

Received October 15, 2021, accepted December 4, 2021, date of publication December 6, 2021, date of current version December 17, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3133465

Finding Optimal Point Features in Transient Multivariate Excursions by Horizontally Integrated Trilateral Hybrid Feature Selection Scheme for Transient Analysis

SEYED ALIREZA BASHIRI MOSAVI 

Imam Khomeini International University-Buin Zahra Higher Education Center of Engineering and Technology, Buin Zahra, Qazvin 3414896818, Iran

e-mail: a.bashiri@bzeng.ikiu.ac.ir


ABSTRACT In transient analysis (TA), the processing time (PT) and prediction accuracy (PA) are the most significant indices be influenced the decision-making of grid operators to conduct timely-accurate corrective actions. In fact, achieving low PT and high PA (high-performance TA) necessitates designing the comprehensive feature selection scheme to select optimal transient point features (OTPFs). Hence, the partial-injective trilateral hybrid (filter-wrapper) scheme called PITHS is introduced in this paper. First, the transient dataset in the form of multivariate time series is constructed by an integrated programming platform. Next, based on PITHS, the first univariate trajectory feature (UTF^1) is entered into the nested trilateral filter phase (NTFP) equipped with intertwined triple criteria of information theory for selecting filter-OTPFs of UTF^1 (f^{-1} OTPFs). Then, f^{-1} OTPFs are fed to the nested trilateral wrapper phase (NTWP) for selecting filter-wrapper- 1 OTPFs (fw^{-1} OTPFs). The NTWP is including the hyperplane-based predictive approach accompanied by the triple kernel. After conducting NTWP, fw^{-1} OTPFs are considered as the first ultimate optimal point features (1 UOPFs). Next, survived fw^{-1} OTPFs injected into the subsequent trajectory (UTF^2), and the neo-formed trajectory (fw^{-1} OTPFs plus UTF^2) drives a new round of NTFP and NTWP for finding fw^{-2} OTPFs (2 UOPFs). By conducting this procedure on the last neo-formed trajectory (the fw^{-k-1} OTPFs + UTF^k), the fw^{-k} OTPFs are obtained (k UOPFs). Finally, the $^1:k$ UOPFs set is tested to verify their efficacy for TA based on the cross-validation technique. The obtained results show that the proposed framework has a prediction accuracy of 98.75% and a processing time of 152.591 milliseconds for TA.

INDEX TERMS Hybrid feature selection scheme, optimal transient point features (OTPFs), support vector machine (SVM), transient stability assessment (TSA).

I. INTRODUCTION

Emerging wide-area monitoring systems (WAMS) like phasor measurement units (PMUs) caused soft-hard restructuring in grid monitoring platforms, which has a direct impact on the precise reliability assessment of power systems [1]. In fact, using the PMU-based monitoring dashboard depicts the real-time variations of dynamic responses for grid operators, which helps them to promote awareness of the synchrony degree of the power system components [2]. In this regard, one of the most significant concerns in the power system exploitation process is related to the importance

of maintaining synchronism of power system components under severe and sudden disturbances, called transient stability [3]. Hence, transient stability assessment (TSA) to identify unstable conditions by mining on the dynamic characteristics of system variables is a vital task. By detecting the transient instability via TSA, makes it possible to take corrective control action for the secure-adequate power supply. However, such actions provide the potential opportunity to keep normal operations, if conducted in a timely manner. An important point to note is that timely corrective action requires the fast detection of transient stability status (stable or unstable case), which is achieved through applying robust lightweight predictive data mining (DM) techniques on a small observation window (SOW) of transient

The associate editor coordinating the review of this manuscript and approving it for publication was Amedeo Andreotti .

features [4], [5]. Taking into cognizance these points, the processing time of transient stability prediction (TSP), which is included prediction and observation time must be less than one second (<1 s) [6]. Besides the pay attention to the time complexity of the classifier for labeling transient samples, SOW plays a major role in reducing the processing time of TSP. However, lack of consideration to the intrinsic characteristics of existing transient features in picked SOW negatively affected the training and testing procedure of classification techniques, which leads to low accuracy on TSP. In other words, selecting the most relevant transient features in the form of the best-laid SOW is the necessary concern to achieve high-performance (time and accuracy) on TSP. This challenge can be solved via conducting the feature selection process, which is known as the most prominent category in DM technology. Hence, studies on designing feature selection scheme (FSS) have become an interesting research topic in the field of TSA and pattern recognition in recent years. Generally, FSS-centric transient studies fall under two categories: 1) Filter method; in Reference [7], ReliefF-based feature selection algorithm is used to select the most discriminative features for diagnosing fault of induction motors. Also, dynamic stability features are selected by the minimum-redundancy and maximum-relevance (mRMR) for large-scale power systems transient stability assessment in [8], [9]. In Reference [10], regarding transient stability constraints, a feature pre-screening strategy for selecting optimal features based on the fast correlation-based filter method (FCBF) is used to achieve a high-performance total transfer capability calculation model, and 2) filter-wrapper method; In Reference [11], the optimal trajectory cluster features are exploited from the large observations of rotor angle and voltage magnitude by conducting hybrid framework in the form of the Relief-support vector machine for TSP. Also, extracting the most relevant features by conducting a segment-oriented filter-wrapper method on reactive power-based two-variate time series has been considered for TSP in [12]. Also, in [13], the bi-mode hybrid feature selection scheme (BMHFSS) finds optimal transient features on multivariate time series by coupling the point and trajectory-based filter-wrapper scheme. Focusing on the structure of the above-mentioned classical FSS shows the fact these approaches were designed based on vertically integrated strategy. For example, in vertical-oriented hybrid FSS, first, entire feature space [13] or fragmented feature space (fragment¹, fragment² and so on) [12] is entered into information theory-based approaches (filter), and then selected primary optimal features are fed to predictive-based algorithms (wrapper) for finding final optimal features. Such a cohesive-mode strategy leads to the extraction of the intrinsic characteristic of some transient multivariate time series as optimal features. Although, the selected optimal features based on vertical mode induced high accuracy in TSP, relevant features of some trajectories (called optimal-blurred features) are overshadowed based on this strategy. In fact, such an approach may cause the loss of the

discriminative transient features per univariate trajectory and negatively affect the TSP performance in the presence of severe transient space. Hence, designing the horizontally integrated hybrid FSS, which considers all trajectories in the generalized form, is essential for selecting the most discriminative features. Applying the proposed hybrid FSS based on the partial-injective scenario makes the opportunity to find optimal-blurred features as best-laid SOW.

According to what was mentioned above, designing the comprehensive FSS based on the proper strategy is one of the main solutions to achieve key indices, namely processing time (PT) and prediction accuracy (PA) on TSA. In fact, achieving low PT and high PA based on the most discriminative transient features in the presence of the severe transient space provides the necessary context to conduct timely-accurate corrective actions in power systems. To this end, as can be seen in Fig. 1, we consider the three-step scenario for TSA, which is included: 1) transient dataset generation based on transient simulation on the New England-New York interconnection (NETS-NYPS) test case was considered in the first step, 2) Next, the partial-injective trilateral hybrid scheme (PITHS) based on horizontally integrated mode is applied on transient multivariate trajectory features (TMTFs) which consist of two nested trilateral phases: a) nested trilateral filter phase (NTFP); TMTFs is entered into the NTFP equipped with intertwined triple information theory criteria for selecting filter-optimal transient point features (f -OTPFs) and b) in the nested trilateral wrapper phase (NTWP); the f -OTPFs is entered into the hyperplane-based predictive approach based on the triple kernel to find the filter-wrapper optimal transient point features (f_w -OTPFs), and 3) in the final step, performance evaluation on TSP based on selected ultimate optimal point features (UOPFs) is considered by cross-validation technique.

The rest of the paper is organized as follows: we describe the structure of the PITHS in Section 2. Experimental results of the proposed framework are presented in Section 3. Finally, the conclusion is interpreted in Section 4.

II. OVERALL PITHS PROCEDURE

The conjoined steps of PITHS to select the best-laid SOW in TMTFs for TSA are depicted in Fig. 2. Have a glance at the structure of the proposed scheme show the fact that the PITHS implements a partial-injective policy on transient multivariate excursions during the feature selection process. In fact, f_w^{-1} -OTPFs of the first univariate trajectory feature (UTF¹) survived by conducting the first round of dual-phase of PITHS (partial-manner) are considered as first ultimate optimal point features (¹UOPFs) and then accompanied by subsequent univariate trajectory feature (UTF²) (injection) for exerting new rounds of dual-phase for finding f_w^{-2} -OTPFs (²UOPFs). This horizontal integration will continue until the last neo-formed trajectory (LnfT) (retrieved from combination f_w^{-k-1} -OTPFs and UTF^k) is obtained. Next, LnfT is entered into the final dual-phase of PITHS to select f_w^{-k} -OTPFs as ^kUOPFs. Regardless of the partial-injective

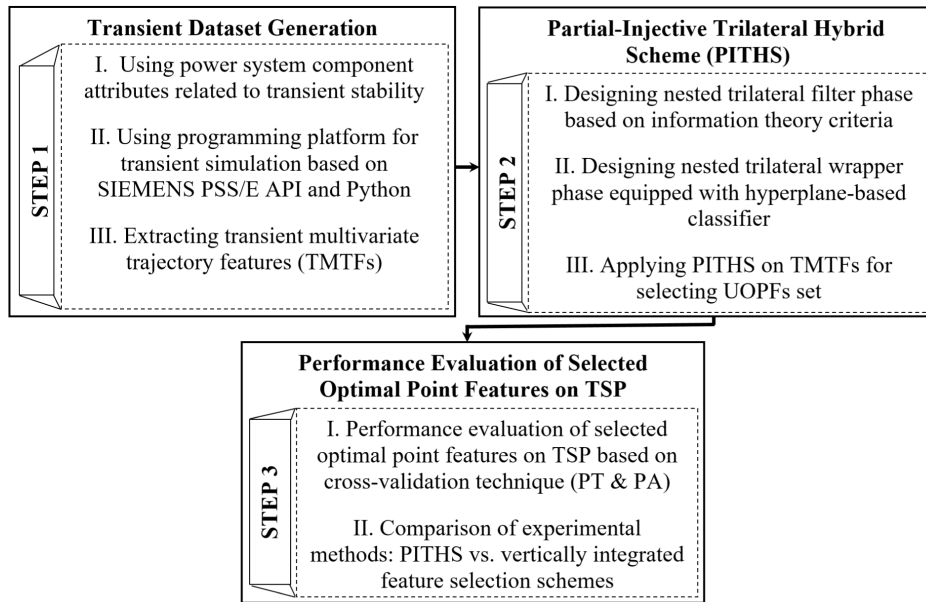


FIGURE 1. Visual summary of the proposed framework for TSA based on PITHS.

