processes. *arXiv preprint arXiv:2101.11179*, 2021. Accepted at the 2021 INFORMS Conference on Service Science (ICSS 2021).

[53] Shixiang Zhu, Hanyu Zhang, Yao Xie, and Pascal Van Hentenryck. Multi-resolution spatio-temporal prediction with application to wind power generation. *arXiv preprint arXiv:2108.13285*, 2021.

Table 1: Average empirical coverage and prediction sets sizes obtained by FCP and all baselines on three real-world datasets, evaluated under different base predictors and varying outcome dimensions d_y. Reported values represent the average and standard deviation over five independent experiments. The target confidence level was set to 0.95. Results with average empirical coverage below the target confidence level are grayed out, and the smallest prediction set sizes, excluding the grayed-out results, are highlighted in bold.

Dataset	Base Predictor	Method	$d_y = 2$		$d_y = 4$		$d_y = 8$	
			Coverage	**Size**	**Coverage**	**Size**	**Coverage**	**Size**
Wind	LOO Bootstrap	FCP (MLP)	$0.951_{\pm.018}$	$\mathbf{0.88}_{\pm.089}$	$0.953_{\pm.006}$	$3.43_{\pm1.37}$	$0.956_{\pm.010}$	$19.4_{\pm10.2}$
		FCP (iResNet)	$0.951_{\pm.021}$	$1.14_{\pm.069}$	$0.954_{\pm.014}$	$\mathbf{1.79}_{\pm.736}$	$0.953_{\pm.018}$	$\mathbf{14.8}_{\pm22.5}$
		MultiDimSPCI	$0.953_{\pm.016}$	$1.31_{\pm.524}$	$0.956_{\pm.018}$	$6.39_{\pm3.90}$	$0.951_{\pm.024}$	$205.5_{\pm161.5}$
		Local Ellipsoid	$0.964_{\pm.015}$	$1.38_{\pm.419}$	$0.971_{\pm.013}$	$8.63_{\pm5.90}$	$0.974_{\pm.011}$	$394.9_{\pm522.4}$
		Empirical Copula	$0.951_{\pm.013}$	$1.22_{\pm.316}$	$0.958_{\pm.019}$	$4.94_{\pm2.57}$	$0.948_{\pm.012}$	$77.4_{\pm26.1}$
		Gaussian Copula	$0.945_{\pm.017}$	$1.17_{\pm.289}$	$0.958_{\pm.019}$	$5.11_{\pm2.40}$	$0.948_{\pm.012}$	$77.4_{\pm26.1}$
		CopulaCPTS	$1.0_{\pm.000}$	$22.3_{\pm19.0}$	$1.0_{\pm.000}$	$611.3_{\pm484.7}$	$1.0_{\pm.000}$	$3.50 \times 10^5_{\pm3.73\times10^5}$
		TFT	$0.723_{\pm.172}$	$1.34_{\pm.588}$	$0.515_{\pm.174}$	$4.26_{\pm3.52}$	$0.187_{\pm.126}$	$6.75_{\pm3.19}$
		DeepAR	$0.909_{\pm.036}$	$1.32_{\pm.445}$	$0.672_{\pm.130}$	$4.84_{\pm3.86}$	$0.320_{\pm.160}$	$52.8_{\pm64.5}$
	LSTM	FCP (MLP)	$0.952_{\pm.054}$	$\mathbf{1.18}_{\pm.215}$	$0.957_{\pm.022}$	$10.8_{\pm1.05}$	$0.953_{\pm.056}$	$\mathbf{2.48 \times 10^3}_{\pm669}$
		FCP (iResNet)	$0.957_{\pm.034}$	$1.84_{\pm.279}$	$0.957_{\pm.018}$	$\mathbf{6.37}_{\pm2.91}$	$0.978_{\pm.015}$	$2.55 \times 10^3_{\pm1.94\times10^3}$
		MultiDimSPCI	$0.974_{\pm.009}$	$3.79_{\pm1.71}$	$0.926_{\pm.045}$	$63.9_{\pm58.4}$	$0.896_{\pm.035}$	$5.53 \times 10^3_{\pm6.31\times10^3}$
		Local Ellipsoid	$0.978_{\pm.043}$	$10.5_{\pm6.97}$	$1.0_{\pm.000}$	$354.4_{\pm406.8}$	$1.0_{\pm.000}$	$2.63 \times 10^5_{\pm2.70\times10^5}$
		Empirical Copula	$0.983_{\pm.035}$	$14.2_{\pm8.19}$	$1.0_{\pm.000}$	$494.5_{\pm196.1}$	$1.0_{\pm.000}$	$4.46 \times 10^5_{\pm9.82\times10^4}$
		Gaussian Copula	$0.983_{\pm.035}$	$14.1_{\pm8.18}$	$1.0_{\pm.000}$	$499.1_{\pm189.5}$	$1.0_{\pm.000}$	$5.24 \times 10^5_{\pm1.89\times10^5}$
		CopulaCPTS	$1.0_{\pm.000}$	$45.7_{\pm45.4}$	$1.0_{\pm.000}$	$4.82 \times 10^3_{\pm3.73\times10^3}$	$1.0_{\pm.000}$	$2.83 \times 10^7_{\pm3.28\times10^7}$
		TFT	$0.550_{\pm.321}$	$1.90_{\pm.695}$	$0.395_{\pm.195}$	$3.93_{\pm2.01}$	$0.136_{\pm.189}$	$23.7_{\pm34.8}$
		DeepAR	$0.786_{\pm.065}$	$1.69_{\pm.489}$	$0.305_{\pm.258}$	$9.88_{\pm10.1}$	$0.00_{\pm.000}$	$22.8_{\pm32.6}$
Traffic	LOO Bootstrap	FCP (MLP)	$0.957_{\pm.014}$	$\mathbf{0.915}_{\pm.119}$	$0.953_{\pm.009}$	$\mathbf{1.06}_{\pm.431}$	$0.965_{\pm.015}$	$\mathbf{1.53}_{\pm.161}$
		FCP (iResNet)	$0.950_{\pm.021}$	$1.21_{\pm.084}$	$0.959_{\pm.014}$	$1.33_{\pm.118}$	$0.970_{\pm.007}$	$2.72_{\pm.215}$
		MultiDimSPCI	$0.963_{\pm.008}$	$1.58_{\pm.446}$	$0.968_{\pm.006}$	$2.62_{\pm.908}$	$0.971_{\pm.004}$	$10.7_{\pm4.60}$
		Local Ellipsoid	$0.970_{\pm.007}$	$2.04_{\pm.505}$	$0.975_{\pm.005}$	$2.95_{\pm1.06}$	$0.980_{\pm.003}$	$3.82_{\pm1.13}$
		Empirical Copula	$0.973_{\pm.006}$	$2.35_{\pm.446}$	$0.972_{\pm.004}$	$5.61_{\pm1.48}$	$0.970_{\pm.005}$	$40.4_{\pm6.04}$
		Gaussian Copula	$0.973_{\pm.006}$	$2.37_{\pm.430}$	$0.972_{\pm.004}$	$5.61_{\pm1.48}$	$0.970_{\pm.005}$	$40.4_{\pm6.04}$
		CopulaCPTS	$1.0_{\pm.000}$	$21.6_{\pm16.3}$	$1.0_{\pm.000}$	$645.8_{\pm645.5}$	$1.0_{\pm.000}$	$3.18 \times 10^5_{\pm4.80\times10^5}$
		TFT	$0.407_{\pm.065}$	$0.292_{\pm.089}$	$0.189_{\pm.306}$	$0.07_{\pm.031}$	$0.09_{\pm.007}$	$0.009_{\pm.007}$
		DeepAR	$0.443_{\pm.095}$	$0.308_{\pm.088}$	$0.197_{\pm.054}$	$0.07_{\pm.030}$	$0.09_{\pm.028}$	$0.004_{\pm.003}$
	LSTM	FCP (MLP)	$0.968_{\pm.022}$	$0.859_{\pm.075}$	$0.966_{\pm.022}$	$\mathbf{1.05}_{\pm.111}$	$0.950_{\pm.010}$	$\mathbf{1.82}_{\pm.287}$
		FCP (iResNet)	$0.957_{\pm.024}$	$\mathbf{0.788}_{\pm.051}$	$0.970_{\pm.010}$	$1.31_{\pm.103}$	$0.956_{\pm.016}$	$2.50_{\pm.328}$
		MultiDimSPCI	$0.957_{\pm.007}$	$0.870_{\pm.383}$	$0.960_{\pm.009}$	$1.59_{\pm.588}$	$0.952_{\pm.014}$	$14.2_{\pm7.56}$
		Local Ellipsoid	$0.957_{\pm.023}$	$0.987_{\pm.413}$	$0.948_{\pm.008}$	$1.48_{\pm.559}$	$0.928_{\pm.017}$	$3.37_{\pm.605}$
		Empirical Copula	$0.955_{\pm.005}$	$3.81_{\pm.629}$	$0.948_{\pm.010}$	$25.8_{\pm5.06}$	$0.920_{\pm.017}$	$1.22 \times 10^3_{\pm281.9}$
		Gaussian Copula	$0.953_{\pm.006}$	$3.74_{\pm.570}$	$0.952_{\pm.011}$	$26.4_{\pm4.00}$	$0.920_{\pm.017}$	$1.22 \times 10^3_{\pm281.9}$
		CopulaCPTS	$1.0_{\pm.000}$	$21.9_{\pm12.7}$	$1.0_{\pm.000}$	$330.0_{\pm219.4}$	$0.992_{\pm.002}$	$4.47 \times 10^4_{\pm4.23\times10^4}$
		TFT	$0.374_{\pm.110}$	$0.285_{\pm.106}$	$0.192_{\pm.048}$	$0.06_{\pm.022}$	$0.062_{\pm.015}$	$0.003_{\pm.002}$
		DeepAR	$0.386_{\pm.065}$	$0.266_{\pm.069}$	$0.211_{\pm.056}$	$0.06_{\pm.017}$	$0.09_{\pm.009}$	$0.003_{\pm.001}$
Solar	LOO Bootstrap	FCP (MLP)	$0.957_{\pm.007}$	$1.48_{\pm.292}$	$0.969_{\pm.003}$	$4.18_{\pm.597}$	-	-
		FCP (iResNet)	$0.952_{\pm.009}$	$\mathbf{1.42}_{\pm.166}$	$0.956_{\pm.003}$	$\mathbf{2.69}_{\pm.196}$	-	-
		MultiDimSPCI	$0.968_{\pm.005}$	$1.97_{\pm.076}$	$0.971_{\pm.003}$	$11.4_{\pm1.20}$	-	-
		Local Ellipsoid	$0.947_{\pm.004}$	$1.44_{\pm.188}$	$0.948_{\pm.005}$	$1.87_{\pm.540}$	-	-
		Empirical Copula	$0.986_{\pm.004}$	$4.47_{\pm.174}$	$0.988_{\pm.004}$	$36.5_{\pm4.03}$	-	-
		Gaussian Copula	$0.986_{\pm.004}$	$4.47_{\pm.174}$	$0.989_{\pm.003}$	$38.2_{\pm1.37}$	-	-
		CopulaCPTS	$1.0_{\pm.000}$	$67.9_{\pm12.6}$	$1.0_{\pm.000}$	$7.25 \times 10^3_{\pm1.86\times10^3}$	-	-
		TFT	$0.782_{\pm.026}$	$0.779_{\pm.056}$	$0.722_{\pm.028}$	$3.18_{\pm.415}$	-	-
		DeepAR	$0.802_{\pm.121}$	$1.03_{\pm.114}$	$0.713_{\pm.086}$	$6.73_{\pm1.09}$	-	-
	LSTM	FCP (MLP)	$0.968_{\pm.009}$	$\mathbf{1.16}_{\pm.092}$	$0.961_{\pm.008}$	$\mathbf{2.09}_{\pm.566}$	-	-
		FCP (iResNet)	$0.955_{\pm.005}$	$1.24_{\pm.076}$	$0.955_{\pm.008}$	$2.42_{\pm.276}$	-	-
		MultiDimSPCI	$0.969_{\pm.004}$	$1.31_{\pm.010}$	$0.976_{\pm.005}$	$6.46_{\pm2.51}$	-	-
		Local Ellipsoid	$0.972_{\pm.005}$	$1.27_{\pm.143}$	$0.978_{\pm.004}$	$2.43_{\pm.996}$	-	-
		Empirical Copula	$0.987_{\pm.002}$	$6.47_{\pm.103}$	$0.990_{\pm.003}$	$67.7_{\pm10.9}$	-	-
		Gaussian Copula	$0.992_{\pm.001}$	$7.11_{\pm.216}$	$0.997_{\pm.001}$	$89.9_{\pm4.69}$	-	-
		CopulaCPTS	$1.0_{\pm.000}$	$44.8_{\pm9.88}$	$1.0_{\pm.000}$	$3.34 \times 10^3_{\pm570}$	-	-
		TFT	$0.746_{\pm.081}$	$0.651_{\pm.095}$	$0.684_{\pm.063}$	$1.63_{\pm.177}$	-	-
		DeepAR	$0.839_{\pm.028}$	$1.01_{\pm.088}$	$0.715_{\pm.043}$	$3.57_{\pm.493}$	-	-

