Advances in Artificial Intelligence and Robotics Research, 2026, 2(3), 147-161
https://www.scirp.org/journal/airr
ISSN Online: 3143-3995
ISSN Print: 3143-3987

Advanced Quantum Support Vector Machine Algorithm for Transient Stability Assessment

Junior Morel Angouah Massaga[1], Claudette Christiane Koupna Eko[2], Patrick Nounamo Dabou[3], Jacques Tagoudjeu[4]

[1]ICT-AI Technical Commission, National Technology Development Committee, Yaoundé, Cameroon
[2]EMET Technical Commission, National Technology Development Committee, Yaoundé, Cameroon
[3]Department of Computer Science & Telecommunications, University of Maroua, Maroua, Cameroon
[4]Department of Mathematics & Physical Science, National Advanced School of Engineering, University of Yaoundé 1, Yaoundé, Cameroon

Email: angouah.cndt1@gmail.com, mclaudette.cndt@gmail.com, patrick.nounamo-dabou@univ-maroua.cm, jacques.tagoudjeu@univ-yaounde1.cm

How to cite this paper: Massaga, J.M.A., Eko, C.C.K., Dabou, P.N. and Tagoudjeu, J. (2026) Advanced Quantum Support Vector Machine Algorithm for Transient Stability Assessment. *Advances in Artificial Intelligence and Robotics Research*, **2**, 147-161.
https://doi.org/10.4236/airr.2026.23009

Received: June 4, 2026
Accepted: September 6, 2026
Published: September 9, 2026

Abstract

The increasing use of smart sensors in electrical grids is generating a massive amount of data, creating new challenges for power systems professionals. To effectively analyze these large datasets for signs of instability, quantum algorithms are a promising solution. This work demonstrates that Advanced Quantum Support Vector Machines (AQSVMs) are a superior alternative to classical SVMs for the rapid and accurate assessment of dynamic stability from this data. We implement a quantum circuit to encode data, train the model, and measure qubits, enabling the AQSVM to evaluate the probability of stability for any given signal. Tested on massive labeled datasets generated from IEEE 39-bus and 68-bus systems, the AQSVM consistently achieved exceptional performance (e.g., 99.13% precision, 99.37% accuracy, 99.99% AUC, and 1.38% Logloss for the 68-bus case), outperforming its classical counterpart.

Keywords

Power System, Quantum Machine Learning, IEEE 39-Bus, IEEE 68-Bus, Quantum Programming, Transient Stability, Assessment

1. Introduction

1.1. Research Background

Power systems stability studies model electrical systems, focusing on synchronous machines with dynamic models to simulate various operating conditions [1]. Machine learning algorithms recognize data patterns without explicit programming

[2]. Advances in microcircuits enable real-time data collection from electrical networks, offering simple, rapid ML stability analysis. Classical systems with many degrees of freedom face dimensionality challenges, but quantum computing's superposition principle provides speedup and parallelism advantages that could benefit quantum machine learning for power stability [3], while recent reviews have highlighted the emerging role of quantum computing in power-system applications [4].

1.2. Literature Review

The ability of a power system to remain synchronized in the face of a severe disturbance, like a transmission line short circuit, is known as transient stability [5]. The system must perform transient stability analysis (TSA) and react promptly to prevent fault expansion to withstand a long-term fault [6]. We differentiate between time-domain [6]-[8], direct [6] [9] [10], and automatic learning approaches for evaluating transient stability in the literature. Time-domains take a lot of time [11], and when detailed models are considered, direct methods get more complicated. The latest study makes use of AI and automatic learning techniques. For this problem, several AI-based TSA strategies have been put forth. ANN-Based TSA [12] [13], SVM-Based [14] [15], TSA Ensemble Learning Based [16] [17], and TSA Deep Learning-Based [18] [19] are some examples of AI-based TSA methodologies. In [20], Yifan used quantum material to construct a quantum TSA method for quantum AI-based TSA and evaluated it against SVM and DNN models on four datasets. Some authors combine various CNN and long short-term memory (LSTM) configurations [10]-[12], and CNN with Gated Recurrent Units (GRU) [21]-[23] to increase accuracy. For online stability evaluation, Weiling proposes a modified support vector machine that can expedite the training stage [24]. An online ELM that can also pick features for TSA is presented by Yang Li [25].

1.3. Motivation and Contributions

Although considerable research has been conducted on transient stability assessment using classical computing and machine learning techniques, recent advances in quantum machine learning (QML) have created new opportunities for exploring quantum-enhanced classification approaches. In this context, this study investigates the application of an Advanced Quantum Support Vector Machine (AQSVM) to transient stability assessment and compares its performance with that of a classical Support Vector Machine (SVM).