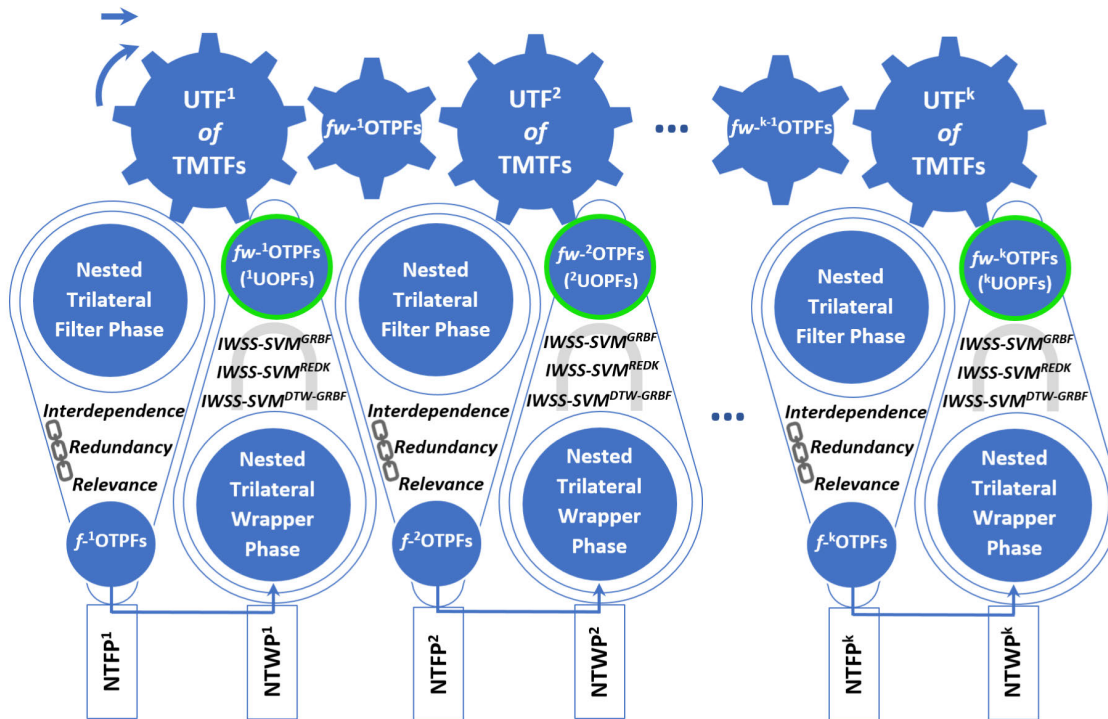


FIGURE 2. The overall process of PITHS for selecting UOPFs set.

scenario coupled with PITHS to obtain UOPFs set (green-face circles in Fig. 2), utilizing two nested trilateral phases (filter and wrapper) called 2NTPs in PITHS has a driven role for extracting UOPFs. The 2NTPs is including the nested trilateral filter phase (NTFP) and nested trilateral wrapper phase (NTWP). In NTFP, filter-optimal transient point features called f -OTPFs are selected via information theory-based triple criteria, namely relevance,

interdependence, and redundancy (RIR). The RIR analysis is exerted based on basic ratios like entropy, conditional entropy, mutual information (MI), and conditional MI. After conducting RIR analysis, the NTWP is applied on f -OTPFs for finding f_w -OTPFs. In fact, the NTWP is considered a supplementary phase in PITHS to cover the weakness of the IRI analysis in ignoring supervised learning-based analysis on feature selection process. In this regard, the

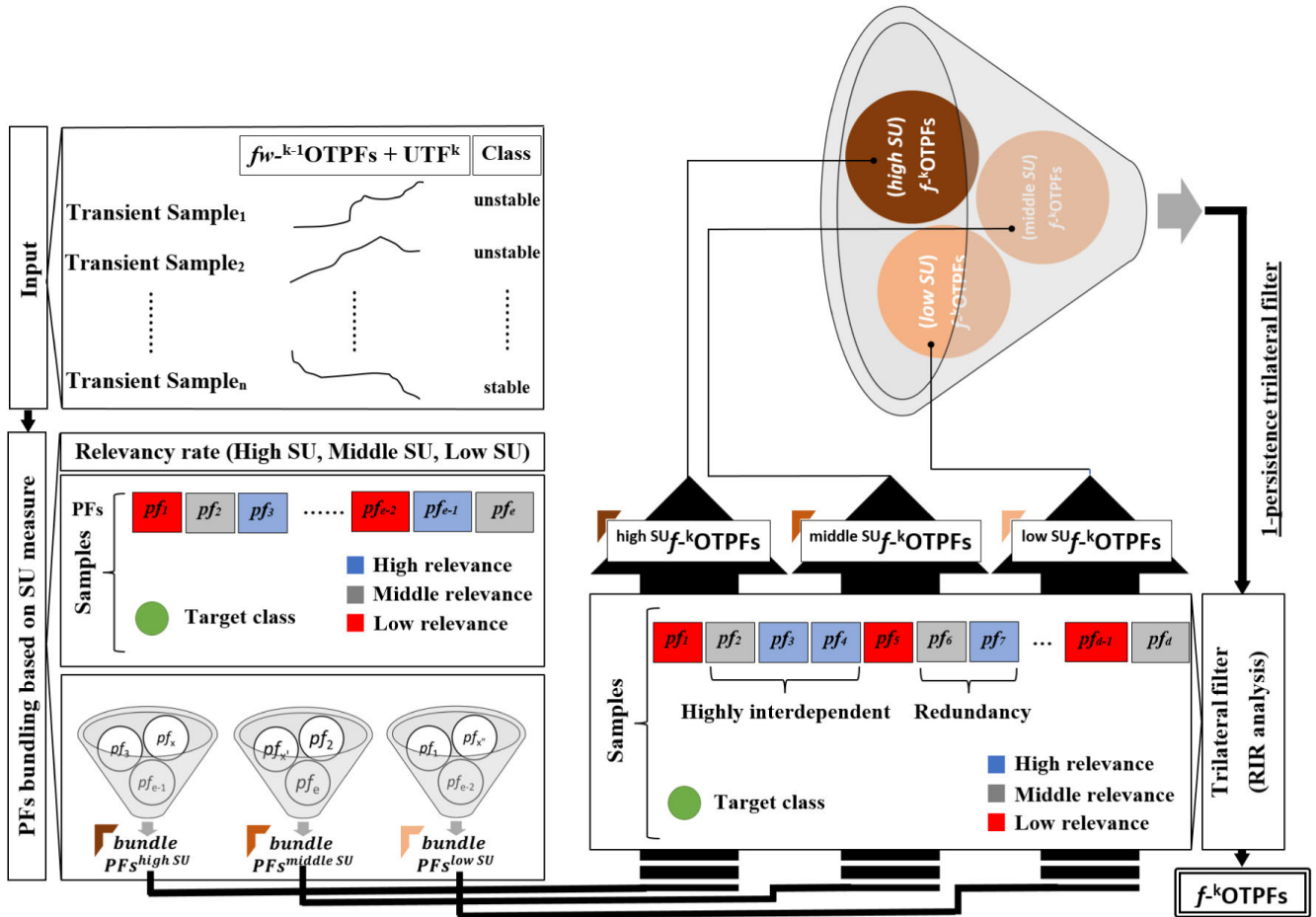


FIGURE 3. The visual summary of NTFP.

hyperplane-based approach equipped with the elastic and non-elastic kernels is considered in NTWP for selecting f_w -OTPFs. The detailed descriptions of the NTFP and NTWP are considered in the following sections.

A. NESTED TRILATERAL FILTER PHASE (NTFP)

In the filter phase of PITHS, we consider statistical and intrinsic characteristics of the trajectory features based on the information theory concept. In fact, three significant criteria of information theory to measure of relatedness state of point features per trajectory to the target class are considered in the NTFP, which is the leading transient data analytic package in

each stage of PITHS (See Fig. 2). As can be seen in Fig. 3, the NTFP of PITHS rounds consists of the nested steps as follow:

Step 1) Specifying the input trajectory: In the first stage of PITHS, the input trajectory, including univariate of the first trajectory called UTF^1 (See Fig. 2), is entered into $NTFP^1$. In the next stage of PITHS (2^{nd} round), trajectory input involved f_w^{-1} -OTPFs (optimal point features extracted based on applying $NTFP^1$ and $NTWP^1$ (See Section 2.2) on UTF^1) and UTF^2 are fed to $NTFP^2$. This procedure will continue until the k^{th} UTF, where f_w^{-k-1} -OTPFs and UTF^k (called $LnfT$) are entered into $NTFP^k$ in the last stage of PITHS. So, we have (1), shown at the bottom of the page, where

$$\begin{aligned}
 & \text{input trajectory per round of PITHS} \\
 & = f_w^{-k-1}OTPFs + UTF^k \text{ (general form)} \\
 & \left\{ \begin{array}{ll} 1^{st} \text{ round of NTFP in PITHS (NTFP}^1), & k = 1; \text{ input trajectory} = f_w^{-0}OTPFs(=nil) + UTF^1 = UTF^1 \\ 2^{nd} \text{ round of NTFP in PITHS (NTFP}^2), & k = 2; \text{ input trajectory} = f_w^{-1}OTPFs + UTF^2 \\ \vdots & \vdots \\ 28^{th} \text{ round of NTFP in PITHS (NTFP}^{28}), & k = 28; \text{ input trajectory} = f_w^{-27}OTPFs + UTF^{28} \end{array} \right. \quad (1)
 \end{aligned}$$

k indicates the number of univariate time series constrained $\{k|k = 1, 2, \dots, 28\}$.

Step 2) Calculating relevance rate: The first component of RIR analysis, namely relevance rate is considered in this step. To this end, symmetric uncertainty (SU) [14] is used for selecting point features (PFs) from univariate trajectory input, which has tightly relation with the target class. In terms of SU, entropy, conditional entropy, and mutual information (MI) are the main factors to measure the value of information shared between $pf \in$ PFs of input trajectory and target class. In this regard, the entropy $H(pf)$ is defined as:

$$H(pf) = - \sum_{x \in pf} p(x) \log p(x) \quad (2)$$

where $pf \in$ PFs be a discrete random variable and probability density function $p(x) = \Pr\{pf = x\}$. Also, conditional entropy calculates the entropy of fp in the presence of target class knowledge as follow:

$$H(pf|class) = - \sum_{x \in pf} \sum_{c \in class} p(x, c) \log p(x|c) \quad (3)$$

Now, MI is defined as Eq. (4):

$$MI(pf; class) = H(pf) - H(pf|class) \quad (4)$$

Regarding Eqs. (2) to (4), The SU index is calculated as normalized form of MI , given by:

$$SU(pf, class) = 2 \frac{MI(pf; class)}{H(pf) + H(class)} \quad (5)$$

Step 3) After calculating SU per $pf \in$ PFs of input trajectory, by setting proper thresholds, each pf is situated in one of the three bundles (high SU PFs, middle SU PFs, and low SU PFs) according to its SU amount. The main reason for making this decision is to avoid absolute reliance on obtained results based on the SU index and the prevention of forced removal of $pf \in$ PFs in the middle and low SU bundles. In fact, not content with SU-oriented analysis gives middle SU pf and low SU pf a chance to be re-analyzed via the interdependence-redundancy (IR) analysis and 1-persistence trilateral filter (See Step 4).

Step 4) After bundling PFs of input trajectory, each bundle (high SU PFs, middle SU PFs, and low SU PFs) entered to complementary analysis based on IR index as (6). In IR analysis, the effect of the presence of pf_j on information shared between pf_i and the target class is measured. If knowledge of pf_j cause to increase in the relevance ratio between pf_i and the target class, two features are interdependent. Also, joining pf_j to pf_i is considered redundant, if knowledge of pf_j causes a negative effect on the relevance between the pf_i and the target class.

$$IR(pf_i, pf_j) = 2 \frac{MI(pf_i; class|pf_j) - MI(pf_i; class)}{H(pf_i) + H(class)} \quad -1 \leq IR(pf_i, pf_j) \leq 1 \quad (6)$$

Consequently, by following Step 1 to Step 4, the filter-optimal transient point features (f -OTPFs) per bundle are

obtained (high SU_f -OTPFs, middle SU_f -OTPFs, and low SU_f -OTPFs). To better understand the details of RIR analysis, consider the pseudo-code of the NTFP (Step 1 to Step 4) as Table 1.

Step 5) The obtained high SU_f -OTPFs, middle SU_f -OTPFs, and low SU_f -OTPFs are joined together and entered into function IR once again (1-persistence scenario) for selecting f -OTPFs of input trajectory.

After conducting the above-mentioned steps, each NTFP^k-specific f -OTPFs are fed to NTWP^k ($k = 1$ to 28) (See Fig. 2 and Section 2.2) for finding NTWP^k-specific f_w -OTPFs at the end of each round of PITHS as ^kUOPFs (e.g.; last round: $k = 28$; f -²⁸OTPFs entered into NTWP²⁸ for finding f_w -²⁸OTPFs as ²⁸UOPFs).

