Applying Cross-Permutation-Based Quad-Hybrid Feature Selection Algorithm on Transient Univariates to Select Optimal Features for Transient Analysis

SEYED ALIREZA BASHIRI MOSAVI
Department of Electrical and Computer Engineering, Buein Zahra Technical University, Buein Zahra, Qazvin 3451866391, Iran
e-mail: abashirimosavi@bzte.ac.ir

ABSTRACT Neglect feature selection matter for high-dimensional transient data obtained from phasor measurement units (PMUs) negatively affect the inconsistent-linked indices, namely data labeling time (DLT) and data labeling accuracy (DLA) in the transient analysis (TA). A reasonable trade-off between DLT and DLA or a win-win solution (low DLT and high DLA) necessitates feature-based mining on transient multivariate excursions (TMEs) via designing the comprehensive feature selection scheme (FSS). Hence, to achieve high-performance TA, we offer the cross-permutation-based quad-hybrid FSS (CPQHFSS) to select optimal features from TMEs. The CPQHFSS consists of four filter-wrapper blocks (FWBs) in the form of twin two-FWBs mounted on two-mechanism of the incremental wrapper, namely incremental wrapper subset selection (IWSS) and IWSS with replacement (IWSSr). The $\text{IWSS}^{2\text{FWBs}}$ and $\text{IWSSr}^{2\text{FWBs}}$ contain filter-fixed and wrapper-varied approaches ($\text{F}^f \text{W}^v$) that first block-specific $\text{F}^f \text{W}^v$ of $\text{IWSS}^{2\text{FWBs}}$ and $\text{IWSSr}^{2\text{FWBs}}$ includes relevancy ratio-support vector machine (RR-SVM) and second block-specific $\text{F}^f \text{W}^v$ of $\text{IWSS}^{2\text{FWBs}}$ and $\text{IWSSr}^{2\text{FWBs}}$ accompanied by relevancy ratio-twin support vector machine (RR-TWSVM). Generally, $^{\text{RR}}\text{IWSS}^{\text{SVM}}$ and $^{\text{RR}}\text{IWSS}^{\text{TWSVM}}$ is in $\text{IWSS}^{2\text{FWBs}}$, and $^{\text{RR}}\text{IWSSr}^{\text{SVM}}$ and $^{\text{RR}}\text{IWSSr}^{\text{TWSVM}}$ is in $\text{IWSSr}^{2\text{FWBs}}$. Besides direct relations in two-$\text{F}^f \text{W}^v$Bs per incremental wrapper mechanism, by plugging different kernels into the hyperplane-based wrapper, all possible cross-permutations of hybrid FSS are applied on transient data to extract the optimal transient features (OTFs). Finally, the evaluation of the effectiveness of the CPQHFSS-based OTFs in TA is conducted based on the cross-validation technique. The obtained results show that the proposed framework has a DLA of 98.87 % and a DLT of 152.525 milliseconds for TA.

INDEX TERMS Hybrid feature selection algorithm, optimal transient features, transient analysis.

ACRONYMS

PMUs	Phasor Measurement Units.
TA	Transient Analysis.
DLT	Data Labeling Time.
DLA	Data Labeling Accuracy.
TMEs	Transient Multivariate Excursions.
FSS	Feature Selection Scheme.
CPQHFSS	Cross-Permutation-based Quad-Hybrid FSS.
FWBs	Filter-Wrapper Blocks.
HBs	Hybrid Blocks.
IWSS	Incremental Wrapper Subset Selection.
IWSSr	IWSS with Replacement.
RR	Relevancy Ratio.
SU	Symmetric Uncertainty.
SVM	Support Vector Machine.
TWSVM	Twin Support Vector Machine.
OTFs	Optimal Transient Features.
IT	Information Technology.
DM	Data Mining.
ML	Machine Learning.
HDD	High-Dimensional Data.
TTP	Training-Testing Procedure.
A&T	Accuracy and Time.
OFs	Optimal Features.
MDFs	Most Discriminative Features.

TSA	Transient Stability Assessment.
TSP	Transient Stability Prediction.
LDOTFs	Low Dimensional Optimal Transient Features.
mRMR	Minimum-Redundancy and Maximum-Relevance.
FCBF	Fast Correlation-based Filter.
DSA	Dynamic Security Assessment.
MI	Mutual Information.
PCC	Pearson Correlation Coefficient.
NMI	Normalized Mutual Information.
SRFS	Strongly Relevant Feature Subset.
WRFS	Weakly Relevant Features Subset.
TFWM	Trajectory-based Filter-Wrapper Method.
PFWM	Point-based Filter-Wrapper Method.
FICA	Fuzzy Imperialist Competitive Algorithm.
MIE	Mutual Information-Entropy.
MHFSS	Multifaceted Hybrid FSS.
IWMs	Incremental Wrapper Mechanisms.
CL	Classification Learner.
CFS	Candidate Features Subset.
GEPSVM	Generalized Proximal Eigenvalue SVM.
KKT	Karush-Kuhn-Tucker.
LinKer	Linear Kernel.
PolKer	Polynomial Kernel.
^{S}GRBF	Standard Gaussian Radial Basis Function.
DTW	Dynamic Time Warping.
REDK	Recursive Edit Distance Kernel.
PFs	Point features.
pOTFs	Preliminary Optimal Transient Features.
TDGW	Transient Dataset Generation Workflow.
OCBF	Output Channels of Basic Feature.
API	Application Program Interface.
PSS/E	Power System Simulator for Engineering.
CONL	Convert Load.
VOLT	Bus Voltages.
VANGLE	Voltage Phase Angle.
PELEC	Machine Active Power.
QELEC	Machine Reactive Power.
QLOAD	Reactive Power Consumption.
TMTD	Transient Multivariate Trajectory Dataset.
Acc	Accuracy.
TPR	True Positive Rate (Sensitivity).
TNR	True Negative Rate (Specificity).
VUFSSs	Vertically Unilateral FSSs.
PITHS	Partial-Injective Trilateral Hybrid FSS.
BMHFSS	Bi-Mode Hybrid FSS.
OT	Observed Time.
NETS-NYPS	New England Test System-New York Power System

NOTATION

f_i	i^{th} feature.
f_{hi}	Feature with i^{th}-highest RR.
$^{Acc}(f_{hi})$	Learning model accuracy based on f_{hi}.

Incr: #i	i^{th} increments in IWSS/ IWSSr tree.
$CL^{train}(f_i)$	Training procedure of classification learner based on f_i.
param	Recording learning parameters.
$CL^{test}(f_i, param)$	Testing procedure of model based on param and f_i.
HB$^{1-2}$	First-second hybrid blocks of IWMs.
S$_i$	i^{th} contingency sample in TMEs.
pf_i	i^{th} point feature.
TU^i	i^{th} transient univariate in multivariate data.
$f_i^{TU^k}$	i^{th} features of kth transient univariate.
$H(X)$	Entropy of X; X:$\{f_i^{TU^k}$, target class$\}$.
$H(f_i^{TU^k}\|C)$	Entropy of $f_i^{TU^k}$ when class is given.
O ()	Stands for worst-case complexity.
^{pfs}TU^i	Point features (observed cycles) of TU^i.
^{Spfs}TU^i	Sorted point features of TU^{i}based on SU.
^{p}OTFsTU^i	Preliminary OTFs of TU^i.
fOTFs$^{TU\#}$	Set of final OTFs per TU.
Max	Maximum function.
Min	Minimum function.
Var	Variance function.
F^{f}W^v	Filter-fixed wrapper-varied in IWMs.
UfOTFs$^{TU1:TU28}$	Union of final OTFs of TU^1 to TU^{28}.
OCBF-X	Output channels of basic feature (X); X: {VOLT/ VANGLE/ PELEC/ QELEC/ QLOAD}.
^{SVM}IWSS^{3ker}	Embedding SVM model equipped with triple kernel in IWSS.
^{TWSVM}IWSS^{3ker}	Embedding TWSVM model equipped with triple kernel in IWSS.
^{SVM}IWSSr^{3ke}	Embedding SVM model equipped with triple kernel in IWSSr.
^{TWSVM}IWSSr^{3ker}	Embedding TWSVM model equipped with triple kernel in IWSSr.

I. INTRODUCTION

Nowadays, information technology (IT) by integrating different data-driven systems, plays the pivot role in collecting a large amount of data in different sensitive industries. The raw data obtained by the IT paradigm provide the necessary conditions for conducting data-oriented actions instead of experience-based operations in all tasks and responsibilities of system operators [1]–[5]. Such restructuring in the decision-making process will be possible through data mining (DM) technology which triangulated machine learning (ML), statistical learning (SL), and dataset to discover useful patterns for predicting different phenomena [6], [7]. Besides the importance of the type of ML and SL methods for achieving efficiency in learning procedures, the high-dimensional data (HDD)with sparse-dissimilar features is the most significant factor that negatively affects the training-testing

procedures (TTP) of learning frameworks. In this regard, the concept of the curse of dimensionality is defined as a great challenge on the way of high-performance DM [8], [9]. Furthermore, the importance of inconsistent-linked indices like accuracy and time (A&T) to make A&T-critical prediction in real-world problems exacerbates the necessitate of focusing on HDD concern. To fill two needs with one deed, DM engineers apply FSS on HDD for extracting the optimal features (OFs) set [10], [11]. The survived OFs based on the FSS will bring two points: first, the low processing time to predict unseen cases due to the mapping HDD to low-dimensional feature space, and second, high accuracy prediction induced by selecting most discriminative features (MDFs).

One of the HDD-oriented real-world problems is transient stability assessment (TSA) related to secure power supply [12], [13]. The IT-based grid infrastructure equipped with phasor measurement units (PMUs) gathers high-dimensional transient features (HDTF) for transient analysis (TA) [14]. In HDTF, the presence of irrelevant and redundant features is problematic for TTP of predictive approaches, which cause low-accuracy transient stability prediction (TSP). By applying the FSS scheme on HDTF, optimal transient features (OTFs) are selected to achieve high data labeling accuracy (DLA) in TSP. Also, the severe-sudden essence of transient stability necessitates using the FSS for compacting the HDTF to decrease data labeling time (DLT) in TSP, including observed time and prediction time [15]. In terms of the low DLT, low dimensional optimal transient features (LDOTF) caused fast learning in TTP scenarios leads to low prediction time, and existing the most relevant features in LDOTF allow picking up small-optimal observations. Consequently, by applying the FSS scheme on HDTF, system operators will be able to take timely-accurate corrective control actions to provide secure-adequate exploitation of the power grid. Hence, to achieve a win-win trade-off (high DLA and low DLT), designing the comprehensive FSS has been widely considered by DM researchers for TA.

II. RELATED WORKS

Reviewing the FSS-based transient studies shows that optimal transient features are selected by filter and filter-wrapper (hybrid) methods. In term of filter-oriented FSS, in Reference [16], [17], mutual information theory applied on transient characteristics related to power and angle to select optimal features regarding two principles: selected transient features have maximum relevance to the target class and have minimum relevance to one another, which is called minimum-redundancy and maximum-relevance (mRMR) FSS. Reference [18] introduce the ReliefF algorithm for calculating relevancy of rotor faults features to predict the health state of induction motor. To calculate total transfer capability regarding transient stability limitations, designing the feature pre-screening scenario based on the fast correlation-based filter (FCBF) is considered in [19]. Based on FCBF, optimal features of active and reactive load power, phase

angles of bus voltages, and the induced electromotive force of generators are selected to achieve training-testing efficiency. In Reference [20], for constructing the dynamic security assessment (DSA) model for predicting the transient stability margin, applying the FSS algorithm based on partial mutual information (PMI) and the Pearson correlation coefficient (PCC) is considered as the main step of the DSA model. In terms of hybrid FSSs, in Reference [21], filter-wrapper FSS includes feature weight ranking by Relief (filter as the preliminary) step, and five-fold cross-validation SVM model (wrapper as the complementary step) are applied on trajectory cluster features for selecting optimal feature set. Based on the proposed hybrid FSS in [22], first, normalized mutual information (NMI) ranks the initial features in the form of strongly relevant feature subset (SRFS) and the weakly relevant feature subset (WRFS). Next, the obtained knowledge of the filter phase is fed to the wrapper phase equipped with an easy-implementing search algorithm called binary particle swarm optimization (BPSO) to improve the effectiveness of FSS results.Considering high-dimensional multivariate time series data obtained by transient simulations, Reference [23] designed hybrid FSS in bi-mode, including trajectory-based filter-wrapper method (TFWM) and point-based filter-wrapper method (PFWM). In TFWM, mutual information-entropy-based (MIE) calculations (filter) and fuzzy imperialist competitive algorithm (FICA)-IWSS-based trihedral kernel-SVM (wrapper) find the optimal transient series. Next, the PFWM, including MIE calculations (filter) and the Gaussian kernel-SVM (wrapper) utilized to find optimal point features per optimal time series.

Regardless of the precise mining on transient feature space by the abovementioned FSSs, which have led to the acceptable performance in TSA, designing the comprehensive hybrid framework to extract masked-relevant transient features is the greatest challenge to achieve timely-accurate TSA. Lack of cross-oriented learning mechanisms in the form of multifaceted hybrid FSS (MHFSS) causes some features with the discriminative character don't survive in the feature selection process. In this regard, focusing on the structure of the filter or hybrid FSSs in previous studies shows the fact that the mining of intrinsic characteristics of transient data for selecting optimal features is based on the unilateral strategy equipped with vertically learning. Such a mechanism may be applicable in selecting the optimal features to improve the TSA, but it will ignore optimal-blurred transient features. Furthermore, the characteristic of transient data is the main parameter in determining how to apply the proposed MHFSS to it. Having a glance at FSS-based studies shows that the FSSs applied on multivariate point or time series data in the whole-manner. Such a strategy tends to select the OTFs without regard to the possibility of optimal features sacrificing related to each univariate trajectory. Streaming k-variate time-series data obtained by the PMU-based synchronized measurement necessitate extracting univariate-specific OTFs in the form of univariate-oriented learning in both filter and wrapper phase.
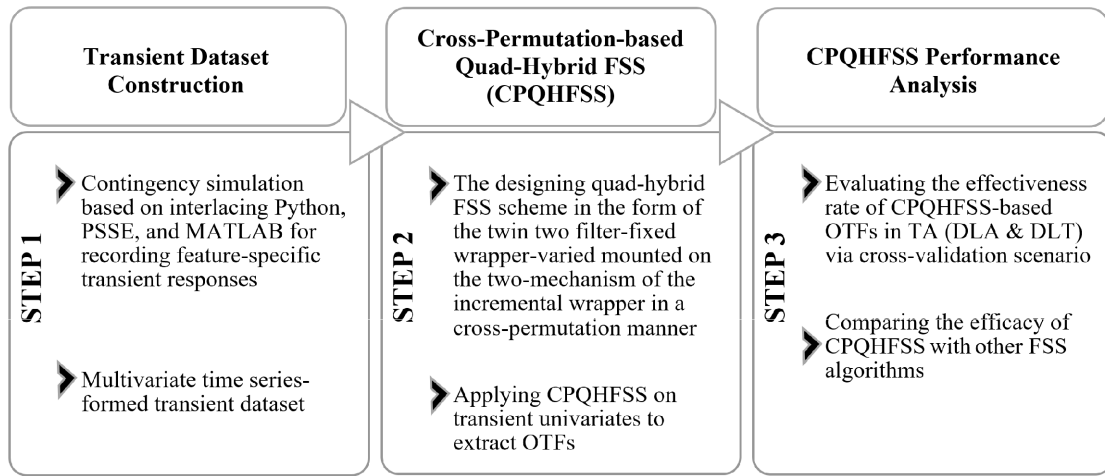
FIGURE 1. The Overall framework of FSS-oriented TA based on CPQHFSS.

The main contributions of this study to solve above-mentioned challenges in FSS-based TSA are summarized as follows:

- A new feature selection algorithm called cross-permutation quad-hybrid FSS (CPQHFSS) was considered to select OTFs to achieve high-performance TA. This scheme was designed via a multifaceted hybrid scenario accompanied by the entropy-based metric in the filter phase and hyperplane-based learning methods in the wrapper phase.
- The CPQHFSS applied on high-dimensional TMEs based on partial-manner learning for extracting univariate-specific MDFs. The optimal features per univariates are survived by the conducting cross-permutation scenario of the proposed FSS. Such a mechanism guaranteed the optimal-blurred transient features extraction to achieve high DLA and low DLT.
- The performance of CPQHFSS-specific OTFs in TA was compared with selected OTFs by other FSS based on the cross-validation technique.

The rest of the paper is organized as follows: The detailed descriptions of the CPQHFSS are remarked in Section 3. Experimental results of applying CPQHFSS on univariates of TMEs for TSA are presented in Section 4. Also, the comparison results between the proposed FSS and the other FSSs are interpreted in Section 4. Finally, the conclusion is depicted in Section 5.

III. CROSS-PERMUTATION-BASED QUAD-HYBRID FEATURE SELECTION SCHEME (CPQHFSS)

The overall framework of FSS-oriented TA based on CPQHFSS is shown in Fig. 1. After transient data gathering phase, we offer CPQHFSS including twin two-filter-wrapper blocks (2FWBs) mounted on the two-mechanism of the incremental wrapper namely incremental wrapper subset selection (IWSS^{2FWBs}) and IWSS with replacement (IWSSr^{2FWBs}). The relevancy ratio-support vector machine (RR-SVM) is

the filter-fixed wrapper-varied ($F^f W^v$) embedded in the first block of IWSS^{2FWBs} and IWSSr^{2FWBs}. The second block-specific $F^f W^v$ of IWSS^{2FWs} and IWSSr^{2FWBs} is designed by relevancy ratio-twin support vector machine (RR-TWSVM). Besides direct relations in four hybrid blocks, cross-permutation-based relations in IWSS^{2FWBs} and IWSSr^{2FWBs} are defined in CPQHFSS. By plugging the different kernels into hyperplane-based wrapper methods of IWSS^{2FWBs} and IWSSr^{2FWBs}, all possible cross-permutations of two states per IWSS^{2FWBs} and IWSSr^{2FWBs} caused to applying different hybrid FSS on each univariate of transient multivariate excursions (TMEs) to extract OTFs. In the third step, transient analysis based on survived OTFs in the presence of cross-validation scenario evaluates the effectiveness rate of the OTFs in achieving high-performance TA.