A Related Works

A.1 Conformal Prediction for Time Series

Conformal prediction has gained widespread popularity for its effectiveness in uncertainty quantification for black-box models, requiring only the exchangeability assumption [45]. However, applying CP methods to time series poses significant challenges, as time series inherently violate the exchangeability assumption due to their sequential and temporal dependencies.

Numerous studies have extended conformal prediction (CP) beyond the exchangeability assumption. A significant line of research focuses on assigning unequal weights to past non-conformity scores or leveraging their historical context, allowing more informative scores to contribute more effectively. Such works include Xu and Xie [50], Xu and Xie [47], Tibshirani et al. [43], and Lee et al. [25].

In particular, Xu and Xie [50] introduced the Sequential Predictive Conformal Inference (SPCI) framework, which incorporates correlations in non-conformity scores to construct more robust prediction intervals by sequentially adopting a quantile regression estimator. Based on this idea, several studies have employed neural networks to enhance CP for time series. For example, Lee et al. [26] utilized the Transformer [44] to capture the correlations in non-conformity scores. Auer et al. [3] proposed HopCPT, which leverages Hopfield networks to achieve a similar objective.

A.2 Conformal Prediction for Multi-dimensional Data

Conformal prediction for multi-dimensional data has been actively studied, as modern data often contain multiple variables. One of the simplest approaches involves constructing coordinate-wise prediction intervals with Bonferroni correction. For instance, Stankeviciute et al. [41] applied this idea to generate coordinate-wise prediction intervals for multi-step time series forecasting by adjusting the significance level using Bonferroni correction. A similar approach has been explored for multivariate functional data [14] and multivariate functional time series data [13].

Recent studies have explored various uncertainty sets, such as copulas and ellipsoids, to construct prediction regions for multidimensional data. For example, Messoudi et al. [30] investigated the use of copulas for constructing prediction regions, while Messoudi et al. [31] and Johnstone and Ndiaye [22] utilized ellipsoidal uncertainty sets. Sun and Yu [42] extended the application of copulas to exchangeable time series. Xu et al. [51] applied the SPCI framework to non-conformity scores defined as the radius of ellipsoidal uncertainty sets, leveraging sequential correlations of the non-conformity scores to handle multi-dimensional outcomes.

Fang et al. [17] used normalizing flow for CP with multi-dimensional outcomes of exchangeable data. They defined non-conformity scores as the distances from the origin and employed split conformal prediction to construct prediction regions. While their study shares some methodological similarities with ours, it differs in two significant aspects: they used discrete normalizing flows and focused exclusively on exchangeable data.

A.3 Conformal Prediction for Time Series and Multi-dimensional Data

Applying CP to time series and multi-dimensional data is challenging due to temporal dependencies and high-dimensional outputs. To address this, recent works have extended CP beyond exchangeability by incorporating sequential correlations through weighted or context-aware non-conformity scores [50, 47, 43, 25], including neural architectures such as Transformers [26] and Hopfield networks [3]. For multi-dimensional outcomes, approaches include coordinate-wise prediction with Bonferroni correction [41, 13], and structured uncertainty sets such as copulas [30, 42] and ellipsoids [31, 22, 51]. Fang et al. [17] further explored multi-dimensional CP using normalizing flows, though their method was limited to exchangeable settings and discrete flows.

A.4 Probabilistic Forecasting using Deep Learning

Probabilistic forecasting is a method of prediction that estimates the distribution of outcomes. Unlike typical time series forecasting, which outputs a point prediction, probabilistic forecasting can be used for uncertainty quantification since it outputs the distribution of the outcomes. With recent advances in deep learning, numerous probabilistic forecasting methods have been developed. Among these, DeepAR [37] and Temporal Fusion Transformer (TFT) [27] are widely used methods. DeepAR leverages RNNs and TFT utilizes attention mechanisms to capture the temporal dependencies for probabilistic forecasting. Rasul et al. [35] also applied conditional normalizing flows for probabilistic forecasting, similar to our approach, but they used a discrete set of normalizing flow layers [12] instead of continuous transformation.

B Problem Setup

We consider a sequence of observations $\{(x_i, y_i) : i = 1, 2, \ldots\}$, where $x_i \in \mathbb{R}^{d_x}$ represents d_x-dimensional feature, and $y_i \in \mathbb{R}^{d_y}$ represents d_y-dimensional continuous scalar outcome. We assume that we have a base predictor $\hat{f}$ that provides a point prediction $\hat{y}_i$ for y_i, given by $\hat{y}_i = \hat{f}(x_i)$. The base predictor $\hat{f}$ can be any black-box model and is not subject to any specific constraints.

Suppose that the first T examples, $\{(x_i, y_i)\}_{i=1}^{T}$, are used for training and validation. Our goal is to sequentially construct a prediction region $\hat{C}_{i-1}(z_i)$ for each new observation, beginning at time $T+1$. Here, z_i denotes the features used to construct $\hat{C}_{i-1}$. In the simplest setting, z_i consists only of x_i, but it may also include additional contextual information such as past features or outcomes.

We aim to construct prediction regions that satisfy the following *marginal coverage*:

$$\mathbb{P}\left(y_i \in \hat{C}_{i-1}(z_i)\right) \geq 1-\alpha, \quad \forall i, \tag{9}$$

and ideally the stronger *conditional coverage*:

$$\mathbb{P}\left(y_i \in \hat{C}_{i-1}(z_i) \mid z_i\right) \geq 1-\alpha, \quad \forall i, \tag{10}$$

where $\alpha \in [0,1]$ denotes a pre-specified significance level. Although trivially large prediction regions can always satisfy marginal coverage, they do not provide useful information for uncertainty quantification. Therefore, the objective is to construct *efficient* prediction regions—the regions that are as small as possible while still guaranteeing the marginal coverage [45]. Throughout this paper, we distinguish between the indices i and t to avoid confusion: the subscript i refers to the discrete time index of the sequence of observations, while the subscript t refers to continuous time in ODEs.

C Preliminary: Guided Flow

A flow is a time-dependent mapping $\psi : [0,1] \times \mathbb{R}^d \to \mathbb{R}^d$ that evolves a random variable $X_0 \in \mathbb{R}^d$ from a source distribution p to $X_t = \psi_t(X_0) \in \mathbb{R}^d$ for time $t \in [0,1]$. The flow ψ is defined by a vector field $u : [0,1] \times \mathbb{R}^d \to \mathbb{R}^d$ through the following ordinary differential equation (ODE):

$$\begin{aligned} \frac{d}{dt}\psi_t(x_0) &= u_t(\psi_t(x_0)), && \text{(flow ODE)} \\ \psi_0(x_0) &= x_0. && \text{(initial condition)} \end{aligned} \tag{11}$$

The flow ψ push-forward the source distribution p to the time-dependent probability density (i.e. probability path) $(p_t)_{0\leq t\leq 1}$ as:

$$([\psi_t]_* p)(x_t) = p(\psi_t^{-1}(x_t)) \det \left| \frac{\partial \psi_t^{-1}}{\partial x_t}(x_t) \right|, \tag{12}$$

where $*$ denotes the push-forward operator, and $x_t = \psi_t(x_0)$. By appropriately designing the vector field u_t that generates the probability path p_t to interpolate between the source distribution $p_0 = p$ and the target distribution $p_1 = q$, the resulting flow ψ transforms samples from p to q. Whether a vector field u_t generates a valid probability path p_t can be verified using the *continuity equation*.

A guided flow models the conditional distribution $q(x_1 \mid h)$ by learning a guided vector field defined by the following ODE:

$$\begin{aligned} \frac{d}{dt}\psi_{t|h}(x_0 \mid h) &= u_{t|h}\left(\psi_{t|h}(x_0 \mid h) \mid h\right), && \text{(guided flow ODE)} \\ \psi_{t=0|h}(x_0 \mid h) &= x_0, && \text{(initial condition)} \end{aligned} \tag{13}$$

where x_0 is from the source distribution, x_1 is the data from conditional distribution, and $h \in \mathbb{R}^{d_h}$ denotes the guidance. The marginal probability path for the guided flow is defined as:

$$p_{t|h}(x \mid h) = \int p_{t|x_1}(x \mid x_1)\, q(x_1 \mid h)\, dx_1, \tag{14}$$

where $p_{t|x_1}(x \mid x_1)$ is a probability path interpolating between $p_{0|x_1}(x \mid x_1) = p$ and $p_{1|x_1}(x \mid x_1) = \delta_{x_1}$, with δ_{x_1} denoting the Dirac delta distribution centered at x_1 from $q(x \mid h)$. Therefore, $p_{t|h}(x_1 \mid h)$ interpolates between the source distribution $p_{0|h}(x \mid h) = p$ and $p_{1|h}(x \mid h) = q(x_1 \mid h)$.

The corresponding guided vector field that generates $p_{t|h}(x \mid h)$ is given by:

$$u_{t|h}(x \mid h) = \int u_{t|x_1}(x \mid x_1) \frac{p_{t|x_1}(x \mid x_1) q(x_1 \mid h)}{p_{t|h}(x \mid h)}\, dx_1. \tag{15}$$

The validity of $u_t(x \mid h)$ in generating the guided probability path $p_{t|h}(x \mid h)$ can be verified using the *continuity equation* (see Proposition E.1). The resulting guided flow enables sampling from the conditional distribution $q(x_1 \mid h)$.