In this study, the term Advanced Quantum Support Vector Machine (AQSVM) refers to an integrated quantum-classical workflow that combines quantum feature mapping, quantum-kernel computation, and a classical SVM with a precomputed kernel. The proposed approach uses generator-speed information obtained from transient stability simulations and evaluates the ability of the AQSVM and classical SVM to distinguish stable and unstable operating conditions.

The study focuses on evaluating the performance of the two approaches using the IEEE 39-bus and IEEE 68-bus benchmark systems. The comparison is performed using standard classification metrics, including accuracy, precision, AUC, and log loss. The objective is to assess the effectiveness of the integrated AQSVM workflow for transient stability assessment and to examine its performance relative to a classical SVM.

The remainder of this paper is organized as follows. Section II introduces the fundamentals of quantum support vector machines and the quantum-classical learning framework used in this study. Section III describes the experimental methodology and presents the results and their analysis. Finally, Section IV concludes the paper and discusses the main findings.

2. Quantum Support Vector Machine (QSVM)

2.1. Generality

Quantum computing is based on the fundamental principles of quantum mechanics. A simple way to convert QSVM to standard SVM is to apply projection over a Hilbert space, where we may express the qubit state as:

$$\begin{aligned} \varphi &: R^n \to S\left(2^q\right) \\ x &\to \left|\varphi(x)\right\rangle\left\langle\varphi(x)\right| \end{aligned} \tag{1}$$

where: $S\left(2^q\right)$ represents the density matrices [26] [27]. The Hilbert-Schmidt inner product gives the kernel function by:

$$\begin{aligned} k(x,y) &= tr\left[\left|\varphi(y)\right\rangle\left\langle\varphi(y)\right|\left|\varphi(x)\right\rangle\left\langle\varphi(x)\right|\right] \\ &= \left|\left\langle\varphi(y)\middle|\varphi(x)\right\rangle\right|^2 \end{aligned} \tag{2}$$

QSVM can be used in a different quantum paradigm. One of them involves estimating the quantum kernel and applying it to conventional SVM; the other involves utilizing quantum circuits; and the third involves quantum annealing.

- Quantum kernel estimation

To create distinct hyperplanes, we employ classical SVM with quantum kernels as opposed to variational quantum circuits. Initially, the phase quantum computer runs over the whole dataset to estimate the kernel $k(x,y)$ during training. Second, we estimate the kernel using all support vectors in the test dataset. Thirdly, by combining the quantum kernel with the classical SVM, we can make predictions.

- QSVM in the form of a quantum circuit

To construct QSVM with circuits, we must encode data using quantum states known as Ansatz circuits, as shown in **Figure 1**, using unitary or transformation functions. The circuit will then be trained using a variational quantum circuit with parameters, and we may measure the results to obtain classical bits.

- QSVM for quantum annealer

To use quantum annealer (QA), the computing problem must be formulated as a non-constrained quadratic binary optimization (QUBO) [28] [29]. The question of training traditional SVM in QA can be formulated as:

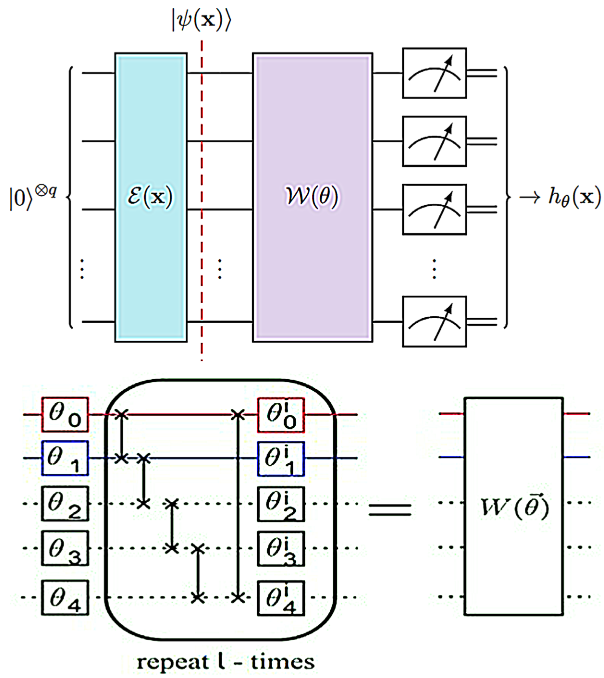


Figure 1. Representation of quantum support vector machine [26].