B. NESTED TRILATERAL WRAPPER PHASE (NTWP)

For more analysis on transient features, the outputs of NTFP are fed to the predictive-oriented analysis called NTWP to extract the most discriminative transient features. As can be seen in Fig. 2, each NTFP^k-specific f -OTPFs are entered to NTWP^k (e.g.; first round: entering f -¹OTPFs into NTWP¹; last round: entering f -²⁸OTPFs into NTWP²⁸), which is considered as a complementary analysis in the proposed FSS. The NTWP is equipped with a hyperplane-based classifier accompanied by the triple kernel to evaluate the efficacy of selected features via NTFP. In fact, the point features in each NTFP^k-specific f -OTPFs inducing high performance in TSP survive by NTWP. Generally, each NTWP in PITHS is formed based on three components as follow:

1) THE WRAPPER APPROACH BASED ON INCREMENTAL MECHANISM

Regardless of the effective role of the filter method in selecting f -OTPFs, using the wrapper method in the form of applying supervised learning models is the key approach to evaluate the predictive capacity of f -OTPFs. In fact, selecting the discriminative transient point features (DTPFs) from the f -OTPFs that lead to the correct prediction of unseen transient samples (stable or unstable) is the main goal to incorporate this approach in the FSS process. However, applying a mechanism that checks the performance rate of the DTPFs subset for TSP in an incremental manner is another important aspect of the wrapper phase. One of the incremental-based mechanisms is incremental wrapper subset selection (IWSS) [15] that regards the SU value of each member of f -OTPFs as the criterion in arranging the entry of features to the learning model in each iteration of IWSS. In IWSS, first, the feature that has the highest SU is added to the DTPFs subset, and it is fed to the learning algorithm. Then, classification accuracy is recorded as the best result. In the next iterations, the feature with the second-highest SU is added to the DTPFs subset, and the training and testing procedure is conducted based on existing members of DTPFs. If by adding this feature to DTPFs, the prediction accuracy increased against the preceding DTPFs subset accuracy, the feature has remained in the DTPFs subset; otherwise, this

TABLE 1. The pseudo-code of the NTFP (step 1 to step 4).

<p>Input: transient trajectory: $f_{W^{k-1}}\text{OTPFs}+\text{UTF}^k$; $k=1, 2, \dots, 28$.</p> <p>Output: optimal transient point features in three bundles: $\text{high}^{SU}f\text{-OTPFs}$, $\text{middle}^{SU}f\text{-OTPFs}$, and $\text{low}^{SU}f\text{-OTPFs}$.</p> <p>Initialize parameters: $\text{threshold}^1=t$, $\text{threshold}^2=tt$, m=number of selected optimal point features per bundle.</p> <p>(1) for $i=1$ to $\text{length}(\text{PFs})$</p> <p>(2) SU_{pf_i} = calculate relevancy rate of $pf_i \in \text{PFs}$; // $SU(pf_i, \text{class})$</p> <p>(3) if $SU_{pf_i} > tt$</p> <p>(4) Add pf_i in bundle $\text{PFs}^{\text{high}^{SU}}$;</p> <p>(5) else if $t \leq SU_{pf_i} \leq tt$</p> <p>(6) Add pf_i in bundle $\text{PFs}^{\text{middle}^{SU}}$;</p> <p>(7) else</p> <p>(8) Add pf_i in bundle $\text{PFs}^{\text{low}^{SU}}$;</p> <p>(9) end</p> <p>(10) end</p> <p>// applying function IR per three bundles (bundle $\text{PFs}^{\text{high}^{SU}}$, bundle $\text{PFs}^{\text{middle}^{SU}}$, bundle $\text{PFs}^{\text{low}^{SU}}$)</p> <p>(11) $\text{level}^{SU}f\text{-OTPFs} = \text{IR}(\text{PFs}^{\text{level}^{SU}})$; // $\text{level} = \{\text{high} \parallel \text{middle} \parallel \text{low}\}$</p>
<p>$\text{level}^{SU}f\text{-OTPFs} = \text{Function IR}(\text{bundle } \text{PFs}^{\text{level}^{SU}})$</p> <p>(1) $\text{level}^{SU}f\text{-OTPFs} = \emptyset$;</p> <p>(2) $W_{pf}^{(0)}$ = initial weight per $pf \in \text{PFs}^{\text{level}^{SU}}$ is 1 equally;</p> <p>(3) pf^{h_1} = find pf with highest value in $SU_{pf \in \text{PFs}^{\text{level}^{SU}}}$;</p> <p>(4) $\text{level}^{SU}f\text{-OTPFs} = \text{level}^{SU}f\text{-OTPFs} \cup pf^{h_1}$;</p> <p>(5) $\text{PFs}^{\text{level}^{SU}} = \text{PFs}^{\text{level}^{SU}} - pf^{h_1}$;</p> <p>(6) for $v=2$ to m</p> <p>(7) $W_{pf}^{(v-1)} = W_{pf}^{(v-2)} \cdot \text{IR}(pf, pf^{h_{(v-1)}})$;</p> <p>(8) $SU_{pf \in \text{PFs}^{\text{level}^{SU}}}^{\text{New}} = W_{pf}^{(v-1)} \cdot (SU_{pf \in \text{PFs}^{\text{level}^{SU}}})$;</p> <p>(9) pf^{h_v} = find pf with highest value from $SU_{pf \in \text{PFs}^{\text{level}^{SU}}}^{\text{New}}$;</p> <p>(10) $\text{level}^{SU}f\text{-OTPFs} = \text{level}^{SU}f\text{-OTPFs} \cup pf^{h_v}$;</p> <p>(11) $\text{PFs}^{\text{level}^{SU}} = \text{PFs}^{\text{level}^{SU}} - pf^{h_v}$;</p> <p>(12) end</p> <p>(13) return $\text{level}^{SU}f\text{-OTPFs}$;</p>

feature is removed, and the next feature is added to DTPFs to be executed subsequent iteration of IWSS.

2) THE HYPERPLANE-BASED CLASSIFIER

Focusing on the type of classification algorithm embedded in the IWSS iterations is a significant issue that affects the learning process (train-test). This becomes doubly important when we find out that the processing time (PT) in the TA is a crucial concern. As mentioned in the introduction section, the reasonable PT for TSP is less than one second (<1s) which such PT constraint provides the necessary condition for timely corrective control action in the power grid. Two factors are influential in achieving low PT namely

observed length of transient cycles (observation window) and the time complexity of the classification model. Taking into cognizance these points, picking a small observation window (SOW) and labeling transient samples in low time is the only definitive solution to handle the PT bottleneck. To this end, we need to apply the robust-lightweight classifier that robust reflects the algorithm’s ability to precisely learn the decision boundary in SOW (high accuracy prediction), and lightweight refers to the algorithm capable of fast detection of transient stability status (low time complexity). As the best option, the support vector machine (SVM) [16] is the robust learning model that employs a separating hyperplane with low structural risk in the classification of

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

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

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, Q_{max} = maximum amount of machine reactive power, Q_{load} = 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, Mean= average.

the limited and not linearly separable transient feature space. Furthermore, several kernels can be embedded in the SVM classifier for optimal matching (point-based or trajectory-based alignment) between transient samples, which increases the generalization capacity of the learning model. On the

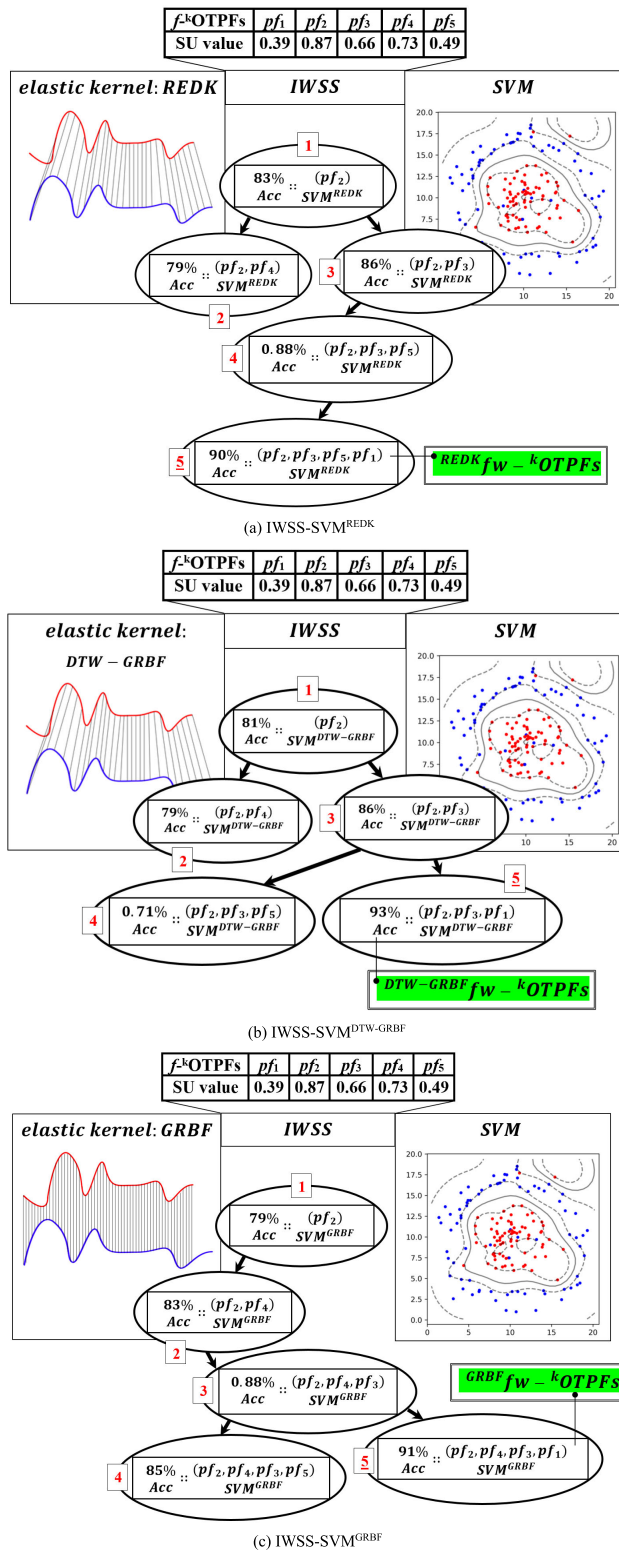


FIGURE 4. The visual summary of NTWP.

other hand, the kernel-based SVM capability to maximize prediction accuracy without overfitting training SOW indirectly affects train-test computational complexity and turns it into a lightweight classifier. Hence, this issue motivated us to use SVM in the IWSS iterations. The optimization problem

