Inconsistent Multivariate Time Series Forecasting

This is the appendix to the paper entitled 'Inconsistent Multivariate Time Series Forecasting', submitted to IEEE Transactions on Knowledge and Data Engineering.

APPENDIX A
ABBREVIATIONS AND NOTATIONS

The abbreviaion table and notation table are shown in Table A.1 and A.2.

APPENDIX B
MORE DETAILS OF FPPFORMER

FPPformer [1] concentrates on improving the temporal feature extraction capability of TSFT. Fig. B.1 shows the differences of FPPformer and a canonical TSFT.

The novelties of FPPformer are three folds:

1) Hierarchical Decoder: As shown in Fig. 2 of the main text, FPPformer possesses a bottom-up encoder and up-down decoder so that it extracts the input sequence from fine to coarse in its encoder and reconstructs the prediction sequence from coarse to fine in its decoder. To comply with this rule, the canonical self-attention and the cross-attention order in the decoder is reversed to first coarsely reconstruct the prediction sequence via the patch-wise cross-attention with input sequence features and then finely deduce the inner-

TABLE A.1

Abbreviations	Meanings
MTSF	Multivariate time series forecasting
CD	Channel-dependent
CI	Channel-independent
MRA	Multi-Resolution Analysis
MODWT	Maximal overlap discrete wavelet transform
MVCI	MODWT-based variable correlation identification
ICVA	Inconsistent Cross-Variable Attention
CVDA	Cross-variable data augmentation
DMD	Dynamic Mode Decomposition
TSFT	Time series forecasting Transformer
FPPformer-MD	FPPformer with MRA and DMD
DFT	Discrete Fourier transform
DWT	Discrete wavelet transform
STFT	Short time Fourier transform
EMD	Empirical mode decomposition
SVD	Singular value decomposition
MSE	Mean Square Error
MAE	Mean Absolute Error
SMAPE	Symmetric Mean Absolute Percentage Error
MED	Median absolute deviation
MAD	Mean absolute deviation

TABLE A.2

Notations	Meanings
V	The number of variables
t_1, t_2, t_3, t_1'	Timestamps
$\boldsymbol{x}_t$	The sequence value at timestamp t
$\boldsymbol{\mathcal{X}}_{t_1:t_2}, \boldsymbol{\mathcal{X}}_{t_2+1:t_3}$	Multivariate time series
L_{in}	Input sequence length
L_{pred}	Prediction sequence length
L	Sequence length
$f(\cdot)$	Forecasting model
θ	The learnable parameters
$\boldsymbol{A}(\cdot)$	Adjacency matrix
j	The level of MODWT wavelet
N_j	The j^{th} level filter width
$\boldsymbol{X}$	Arbitrary univariate sequence
$\boldsymbol{w}_j$	The j^{th} level MODWT wavelet coefficients
$\boldsymbol{v}_j$	The j^{th} level MODWT scaling coefficients
h_j	The j^{th} level MODWT wavelet filters
g_j	The j^{th} level MODWT scaling filters
$\boldsymbol{S}_j$	The j^{th} MODWT smooth
$\boldsymbol{D}_j$	The j^{th} MODWT detail
h_j°	The filters whose DFTs are the complex conjugate of DFT(h_j)
g_j°	The filters whose DFTs are the complex conjugate of DFT(g_j)
$\boldsymbol{\mathcal{X}}^+$	The Moore–Penrose pseudo inverse of $\boldsymbol{\mathcal{X}}$
m	The number of time snapshots in each set
n	The variable number per time snapshot
$\boldsymbol{\mathcal{A}}$	The best-fit linear operator in the least square sense
$\boldsymbol{\Omega}$	The eigenvalues of $\boldsymbol{\mathcal{A}}$
$\boldsymbol{\Phi}$	The eigenvector matrix of $\boldsymbol{\mathcal{A}}$
$diag(\cdot)$	Diagonal matrix transforming function
$\boldsymbol{b}$	The initial values of $\boldsymbol{\mathcal{X}}$
C	Hidden dimension size
$\boldsymbol{W}$	Weight
σ	Biweight midvariance
$\mathrm{I}(\cdot)$	The indicator function
M_j	The size of $\boldsymbol{w}_j$ excluding the boundary coefficients
$\boldsymbol{\mathcal{F}}$	Frequency scale set
F_i	Frequency scales
$\tilde{\boldsymbol{\mathcal{X}}}_{t_1:t_2}$	The input features concatenated by $\boldsymbol{\mathcal{X}}_{t_1:t_2}$ and its MODWT smooths
$\tilde{\boldsymbol{\mathcal{X}}}_{embed}$	The embedded feature map of $\tilde{\boldsymbol{\mathcal{X}}}_{t_1:t_2}$
$\tilde{\boldsymbol{\mathcal{X}}}_{patch}$	The input feature map to the i_{th} encoder stage of FPPformer-MD
p_i	The patch size of the i_{th} encoder stage of FPPformer-MD
$\top(a, b)$	The transposing operation to swap the a and b dimension
$\boldsymbol{q}, \boldsymbol{k}, \boldsymbol{v}$	Query, key and value in attention
$Attn(\boldsymbol{q}, \boldsymbol{k})$	Attention score
$\boldsymbol{Y}$	Attention output
$Mask_A(*)$	The mask matrix in $Attn_{variable}(\boldsymbol{q}, \boldsymbol{k})$
$\boldsymbol{\mathcal{Y}}_{t_2+1:t_3}$	Model prediction result
$\boldsymbol{\mathcal{X}}^D_{t_1:t_2}$	The augmented result of $\boldsymbol{\mathcal{X}}_{t_1:t_2}$