When using an affine probability path with a Gaussian source distribution, the interpolating probability paths remain Gaussian. This yields the Gaussian probability path:

$$p_{t|x_1}(x \mid x_1) = \mathcal{N}(x \mid \alpha_t x_1, \sigma_t^2 I_d), \tag{16}$$

where $\mathcal{N}$ denotes the Gaussian kernel and $I_d \in \mathbb{R}^{d \times d}$ denotes the identity matrix. $\alpha_t, \sigma_t : [0,1] \to [0,1]$ are interpolating scheduler, which are smooth functions satisfying $\alpha_0, \sigma_1 = 0$, $\alpha_1, \sigma_0 = 1$, and $\frac{d}{dt}\alpha_t - \frac{d}{dt}\sigma_t > 0$ for $t \in (0,1)$. Under the Gaussian probability path, the guided vector field $u_{t|h}(x \mid h)$ can be reformulated as:

$$u_{t|h}(x \mid h) = u_t(x) + b_t \nabla_x \log p_{h|t}(h \mid x), \tag{17}$$

where $u_t(x)$ is a vector field trained without the guidance, b_t is a scalar constant, and $\log p_{h|t}(h \mid x)$ is a time-dependent classifier predicting the guidance h given $x \sim p_t(x)$. A detailed derivation is provided in Proposition E.2. Early approaches [11, 40] proposed training a separate classifier for $\nabla \log p_{h|t}(h \mid x)$, and found that a guidance scale $w > 1$ to amplify the signal from the classifier is beneficial in practice:

$$\tilde{u}_{t|h}(x \mid h) = u_t(x) + w b_t \nabla_x \log p_{h|t}(h \mid x). \tag{18}$$

Using the identity $\nabla_x \log p_{t|h}(x_t \mid h) = \nabla_x \log p_t(x_t) + \nabla_x \log p_{h|t}(h \mid x_t)$, equation (18) can be equivalently rewritten as:

$$\tilde{u}_{t|h}(x \mid h) = (1-w) u_{t|h}(x) + w u_{t|h}(x \mid h). \tag{19}$$

Instead of modeling $u_t(x_t)$ and $u_t(x_t \mid h)$ separately, Ho and Salimans [20] proposed classifier-free guidance (CFG) to use a unified vector field to model both by assigning a null condition $\varnothing$ to represent the unguided vector field. In CFG, the unconditional vector field $u_t(x_t)$ is represented as $u_t(x \mid \varnothing)$, allowing equation (19) to be formulated as:

$$\tilde{u}_{t|h}(x \mid h) = (1-w) u_{t|h}(x \mid \varnothing) + w u_{t|h}(x \mid h). \tag{20}$$

D Theoretical Analysis

In this section, we present a theoretical analysis of our method, establishing an exact non-asymptotic marginal coverage and a finite-sample bound on conditional coverage. We assume that $y_i \in \mathbb{R}^{d_y}$ is generated from an unknown true function f with additive noise $\epsilon_i \in \mathbb{R}^{d_y}$ according to $y_i = f(x_i) + \epsilon_i$. We further assume that the domains of x_i and y_i are compact, which ensures that the encoder output is also compact, as formalized in Assumption D.1 and Remark D.2. Detailed proofs are presented in Appendix E.

Assumption D.1 (Compact feature and outcome domains)**.** The feature and outcome domains are compact. That is, $x_i \in \mathcal{X} \subset \mathbb{R}^{d_x}$ and $y_i \in \mathcal{Y} \subset \mathbb{R}^{d_y}$, where $\mathcal{X}$ and $\mathcal{Y}$ are compact sets.

Remark D.2. Under Assumption D.1, if the encoder $g : \mathbb{R}^{k \times d_x} \to \mathbb{R}^{d_h}$ is a continuous function mapping an input sequence $[x_{i-k:i-1}]$ of context length k to a representation $h \in \mathbb{R}^{d_h}$, then the image of the encoder $\mathcal{H} \subset \mathbb{R}^{d_h}$ is compact.

Assumption D.3 (Flow existence, uniqueness, and Lipschitz continuity)**.** The guided vector field $u_t(x \mid h)$ is continuously differentiable in x and uniformly Lipschitz continuous in x for all $t \in [0,1]$ and $h \in \mathcal{H}$. That is, there exists a constant $L_u > 0$ such that

$$\|u_t(x \mid h) - u_t(x' \mid h)\| \le L_u \|x - x'\|, \quad \forall t, h, x, x'. \tag{21}$$

Consequently, the guided flow ψ_t defined by this vector field u_t is uniformly Lipschitz continuous in x for all $t \in [0,1]$ and $h \in \mathcal{H}$. That is, there exists a constant $L_\psi > 0$ such that

$$\|\psi_t(x|h) - \psi_t(x'|h)\| \le L_\psi \|x - x'\|, \quad \forall t, h, x, x'. \tag{22}$$

Remark D.4. Assumption D.3 ensures the existence and uniqueness of solutions to the guided flow ODE. Lemma E.5 establishes that Lipschitz continuity of the vector field implies Lipschitz continuity of the flow. In practice, the vector field can be modeled using neural network architectures that satisfy this assumption, such as multi-layer perceptrons (MLP) with smooth activation functions.

D.1 Marginal Coverage

We first establish that prediction sets generated by our method achieve exact non-asymptotic marginal coverage. This result relies on a fundamental property of flows: probability mass preservation under push-forward operations. When any measurable set is transformed through the push-forward operation of a flow, its probability mass is preserved. Lemma D.5 formalizes this property and suffices to prove the exact non-asymptotic marginal coverage stated in Proposition 2.1.

Lemma D.5 (Probability mass preserving property of flows)**.** *Let $X \sim p_X$ be a continuous random variable on $\mathbb{R}^d$, and let $\psi : \mathbb{R}^d \to \mathbb{R}^d$ be a C^1 diffeomorphism. Define $Y := \psi(X)$ with density p_Y given by the push-forward of p_X under ψ. Then, for any measurable set $\mathcal{A} \subset \mathbb{R}^d$, the transformed set $\mathcal{A}' := \psi(\mathcal{A})$ satisfies:*

$$\mathbb{P}(X \in \mathcal{A}) = \mathbb{P}(Y \in \mathcal{A}') \tag{23}$$

D.2 Conditional Coverage

We next establish a finite-sample bound on conditional coverage. Let the non-conformity score based on the prediction residual be defined as $\hat{e}(y_i) = ||\psi^{-1}(\hat{\epsilon_i} \mid h_i)||$, and the non-conformity score based on the true noise be defined as $e(y_i) = ||\psi^{-1}(\epsilon_i \mid h_i)||$. For notational simplicity, we denote $\hat{e}(y_i)$ by $\hat{e}_i$ and $e(y_i)$ by e_i. The guided flow ψ is trained on $\{(x_i, y_i)\}_{i=1}^T$ until convergence and then fixed for computing e and $\hat{e}$. We define the empirical cumulative distribution function (CDF) of $\hat{e}$ and e as:

$$\hat{F}_{T+1}(u) = \frac{1}{T}\sum_{i=1}^{T} \mathbb{1}\{\hat{e}_i \le u\}, \quad \tilde{F}_{T+1}(u) = \frac{1}{T}\sum_{i=1}^{T} \mathbb{1}\{e_i \le u\}. \tag{24}$$

Since the source distribution of the flow is set to be identical across time, the marginal distribution for e_i and $\hat{e}_i$ can be considered to be identical for all i. We denote $F_e(u) = \mathbb{P}(e \le u)$ as the CDF of the true non-conformity scores. Although the marginal distribution of e_i is identical for all i, they can be dependent through h_i. Therefore, we analyze two cases: (1) when $\{e_i\}_{i=1}^T$ are i.i.d., and (2) when $\{e_i\}_{i=1}^T$ are stationary and strongly mixing. We first establish a finite-sample bound on conditional coverage under the assumption of i.i.d. non-conformity scores.

Assumption D.6 (i.i.d. non-conformity scores)**.** The true non-conformity scores $\{e_i\}_{i=1}^T$ are i.i.d.

Assumption D.7 (Bi-Lipschitz flow)**.** We assume that the guided flow $\psi_t(x \mid h)$ is bi-Lipschitz continuous in x for all $t \in [0, 1]$ and $h \in \mathcal{H}$. That is, there exist constants $L_\psi > 0$ and $L_{\psi^{-1}} > 0$ such that

$$\|\psi_t(x \mid h) - \psi_t(x' \mid h)\| \le L_\psi \|x - x'\|, \quad \forall t, h, x, x', \tag{25}$$

and

$$\|\psi_t^{-1}(x \mid h) - \psi_t^{-1}(x' \mid h)\| \le L_{\psi^{-1}} \|x - x'\|, \quad \forall t, h, x, x'. \tag{26}$$

Remark D.8. Lemma E.6 shows that bi-Lipschitz guided vector field results in bi-Lipschitz guided flow. Therefore, the vector field $u_t(x \mid h)$ can be modeled using neural network architectures that satisfy this assumption. For example, one can use invertible Residual Networks (iResNet) [5, 9] with smooth activation functions.

Assumption D.9 (Lipschitz continuous of the CDF of the true non-conformity scores)**.** Assume that $F_e(x)$ is Lipschitz continuous with Lipschitz constant $L_{T+1} > 0$, and that F_e is strictly increasing in x.

Assumption D.10 (Estimation quality)**.** Define $\Delta_i = \hat{\epsilon}_i - \epsilon_i$. There exists a sequence $\{\delta_T\}_{T \ge 1}$ such that

$$\frac{1}{T}\sum_{i=1}^{T} \|\Delta_i\|^2 \le \delta_T^2, \quad \|\Delta_{T+1}\| \le \delta_T. \tag{27}$$

Lemma D.11 (Convergence of empirical CDF of i.i.d. $\{e_i\}_{i=1}^T$)**.** *Under Assumption D.3 and D.6, for any T, there exists an event A_T with probability at least $1 - \sqrt{\frac{\log(16T)}{T}}$, such that conditioned on A_T,*

$$\sup_x \left| \tilde{F}_{T+1}(x) - F_e(x) \right| \le \sqrt{\frac{\log(16T)}{T}}. \tag{28}$$

Lemma D.12 (Norm concentration of isotropic Gaussian random vectors)**.** *Let $X_i \sim \mathcal{N}(\mathbf{0}, \gamma I_d)$ be an isotropic Gaussian random vector in $\mathbb{R}^d$, and $\|\cdot\|$ be 2-norm. Then for any $\delta \in (0,1)$, with probability at least $1-\delta$, we have:*

$$\max_{1\le i\le T} \|X_i\| \le M_T, \tag{29}$$

where $M_T = \sqrt{\gamma}\left(\sqrt{d} + \sqrt{2\log(T/\delta)}\right)$.