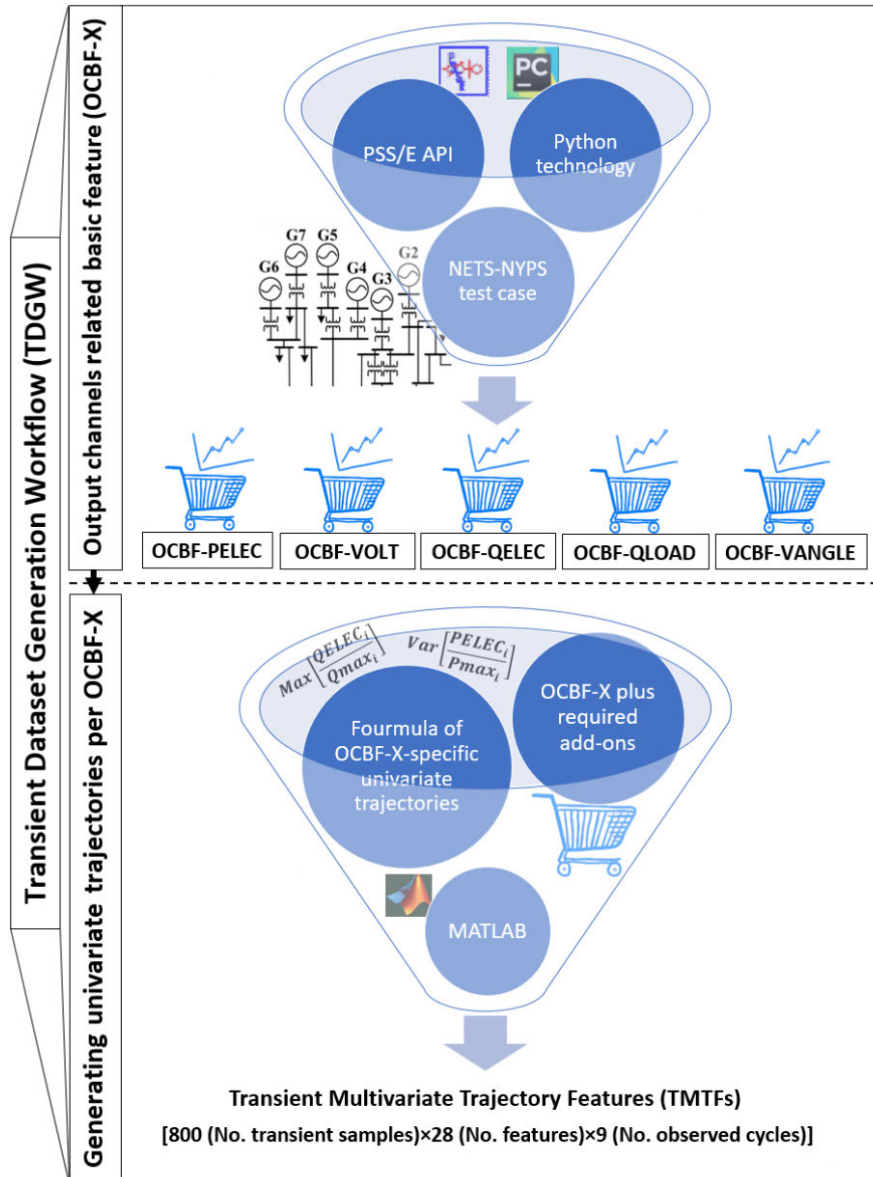


FIGURE 5. Transient dataset generation workflow (TDGW).

and the constraints of SVM are defined according to (7):

$$\begin{aligned}
 a^* &= \arg \min_{\alpha} \frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l \alpha_i \alpha_j y_i y_j K(x_i, x_j) - \sum_{k=1}^l \alpha_k; \\
 0 &\leq \alpha_i \leq C, \quad \sum_{j=1}^l \alpha_j y_j = 0, \quad i, j = 1, \dots, l \quad (7)
 \end{aligned}$$

The optimal separating surface in transient feature space is solved by (8):

$$\begin{aligned}
 f(x) &= \text{sgn} \left(\sum_{i \in S} \alpha_i y_i K(x_i, x) + b \right); \\
 b &= \frac{1}{s} \sum_{i \in S} \left[y_i - \sum_j \alpha_j y_j K(x_j, x_i) \right] \quad (8)
 \end{aligned}$$

3) TYPE OF KERNEL FUNCTION

In (7) and (8), $K(\cdot, \cdot)$ indicates the kernel function plugged into SVM to learn optimal decision boundary without raising the computational complexity. In this regard, three efficient kernel-based on elastic and non-elastic functions are introduced as follow:

a: THE NON-ELASTIC KERNEL

Standard Gaussian radial basis function (Standard GRBF) kernel [16]: The GRBF kernel works based on point to point alignment as (9):

$$K(x, x') = \exp \left(-\frac{\|x - x'\|^2}{2\sigma^2} \right) \quad (9)$$

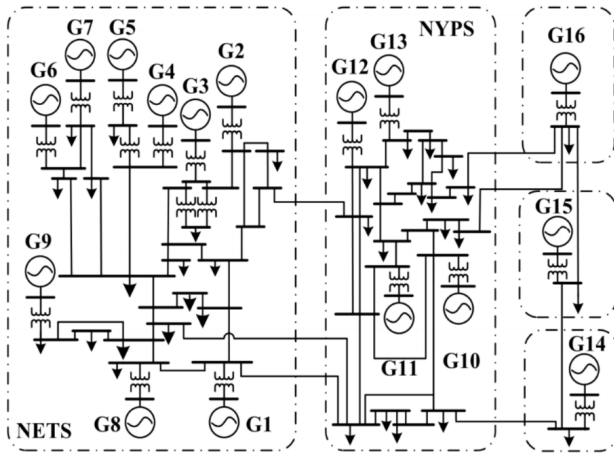


FIGURE 6. Single line diagram of NETS-NYPS test system.

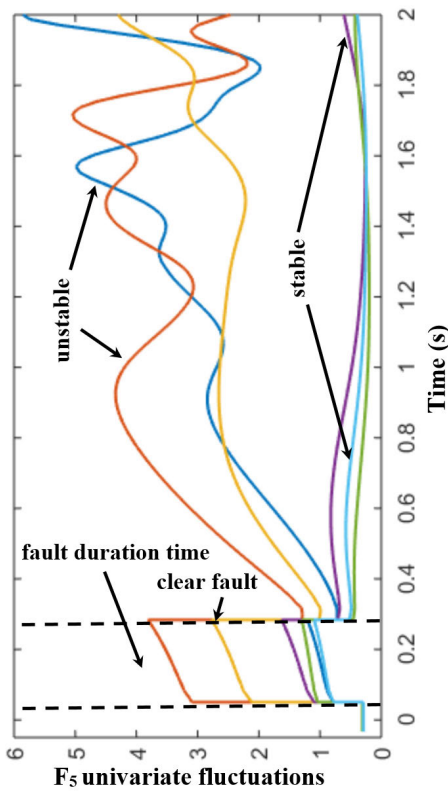


FIGURE 7. Stable and unstable sample based on F_5 variations.

where $\|x - x'\|^2$ is squared Euclidean distance between the two time series feature.

b: THE ELASTIC KERNEL

DTW in GRBF kernel (DTW-GRBF kernel) [17]: Since the pattern matching with DTW outperforms Euclidean distance in most cases because of its non-linear alignment, using DTW distance in the GRBF kernel can help to build the high-performance SVM model for time series classification.

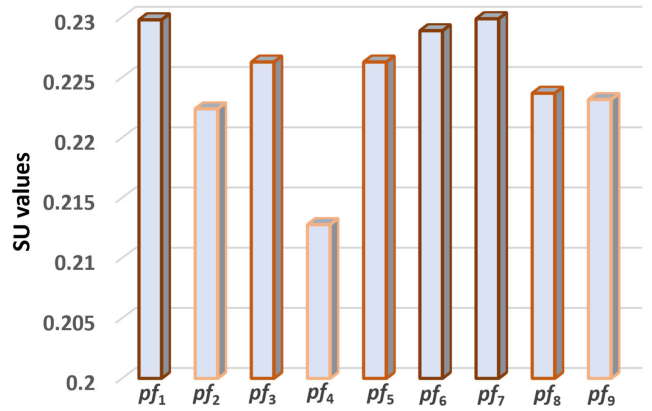
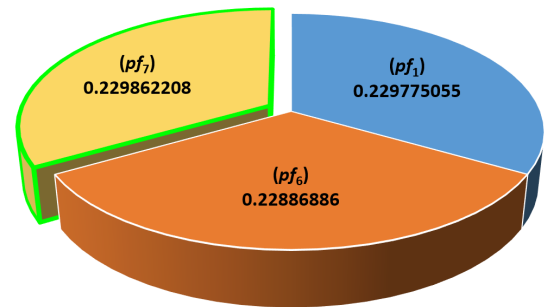


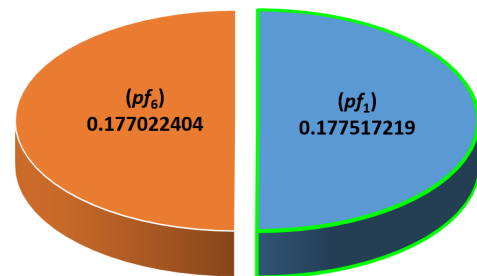
FIGURE 8. The relevancy rate (SU amount) for PFs-UTF¹.

Member¹ of high SU_f -OTPFs selected from high SU -UTF¹= pf_7



(a) Member¹

Member² of high SU_f -OTPFs selected from high SU -UTF¹= pf_1



(b) Member²

FIGURE 9. Selecting high SU_f -OTPFs of high SU PFs-UTF¹ via IR analysis.

Hence, the DTW-GRBF kernel is defined as (10):

$$K(x, x') = \exp\left(-\frac{[distance^{DTW}(A_1^p, B_1^q)]^2}{2\sigma^2}\right) \quad (10)$$

where

$$distance^{DTW}(A_1^p, B_1^q) = d(a(p), b(q)) + \text{Min} \begin{pmatrix} distance^{DTW}(A_1^{p-1}, B_1^q) \\ distance^{DTW}(A_1^p, B_1^{q-1}) \\ distance^{DTW}(A_1^p, B_1^{q-1}) \end{pmatrix}$$

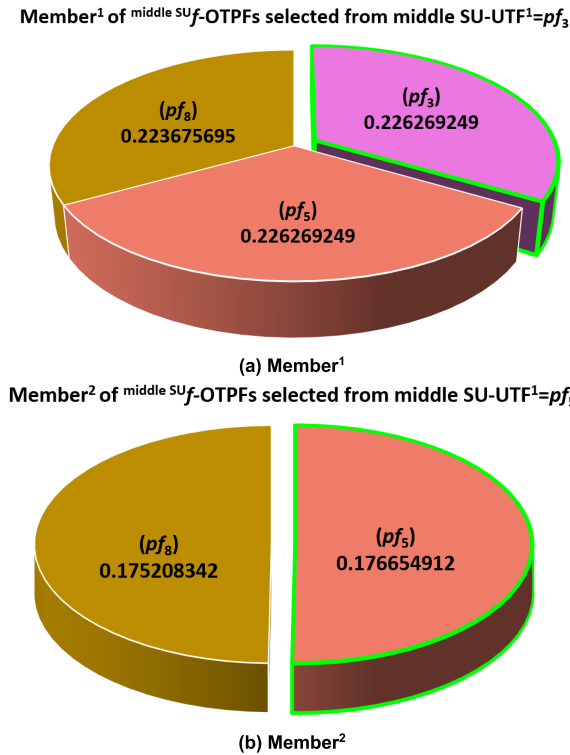


FIGURE 10. Selecting middle SU_f -OTPFs of middle SU PFs-UTF¹ via IR analysis.

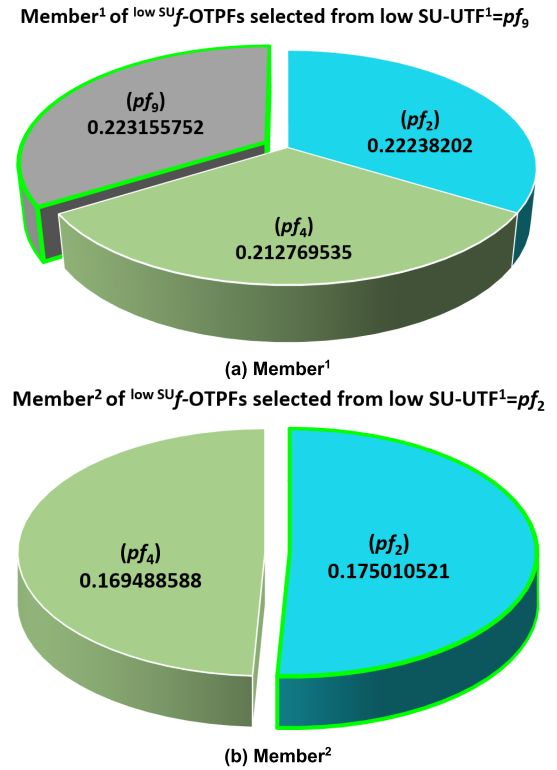


FIGURE 11. Selecting low SU_f -OTPFs of low SU PFs-UTF¹ via IR analysis.

c: THE ELASTIC KERNEL

Recursive edit distance kernel (REDK): Defining a positive definite symmetric (PDS) kernel as a valid alternative to conventional elastic kernel like DTW-based functions which does not always satisfy PDS acceptable by SVM has been considered in [18]. The function $\langle \cdot, \cdot \rangle : U \times U \rightarrow R$ is called REDK if, for any pair of trajectory. A_1^p, B_1^q there exists a function $f : R \rightarrow R$ such that the following recursive equation is satisfied:

$$\langle A_1^p, B_1^q \rangle = \sum \begin{cases} \langle A_1^{p-1}, B_1^q \rangle f(\Gamma(A(p) \rightarrow \Lambda)) \\ \langle A_1^p, B_1^{q-1} \rangle f(\Gamma(A(p) \rightarrow B(q))) \\ \langle A_1^p, B_1^q \rangle f(\Lambda \rightarrow B(q)) \end{cases} \quad (11)$$

Let U be the set of finite trajectory; $U = \{A_1^p | p \in \mathbb{N}\}$. A_1^p is a sequence with a discrete index varying between 1 and p . Also, $\Gamma(h)$ is the cost function for edit operation.