$$\begin{aligned}&\min E = \frac{1}{2}\sum_{nm} a_n a_m t_n t_m k\left(x_n, x_m\right) - \sum_n a_n \\ &\text{subject to } 0 < a_n < C \\ &\text{and } \sum a_n t_n = 0\end{aligned} \tag{3}$$

with $a_n \in \mathbb{R}$, K is the kernel function of SVM, and C is a regularization parameter. For QA implementation we use an encoding in the form of:

$$a_n = \sum_{k=0}^{k-1} B^k \sigma_{kn+k} \tag{4}$$

where $\sigma_{kn+k} \in \{0,1\}$, B is the base used for encoding and K is the number of binary variables used to encode a_n.

To transform quadratic problem given by (3)-(4) into QUBO, we use a_n expression and introduce a multiplier ξ to include last constraint $\sum_n a_n t_n = 0$ as a squared penalty term [30] and we obtain:

$$E = \sum_{n,m=0}^{N-1} \sum_{k,j=0}^{K-1} \sigma_{kn+k} \tilde{Q}_{kn+k,km+j} \sigma_{km+j} \tag{5}$$

where: $\tilde{Q}$ is a matrix of size KN x KN given by:

$$\tilde{Q}_{kn+k,km+j} = \frac{1}{2} B^{k+j} t_n t_m (k(x_n \xi x_m) - \sigma_{nm} \sigma_k jB^k \tag{6}$$

This formulation allows us to determine the lower energy that would be the QSVM solution.

2.2. Search Optimization Algorithms

The best settings are found using this method. In this study, we employ several optimization strategies to raise the model's effectiveness and performance. Early stopping prevents overfitting and conserves computational resources during cross-validation by terminating training when the F1 score does not increase for successive folds. Furthermore, threshold optimization is utilized to determine the appropriate classification threshold that maximizes the F1 score, guaranteeing optimal decision-making for datasets that are not balanced. To speed up processing, particularly with large datasets, joblib is used to parallelize the kernel matrix computation. To increase performance on unbalanced data, the Support Vector Classifier (SVC) is optimized with a precomputed kernel and class balancing. This allows for more accurate classification judgments by changing class weights and permitting probability outputs. All these adjustments are meant to improve the management of unbalanced data, decrease calculation time, and increase model validity.

2.3. Performance Evaluation

An effective method for comparing models is performance evaluation. Depending on the kind of problem or machine learning method, the measure is used to evaluate in the literature. Four metrics—accuracy, precision, AUC, and loss—are computed using a confusion matrix.

- Confusion matrix

The structure of Table 1 is utilized to compute the fundamental metrics used to assess the performance of the QSVM and SVM algorithms. For classification issues involving two or more classes, it is employed. It evaluates the classification quality by recording samples that are properly and mistakenly recognized for each class. Table 1 shows a binary classification confusion matrix.

Table 1. Confusion matrix.

Class	Positive: stable	Negative: unstable
Positive: stable	True Positive (TP)	False Negative (FN)
Negative: unstable	False Positive (FP)	True Negative (TN)

- Accuracy

Accuracy in classification issues refers to how many stable and unstable cases the model correctly predicts. The mathematical expression of accuracy is:

$$\text{accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}} \tag{7}$$

- Precision

The percentage of all positive classifications in the model that are truly positive is known as precision.

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{8}$$

- Area Under the ROC Curve (AUC)

When evaluating the performance of different models, AUC (Area Under the Curve) is a useful metric—if the dataset is balanced. The model that performs the best is usually the one with the biggest area under the curve, as illustrated in **Figure 2**.

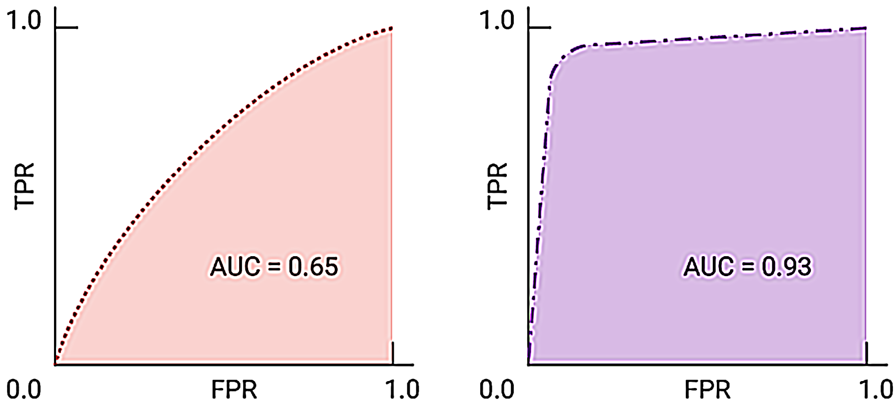


Figure 2. ROC curve with AUC metric.