The CPQHFSS by interlacing the filter-wrapper methods, incremental wrapper mechanisms, and cross-permutation scenario, selects OTFs of TMEs for high-performance TA. According to Fig. 2, each univariates of the transient multivariate trajectories dataset is entered into the filter phase as the first step of CPQHFSS (See Fig. 2, filter funnel). Then, the obtained filter-based results per univariate are used in the wrapper phase of the CPQHFSS, which consists of four hybrid blocks categorized in twin two-FWBs, which are mounted on dual incremental wrapper mechanisms (IWSS and IWSSr). Overall, in the CPQHFSS, four hybrid blocks formed as IWSS^{2FWBs} (the left rectangle box of Fig. 2) and IWSSr^{2FWBs}(the right rectangle box of Fig. 2). The direct relations in each block of IWSS^{2FWBs} or IWSSr^{2FWBs} be caused that two $F^f W^v$-states (totally four $F^f W^v$-states). In CPQHFSS, due to plugging elastic and non-elastic kernels into hyperplane-based approaches situated in the wrapper methods, different cross-permutation hybrid FSS can be considered for two filter-wrapper-states of IWSS^{2FWBs} and IWSSr^{2FWBs} (See cross-permutation box in Fig. 2). Taking into cognizance the concise explanation of CPQHFSS depicted in Fig. 2, the pseudocode of CPQHFSS is shown
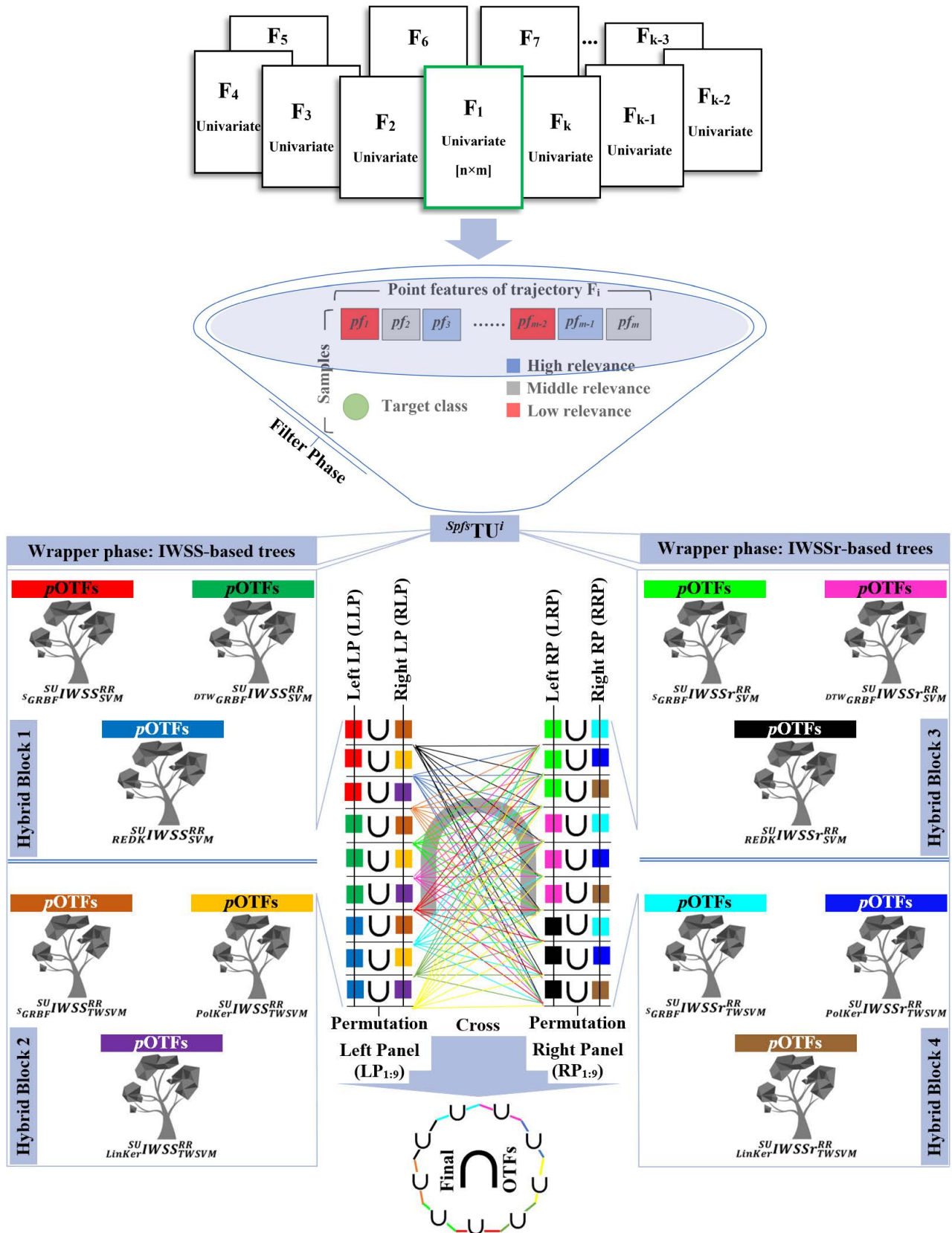
FIGURE 2. Overall process of CPQHFSS.

TABLE 1. The pseudocode of the CPQHFSS.

Main body of CPQHFSS

Input: transient univariates (TU) of multivariate trajectory dataset; $\{TU^k \mid k=1, 2, \ldots, n\}$.

Output: union of final optimal transient features ($UfOTFs^{TU1: TUn}$).

(1) **for** k=1 to n

(2)   **for** i=1 to m // m=length (pfs), pfs: point features (observed cycles) in TU^k.

(3) $SU\dfrac{TU^k}{pf_i}$ = calculate relevancy rate (RR) of $pf_i \in TU^k$;

(4)   **end**

(5) **end**

(6) **for** k=1 to n

(7) $^{Spfs}TU^k$ =**Sort** ($SU^{TU^k}_{pfs}$); // Sort $SU\dfrac{TU^k}{pf_{1:m}}$ array in descending manner.

(8) $^{pOTFs}_{IWSS}TU^k$ =**IWSSr** ($^{Spfs}TU^k$); // Recording preliminary optimal transient features ($pOTFs$) per TU^k in ($\left[^{pOTFs}_{IWSS}TU^k \right]_{2\times3}$);

(9) $^{pOTFs}_{IWSSr}TU^k$ =**IWSS** ($^{Spfs}TU^k$); // Recording preliminary optimal transient features ($pOTFs$) per TU^k ($\left[^{pOTFs}_{IWSSr}TU^k \right]_{2\times3}$);

(10)   **if** k=1

(11) $^{pOTFs}IWSS^{TU}$=struct($^{pOTFs}TUs'$, $\left[^{pOTFs}_{IWSS}TU^k \right]_{2\times3}$); $^{pOTFs}IWSSr^{TU}$=struct($^{pOTFs}TUs'$, $\left[^{pOTFs}_{IWSSr}TU^k \right]_{2\times3}$);

(12)   **else**

(13) $^{pOTFs}IWSS^{TU}$(end+1)= struct($^{pOTFs}TUs'$, $\left[^{pOTFs}_{IWSS}TU^k \right]_{2\times3}$); $^{pOTFs}IWSSr^{TU}$(end+1)= struct($^{pOTFs}TUs'$, $\left[^{pOTFs}_{IWSSr}TU^k \right]_{2\times3}$);

(14)   **end**

(15) **end**

(16) $fOTFs^{TUk}$= **cross-permutation** ($^{pOTFs}IWSS^{TU}$, $^{pOTFs}IWSSr^{TU}$);

(17) $UfOTFs^{TU1:TUn}$= $fOTFs^{TU1} \cup fOTFs^{TU2} \cup \ldots \cup fOTFs^{TUn}$;

Function: IWSSr

(1) Classifier: {SVM, TWSVM}, Kernel^{Cat1}: {^{S}GRBF, ^{DTW}GRBF, REDK}, Kernel^{Cat2}: {^{S}GRBF, PolKer, LinKer};

(2) Sel=$^{Spfs}TU^k$ {1}; // first element of $^{Spfs}TU^k$ array (feature with highest SU insert in Sel).

(3) SVM^{Kernelx}= Classifier {1} + Kernel^{Cat1}{x}; // x=1 | 2 | 3, Learning model: {SVM^{Kernelx}}, Three learning modes.

(4) TWSVM^{Kernely}= Classifier {2} + Kernel^{Cat2} {y}; // y=1 | 2 | 3, Learning model: {TWSVM^{Kernely}}, Three learning modes.

(5) AccSel=PerEval (Learning Model, $\left[data \right]_{m\times Sel\ with\ C}$);   // C: [target class]$_{m\times1}$; m: number of sample, PerEval: Performance Evaluation.

(6) **for** i=2 to length ($^{Spfs}TU^k$)

(7)   OptFea=$\varnothing$;

(8)   **for** r=1 to length (Sel)

(9)   Sel^{temp}=update^{SelSub} (copy(Sel), swap(Sel{r}, $^{Spfs}TU^k$ {i}));

(10)   AccSel^{temp}= PerEval (Learning Model, $\left[data \right]_{m\times Sel^{temp}\ with\ C}$);

(11)   **if** (AccSel^{temp}>AccSel)

(12)   OptFea=swap (Sel{r}, $^{Spfs}TU^k$ {i});

(13)   Acc= AccSel^{temp};

(14)   **end**

(15)   **end**

(16)   Sel^{temp}=update^{SelSub} (copy(Sel), add ($^{Spfs}TU^k$ {i}));

(17)   AccSel^{temp}=PerEval (Learning model, $\left[data \right]_{m\times Sel^{temp}\ with\ C}$);

(18)   **if** (AccSel^{temp}>AccSel)

(19)   OptFea=Sel^{temp};

(20)   AccSel=AccSel^{temp};

(21)   **end**

(22)   **if** (OptFea != null)

(23)   update (Sel, OptFea);

(24)   **end**

(25) **end**

(26) **return** Sel; // Sel returned as matrix $\left[^{pOTFs}_{IWSSr}TU^k \right]_{2\times3}$, first row of matrix related to selected {Sel} by SVM^{Kernelx} and second row of matrix related to selected {Sel} by TWSVM^{Kernely}.

Function: IWSS

(1) Sel=$^{Spfs}TU^k$ {1};

(2) AccSel=PerEval (Learning Model, $\left[data \right]_{m\times Sel\ with\ C}$);

(3) **for** i=2 to length ($^{Spfs}TU^k$)

(4)   Sel^{temp}=add (copy(Sel), $^{Spfs}TU^k$ {i});

(5)   AccSel^{temp}=PerEval (Learning Model, $\left[data \right]_{m\times Sel^{temp}\ with\ C}$);

TABLE 1. *(Continued.)* The pseudocode of the CPQHFSS.

(6)	**if** (AccSel$^{\text{temp}}$>AccSel)
(7)	add (Sel, ^{Spfs}TU^k {i});
(8)	AccSel=AccSel$^{\text{temp}}$;
(9)	**end**
(10)	**end**

(11) **return** Sel; // Sel returned as matrix $\left[\begin{smallmatrix} pOTFs \\ IWSS \end{smallmatrix} TU^k \right]_{2\times3}$, first row of matrix related to selected {Sel} by SVM$^{\text{Kernelx}}$ and second row of matrix related to selected {Sel} by TWSVM$^{\text{Kernely}}$.

Function: Cross-Permutation

(1) **for** k=1 to n

(2) $^{\text{IWSS}}$mat=$^{p\text{OTFs}}$IWSS^{TU}(k).$^{p\text{OTFs}}$TUs; // insert k^{th} 6-element matrix ($\left[\begin{smallmatrix} pOTFs \\ IWSS \end{smallmatrix} TU^k \right]_{2\times3}$) of struct $^{p\text{OTFs}}$IWSS^{TU} in $^{\text{IWSS}}$mat.

(3) $^{\text{IWSSr}}$mat=$^{p\text{OTFs}}$IWSSr^{TU}(k).$^{p\text{OTFs}}$TUs; // insert k^{th} 6-element matrix ($\left[\begin{smallmatrix} pOTFs \\ IWSSr \end{smallmatrix} TU^k \right]_{2\times3}$) of struct $^{p\text{OTFs}}$IWSSr^{TU} in $^{\text{IWSSr}}$mat.

// first row of matrix $\left[^{IWSS / IWSSr} mat \right]_{2\times3}$ (elements {1, 3, 5}) related to selected {Sel} by SVM$^{\text{Kernelx}}$.

// second row of matrix $\left[^{IWSS / IWSSr} mat \right]_{2\times3}$ (elements {2, 4, 6}) related to selected {Sel} by TWSVM$^{\text{Kernely}}$.

// Permutations of IWSS-based learning models (9 permutations (LP1: LP9)).

(4) LP1=$^{\text{IWSS}}$mat (1) ∪ $^{\text{IWSS}}$mat (2); LP2= $^{\text{IWSS}}$mat (1) ∪ $^{\text{IWSS}}$mat (4); LP3=$^{\text{IWSS}}$mat (1) ∪ $^{\text{IWSS}}$mat (6);

(5) LP4=$^{\text{IWSS}}$mat (3) ∪ $^{\text{IWSS}}$mat (2); LP5= $^{\text{IWSS}}$mat (3) ∪ $^{\text{IWSS}}$mat (4); LP6= $^{\text{IWSS}}$mat (3) ∪ $^{\text{IWSS}}$mat (6);

(6) LP7=$^{\text{IWSS}}$mat (5) ∪ $^{\text{IWSS}}$mat (2); LP8= $^{\text{IWSS}}$mat (5) ∪ $^{\text{IWSS}}$mat (4); LP9= $^{\text{IWSS}}$mat (5) ∪ $^{\text{IWSS}}$mat (6);

// Permutations of IWSSr-based learning models (9 permutations (RP1: RP9)).

(7) RP1=$^{\text{IWSSr}}$mat (1) ∪ $^{\text{IWSSr}}$mat (2); RP2= $^{\text{IWSSr}}$mat (1) ∪ $^{\text{IWSSr}}$mat (4); RP3=$^{\text{IWSSr}}$mat (1) ∪ $^{\text{IWSSr}}$mat (6);

(8) RP4=$^{\text{IWSSr}}$mat (3) ∪ $^{\text{IWSSr}}$mat (2); RP5= $^{\text{IWSSr}}$mat (3) ∪ $^{\text{IWSSr}}$mat (4); RP6=$^{\text{IWSSr}}$mat (3) ∪ $^{\text{IWSSr}}$mat (6);

(9) RP7=$^{\text{IWSSr}}$mat (5) ∪ $^{\text{IWSSr}}$mat (2); RP8= $^{\text{IWSSr}}$mat (5) ∪ $^{\text{IWSSr}}$mat (4); RP9=$^{\text{IWSSr}}$mat (5) ∪ $^{\text{IWSSr}}$mat (6);

// Cross relations.

(10) C1=∪ [LP1 ∩ (RP1: RP9]; C2=∪ [LP2 ∩ (RP1: RP9]; C3=∪ [LP3 ∩ (RP1: RP9];
C4=∪ [LP4 ∩ (RP1: RP9]; C5=∪ [LP5 ∩ (RP1: RP9]; C6=∪ [LP6 ∩ (RP1: RP9];
C7=∪ [LP7 ∩ (RP1: RP9]; C8=∪ [LP8 ∩ (RP1: RP9]; C9=∪ [LP9 ∩ (RP1: RP9];

(11) fOTFs$^{\text{TUk}}$= C1 ∩ C2 ∩ … ∩ C9;

(12) **end**

in Table 1. As can be seen in Table 1, the main body of CPQHFSS includes filter phase (RR analysis), incremental wrapper mechanisms (IWMs), and cross-permutation function. In the main body of CPQHFSS, symmetric uncertainty values(Line 3) of point features per transient univariate (TU) are calculated. After sorting point features (pfs) of TU^k based on SU measure (Line 7), these features are entered into IWSS and IWSSr functions (Line 8 and Line 9) to extract preliminary optimal transient features (pOTFs). Each IWM is equipped with dual hyperplane-based classifiers (SVM and TWSVM) to exert the train-test procedures. Also, to find optimal separating hyperplane, different kernels are plugged into SVM and TWSVM classifiers. After selecting pOTFs per TU in a six-states manner by IWMs ($\left[\begin{smallmatrix} pOTFs \\ IWSS \end{smallmatrix} TU^k \right]_{2\times3}$ and $\left[\begin{smallmatrix} pOTFs \\ IWSSr \end{smallmatrix} TU^k \right]_{2\times3}$), these matrices for each TU are recorded in the structure arrays ($^{p\text{OTFs}}$IWSS^{TU}, $^{p\text{OTFs}}$IWSSr^{TU}) (Line 10-14). Finally, structure arrays are entered into the cross-permutation function to extract final OTFs (fOTFs) per TU (Line 16). After conducting union-intersection operations on subsets of optimal features based on cross-permutation scenario, the union of fOTFs (UfOTFs) is obtained (Line 17). To better understand the details of the pseudocode of CPQHFSS, we elaborate on the triple components of it (filter, IWMs, and cross-permutation) in Sections III-A to III-C.

Besides the above-mentioned describing the main body of pseudocode of CPQHFSS, we present the complexity of CPQHFSS for readers at a glance. The complexity of CPQHFSS is related to IWMs accompanied by hyperplane-based learning methods. By analyzing these main functions, we can approximate the complexity of CPQHFSS. In the worst case, the complexity of IWSS and IWSSr is O(n) and O(n^2), respectively [24]. Also, the complexity of SVM and TWSVM is O(n^3) and O(2×(n/2)^3), respectively [25]. Hence, the complexity of $^{\text{SVM}}$IWSS$^{\text{TWSVM}}$ is O(max{(n×n^3), (n×2×(n/2)^3)}) and $^{\text{SVM}}$IWSSr$^{\text{TWSVM}}$ has O(max{(n^2×n^3), (n^2×2×(n/2)^3)}) complexity. Since the complexity of the SVM is 4 times larger than of the TWSVM, the complexity of $^{\text{SVM}}$IWSS$^{\text{TWSVM}}$ and $^{\text{SVM}}$IWSSr$^{\text{TWSVM}}$ will be equal to O(n×n^3) and O(n^2×n^3), respectively. On the other hand, the experiment results of Reference [24] show the fact that the complexity of IWSSr is near to IWSS when the number of variables to be selected is a very small number of wrapper evaluations. Consequently, the CPQHFSS has O(2×n^4) complexity.

A. INCREMENTAL WRAPPER MECHANISMS (IWMs)
1) IWSS
The IWSS mechanism [26] is utilized in the first set of twin 2FWBs (IWSS$^{\text{2FWBs}}$) in CPQHFSS as IWM. How to

Features	f_1	f_2	f_3	f_4
Values of RR	0.56	0.49	0.81	0.76
Sorted array	$f_{h1}{:}f_3$	$f_{h2}{:}f_4$	$f_{h3}{:}f_1$	$f_{h4}{:}f_2$
	0.81	0.76	0.56	0.49

FIGURE 3. The IWSS algorithm.

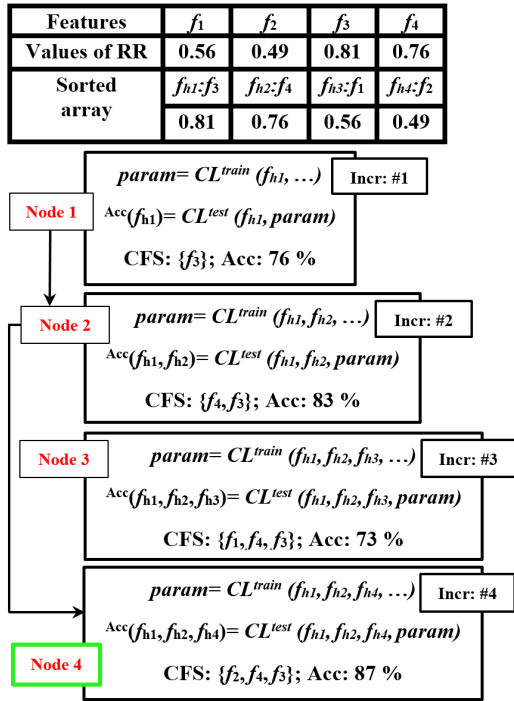| Features	f_1	f_2	f_3
Values of RR	0.76	0.65	0.81
Sorted array	$f_{h1}{:}f_3$	$f_{h2}{:}f_1$	$f_{h3}{:}f_2$
	0.81	0.76	0.65

FIGURE 4. The IWSSr algorithm.

navigate in the incremental process of IWSS to select optimal features depends on the results of the embedded filter and wrapper method. As the preliminary step of IWSS, by conducting the filter method on feature set based on relevancy ratio (RR), the features are sorted based on RR values in descending manner. Then, for completing the first increment of IWSS, the feature inserted into the first position of the sorted array (f_{h1}: feature with highest RR) is fed to the classification learner (CL), and then f_{h1} with the prediction accuracy ($^{Acc}(f_{h1})$) is recorded in the candidate features subset (CFS) based on the TTP. In the next increment, the feature with second-highest RR (f_{h2}) is added to the CFS, and the updated CFS-based learning model reports the $^{Acc}(f_{h1},f_{h2})$. If the classification performance of CFS including f_{h1} and f_{h2} is higher than the performance of f_{h1}, the third increase (adding f_{h3}) is accompanied by f_{h1} & f_{h2}; otherwise, f_{h2} is deleted from CFS, and f_{h3} is added to the CFS and placed next to f_{h1}. Fig. 3 shows how to select OTFs by the IWSS in the form of a numerical example.