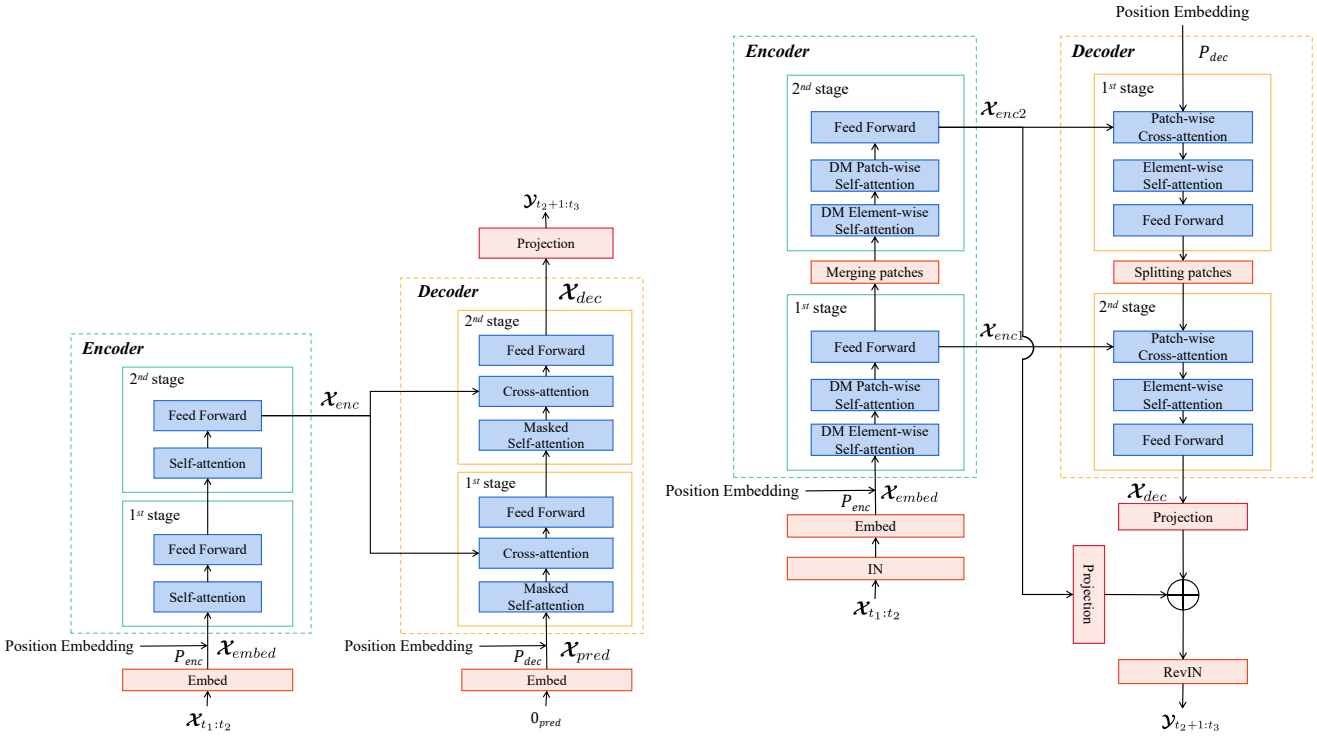

Fig. B.1. (a) A schematic of a vanilla TSFT with two-stage encoder (Green dashed box containing green solid boxes in the left) and two-stage decoder (Orange dashed box containing orange solid boxes in the right). (b) An overview of FPPformer's hierarchical architecture with two-stage encoder and two-stage decoder. Different from the vanilla one in (a), the encoder owns bottom-up structure while the decoder owns top-down structure. Note that the direction of the propagation flow in decoder is opposite to that in (a) to highlight the top-down structure. 'DM' in the stages of encoder refers to 'Diagonal-Masked'.

relationships via the element-wise self-attention.

2) Combined Element-wise and Patch-wise Attention: The patch-wise attention is prevailing in TSFTs, e.g., PatchTST [2] and Crossformer [3] since it slices the sequence into segments and the attentions are performed among these segments to enhance the model robustness. However, the patch-wise attention loses the fine-grained details of the sequences so that the patch-wise attention and element-wise attention mechanisms are combined in FPPformer to integrate their advantages as shown in Fig. B.2. To maintain the model efficiency, the element-wise attention is performed to extract the inner-relationships of each patch, which means that the elements of each patch are merely allowed to interact with the elements from the same patch. The patches in the encoder are merged into larger patches with deeper stages while the patches in the decoder are split into smaller patches with deeper stages, which means that the

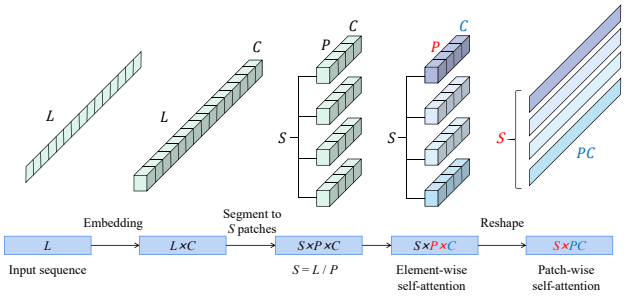

Fig. B.2. The changes in the size of a single input sequence when propagating through the first encoder stage. The batch size and the variable dimension are omitted. The red and blue letters in the last two sizes separately refer to the token dimension and its latent representation dimension. The reshaping operation is used to treat the features of all elements in a single patch as a unity for the sake of connecting element-wise self-attention and patch-wise attention. L is the sequence length, S is the number of patches, P is the patch size and C is the hidden dimensionality.

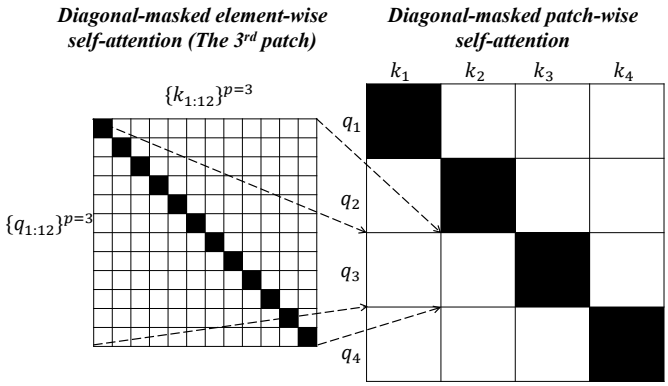

Fig. B.3. An example of query-key matching matrices in diagonal-masked element-wise self-attention and patch-wise self-attention. The total sequence length is 48 and the sequence is divided into 4 patches, each with the length of 12, in this example. The white hollow boxes denote the normal unchanged matrix elements while the black solid boxes, i.e., the matrix elements at the diagonal, denote the masked matrix elements.

features maps of the input sequence in the encoder are getting coarser while those of the prediction sequence in the decoder are getting finer. This is identical to the concept in the previous point.

3) Diagonal-Masked (DM) Self-Attention: To tackle the outliers of input sequence, FPPformer masks the diagonal of query-key matching matrix in both the element-wise and patch-wise self-attention modules of encoder as shown in Fig. B.3. Thereby, any element (patch) during the attention can merely be expressed by the values of the other elements (patches). Those normal elements or patches whose characteristics comply with the general ones are hardly affected, however the outliers are impossible to be expressed by the other normal elements (patches), hence their values are restored to approach the general level and the negative effects of them are mitigated.

APPENDIX C
INTRODUCTIONS OF BENCHMARKS AND BASELINES

C.1 *Introductions of Benchmarks*

This section provides more details of the 13 employed benchmarks in the main text:

1) ETT (Electricity Transformer Temperature) [4]: This dataset is composed of two types of subsets: 1-hour-level datasets $\{\text{ETTh}_1, \text{ETTh}_2\}$ and 15-min-level datasets $\{\text{ETTm}_1, \text{ETTm}_2\}$. They respectively contain 2-year data from two electric stations in China.

2) ECL (Electricity Consuming Load) [5]: This dataset contains the electricity consumption (Kwh) of 321 clients lasting for 2 years. We follow the usage of [4] to transform it into a 1-hour-level dataset.

3) Traffic [6]: This dataset consists of the hourly road occupation rates in San Francisco Bay area freeways for 2 years.

4) Weather [7]: This dataset is a 10-minute-level dataset which describes the values of 21 meteorological indicators in Germany during 2020.

5) Solar [8]: This dataset describes the solar power productions of 137 different photovoltaic (PV) power plants every 10 minutes in Alabama State during 2006.

6) PeMS: This traffic forecasting dataset contains four subsets PeMSD3, PeMSD4, PeMSD7, PeMSD8 collected from California Transportation Agencies (CalTrans) Performance Measurement System (PeMS). We follow [9] to aggregate the traffic flow data into 5-minute-level intervals.

7) PeMS-Bay: This traffic forecasting dataset also emanates from CalTrans PeMS. It is a 5-minute-level dataset collected by [10] and spans from Jan 1st 2017 to May 31th 2017.

C.2 *Introductions of Baselines*

This section introduces the baselines which are used in the main text for experiments.

1) ARM [11]: This Transformer-based model is named according to its three unique components: adaptive univariate effect learning (AUEL), random dropping (RD) training strategy and multi-kernel local smoothing (MKLS). The first one and the third one are involved with the temporal dimension. The second one is used to overcome the over-fitting problem of the conventional CD strategy. RD randomly pads certain number of univariate sequences to zero in arbitrary multivariate input sequence, thus the model has a chance to neglect uncorrelated variables, which is not stable and sufficiently interpretable.

2) iTransformer [12] and SAMformer [13]: These two Transformer-based CD forecasting models extract the full variable correlations via cross-variable attention modules. The difference is that iTransformer owns temporal attention modules, which are abandoned in SAMformer, before the cross-variable attention modules. They completely rely on the attention module itself to suppress the interactions of uncorrelated variables during the training phases, thus the over-fitting problem of CD forecasting approach is alleviated but not eliminated.

3) FITS [14] and SparseTSF [15]: FITS and SparseTSF are two MLP-based CI forecasting models which extract merely the temporal features of input sequences in the frequency domain. Their advantages are the lightweight architectures and the cheap computation costs. However, they excessively rely on long input sequence length to achieve competitive forecasting results. They are also incompetent in tackling non-stationary time-series in virtue of the limitations of pure Fourier spectral analysis.