Lemma D.13 (Distance between the empirical CDF of $\{e_i\}_{i=1}^T$ and $\{\hat{e}_i\}_{i=1}^T$)**.** *Under Assumption D.7, D.9, and D.10, with probability $1-\delta$, $\hat{F}_{T+1}(x)$ and $\tilde{F}_{T+1}(x)$ satisfy*

$$\sup_x \left|\hat{F}_{T+1}(x) - \tilde{F}_{T+1}(x)\right| \le (2L_{T+1}+1)C_+ 2 \sup_x \left|\tilde{F}_{T+1}(x) - F_e(x)\right|, \tag{30}$$

where $C = \sqrt{M_T L_{\psi^{-1}}\delta_T + L^2_{\psi^{-1}}\delta_T^2}$.

As a result of Lemma D.11 and D.13, Theorem 2.2 establishes the finite-sample bound for conditional coverage under i.i.d. non-conformity scores.

Definition D.14. A sequence of random variables $\{X_n\}$ is said to be *strictly stationary* if for every $k \ge 1$, any integers $n_1, \dots, n_k$, and any integer h, the joint distribution of the random variables $(X_{n_1}, \dots, X_{n_k})$ is the same as the joint distribution of $(X_{n_1+h}, \dots, X_{n_k+h})$.

Definition D.15. A sequence of random variables $\{X_n\}$ is said to be *strongly mixing* (or α-*mixing*) if the mixing coefficients $\alpha(k)$ defined by

$$\alpha(k) = \sup_{n\in\mathbb{N}} \sup_{A\in\mathcal{F}_1^n,\, B\in\mathcal{F}_{n+k}^\infty} |\mathbb{P}(A\cap B) - \mathbb{P}(A)\mathbb{P}(B)| \tag{31}$$

satisfy $\alpha(k) \to 0$ as $k \to \infty$, where $\mathcal{F}_a^b$ denotes the σ-algebra generated by $\{X_a, \dots, X_b\}$.

Assumption D.16 (Strictly stationary and strongly mixing non-conformity scores)**.** Assume that the sequence $\{e_i\}_{i=1}^T$ is strictly stationary and strongly mixing, with mixing coefficients satisfying $0 < \sum_{k>0}\alpha(k) < M < \infty$.

Lemma D.17 (Convergence of empirical CDF of stationary and strongly mixing $\{e_i\}_{i=1}^T$)**.** *Under Assumption D.16, for any T, there exists an event A_T with probability at least $1 - (\frac{M(\log T)^2}{2T})^{1/3}$, such that conditioned on A_T,*

$$\sup_x |\tilde{F}_{T+1}(x) - F_e(x)| \le \frac{(\frac{M}{2})^{1/3}(\log T)^{2/3}}{T^{1/3}}. \tag{32}$$

The bounds in Theorem 2.2 and Corollary 2.3 depend on the sample size T and the estimation error δ_T. Both bounds converge to $1-\alpha$ as $T \to \infty$, provided that $\delta_T = \mathcal{O}(T^{-a})$ for some $a > 0$. Intuitively, with sufficiently large training data and an accurate base predictor $\hat{f}$, the conditional coverage is guaranteed. The condition on δ_T can be satisfied by a broad class of estimators. For example, sieve estimators based on general neural networks achieve $\delta_T = o_p(T^{-1/4})$ when f is sufficiently smooth [10]. The Lasso estimator and Dantzig selector achieve $\delta_T = o_p(T^{-1/2})$ when f is a sparse high-dimensional linear model [6].

E Proofs

Proposition E.1. *Let $u_{t|x_1}(x \mid x_1)$ be the vector field generating the probability path $p_{t|x_1}(x \mid x_1)$. Then, the vector field $u_{t|h}(x \mid h)$ is a valid vector field generating $p_{t|h}(x \mid h)$.*

Proof. Since $u_{t|x_1}(x \mid x_1)$ generates the probability path $p_{t|x_1}(x \mid x_1)$, the continuity equation holds for each x_1:

$$\frac{\partial p_{t|x_1}(x \mid x_1)}{\partial t} + \operatorname{div}\left(u_{t|x_1}(x \mid x_1) p_{t|x_1}(x \mid x_1)\right) = 0. \tag{33}$$

The time derivative of $p_{t|h}(x \mid h)$ is:

$$\begin{aligned}\frac{\partial p_{t|h}(x \mid h)}{\partial t} &= \frac{\partial}{\partial t}\int p_{t|x_1}(x \mid x_1)q(x_1 \mid h)\,dx_1 \\ &= \int \frac{\partial p_{t|x_1}(x \mid x_1)}{\partial t} q(x_1 \mid h)\,dx_1 \\ &= -\int \text{div}\left(u_{t|x_1}(x \mid x_1)p_{t|x_1}(x \mid x_1)\right) q(x_1 \mid h)\,dx_1 \\ &= -\text{div}\left(\int u_{t|x_1}(x \mid x_1)p_{t|x_1}(x \mid x_1)q(x_1 \mid h)\,dx_1\right).\end{aligned} \tag{34}$$

Since the marginal guided vector field is defined as:

$$u_{t|h}(x \mid h) := \int u_{t|x_1}(x \mid x_1)\frac{p_{t|x_1}(x \mid x_1)q(x_1 \mid h)}{p_{t|h}(x \mid h)}\,dx_1, \tag{35}$$

we can rewrite as:

$$u_{t|h}(x \mid h)p_{t|h}(x \mid h) = \int u_{t|x_1}(x \mid x_1)p_{t|x_1}(x \mid x_1)q(x_1 \mid h)\,dx_1. \tag{36}$$

Substituting equation (36) into equation (34), we have:

$$\frac{\partial p_{t|h}(x \mid h)}{\partial t} = -\text{div}\left(u_{t|h}(x \mid h)p_{t|h}(x \mid h)\right), \tag{37}$$

which is the continuity equation for $p_{t|h}(x \mid h)$ under the vector field $u_{t|h}(x \mid h)$. Therefore, $u_{t|h}(x \mid h)$ is a valid vector field generating $p_{t|h}(x \mid h)$. □

Proposition E.2. *With a given Gaussian probability path $p_{t|x_1}(x \mid x_1) = \mathcal{N}(x \mid \alpha_t x_1, \sigma_t^2 I_d)$, the guided vector field $u_{t|h}(x \mid h)$ can be reformulated as:*

$$u_{t|h}(x \mid h) = u_t(x) + b_t \nabla_x \log p_{h|t}(h \mid x). \tag{38}$$

Proof. By the definition of the guided marginal probability path:

$$p_{t|h}(x \mid h) = \int p_{t|x_1}(x \mid x_1)q(x_1 \mid h)dx_1, \tag{39}$$

where $p_{t|x_1}(x \mid x_1) = \mathcal{N}(x \mid \alpha_t x_1, \sigma_t^2 I)$. We express the score function as

$$\nabla_x \log p_{t|h}(x \mid h) = \frac{\nabla_x p_{t|h}(x \mid h)}{p_{t|h}(x \mid h)} \tag{40}$$

$$= \frac{\int \nabla_x p_{t|x_1}(x \mid x_1)q(x_1 \mid y)dx_1}{p_{t|h}(x \mid h)} \tag{41}$$

$$= \int \nabla_x \log p_{t|x_1}(x \mid x_1)\frac{p_{t|x_1}(x \mid x_1)q(x_1 \mid y)}{p_{t|h}(x \mid h)}dx_1. \tag{42}$$

Since $p_{t|x_1}(x \mid x_1) = \mathcal{N}(x \mid \alpha_t x_1, \sigma_t^2 I)$, we have:

$$u_t(x \mid x_1) = \frac{\dot{\alpha}_t}{\sigma_t}(x - \alpha_t x_1) + \dot{\alpha}_t x_1 \tag{43}$$

$$= \frac{\dot{\alpha}_t}{\sigma_t}x - \frac{\dot{\alpha}_t}{\sigma_t}\alpha_t x_1 + \dot{\alpha}_t x_1 \tag{44}$$

$$= \frac{\dot{\alpha}_t}{\sigma_t}x + (\dot{\alpha}_t - \frac{\dot{\alpha}_t}{\sigma_t}\alpha_t)x_1 \tag{45}$$

$$= \frac{\dot{\alpha}_t}{\alpha_t}x + (\dot{\alpha}_t\sigma_t - \alpha_t\dot{\sigma}_t)\frac{1}{\alpha_t\sigma_t}(x - \alpha_t x_1) \tag{46}$$

$$= \frac{\dot{\alpha}_t}{\alpha_t}x + (\dot{\alpha}_t\sigma_t - \alpha_t\dot{\sigma}_t)\frac{\sigma_t}{\alpha_t}\nabla \log p_t(x \mid x_1), \tag{47}$$

where $\dot{\alpha_t}$ denotes $\frac{d}{dt}\alpha_t$, and $\dot{\sigma_t}$ denotes $\frac{d}{dt}\sigma_t$. The last equality holds since $\nabla_x \log p_{t|x_1}(x \mid x_1) = -\frac{1}{\sigma_t^2}(x - \alpha_t x_1)$.

The guided velocity field is defined as:

$$u_{t|h}(x \mid h) = \int u_{t|x_1}(x \mid x_1) \frac{p_{t|x_1}(x \mid x_1) q(x_1 \mid h)}{p_{t|h}(x \mid h)} dx_1. \tag{48}$$

Therefore,

$$u_{t|h}(x \mid h) = a_t x + b_t \nabla_x \log p_t(x \mid h), \tag{49}$$

where $a_t = \frac{\dot{\alpha_t}}{\alpha_t}$, and $b_t = (\dot{\alpha_t}\sigma_t - \alpha_t \dot{\sigma_t})\frac{\sigma_t}{\alpha_t}$.

By using the identity $\nabla_x \log p_{t|h}(x \mid h) = \nabla_x \log p_{h|t}(h \mid x) + \nabla_x \log p_t(x)$, we have:

$$u_t(x \mid h) = a_t x + b_t \left(\nabla \log p_{h|t}(h \mid x) + \nabla \log p_t(x)\right) = u_t(x) + b_t \nabla_x \log p_{h|t}(h \mid x). \tag{50}$$

□

Proposition E.3. *The log-determinant Jacobian ODE defined in equation* (6) *is equivalent to the divergence of the guided vector field.*

Proof. The Jacobian ODE is defined as:

$$\frac{d}{dt} J_{\psi_{t|h}}(x \mid h) = \frac{\partial u_{t|h}(\psi_{t|h}(x \mid h))}{\partial \psi_{t|h}(x \mid h)} \frac{\partial \psi_{t|h}(x \mid h)}{\partial x} = \frac{\partial u_{t|h}(\psi_{t|h}(x \mid h))}{\partial \psi_{t|h}(x \mid h)} J_{\psi_{t|h}}(x \mid h), \tag{51}$$

with the initial condition:

$$J_{\psi_{t=0|h}}(x \mid h) = I. \tag{52}$$

By using Jacobi's formula,

$$\frac{d}{dt} \det J_{\psi_{t|h}}(x \mid h) = \det J_{\psi_{t|h}}(x \mid h) \cdot \text{tr}\left(J^{-1}_{\psi_{t|h}}(x \mid h) \frac{d}{dt} J_{\psi_{t|h}}(x \mid h)\right). \tag{53}$$

Substituting equation (51) into equation (53), we obtain:

$$\frac{d}{dt} \det J_{\psi_{t|h}}(x \mid h) = \det J_{\psi_{t|h}}(x \mid h) \cdot \text{tr}\left(\frac{\partial u_{t|h}(\psi_{t|h}(x \mid h))}{\partial \psi_{t|h}(x \mid h)}\right). \tag{54}$$

Therefore,

$$\frac{d}{dt} \log |\det J_{\psi_{t|h}}(x \mid h)| = \text{tr}\left(\frac{\partial u_{t|h}(\psi_{t|h}(x \mid h))}{\partial \psi_{t|h}(x \mid h)}\right). \tag{55}$$

Since the trace of the Jacobian of a vector field corresponds to its divergence, we have:

$$\text{tr}\left(\frac{\partial u_{t|h}(\psi_{t|h}(x \mid h))}{\partial \psi_{t|h}(x \mid h)}\right) = \text{div}\left(u_{t|h}(\psi_{t|h}(x \mid h))\right), \tag{56}$$

where $\text{div}(\cdot)$ denotes the divergence operator.