2) IWSSr

In the second set of twin 2FWBs, filter-wrapper blocks are mounted on IWSSr [24] algorithm (IWSSr^{2FWBs}) as IWM. In IWSSr, similar to the dependence of the IWSS on filter and wrapper method results, based on sorted RR values of features, in the first increment, f_{h1} is added to CFS, then CL trained by f_{h1} and $^{Acc}(f_{h1})$ is recorded. In the second increment, f_{h2} is added to the preceding CFS in two modes. In the first mode, f_{h1} is replaced by f_{h2} (only f_{h2} added to CFS) and in the second mode, f_{h1} and f_{h2} are added to CL together.

Now, in the second increment, $^{Acc}(f_{h1})$ and $^{Acc}(f_{h1},f_{h2})$ are obtained. Fig. 4, shows the process of IWSSr which the third increment starts from node 3 (create node 4 to 6).

B. TWIN TWO FILTER-WRAPPER BLOCKS (2FWBs)

Twin 2FWBs in CPQHFSS refer to applying two sets of 2FWBs in the presence of IWMs which are called IWSS^{2FWBs} and IWSSr^{2FWBs}. The 2FWBs include frozen information theory concept (filter-fixed) and unfixed machine learning classifier (wrapper-varied). Filter-related model of 2FWBs including the relevancy ratio (RR) [27] in the first-second hybrid block (HB$^{1-2}$) of IWSS^{2FWBs} ($^{RR}IWSS_{HB^{1-2}}^{2FWBs}$) and IWSSr^{2FWBs} ($^{RR}IWSSr_{HB^{1-2}}^{2FWBs}$). Wrapper-related models including support vector machine in the HB^1 of IWSS^{2FWBs} ($_{SVM}IWSS_{HB^1}^{2FWBs}$) and IWSSr^{2FWBs} ($_{SVM}IWSSr_{HB^1}^{2FWBs}$) and twin support vector machine (TWSVM) in HB^2 of IWSS^{2FWBs} ($_{TWSVM}IWSS_{HB^2}^{2FWBs}$) and IWSSr^{2FWBs}($_{TWSVM}IWSSr_{HB^2}^{2FWBs}$). Generally, in respect of twin 2FWBs, one set of 2FWBs is situated in IWSS formed as $_{SVM}^{RR}IWSS_{HB^1}^{2FWBs}$, and $_{TWSVM}^{RR}IWSS_{HB^2}^{2FWBs}$. Also, another set of 2FWBs is associated with IWSSr raised as $_{SVM}^{RR}IWSSr_{HB^1}^{2FWBs}$, and $_{TWSVM}^{RR}IWSSr_{HB^2}^{2FWBs}$. The detailed descriptions of the filter and wrapper models of 2FWBs are discussed in the following subsections.

1) FILTER-FIXED METHODS IN 2FWBS

Relevancy ratio (RR): The symmetric uncertainty (SU) is considered as the HB$^{1-2}$-specific filter index in IWSS^{2FWBs} and IWSSr^{2FWBs}, which triangulated the entropy, conditional entropy, and mutual information (MI) to measure the relevancy rate between features of transient univariates and class

label. The SU index is calculated as:

$$SU(f_i^{TU^k}, C) = 2 \frac{MI(f_i^{TU^k}; C)}{H(f_i^{TU^k}) + H(C)} \tag{1}$$

where $f_i^{TU^k}$ represents i^{th} features of kth transient univariate, and C is the class label of transient samples. In (1), the entropy $H(Z)$ is defined as:

$$H(Z) = - \sum_{z \in Z} p(z) \log p(z) \tag{2}$$

where Z is a discrete random variable and $p(z) = \Pr\{Z = z\}$ is a probability density function. Mutual information (*MI*) in (1) is defined as:

$$MI(f_i^{TU^k}; C) = H(f_i^{TU^k}) - H(f_i^{TU^k}|C) \tag{3}$$

where $H(f_i^{TU^k}|C)$ in (3) is called conditional entropy as follow:

$$H(f_i^{TU^k}|C) = - \sum_{x \in f_i^{TU^k}} \sum_{c \in C} p(x, c) \log p(x|c) \tag{4}$$

2) WRAPPER METHODS IN 2FWBS

a: SUPPORT VECTOR MACHINE (SVM)

SVM in [28] introduced as a supervised learning model that draws hyperplane in feature space for classifying binary or multi-class data in the form of the linear SVM (hard margin or soft margin approach) and nonlinear SVM (kernel-based approach). Regardless of the different factors that affected the SVM formula, SVM aims a low structural risk without overfitting data to achieve high accuracy in train-test procedures. For example, achieving such a goal in the presence of data that has a nonlinear decision boundary (e.g., PMU-gathered HDTF), necessitates plugging the kernel trick into SVM computations as follows:

$$a^* = \arg \min_\alpha \frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l \alpha_i \alpha_j y_i y_j K(x_i, x_j) - \sum_{k=1}^l \alpha_k;$$

$$0 \le \alpha_i \le C, \quad \sum_{j=1}^l \alpha_i y_i = 0, \ i, j = 1, \dots, l \tag{5}$$

where $K(x_i, x_j)$ in (5) is employed for mapping the data from the main space to the new space (high dimensional space) so that in the new space the data are linearly separable. The maximum-margin separating hyperplane in feature space is solved by (6):

$$f(x) = \text{sgn} \left(\sum_{i \in s} \alpha_i y_i K(x_i, x) + b \right);$$

$$b = \frac{1}{s} \sum_{i \in s} \left[y_i - \sum_j \alpha_j y_j K(x_j, x_i) \right] \tag{6}$$

b: TWIN SUPPORT VECTOR MACHINE (TWSVM)

The standard SVM is formulated based on finding the middle boundary (maximum margin) between two parallel planes with the maximum distance from each of the existing classes. SVM geometry space can be reshaped by cross planes in which each plane could nearest distance to the samples of one class and farthest from the samples of the other class. This idea was raised as the generalized proximal eigenvalue support vector machine (GEPSVM) [28]. In another effort by [29], the spirit of the GEPSVM was kept into a new skeleton (new formulation) was termed TWSVM. In TWSVM, cross planes are obtained by solving the following optimization problems:

$$\min_{w_1, b_1, q} \frac{1}{2} ||Pw_1 + e_1 b_1||^2 + c_1 e_2^T q$$
$$s.t. \quad -(Qw_1 + e_2 b_1) + q \ge e_2, \quad q \ge 0 \tag{7}$$

$$\min_{w_2, b_2, q} \frac{1}{2} ||Qw_2 + e_2 b_2||^2 + c_2 e_1^T q$$
$$s.t. \quad (Pw_2 + e_1 b_2) + q \ge e_1, \quad q \ge 0 \tag{8}$$

where c_1, c_2, e_1, and e_2 are vectors with a value of 1 and a proper dimension. By calculating the Lagrangian function for (7) and (8), the Karush–Kuhn–Tucker (KKT) equations are formed. By placing the KKT terms and relations in the Lagrangian function for each of the relations (7) and (8), the dual optimal relations are obtained according to the following relations:

$$^{dual}TWSVM^1: \quad \max\{e_2^T \alpha - \frac{1}{2} \alpha^T G(H^T H)^{-1} G^T \alpha\} \tag{9}$$

$$^{dual}TWSVM^2: \quad \max\{e_1^T \psi - \frac{1}{2} \psi^T P(Q^T Q)^{-1} P^T \psi\} \tag{10}$$

Based on the dual optimization problems, the value α and ψ is obtained via quadratic programming and by placing these values in the KKT relations, the values $[w^{(1)}, b^{(1)}]$ and $[w^{(2)}, b^{(2)}]$ related to hyperplanes of the binary-class task are obtained:

$$X^T w^{(1)} + b^{(1)} = 0 \quad \text{and} \quad X^T w^{(2)} + b^{(2)} = 0 \tag{11}$$

Finally, the class label of the unseen point $x \in \mathbb{R}^n$ is determined from the plane that is close to this point:

$$Class \ x = \arg_v \min \left| x^T w^{(v)} + b^{(v)} \right|; \quad v = 1, 2 \tag{12}$$

Based on the above-mentioned principle of TWSVM, the nonlinear classification of HDTF can be considered via kernel-based cross planes [29]:

$$K(x^T, C^T)u^{(1)} + b^{(1)} = 0 \quad \text{and} \quad K(x^T, C^T)u^{(2)} + b^{(2)} = 0 \tag{13}$$

where $C^T = [A \ B]^T$ and K denote the kernel. The vector $[u^{(1)} \ b^{(1)}]^T$ and $[u^{(2)} \ b^{(2)}]^T$ are obtained by solving the following optimization problem:

$$^K TWSVM^1: \quad \min_{u^{(1)}, b^{(1)}, q} \frac{1}{2} ||(K(A, C^T)u^{(1)} + e_1 b^{(1)}||^2$$
$$+ c_1 e_2^T q$$

$$s.t. \quad -(K(B, C^T)u^{(1)} + e_2 b^{(1)})$$
$$+ q \geq e_2, \quad q \geq 0 \tag{14}$$

$$^K TWSVM^2: \quad \min_{u^{(2)}, b^{(2)}, q} \frac{1}{2}||(K(B, C^T)u^{(2)} + e_2 b^{(2)}||^2$$
$$+ c_2 e_1^T q$$
$$s.t. \quad (K(A, C^T)u^{(2)} + e_1 b^{(2)})$$
$$+ q \geq e_1, \quad q \geq 0 \tag{15}$$

C. CROSS-PERMUTATION SCENARIO BASED ON WRAPPER KERNELS

The cross-permutation scenario in CPQHFSS is conducted based on plugged triple kernel into wrapper approaches of twin 2FWBs. The kernels situated in wrapper approaches of IWSS^{2FWBs} and IWSSr^{2FWBs} are categorized into two types: 1) non-elastic kernel: linear kernel (LinKer) [31], polynomial kernel (PolKer) [31], and standard Gaussian radial basis function (^{S}GRBF) [28], and 2) elastic kernels: dynamic time warping in GRBF (^{DTW}GRBF) [32] and recursive edit distance kernel (REDK) [33]. The definitions per kernel are summarized below:

1) LINKER

LineKer is calculated based on the inner product plus constant c, which is considered as the simplest kernel function:

$$K(x, x') = x^T x' + c \tag{16}$$

2) POLKER

The degree-based variant of LinKer (the d value more than 1) is known as PolKer, which is defined as follow:

$$K(x, x') = (\alpha x^T x' + c)^d \tag{17}$$

3) ^{S}GRBF

^{S}GRBF is known as a non-elastic kernel due to linear alignment (point to point) in pattern matching in feature space. The ^{S}GRBF for using as $K(x, x')$ in (5) and (13) is defined as follows:

$$K(x, x') = \exp\left(-\frac{||x - x'||^2}{2\sigma^2}\right) \tag{18}$$

The distance between two trajectories is calculated based on squared Euclidean distance denoted by $||x - x'||^2$ in (18).

4) ^{DTW}GRBF

The elastic behavior of DTW motivates to replace Euclidean distance with DTW distance in ^{S}GRBF to have a robust and non-linear alignment in feature space. The DTW distance between two-time series is calculated by (19):

$$\text{distance}^{DTW}(A_1^p, B_1^q)$$
$$= d(a(p), b(q)) + Min\begin{pmatrix} \text{distance}^{DTW}(A_1^{p-1}, B_1^q) \\ \text{distance}^{DTW}(A_1^{p-1}, B_1^{q-1}) \\ \text{distance}^{DTW}(A_1^p, B_1^{q-1}) \end{pmatrix} \tag{19}$$
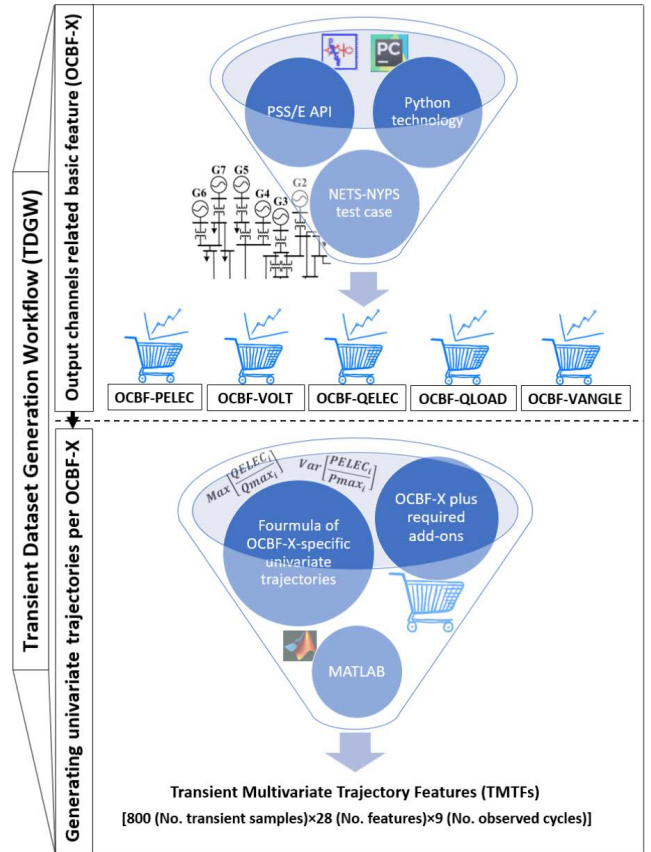
FIGURE 5. Transient dataset generation workflow (TDGW) [34].
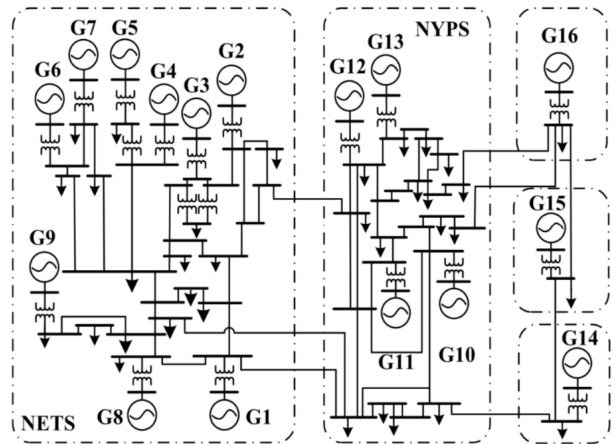
FIGURE 6. Single line diagram of NETS-NYPS test system.

According to (19), $K(x, x')$ can be defined as the DTW-based elastic kernel as follow:

$$K(x, x') = \exp\left(-\frac{\left[\text{distance}^{DTW}(A_1^p, B_1^q)\right]^2}{2\sigma^2}\right) \tag{20}$$

5) REDK

In [33], constructing the kernel based on the aggregation of scores recursively caused that the REDK is introduced as an
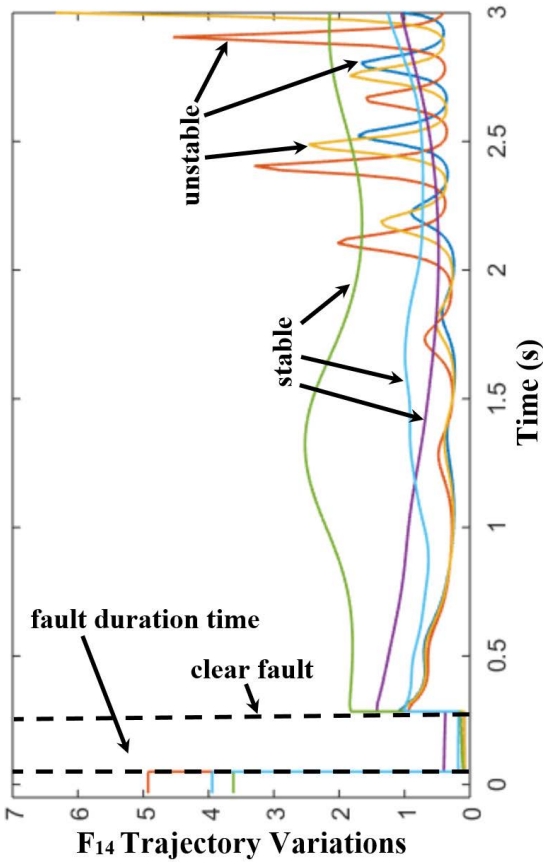
FIGURE 7. Stable and unstable samples based on F_{14} variations.
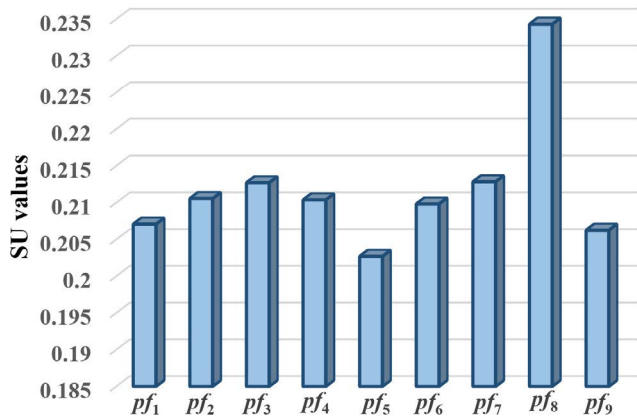
FIGURE 8. SU amount of $^{pfs}TU^6$.

efficient elastic similarity measure against classical elastic distances. According to (21), if for any pair of trajectory, such following equation is satisfied, the function $\langle \cdot, \cdot \rangle : U \times U \to R$ termed as REDK:

$$\langle A_1^p, B_1^q \rangle = \sum \begin{cases} \langle A_1^{p-1}, B_1^q \rangle f\left(\Gamma(A(p) \to \Lambda)\right) \\ \langle A_1^{p-1}, B_1^{q-1} \rangle f\left(\Gamma(A(p) \to B(q))\right) \\ \langle A_1^p, B_1^{q-1} \rangle f\left(\Lambda \to B(q)\right) \end{cases} \quad (21)$$

TABLE 2. Transient multivariate time series features (28-variate).