4) LIFT [16]: LIFT is a plug-and-play postprocessing module that can be arranged after any CI-based or CD-based forecasting method. It is supposed to make an arbitrary lagged variable able to utilize the advance information from the corresponding leading indicators via mixing their features in the frequency domain. However, it is applied to the forecasting results of the target forecasting method, which is highly possible to be flawed attributable to the drawbacks of CI and CD approaches. Moreover, the number of lead indicator of each variable is fixed in LIFT. Such inflexibility makes LIFT utilize the cross-variable features neither sufficiently nor appropriately.

5) ModernTCN [17]: ModernTCN is a CNN-based CD forecasting method which alternately extracts the temporal features and cross-variable features of input sequence.

APPENDIX D
SUPPLEMENTARY EXPERIMENT

D.1 *Supplementary Multivariate Forecasting Results*

The full multivariate forecasting results (Input length=96) of $\{\text{ETTh}_1, \text{ETTh}_2, \text{ETTm}_1, \text{ETTm}_2, \text{PeMSD}_3, \text{PeMSD}_4, \text{PeMSD}_7, \text{PeMSD}_8\}$ are shown in Table D.1. Besides the datasets included in the previous manuscript, two additional datasets, Exchange [18] and ILI [19], which are two prevailing datasets involving finance and disease, are adopted for comparison experiment here. The numerical details for them are given in Table D.2. Following the setting of [20], the input sequence length for Exchange is 96 and the prediction sequence lengths are within $\{96, 192, 336, 720\}$. Similarly, the input sequence length for ILI is 36 and the prediction sequence lengths are within $\{24, 36, 48, 60\}$. Five additional baselines are adopted for comparison. One of them is the well-known transformer-based model PatchTST [2]. Two of them are the recent

TABLE D.1
FULL MULTIVARIATE FORECASTING RESULTS (INPUT LENGTH = 96)

Methods	Metrics	$ETTh_1$				$ETTh_2$				$ETTm_1$				$ETTm_2$			
		192	336	720	Avg.	192	336	720	Avg.	192	336	720	Avg.	192	336	720	Avg.
FPPformer-MD	MSE	***0.411***	***0.435***	***0.461***	***0.436***	***0.315***	***0.352***	***0.412***	***0.360***	***0.343***	***0.387***	***0.440***	***0.390***	***0.235***	***0.296***	***0.388***	***0.306***
	MAE	***0.413***	***0.429***	***0.459***	***0.434***	***0.358***	***0.389***	***0.432***	***0.393***	***0.369***	***0.395***	***0.430***	***0.398***	***0.295***	***0.333***	***0.385***	***0.338***
FPPformer	MSE	*0.425*	*0.470*	0.479	*0.458*	*0.372*	0.418	0.422	*0.404*	*0.362*	*0.393*	*0.448*	*0.401*	*0.243*	*0.302*	*0.398*	*0.314*
	MAE	*0.421*	*0.442*	*0.463*	*0.442*	*0.392*	0.427	*0.435*	*0.418*	*0.377*	*0.401*	*0.437*	*0.405*	*0.301*	*0.340*	*0.396*	*0.346*
ARM	MSE	0.547	0.648	0.765	0.653	0.651	0.754	0.808	0.738	0.428	0.551	0.608	0.529	0.503	0.834	1.186	0.841
	MAE	0.504	0.569	0.637	0.570	0.535	0.597	0.620	0.584	0.430	0.508	0.543	0.494	0.451	0.571	0.720	0.581
iTransformer	MSE	0.442	0.484	0.532	0.486	0.384	0.448	0.453	0.429	0.393	0.426	0.512	0.444	0.261	0.320	0.422	0.334
	MAE	0.436	0.458	0.506	0.467	0.403	0.444	0.456	0.435	0.400	0.421	0.465	0.429	0.317	0.355	0.411	0.361
ModernTCN	MSE	0.446	0.487	0.531	0.488	0.400	*0.412*	*0.416*	0.409	0.443	0.500	0.604	0.516	0.298	0.366	0.480	0.381
	MAE	0.432	0.451	0.487	0.457	0.406	*0.422*	0.436	0.422	0.435	0.470	0.524	0.476	0.339	0.379	0.442	0.387
LIFT	MSE	0.486	0.537	0.555	0.526	0.439	0.527	0.533	0.500	0.404	0.426	0.491	0.440	0.269	0.479	0.510	0.419
	MAE	0.458	0.483	0.515	0.485	0.429	0.480	0.493	0.467	0.407	0.424	0.463	0.431	0.328	0.405	0.458	0.397
FITS	MSE	0.440	0.481	*0.469*	0.463	0.377	0.417	0.421	0.405	0.393	0.430	0.489	0.437	0.247	0.308	0.408	0.321
	MAE	0.427	0.449	0.467	0.448	*0.392*	0.426	0.439	0.419	0.394	0.418	0.450	0.421	0.306	0.343	0.398	0.349
SAMformer	MSE	0.444	0.485	0.494	0.474	0.384	0.430	0.454	0.423	0.383	0.421	0.481	0.428	0.249	0.313	0.411	0.324
	MAE	0.431	0.451	0.480	0.454	0.399	0.438	0.458	0.432	0.390	0.416	0.447	0.418	0.308	0.348	0.405	0.354
SparseTSF	MSE	0.443	0.480	0.493	0.472	0.388	0.424	0.423	0.412	0.412	0.443	0.499	0.451	0.259	0.317	0.414	0.330
	MAE	0.427	0.443	0.477	0.449	0.396	0.429	0.439	0.421	0.403	0.423	0.455	0.427	0.314	0.349	0.401	0.355

Methods	Metrics	PeMSD3				PeMSD4				PeMSD7				PeMSD8			
		192	336	720	Avg.	192	336	720	Avg.	192	336	720	Avg.	192	336	720	Avg.
FPPformer-MD	MSE	***0.220***	***0.222***	***0.265***	***0.236***	***0.191***	***0.203***	***0.228***	***0.207***	***0.190***	***0.180***	***0.210***	***0.193***	***0.240***	***0.228***	***0.265***	***0.244***
	MAE	***0.313***	***0.316***	***0.346***	***0.325***	***0.304***	***0.308***	***0.331***	***0.314***	***0.284***	***0.272***	***0.300***	***0.285***	***0.326***	***0.305***	***0.338***	***0.323***
FPPformer	MSE	0.450	0.376	0.449	0.425	0.609	0.485	0.574	0.556	0.477	0.380	0.457	0.438	0.600	0.511	0.594	0.568
	MAE	0.468	0.412	0.461	0.447	0.562	0.482	0.536	0.527	0.475	0.412	0.462	0.450	0.536	0.479	0.531	0.515
ARM	MSE	0.342	0.340	0.366	0.349	0.255	0.274	0.363	0.297	0.345	0.286	0.360	0.330	0.319	0.327	0.379	0.342
	MAE	0.415	0.408	0.411	0.411	0.365	0.369	0.432	0.389	0.387	0.359	0.389	0.378	0.405	0.392	0.427	0.408
iTransformer	MSE	0.335	0.300	0.368	0.334	0.377	0.328	0.398	0.368	0.275	0.241	0.302	0.273	0.426	0.394	0.483	0.434
	MAE	0.410	0.375	0.423	0.403	0.440	0.399	0.450	0.430	0.364	0.332	0.381	0.359	0.453	0.422	0.482	0.452
ModernTCN	MSE	*0.239*	*0.235*	*0.287*	*0.254*	*0.202*	*0.205*	*0.230*	*0.212*	*0.209*	*0.189*	*0.238*	*0.212*	0.296	0.309	0.372	0.326
	MAE	*0.334*	*0.325*	*0.361*	*0.340*	*0.315*	*0.315*	*0.337*	*0.322*	*0.298*	*0.282*	*0.327*	*0.302*	0.364	0.361	0.403	0.376
LIFT	MSE	*0.239*	0.240	*0.287*	0.255	0.252	0.251	0.293	0.265	0.284	0.269	0.294	0.282	*0.267*	*0.279*	*0.301*	*0.282*
	MAE	0.341	0.337	0.374	0.351	0.356	0.350	0.387	0.364	0.350	0.335	0.352	0.346	*0.358*	*0.353*	*0.370*	*0.360*
FITS	MSE	1.118	0.823	0.954	0.965	1.172	0.871	1.008	1.017	1.201	0.865	1.019	1.028	1.241	0.947	1.100	1.096
	MAE	0.806	0.650	0.721	0.726	0.835	0.679	0.755	0.756	0.838	0.668	0.746	0.751	0.855	0.702	0.782	0.780
SAMformer	MSE	0.550	0.441	0.515	0.502	0.607	0.495	0.567	0.556	0.617	0.514	0.569	0.567	0.775	0.607	0.715	0.699
	MAE	0.541	0.459	0.510	0.503	0.568	0.490	0.538	0.532	0.573	0.491	0.542	0.535	0.614	0.537	0.604	0.585
SparseTSF	MSE	1.391	0.987	1.143	1.174	1.452	1.044	1.183	1.226	1.469	1.041	1.206	1.239	1.500	1.110	1.274	1.295
	MAE	0.934	0.720	0.800	0.818	0.965	0.759	0.825	0.850	0.965	0.746	0.830	0.847	0.973	0.771	0.847	0.864