Therefore, the log-determinant of the Jacobian ODE is defined as:

$$\frac{d}{dt} \log |\det J_{\psi_{t|h}}(x \mid h)| = \text{div}\left(u_{t|h}(\psi_{t|h}(x \mid h))\right) \tag{57}$$

with the initial condition:

$$\log |\det J_{\psi_{t=0|h}}(x \mid h)| = 0. \tag{58}$$

□

Theorem E.4 (Closed and connected sets under a continuous map, Munkres [33])**.** *Let Z and Y be topological spaces, and let $\psi : Z \to Y$ be a continuous map. If $E \subset Z$ is closed and connected, then $\psi(E) \subset Y$ is also closed and connected.*

Lemma E.5 (Lipschitz continuous of the guided flow)**.** *Let ψ_t denote the guided flow defined by a guided vector field u_t. If the guided vector field $u_t(x \mid h)$ is Lipschitz continuous in x uniformly over $t \in [0,1]$ and $h \in \mathcal{H}$, i.e., there exists a constant $L_u > 0$ such that*

$$\|u_t(x \mid h) - u_t(x' \mid h)\| \leq L_u \|x - x'\| \quad \forall x, x', t, h, \tag{59}$$

then the guided flow $\psi_t(x \mid h)$ is Lipschitz continuous in x over $t \in [0,1]$ and $h \in \mathcal{H}$. That is, there exists a constant $L_\psi > 0$ such that

$$\|\psi_t(x \mid h) - \psi_t(x' \mid h)\| \leq L_\psi \|x - x'\| \quad \forall x, x', t, h. \tag{60}$$

Proof. Let $d(t) = \|\psi_t(x \mid h) - \psi_t(x' \mid h)\|$

Since the guided vector field is Lipschitz continuous, there exists L_u such that

$$\|u_t(x \mid h) - u_t(x' \mid h)\| \leq L_u \|x - x'\|, \quad \forall t, h, x, x'. \tag{61}$$

This is equivalent to

$$\|u_t(\psi_t(x \mid h) \mid h) - u_t(\psi_t(x' \mid h) \mid h)\| \leq L_u \|\psi_t(x \mid h) - \psi_t(x' \mid h)\|, \quad \forall t, h, x, x'. \tag{62}$$

Let $z(t) = \psi_t(x \mid h) - \psi_t(x' \mid h)$, then

$$\frac{d}{dt} d(t) = \frac{1}{\|z(t)\|} \langle z(t), \frac{d}{dt} z(t) \rangle = \langle \frac{z(t)}{\|z(t)\|}, \frac{d}{dt} z(t) \rangle \tag{63}$$

Since $\frac{d}{dt} z(t) = u_t(\psi_t(x \mid h) \mid h) - u_t(\psi_t(x' \mid h) \mid h)$, by Cauchy-Schwarz inequality,

$$|\langle \frac{z(t)}{\|z(t)\|}, \frac{d}{dt} z(t) \rangle| \leq \|u_t(\psi_t(x \mid h) \mid h) - u_t(\psi_t(x' \mid h) \mid h)\| \tag{64}$$

Therefore,

$$\frac{d}{dt} d(t) \leq \|u_t(\psi_t(x \mid h) \mid h) - u_t(\psi_t(x' \mid h) \mid h)\| \tag{65}$$

Since the guided vector field is Lipschitz continuous,

$$\frac{d}{dt} d(t) \leq L_u d(t) \tag{66}$$

Based on Gronwall's inequality [18, 19],

Assuming that $d(t) > 0$ divide both sides by $d(t)$. If $d(t) = 0$, the inequality holds.

$$\frac{1}{d(t)} \frac{d}{dt} d(t) \leq L \Rightarrow \frac{d}{dt} \log d(t) \leq L \tag{67}$$

Now integrate both sides from 0 to t:

$$\log d(t) - \log d(0) \leq Lt \Rightarrow \log \left(\frac{d(t)}{d(0)} \right) \leq Lt \Rightarrow \frac{d(t)}{d(0)} \leq e^{Lt} \Rightarrow d(t) \leq d(0) e^{Lt} \tag{68}$$

Since $d(0) = \|\psi_0(x \mid h) - \psi_0(x' \mid h)\| = \|x - x'\|$,

$$\|\psi_t(x \mid h) - \psi_t(x' \mid h)\| \leq e^{L_u t} \|x - x'\| \tag{69}$$

Therefore, we know that

$$\|\psi_t(x \mid h) - \psi_t(x' \mid h)\| \leq e^{L_u} \|x - x'\| \quad \forall x, x', t, h \tag{70}$$

□

Proof of Lemma D.5. The probability density function of $Y = \psi(X)$ is given by the change-of-variables formula:

$$p_Y(y) = p_X(\psi^{-1}(y)) \left|\det J_{\psi^{-1}}(y)\right|, \tag{71}$$

where $J_{\psi^{-1}}(y) = \frac{\partial \psi^{-1}(y)}{\partial y}$ is the Jacobian of ψ^{-1}.

The probability mass of the transformed set $\mathcal{A}' = \psi(\mathcal{A})$ is:

$$\mathbb{P}(Y \in \mathcal{A}') = \int_{\mathcal{A}'} p_Y(y)\, dy. \tag{72}$$

Using the change-of-variables $y = \psi(x)$ with $dy = |\det J_\psi(x)| dx$:

$$\int_{\mathcal{A}'} p_Y(y)\, dy = \int_{\mathcal{A}} p_Y(\psi(x)) \left|\det J_\psi(x)\right|\, dx. \tag{73}$$

Substituting from equation (71):

$$\int_{\mathcal{A}} p_Y(\psi(x)) \left|\det J_\psi(x)\right|\, dx = \int_{\mathcal{A}} p_X(x) \left|\det J_{\psi^{-1}}(\psi(x))\right| \left|\det J_\psi(x)\right|\, dx. \tag{74}$$

Since $J_{\psi^{-1}}(\psi(x)) = J_\psi(x)^{-1}$, we have $|\det J_{\psi^{-1}}(\psi(x))| \cdot |\det J_\psi(x)| = 1$. Therefore,

$$\int_{\mathcal{A}'} p_Y(y)\, dy = \int_{\mathcal{A}} p_X(x)\, dx. \tag{75}$$

□

Lemma E.6 (bi-Lipschitz guided flow)**.** *Assume that the guided vector field is bi-Lipschitz uniformly in x over $t \in [0,1]$ and $h \in \mathcal{H}$, i.e., there exists L_u and l_u such that*

$$l_u \|x - x'\| \le \|u_t(x \mid h) - u_t(x' \mid h)\| \le L_u \|x - x'\| \quad \forall t, h, x, x'. \tag{76}$$

Then the guided flow ψ is bi-Lipschitz. There exists L_ψ and l_ψ such that

$$l_\psi \|x - x'\| \le \|\psi_t(x \mid h) - \psi_t(x' \mid h)\| \le L_\psi \|x - x'\| \quad \forall t, h, x, x'. \tag{77}$$

Proof. Proof follows similarly to Lemma E.5. The upper Lipschitz bound follows from Lemma E.5. Let $z(t) = \psi_t(x \mid h) - \psi_t(x' \mid h)$ and $d(t) = \|\psi_t(x \mid h) - \psi_t(x' \mid h)\| = \|z_t\|$.

$$\frac{d}{dt}\|z(t)\|^2 = 2\langle z(t), \frac{d}{dt} z(t)\rangle \tag{78}$$

By Cauchy-Schwarz inequality,

$$\frac{d}{dt}\|z(t)\|^2 = \frac{d}{dt} d(t)^2 \ge -2\|z(t)\| \|\frac{d}{dt} z(t)\| \tag{79}$$

Since $\frac{d}{dt} z(t) = u_t(x \mid h) - u_t(x' \mid h)$ and $\|u_t(x \mid h) - u_t(x' \mid h)\| \ge l_u \|x - x'\| = l_u \|\psi_t(x \mid h) - \psi_t(x' \mid h)\|$,

we obtain

$$\frac{d}{dt} d(t)^2 \ge -2 l_u \|z(t)\|^2 = -2 l_u d(t)^2 \tag{80}$$

Using Gronwall's inequality,

$$\|\psi_t(x \mid h) - \psi_t(x' \mid h)\| \ge e^{-l_u t} \|x - x'\| \tag{81}$$

Therefore, we know that

$$\|\psi_t(x \mid h) - \psi_t(x' \mid h)\| \ge e^{-l_u} \|x - x'\| \quad \forall x, x', t, h \tag{82}$$

Combining with the upper Lipschitz bound, we get

$$e^{-l_u} \|x - x'\| \le \|\psi_t(x \mid h) - \psi_t(x' \mid h)\| \le e^{L_u} \|x - x'\| \quad \forall x, x', t, h \tag{83}$$

□

Lemma E.7. *Under Assumption D.9,* $F_e(e_{T+1}) \sim \mathrm{Unif}[0,1]$.