Math formula
$F_1^{t_m} = Max([\frac{PELEC_i}{P\max_i}]^{i=1:N_{genbus}})$
$F_2^{t_m} = Var([\frac{PELEC_i}{P\max_i}]^{i=1:N_{genbus}})$
$F_3^{t_m} = Max([\frac{QELEC_i}{Q\max_i}]^{i=1:N_{genbus}})$
$F_4^{t_m} = Min([\frac{QELEC_i}{Q\max_i}]^{i=1:N_{genbus}})$
$F_5^{t_m} = Var([\frac{QELEC_i}{Q\max_i}]^{i=1:N_{genbus}})$
$F_6^{t_m} = Max([VOLT_i]^{i=1:N_{bus}})$
$F_7^{t_m} = Var([VOLT_i]^{i=1:N_{bus}})$
$F_8^{t_m} = Max([VANGLE_i]^{i=1:N_{bus}}); \; slack\ bus = 0$
$F_9^{t_m} = Min([VANGLE_i]^{i=1:N_{bus}}); \; slack\ bus = 0$
$F_{10}^{t_m} = Var([VANGLE_i]^{i=1:N_{bus}}); slack\ bus = 0$
$F_{11}^{t_m} = Max(abs([VANGLE_i - VANGLE_j]^{i,j=1:N_{bus}}))$
$F_{12}^{t_m} = Mean(abs([VANGLE_i - VANGLE_j]^{i,j=1:N_{bus}}))$
$F_{13}^{t_m} = Var(abs([VANGLE_i - VANGLE_j]^{i,j=1:N_{bus}}))$
$F_{14}^{t_m} = \dfrac{\sum_{i=1}^{N_{busgen}} QLOAD_i}{\sum_{i=1}^{N_{busgen}} QELEC_i}$
$F_{15:28}^{t_m} = Gradient\ of\ features\ F_1 - F_{14}$

Symbol: t_m= moments in simulation time [1: s], $N_{bus\ gen}$ = number of bus generator in test case, PELEC= machine electrical power (pu), P_{max}= maximum amount of machine electrical power, QELEC=machine reactive power, Qmax= maximum amount of machine reactive power, Qload= reactive power consumption, Volt= bus pu voltages, N_{bus}= number of buses in test case, VANGLE= voltage phase angle, Var= variance, Max= maximum, Min= Minimum.

where $U = \{A_1^p | p \in \mathbb{N}\}$. A_1^p (or B_1^q) is the trajectory with a discrete index varying between 1 and p (or q). Also, $\Gamma(h)$ is the cost function for edit operation.

After applying different kernels in the hyperplane classifiers of wrapper phase (^{S}GRBF, ^{DTW}GRBF, and REDK in

TABLE 3. Filter-based ranking of point features of univariates.

Input	^{Spfs}TU${}^{\#}$ based on SU	Input	^{Spfs}TU${}^{\#}$ based on SU
TU^{1}	{${}^{1}pf_{7}$, ${}^{1}pf_{1}$, ${}^{1}pf_{6}$, ${}^{1}pf_{3}$, ${}^{1}pf_{5}$, ${}^{1}pf_{8}$, ${}^{1}pf_{9}$, ${}^{1}pf_{2}$, ${}^{1}pf_{4}$}	TU^{15}	{${}^{15}pf_{9}$, ${}^{15}pf_{1}$, ${}^{15}pf_{8}$, ${}^{15}pf_{6}$, ${}^{15}pf_{3}$, ${}^{15}pf_{7}$, ${}^{15}pf_{5}$, ${}^{15}pf_{2}$, ${}^{15}pf_{4}$}
TU^{2}	{${}^{2}pf_{1}$, ${}^{2}pf_{2}$, ${}^{2}pf_{3}$, ${}^{2}pf_{4}$, ${}^{2}pf_{5}$, ${}^{2}pf_{6}$, ${}^{2}pf_{7}$, ${}^{2}pf_{8}$, ${}^{2}pf_{9}$}	TU^{16}	{${}^{16}pf_{1}$, ${}^{16}pf_{2}$, ${}^{16}pf_{3}$, ${}^{16}pf_{4}$, ${}^{16}pf_{5}$, ${}^{16}pf_{6}$, ${}^{16}pf_{7}$, ${}^{16}pf_{8}$, ${}^{16}pf_{9}$}
TU^{3}	{${}^{3}pf_{3}$, ${}^{3}pf_{1}$, ${}^{3}pf_{3}$, ${}^{3}pf_{4}$, ${}^{3}pf_{6}$, ${}^{3}pf_{8}$, ${}^{3}pf_{9}$, ${}^{3}pf_{2}$, ${}^{3}pf_{4}$}	TU^{17}	{${}^{17}pf_{9}$, ${}^{17}pf_{2}$, ${}^{17}pf_{4}$, ${}^{17}pf_{3}$, ${}^{17}pf_{1}$, ${}^{17}pf_{7}$, ${}^{17}pf_{8}$, ${}^{17}pf_{5}$, ${}^{17}pf_{6}$}
TU^{4}	{${}^{4}pf_{4}$, ${}^{4}pf_{9}$, ${}^{4}pf_{1}$, ${}^{4}pf_{3}$, ${}^{4}pf_{5}$, ${}^{4}pf_{6}$, ${}^{4}pf_{7}$, ${}^{4}pf_{8}$, ${}^{4}pf_{2}$}	TU^{18}	{${}^{18}pf_{9}$, ${}^{18}pf_{1}$, ${}^{18}pf_{3}$, ${}^{18}pf_{8}$, ${}^{18}pf_{6}$, ${}^{18}pf_{5}$, ${}^{18}pf_{4}$, ${}^{18}pf_{7}$, ${}^{18}pf_{2}$}
TU^{5}	{${}^{5}pf_{1}$, ${}^{5}pf_{2}$, ${}^{5}pf_{3}$, ${}^{5}pf_{4}$, ${}^{5}pf_{5}$, ${}^{5}pf_{6}$, ${}^{5}pf_{7}$, ${}^{5}pf_{8}$, ${}^{5}pf_{9}$}	TU^{19}	{${}^{19}pf_{1}$, ${}^{19}pf_{2}$, ${}^{19}pf_{3}$, ${}^{19}pf_{4}$, ${}^{19}pf_{5}$, ${}^{19}pf_{6}$, ${}^{19}pf_{7}$, ${}^{19}pf_{8}$, ${}^{19}pf_{9}$}
TU^{6}	{${}^{6}pf_{8}$, ${}^{6}pf_{7}$, ${}^{6}pf_{3}$, ${}^{6}pf_{2}$, ${}^{6}pf_{4}$, ${}^{6}pf_{6}$, ${}^{6}pf_{1}$, ${}^{6}pf_{9}$, ${}^{6}pf_{5}$}	TU^{20}	{${}^{20}pf_{5}$, ${}^{20}pf_{7}$, ${}^{20}pf_{3}$, ${}^{20}pf_{8}$, ${}^{20}pf_{9}$, ${}^{20}pf_{6}$, ${}^{20}pf_{4}$, ${}^{20}pf_{2}$, ${}^{20}pf_{1}$}
TU^{7}	{${}^{7}pf_{1}$, ${}^{7}pf_{2}$, ${}^{7}pf_{3}$, ${}^{7}pf_{4}$, ${}^{7}pf_{5}$, ${}^{7}pf_{6}$, ${}^{7}pf_{7}$, ${}^{7}pf_{8}$, ${}^{7}pf_{9}$}	TU^{21}	{${}^{21}pf_{1}$, ${}^{21}pf_{2}$, ${}^{21}pf_{3}$, ${}^{21}pf_{4}$, ${}^{21}pf_{5}$, ${}^{21}pf_{6}$, ${}^{21}pf_{7}$, ${}^{21}pf_{8}$, ${}^{21}pf_{9}$}
TU^{8}	{${}^{8}pf_{2}$, ${}^{8}pf_{5}$, ${}^{8}pf_{7}$, ${}^{8}pf_{8}$, ${}^{8}pf_{9}$, ${}^{8}pf_{3}$, ${}^{8}pf_{1}$, ${}^{8}pf_{4}$, ${}^{8}pf_{6}$}	TU^{22}	{${}^{22}pf_{5}$, ${}^{22}pf_{8}$, ${}^{22}pf_{7}$, ${}^{22}pf_{4}$, ${}^{22}pf_{1}$, ${}^{22}pf_{6}$, ${}^{22}pf_{9}$, ${}^{22}pf_{3}$, ${}^{22}pf_{2}$}
TU^{9}	{${}^{9}pf_{2}$, ${}^{9}pf_{1}$, ${}^{9}pf_{3}$, ${}^{9}pf_{5}$, ${}^{9}pf_{8}$, ${}^{9}pf_{6}$, ${}^{9}pf_{9}$, ${}^{9}pf_{4}$, ${}^{9}pf_{7}$}	TU^{23}	{${}^{23}pf_{6}$, ${}^{23}pf_{9}$, ${}^{23}pf_{3}$, ${}^{23}pf_{5}$, ${}^{23}pf_{7}$, ${}^{23}pf_{8}$, ${}^{23}pf_{4}$, ${}^{23}pf_{2}$, ${}^{23}pf_{1}$}
TU^{10}	{${}^{10}pf_{1}$, ${}^{10}pf_{2}$, ${}^{10}pf_{3}$, ${}^{10}pf_{4}$, ${}^{10}pf_{5}$, ${}^{10}pf_{6}$, ${}^{10}pf_{7}$, ${}^{10}pf_{8}$, ${}^{10}pf_{9}$}	TU^{24}	{${}^{24}pf_{1}$, ${}^{24}pf_{2}$, ${}^{24}pf_{3}$, ${}^{24}pf_{4}$, ${}^{24}pf_{5}$, ${}^{24}pf_{6}$, ${}^{24}pf_{7}$, ${}^{24}pf_{8}$, ${}^{24}pf_{9}$}
TU^{11}	{${}^{11}pf_{3}$, ${}^{11}pf_{1}$, ${}^{11}pf_{2}$, ${}^{11}pf_{4}$, ${}^{11}pf_{6}$, ${}^{11}pf_{7}$, ${}^{11}pf_{5}$, ${}^{11}pf_{8}$, ${}^{11}pf_{9}$}	TU^{25}	{${}^{25}pf_{9}$, ${}^{25}pf_{6}$, ${}^{25}pf_{3}$, ${}^{25}pf_{5}$, ${}^{25}pf_{7}$, ${}^{25}pf_{2}$, ${}^{25}pf_{8}$, ${}^{25}pf_{4}$, ${}^{25}pf_{1}$}
TU^{12}	{${}^{12}pf_{1}$, ${}^{12}pf_{2}$, ${}^{12}pf_{3}$, ${}^{12}pf_{4}$, ${}^{12}pf_{5}$, ${}^{12}pf_{6}$, ${}^{12}pf_{7}$, ${}^{12}pf_{8}$, ${}^{12}pf_{9}$}	TU^{26}	{${}^{26}pf_{1}$, ${}^{26}pf_{2}$, ${}^{26}pf_{3}$, ${}^{26}pf_{4}$, ${}^{26}pf_{5}$, ${}^{26}pf_{6}$, ${}^{26}pf_{7}$, ${}^{26}pf_{8}$, ${}^{26}pf_{9}$}
TU^{13}	{${}^{13}pf_{1}$, ${}^{13}pf_{2}$, ${}^{13}pf_{3}$, ${}^{13}pf_{4}$, ${}^{13}pf_{5}$, ${}^{13}pf_{6}$, ${}^{13}pf_{7}$, ${}^{13}pf_{8}$, ${}^{13}pf_{9}$}	TU^{27}	{${}^{27}pf_{1}$, ${}^{27}pf_{2}$, ${}^{27}pf_{3}$, ${}^{27}pf_{4}$, ${}^{27}pf_{5}$, ${}^{27}pf_{6}$, ${}^{27}pf_{7}$, ${}^{27}pf_{8}$, ${}^{27}pf_{9}$}
TU^{14}	{${}^{14}pf_{1}$, ${}^{14}pf_{2}$, ${}^{14}pf_{3}$, ${}^{14}pf_{4}$, ${}^{14}pf_{5}$, ${}^{14}pf_{6}$, ${}^{14}pf_{7}$, ${}^{14}pf_{8}$, ${}^{14}pf_{9}$}	TU^{28}	{${}^{28}pf_{1}$, ${}^{28}pf_{2}$, ${}^{28}pf_{3}$, ${}^{28}pf_{4}$, ${}^{28}pf_{5}$, ${}^{28}pf_{6}$, ${}^{28}pf_{7}$, ${}^{28}pf_{8}$, ${}^{28}pf_{9}$}

Symbol: *pf*: point feature; TU: transient univariate; ${}^{j}pf_{i}$: ith *pf* of TU j; ^{Spfs}TU${}^{\#}$: Sorted *pfs* of TU based on SU

TABLE 4. Results of wrapper phase based on IWSS-SVM regarding different kernels per transient univariate.

Input	${}^{pOTFs}IWSS\text{-}SVM^{SGRBF}$: TSP Acc	${}^{pOTFs}IWSS\text{-}SVM^{DTW\text{-}GRBF}$: TSP Acc	${}^{pOTFs}IWSS\text{-}SVM^{REDK}$: TSP Acc
^{Spfs}TU^{1}	{${}^{1}pf_{7}$}: 87.80%	{${}^{1}pf_{7}$}: 87.80%	{${}^{1}pf_{7}$}: 80.48%
^{Spfs}TU^{2}	{${}^{2}pf_{1}$, ${}^{2}pf_{3}$, ${}^{2}pf_{4}$}: 87.80%	{${}^{2}pf_{1}$, ${}^{2}pf_{7}$}: 75.60%	{${}^{2}pf_{9}$, ${}^{2}pf_{1}$, ${}^{2}pf_{3}$, ${}^{2}pf_{4}$}: 87.80%
^{Spfs}TU^{3}	{${}^{3}pf_{1}$, ${}^{3}pf_{5}$, ${}^{3}pf_{6}$}: 90.24%	{${}^{3}pf_{1}$, ${}^{3}pf_{5}$, ${}^{3}pf_{6}$, ${}^{3}pf_{8}$}: 90.24%	{${}^{3}pf_{1}$, ${}^{3}pf_{3}$, ${}^{3}pf_{5}$}: 90.24%
^{Spfs}TU^{4}	{${}^{4}pf_{1}$, ${}^{4}pf_{2}$, ${}^{4}pf_{4}$}: 78.04%	{${}^{4}pf_{1}$, ${}^{4}pf_{4}$, ${}^{4}pf_{7}$, ${}^{4}pf_{9}$}: 78.04%	{${}^{4}pf_{3}$, ${}^{4}pf_{4}$, ${}^{4}pf_{9}$}: 70.73%
^{Spfs}TU^{5}	{${}^{5}pf_{1}$, ${}^{5}pf_{3}$, ${}^{5}pf_{4}$, ${}^{5}pf_{5}$, ${}^{5}pf_{6}$}: 90.24%	{${}^{5}pf_{6}$, ${}^{5}pf_{1}$, ${}^{5}pf_{2}$, ${}^{5}pf_{3}$}: 87.80%	{${}^{5}pf_{1}$, ${}^{5}pf_{4}$, ${}^{5}pf_{6}$}: 90.24%
^{Spfs}TU^{6}	{${}^{6}pf_{8}$}: 80.48%	{${}^{6}pf_{1}$, ${}^{6}pf_{8}$}: 73.17%	{${}^{6}pf_{8}$}: 80.48%
^{Spfs}TU^{7}	{${}^{7}pf_{1}$}: 87.80%	{${}^{7}pf_{1}$, ${}^{7}pf_{2}$}: 87.80%	{${}^{7}pf_{1}$}: 87.80%
^{Spfs}TU^{8}	{${}^{8}pf_{2}$, ${}^{8}pf_{5}$, ${}^{8}pf_{7}$, ${}^{8}pf_{8}$, ${}^{8}pf_{9}$}: 95.12%	{${}^{8}pf_{2}$, ${}^{8}pf_{3}$, ${}^{8}pf_{5}$, ${}^{8}pf_{8}$}: 92.68%	{${}^{8}pf_{6}$, ${}^{8}pf_{2}$, ${}^{8}pf_{5}$, ${}^{8}pf_{7}$, ${}^{8}pf_{9}$}: 95.12%
^{Spfs}TU^{9}	{${}^{9}pf_{2}$, ${}^{9}pf_{5}$, ${}^{9}pf_{8}$}: 58.53%	{${}^{9}pf_{2}$, ${}^{9}pf_{5}$}: 56.09%	{${}^{9}pf_{2}$, ${}^{9}pf_{5}$}: 58.53%
^{Spfs}TU^{10}	{${}^{10}pf_{9}$, ${}^{10}pf_{1}$, ${}^{10}pf_{2}$, ${}^{10}pf_{3}$, ${}^{10}pf_{6}$}: 90.24%	{${}^{10}pf_{1}$, ${}^{10}pf_{2}$, ${}^{10}pf_{3}$}: 85.36%	{${}^{10}pf_{1}$, ${}^{10}pf_{2}$, ${}^{10}pf_{3}$}: 85.36%
^{Spfs}TU^{11}	{${}^{11}pf_{1}$, ${}^{11}pf_{3}$, ${}^{11}pf_{4}$, ${}^{11}pf_{8}$}: 90.24%	{${}^{11}pf_{1}$, ${}^{11}pf_{3}$, ${}^{11}pf_{4}$, ${}^{11}pf_{6}$}: 87.80%	{${}^{11}pf_{1}$, ${}^{11}pf_{3}$, ${}^{11}pf_{4}$, ${}^{11}pf_{6}$}: 90.24%
^{Spfs}TU^{12}	{${}^{12}pf_{1}$, ${}^{12}pf_{2}$}: 92.68%	{${}^{12}pf_{1}$, ${}^{12}pf_{2}$, ${}^{12}pf_{4}$, ${}^{12}pf_{5}$, ${}^{12}pf_{6}$}: 82.92%	{${}^{12}pf_{1}$, ${}^{12}pf_{2}$, ${}^{12}pf_{3}$}: 90.24%
^{Spfs}TU^{13}	{${}^{13}pf_{1}$, ${}^{13}pf_{3}$, ${}^{13}pf_{4}$}: 85.36%	{${}^{13}pf_{1}$, ${}^{13}pf_{2}$, ${}^{13}pf_{3}$}: 85.36%	{${}^{13}pf_{1}$, ${}^{13}pf_{2}$, ${}^{13}pf_{3}$, ${}^{13}pf_{4}$, ${}^{13}pf_{5}$, ${}^{13}pf_{8}$}: 92.68%
^{Spfs}TU^{14}	{${}^{14}pf_{1}$, ${}^{14}pf_{2}$, ${}^{14}pf_{4}$, ${}^{14}pf_{5}$}: 92.68%	{${}^{14}pf_{1}$, ${}^{14}pf_{2}$, ${}^{14}pf_{3}$, ${}^{14}pf_{6}$}: 87.80%	{${}^{14}pf_{9}$, ${}^{14}pf_{8}$, ${}^{14}pf_{1}$, ${}^{14}pf_{2}$, ${}^{14}pf_{3}$, ${}^{14}pf_{5}$}: 97.56%
^{Spfs}TU^{15}	{${}^{15}pf_{2}$, ${}^{15}pf_{9}$}: 80.48%	{${}^{15}pf_{5}$, ${}^{15}pf_{6}$, ${}^{15}pf_{8}$, ${}^{15}pf_{9}$}: 70.73%	{${}^{15}pf_{2}$, ${}^{15}pf_{3}$, ${}^{15}pf_{9}$}: 75.60%
^{Spfs}TU^{16}	{${}^{16}pf_{1}$, ${}^{16}pf_{2}$, ${}^{16}pf_{3}$}: 82.92%	{${}^{16}pf_{1}$, ${}^{16}pf_{2}$, ${}^{16}pf_{4}$, ${}^{16}pf_{5}$}: 87.80%	{${}^{16}pf_{1}$, ${}^{16}pf_{2}$, ${}^{16}pf_{3}$, ${}^{16}pf_{4}$}: 85.36%
^{Spfs}TU^{17}	{${}^{17}pf_{2}$, ${}^{17}pf_{4}$, ${}^{17}pf_{5}$, ${}^{17}pf_{9}$}: 90.24%	{${}^{17}pf_{2}$, ${}^{17}pf_{9}$}: 82.92%	{${}^{17}pf_{3}$, ${}^{17}pf_{4}$, ${}^{17}pf_{5}$, ${}^{17}pf_{7}$, ${}^{17}pf_{9}$}: 90.24%
^{Spfs}TU^{18}	{${}^{18}pf_{9}$}: 73.17%	{${}^{18}pf_{1}$, ${}^{18}pf_{9}$}: 70.73%	{${}^{18}pf_{1}$, ${}^{18}pf_{6}$, ${}^{18}pf_{9}$}: 73.17%
^{Spfs}TU^{19}	{${}^{19}pf_{1}$, ${}^{19}pf_{3}$, ${}^{19}pf_{4}$, ${}^{19}pf_{5}$}: 90.24%	{${}^{19}pf_{1}$, ${}^{19}pf_{5}$, ${}^{19}pf_{7}$}: 80.48%	{${}^{19}pf_{1}$, ${}^{19}pf_{3}$, ${}^{19}pf_{4}$, ${}^{19}pf_{5}$}: 90.24%
^{Spfs}TU^{20}	{${}^{20}pf_{3}$, ${}^{20}pf_{5}$}: 78.04%	{${}^{20}pf_{5}$, ${}^{20}pf_{7}$}: 82.92%	{${}^{20}pf_{5}$, ${}^{20}pf_{6}$, ${}^{20}pf_{9}$}: 75.60%
^{Spfs}TU^{21}	{${}^{21}pf_{1}$, ${}^{21}pf_{6}$}: 73.17%	{${}^{21}pf_{1}$}: 68.29%	{${}^{21}pf_{1}$, ${}^{21}pf_{6}$}: 73.17%
^{Spfs}TU^{22}	{${}^{22}pf_{5}$, ${}^{22}pf_{8}$}: 92.68%	{${}^{22}pf_{3}$, ${}^{22}pf_{5}$, ${}^{22}pf_{6}$}: 95.12%	{${}^{22}pf_{5}$, ${}^{22}pf_{7}$, ${}^{22}pf_{8}$, ${}^{22}pf_{9}$}: 95.12%
^{Spfs}TU^{23}	{${}^{23}pf_{6}$}: 63.41%	{${}^{23}pf_{2}$, ${}^{23}pf_{6}$}: 65.85%	{${}^{23}pf_{6}$, ${}^{23}pf_{7}$, ${}^{23}pf_{9}$}: 63.41%
^{Spfs}TU^{24}	{${}^{24}pf_{1}$, ${}^{24}pf_{2}$, ${}^{24}pf_{3}$}: 85.36%	{${}^{24}pf_{1}$, ${}^{24}pf_{2}$, ${}^{24}pf_{3}$, ${}^{24}pf_{6}$}: 87.80%	{${}^{24}pf_{1}$, ${}^{24}pf_{2}$, ${}^{24}pf_{3}$}: 87.80%
^{Spfs}TU^{25}	{${}^{25}pf_{1}$, ${}^{25}pf_{2}$, ${}^{25}pf_{3}$, ${}^{25}pf_{9}$}: 95.12%	{${}^{25}pf_{2}$, ${}^{25}pf_{9}$}: 92.68%	{${}^{25}pf_{2}$, ${}^{25}pf_{3}$, ${}^{25}pf_{9}$}: 92.68%
^{Spfs}TU^{26}	{${}^{26}pf_{9}$, ${}^{26}pf_{1}$, ${}^{26}pf_{2}$, ${}^{26}pf_{3}$}: 80.48%	{${}^{26}pf_{1}$, ${}^{26}pf_{2}$, ${}^{26}pf_{3}$, ${}^{26}pf_{4}$, ${}^{26}pf_{6}$}: 78.04%	{${}^{26}pf_{1}$, ${}^{26}pf_{2}$, ${}^{26}pf_{3}$}: 78.04%
^{Spfs}TU^{27}	{${}^{27}pf_{1}$, ${}^{27}pf_{2}$}: 95.12%	{${}^{27}pf_{9}$, ${}^{27}pf_{1}$, ${}^{27}pf_{2}$, ${}^{27}pf_{4}$}: 97.56%	{${}^{27}pf_{1}$, ${}^{27}pf_{2}$, ${}^{27}pf_{4}$}: 95.12%
^{Spfs}TU^{28}	{${}^{28}pf_{9}$, ${}^{28}pf_{1}$, ${}^{28}pf_{2}$, ${}^{28}pf_{4}$, ${}^{28}pf_{6}$, ${}^{28}pf_{7}$}: 82.92%	{${}^{28}pf_{1}$, ${}^{28}pf_{2}$, ${}^{28}pf_{3}$, ${}^{28}pf_{4}$, ${}^{28}pf_{5}$}: 73.17%	{${}^{28}pf_{9}$, ${}^{28}pf_{1}$, ${}^{28}pf_{2}$}: 73.17%