state-of-the-art models, CATS [21] and TimeXer [22]. The last two of them are well-known anomaly detection models, Anomaly Transformer (AT) [23] and MEMTO [24], wherein their heads are replaced with the regressors for forecasting. As shown in Table D.3, FPPformer-MD still keeps the leading position when handling the forecasting scenarios of these two datasets. Specifically, under the forecasting scenarios of ILI dataset, FPPformer-MD surpasses the second best forecasting performance by achieving an MSE reduction of 16.5%, 18.5%, 19.7% and 22.4% when the prediction sequence lengths are 24, 36, 48, 60, respectively, demonstrating that FPPformer-MD also excels in short-term time series forecasting.

D.2 *Efficiency Analysis*

As a novel forecasting approach that inconsistently extracts both temporal and cross-variable features, the inconsistent MTSF approach definitely requires more space and time complexity. As shown in Fig. D.1, even compared with the existing TSFTs, FPPformer-MD is far from cheap, let alone those efficient MLP-based forecasting models, e.g., FITS [14] and SparseTSF [15]. However, we shall also notice two facts:

1) The efficiency of a MTSF model is manifested not only by its time and computation complexity, but also by its capability to fully exploit the given data. However, many researches focus on performing MTSF with long input sequence length, which veils their deficiency of incompetent in sufficiently and rationally handling variable correlations. The best evidence is that FPPformer-MD owns apparently better performances than other baselines in Table III of the main text, where the input sequence length is fixed to 96, whereas the performance

TABLE D.2
THE NUMERICAL DETAILS OF EXCHANGE AND ILI DATASETS

Datasets	Sizes	Variates	Granularity	Domains	Partitions (train/val/test)
Exchange	7588	8	24h	Finance	7/1/2
ILI	966	7	168h	Disease	7/1/2

TABLE D.3
MULTIVARIATE FORECASTING RESULTS ON EXCHANGE (INPUT LENGTH = 96) AND ILI (INPUT LENGTH = 36) DATASETS

Methods	Metrics	Exchange				ILI			
		96	192	336	720	24	36	48	60
FPPformer-MD	MSE	***0.078***	***0.165***	***0.298***	***0.802***	***2.455***	***2.253***	***2.246***	***2.147***
	MAE	***0.190***	***0.284***	***0.382***	***0.674***	***0.892***	***0.934***	***0.911***	***0.897***
FPPformer	MSE	*0.082*	*0.169*	*0.318*	*0.815*	*2.512*	*2.268*	*2.364*	*2.209*
	MAE	*0.195*	0.295	*0.407*	*0.679*	*0.934*	*0.941*	*0.921*	*0.947*
ARM	MSE	0.105	0.214	0.387	1.018	3.627	3.418	3.115	2.909
	MAE	0.253	0.362	0.484	0.806	1.418	1.330	1.268	1.190
iTransformer	MSE	0.094	0.195	0.361	1.037	3.373	3.062	2.869	2.816
	MAE	0.209	0.313	0.416	0.763	1.226	1.114	1.090	1.104
ModernTCN	MSE	0.090	0.186	0.355	1.044	3.181	2.972	2.885	2.925
	MAE	0.211	0.307	0.427	0.767	1.202	1.137	1.120	1.132
LIFT	MSE	0.093	0.196	0.352	1.211	3.331	3.171	2.912	3.362
	MAE	0.219	0.334	0.434	0.871	1.297	1.249	1.160	1.275
FITS	MSE	0.105	0.213	0.375	1.018	3.612	3.404	2.983	2.821
	MAE	0.233	0.332	0.432	0.743	1.301	1.242	1.131	1.074
SAMformer	MSE	0.097	0.206	0.369	1.010	3.339	3.209	3.024	2.976
	MAE	0.217	0.330	0.434	0.743	1.199	1.191	1.161	1.154
SparseTSF	MSE	0.105	0.222	0.382	1.087	3.679	3.467	3.126	3.044
	MAE	0.233	0.351	0.437	0.802	1.302	1.265	1.156	1.184
PatchTST	MSE	0.083	0.172	0.324	0.821	2.585	2.509	2.449	2.475
	MAE	0.199	*0.294*	0.411	0.682	1.030	1.014	1.015	1.037
CATS	MSE	0.088	0.190	0.347	0.895	2.530	2.279	2.397	2.248
	MAE	0.206	0.310	0.427	0.712	0.940	0.948	0.927	0.969
TimeXer	MSE	0.090	0.187	0.362	0.888	2.724	2.390	2.523	2.480
	MAE	0.209	0.308	0.434	0.712	1.054	0.970	0.996	0.995
AT	MSE	0.120	0.238	0.411	1.347	3.741	3.526	3.248	3.384
	MAE	0.248	0.377	0.459	0.986	1.355	1.346	1.192	1.286
MEMTO	MSE	0.126	0.234	0.405	1.259	3.728	3.514	3.352	3.396
	MAE	0.261	0.364	0.449	0.924	1.342	1.333	1.224	1.293

gaps are much smaller in Table V, where the long input sequence length is allowed. Moreover, we offer the other TSFTs more complicated architectures in Fig. D.2 or longer input sequence lengths in Fig. D.3. However, they fail to outperform the original FPPformer-MD from the perspectives of both accuracy and efficency. Therefore, albeit seemingly more expensive, FPPformer-MD is more efficient in exploiting the multivariate input sequence data and does not rely on long input sequence length to achieve shiny forecasting performance, which is a distinct advantage.

2) In practice, deep MTSF method is merely needed when meeting long-term forecasting conditions [4], which means that the accuracy is more significant if the inference time is not unacceptable. Though the inference time of FPPformer-MD may be several times longer than other baselines, it is still millisecond-level, which is minuscule compared with the forecasting duration lasting for several days or even longer.

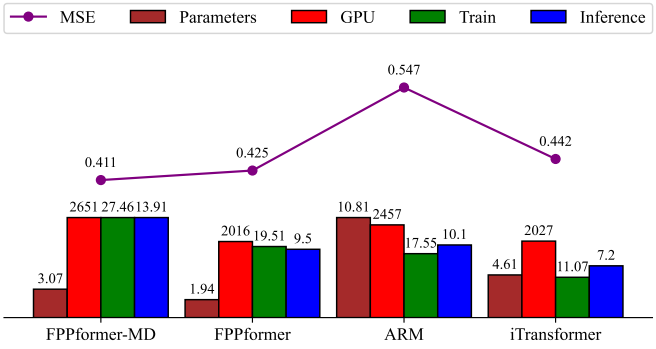

Fig. D.1. The complexity comparison of four TSFTs. The purple curve indicates the MSE results. The brown, red, blue and green bars are respectively the number of learnable parameters (MB), the GPU memory occupation (MB), the training time per epoch (s) and the inference speed (ms/instance). The batch size is 16. The dataset is ETTh_1. The input and prediction lengths are 96 and 192, respectively.