According to what was mentioned about the main components of NTWP, the NTWP is exerted on f -OTPFs as shown in Fig. 4. As can be seen in Fig. 4, each NTWP^k-specific f -^kOTPFs are entered into the three stages of NTWP^k. In the first stage, f -^kOTPFs are entered into IWSS-SVM^{REDK}. By applying SVM^{REDK} in IWSS iterations, the REDK _{f_w -^kOTPFs are obtained (See panel (a) of Fig. 4). Next, the f -^kOTPFs are fed to the IWSS-SVM^{DTW-GRBF} and consequently, the DTW-GRBF _{f_w -^kOTPFs is obtained as the output of the second stage of NTWP^k (See panel (b) of Fig 4).}}

Unlike the previous two stages of the NTWP^k which plugged elastic kernel into SVM to feed IWSS iterations, the obtained f -^kOTPFs are entered into the non-elastic face of NTWP^k accompanied with IWSS-SVM^{GRBF}. In this way, the GRBF _{f_w -^kOTPFs are extracted as the DTPFs subset of f -^kOTPFs. Next, intersection of kernel _{f_w -^kOTPFs in pairs ($[\text{REDK}_{f_w\text{-}^k\text{OTPFs}} \cap \text{DTW-GRBF}_{f_w\text{-}^k\text{OTPFs}}]$, $[\text{REDK}_{f_w\text{-}^k\text{OTPFs}} \cap \text{GRBF}_{f_w\text{-}^k\text{OTPFs}}]$, and $[\text{DTW-GRBF}_{f_w\text{-}^k\text{OTPFs}} \cap \text{GRBF}_{f_w\text{-}^k\text{OTPFs}}]$) are calculated and result with the most members is considered as the f_w -^kOTPFs of NTWP^k, which is called the ^kUOPFs. Furthermore, if intersection kernel _{f_w -^kOTPFs in pairs have the same length of members, we combine their members as ^kUOPFs. Also, if the difference in the prediction accuracy of each set compared to the other two sets is more than 10%, the members of that are added to ^kUOPFs (numeric example in Fig. 4, UOPFs: pf_1, pf_2, pf_3).}}}

III. EXPERIMENTAL DESIGN

A. TRANSIENT DATASET GENERATION

As can be seen in Fig. 1, transient dataset generation based on dynamic contingency simulation is the preliminary step of the proposed framework for TSP. For implementing this step, the transient dataset generation workflow (TDGW), including two parts followed as Fig. 5. In the first part of TDGW, Python technology, SIEMENS power system simulator for engineering (PSS/E) planning tools, and case study (top-funnel in Fig. 5) triangulated to record transient

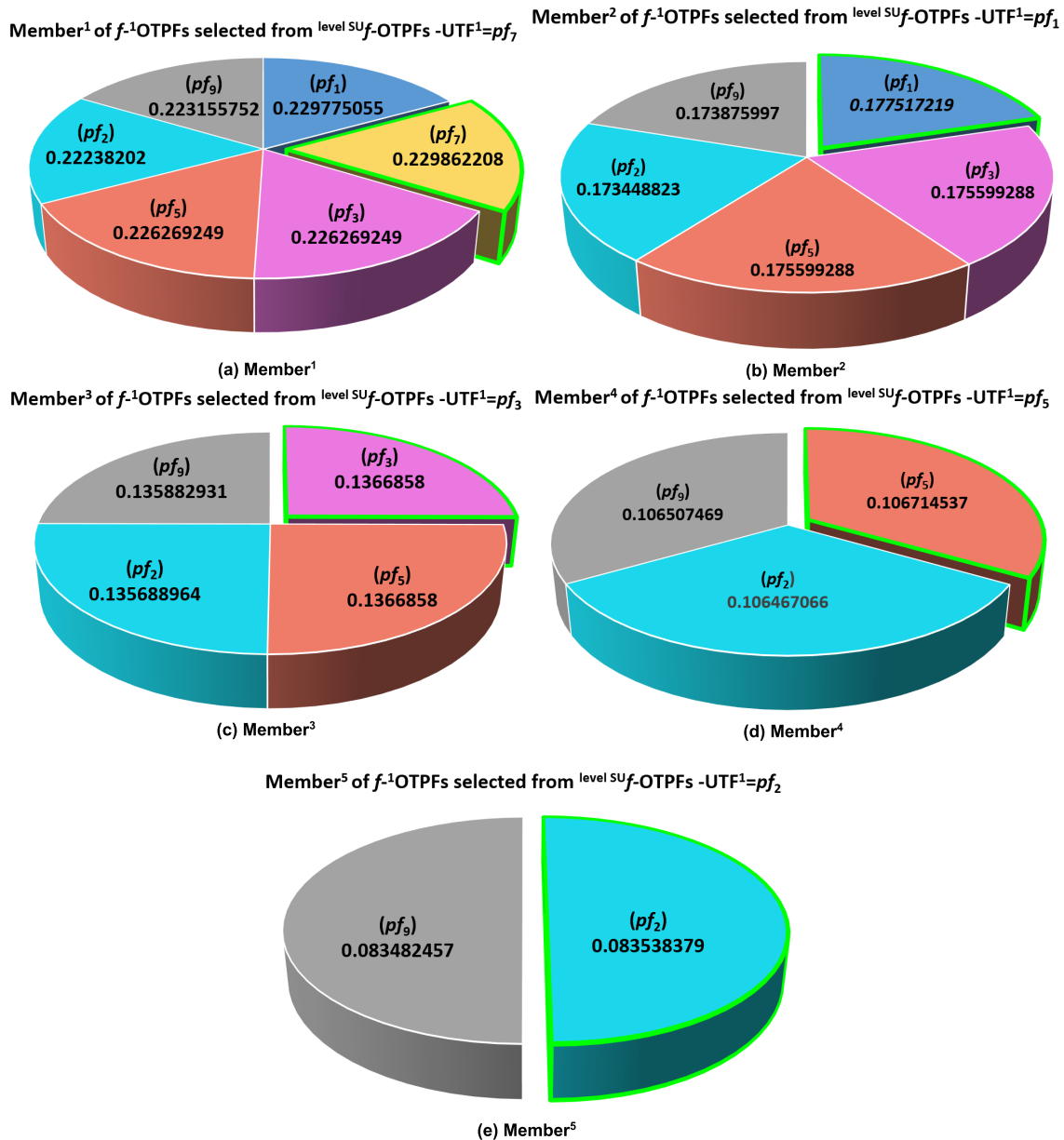
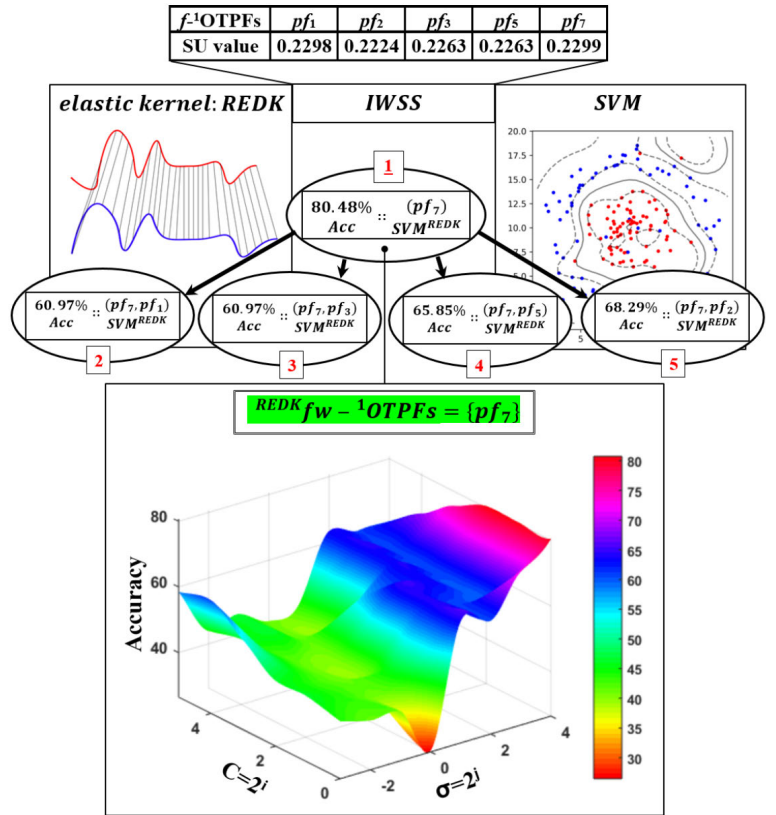


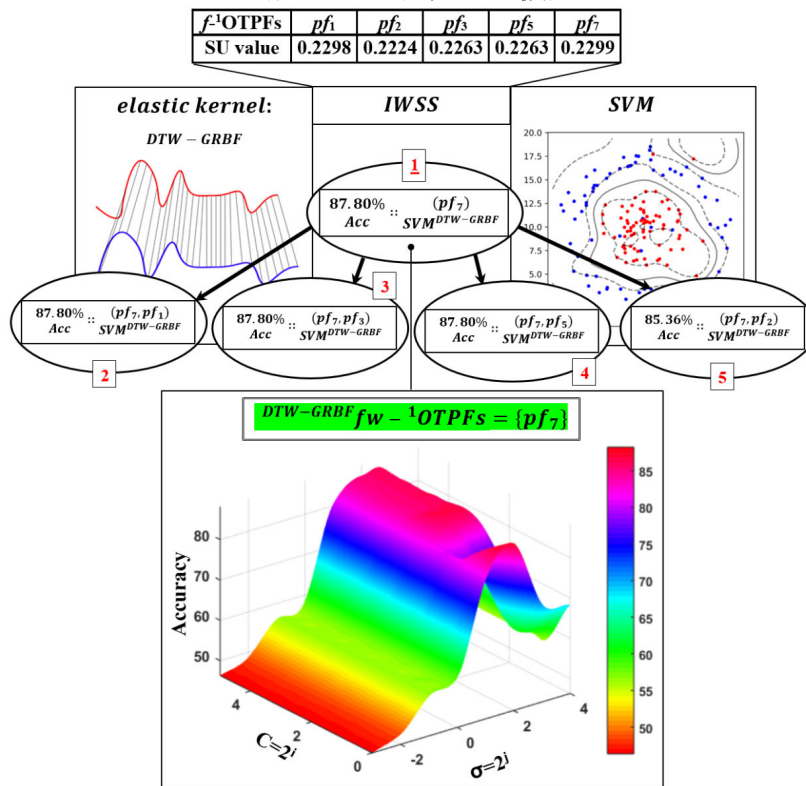
FIGURE 12. Selecting f^{-1} OTPFs of UTF¹.

data from output channels of basic features (OCBF-X). The X in OCBF-X is the symbol of basic features, including bus voltages (VOLT), voltage phase angle (VANGLE), machine active power (PELEC), machine reactive power (QELEC), and reactive power consumption (QLOAD) [19]. In this part, the transient sequence of OCBF-X is recorded based on coupling Python script and PSS/E application program interface (API) routine [20], which is executed on 68-Bus New England-New York interconnection system (NETS-NYPS) (See Fig. 6) [21]. An important point to note is that the obtained transient data of OCBF-X is related to applying several disturbances like substation outages, generator outages, and line outages on the NETS-NYPS grid case. In terms of simulation time, 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. Furthermore, for recording severe transient samples, different load characteristics are considered by converting the constant MVA load for a specified grouping of network loads to a specified mixture of the constant MVA, constant current, and constant admittance load characteristics. In the second step of TDGW, OCBF-X-specific univariate trajectories accompanied with required add-ons are entered into MATLAB-based feature calculation module (bottom-funnel in Fig. 5) which leads to construct transient multivariate trajectory features (TMTFs). The math formula of 28 univariate trajectory features (28-variate trajectory) [12], [13], [22] listed in Table 2 (e.g., OCBF-PELEC



(a) IWSS-SVM^{REDK} (^{REDK} f_W^{-1} OTPFs: { pf_7 })



(b) IWSS-SVM^{DTW-GRBF} (^{DTW-GRBF} f_W^{-1} OTPFs: { pf_7 })

FIGURE 13. The NTWP¹ for selecting f_W^{-1} OTPFs (¹UOPFs).