Symbol: *pf*: point feature; TU: transient univariate; ^{Spfs}TU^{x}: sorted *pfs* of TU^{x}; ${}^{j}pf_{i}$: ith *pf* of TU j

SVM; ^{S}GRBF, LinKer, and PolKer in TWSVM) in the form of proposed IWMs (${}^{SU}_{Kernels}IWSS^{RR}_{SVM}$ and ${}^{SU}_{Kernels}IWSS^{RR}_{TWSVM}$ in IWSS^{2FWBs}; ${}^{SU}_{Kernels}IWSSr^{RR}_{SVM}$ and ${}^{SU}_{Kernels}IWSSr^{RR}_{TWSVM}$ in IWSSr^{2FWBs}), first, by regarding possible kernel-based permutation between two-state of IWSS^{2FWBs}(9 permutations), for each permutation of IWSS^{2FWBs}, the union of the selected preliminary OTFs (*p*OTFs) per the state of IWSS^{2FWBs} is recorded (See Fig. 2, left panel (LLP and RLP) of the cross-permutation box). In terms of IWSSr^{2FWBs}, the mentioned scenario is conducted to record union results per permutation in IWSSr^{2FWBs}, which is shown in the right panel (LRP and RRP) of the permutation box. Next, based on the cross-manner scenario, each left union is linked with right unions (9 links are established; e.g., 9 black links, 9 blue links,
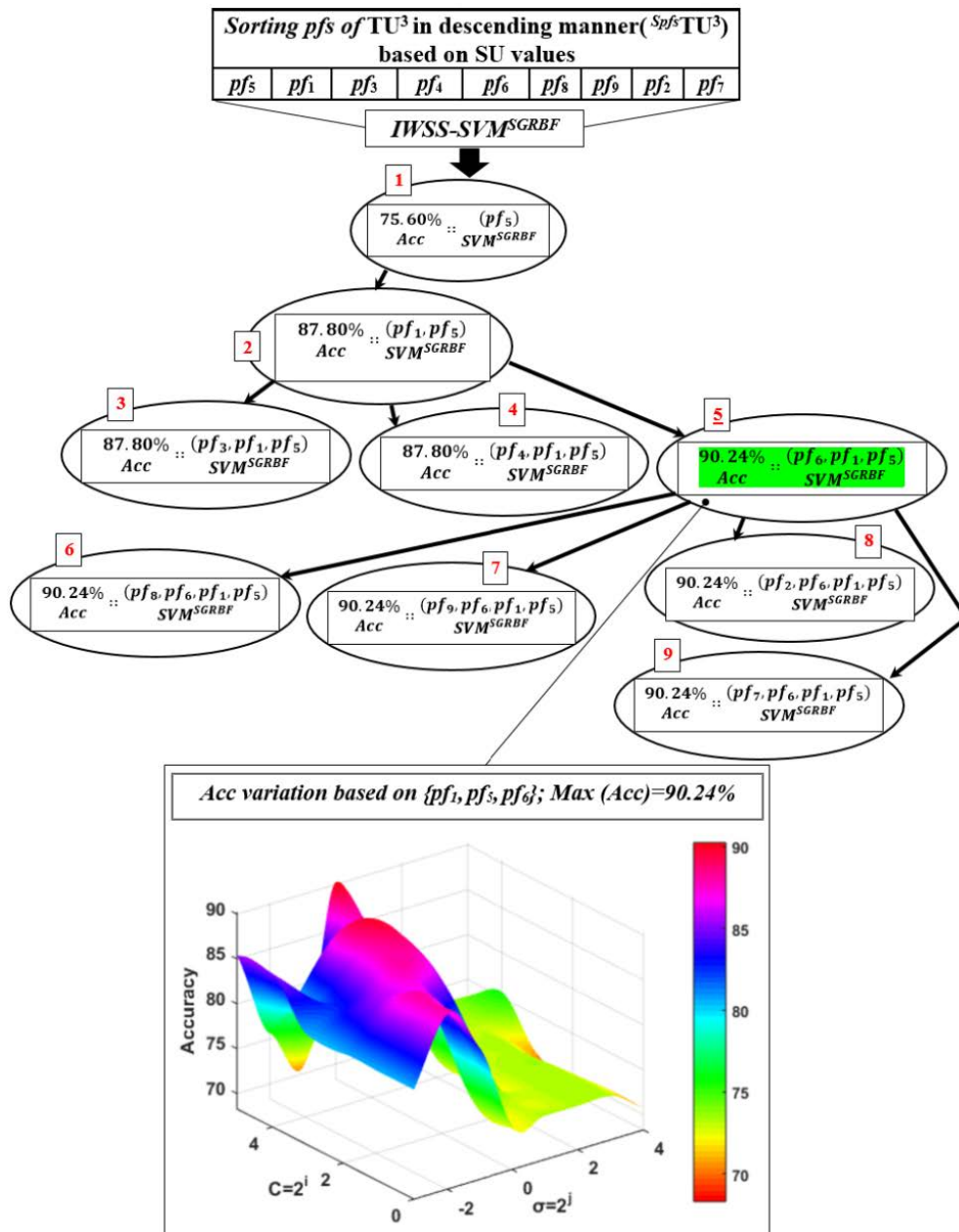
FIGURE 9. Structure of IWSS tree in selecting IWSS-SVM^{SGRBF}-based $^{POTFs}TU^3$ regarding Acc variations in optimal node (node 5).

and so on). Then, the intersection of both sides of the link in each colored 9 links (e.g., the intersection of both sides of the first black link) is calculated. After calculating the intersection of both sides of all colored links (e.g., both sides of the second black link to both sides of the ninthblack link), the union of sets is recorded (See Fig. 2), union sign in the polygonal black side). Such mechanism (intersection and union) is conducted on other colored 9 links (9 blue links, 9 yellow links, and so on). Finally, the intersection of the obtained sets in polygonal sides is considered as univariate-specific final OTFs (intersection of polygonal black side, polygonal brown side, and so on).

IV. EXPERIMENTAL DESIGN

A. TRANSIENT DATASET CONSTRUCTION

As can be seen in Fig. 1, transient dataset construction containing multivariate trajectories is considered as the first step of the proposed framework for FSS-based TSA. In this step, we executed contingency simulation based on the two-step transient dataset generation workflow (TDGW) proposed in [34], which is shown in Fig. 5. In TDGW, first, transient responses are extracted from output channels of basic features (OCBF-X; X indicated basic features namely bus voltages (VOLT), voltage phase angle (VANGLE), machine active power (PELEC), machine reactive power (QELEC),

TABLE 5. Results of wrapper phase based on IWSS-TWSVM regarding different kernels per transient univariate.

Input	$^{pOTFs}IWSS\text{-}TWSVM^{LinKer}$: TSP Acc	$^{pOTFs}IWSS\text{-}TWSVM^{PolKer}$: TSP Acc	$^{pOTFs}IWSS\text{-}TWSVM^{SGRBF}$: TSP Acc
$^{Spfs}TU^1$	$\{^1pf_1, ^1pf_7\}$: 78.04%	$\{^1pf_1, ^1pf_6, ^1pf_7\}$: 80.48%	$\{^1pf_7\}$: 60.97%
$^{Spfs}TU^2$	$\{^2pf_1, ^2pf_3, ^2pf_7\}$: 75.60%	$\{^2pf_9, ^2pf_1, ^2pf_3\}$: 75.60%	$\{^2pf_1, ^2pf_2, ^2pf_3, ^2pf_4\}$: 85.36%
$^{Spfs}TU^3$	$\{^3pf_3, ^3pf_5, ^3pf_8, ^3pf_9\}$: 80.48%	$\{^3pf_3, ^3pf_5, ^3pf_9\}$: 80.48%	$\{^3pf_7, ^3pf_1, ^3pf_3, ^3pf_5, ^3pf_6\}$: 90.24%
$^{Spfs}TU^4$	$\{^4pf_1, ^4pf_3, ^4pf_4, ^4pf_9\}$: 70.73%	$\{^4pf_3, ^4pf_4, ^4pf_9\}$: 65.85%	$\{^4pf_1, ^4pf_4\}$: 80.48%
$^{Spfs}TU^5$	$\{^5pf_9, ^5pf_1\}$: 68.29%	$\{^5pf_1, ^5pf_7\}$: 73.17%	$\{^5pf_1, ^5pf_2, ^5pf_5\}$: 87.80%
$^{Spfs}TU^6$	$\{^6pf_3, ^6pf_8\}$: 78.04%	$\{^6pf_8, ^6pf_9\}$: 80.48%	$\{^6pf_2, ^6pf_4, ^6pf_7, ^6pf_8\}$: 85.36%
$^{Spfs}TU^7$	$\{^7pf_1\}$: 87.80%	$\{^7pf_1\}$: 87.80%	$\{^7pf_1, ^7pf_5\}$: 87.80%
$^{Spfs}TU^8$	$\{^8pf_2\}$: 70.73%	$\{^8pf_2\}$: 78.04%	$\{^8pf_2, ^8pf_5, ^8pf_9\}$: 87.80%
$^{Spfs}TU^9$	$\{^9pf_2\}$: 58.53%	$\{^9pf_2\}$: 58.53%	$\{^9pf_2\}$: 56.09%
$^{Spfs}TU^{10}$	$\{^{10}pf_1\}$: 75.60%	$\{^{10}pf_1, ^{10}pf_6\}$: 78.04%	$\{^{10}pf_1\}$: 75.60%
$^{Spfs}TU^{11}$	$\{^{11}pf_3, ^{11}pf_6\}$: 56.09%	$\{^{11}pf_1, ^{11}pf_3, ^{11}pf_7\}$: 68.29%	$\{^{11}pf_1, ^{11}pf_2, ^{11}pf_3, ^{11}pf_4\}$: 85.36%
$^{Spfs}TU^{12}$	$\{^{12}pf_1, ^{12}pf_2\}$: 75.60%	$\{^{12}pf_1, ^{12}pf_5\}$: 78.04%	$\{^{12}pf_1, ^{12}pf_2\}$: 92.68%
$^{Spfs}TU^{13}$	$\{^{13}pf_1, ^{13}pf_3\}$: 56.09%	$\{^{13}pf_1, ^{13}pf_2, ^{13}pf_3, ^{13}pf_7\}$: 65.85%	$\{^{13}pf_1, ^{13}pf_4, ^{13}pf_5, ^{13}pf_6, ^{13}pf_7\}$: 87.80%
$^{Spfs}TU^{14}$	$\{^{14}pf_1\}$: 78.04%	$\{^{14}pf_1, ^{14}pf_3\}$: 78.04%	$\{^{14}pf_1, ^{14}pf_2, ^{14}pf_6\}$: 95.12%
$^{Spfs}TU^{15}$	$\{^{15}pf_4, ^{15}pf_5, ^{15}pf_8, ^{15}pf_9\}$: 58.53%	$\{^{15}pf_6, ^{15}pf_9\}$: 53.65%	$\{^{15}pf_4, ^{15}pf_3, ^{15}pf_9\}$: 65.85%
$^{Spfs}TU^{16}$	$\{^{16}pf_9, ^{16}pf_1, ^{16}pf_2\}$: 70.73%	$\{^{16}pf_1, ^{16}pf_2\}$: 75.60%	$\{^{16}pf_1\}$: 70.73%
$^{Spfs}TU^{17}$	$\{^{17}pf_6, ^{17}pf_9\}$: 80.48%	$\{^{17}pf_6, ^{17}pf_9\}$: 80.48%	$\{^{17}pf_4, ^{17}pf_9\}$: 90.24%
$^{Spfs}TU^{18}$	$\{^{18}pf_9\}$: 51.21%	$\{^{18}pf_4, ^{18}pf_6, ^{18}pf_9\}$: 73.17%	$\{^{18}pf_9\}$: 73.17%
$^{Spfs}TU^{19}$	$\{^{19}pf_1, ^{19}pf_3, ^{19}pf_4\}$: 80.48%	$\{^{19}pf_1, ^{19}pf_2, ^{19}pf_3, ^{19}pf_4\}$: 78.04%	$\{^{19}pf_1, ^{19}pf_2, ^{19}pf_3, ^{19}pf_4\}$: 87.80%
$^{Spfs}TU^{20}$	$\{^{20}pf_5\}$: 51.21%	$\{^{20}pf_5, ^{20}pf_6\}$: 75.60%	$\{^{20}pf_2, ^{20}pf_5, ^{20}pf_7\}$: 82.92%
$^{Spfs}TU^{21}$	$\{^{21}pf_1, ^{21}pf_3, ^{21}pf_6, ^{21}pf_7\}$: 75.60%	$\{^{21}pf_1\}$: 68.29%	$\{^{21}pf_1, ^{21}pf_3, ^{21}pf_5\}$: 82.92%
$^{Spfs}TU^{22}$	$\{^{22}pf_5, ^{22}pf_8\}$: 75.60%	$\{^{22}pf_5, ^{22}pf_7, ^{22}pf_8\}$: 75.60%	$\{^{22}pf_4, ^{22}pf_5\}$: 85.36%
$^{Spfs}TU^{23}$	$\{^{23}pf_3, ^{23}pf_5, ^{23}pf_6\}$: 63.41%	$\{^{23}pf_6, ^{23}pf_9\}$: 63.41%	$\{^{23}pf_1, ^{23}pf_6, ^{23}pf_9\}$: 58.53%
$^{Spfs}TU^{24}$	$\{^{24}pf_1, ^{24}pf_2\}$: 73.17%	$\{^{24}pf_1, ^{24}pf_3\}$: 78.04%	$\{^{24}pf_9, ^{24}pf_1, ^{24}pf_2, ^{24}pf_5\}$: 80.48%
$^{Spfs}TU^{25}$	$\{^{25}pf_2, ^{25}pf_9\}$: 60.97%	$\{^{25}pf_6, ^{25}pf_7, ^{25}pf_9\}$: 78.04%	$\{^{25}pf_8, ^{25}pf_9\}$: 87.80%
$^{Spfs}TU^{26}$	$\{^{26}pf_1, ^{26}pf_2, ^{26}pf_3\}$: 73.17%	$\{^{26}pf_1, ^{26}pf_2, ^{26}pf_5\}$: 75.60%	$\{^{26}pf_1, ^{26}pf_2, ^{26}pf_3, ^{26}pf_7\}$: 80.48%
$^{Spfs}TU^{27}$	$\{^{27}pf_1, ^{27}pf_4, ^{27}pf_6, ^{27}pf_8\}$: 60.97%	$\{^{27}pf_1, ^{27}pf_2, ^{27}pf_8\}$: 58.53%	$\{^{27}pf_1, ^{27}pf_2, ^{27}pf_4, ^{27}pf_5\}$: 90.24%
$^{Spfs}TU^{28}$	$\{^{28}pf_9, ^{28}pf_1, ^{28}pf_3, ^{28}pf_5\}$: 68.29%	$\{^{28}pf_1, ^{28}pf_4\}$: 53.65%	$\{^{28}pf_1, ^{28}pf_2\}$: 70.73%

Symbol: pf: point feature; TU: transient univariate; $^{Spfs}TU^x$: sorted pfs of TU^x; ^jpf_i: i^{th} pf of TU j

and reactive power consumption (QLOAD)). The application program interface (API) functions of the SIEMENS power system simulator for engineering (PSS/E) [35] in the form of Python-based automation file are used for exerting contingency simulation on the New England test system- New York power system (NETS-NYPS) (See Fig. 6) [36]. The generated transient samples are derived by substation outages, generator outages, and line outages where the fault duration time is set to 0.23 seconds (the time step is 0.0167 seconds). Also, the fault clearing time is set after the end of fault duration time. As an important uncertainty parameter in generating transient samples, the convert load (CONL) API is added to the PSS/E-Python automation file to regard different load characteristics which the percent of active and reactive power load to be converted to the constant current and constant admittance load characteristics [34], [36]. In the second step of TDGW, a set of Matlab-based commands cause adding factors to OCBF-X which leads to the defined 28 univariates trajectory features listed in Table 2 [37]–[39]. After conducting TDGW, the transient dataset contains 800 (No. transient samples) $\times$ 28 (No. features) $\times$ 9 (No. observed cycles)]. For example, some stable and unstable samples streamed based on F_{14} trajectory (The proportion of total QLOAD to total QELEC) is shown in Fig. 7.