2.4. Time Series Database for Quantum Machine Learning

There are more synchronous generators combined with each other via the power grid and act on each other, which leads to the complexity of transient stability analysis. The imbalance between the mechanical power Pm and the electromagnetic power P_e of the generator is associated with the variation of the rotor angles and speeds. The rotor speed signal thus influences P_m. When we predict transient stability, a set of elements $\vec{x} = (x_1, \cdots, x_n)$ should be selected to describe the state of power systems. Then, a dataset $D = \{(x_i, y_i) \mid i = 1, \cdots, n\}$ is obtained considering different load levels, network configurations, and contingencies by simulation or historical measurements. By training from massive data, the model can find a correspondence between the input characteristics x_i and the result of the transient stability analysis $f(x_i) = y_i$. The transient stability directly depends on the dynamics of the generator speed. When a fault occurs, the electromagnetic power will suddenly change. After the fault is eliminated, the speed will return to normal. In this research, the speeds of all generators (IEEE 39bus and IEEE 68bus) are used as input and the corresponding stability state is considered as output. In quantum machine learning, the encoding of data $x \in X$ in a quantum system is a mathematical mapping of the input set X to an output set Y of the quantum state. In pratique, we use parameterized quantum circuit also call Ansatz circuit to encode our dataset.

Let $U(\alpha, \beta, \gamma)$ our Ansatz circuit function with three parameters. Encode x with two qubits state would consist to operate the transformation:

$$\begin{aligned} U(\alpha, \beta)(x) &= U_\alpha\left(U_\beta(x)\right) \\ &= \sum_{\alpha\beta, i \in [2]} U_{\alpha\beta} \left| e_i \right\rangle \end{aligned} \tag{9}$$

where e_i could be 00,01,10,11.

Once the data is encoded, it can be used to train our AQSVM in the next section.

2.5. Time-Series Feature Extraction and Dimensionality Reduction

The generator-speed responses obtained from the transient stability simulations are initially represented as time-series data. Since the complete trajectories may contain a large number of highly correlated temporal samples, Principal Component Analysis (PCA) is applied as a dimensionality-reduction and feature-extraction technique before quantum encoding. PCA transforms the original generator-speed observations into a set of orthogonal principal components that capture the dominant variations in the simulated system responses.

For each simulated case, the PCA-transformed data are used to construct a fixed-length input vector for the classification models. The resulting feature representation contains 11 components for the IEEE 39-bus system and 17 components for the IEEE 68-bus system. These PCA-based feature vectors, rather than the complete raw generator-speed trajectories, are subsequently used as inputs to the quantum feature map. The same feature-extraction procedure is applied to the data used by both the AQSVM and the classical SVM to ensure a consistent and fair comparison.