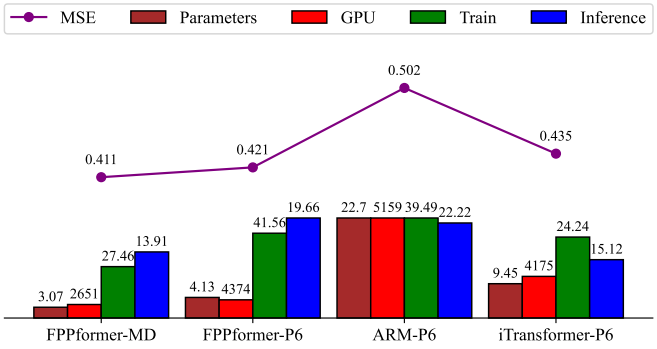

Fig. D.2. The complexity comparison of four TSFTs. The numbers of layers of FPPformer, ARM and iTransformer are doubled (from 3 to 6). '-P6' indicates that the number of layers of certain model is six.

D.3 *Comparison with Similar Methods*

To further highlight the state-of-the-art performances of the ICVA mechanism and CVDA method, we compare them with several recently proposed methods with similar functions. Four additional FPPformer-MD ablation variants are used for comparison purposes, as follows.

1) *w iA*: Imitating iTransformer [12], the cross-variable attention modules are all placed at the end of the encoder (rather than spread out at the end of each encoder stage) in FPPformer-MD. The adjacency matrix $\boldsymbol{A}$ is also removed.
2) *w LIFT*: The ICVA modules in FPPformer-MD are removed and the LIFT [16] module is attached to the prediction results of FPPformer-MD to refine each univariate prediction sequence with its leading indicators stemming from other univariate prediction sequences.
3) *w StiefelGen*: The CVDA method is replaced with StiefelGen [25], which leverages the matrix differential

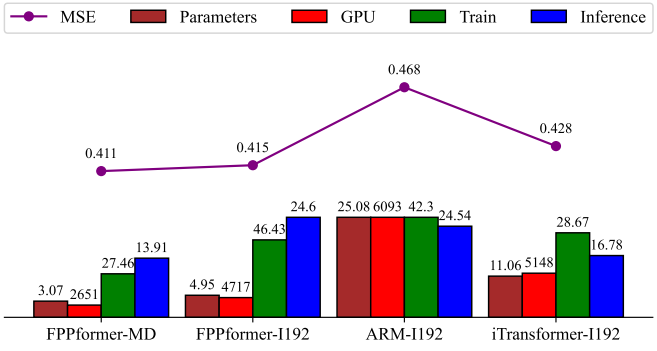

Fig. D.3. The complexity comparison of four TSFTs. The input sequence length of FPPformer-MD is 96 while those of others are 192. '-I192' indicates that the input sequence length of certain model is 192.

TABLE D.4
ABLATION STUDY ON MECHANISM VARIANTS

Methods	$ETTh_1$			ECL		
	192	336	720	192	336	720
FPPformer-MD	***0.411***	***0.435***	***0.461***	***0.144***	***0.161***	***0.197***
w iA	0.432	0.477	0.519	0.172	0.192	0.225
w LIFT	0.423	0.470	0.482	0.168	0.188	0.221
w StiefelGen	0.416	0.455	0.469	0.155	0.178	0.204
w FOC	0.415	0.451	0.468	0.151	0.169	0.201

geometry of the Stiefel manifold to generate new training instances.

4) *w FOC*: The CVDA method is replaced with FOC [26], which is a state-of-the-art mixup-based data augmentation approach.

As shown in Table D.4, replacing any proposed method with other similar existing methods yields performance degradations, thereby demonstrating that our proposed methods achieve state-of-the-art performance.

Additionally, we evaluate the effects of other wavelet transforms on FPPformer-MD, including the conventional discrete wavelet transform (DWT), discrete wavelet packet transform (DWPT) and maximal-overlap discrete wavelet packet transform (MODWPT). As shown in Table D.5, the FPPformer-MD model variants with the former two wavelet transforms suffer from apparent performance degradation due to the negative effects of circular shifts and the fact that the energy in the high frequency scales is normally minuscule, which makes more fine-grained division in the high-frequency spectrum useless for MCVI and ICVA methods. The conclusion that wavelet packet transform is not more useful than wavelet transform for MCVI and ICVA methods is once more verified by the phenomenon that the performances of FPPformer-MD models with MODWT and MODWPT are similar. This experiment demonstrates that the technique of 'maximal-overlap' is more important for the proposed methods.

D.4 *Parameter Sensitivity of CVDA*

We additionally supplement the parameter sensitivity analysis of the hyper-parameters in CVDA, including the augmentation ratio (γ), augmented segment size (s) and the error threshold to filter the low-quality augmented instances (η). The experiment benchmark is PeMSD7, which owns the largest variable number to ensure the sufficiency of page matrix rank.

TABLE D.5
ABLATION STUDY ON WAVELET TRANSFORMS

Methods	$ETTh_1$			ECL		
	192	336	720	192	336	720
FPPformer-MD	***0.411***	0.435	***0.461***	***0.144***	***0.161***	0.197
w DWT	0.445	0.482	0.534	0.174	0.198	0.235
w DWPT	0.436	0.479	0.523	0.179	0.194	0.236
w MODWPT	0.413	***0.433***	0.462	0.147	0.162	***0.195***

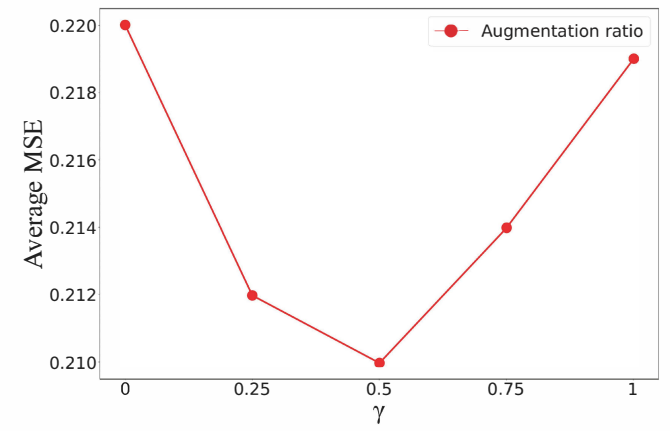

Fig. D.4. The performances of FPPformer-MD on PeMSD7 with augmentation ratio γ = 0, 0.25, 0.5, 0.75, 1

Input sequence length is 96 and prediction sequence length is 720. The results are respectively shown in Fig. D.4, D.5 and D.6.

It can be observed from Fig. D.4 and D.5 that it is not appropriate to set γ or s too large or too small. Since RevIN is applied to FPPformer-MD, the statistical characteristics of input sequences shall not be changed thoroughly, otherwise the prediction result would be forced to fit the wrong distribution, leading to enormous training loss and unstable training process. This explains why setting γ or s too big could greatly degrade the model performance. On contrary, setting γ or s too large makes the data augmentation rarely be applied so that the model generalization capability is not strengthened and the model performance also degrades.