B. FINAL OTFs (fOTFs) SET PER TRANSIENT UNIVARIATE (TU)

Extracting the transient univariate-specific final OTFs set called $fOTFs^{TU\#}$ (namely $fOTFs^{TU1}$ to $fOTFs^{TU28}$) by applying CPQHFSS on each univariate (TU^1 to TU^{28}) of transient multivariate trajectory dataset (TMTD) is elaborated in this section. According of Fig. 2, in the first step of CPQHFSS, each univariate of TMTD is entered to the filter-fixed phase of twin 2FWBs. According to (1), RR of point features per TU of TMTD is calculated based on SU measure. According to was what mentioned about the filter method in Section 2.B.1, for example, the obtained RR of pfs of TU^6 ($^{pfs}TU^6$; 9 cycles) based on SU is shown in Fig. 8. As can be seen in Fig. 8, based on SU values of $^{pfs}TU^6$, $\{pf_8\}$ is ranked as a high SU feature, and $\{pf_5\}$ is considered as the low SU pfs. Next, by sorting $^{pfs}TU^6$ based on SU values in descending manner ($^{Spfs}TU^6$), the order in which the $^{pfs}TU^6$ enter the wrapper phase in twin 2FWBs is specified (See Table 3; sixth row). For more clarity, Table 3 show the $^{Spfs}TU^1$ to $^{Spfs}TU^{28}$ based on SU values.

After calculating the SU amount of $^{pfs}TU^{1:28}$ and sorting the pfs of $TU^{1:28}$ ($^{Spfs}TU^{1:28}$), the $^{Spfs}TU^{1:28}$ are fed to the wrapper phase of twin 2FWBs in the form of $IWSS^{2FWBs}$ and $IWSSr^{2FWBs}$. In $IWSS^{2FWBs}$(the left

TABLE 6. Results of wrapper phase based on IWSSr-SVM regarding different kernels per transient univariate.

Input	$^{pOTFs}IWSSr\text{-}SVM^{SGRBF}$	$^{pOTFs}IWSSr\text{-}SVM^{DTW\text{-}GRBF}$	$^{pOTFs}IWSSr\text{-}SVM^{REDK}$
$^{Spfs}TU^1$	$\{^1pf_7\}$: 87.80%	$\{^1pf_7\}$: 87.80%	$\{^1pf_8\}$: 87.80%
$^{Spfs}TU^2$	$\{^2pf_5, ^2pf_1\}$: 92.68%	$\{^2pf_5\}$: 70.73%	$\{^2pf_4, ^2pf_1\}$: 90.24%
$^{Spfs}TU^3$	$\{^3pf_4, ^3pf_1\}$: 90.24%	$\{^3pf_9, ^3pf_6\}$: 92.68%	$\{^3pf_6, ^3pf_3, ^3pf_5\}$: 92.68%
$^{Spfs}TU^4$	$\{^4pf_2, ^4pf_5, ^4pf_3\}$: 78.04%	$\{^4pf_7, ^4pf_3, ^4pf_9, ^4pf_4\}$: 78.04%	$\{^4pf_2, ^4pf_5, ^4pf_4\}$: 70.73%
$^{Spfs}TU^5$	$\{^5pf_6, ^5pf_1, ^5pf_4\}$: 90.24%	$\{^5pf_9\}$: 78.04%	$\{^5pf_6, ^5pf_1\}$: 90.24%
$^{Spfs}TU^6$	$\{^6pf_8\}$: 80.48%	$\{^6pf_1\}$: 75.60%	$\{^6pf_3\}$: 82.92%
$^{Spfs}TU^7$	$\{^7pf_1\}$: 87.80%	$\{^7pf_2, ^7pf_1\}$: 87.80%	$\{^7pf_1\}$: 87.80%
$^{Spfs}TU^8$	$\{^8pf_3, ^8pf_7\}$: 95.12%	$\{^8pf_6, ^8pf_3, ^8pf_8, ^8pf_2\}$: 95.12%	$\{^8pf_6, ^8pf_3, ^8pf_7\}$: 97.56%
$^{Spfs}TU^9$	$\{^9pf_8, ^9pf_2\}$: 58.53%	$\{^9pf_5\}$: 56.09%	$\{^9pf_5, ^9pf_2\}$: 58.53%
$^{Spfs}TU^{10}$	$\{^{10}pf_7, ^{10}pf_5, ^{10}pf_1, ^{10}pf_2\}$: 92.68%	$\{^{10}pf_3, ^{10}pf_2, ^{10}pf_1\}$: 85.36%	$\{^{10}pf_3, ^{10}pf_1\}$: 85.36%
$^{Spfs}TU^{11}$	$\{^{11}pf_5, ^{11}pf_2\}$: 90.24%	$\{^{11}pf_6, ^{11}pf_3\}$: 90.24%	$\{^{11}pf_8, ^{11}pf_7, ^{11}pf_4\}$: 92.68%
$^{Spfs}TU^{12}$	$\{^{12}pf_2, ^{12}pf_1\}$: 92.68%	$\{^{12}pf_7, ^{12}pf_2\}$: 82.92%	$\{^{12}pf_5, ^{12}pf_4, ^{12}pf_2\}$: 90.24%
$^{Spfs}TU^{13}$	$\{^{13}pf_4, ^{13}pf_1\}$: 85.36%	$\{^{13}pf_3, ^{13}pf_2, ^{13}pf_1\}$: 85.36%	$\{^{13}pf_7, ^{13}pf_1\}$: 92.68%
$^{Spfs}TU^{14}$	$\{^{14}pf_3\}$: 92.68%	$\{^{14}pf_9, ^{14}pf_4\}$: 92.68%	$\{^{14}pf_3\}$: 92.68%
$^{Spfs}TU^{15}$	$\{^{15}pf_2, ^{15}pf_9\}$: 80.48%	$\{^{15}pf_4, ^{15}pf_3, ^{15}pf_9\}$: 80.48%	$\{^{15}pf_6, ^{15}pf_1\}$: 60.97%
$^{Spfs}TU^{16}$	$\{^{16}pf_3, ^{16}pf_2\}$: 87.80%	$\{^{16}pf_4\}$: 85.36%	$\{^{16}pf_4, ^{16}pf_3, ^{16}pf_2\}$: 87.80%
$^{Spfs}TU^{17}$	$\{^{17}pf_3, ^{17}pf_4\}$: 90.24%	$\{^{17}pf_2, ^{17}pf_6\}$: 82.92%	$\{^{17}pf_4\}$: 85.36%
$^{Spfs}TU^{18}$	$\{^{18}pf_9\}$: 73.17%	$\{^{18}pf_6, ^{18}pf_9\}$: 73.17%	$\{^{18}pf_8, ^{18}pf_9\}$: 73.17%
$^{Spfs}TU^{19}$	$\{^{19}pf_4\}$: 90.24%	$\{^{19}pf_4\}$: 80.48%	$\{^{19}pf_5, ^{19}pf_1\}$: 90.24%
$^{Spfs}TU^{20}$	$\{^{20}pf_7\}$: 75.60%	$\{^{20}pf_7, ^{20}pf_5\}$: 82.92%	$\{^{20}pf_9\}$: 73.17%
$^{Spfs}TU^{21}$	$\{^{21}pf_6, ^{21}pf_1\}$: 73.17%	$\{^{21}pf_1\}$: 68.29%	$\{^{21}pf_6, ^{21}pf_1\}$: 73.17%
$^{Spfs}TU^{22}$	$\{^{22}pf_9, ^{22}pf_5\}$: 95.12%	$\{^{22}pf_3, ^{22}pf_9\}$: 95.12%	$\{^{22}pf_1, ^{22}pf_7, ^{22}pf_5\}$: 95.12%
$^{Spfs}TU^{23}$	$\{^{23}pf_6\}$: 63.41%	$\{^{23}pf_3\}$: 63.41%	$\{^{23}pf_2, ^{23}pf_9\}$: 87.80%
$^{Spfs}TU^{24}$	$\{^{24}pf_3, ^{24}pf_2\}$: 87.80%	$\{^{24}pf_9, ^{24}pf_1, ^{24}pf_3\}$: 90.24%	$\{^{24}pf_2, ^{24}pf_9\}$: 87.80%
$^{Spfs}TU^{25}$	$\{^{25}pf_1, ^{25}pf_2, ^{25}pf_9\}$: 95.12%	$\{^{25}pf_1, ^{25}pf_2, ^{25}pf_9\}$: 95.12%	$\{^{25}pf_2, ^{25}pf_9\}$: 87.80%
$^{Spfs}TU^{26}$	$\{^{26}pf_8, ^{26}pf_7, ^{26}pf_2, ^{26}pf_3\}$: 87.80%	$\{^{26}pf_5, ^{26}pf_3\}$: 78.04%	$\{^{26}pf_7, ^{26}pf_3\}$: 78.04%
$^{Spfs}TU^{27}$	$\{^{27}pf_2, ^{27}pf_1\}$: 95.12%	$\{^{27}pf_5, ^{27}pf_1, ^{27}pf_4\}$: 97.56%	$\{^{27}pf_5, ^{27}pf_4, ^{27}pf_1\}$: 97.56%
$^{Spfs}TU^{28}$	$\{^{28}pf_4, ^{28}pf_3\}$: 82.92%	$\{^{28}pf_6, ^{28}pf_3\}$: 82.92%	$\{^{28}pf_5, ^{28}pf_3\}$: 82.92%

Symbol: pf: point feature; TU: transient univariate; $^{Spfs}TU^x$: sorted pfs of TU^x; ^jpf_i: i^{th} pf of TU j

rectangle box in Fig. 2), the first hybrid block including SVM classifier ($_{SVM}IWSS_{HB1}^{2FWBs}$) accompanied with ^SGRBF, $^{DTW}GRBF$, and REDK kernels. Also, the second hybrid block of $IWSS^{2FWBs}$ includes the TWSVM classifier ($_{TWSVM}IWSS_{HB2}^{2FWBs}$) with LinKer, PolKer, and ^SGRBF kernels. In terms of $IWSSr^{2FWBs}$ (the right rectangle box in Fig. 2), the $^{Spfs}TU^{1:28}$ entered into $_{SVM}IWSSr_{HB1}^{2FWBs}$ (kernels: ^SGRBF, $^{DTW}GRBF$, and REDK) and $_{TWSVM}IWSSr_{HB2}^{2FWBs}$ (kernels: LinKer, PolKer and ^SGRBF).

After conducting the wrapper phase of CPQHFSS in the form $IWSS^{2FWBs}$ and $IWSSr^{2FWBs}$, the obtained results are shown in Table 4 to Table 7. Table 4 show the obtained preliminary OTFs ($pOTFs$) of TU^x ($^{pOTFs}TU^x$) based on IWSS-SVM regarding SVM-specific kernels in the form of $^{pOTFs}IWSS\text{-}SVM^{SGRBF}$, $^{pOTFs}IWSS\text{-}SVM^{DTW-GRBF}$, and $^{pOTFs}IWSS\text{-}SVM^{REDK}$ columns. Also, Table 5 show $^{pOTFs}TU^x$ based on IWSS-TWSVM regarding TWSVM-specific kernels in the form of $^{pOTFs}IWSS\text{-}TWSVM^{LinKer}$, $^{pOTFs}IWSS\text{-}TWSVM^{PolKer}$, and $^{pOTFs}IWSS\text{-}TWSVM^{SGRBF}$ columns. In terms of IWSSr-based results, Table 6 and Table 7, show the $^{pOTFs}TU^x$ based on IWSSr-SVM ($^{pOTFs}IWSSr\text{-}SVM^{SGRBF}$, $^{pOTFs}IWSSr\text{-}SVM^{DTW-GRBF}$, $^{pOTFs}IWSSr\text{-}SVM^{REDK}$) and IWSSr-TWSVM ($^{pOTFs}IWSSr\text{-}TWSVM^{LinKer}$, $^{pOTFs}IWSSr\text{-}TWSVM^{PolKer}$, $^{pOTFs}IWSSr\text{-}TWSVM^{SGRBF}$) respectively. For example, after applying the wrapper phase of CPQHFSS on $^{Spfs}TU^3$, the IWSS-

based $^{pOTFs}TU^3$ (See third row of Table 4 and Table 5) and IWSSr-based $^{pOTFs}TU^3$ (See third row of Table 6 and Table 7) are extracted. By applying IWSS-SVM^{SGRBF}, IWSS-SVM$^{DTW-GRBF}$, and IWSS-SVM^{REDK} on $^{Spfs}TU^3$, the ^{pOTFs}IWSS-SVM^{SGRBF}($\{pf_1, pf_5, pf_6\}$: 90.24%), ^{pOTFs}IWSS-SVM$^{DTW-GRBF}$($\{pf_1, pf_5, pf_6, pf_8\}$: 90.24%), and ^{pOTFs}IWSS-SVM^{REDK} ($\{pf_1, pf_3, pf_5\}$: 90.24%) are obtained, respectively. Also, by conducing IWSS-TWSVM^{LinKer}, IWSS-TWSVM^{PolKer}, and IWSS-TWSVM^{SGRBF} on $^{Spfs}TU^3$, the ^{pOTFs}IWSS-TWSVM^{LinKer}($\{pf_3, pf_5, pf_8, pf_9\}$: 80.48%), ^{pOTFs}IWSS-TWSVM^{PolKer}($\{pf_3, pf_5, pf_9\}$: 80.48%), and ^{pOTFs}IWSS-TWSVM^{SGRBF} ($\{pf_7, pf_1, pf_3, pf_5, pf_6\}$: 90.24%) are obtained, respectively. In the case of $^{Spfs}UT^3$-specific $pOTFs$ based on IWSSr-oriented wrapper phase, the ^{pOTFs}IWSSr-SVM^{SGRBF} is $\{pf_4, pf_1\}$: 90.24%, ^{pOTFs}IWSSr-SVM$^{DTW-GRBF}$ is $\{pf_9, pf_6\}$: 92.68%, and ^{pOTFs}IWSSr-SVM^{REDK} is $\{pf_6, pf_3, pf_5\}$: 92.68%. Also, the ^{pOTFs}IWSSr-TWSVM^{LinKer} is $\{pf_7, pf_3, pf_9\}$: 85.36%, ^{pOTFs}IWSSr-TWSVM^{PolKer} is $\{pf_7, pf_9\}$: 85.36%, and ^{pOTFs}IWSSr-TWSVM^{SGRBF} is $\{pf_8, pf_1, pf_6\}$: 90.20%. For example, the IWSS-oriented tree is depicted in Fig. 9 show how IWSS-SVM^{SGRBF}-based$^{pOTFs}TU^3$ are selected in IWSS iterations. Also, the Acc metric (22) measured the performance of $^{SVM}IWSS^{TWSVM}$ and $^{SVM}IWSSr^{TWSVM}$. Furthermore, the fine-tuning on learning parameters (C and σ) in each iteration of IWMs is considered for TTP which for the SVM and

TABLE 7. Results of wrapper phase based on IWSSr-TWSVM regarding different kernels per transient univariate.

Input	$_{p\text{OTFs}}IWSSr\text{-}TWSVM^{LinKer}$	$_{p\text{OTFs}}IWSSr\text{-}TWSVM^{PolKer}$	$_{p\text{OTFs}}IWSSr\text{-}TWSVM^{SGRBF}$
$^{Spfs}TU^1$	{$^1pf_1$, $^1pf_7$}: 78.04%	{$^1pf_6$, $^1pf_1$, $^1pf_7$}: 80.48%	{$^1pf_4$, $^1pf_6$}: 68.29%
$^{Spfs}TU^2$	{$^2pf_7$, $^2pf_3$, $^2pf_1$}: 73.17%	{$^2pf_8$, $^2pf_1$}: 75.60%	{$^2pf_5$, $^2pf_1$}: 90.24%
$^{Spfs}TU^3$	{$^3pf_7$, $^3pf_3$, $^3pf_9$}: 85.36%	{$^3pf_7$, $^3pf_9$}: 85.36%	{$^3pf_8$, $^3pf_1$, $^3pf_6$}: 90.20%
$^{Spfs}TU^4$	{$^4pf_3$, $^4pf_1$, $^4pf_9$, $^4pf_4$}: 70.73%	{$^4pf_5$, $^4pf_9$}: 73.17%	{$^4pf_1$, $^4pf_4$}: 80.48%
$^{Spfs}TU^5$	{$^5pf_6$, $^5pf_2$}: 82.92%	{$^5pf_9$, $^5pf_7$, $^5pf_6$}: 85.36%	{$^5pf_6$, $^5pf_1$}: 90.24%
$^{Spfs}TU^6$	{$^6pf_3$, $^6pf_8$}: 78.04%	{$^6pf_9$, $^6pf_8$}: 80.48%	{$^6pf_1$, $^6pf_2$, $^6pf_6$}: 90.24%
$^{Spfs}TU^7$	{$^7pf_1$}: 87.80%	{$^7pf_1$}: 87.80%	{$^7pf_2$, $^7pf_5$}: 87.80%
$^{Spfs}TU^8$	{$^8pf_2$}: 70.73%	{$^8pf_5$}: 80.48%	{$^8pf_8$, $^8pf_7$}: 92.68%
$^{Spfs}TU^9$	{$^9pf_2$}: 58.53%	{$^9pf_2$}: 58.53%	{$^9pf_1$}: 58.53%
$^{Spfs}TU^{10}$	{$^{10}pf_3$}: 80.48%	{$^{10}pf_3$}: 80.48%	{$^{10}pf_9$}: 80.48%
$^{Spfs}TU^{11}$	{$^{11}pf_2$, $^{11}pf_1$}: 58.53%	{$^{11}pf_7$, $^{11}pf_1$, $^{11}pf_3$}: 68.29%	{$^{11}pf_2$, $^{11}pf_3$}: 85.36%
$^{Spfs}TU^{12}$	{$^{12}pf_2$}: 75.60%	{$^{12}pf_5$, $^{12}pf_1$}: 78.04%	{$^{12}pf_2$, $^{12}pf_1$}: 92.68%
$^{Spfs}TU^{13}$	{$^{13}pf_3$}: 58.53%	{$^{13}pf_6$, $^{13}pf_5$}: 68.29%	{$^{13}pf_8$, $^{13}pf_4$}: 87.80%
$^{Spfs}TU^{14}$	{$^{14}pf_1$}: 78.04%	{$^{14}pf_8$, $^{14}pf_3$}: 85.36%	{$^{14}pf_4$}: 92.68%
$^{Spfs}TU^{15}$	{$^{15}pf_1$}: 53.65%	{$^{15}pf_6$, $^{15}pf_9$}: 53.65%	{$^{15}pf_4$, $^{15}pf_3$, $^{15}pf_9$}: 65.85%
$^{Spfs}TU^{16}$	{$^{16}pf_3$}: 75.60%	{$^{16}pf_6$, $^{16}pf_3$}: 85.36%	{$^{16}pf_2$}: 82.92%
$^{Spfs}TU^{17}$	{$^{17}pf_5$}: 80.48%	{$^{17}pf_3$, $^{17}pf_4$}: 85.36%	{$^{17}pf_4$}: 90.24%
$^{Spfs}TU^{18}$	{$^{18}pf_9$}: 51.21%	{$^{18}pf_8$, $^{18}pf_1$}: 58.53%	{$^{18}pf_9$}: 73.17%
$^{Spfs}TU^{19}$	{$^{19}pf_5$}: 80.48%	{$^{19}pf_9$, $^{19}pf_4$}: 85.36%	{$^{19}pf_6$, $^{19}pf_5$}: 92.68%
$^{Spfs}TU^{20}$	{$^{20}pf_5$}: 51.21%	{$^{20}pf_6$, $^{20}pf_5$}: 75.60%	{$^{20}pf_2$, $^{20}pf_7$, $^{20}pf_5$}: 82.92%
$^{Spfs}TU^{21}$	{$^{21}pf_3$, $^{21}pf_1$}: 65.85%	{$^{21}pf_1$}: 68.29%	{$^{21}pf_5$, $^{21}pf_1$}: 85.36%
$^{Spfs}TU^{22}$	{$^{22}pf_8$, $^{22}pf_5$}: 75.60%	{$^{22}pf_7$, $^{22}pf_8$, $^{22}pf_5$}: 75.60%	{$^{22}pf_6$, $^{22}pf_1$, $^{22}pf_4$}: 90.24%
$^{Spfs}TU^{23}$	{$^{23}pf_5$, $^{23}pf_3$, $^{23}pf_6$}: 63.41%	{$^{23}pf_9$, $^{23}pf_6$}: 63.41%	{$^{23}pf_9$}: 60.97%
$^{Spfs}TU^{24}$	{$^{24}pf_3$, $^{24}pf_2$}: 78.04%	{$^{24}pf_3$}: 75.60%	{$^{24}pf_1$, $^{24}pf_2$}: 78.04%
$^{Spfs}TU^{25}$	{$^{25}pf_4$}: 56.09%	{$^{25}pf_5$, $^{25}pf_6$}: 80.48%	{$^{25}pf_8$, $^{25}pf_9$}: 87.80%
$^{Spfs}TU^{26}$	{$^{26}pf_3$, $^{26}pf_2$, $^{26}pf_1$}: 73.17%	{$^{26}pf_3$, $^{26}pf_1$}: 75.60%	{$^{26}pf_8$, $^{26}pf_3$, $^{26}pf_4$}: 90.24%
$^{Spfs}TU^{27}$	{$^{27}pf_4$, $^{27}pf_3$, $^{27}pf_2$}: 65.85%	{$^{27}pf_9$, $^{27}pf_7$, $^{27}pf_5$, $^{27}pf_1$}: 80.48%	{$^{27}pf_9$, $^{27}pf_8$}: 90.24%
$^{Spfs}TU^{28}$	{$^{28}pf_8$, $^{28}pf_3$}: 82.92%	{$^{28}pf_4$}: 73.17%	{$^{28}pf_4$, $^{28}pf_1$}: 75.60%