Algorithm 1: Advanced Quantum Support Vector Machine

Require:
Training data $X \in \mathbb{R}^{n \times d}$, labels $y \in \{0,1\}^n$,
max qubits M, # folds K, patience P **Ensure :**
Best ansatz circuit, trained QSVM classifier $\mathcal{C}$,
optimal threshold τ^*
$X_{\text{proc}} \leftarrow \texttt{FeatureEngineering}(X)$;
Initialize best_mean_f1 $\leftarrow 0$, stagnation $\leftarrow 0$;
$\mathcal{M} \leftarrow [\,]$; // metrics per fold
for $\ell = 1$ **to** K ***and*** *stagnation* $< P$ **do**
  $(train\_idx, test\_idx) \leftarrow \texttt{StratifiedKFoldSplit}(X_{\text{proc}}, y, \ell)$;
  $X_{\text{tr}}, X_{\text{te}} \leftarrow X_{\text{proc}}[train\_idx], X_{\text{proc}}[test\_idx]$;
  $y_{\text{tr}}, y_{\text{te}} \leftarrow y[train\_idx], y[test\_idx]$;
  ; // Quantum Embedding
  $\Psi_{\text{tr}} \leftarrow [\text{feature\_map}(x) \mid x \in X_{\text{tr}}]$;
  $\Psi_{\text{te}} \leftarrow [\text{feature\_map}(x) \mid x \in X_{\text{te}}]$;
  ; // Kernel Computation
  $K_{\text{tr}} \leftarrow \texttt{ComputeKernel}(\Psi_{\text{tr}}, \Psi_{\text{tr}})$;
  $K_{\text{te}} \leftarrow \texttt{ComputeKernel}(\Psi_{\text{te}}, \Psi_{\text{tr}})$;
  ; // Train & Evaluate
  $\mathcal{C} \leftarrow \text{SVC}(\text{kernel} = \texttt{precomputed}, \dots)$;
  $\mathcal{C}.\text{fit}(K_{\text{tr}}, y_{\text{tr}})$;
  $p \leftarrow \mathcal{C}.\text{predict\_proba}(K_{\text{te}})[:, 1]$;
  $\hat{y} \leftarrow (p > 0.5)$;
  $m \leftarrow \texttt{EvaluateMetrics}(y_{\text{te}}, \hat{y}, p)$;
  Append m to $\mathcal{M}$;
  $\mu_f \leftarrow \frac{1}{|\mathcal{M}|} \sum_{m \in \mathcal{M}} m.\text{f1}$;
  if $\mu_f >$ *best_mean_f1* **then**
    best_mean_f1 $\leftarrow \mu_f$;
    stagnation $\leftarrow 0$;
  else
    stagnation $\leftarrow$ stagnation $+ 1$;
; // Threshold Optimization
Concatenate all p and y from $\mathcal{M}$ into (P, Y);
$\tau^* \leftarrow \underset{\tau \in [0.01, 0.99]}{\arg\max}\ \text{F1}(Y, P > \tau)$;
return Best metric ;

3. Experimental Result and Analysis

3.1. Experimental Dataset

The IEEE 39-bus and IEEE 68-bus systems are used as benchmark power networks to generate the transient stability datasets. The IEEE 39-bus system consists of 39 buses and 10 synchronous generators, whereas the IEEE 68-bus system consists of 68 buses and 16 generators. Both systems are modeled and simulated in MATLAB/Simulink using the Power Systems simulation environment.

For each test system, transient disturbances are introduced to generate different operating scenarios. Two types of disturbances are considered in this study: three-phase faults and line-to-ground faults. For each simulated scenario, the system response is recorded over a variable simulation horizon not exceeding 10 s. The resulting generator-speed trajectories are used to characterize the dynamic response of the system following each disturbance.

The stability status of each simulated case is determined from the outcome of the Power Systems solvers implemented in Simulink. Each simulation is therefore assigned a stability label according to the stability condition identified by the simulation solver. The resulting labeled simulations constitute the dataset used for the subsequent machine-learning classification task. An example of the simulated response is shown in **Figure 3**.

A total of 3825 samples with 11 input features were generated for the IEEE 39-bus system, while 4725 samples with 17 input features were generated for the IEEE 68-bus system. The resulting datasets are imbalanced, with the number of stable cases exceeding the number of unstable cases in both benchmark systems. This class imbalance is taken into consideration during the machine-learning stage through class-weight balancing and decision-threshold optimization.