Moreover, the apparent error increasing with bigger η in Fig. D.6 shows that it is necessary to use MAPE to filter the pathological augmented instances. This also illuminates the significance of variable correlation identification. Even with MCVI, which ensures the correlated variables are performed CVDA in most of the cases, there still exist considerable pathological augmented cases, let alone directly applying DMD to all variables for data augmentation. Indeed, as shown

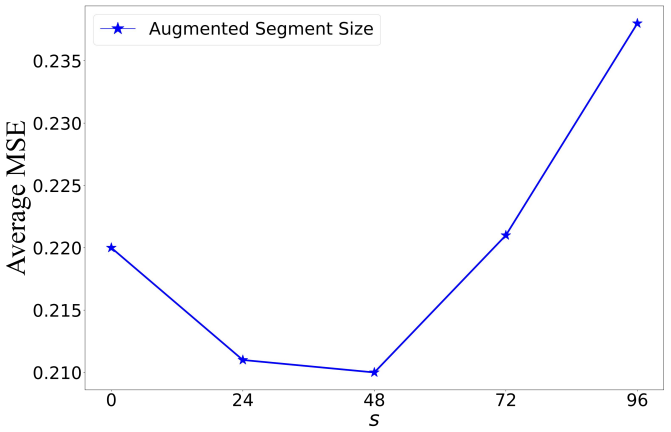

Fig. D.5. The performances of FPPformer-MD on PeMSD7 with augmented segment size s = 0, 24, 48, 72, 96.

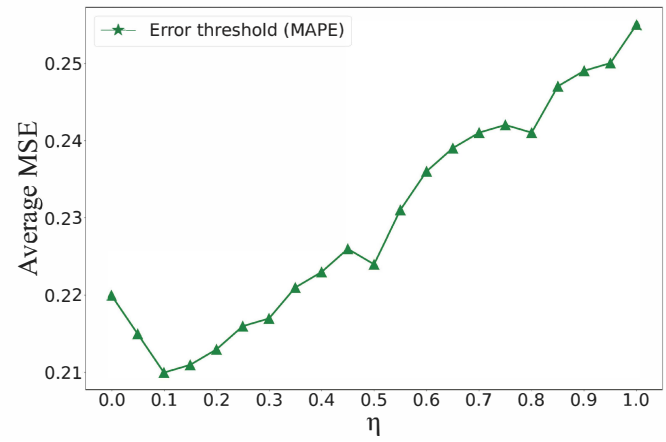

Fig. D.6. The performances of FPPformer-MD on PeMSD7 with augmentation error threshold η = 0, 0.1 , 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.

in Fig. D.7, the pathological augmented cases (MAPE $>$ 0.1) in (b), where CVDA is not applied with MCVI, are two times more than those in (a), where CVDA and MCVI are collectively applied. This phenomenon demonstrates the effectiveness of our proposed MCVI.

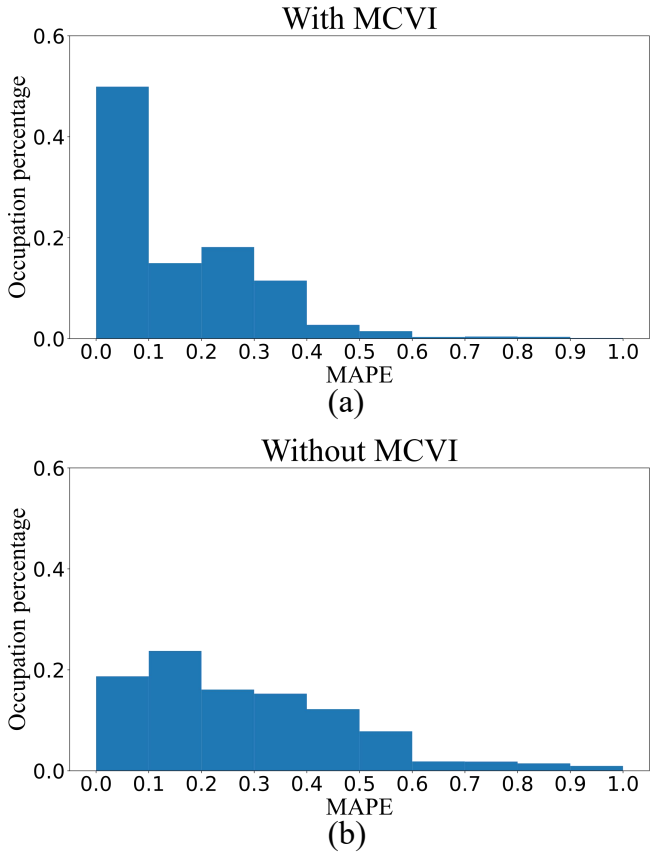

Fig. D.7. The MAPE histogram of the augmented sequences and the original input sequences. The sample space includes ten epochs of the training instances belonging to PeMSD7. (a) Applying CVDA with MCVI. (b) Applying CVDA without MCVI.

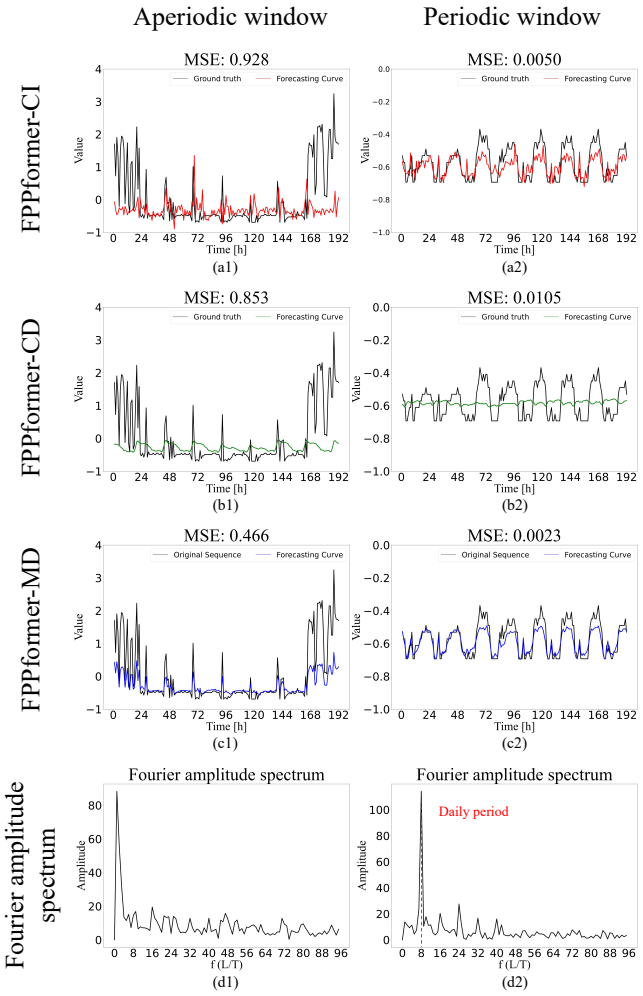

Fig. D.8. The forecasting results on ECL dataset of FPPformer-CI (a), FPPformer-CD (b) and FPPformer-MD (c). The last row are the Fourier amplitude spectra (d). The title of each sub-figure is the corresponding MSE result.

D.5 *Supplementary Case Study*

The second case study in the main text has shown that the variable correlation identification is significant for the multivariate time-series of each rolling window. However, it does not fully manifest the importance of the inconsistent forecasting approach. Indeed, the most difficult point of MTSF is not that the variables are partially related, which is already not well handled by CI and CD approaches, but that their correlations might alter heavily with time due to the nonstationarity. Using the 'MT_001' variable in ECL dataset as an example, the sub-figures in the first column of Fig. D.8 and the sub-figures in the second column of Fig. D.8 are respectively two rolling windows of 'MT_001' with different time spans. Obviously, the periodicity does not exist in the first rolling window according to its Fourier amplitude spectrum, which means that it is highly possible that it is not correlated to any other variable, whereas the second rolling window possesses apparent daily periods, rendering it correlated to plenty of other variables also with daily periods during this time span. Similar to the results in the main text, only FPPformer-MD is