Symbol: *pf*: point feature; TU: transient univariate; $^{Spfs}TU^x$: sorted *pfs* of TU^x; ^jpf_i: i^{th} *pf* of TU j

TABLE 8. The obtained final OTFs per univariate (fOTFs$^{TU1:TU28}$) and union of fOTFs$^{TU1:TU28}$ based on cross-permutation phase of CPQHFSS.

Input	Permutation: LP	Permutation: RP	Cross	fOTFs^{TU1}
$_{pOTFs}TU^1$	LLP1 U RLP1= {$^1pf_7$} LLP2 U RLP2= {$^1pf_1$, $^1pf_6$, $^1pf_7$} LLP3 U RLP3= {$^1pf_1$, $^1pf_7$} LLP4 U RLP4= {$^1pf_7$} LLP5 U RLP5= {$^1pf_1$, $^1pf_6$, $^1pf_7$} LLP6 U RLP6= {$^1pf_1$, $^1pf_7$} LLP7 U RLP7= {$^1pf_7$} LLP8 U RLP8= {$^1pf_1$, $^1pf_6$, $^1pf_7$} LLP9 U RLP9= {$^1pf_1$, $^1pf_7$}	LLP1 U RLP1= {$^1pf_4$, $^1pf_6$, $^1pf_7$} LLP2 U RLP2= {$^1pf_1$, $^1pf_6$, $^1pf_7$} LLP3 U RLP3= {$^1pf_1$, $^1pf_7$} LLP4 U RLP4= {$^1pf_4$, $^1pf_6$, $^1pf_7$} LLP5 U RLP5= {$^1pf_1$, $^1pf_6$, $^1pf_7$} LLP6 U RLP6= {$^1pf_1$, $^1pf_7$} LLP7 U RLP7= {$^1pf_4$, $^1pf_6$, $^1pf_7$} LLP8 U RLP8= {$^1pf_1$, $^1pf_6$, $^1pf_7$} LLP9 U RLP9= {$^1pf_1$, $^1pf_7$}	polygonal black side= {$^1pf_7$} polygonal dark blue side= {$^1pf_1$, $^1pf_6$, $^1pf_7$} polygonal brown side = {$^1pf_1$, $^1pf_7$} polygonal light green side= {$^1pf_7$} polygonal pink side= {$^1pf_1$, $^1pf_6$, $^1pf_7$} polygonal red side= {$^1pf_1$, $^1pf_7$} polygonal light blue side= {$^1pf_7$} polygonal dark green side= {$^1pf_1$, $^1pf_6$, $^1pf_7$} polygonal yellow side= {$^1pf_1$, $^1pf_7$}	{$^1pf_7$}

Input	fOTFs$^{TU2:TU10}$	Input	fOTFs$^{TU11:TU119}$	Input	fOTFs$^{TU20:TU28}$
$_{pOTFs}TU^2$	{$^2pf_1$, $^2pf_5$}	$_{pOTFs}TU^{11}$	{$^{11}pf_1$, $^{11}pf_3$, $^{11}pf_4$}	$_{pOTFs}TU^{20}$	{$^{20}pf_5$}
$_{pOTFs}TU^3$	{$^3pf_1$, $^3pf_3$, $^3pf_5$}	$_{pOTFs}TU^{12}$	{$^{12}pf_1$, $^{12}pf_2$}	$_{pOTFs}TU^{21}$	{$^{21}pf_1$}
$_{pOTFs}TU^4$	{$^4pf_4$}	$_{pOTFs}TU^{13}$	{$^{13}pf_1$, $^{13}pf_3$}	$_{pOTFs}TU^{22}$	{$^{22}pf_5$}
$_{pOTFs}TU^5$	{$^5pf_1$}	$_{pOTFs}TU^{14}$	{$^{14}pf_1$, $^{14}pf_3$}	$_{pOTFs}TU^{23}$	{$^{23}pf_6$}
$_{pOTFs}TU^6$	{$^6pf_8$}	$_{pOTFs}TU^{15}$	{$^{15}pf_9$}	$_{pOTFs}TU^{24}$	{$^{24}pf_1$, $^{24}pf_2$, $^{24}pf_3$}
$_{pOTFs}TU^7$	{$^7pf_1$}	$_{pOTFs}TU^{16}$	{$^{16}pf_2$}	$_{pOTFs}TU^{25}$	{$^{25}pf_2$, $^{25}pf_9$}
$_{pOTFs}TU^8$	{$^8pf_2$, $^8pf_5$}	$_{pOTFs}TU^{17}$	{$^{17}pf_9$}	$_{pOTFs}TU^{26}$	{$^{26}pf_1$, $^{26}pf_2$, $^{26}pf_3$}
$_{pOTFs}TU^9$	{$^9pf_2$, $^9pf_5$}	$_{pOTFs}TU^{18}$	{$^{18}pf_9$}	$_{pOTFs}TU^{27}$	{$^{27}pf_1$, $^{27}pf_2$}
$_{pOTFs}TU^{10}$	{$^{10}pf_1$, $^{10}pf_2$, $^{10}pf_3$}	$_{pOTFs}TU^{19}$	{$^{19}pf_1$, $^{19}pf_4$, $^{19}pf_5$}	$_{pOTFs}TU^{28}$	{$^{28}pf_1$}

Union of fOTFs$^{TU1:TU28}$ (UfOTFs$^{TU1:TU28}$)

{$^1pf_7$, $^2pf_{1,3}$, $^3pf_{1,3,5}$, $^4pf_4$, $^5pf_1$, $^6pf_8$, $^7pf_1$, $^8pf_{2,5}$, $^9pf_{2,5}$, $^{10}pf_{1,2,3}$, $^{11}pf_{1,3,4}$, $^{12}pf_{1,2}$, $^{13}pf_{1,3}$, $^{14}pf_{1,3}$, $^{15}pf_9$, $^{16}pf_2$, $^{17}pf_9$, $^{18}pf_9$, $^{19}pf_{1,4,5}$, $^{20}pf_5$, $^{21}pf_1$, $^{22}pf_5$, $^{23}pf_6$, $^{24}pf_{1,2,3}$, $^{25}pf_{2,9}$, $^{26}pf_{1,2,3}$, $^{27}pf_{1,2}$, $^{28}pf_1$}

Symbol: *pf*: point feature; TU: transient univariate; $^{pOTFs}TU^x$: survived preliminary optimal transient features of TU^x by wrapper phase; ^jpf_i: i^{th} *pf* of TU j, U: union

(Fold 2)

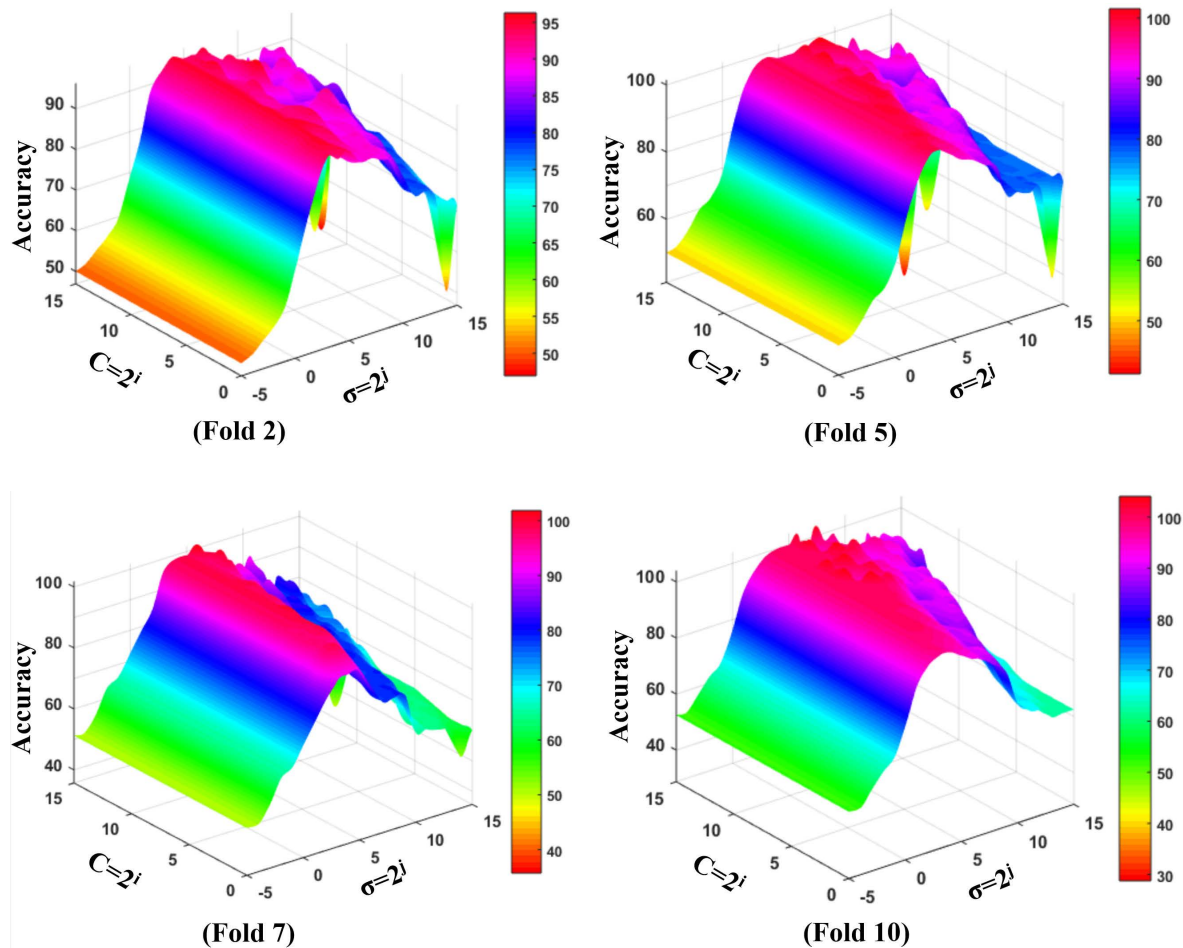**(Fold 5)**

(Fold 7)

(Fold 10)

FIGURE 10. Acc variations based on learning parameters in some folds (fold^{2,5,7,10}) for TSP based on UfOTFs$^{TU1:TU28}$.

TWSVM classifiers and their plugged kernels as (23). In the case of IWSS-SVM^{SGRBF}(See Fig. 9), in each iteration, the maximum value of the Acc (retrieved by optimal pair of learning parameters) is recorded. As can be seen in Fig. 9, the ^{p}OTFsTU^3 based on IWSS-SVM^{SGRBF} is {pf_1, pf_5, pf_6}, which Acc variations related to optimal iteration (node 5: green-face) is depicted in 3-D plot.

$$Accuracy(Acc) = (TP + TN)\big/(TP + TN + FP + FN)$$

$$\begin{cases} P : stable\ sample; & T : predicted\ correctly \\ N : unstable\ sample; & F : predictedin\ correctly \end{cases} \quad (22)$$

$$SVM\ (plugged\ kernel) \begin{cases} C = 2^i | i = 0, 1, \ldots, 5 \\ \sigma = 2^j | j = -3, -2, \ldots, 4 \end{cases}$$

$TWSVM\ (plugged\ kernel)$

$$\begin{cases} C = 2^i | i = 0, 1, \ldots, 5 \\ \sigma = 2^j | j = -3, -2, \ldots, 4 \\ \sigma = 2^j | j = 0, 1, .., 4\ (PolKer) \end{cases} \quad (23)$$

After selecting ^{SVM}IWSS^{3ker}-based three packages-pOTFs and ^{TWSVM}IWSS^{3ker}-based three packages-pOTFs

TABLE 9. The evaluation metrics.

Metrics	Formula
Accuracy	Acc=(TP+TN)/(TP+TN+FP+FN)
Sensitivity	TPR=TP/(TP+FN)
Specificity	TNR=TN/(TN+FP)

(totally six packages pOTFs-IWSS); and ^{SVM}IWSSr^{3ker}-based three packages-pOTFs and ^{TWSVM}IWSSr^{3ker}–based three packages-pOTFs (totally six packages pOTFs-IWSSr) per TU^x, as the final phase of CPQHFSS which is shown in Fig. 2, the cross-permutation scenario is conducted on obtained 12 packages of pOTFs per TU^x for extracting fOTFs$^{TU1:TU28}$. Finally, based on fOTFs$^{TU1:TU28}$, the union of fOTFs$^{TU1:TU28}$ (UfOTFs$^{TU1:TU28}$) is obtained as the final optimal transient feature set for use in TSP. The obtained fOTFs$^{TU1:TU28}$ and UfOTFs$^{TU1:TU28}$ based on the cross-permutation phase of CPQHFSS is shown in Table 8. In Table 8, the details of obtaining the optimal transient

TABLE 10. Results of tsp based on UfOTFs$^{TU1:TU28}$ set.

Classifier	Test case	10-fold cross validation			
		Max(Acc.) per fold based on fine-tuning on C and σ Accuracy [TPR / TNR]			
		fold 1	**fold 2**	**fold 3**	**fold 4**
		98.75 [97.5 / 100]	95 [95 / 95]	97.5 [95 / 100]	98.75 [100 / 97.5]
		fold 5	**fold 6**	**fold 7**	**fold 8**
SVM(^{s}GRBF)	NETS-NYPS	100 [100 / 100]	100 [100 / 100]	100 [100 / 100]	100 [100 / 100]
		fold 9		**fold 10**	
		98.75 [97.5 / 100]		100 [100 / 100]	
		Mean(measure) of folds: Accuracy [TPR / TNR]			
		98.87 [98. 5 / 99.25]			

features of TU^1 (fOTFs^{TU1}) based on cross-permutation are elaborated.

C. TSP BASED ON UNION OF fOTFs$^{TU1:TU28}$ (UfOTFs$^{TU1:TU28}$) SET

After selecting the univariate-specific fOTFs set for 28-variate trajectories dataset namely fOTFs^{TU1} to fOTFs^{TU28}(fOTFs$^{TU1:TU28}$) based on CPQHFSS, evaluating the efficacy of union of fOTFs$^{TU1:TU28}$(UfOTFs$^{TU1:TU28}$) (See Table 8, last row) in achieving high-performance TSP is considered in this section. The UfOTFs$^{TU1:TU28}$-oriented TTP are based on the 10-fold cross-validation technique that in each fold, the SVM^{sGRBF} is used as a learning model. Also, fine-tuning of the learning parameters for pair (C, σ) in SVM^{sGRBF} is adjusted from $\{C = 2^i | i = 0, 1, \ldots, 15\}$ and $\{\sigma = 2^j | j = -5, -4, \ldots, 15\}$ to find an optimal value for pair (C, σ) in the TTP per fold. In such an experimental design, the performance evaluation of the class labeling by SVM^{sGRBF} in the presence of UfOTFs$^{TU1:TU28}$ is measured by classification metrics which are shown in Table 9.

By regarding the above-mentioned points in TTP for TSP, the value of triple indices in TSP (Acc, TPR, and TNR) per fold is shown in Table 10. Based on setting the different value for learning parameters, the fluctuations of the Acc index per fold is recorded, and these values are inserted into Table 10. For example, the Acc variation of fold^2, fold^5, fold^7, and fold^{10} are depicted in Fig. 10. Furthermore, the Table 10 is accompanied by TPR and TNR indices, which is commensurate with the maximum value of Acc in each fold. After extracting all the results based on the evaluation indices, as the overall result of employing the UfOTFs$^{TU1:TU28}$ in SVM^{sGRBF} for TSP, calculating the average of the Acc, TPR, and TNR is regarded in Table 10. The results of Table 10 show the fact that the UfOTFs$^{TU1:TU28}$ set plays the pivot role in raising the capacity of prediction (high DLA) in TSA (Acc: 98.87 %, TPR: 98.5 %, and TNR: 99.25 %). Besides the importance of DLA, we concentrated on the DLT factor (including observed time and prediction time) in the performance evaluation of proposed framework in TSA. For DLT calculation, we need to focus

TABLE 11. ODLT for TSP based on UfOTFs$^{TU1:TU28}$ set.