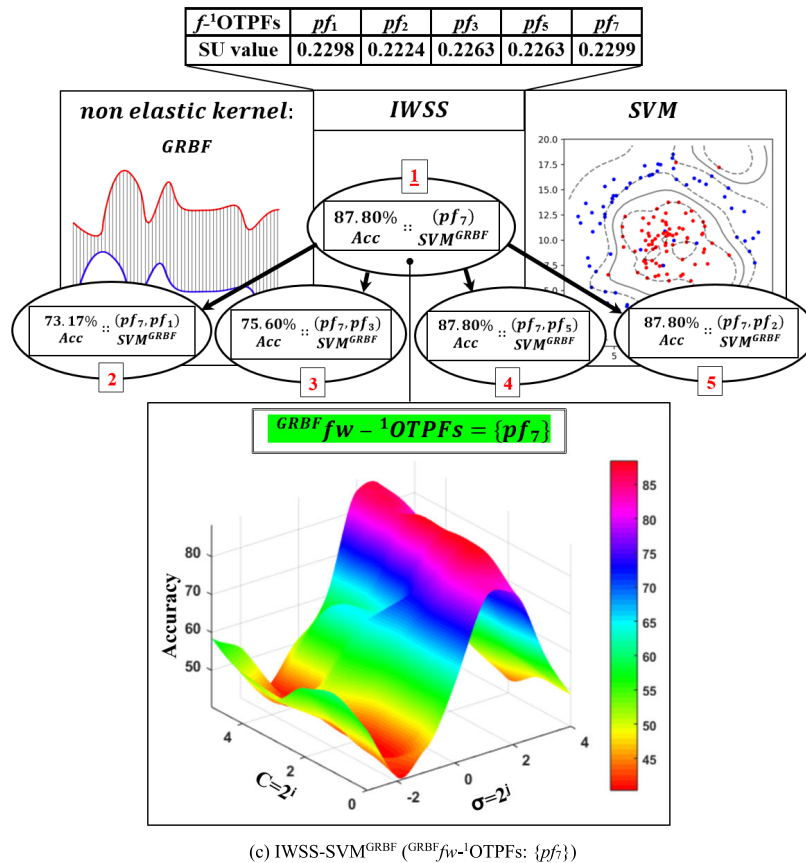


FIGURE 13. (Continued.) The NTWP¹ for selecting f_w^{-1} OTPFs (¹UOPFs).

plus P_{\max} : $F_1^{tm} = \text{Max}([\text{PELEC}_i / P_{\max_i}]^{i=1:N_{\text{genbus}}})$. Consequently, 800×28 transient samples were gathered based on TDGW, which is 9 cycles (0.1503 seconds) per univariate time series are observed after fault clearing time (we have: 800 (No. transient samples) \times 28 (No. features) \times 9 (No. observed cycles)). For example, Fig. 7 shows the fluctuations of the F_5 univariate trajectory (variance [proportion of QELEC to the Q_{\max}]) for stable and unstable samples of the NETS-NYPS test case.

B. UOPFs SET

The UOPFs set (¹UOPFs to ²⁸UOPFs) obtained by applying 2NTPs (NTFP and NTWP) in each round of PITHS (rounds 1 to 28) is elaborated in this section. According to (1), in the first round of PITHS, UTF¹ is entered into NTFP. According to what mentioned about the NTFP scenario (See Section 2.1), based on the first three steps of NTFP, the PFs of UTF¹ (9 cycles) based on relevance rate (SU measure) fall into the three bundles, namely high SU, middle SU, and low SU (Lines 1-10 of Table 1). As can be seen in Fig. 8, which is depicted the SU amounts of PFs-UTF¹, based on setting proper thresholds, $\{pf_1, pf_6, pf_7\}$ is situated in high SU PFs bundle, middle SU PFs bundle including $\{pf_3, pf_5, pf_8\}$, and $\{pf_2, pf_4, pf_9\}$ is considered as the low SU PFs.

After bundling PFs of UTF¹ in the form of high SU PFs, middle SU PFs, and low SU PFs, each bundle is fed to IR analysis (See Step 4 of Section 2.1; Line 1-13 of Table 1 (function IR)). By conducting IR analysis, the filter-optimal transient point features per bundle called ^{high} SU_f -OTPFs, ^{middle} SU_f -OTPFs, and ^{low} SU_f -OTPFs are obtained (See Fig. 3). The Fig. 9 to Fig. 11, shows how ^{level} SU_f -OTPFs is selected by IR analysis. As can be seen in Fig. 9 to Fig. 11 $\{pf_1, pf_7\}$, $\{pf_3, pf_5\}$, and $\{pf_2, pf_9\}$ are selected as ^{high} SU_f -OTPFs, ^{middle} SU_f -OTPFs, and ^{low} SU_f -OTPFs, respectively (See green-border point explosion in 3-D pies).

As the final step of NTFP (Step 5 of Section 2.1), the members of ^{high} SU_f -OTPFs, ^{middle} SU_f -OTPFs, and ^{low} SU_f -OTPFs combined ($\{pf_1, pf_2, pf_3, pf_5, pf_7, pf_9\}$) and then entered into the 1-persistence scenario of IR analysis for selecting f -OTPFs of UTF¹ (f^{-1} OTPFs) as the final output of the NTWP¹. As can be seen in Fig. 12, five PFs, namely $\{pf_1, pf_2, pf_3, pf_5, pf_7\}$ are selected from ^{level} SU_f -OTPFs members (six PFs) as the f^{-1} OTPFs.

After conducting the NTFP¹ on UTF¹ for selecting f^{-1} OTPFs, the f^{-1} OTPFs are fed to NTWP (See Section 2.2) for finding NTWP¹-specific f_w^{-1} OTPFs at the end of the first round of PITHS as ¹UOPFs. To this end, by applying IWSS-SVM^{REDK}, IWSS-SVM^{DTW-GRBF}, and IWSS-SVM^{GRBF}, the ^{REDK} f_w^{-1} OTPFs ($\{pf_7\}$):80.48%

TABLE 4. Results of PITHS round # (NTWP).