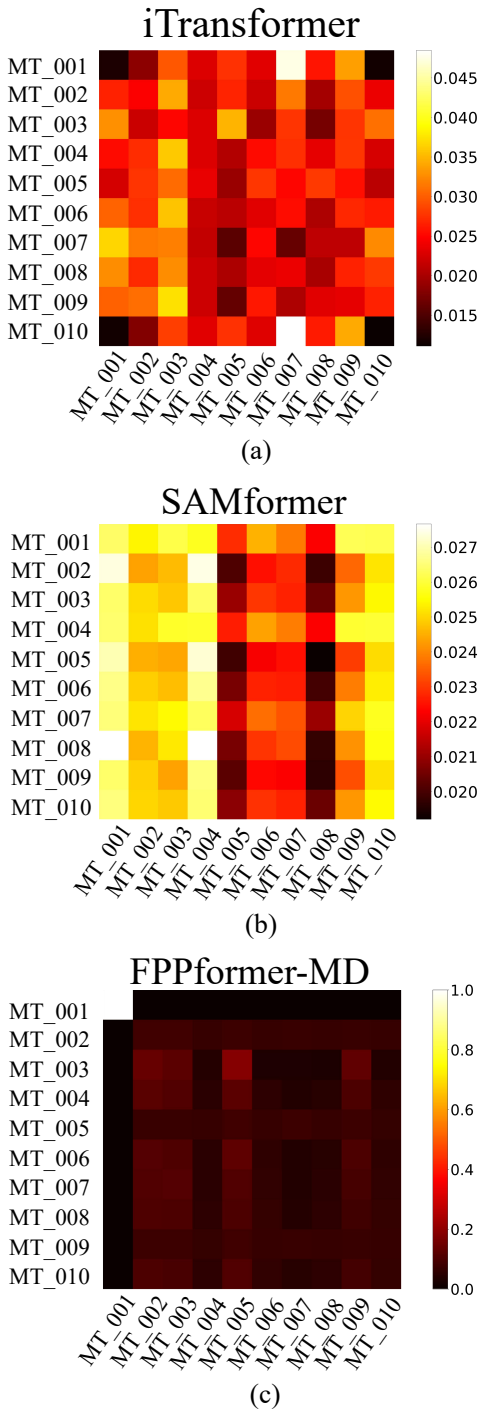

Fig. D.9. The visualization of the attention scores of cross-variable attentions. (a) iTransformer. (b) SAMformer. (c) FPPformer-MD.

able to tackle this sort of occasions.

We also visualize the attention scores to illuminate that the MCVI process is requisite to inconsistently identify the uncorrelated variables and the conventional attention module is not trustworthy to consistently suppress the interactions of uncorrelated variables. iTransformer [12] and SAMformer [13] are used for comparison since they both employ the conventional cross-variable attentions. An instance with ten variables, where the first one is unrelated to others and the rest are all correlated due to the periodicity properties, of ECL is used for experiment. As shown in Fig. D.9, neither iTransformer nor SAMformer is capable of weakening the connections of the first variable and the others, indicating that the conventional attention is not trustworthy to automatically distinguish the uncorrelated variables. In contrast, FPPformer-MD completely isolates the first variable and the others thanks to the MCVI process and ICVA mechanism, which verifies the necessity of the prior variable correlation identification.

D.6 *Imputation Experiment*

An imputation experiment is supplemented here to quantitatively illustrate the forecasting ability of FPPformer-MD when tackling the miss data or noisy data. ECL, Traffic and PeMSD7, which possess the largest variable numbers, are adopted in this experiment and the prediction length is set to 720 for widening the performance gaps of different models. The format of manually making missing data follows S4M [27] and the missing rates are within 0.03, 0.06, 0.12, 0.24. Thus, the S4M model is also used for comparison. The format of manually adding noises is similar, whereas the data elimination is replaced with adding Gaussian noises. The statistical characteristics of the Gaussian noises are proportional to those of the local data for imposing more challenges. The results in Fig. D.10 and Fig. D.11 show that FPPformer-MD not only outperforms all other forecasting models under the forecasting scenarios with missing or noisy data, but also suffers from the lowest increasing speeds of errors. These phenomena demonstrate that the inconsistent multivariate time series forecasting approach, as well as its inner methods proposed by this work, literally enhances the performance of FPPformer-MD and strengthens its robustness.

APPENDIX E
PSEUDOCODE OF AN ICVA MODULE

Algorithm 1 The Process of the ICVA Module

Input: Input feature map $\boldsymbol{\mathcal{X}}_{in}$, the adjacency matrix $\boldsymbol{A}$ from the MCVI method

$[\boldsymbol{q}, \boldsymbol{k}, \boldsymbol{v}] = [\boldsymbol{\mathcal{X}}_{in}\boldsymbol{W}_q, \boldsymbol{\mathcal{X}}_{in}\boldsymbol{W}_k, \boldsymbol{\mathcal{X}}_{in}\boldsymbol{W}_v]$
$\boldsymbol{E} = \boldsymbol{q} * \boldsymbol{k}^{\top(2,3)}$
for i = 1 to G **do**
  for j = 1 to V **do**
    for k = 1 to V **do**
      if $\boldsymbol{A}(j,k) = 1$ **then**
        $\boldsymbol{E}(i,j,k) = -\infty$
      else
        $\boldsymbol{E}(i,j,k) = \boldsymbol{E}(i,j,k)$
      end if
    end for
  end for
end for
$Attn = softmax(\frac{\boldsymbol{E}}{\sqrt{D}})$
$\boldsymbol{\mathcal{X}}_1 = Attn * \boldsymbol{v}$
$\boldsymbol{\mathcal{X}}_{out} = FFN(\boldsymbol{\mathcal{X}}_1)$

Output: Output feature map $\boldsymbol{\mathcal{X}}_{out}$

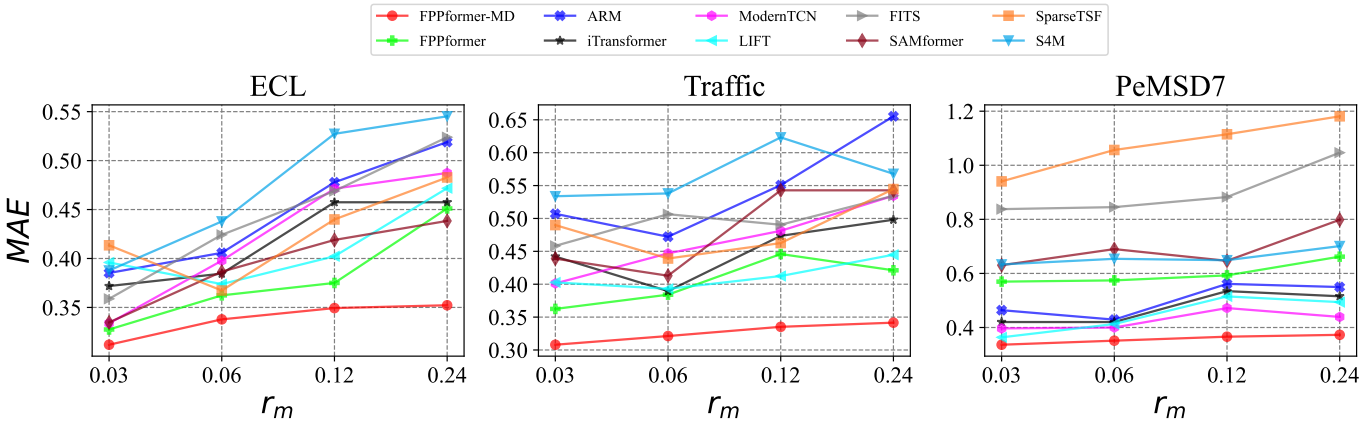

Fig. D.10. The forecasting results of FPPformer-MD and the other nine forecasting models on three datasets under the forecasting scenarios with missing data. r_m denotes the missing rate.