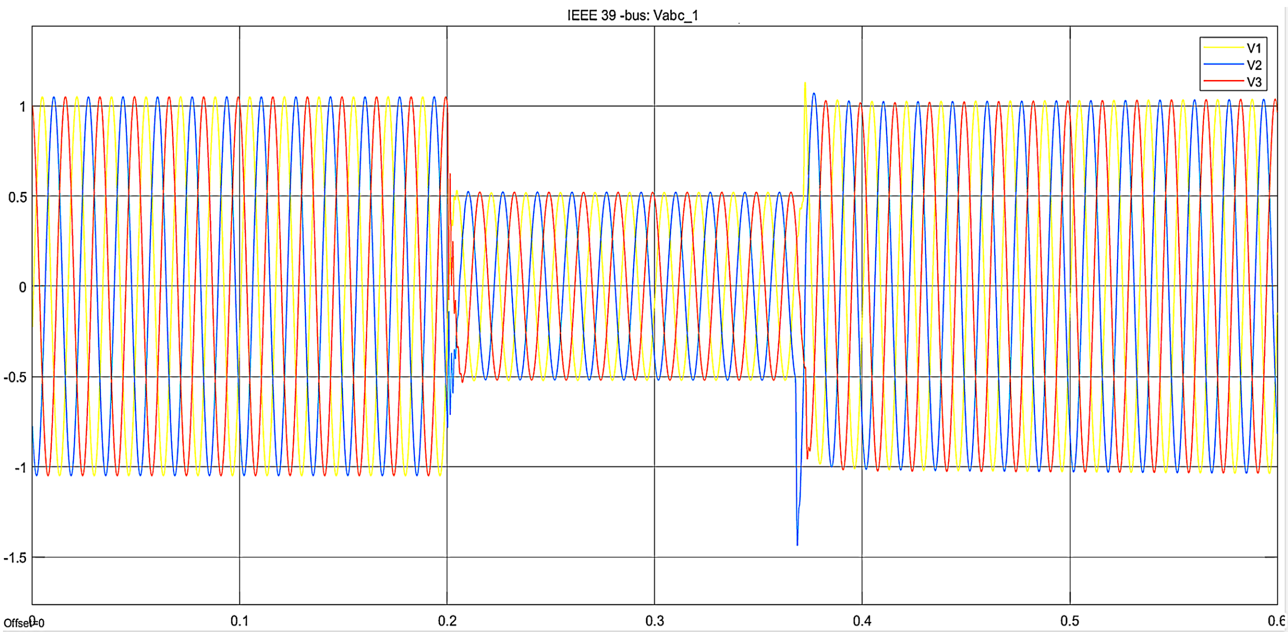


Figure 3. Three-phase voltages in EMT mode at bus 1 in IEEE 39-bus.

3.2. Experimental Environment

For the current power system stability analysis, we use an ASUS TUF GAMING F15 11th Gen Intel(R) Core (TM) i5-11400H @ 2.70 GHz 2.69 GHz with 8Go of computer RAM to run all our programs. All programs were written in MATLAB and Python. Anaconda 2016 edition tool was used to write the Python scripts. Keras was used to write the classical SVM programs, while Qskit was used for the implementation of quantum advanced SVM algorithms. The quantum component of the proposed AQSVM workflow is executed using a classical quantum simulator rather than physical quantum hardware. The quantum stage consists of encoding the PCA-based input features into quantum states through the selected quantum feature map and evaluating the corresponding quantum kernel. These quantum computations are simulated on a classical computer using the Qiskit-based implementation. The remaining stages of the workflow, including transient-stability simulation, PCA-based feature extraction, data shuffling and stratified cross-validation, SVM training, class weighting, decision-threshold optimization, and performance evaluation, are performed using classical computing procedures. Therefore, the reported AQSVM results represent the performance of a simulated quantum-kernel approach and should not be interpreted as measurements obtained from a physical quantum processor.

3.3. Results and Analysis

Before model training, the generated samples are randomly shuffled to remove any potential ordering effects. The shuffled dataset is then partitioned using stratified cross-validation, while preserving the relative proportions of stable and unstable cases across the folds. The number of cross-validation folds (K) is determined as part of the optimization procedure according to the characteristics of the generated dataset rather than being imposed as a fixed constant. For each fold, the training and validation subsets are kept separate during model fitting and prediction. The decision threshold is optimized using the prediction probabilities obtained from the validation procedure. This protocol accounts for the imbalanced nature of the datasets while allowing the cross-validation configuration to be adapted during the optimization process.

According to the experimental results, the suggested Advanced Quantum Support Vector Machine (AQSVM) outperforms the traditional SVM on several evaluation metrics on the IEEE 39-bus and IEEE 68-bus power systems. The precision and recall results demonstrate the overall effectiveness of the proposed AQSVM approach, although the performance varies between the two classification methods. The confusion matrices for the IEEE 39-bus and IEEE 68-bus systems presented in **Figure 4** and **Figure 5** support this. The AQSVM achieves precision values above 99% on both benchmark systems, while the classical SVM obtains lower precision and recall values. For the IEEE 39-bus system, the SVM achieves a precision of 94.36%, whereas for the IEEE 68-bus system, its recall reaches 94.20%. These results indicate that the AQSVM provides more consistent