NTWP round #	Input	REDK f_w -#OTPFs	DTW-GRBF f_w -#OTPFs	GRBF f_w -#OTPFs	f_w -#OTPFs (#UOPFs)
1	f - ¹ OTPFs	{ ¹ pf_7 }:80.48%	{ ¹ pf_7 }:87.80%	{ ¹ pf_7 }:87.80%	{ ¹ pf_7 }
2	f - ² OTPFs	{ ¹ pf_7 , ² pf_3 , ² pf_4 }:90.24%	{ ² pf_1 }:68.29%	{ ¹ pf_7 , ² pf_3 , ² pf_4 }:92.68%	{ ¹ pf_7 , ² pf_3 , ² pf_4 }
3	f - ³ OTPFs	{ ² pf_1 , ² pf_3 , ³ pf_5 , ³ pf_6 }:95.12%	{ ² pf_1 , ³ pf_5 }:92.68%	{ ² pf_1 , ² pf_3 , ³ pf_5 , ³ pf_6 }:92.68%	{ ² pf_1 , ² pf_3 , ³ pf_5 }
4	f - ⁴ OTPFs	{ ² pf_1 , ³ pf_5 , ⁴ pf_4 }:90.24%	{ ² pf_1 , ³ pf_5 , ⁴ pf_4 }:87.80%	{ ² pf_1 , ³ pf_5 , ⁴ pf_4 }:87.80%	{ ² pf_1 , ³ pf_5 , ⁴ pf_4 }
5	f - ⁵ OTPFs	{ ³ pf_5 , ³ pf_6 , ³ pf_7 }:95.12%	{ ² pf_1 , ³ pf_5 }:92.68%	{ ² pf_1 , ³ pf_5 , ³ pf_7 }:92.68%	{ ² pf_1 , ³ pf_5 , ³ pf_7 }
6	f - ⁶ OTPFs	{ ⁶ pf_8 }:80.48%	{ ² pf_1 , ³ pf_5 , ⁶ pf_8 }:92.68%	{ ⁶ pf_8 }:80.48%	{ ² pf_1 , ³ pf_5 , ⁶ pf_8 }
7	f - ⁷ OTPFs	{ ⁶ pf_8 }:80.48%	{ ² pf_1 , ⁶ pf_8 }:68.29%	{ ⁶ pf_8 }:80.48%	{ ⁶ pf_8 }
8	f - ⁸ OTPFs	{ ⁶ pf_8 , ⁸ pf_9 , ⁸ pf_7 }:90.24%	{ ⁶ pf_8 , ⁸ pf_9 , ⁸ pf_7 }:87.80%	{ ⁶ pf_8 , ⁸ pf_9 , ⁸ pf_7 }:92.68%	{ ⁶ pf_8 , ⁸ pf_9 , ⁸ pf_7 }
9	f - ⁹ OTPFs	{ ⁶ pf_8 , ⁸ pf_9 , ⁸ pf_7 }:90.24%	{ ⁶ pf_8 , ⁸ pf_9 , ⁸ pf_7 }:87.80%	{ ⁶ pf_8 , ⁸ pf_9 , ⁸ pf_7 }:90.24%	{ ⁶ pf_8 , ⁸ pf_9 , ⁸ pf_7 }
10	f - ¹⁰ OTPFs	{ ⁶ pf_8 , ⁸ pf_9 , ¹⁰ pf_1 }:87.80%	{ ⁶ pf_8 , ⁸ pf_9 , ¹⁰ pf_1 }:90.24%	{ ⁶ pf_8 , ⁸ pf_9 , ¹⁰ pf_1 , ¹⁰ pf_3 }:90.24%	{ ⁶ pf_8 , ⁸ pf_9 , ¹⁰ pf_1 }
11	f - ¹¹ OTPFs	{ ⁶ pf_8 , ⁸ pf_9 , ¹⁰ pf_1 , ¹¹ pf_1 }:92.68%	{ ⁶ pf_8 , ⁸ pf_9 , ¹⁰ pf_1 }:90.24%	{ ¹¹ pf_1 , ¹¹ pf_5 , ⁶ pf_8 , ⁸ pf_9 , ¹⁰ pf_1 }:95.12%	{ ⁶ pf_8 , ⁸ pf_9 , ¹⁰ pf_1 }
12	f - ¹² OTPFs	{ ¹² pf_2 , ⁶ pf_8 , ¹² pf_2 }:85.36%	{ ⁶ pf_8 , ¹⁰ pf_1 }:73.17%	{ ⁶ pf_8 , ¹² pf_2 , ¹² pf_2 }:92.68%	{ ⁶ pf_8 , ¹² pf_2 }
13	f - ¹³ OTPFs	{ ⁶ pf_8 , ¹² pf_2 }:82.92%	{ ⁶ pf_8 , ¹² pf_2 , ¹³ pf_3 , ¹³ pf_3 }:85.36%	{ ⁶ pf_8 , ¹² pf_2 }:85.36%	{ ⁶ pf_8 , ¹² pf_2 }
14	f - ¹⁴ OTPFs	{ ⁶ pf_8 , ¹² pf_2 , ¹⁴ pf_1 }:85.36%	{ ⁶ pf_8 , ¹² pf_2 , ¹⁴ pf_1 , ¹⁴ pf_4 }:85.36%	{ ¹⁴ pf_2 , ¹⁴ pf_3 , ⁶ pf_8 , ¹² pf_2 }:92.68%	{ ⁶ pf_8 , ¹² pf_2 , ¹⁴ pf_1 }
15	f - ¹⁵ OTPFs	{ ⁶ pf_8 , ¹² pf_2 , ¹⁴ pf_1 }:85.36%	{ ⁶ pf_8 , ¹² pf_2 , ¹⁴ pf_1 }:80.48%	{ ¹⁵ pf_2 , ¹⁵ pf_6 , ⁶ pf_8 , ¹² pf_2 }:90.24%	{ ⁶ pf_8 , ¹² pf_2 , ¹⁴ pf_1 }
16	f - ¹⁶ OTPFs	{ ¹⁶ pf_4 , ⁶ pf_8 , ¹² pf_2 , ¹⁴ pf_1 }:90.24%	{ ⁶ pf_8 , ¹² pf_2 , ¹⁴ pf_1 }:80.48%	{ ¹⁶ pf_4 , ⁶ pf_8 , ¹² pf_2 }:87.80%	{ ⁶ pf_8 , ¹² pf_2 , ¹⁴ pf_1 , ¹⁶ pf_4 }
17	f - ¹⁷ OTPFs	{ ¹⁷ pf_5 , ⁶ pf_8 , ¹² pf_2 }:85.36%	{ ¹⁷ pf_5 , ⁶ pf_8 , ¹⁷ pf_6 }:82.92%	{ ⁶ pf_8 , ¹² pf_2 , ¹⁷ pf_4 , ¹⁷ pf_7 }:92.68%	{ ⁶ pf_8 , ¹² pf_2 , ¹⁷ pf_5 }
18	f - ¹⁸ OTPFs	{ ⁶ pf_8 , ¹⁸ pf_3 }:82.92%	{ ⁶ pf_8 , ¹⁸ pf_3 }:60.97%	{ ⁶ pf_8 , ¹⁸ pf_6 }:82.92%	{ ⁶ pf_8 }
19	f - ¹⁹ OTPFs	{ ¹⁹ pf_5 , ¹⁹ pf_4 , ⁶ pf_8 }:90.24%	{ ⁶ pf_8 , ¹⁹ pf_3 , ¹⁹ pf_4 }:82.92%	{ ⁶ pf_8 , ¹⁹ pf_3 , ¹⁹ pf_4 }:90.24%	{ ⁶ pf_8 , ¹⁹ pf_3 , ¹⁹ pf_4 }
20	f - ²⁰ OTPFs	{ ⁶ pf_8 , ¹⁹ pf_4 }:85.36%	{ ⁶ pf_8 , ¹⁹ pf_3 , ¹⁹ pf_4 }:82.92%	{ ⁶ pf_8 , ¹⁹ pf_3 , ¹⁹ pf_4 }:90.24%	{ ⁶ pf_8 , ¹⁹ pf_3 , ¹⁹ pf_4 }
21	f - ²¹ OTPFs	{ ⁶ pf_8 , ¹⁹ pf_4 }:85.36%	{ ⁶ pf_8 , ¹⁹ pf_3 , ¹⁹ pf_4 }:82.92%	{ ⁶ pf_8 , ¹⁹ pf_3 , ¹⁹ pf_4 }:90.24%	{ ⁶ pf_8 , ¹⁹ pf_3 , ¹⁹ pf_4 }
22	f - ²² OTPFs	{ ²² pf_3 , ²² pf_6 , ²² pf_6 , ⁶ pf_8 , ¹⁹ pf_4 }:95.12%	{ ²² pf_3 , ⁶ pf_8 , ¹⁹ pf_4 , ¹⁹ pf_4 , ²² pf_6 }:92.68%	{ ⁶ pf_8 , ¹⁹ pf_3 , ¹⁹ pf_4 }:90.24%	{ ⁶ pf_8 , ¹⁹ pf_4 , ²² pf_3 , ²² pf_6 }
23	f - ²³ OTPFs	{ ⁶ pf_8 , ¹⁹ pf_4 , ²² pf_6 , ²³ pf_6 }:95.12%	{ ⁶ pf_8 , ¹⁹ pf_4 , ²² pf_6 , ²³ pf_6 , ²³ pf_6 }:95.12%	{ ²³ pf_2 , ⁶ pf_8 , ¹⁹ pf_4 }:92.68%	{ ⁶ pf_8 , ¹⁹ pf_4 , ²² pf_6 , ²³ pf_6 }
24	f - ²⁴ OTPFs	{ ²² pf_6 , ⁶ pf_8 , ¹⁹ pf_4 }:87.80%	{ ²² pf_6 , ⁶ pf_8 , ¹⁹ pf_4 }:87.80%	{ ⁶ pf_8 , ¹⁹ pf_4 }:90.24%	{ ⁶ pf_8 , ¹⁹ pf_4 , ²² pf_6 }
25	f - ²⁵ OTPFs	{ ⁶ pf_8 , ¹⁹ pf_4 , ²⁵ pf_6 , ²⁵ pf_6 }:92.68%	{ ⁶ pf_8 , ²⁵ pf_6 , ²⁵ pf_6 , ²⁵ pf_6 }:95.12%	{ ⁶ pf_8 , ¹⁹ pf_4 , ²⁵ pf_6 }:92.68%	{ ⁶ pf_8 , ¹⁹ pf_4 , ²⁵ pf_6 , ²⁵ pf_6 }
26	f - ²⁶ OTPFs	{ ⁶ pf_8 , ¹⁹ pf_4 }:85.36%	{ ⁶ pf_8 , ¹⁹ pf_4 , ²⁶ pf_1 }:85.36%	{ ⁶ pf_8 , ¹⁹ pf_4 }:90.24%	{ ⁶ pf_8 , ¹⁹ pf_4 }
27	f - ²⁷ OTPFs	{ ⁶ pf_8 , ¹⁹ pf_4 , ²⁷ pf_1 , ²⁷ pf_2 , ²⁷ pf_4 }:95.12%	{ ⁶ pf_8 , ¹⁹ pf_4 , ²⁷ pf_4 }:85.36%	{ ⁶ pf_8 , ¹⁹ pf_4 }:90.24%	{ ⁶ pf_8 , ¹⁹ pf_4 , ²⁷ pf_4 }
28	f - ²⁸ OTPFs	{ ⁶ pf_8 , ¹⁹ pf_4 , ²⁷ pf_4 , ²⁸ pf_1 }:90.24%	{ ⁶ pf_8 , ¹⁹ pf_4 , ²⁷ pf_4 , ²⁸ pf_1 }:87.80%	{ ⁶ pf_8 , ¹⁹ pf_4 }:90.24%	{ ⁶ pf_8 , ¹⁹ pf_4 , ²⁷ pf_4 , ²⁸ pf_1 }

TABLE 5. UOPFs set.

Members of UOPFs set: UTF ^l : {cycle #}	
UTF ¹ : { pf_7 }	UTF ² : { pf_1 , pf_3 , pf_4 }
UTF ³ : { pf_5 }	UTF ⁴ : { pf_4 }
UTF ⁵ : { pf_1 }	UTF ⁶ : { pf_6 }
UTF ⁸ : { pf_5 , pf_7 }	UTF ¹⁰ : { pf_1 }
UTF ¹² : { pf_2 }	UTF ¹⁴ : { pf_1 }
UTF ¹⁶ : { pf_4 }	UTF ¹⁷ : { pf_5 }
UTF ¹⁹ : { pf_5 , pf_4 }	UTF ²² : { pf_5 , pf_6 }
UTF ²³ : { pf_6 }	UTF ²⁵ : { pf_5 , pf_6 }
UTF ²⁷ : { pf_4 }	UTF ²⁸ : { pf_1 }

TABLE 6. The performance metrics.

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

TSP accuracy), DTW-GRBF f_w -¹OTPFs ({ pf_7 }:87.80% TSP accuracy), and GRBF f_w -¹OTPFs ({ pf_7 }:87.80% TSP accuracy) are obtained, respectively. Next, REDK f_w -¹OTPFs \cap DTW-GRBF f_w -¹OTPFs (pf_7), REDK f_w -¹OTPFs \cap GRBF f_w -¹OTPFs (pf_7), and DTW-GRBF f_w -¹OTPFs \cap GRBF f_w -¹OTPFs

(pf_7) are calculated and pf_7 is considered as the f_w -¹OTPFs of NTWP¹. A graphical report of how f_w -¹OTPFs (¹UOPFs) are selected based on the NTWP¹ is shown in Fig. 13. The performance metric (12) is considered to evaluate IWSS-SVM^{kernel} performance via fine-tuning on learning parameters (C and σ in SVM and its plugged kernels). The C and σ in SVM^{kernel} are selected from { $C = 2^i | i = 0, 1, \dots, 5$ } and { $\sigma = 2^j | j = -3, -2, \dots, 4$ }.

Accuracy(Acc)

$$= (TP + TN)/(TP + TN + FP + FN)$$

$$\begin{cases} P : \text{stable sample}; & T : \text{predicted correctly} \\ N : \text{unstable sample}; & F : \text{predicted incorrectly} \end{cases} \quad (12)$$

In each iteration of IWSS based on SVM^{kernel}, the maximum value of the Acc package (retrieved by optimal pair of learning parameters) is recorded. In Fig. 13, we depicted the IWSS-SVM^{kernel} Acc variations based on learning parameters (C , σ) related to selected ^{kernel} f_w -¹OTPFs. For example, the illustration of SVM^{REDK} performance variations (Acc) in the selected iteration of IWSS is shown in panel (a) of Fig. 13 (in the first iteration: pf_7 as REDK f_w -¹OTPFs). Also, SVM^{DTW-GRBF} and SVM^{GRBF} performance variations

TABLE 7. Results of TSP based on UOPFs 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	
SVM ^{GRBF}	NETS-NYPS	98.75	96.25	97.5	98.75	
		[97.5 / 100]	[100 / 92.5]	[95 / 100]	[100 / 97.5]	
		fold 5	fold 6	fold 7	fold 8	
		100	100	98.75	100	
		[100 / 100]	[100 / 100]	[97.5 / 100]	[100 / 100]	
		fold 9	fold 10			
		97.5	100			
		[97.5 / 97.5]	[100 / 100]			
		Mean(measure) of folds: Accuracy [TPR / TNR]				
		98.75 [98.75 / 98.75]				

TABLE 8. Processing time for TSP based on UOPFs set.

Observation window (OW) in cycle / second	Processing Time (OW + prediction time)
9 / 0.1503	150.3 ms + 2.291 ms = 152.591 ms

related to DTW-GRBF _{f_w} ⁻¹OTPFs (first iteration with pf_7) and GRBF _{f_w} ⁻¹OTPFs (first iteration with pf_7) are shown in panel (b) and (c) of Fig. 13, respectively.

An important point to note is that the obtained ¹UOPFs are considered as the final output of the first round (NTFP¹ and NTWP¹) of PITHS and will be injected into the second round of PITHS (Refer to (1)). For more clarity, the round-specific results related to 2nd to 28th rounds of PITHS are shown in Table 3 (including input and outputs of NTFP per round) and Table 4 (including input and outputs of NTWP per round). In the final step of PITHS, the obtained f_w -OTPFs (UOPFs) per round of PITHS (See the last column of Table 4) are joined together to obtain the UOPFs set. The members of the UOPFs set including selected cycles of each UTF that will be used for transient stability prediction (See Section 3.3) are listed in Table 5.