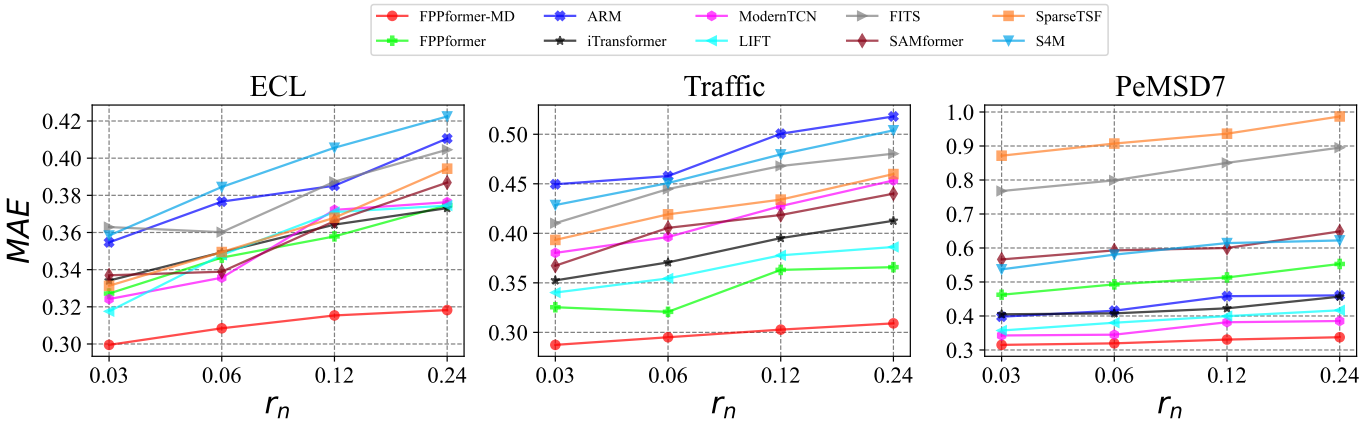

Fig. D.11. The forecasting results of FPPformer-MD and the other nine forecasting models on three datasets under the forecasting scenarios with noisy data. r_n denotes the rate of the noisy data.

where $\boldsymbol{\mathcal{X}}_{in}, \boldsymbol{\mathcal{X}}_{out} \in \mathbb{R}^{G\times V\times D}$ are the input and output feature maps of an ICVA module. G is the number of patches and V is the number of variables. $\boldsymbol{W}_q, \boldsymbol{W}_k, \boldsymbol{W}_v$ are the linear projection layers to obtain the query ($\boldsymbol{q}$), key ($\boldsymbol{k}$) and value ($\boldsymbol{v}$). is the dot-product ('$*$') of query and key. The adjacency matrix $\boldsymbol{A} \in Bool^{V\times V}$ is provided by the MCVI method and it determines whether an arbitrary pair of variables is correlated. The value one indicates an uncorrelated variable pair and the value zero indicates the opposite. $\boldsymbol{A}$ masks the corresponding elements of uncorrelated variable pairs in $\boldsymbol{E}$ with $-\infty$ so that the attention scores of them, i.e., the corresponding elements of $Attn$, are zero due to the softmax function. $FFN(\cdot)$ is the feed-forward network.

References

[1] L. Shen, Y. Wei, Y. Wang, and H. Qiu, "Take an irregular route: Enhance the decoder of time-series forecasting transformer," *IEEE Internet of Things Journal*, vol. 11, no. 8, pp. 14 344–14 356, 2024.

[2] Y. Nie, N. H. Nguyen, P. Sinthong, and J. Kalagnanam, "A time series is worth 64 words: Long-term forecasting with transformers," in *International Conference on Learning Representations*, 2023.

[3] Y. Zhang and J. Yan, "Crossformer: Transformer utilizing cross-dimension dependency for multivariate time series forecasting," in *International Conference on Learning Representations*, 2023.

[4] H. Zhou, S. Zhang, J. Peng, S. Zhang, J. Li, H. Xiong, and W. Zhang, "Informer: Beyond efficient transformer for long sequence time-series forecasting," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 12, 2021, pp. 11 106–11 115.

[5] A. Trindade, "ElectricityLoadDiagrams20112014," UCI Machine Learning Repository, 2015, DOI: https://doi.org/10.24432/C58C86.

[6] (n.d.) Caltrans pems. [Online]. Available: http://pems.dot.ca.gov/

[7] (n.d.) Max-planck-institut fuer biogeochemie - wetterdaten. [Online]. Available: https://www.bgc-jena.mpg.de/wetter/

[8] (n.d.) Solar power data for integration studies. [Online]. Available: https://www.nrel.gov/grid/solar-power-data.html

[9] W. Duan, X. He, Z. Zhou, L. Thiele, and H. Rao, "Localised adaptive spatial-temporal graph neural network," in *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2023, pp. 448–458.

[10] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting," *arXiv preprint arXiv:1707.01926*, 2017.

[11] J. Lu, X. Han, and S. Yang, "ARM: Refining multivariate forecasting with adaptive temporal-contextual learning," in *International Conference on Learning Representations*, 2024.

[12] Y. Liu, T. Hu, H. Zhang, H. Wu, S. Wang, L. Ma, and M. Long, "itransformer: Inverted transformers are effective for time series forecasting," in *International Conference on Learning Representations*, 2024.

[13] R. Ilbert, A. Odonnat, V. Feofanov, A. Virmaux, G. Paolo, T. Palpanas, and I. Redko, "Unlocking the potential of transformers in time series forecasting with sharpness-aware minimization and channel-wise attention," *arXiv preprint arXiv:2402.10198*, 2024.

[14] Z. Xu, A. Zeng, and Q. Xu, "FITS: Modeling time series with $10k$

parameters," in *International Conference on Learning Representations*, 2024.
[15] S. Lin, W. Lin, W. Wu, H. Chen, and J. Yang, "Sparsetsf: Modeling long-term time series forecasting with 1k parameters," *arXiv preprint arXiv:2405.00946*, 2024.
[16] L. Zhao and Y. Shen, "Rethinking channel dependence for multivariate time series forecasting: Learning from leading indicators," in *International Conference on Learning Representations*, 2024.
[17] L. donghao and wang xue, "ModernTCN: A modern pure convolution structure for general time series analysis," in *International Conference on Learning Representations*, 2024.
[18] G. Lai, W.-C. Chang, Y. Yang, and H. Liu, "Modeling long-and short-term temporal patterns with deep neural networks," in *The 41st international ACM SIGIR conference on research & development in information retrieval*, 2018, pp. 95–104.
[19] (n.d.) National, regional, and state level outpatient illness and viral surveillance. [Online]. Available: https://gis.cdc.gov/grasp/fluview/fluportaldashboard.html
[20] H. Wu, J. Xu, J. Wang, and M. Long, "Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting," *Advances in neural information processing systems*, vol. 34, pp. 22 419–22 430, 2021.
[21] J. Chen, J. E. Lenssen, A. Feng, W. Hu, M. Fey, L. Tassiulas, J. Leskovec, and R. Ying, "From similarity to superiority: Channel clustering for time series forecasting," in *Thirty-eighth Conference on Neural Information Processing Systems*, 2024.
[22] Y. Wang, H. Wu, J. Dong, G. Qin, H. Zhang, Y. Liu, Y. Qiu, J. Wang, and M. Long, "Timexer: Empowering transformers for time series forecasting with exogenous variables," in *The Thirty-eighth Conference on Neural Information Processing Systems*, 2024. [Online]. Available: https://openreview.net/forum?id=INAeUQ04lT
[23] J. Xu, H. Wu, J. Wang, and M. Long, "Anomaly transformer: Time series anomaly detection with association discrepancy," in *International Conference on Learning Representations*, 2022.
[24] J. Song, K. Kim, J. Oh, and S. Cho, "Memto: Memory-guided transformer for multivariate time series anomaly detection," *Advances in Neural Information Processing Systems*, vol. 36, pp. 57 947–57 963, 2023.
[25] P. Cheema and M. Sugiyama, "Stiefelgen: A simple, model agnostic approach for time series data augmentation over riemannian manifolds," *arXiv preprint arXiv:2402.19287*, 2024.
[26] B. U. Demirel and C. Holz, "Finding order in chaos: A novel data augmentation method for time series in contrastive learning," *Advances in Neural Information Processing Systems*, vol. 36, 2024.
[27] P. Jing, M. Yang, Q. Zhang, and X. Li, "S4m: S4 for multivariate time series forecasting with missing values," in *The Thirteenth International Conference on Learning Representations*.