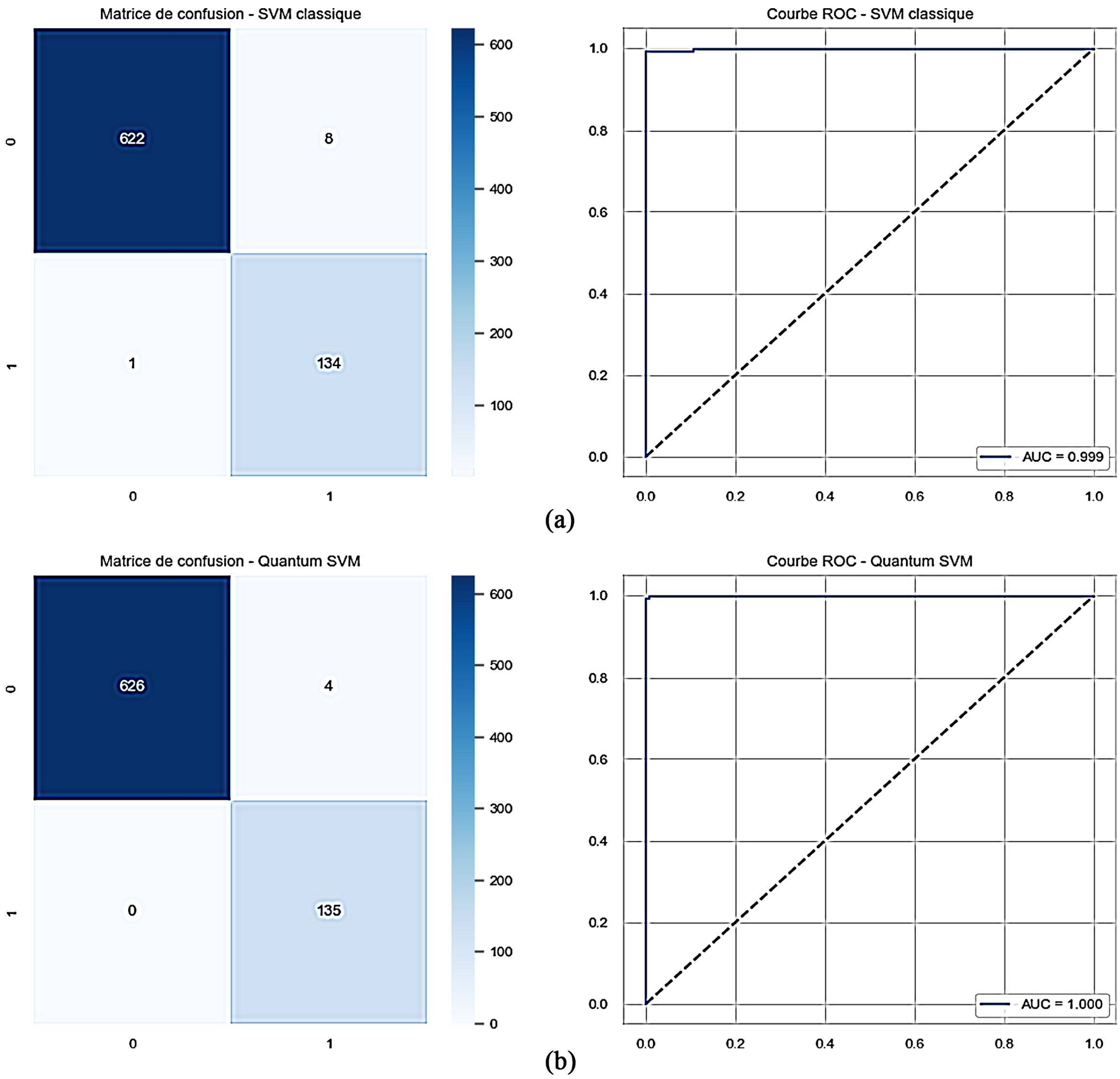
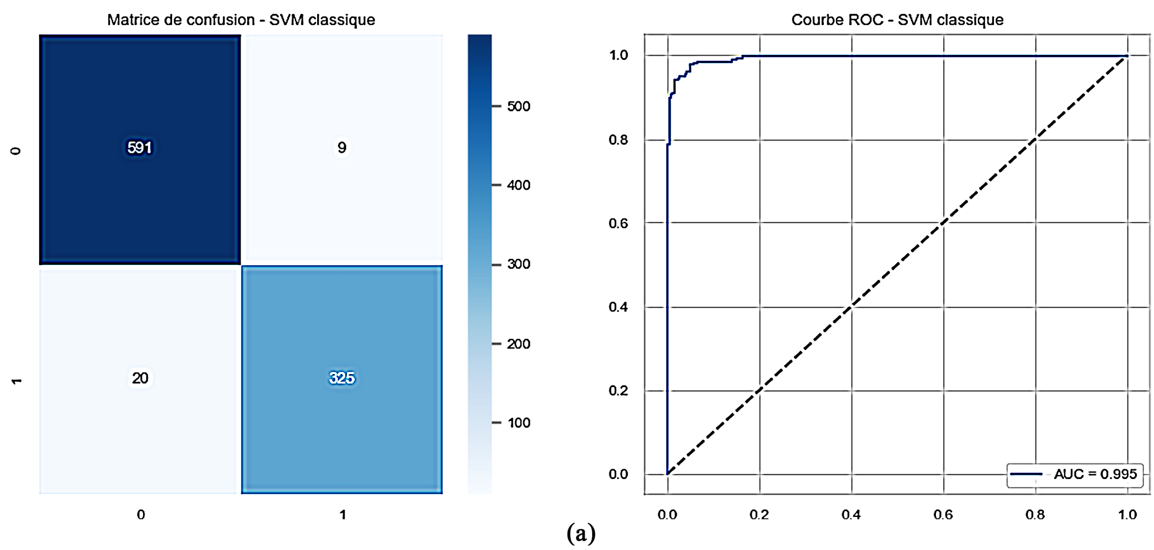
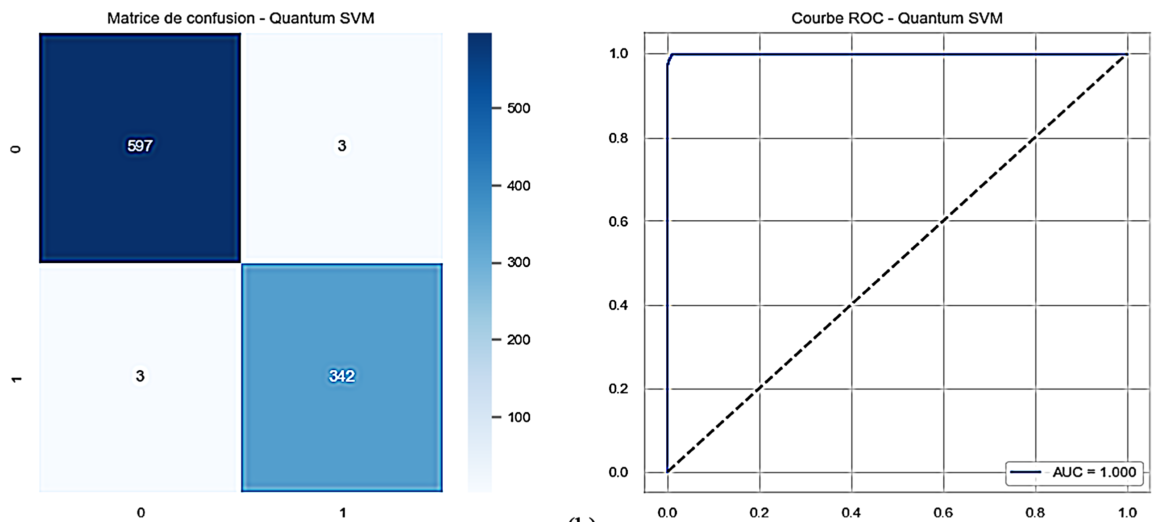