Proof. Since F_e is strictly increasing and continuous under Assumption D.9, the Lemma holds for $e_{T+1} \sim F_e$. □

Proof of Lemma D.11. The proof follows the proof of Lemma 1 in Xu and Xie [49]. Under the assumption that $\{e_i\}_{i=1}^{T+1}$ are i.i.d., the Dvoretzky–Kiefer–Wolfowitz (DKW) inequality [16, 24] implies:

$$\mathbb{P}\left(\sup_x \left|\tilde{F}_{T+1}(x) - F_e(x)\right| > s_T\right) \leq 2e^{-2Ts_T^2}. \tag{84}$$

Choose $s_T = \sqrt{W(16T)/(2\sqrt{T})}$, where $W(T)$ denotes the Lambert W function satisfying $W(T)e^{W(T)} = T$. Since $W(16T) \leq \log(16T)$, it follows that $s_T \leq \sqrt{\log(16T)/T}$. Define the event A_T on which $\sup_x \left|\tilde{F}_{T+1}(x) - F_e(x)\right| \leq \sqrt{\log(16T)/T}$, so that we have:

$$\sup_x \left|\tilde{F}_{T+1}(x) - F_e(x)\right| \Big| A_T \leq \sqrt{\frac{\log(16T)}{T}}, \tag{85}$$

and

$$\mathbb{P}(A_T) > 1 - \sqrt{\frac{\log(16T)}{T}}. \tag{86}$$

□

Lemma E.8 (Gaussian concentration inequality, Theorem 5.6 in Boucheron et al. [7])**.** *Let* $X \sim \mathcal{N}(0, I_d)$ *be a standard Gaussian random vector in* $\mathbb{R}^d$ *and let* $f : \mathbb{R}^d \to \mathbb{R}$ *be an* L_f*-Lipschitz continuous function. Then, for all* $t > 0$*,*

$$\mathbb{P}(f(X) \geq \mathbb{E}[f(X)] + t) \leq \exp\left(\frac{-t^2}{2{L_f}^2}\right), \tag{87}$$

Proposition E.9 (Gaussian concentration inequality with isotropic covariance)**.** *Let* $X \sim \mathcal{N}(0, \gamma I_d)$ *be an isotropic Gaussian random vector in* $\mathbb{R}^d$ *with covariance matrix* $\gamma I_d \in \mathbb{R}^d$ *for some* $\gamma > 0$ *and let* $f : \mathbb{R}^d \to \mathbb{R}$ *be an* L_f*-Lipschitz continuous function. Then, for all* $t > 0$*,*

$$\mathbb{P}(f(X) \geq \mathbb{E}[f(X)] + t) \leq \exp\left(\frac{-t^2}{2\gamma{L_f}^2}\right), \tag{88}$$

Proof. Let $X' \sim \mathcal{N}(0, I_d)$, and define $X = \sqrt{\gamma}X'$, so that $X \sim \mathcal{N}(0, \gamma I_d)$. Define the function $f_\gamma(x) := f(\sqrt{\gamma}x)$. Then f_γ is $\sqrt{\gamma}L_f$-Lipschitz. Applying Lemma E.8 to $f_\gamma(X')$, we obtain:

$$\mathbb{P}\left(f_\gamma(X') \geq \mathbb{E}[f_\gamma(X')] + t\right) \leq \exp\left(-\frac{t^2}{2\gamma L_f^2}\right). \tag{89}$$

Since $f(X) = f_\gamma(X')$,

$$\mathbb{P}(f(X) \geq \mathbb{E}[f(X)] + t) = \mathbb{P}\left(f_\gamma(X') \geq \mathbb{E}[f_\gamma(X')] + t\right) \leq \exp\left(-\frac{t^2}{2\gamma L_f^2}\right). \tag{90}$$

□

Proof of Lemma D.12. Let $X \sim \mathcal{N}(0, \gamma I_d)$ be an isotropic Gaussian random vector in $\mathbb{R}^d$ with covariance matrix $\gamma I_d \in \mathbb{R}^d$ for some $\gamma > 0$ and let $f : \mathbb{R}^d \to \mathbb{R}$ be 2-norm, i.e., $f(X) = \|X\|$.

Using Proposition E.9 and since f is 1-Lipschitz continuous, we have for all $t > 0$:

$$\mathbb{P}(\|X\| \geq \mathbb{E}[\|X\|] + t) \leq \exp\left(-\frac{t^2}{2\gamma}\right). \tag{91}$$

Using Jensen's inequality and since $X \sim \mathcal{N}(0, \gamma I_d)$,

$$\mathbb{E}[\|X\|] \leq \sqrt{\mathbb{E}[\|X\|^2]} = \sqrt{\mathbb{E}[X^\top X]} = \sqrt{\mathrm{tr}(\gamma I_d)} = \sqrt{\gamma d}. \tag{92}$$

Therefore, for any $t > 0$,

$$\mathbb{P}\left(\|X\| \geq \sqrt{\gamma d} + t\right) \leq \exp\left(-\frac{t^2}{2\gamma}\right). \tag{93}$$

By the union bound,

$$\mathbb{P}\left(\max_{1\leq i\leq T} \|X_i\| \geq \sqrt{\gamma d} + t\right) \leq \sum_{i=1}^{T} \mathbb{P}\left(\|X_i\| \geq \sqrt{\gamma d} + t\right) \leq T \cdot \exp\left(-\frac{t^2}{2\gamma}\right). \tag{94}$$

By setting $T \cdot \exp\left(-t^2/2\gamma\right) \leq \delta$, we obtain:

$$t \geq \sqrt{2\gamma \log\left(\frac{T}{\delta}\right)}. \tag{95}$$

Therefore, with probability at least $1 - \delta$,

$$\max_{1\leq i\leq T} \|X_i\| \leq \sqrt{\gamma d} + \sqrt{2\gamma \log\left(\frac{T}{\delta}\right)}. \tag{96}$$

Defining $M_T := \sqrt{\gamma}\left(\sqrt{d} + \sqrt{2\log(T/\delta)}\right)$, we conclude:

$$\max_{1\leq i\leq T} \|X_i\| \leq M_T. \tag{97}$$

□

Lemma E.10 (Bound on the sum of differences between true and estimated non-conformity scores)**.** *Under Assumption D.3, D.7, and D.10, with probability at least* $1 - \delta$*,*

$$\sum_{i=1}^{T} |\hat{e}_i - e_i| \leq 2T(M_T L_{\psi^{-1}} \delta_T + L^2_{\psi^{-1}} \delta_T^2). \tag{98}$$

Proof. Since the encoder is fixed after convergence, it generates the same h for $\hat{\epsilon}$ and ϵ. Let $\hat{s}_i = \psi^{-1}(\hat{\epsilon}_i \mid h)$ and $s_i = \psi^{-1}(\epsilon_i \mid h)$.

Using the identity for the difference of squared norms:

$$\begin{aligned} \|\hat{s}_i\| &= \|s_i + (\hat{s}_i - s_i)\|^2 \\ &= \|s_i\|^2 + 2\langle s_i, \hat{s}_i - s_i\rangle + \|\hat{s}_i - s_i\|^2, \end{aligned} \tag{99}$$

we obtain:

$$\|\hat{s}_i\|^2 - \|s_i\|^2 = 2\langle s_i, \hat{s}_i - s_i\rangle + \|\hat{s}_i - s_i\|^2 \tag{100}$$

Therefore,

$$\begin{aligned} |\hat{e}_i - e_i| &= \left|\|\hat{s}_i\|^2 - \|s_i\|^2\right| \\ &= \left|2\langle s_i, \hat{s}_i - s_i\rangle + \|\hat{s}_i - s_i\|^2\right|. \end{aligned} \tag{101}$$

By the Cauchy-Schwarz inequality,

$$|\langle s_i, \hat{s}_i - s_i \rangle| \leq \|s_i\| \cdot \|\hat{s}_i - s_i\|. \tag{102}$$

Since ψ^{-1} is Lipschitz continuous with Lipschitz constant $L_{\psi^{-1}}$, we have:

$$\|\hat{s}_i - s_i\| \leq L_{\psi^{-1}} \|\hat{\epsilon}_i - \epsilon_i\| = L_{\psi^{-1}} \|\Delta_i\|. \tag{103}$$

Substituting inequality (103) into the inner product bound in equation (102),

$$|\langle s_i, \hat{s}_i - s_i \rangle| \leq \|s_i\| \cdot \|\hat{s}_i - s_i\| \leq L_{\psi^{-1}} \|s_i\| \|\Delta_i\|. \tag{104}$$

Then, by the triangle inequality,

$$|\hat{e}_i - e_i| \leq 2L_{\psi^{-1}} \|s_i\| \|\Delta_i\| + L_{\psi^{-1}}^2 \|\Delta_i\|^2. \tag{105}$$

By Lemma D.12, we have with probability at least $1 - \delta$ that $\|s_i\| \leq M_T$ for all i, and by Assumption D.10, $\|\Delta_i\| \leq \delta_T$. Substituting these into the inequality (105),

$$|\hat{e}_i - e_i| \leq 2M_T L_{\psi^{-1}} \delta_T + L_{\psi^{-1}}^2 \delta_T^2. \tag{106}$$

Summing over all $i = 1, \ldots, T$, we conclude:

$$\sum_{i=1}^{T} |\hat{e}_i - e_i| \leq T \left(2M_T L_{\psi^{-1}} \delta_T + L_{\psi^{-1}}^2 \delta_T^2 \right). \tag{107}$$

□

Proof of Lemma D.13. By Lemma E.10, we have with probability at least $1 - \delta$,

$$\sum_{t=1}^{T} |\hat{e}_t - e_t| \leq T \left(2M_T L_{\psi^{-1}} \delta_T + L_{\psi^{-1}}^2 \delta_T^2 \right). \tag{108}$$

Let $C = \left(2M_T L_{\psi^{-1}} \delta_T + L_{\psi^{-1}}^2 \delta_T^2 \right)^{1/2}$. Then,

$$\sum_{i=1}^{T} |\hat{e}_i - e_i| \leq TC^2. \tag{109}$$

Define $S = \{t : |\hat{e}_t - e_t| \geq C\}$. Then,

$$|S| \cdot C \leq \sum_{t=1}^{T} |\hat{e}_t - e_t| \leq TC^2, \tag{110}$$

which implies $|S| \leq TC$.

We can bound the difference between the empirical CDFs of $\hat{e}_i$ and e_i as follows:

$$\begin{aligned}
|\widehat{F}_{T+1}(x) - \widetilde{F}_{T+1}(x)| &\leq \frac{1}{T}\sum_{t=1}^{T} |\mathbb{1}\{\hat{e}_t \leq x\} - \mathbb{1}\{e_t \leq x\}| \\
&\leq \frac{1}{T}\left(|S| + \sum_{t\notin S} |\mathbb{1}\{\hat{e}_t \leq x\} - \mathbb{1}\{e_t \leq x\}|\right) \\
&\overset{(i)}{\leq} \frac{1}{T}\left(|S| + \sum_{t\notin S} \mathbb{1}\{|e_t - x| \leq C\}\right) \\
&\leq \frac{1}{T}\left(|S| + \sum_{t=1}^{T} \mathbb{1}\{|e_t - x| \leq C\}\right) \\
&\leq C + \mathbb{P}(|e_{T+1} - x| \leq C) \\
&\quad + \sup_x \left|\frac{1}{T}\sum_{t=1}^{T} \mathbb{1}\{|e_t - x| \leq C\} - \mathbb{P}(|e_{T+1} - x| \leq C)\right| \\
&\overset{(ii)}{=} C + [F_e(x + C) - F_e(x - C)] \\
&\quad + \sup_x \left|\left[\widetilde{F}_{T+1}(x + C) - \widetilde{F}_{T+1}(x - C)\right] - [F_e(x + C) - F_e(x - C)]\right| \\
&\overset{(iii)}{\leq} (2L_{T+1} + 1)C + 2\sup_x |\widetilde{F}_{T+1}(x) - F_e(x)|.
\end{aligned} \tag{111}$$

Here, (i) follows from the inequality $|\mathbb{1}\{a \leq x\} - \mathbb{1}\{b \leq x\}| \leq \mathbb{1}\{|b - x| \leq |a - b|\}$ for $a, b \in \mathbb{R}$, (ii) follows from the identity $\mathbb{P}(|e_{T+1} - x| \leq C) = F_e(x + C) - F_e(x - C)$, and (iii) uses the Lipschitz continuity of $F_e(x)$.