C. TSP BASED ON UOPFs SET

After selecting the UOPFs set based on PITHS, the performance evaluation of UOPFs set in TSP based on cross-validation technique which is accompanied with SVM^{GRBF} was exerted on the transient dataset. In the learning scenario (SVM^{GRBF}) plugged into the cross-validation technique, for finding the optimal value of learning parameters (C , σ) to achieve high-performance prediction in each fold (train-test in fold 1 to fold 10), values of C and σ were selected from $\{C = 2^i | i = 0, 1, \dots, 15\}$ and $\{\sigma = 2^j | j = -5, -4, \dots, 15\}$, respectively. Furthermore, for performance evaluation of TSP in each fold, three metrics were considered in this section as Table 6.

After conducting the above-mentioned train-test procedure, the value of the Acc index per fold in TSP is shown in Table 7. The Acc results per fold are related to the maximum value of Acc variations. For more details of

Acc variations based on the different values of learning parameters, we depicted Acc variation of some folds as Fig. 14. Also, TPR and TNR related to obtained Acc per fold are considered for in-depth analysis on UOPFs set capacity in TSP. Also, the mean value of Acc, TPR, and TNR in 10-folds are given in Table 7 (in the last row) as the final report of UOPFs set efficacy on TSP. As can be seen in Table 7, the classification accuracy based on the UOPFs set shows the high-performance capacity (high PA) for TSP (Acc: 98.75 %, TPR: 98.75%, and TNR: 98.75%). In addition to PA analysis, the PT index, including observation time and prediction time was considered in the performance evaluation process. Regarding selected cycles of UTFs caused to construct UOPFs set show this fact that the maximum observing time for recording UTF cycles will not take more than 9 cycles (e.g., 9th cycle of UTF²³ and UTF²⁵, in the rest of UTF, the observation window will take less than 9 cycles). Hence, the observation window (OW) is equal to 9 cycles (150.3 milliseconds (ms)). Also, the prediction time based on applied SVM^{GRBF} on UOPFs is 2.291 ms. Consequently, the processing time is 152.591 ms (See Table 8) which in this way, suitable time conditions are provided for the system operator to take corrective actions.

D. COMPARISON OF EXPERIMENTAL METHODS: PITHS SCHEME VS. VERTICALLY INTEGRATED FSSs

For more clarity on the efficacy of horizontally integrated PITHS, we compare it with vertically integrated feature selection methods in this section. Hence, we consider four vertical-oriented FSS (4vFSS), including ReliefF [7], minimum redundancy maximum relevance (mRMR) [8], fast correlation-based filter (FCBF) [10], and bi-mode hybrid feature selection scheme (BMHFSS) [13], which are applied on transient data in FSS-based TSA studies. First, optimal transient features of 28-variate trajectory features are selected by 4vFSS [13]. Next, the selected optimal cycles by 4vFSS are entered into the SVM^{GRBF} classifier for TA. Based on obtained results (See Table 9), survived features by PITHS have better performance in TSP than selected features by

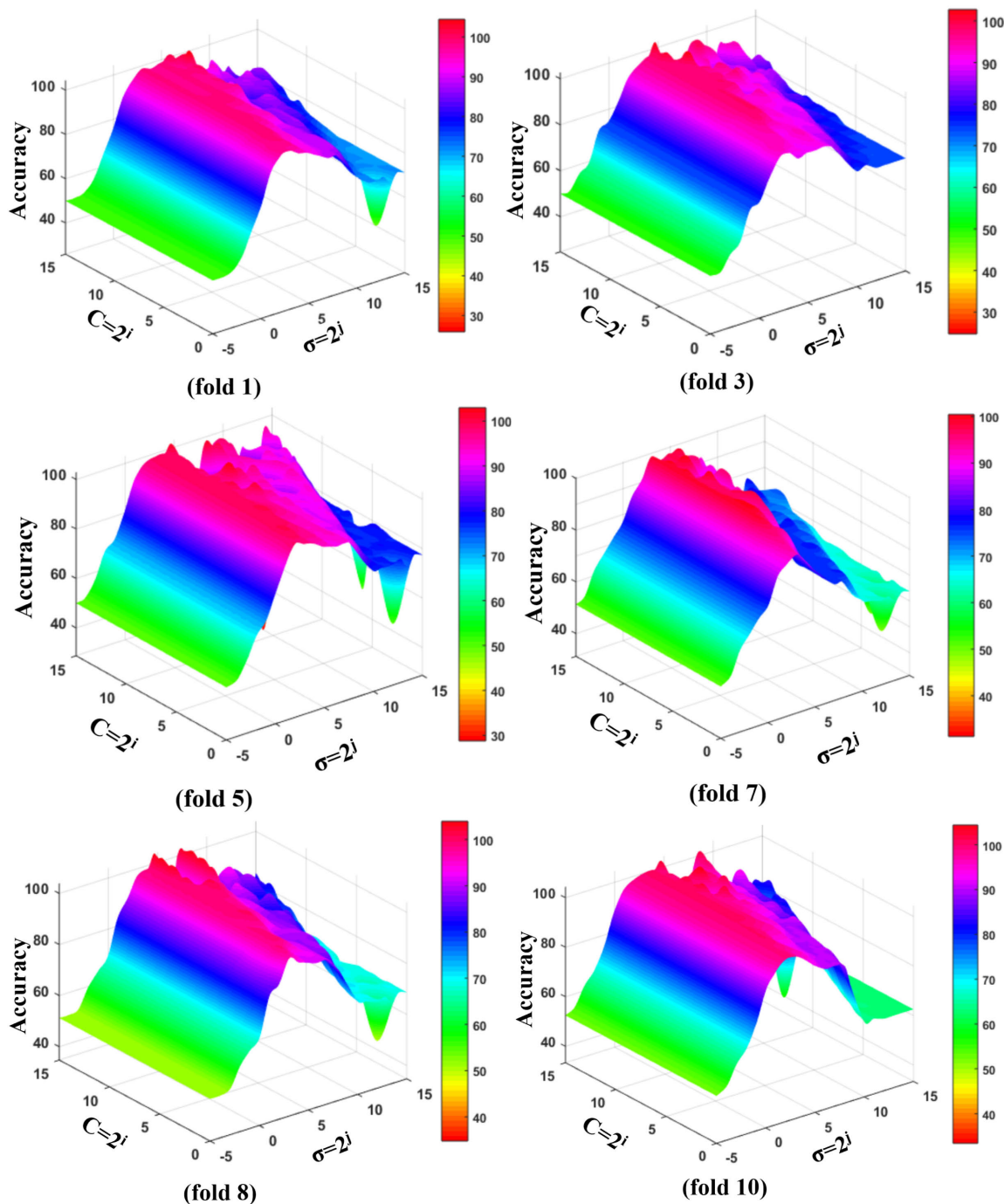


FIGURE 14. Acc variations based on learning parameters in some folds for TSP based on UOPFs set.

4vFSS with regarding same train-test scenario (ignoring only 0.25% less than TNR than ReliefF and FCBF; TNR equal to BMHFSS). The obtained results of Table 9 show that PITHS in the presence of 24-cycles of 18-variate trajectory (See Table 5) has better performance (Acc, TPR, and TNR) than mRMR, which uses 9-cycle of 4-variate trajectory features [13]. In the case of the FCBF, ReliefF, and BMHFSS methods, the selected cycles via PITHS leading to high Acc

and TPR than FCBF, ReliefF, and BMHFSS schemes (9-cycle of 3-variate trajectory features selected by FCBF, ReliefF, and BMHFSS [13]). Also, based on a 9 cycles observation window, PITHS has higher processing time (152.591 ms) than 4vFSS (mRMR: 68.793 ms, FCBF: 68.930 ms, ReliefF: 68.910 ms, BMHFSS: 52.948 ms), which uses a four-cycle observation window for TSP [13] (See Table 10). However, the processing time of PITHS causes the system operator to

TABLE 9. Results of TSP based on selected optimal cycles by 4vFSS schemes.

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

TABLE 10. Processing time for TSP based 4vFSS schemes.

4vFSS	Observation window (OW) in cycle / second	Processing time (OW + 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

have enough time to take corrective actions. For more details on the processing time of PITHS and 4vFSS, refer to Table 8 and Table 10 [13].

IV. CONCLUSION

This study aimed to achieve high prediction accuracy and low processing time on transient stability assessment based on data mining technology. In this paper, to satisfy coupled indices, the horizontally integrated feature selection scheme is introduced for finding discriminative transient point features. We propose the partial-injective trilateral hybrid scheme (PITHS), including the nested trilateral filter phase (NTFP) and nested trilateral wrapper phase. In NTFP, information theory-based analysis by relevance, interdependence, and redundancy (RIR) criteria and in NTWP, the supervised learning-based analysis by non-linear

support vector machine (SVM) applied on 28-variate trajectory features. According to two nested trilateral phases (2NTPs) mounted on the partial-injective scenario, the first univariate trajectory feature (UTF^1) is entered into the NTFP for selecting filter-OTPFs of UTF^1 (f^{-1} OTPFs). Next, the obtained f^{-1} OTPFs are fed to NTWP for finding filter-wrapper- 1 OTPFs (fw^{-1} OTPFs) as 1 UOPFs. After conducting the first round of PITHS, the new round is triggered by feeding the fw^{-1} OTPFs and UTF^2 to 2NTPs for finding fw^{-2} OTPFs (2 UOPFs). By conducting the 28th 2NTPs of PITHS on the last input (fw^{-27} OTPFs plus UTF^{28}), the survived fw^{-28} OTPFs are considered as final UOPFs (28 UOPFs). Finally, $1:28$ UOPFs set are tested to verify their efficacy for TSP based on the cross-validation technique. The obtained results show that the survived UOPFs based on the PITHS have a high performance (Acc 98.75%, TPR 98.75%, TNR 98.75%, and processing time of 152.591 ms) for TSP. Also, to evaluate the effectiveness of the PITHS, we compared it with the other vertically integrated feature selection schemes regarding the same train-test condition. The obtained results show that the selected UOPFs by PITHS have better performance than selected optimal features by mRMR, ReliefF, FCBF, and BMHFSS algorithms on TSP.

REFERENCES

- [1] C.-S.-G. Karavas, K. A. Plakas, K. F. Krommydas, A. S. Kurashvili, C. N. Dikaiaikos, 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.
- [2] 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.
- [3] 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 definitions," *IEEE Trans. Power Syst.*, vol. 19, no. 3, pp. 1387–1401, Aug. 2004.
- [4] 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.
- [5] F. R. Gomez, A. Rajapakse, U. Annakkage, and I. Fernando, "Support vector machine-based algorithm for post-fault transient stability status prediction using synchronized measurements," *IEEE Trans. Power Syst.*, vol. 26, no. 3, pp. 1474–1483, Aug. 2011.
- [6] 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.
- [7] A. Stief, J. R. Ottewill, and J. Baranowski, "Relief F-based feature ranking and feature selection for monitoring induction motors," in *Proc. Int. Conf. Methods Models Automat. Robot. (MMAR)*, Miedzydroje, Poland, 2018, pp. 171–176.
- [8] 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.
- [9] 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.
- [10] 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.
- [11] 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.
- [12] S. 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.
- [13] 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.
- [14] Y. Piao and K. H. Ryu, "A hybrid feature selection method based on symmetrical uncertainty and support vector machine for high-dimensional data classification," in *Proc. Asian Conf. Intell. Inf. Database Syst.*, Kanazawa, Japan, 2017, pp. 721–727.
- [15] 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.
- [16] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, pp. 273–297, Apr. 1995.
- [17] 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.
- [18] 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.
- [19] 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>
- [20] M. U. Khalid, F. Rashid, W. Akhtar, M. H. Ghauri, and O. Lateef, *Python Based Power System Automation in PSS/E*. Durham, NC, USA: Lulu, 2014.
- [21] C. Canizares, T. Fernandes, E. Geraldi, Jr., 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. PES-TR18, Aug. 2015.
- [22] S. B. Mosavi and S. 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, Iran, in 2019. He is currently an Assistant Professor with the Department of Electrical and Computer Engineering, Imam Khomeini International University-Buin Zahra Higher Education Center of Engineering and Technology, 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.

...