Observed Time (OT) in cycle / second	Data Labeling Time (DLT) (OT + prediction time)
9 / 0.1503	150.3 ms + 2.225 ms= 152.525 ms

on selected point features UfOTFs$^{TU1:TU28}$(See Table 8). The optimal features in the UfOTFs$^{TU1:TU28}$ set show the fact the selected cycles of some univariates are related to the 9th cycle. (e.g., fOTFs^{TU15}, fOTFs^{TU17}, fOTFs^{TU18}, and fOTFs^{TU25}). Hence, observed time (OT) is equal to 9 cycles (150.3 milliseconds (ms)), and the prediction time based on SVM^{sGRBF}-UfOTFs$^{TU1:TU28}$ is 2.225 ms. Consequently, the DLT is 152.525 ms (See Table 11), indicating a low DLT to address corrective control actions.

D. COMPARISON OF EXPERIMENTAL METHODS: CPQHFSS VS. VERTICALLY UNILATERAL FSSs (VUFSSs) AND MHFSS

For further evaluation on the efficacy of CPQHFSS-based UfOTFs$^{TU1:TU28}$ in TSP, the comparison between with three VUFSSs^{OTFs} (3VUFSSs^{OTFs}) is considered in this section. The 3VUFSSs includes mRMR [16], [17], ReliefF [18] and fast correlation-based filter (FCBF) [19]. Also, we compared the CPQHFSS with the bi-mode hybrid feature selection scheme (BMHFSS) [23] and partial-injective trilateral hybrid (filter-wrapper) scheme called PITHS [34] as the MHFSS. By applying the 28-variate time-series data to 3VUFSSs and 2MHFSS, the 3VUFSSs-based OTFs and 2MHFSS-based OTFs are extracted. Next, the 3VUFSSs^{OTFs} and 2MHFSS^{OTFs} are entered into the SVM^{sGRBF} learning model in the same learning conditions expressed in subsection C of section 3 (10-fold cross-validation technique and the fine-tuning of the learning parameters).

As can be seen in Table 12, CPQHFSS^{OTFs} have better performance in TSP than 3VUFSSs^{OTFs} and 2MHFSS^{OTFs} (ignoring only 0.25% less than TPR than PITHS). The obtained results of Table 12 show that CPQHFSS in the presence of 48-cycles of 28-variate trajectory (See Table 8)

TABLE 12. Results of TSP based on selected OTFS by 3VUFSS and 2MHFSS.

(FSS) Classifier	Test case	10-fold cross validation			
		Max(Acc.) per fold based on fine-tuning on C and σ Accuracy [TPR / TNR]			
(mRMR) SVM(^{S}GRBF)	NETS-NYPS	fold 1	fold 2	fold 3	fold 4
		93.75 [97.5 / 90]	93.75 [97.5 / 90]	90 [95 / 85]	90 [82.5 / 97.5]
		fold 5	fold 6	fold 7	fold 8
		95 [97.5 / 92.5]	95 [95 / 95]	95 [100 / 90]	91.25 [87.5 / 95]
		fold 9		fold 10	
		88.75 [95 / 82.5]		93.75 [92.5 / 95]	
		Mean(measure) of folds: Accuracy [TPR / TNR]			
		92.62 [94 / 91.25]			
(FCBF) SVM(^{S}GRBF)	NETS-NYPS	fold 1	fold 2	fold 3	fold 4
		98.75 [97.5 / 100]	95 [95 / 95]	96.25 [92.5 / 100]	97.5 [95 / 100]
		fold 5	fold 6	fold 7	fold 8
		96.25 [92.5 / 100]	97.5 [97.5 / 97.5]	97.5 [95 / 100]	97.5 [97.5 / 97.5]
		fold 9		fold 10	
		97.5 [95 / 100]		100 [100 / 100]	
		Mean(measure) of folds: Accuracy [TPR / TNR]			
		97.37 [95.75 / 99]			
(ReliefF) SVM(^{S}GRBF)	NETS-NYPS	fold 1	fold 2	fold 3	fold 4
		98.75 [97.5 / 100]	95 [95 / 95]	96.25 [92.5 / 100]	97.5 [95 / 100]
		fold 5	fold 6	fold 7	fold 8
		96.25 [92.5 / 100]	97.5 [97.5 / 97.5]	97.5 [95 / 100]	97.5 [97.5 / 97.5]
		fold 9		fold 10	
		97.5 [95 / 100]		100 [100 / 100]	
		Mean(measure) of folds: Accuracy [TPR / TNR]			
		97.37 [95.75 / 99]			
(BMHFSS) SVM(^{S}GRBF)	NETS-NYPS	fold 1	fold 2	fold 3	fold 4
		100 [100 / 100]	97.5 [100 / 95]	96.25 [92.5 / 100]	95 [92.5 / 97.5]
		fold 5	fold 6	fold 7	fold 8
		100 [100 / 100]	100 [100 / 100]	97.5 [97.5 / 97.5]	98.75 [97.5 / 100]
		fold 9		fold 10	
		97.5 [97.5 / 97.5]		100 [100 / 100]	
		Mean(measure) of folds: Accuracy [TPR / TNR]			
		98.25 [97.75 / 98.75]			
(PITHS) SVM(^{S}GRBF)	NETS-NYPS	fold 1	fold 2	fold 3	fold 4
		98.75 [97.5 / 100]	96.25 [100 / 92.5]	97.5 [95 / 100]	98.75 [100 / 97.5]
		fold 5	fold 6	fold 7	fold 8
		100 [100 / 100]	100 [100 / 100]	98.75 [97.5 / 100]	100 [100 / 100]
		fold 9		fold 10	
		97.5 [97.5 / 97.5]		100 [100 / 100]	
		Mean(measure) of folds: Accuracy [TPR / TNR]			
		98.75 [98.75 / 98.75]			

has better performance (Acc, TPR, and TNR) than mRMR (9-cycle of 4-variate trajectory features), FCBF, ReliefF, and BMHFSS (9-cycle of 3-variate trajectory features [23]. In terms of the PITHS method, the selected cycles via

TABLE 13. ODLT for TSP based on 3VUFSSs and 2MHFSS.

3VUFSSs & 2MHFSS	Observed Time (OT) in cycle / second	DLT (OT + prediction time)
mRMR	4 / 0.0668	66.8 ms + 1.993 ms= 68.793 ms
FCBF	4 / 0.0668	66.8 ms + 2.130 ms= 68.930 ms
ReliefF	4 / 0.0668	66.8 ms + 2.110 ms= 68.910 ms
BMHFSS	3 / 0.0501	50.1 ms + 2.848 ms= 52.948 ms
PITHS	9 / 0.1503	150.3 ms + 2.291 ms= 152.591 ms

CPQHFSS leading to high Acc and TPR than PITHS (24-cycle of 18-variate trajectory features [34]). In terms of DLT, CPQHFSS has higher DLT (152.525 ms) than 3VUFSSs (mRMR: 68.793 ms, FCBF: 68.930 ms, ReliefF: 68.910 ms) and MHFSS$^{\text{BMHFSS}}$ (BMHFSS: 52.948 ms). Also, based on DLT index, CPQHFSS has lower DLT than MHFSS$^{\text{PITHS}}$ (PITHS: 152.591 ms) (See Table 13). Generally, the CPQHFSS$^{\text{DLT}}$ causes the system operator to have enough time to take corrective actions. The detail information about CPQHFSS$^{\text{DLT}}$, 3VUFSSs$^{\text{DLT}}$, and 2MHFSS$^{\text{DLT}}$ are shown in Table 11 and Table 13.

V. CONCLUSION

The presence of tightly correlated indices in the TSA problem, namely, accuracy and time of transient prediction, extracting optimal transient features (OTFs) from HDTF space via multifaceted FSS to meet low DLT and high DLA, was defined as the main agenda of this paper. To this end, we proposed cross-permutation-based quad-hybrid FSS (CPQHFSS) designed by integrating four filter-wrapper blocks (FWBs) in the form of twin two-FWBs mounted on dual incremental wrapper mechanisms (IWMs) called IWSS$^{\text{2FWBs}}$ and IWSSr$^{\text{2FWBs}}$. Regarding filter-fixed and wrapper-varied approaches (F^{f}W^v) embedded into IWMs$^{\text{2FWBs}}$ of CPQHFSS, the relevancy ratio-support vector machine (RR-SVM) raised as F^{f}W^v in the first block of IWSS$^{\text{2FWBs}}$ and IWSSr$^{\text{2FWBs}}$, and relevancy ratio-twin support vector machine (RR-TWSVM) situated in the second block of IWSS$^{\text{2FWBs}}$ and IWSSr$^{\text{2FWBs}}$. Furthermore, the nonlinear nature of transient feature space makes the plugging elastic and non-elastic kernels into the wrapper phase an integral part of CPQHFSS. Hence, by selecting the triple nonlinear kernel, cross-permutations of hybrid FSS are applied on HDTF to select OTFs. After applying CPQHFSS on transient multivariate trajectories, selected optimal features are fed to the cross-validation scenario to evaluate their effectiveness on TSA. Experimental results show that the selected OTFs based on the CPQHFSS have a high performance (Acc 98.87%, TPR 98.5%, TNR 99.25%, and DLT of 152.525 ms) for TSP. For more clarity, the performance of CPQHFSS compared with 3VUFSSs and 2MHFSS. The obtained results show that the selected CPQHFSS-based OTFS have better performance than selected OTFs by mRMR, ReliefF, FCBF, BMHFSS, and PITHS algorithms on TSP.

In future transient studies, we evaluate the performance of learning models in TSA by focusing on the environmental conditions of power systems that negatively affect the quality of the transient dataset. Selecting the most relevant features from the contaminated transient dataset via filter-wrapper strategy mounted on the convolutional neural network can be considered as one of the most significant solutions to achieve timely-accurate TSA in the presence of missing and noisy data.

REFERENCES

[1] S. Gao, S. Qiu, Z. Ma, R. Tian, and Y. Liu, "SVAE-WGAN-based soft sensor data supplement method for process industry," *IEEE Sensors J.*, vol. 22, no. 1, pp. 601–610, Jan. 2022.

[2] Z. You and L. Feng, "Integration of Industry 4.0 related technologies in construction industry: A framework of cyber-physical system," *IEEE Access*, vol. 8, pp. 122908–122922, 2020.

[3] M. I. Alabsi and A. Q. Gill, "A review of passenger digital information privacy concerns in smart airports," *IEEE Access*, vol. 9, pp. 33769–33781, 2021.

[4] L. Nardy, O. Pinheiro, and H. Lepikson, "Computer system integrated with digital models for reconstruction of underwater structures with high definition," *IEEE Latin Amer. Trans.*, vol. 20, no. 2, pp. 283–290, Feb. 2022.

[5] M. A. A. Sufyan, M. Zuhaib, M. A. Anees, A. Khair, and M. Rihan, "Implementation of PMU-based distributed wide area monitoring in smart grid," *IEEE Access*, vol. 9, pp. 140768–140778, 2021.

[6] J. Han, M. Kamber, and J. Peripei, *Data Mining: Concepts and Techniques*, 3rd ed. Burlington, MA, USA: Morgan Kaufmann, 2011.

[7] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed. New York, NY, USA: Springer, 2009.

[8] G. V. Trunk, "A problem of dimensionality: A simple example," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-1, no. 3, pp. 306–307, Jul. 1979.

[9] K. Koutroumbas and S. Theodoridis, *Pattern Recognition*, 1st ed. New York, NY, USA: Academic, 2008.

[10] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning*, 1st ed. New York, NY, USA: Springer, 2013.

[11] V. Bolón-Canedo, N. Sánchez-Maroño, and A. Alonso-Betanzos, *Feature Selection for High-Dimensional Data*, 1st ed. Cham, Switzerland: Springer, 2016.

[12] S. Das, S. P. Singh, and B. K. Panigrahi, "Transmission line fault detection and location using wide area measurements," *Electr. Power Syst. Res.*, vol. 151, pp. 96–105, Oct. 2017.

[13] P. Kundur, J. Paserba, V. Ajjarapu, G. Andersson, A. Bose, C. Canizares, N. Hatziargyriou, D. Hill, A. Stankovic, C. Taylor, T. Van Cutsem, and V. Vittal, "Definition and classification of power system stability IEEE/CIGRE joint task force on stability terms and definition," *IEEE Trans. Power Syst.*, vol. 19, no. 3, pp. 1387–1401, Aug. 2004.

[14] C.-S.-G. Karavas, K. A. Plakas, K. F. Krommydas, A. S. Kurashvili, C. N. Dikaiakos, and G. P. Papaioannou, "A review of wide-area monitoring and damping control systems in Europe," in *Proc. IEEE Madrid PowerTech*, Madrid, Spain, Jun. 2021, pp. 1–6.

[15] M. Pavella, M. Ernest, and D. Ruiz-Vega, *Transient Stability of Power Systems: A Unified Approach to Assessment and Control*, 1st ed. New York, NY, USA: Springer, 2000.

[16] X. Li, Z. Zheng, L. Wu, R. Li, J. Huang, X. Hu, and P. Guo, "A stratified method for large-scale power system transient stability assessment based on maximum relevance minimum redundancy arithmetic," *IEEE Access*, vol. 7, pp. 61414–61432, 2019.

[17] J. Liu, H. Sun, Y. Li, W. Fang, and S. Niu, "An improved power system transient stability prediction model based on mRMR feature selection and WTA ensemble learning," *Appl. Sci.*, vol. 10, no. 7, p. 2255, Mar. 2020.

[18] A. Stief, J. R. Ottewill, and J. Baranowski, "Relief F-based feature ranking and feature selection for monitoring induction motors," in *Proc. 23rd Int. Conf. Autom. Robot. (MMAR)*, Miedzyzdroje, Poland, Aug. 2018, pp. 171–176.

[19] J. Yan, C. Li, and Y. Liu, "Deep learning based total transfer capability calculation model," in *Proc. Int. Conf. Power Syst. Technol. (POWERCON)*, Guangzhou, China, Nov. 2018, pp. 952–957.

[20] S. Liu, L. Liu, Y. Fan, L. Zhang, Y. Huang, T. Zhang, J. Cheng, L. Wang, M. Zhang, R. Shi, and D. Mao, "An integrated scheme for online dynamic security assessment based on partial mutual information and iterated random forest," *IEEE Trans. Smart Grid*, vol. 11, no. 4, pp. 3606–3619, Jul. 2020.

[21] L. Ji, J. Wu, Y. Zhou, and L. Hao, "Using trajectory clusters to define the most relevant features for transient stability prediction based on machine learning method," *Energies*, vol. 9, no. 11, p. 898, Nov. 2016.

[22] Z. Chen, X. Han, C. Fan, T. Zheng, and S. Mei, "A two-stage feature selection method for power system transient stability status prediction," *Energies*, vol. 12, no. 4, p. 689, Feb. 2019.

[23] S. A. B. Mosavi, "Extracting most discriminative features on transient multivariate time series by bi-mode hybrid feature selection scheme for transient stability prediction," *IEEE Access*, vol. 9, pp. 121087–121110, 2021.

[24] P. Bermejo, J. A. Gamez, and J. M. Puerta, "Incremental wrapper-based subset selection with replacement: An advantageous alternative to sequential forward selection," in *Proc. IEEE Symp. Comput. Intell. Data Mining*, Nashville, TN, USA, Mar. 2009, pp. 367–374.

[25] D. Tomar and S. Agarwal, "Twin support vector machine," *Egyptian Informat. J.*, vol. 16, no. 1, pp. 55–69, Mar. 2015.

[26] R. Ruiz, J. C. Riquelme, and J. S. Aguilar-Ruiz, "Incremental wrapper-based gene selection from microarray data for cancer classification," *Pattern Recognit.*, vol. 39, no. 12, pp. 2383–2392, Dec. 2006.

[27] I. H. Witten, E. Frank, and M. A. Hall, *Data Mining: Practical Machine Learning Tools and Techniques*. Burlington, MA, USA: Morgan Kaufmann, 2011.

[28] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, pp. 273–297, Apr. 1995.

[29] O. L. Mangasarian and E. W. Wild, "Multisurface proximal support vector machine classification via generalized eigenvalues," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 1, pp. 69–74, Jan. 2006.

[30] Jayadeva, R. Khemchandani, and S. Chandra, "Twin support vector machines for pattern classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 5, pp. 905–910, May 2007. [Online]. Available: https://ieeexplore.ieee.org/document/4135685/authors#authors and http://ctech.iitd.ac.in/jayadeva.html

[31] C. M. Bishop, *Pattern Recognition and Machine Learning*, 1st ed. New York, NY, USA: Springer, 2006.

[32] H. Shimodaira, K.-I. Noma, M. Nakai, and S. Sagayama, "Support vector machine with dynamic time-alignment kernel for speech recognition," in *Proc. Eur. Conf. Speech Commun. Technol. (EUROSPEECH)*, Aalborg, Denmark, 2001, pp. 1–4.

[33] P.-F. Marteau and S. Gibet, "On recursive edit distance kernels with application to time series classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 6, pp. 1121–1133, Jun. 2015.

[34] S. A. B. Mosavi, "Finding optimal point features in transient multivariate excursions by horizontally integrated trilateral hybrid feature selection scheme for transient analysis," *IEEE Access*, vol. 9, pp. 163297–163315, 2021.

[35] Siemens Industry, Schenectady, NY, USA. (2013). *Siemens Power Technologies International, PSS/E 33.4 Application Program Interface (API)*. [Online]. Available: https://www.siemens.com/power-technologies

[36] C. Canizares, T. Fernandes, E. Geraldi, L. Gérin-Lajoie, M. Gibbard, I. Hiskens, J. Kersulis, R. Kuiava, L. Lima, F. de Marco, N. Martins, B. Pal, A. Piardi, R. Ramos, J. dos Santos, D. Silva, A. Singh, B. Tamimi, and D. Vowles, "Benchmark systems for small-signal stability analysis and control," IEEE Power Energy Soc., Piscataway, NJ, USA, Tech. Rep. PESTR18, Aug. 2015.

[37] A. B. Mosavi, A. Amiri, and H. Hosseini, "A learning framework for size and type independent transient stability prediction of power system using twin convolutional support vector machine," *IEEE Access*, vol. 6, pp. 69937–69947, 2018.

[38] S. A. B. Mosavi, "Extracting discriminative features in reactive power variations by 1-persistence parallel fragmented hybrid feature selection scheme for transient stability prediction," *Int. J. Intell. Eng. Syst.*, vol. 14, no. 4, pp. 500–513, Aug. 2021.

[39] S. A. B. Mosavi and S. Y. Banihashem, "Defining geometric cross-relevance multivariate trajectory features for transient stability analysis based on elastic kernel support vector machine," *Int. J. Intell. Eng. Syst.*, vol. 14, no. 5, pp. 369–385, Oct. 2021.

SEYED ALIREZA BASHIRI MOSAVI received the B.Sc. degree in computer engineering-software from the Azad University of Zanjan, Zanjan, Iran, in 2011, the M.Sc. degree in information technology engineering-electronic commerce from the University of Qom, Qom, Iran, in 2013, and the Ph.D. degree in computer engineering-artificial intelligence and robotics from the University of Zanjan, Zanjan, in 2019. He is currently an Assistant Professor with the Department of Electrical and Computer Engineering, Buein Zahra Technical University, Buein Zahra, Qazvin, Iran. His master's thesis on customer value analysis won the Tejarat Bank Award as the best thesis related to quality. His doctoral thesis on designing a novel learning framework for size and type independent transient stability prediction of the power systems. He has published several papers and conferences in the field of customer relationship management (CRM), transient processes in power systems, and machine learning scope. His main research interests include CRM, data mining, pattern recognition, and transient analysis based on machine learning methods.

• • •