□

Proof of Theorem 2.2. For any $\beta \in [0, \alpha]$,

$$\begin{aligned}
&\left|\mathbb{P}\left(Y_{T+1} \in \widehat{C}^{\alpha}_{T+1} \mid Z_{T+1} = z_{T+1}\right) - (1 - \alpha)\right| \\
&= \left|\mathbb{P}\left(\hat{e}_{T+1} \in \left[\widehat{F}^{-1}_{T+1}(\beta),\, \widehat{F}^{-1}_{T+1}(1 - \alpha + \beta)\right] \mid Z_{T+1} = z_{T+1}\right) - (1 - \alpha)\right| \\
&\overset{(i)}{=} \left|\mathbb{P}\left(\beta \leq \widehat{F}_{T+1}(\hat{e}_{T+1}) \leq 1 - \alpha + \beta\right) - \mathbb{P}\left(\beta \leq F_e(e_{T+1}) \leq 1 - \alpha + \beta\right)\right|.
\end{aligned} \tag{112}$$

Equality (i) follows from Lemma E.7, which states that $F_e(e_{T+1}) \sim \mathrm{Unif}[0, 1]$. This can be further bounded by:

$$\begin{aligned}
&\left|\mathbb{P}\left(\beta \leq \widehat{F}_{T+1}(\hat{e}_{T+1}) \leq 1 - \alpha + \beta\right) - \mathbb{P}\left(\beta \leq F_e(e_{T+1}) \leq 1 - \alpha + \beta\right)\right| \\
&\leq \mathbb{E}\left|\mathbb{1}\left\{\beta \leq \widehat{F}_{T+1}(\hat{e}_{T+1}) \leq 1 - \alpha + \beta\right\} - \mathbb{1}\left\{\beta \leq F_e(e_{T+1}) \leq 1 - \alpha + \beta\right\}\right| \\
&\overset{(i)}{\leq} \mathbb{E}\left(\left|\mathbb{1}\left\{\beta \leq \widehat{F}_{T+1}(\hat{e}_{T+1})\right\} - \mathbb{1}\left\{\beta \leq F_e(e_{T+1})\right\}\right|\right. \\
&\qquad \left. + \left|\mathbb{1}\left\{\widehat{F}_{T+1}(\hat{e}_{T+1}) \leq 1 - \alpha + \beta\right\} - \mathbb{1}\left\{F_e(e_{T+1}) \leq 1 - \alpha + \beta\right\}\right|\right)
\end{aligned} \tag{113}$$

Here, inequality (i) follows from the fact that for any $a, b \in \mathbb{R}$ and real values $x, y \in \mathbb{R}$,

$$|\mathbb{1}\{a \leq x \leq b\} - \mathbb{1}\{a \leq y \leq b\}| \leq |\mathbb{1}\{a \leq x\} - \mathbb{1}\{a \leq y\}| + |\mathbb{1}\{x \leq b\} - \mathbb{1}\{y \leq b\}|. \tag{114}$$

By triangle inequality,

$$
\begin{aligned}
&\mathbb{E}\left(\left|\mathbb{1}\left\{\beta \le \widehat{F}_{T+1}(\hat{e}_{T+1})\right\} - \mathbb{1}\left\{\beta \le F_e(e_{T+1})\right\}\right|\right. \\
&\left. + \left|\mathbb{1}\left\{\widehat{F}_{T+1}(\hat{e}_{T+1}) \le 1-\alpha+\beta\right\} - \mathbb{1}\left\{F_e(e_{T+1}) \le 1-\alpha+\beta\right\}\right|\right) \\
&\le \underbrace{\mathbb{E}\left(\left|\mathbb{1}\{\beta \le \widehat{F}_{T+1}(\hat{e}_{T+1})\} - \mathbb{1}\{\beta \le F_e(e_{T+1})\}\right|\right)}_{(a)} \\
&+ \underbrace{\mathbb{E}\left(\left|\mathbb{1}\left\{\widehat{F}_{T+1}(\hat{e}_{T+1}) \le 1-\alpha+\beta\right\} - \mathbb{1}\left\{F_e(e_{T+1}) \le 1-\alpha+\beta\right\}\right|\right)}_{(b)}
\end{aligned} \tag{115}
$$

For term (a), we have:

$$
\begin{aligned}
&\mathbb{E}\left(\left|\mathbb{1}\{\beta \le \widehat{F}_{T+1}(\hat{e}_{T+1})\} - \mathbb{1}\{\beta \le F_e(e_{T+1})\}\right|\right) \\
&\le \mathbb{P}\left(|F_e(e_{T+1}) - \beta| \le |\widehat{F}_{T+1}(\hat{e}_{T+1}) - F_e(e_{T+1})|\right).
\end{aligned} \tag{116}
$$

This inequality follows from the fact that for $a, b \in \mathbb{R}$, $|\mathbb{1}\{a \le x\} - \mathbb{1}\{b \le x\}| \le \mathbb{1}\{|b-x| \le |a-b|\}$, and $\mathbb{E}[\mathbb{1}\{A\}] = \mathbb{P}(A)$.

Similarly, for term (b), we have:

$$
\begin{aligned}
&\mathbb{E}\left(\left|\mathbb{1}\left\{\widehat{F}_{T+1}(\hat{e}_{T+1}) \le 1-\alpha+\beta\right\} - \mathbb{1}\left\{F_e(e_{T+1}) \le 1-\alpha+\beta\right\}\right|\right) \\
&\le \mathbb{P}\left(|F_e(e_{T+1}) - (1-\alpha+\beta)| \le \left|\widehat{F}_{T+1}(\hat{e}_{T+1}) - F_e(e_{T+1})\right|\right).
\end{aligned} \tag{117}
$$

Therefore,

$$
\begin{aligned}
&\left|\mathbb{P}\left(Y_{T+1} \in \widehat{C}^{\alpha}_{T+1} \mid Z_{T+1} = z_{T+1}\right) - (1-\alpha)\right| \\
&\le \mathbb{P}\left(|F_e(e_{T+1}) - \beta| \le |\widehat{F}_{T+1}(\hat{e}_{T+1}) - F_e(e_{T+1})|\right) \\
&\quad + \mathbb{P}\left(|F_e(e_{T+1}) - (1-\alpha+\beta)| \le |\widehat{F}_{T+1}(\hat{e}_{T+1}) - F_e(e_{T+1})|\right)
\end{aligned} \tag{118}
$$

In Lemma D.11, we defined A_T as the event on which

$$
\sup_x |\tilde{F}_{T+1}(x) - F_e(x)| \big| A_T \le \sqrt{\frac{\log(16T)}{T}},
$$

where $\mathbb{P}(A_T) > 1 - \sqrt{\frac{\log(16T)}{T}}$. Let A_T^C denote the complement of the event A_T. For any $\gamma \in [0,1]$, we have:

$$
\begin{aligned}
&\mathbb{P}\left(|F_e(e_{T+1}) - \gamma| \le |\hat{F}_{T+1}(\hat{e}_{T+1}) - F_e(e_{T+1})|\right) \\
&\le \mathbb{P}\left(|F_e(e_{T+1}) - \gamma| \le |\hat{F}_{T+1}(\hat{e}_{T+1}) - F_e(e_{T+1})| \mid A_T\right) + \mathbb{P}(A_T^C) \\
&\le \mathbb{P}\Big(|F_e(e_{T+1}) - \gamma| \le |\hat{F}_{T+1}(\hat{e}_{T+1}) - F_e(\hat{e}_{T+1})| + |F_e(\hat{e}_{T+1}) - F_e(e_{T+1})| \,\Big|\, A_T\Big) \\
&\quad + \sqrt{\frac{\log(16T)}{T}}.
\end{aligned} \tag{119}
$$

To bound the conditional probability above, we note that with probability $1-\delta$, conditioning on the event A_T,

$$
\begin{aligned}
&|\widehat{F}_{T+1}(\hat{e}_{T+1}) - F_e(e_{T+1})| + |F_e(\hat{e}_{T+1}) - F_e(e_{T+1})| \,\big|\, A_T \\
&\overset{(i)}{\leq} \sup_x |\widehat{F}_{T+1}(x) - F_e(x)| \,\big|\, A_T + L_{T+1}|\hat{e}_{T+1} - e_{T+1}| \\
&\leq \sup_x |\widehat{F}_{T+1}(x) - \widetilde{F}_{T+1}(x)| \,\big|\, A_T + \sup_x |\widetilde{F}_{T+1}(x) - F_e(x)| \,\big|\, A_T + L_{T+1}|\hat{e}_{T+1} - e_{T+1}| \\
&\overset{(ii)}{\leq} (2L_{T+1}+1)C + 3\sup_x |\widetilde{F}_{T+1}(x) - F_e(x)| \,\big|\, A_T + L_{T+1}\delta_T \\
&\overset{(iii)}{\leq} 3\sqrt{\frac{\log(16T)}{T}} + \left(L_{T+1} + \frac{1}{2}\right)(2C + \delta_T).
\end{aligned} \tag{120}
$$

Here, inequality (i) holds due to the supremum of $|\widehat{F}_{T+1}(x) - F_e(x)|$ over x and Lipschitz continuity of F_e from Assumption D.9. Inequality (ii) follows from Lemma D.13. Inequality (iii) follows from Lemma D.11.

Since $F_e(e_{T+1}) \sim \text{Unif}[0,1]$,

$$
\begin{aligned}
&\mathbb{P}\left(|F_e(e_{T+1}) - \gamma| \leq \left|\widehat{F}_{T+1}(\hat{e}_{T+1}) - F_e(\hat{e}_{T+1})\right| + |F_e(\hat{e}_{T+1}) - F_e(e_{T+1})| \,\middle|\, A_T\right) \\
&\leq 6\sqrt{\frac{\log(16T)}{T}} + 2\left(L_{T+1} + \frac{1}{2}\right)(2C + \delta_T).
\end{aligned} \tag{121}
$$

Therefore, by substituting inequality (121) to inequality (118), we obtain:

$$
\begin{aligned}
&\left|\mathbb{P}\left(Y_{T+1} \in \widehat{C}^{\alpha}_{T+1} \mid Z_{T+1} = z_{T+1}\right) - (1-\alpha)\right| \\
&\leq 12\sqrt{\frac{\log(16T)}{T}} + 4(L_{T+1} + \frac{1}{2})(2C + \delta_T).
\end{aligned} \tag{122}
$$

□

Proof of Lemma D.17. The proof follows similarly in the proof of Lemma B.11 in Xu et al. [51]. Define $v_T(x) := \sqrt{T}(\tilde{F}_{T+1}(x) - F_e(x))$. By using Proposition 7.1 in Rio et al. [36], we have:

$$
\mathbb{E}\left(\sup_x |v_T(x)|^2\right) \leq \left(1 + 4\sum_{k=0}^{T}\alpha(k)\right)\left(3 + \frac{\log T}{2\log 2}\right)^2, \tag{123}
$$

where $\alpha(k)$ denots the k-th mixing coefficient. Under Assumption D.16, we have $\sum_{k\geq 0}\alpha(k) \leq M < \infty$. Applying Markov's inequality yields:

$$
\mathbb{P}\left(\sup_x \left|\widetilde{F}_{T+1}(x) - F_e(x)\right| \geq s_T\right) \leq \frac{\mathbb{E}\left(\sup_x |v_T(x)|^2/T\right)}{s_T^2} \leq \frac{1+4M}{Ts_T^2}\left(3 + \frac{\log T}{2\log 2}\right)^2. \tag{124}
$$

By setting

$$
s_T := \left(\frac{1+4M}{T}\left(3 + \frac{\log T}{2\log 2}\right)^2\right)^{1/3} \approx \left(\frac{M(\log T)^2}{2T}\right)^{1/3}, \tag{125}
$$

we then have:

$$
\mathbb{P}\left(\sup_x \left|\widetilde{F}_{T+1}(x) - F_e(x)\right| \leq \left(\frac{M(\log T)^2}{2T}\right)^{1/3}\right) \geq 1 - \left(\frac{M(\log T)^2}{2T}\right)^{1/3}. \tag{126}
$$

Define the event A_T on which $\sup_x \left|\tilde{F}_{T+1}(x) - F_e(x)\right| \leq \left(\frac{M(\log T)^2}{2T}\right)^{1/3}$, so that we have:

$$\sup_x \left|\tilde{F}_{T+1}(x) - F_e(x)\right| \Big| A_T \leq \left(\frac{M(\log T)^2}{2T}\right)^{1/3} \tag{127}$$

and

$$\mathbb{P}(A_T) > 1 - \left(\frac{M(\log T)^2}{2T}\right)^{1/3}. \tag{128}$$

□

Proof of Corollary 2.3. Under Assumption D.16, the result follows by combining Lemma D.13 and D.17, using an argument analogous to the proof of Theorem 2.2.

□

F Experiment Setup

For notational convenience, we refer to our method as FCP, which stands for **F**low-based **C**onformal **P**rediction. We evaluated FCP using two different architectures to model the guided vector field. The first one is MLP with Softplus activation, and the second one is iResNet [5] with Softplus activation. Both architectures are smooth and continuously differentiable, satisfying the conditions needed to ensure the existence and uniqueness of the guided flow (Assumption D.3). Moreover, the iResNet architecture satisfies the bi-Lipschitz condition of the guided flow, which is required to derive the conditional coverage bound of FCP (Assumption D.7).

`dopri5` [15] at absolute and relative tolerances of 1e-5 was used to solve all ODEs in FCP. A grid search was conducted to select the optimal hyperparameters for FCP, including the number of MLP or iResNet layers in the vector field, hidden dimensions of these layers, the number of Transformer heads and layers, the Transformer model dimension, and the covariance scale γ of the source Gaussian distribution. Detailed descriptions of the hyperparameter search are provided in Appendix G.

To determine an appropriate sample size N for reducing the variance in the prediction set size estimation using quasi-Monte Carlo sampling, we computed the relative standard error of the Jacobian determinants of ψ, defined as $\text{SE}(\det J_{\psi,h})/\text{Avg}(\det J_{\psi,h})$, where $\det J_{\psi,h} = \{\det J_\psi(x_j \mid h)\}_{j=1}^N$ are the sampled Jacobian determinants conditioned on h. We selected the smallest N such that the average relative standard error across all h falls below 0.01.

Datasets. We evaluated FCP and baselines using three real-world time series datasets: wind, traffic, and solar datasets. The wind dataset contains wind speed records measured at 30 different wind farms [53]. Each wind farm location provides 768 records with 5 features at each timestamp. The traffic dataset contains traffic flow collected at 15 different traffic sensor locations [48]. Each sensor location provides 8778 observations with 5 features at each timestamp. The solar dataset considers solar radiation in Diffused Horizontal Irradiance (DHI) units at 9 different solar sensor locations [52]. Each location provides 8755 records with 5 features at each timestamp. For the wind and traffic datasets, we randomly selected $d_y \in \{2, 4, 8\}$ locations to construct five sequences of d_y-dimensional time series. For the solar dataset, we randomly selected $d_y \in \{2, 4\}$ locations to construct five sequences of d_y-dimensional time series. We did not construct sequences with $d_y = 8$ for the solar dataset due to the limited number of unique locations, which could lead to overlapping sequences across different trials of experiments.

Base predictor $\hat{f}$ is required to provide a point prediction $\hat{y}$. We used two types of base predictors for each dataset: (1) leave-one-out (LOO) bootstrap multivariate linear regression, and (2) recurrent neural network (RNN) with long short-term memory (LSTM) units [21]. For each method, we conducted five independent experiments using the five constructed sequences, across all combinations of datasets, dimensions, and base predictors. The first 80% of each sequence was used as the training set, while the remaining 20% was split equally into validation and test sets. A validation set was used for the methods requiring a calibration set. For methods that did not require validation or calibration, the validation set was merged into the training set. Note that the effective sequence length available for evaluation varies depending on the base predictor: the RNN base predictor requires the data for

training the base predictor itself, whereas the LOO bootstrap base predictor can utilize the entire sequence.

Baselines. We evaluated our method against several conformal prediction methods designed for multi-dimensional time series or i.i.d. data: MultiDimSPCI [51], conformal prediction using local ellipsoid [31], CopulaCPTS [42], and conformal prediction using empirical and Gaussian copulas [30]. We also included two probabilistic time series forecasting methods as baselines: Temporal Fusion Transformer (TFT) [27] and DeepAR [37]. Although TFT and DeepAR were originally developed for time series with univariate outcome, we adapted them to our multi-dimensional setting by constructing independent copulas using the predicted intervals for each output dimension.

Evaluation metrics. *Efficient* prediction sets are those that are as small as possible while satisfying the desired coverage. Therefore, we used two evaluation metrics: empirical coverage and the average size of the prediction sets. The empirical coverage at a target confidence level α is defined as:

$$\frac{1}{|\mathcal{D}_{\text{test}}|} \sum_{\{z_i, y_i\} \in \mathcal{D}_{\text{test}}} \mathbb{1}\left(y_i \in \hat{C}_{i-1}(z_i, \alpha)\right), \tag{129}$$

where $\mathcal{D}_{\text{test}}$ denotes the test set. The definition of prediction set size varies depending on the geometric structure of the prediction set. In FCP, the size of the prediction set is estimated using the determinant of the Jacobian of the guided flow transformation, as described in equation (5).

G Implementation Details

MultiDimSPCI. We implemented MultiDimSPCI using the source code provided by the authors [51]. The context window size was set to 50 for experiments on all real-world datasets similarly to FCP. Thu number of trees was set to 15 as suggested by the implementation by authors on Github.

Comformal prediction using copulas. We implemented the method using the source code provided by the authors [30].

Conformal prediction using local ellipsoids We implemented the method using the source code provided by the authors [31].

CopulaCPTS We implemented the method using the source code provided by the authors [42].

Temporal Fusion Transformer We implemented Temporal Fusion Transformer (TFT) [27] using `pytorch_forecasting`. A hyperparameter grid search was conducted on the training set of each dataset with $d_y = 2$ to determine the optimal configuration. We believe this hyperparameter search generalizes well to higher d_y within each dataset, since TFT makes predictions for each outcome dimension independently in our setup. Performance was observed to saturate at a model dimension of 32, with two attention heads and two layers, therefore these settings were used for all experiments. For consistency with FCP, the context window size was fixed at 50 across all experiments. We trained the models using the Adam optimizer [23] with a learning rate of 0.001, a maximum of 50 epochs, and a dropout rate of 0.1. Quantile loss with $q \in \{0.025, 0.975\}$ was used for 0.95 target coverage.

DeepAR We implemented DeepAR [37] using `pytorch_forecasting`. A hyperparameter grid search was conducted on the training set of each dataset with $d_y = 2$ to determine the optimal configuration similarly to TFT. Performance was observed to saturate at a model dimension of 32 with two layers, therefore these settings were used for all experiments. For consistency with FCP, the context window size was fixed at 50 across all experiments. We trained the models using the Adam optimizer [23] with a learning rate of 0.001, a maximum of 50 epochs, and a dropout rate of 0.1. Multivariate normal distribution loss with $q \in \{0.025, 0.975\}$ was used for 0.95 target coverage.

FCP We used multilayer perceptions (MLP) to model the guided vector field $u_{t|h} : [0, 1] \times \mathbb{R}^{d_h} \times \mathbb{R}^{d_y} \to \mathbb{R}^{d_y}$. The time variable $t \in [0, 1]$ was concatenated with the input and fed into the vector field. A hyperparameter grid search was conducted on the training set of each dataset with different d_y to determine the optimal configuration. We set the hidden dimension of the vector field identical to

the model dimension of the encoder, so that additional layer is not required between the vector field and the encoder. Table 2 shows the hyperparameter search space and Table 3 shows the optimized hyperparameter configuration. The context window size for the encoder was set to 50. We trained the model with Adam optimizer [23] with a maximum of 50 epochs for all experiments.

To determine an appropriate sample size N for the set size estimation using quasi-Monte Carlo sampling, we computed the relative standard error of the Jacobian determinants of ψ, defined as $\mathrm{SE}(\det J_{\psi,h})/\mathrm{Avg}(\det J_{\psi,h})$, where $\det J_{\psi,h} = \{\det J_\psi(x_j \mid h)\}_{j=1}^N$ are the sampled Jacobian determinants conditioned on h. We selected the smallest N such that the average relative standard error across all h falls below 0.01. We used $N = 4096$ for experiments with $d_y = 2$, $N = 8192$ for experiments with $d_y = 4$, and $N = 16384$ for experiments with $d_y = 8$.

Table 2: The optimized hyperparameter settings for FCP.

	Hyperparameter	Search Space
Vector Field	the number of layers	{ 2, 4, 6 }
	hidden dimension	{ 16, 32, 64 }
Encoder	the number of layers	{ 2, 4, 6 }
	the number of heads	{ 2, 4, 8 }
	model dimension	{ 16, 32, 64 }
	dropout	{ 0, 0.1 }
General	covariance scale γ	{ 1, 2, 4, 8 }
	learning rate	{ 0.0005, 0.0001 }
	batch size	{ 8, 16 }

Table 3: The optimized hyperparameter configuration for FCP based on the grid search.

Dataset	**Hyperparameter**	$\mathbf{d_y = 2}$	$\mathbf{d_y = 4}$	$\mathbf{d_y = 8}$
Wind	the number of layers of the vector field	4	4	4
	the number of heads of the encoder	2	2	2
	the number of layers of the encoder	4	4	4
	the hidden dimension of the vector field and encoder	32	32	32
	covariance scale γ	1	1	2
	encoder dropout	0.1	0.1	0.1
	batch size	4	4	4
	learning rate	0.0005	0.0005	0.0005
	null condition probability	0.05	0.05	0.05
Traffic	the number of layers of the vector field	4	4	4
	the number of heads of the encoder	2	2	2
	the number of layers of the encoder	4	4	4
	the hidden dimension of the vector field and encoder	32	32	32
	covariance scale γ	1	1	1
	encoder dropout	0.1	0.1	0.1
	batch size	8	8	8
	learning rate	0.0001	0.0001	0.0001
	null condition probability	0.05	0.05	0.05
Solar	the number of layers of the vector field	4	4	-
	the number of heads of the encoder	2	2	-
	the number of layers of the encoder	4	4	-
	the hidden dimension of the vector field and encoder	32	32	-
	covariance scale γ	1	1	-
	encoder dropout	0.1	0.1	-
	batch size	8	8	-
	learning rate	0.0005	0.0005	-
	null condition probability	0.05	0.05	-