Figure 4. Confusion matrix (a) and SVM (b) AQSVM for IEEE 39-bus.





(b)

Figure 5. Confusion matrice: (a) SVM and (b) AQSVM for IEEE 68-bus.

classification performance than the classical SVM for the considered transient stability assessment datasets. The most notable distinction was seen in the log loss metric, where the AQSVM consistently produced well-calibrated probabilistic outputs by maintaining extremely low values (down to 0.0109). However, the log loss of the classical SVM was much larger (up to 0.1024), indicating less trustworthy prediction confidence. Together, these findings support the efficacy of quantum-based learning in managing challenging classification tasks in power system stability evaluation, providing improved performance scalability, generalization, and reliability. A quantitative comparison of the two approaches is provided in Table 2.

Table 2. Algorithms comparisons.

Test systems	Computer	Algorithm	Precision	Accuracy	auc	log_loss
IEEE 39-bus	Quantum	AQSVM	0.992592	0.997385	0.999858	0.010929
	Classical	SVM	0.943661	0.988235	0.999212	0.028914
IEEE 68-bus	Quantum	AQSVM	0.991304	0.993650	0.999893	0.013810
	Classical	SVM	0.973053	0.969312	0.995028	0.102443

4. Conclusion

The IEEE 39- and 68-bus networks are two common power system datasets on which we compared the performance of a Quantum Assisted Support Vector Machine (AQSVM) and a Classical Support Vector Machine (SVM). The AQSVM continuously beat the traditional SVM for both test systems in every evaluation criterion, as summarized in Figure 6. The quantum model demonstrated more confident and dependable predictions on the IEEE 39-bus system with a higher accuracy (99.73% vs. 98.82%) and a much lower Log Loss (0.0109 vs. 0.0289).

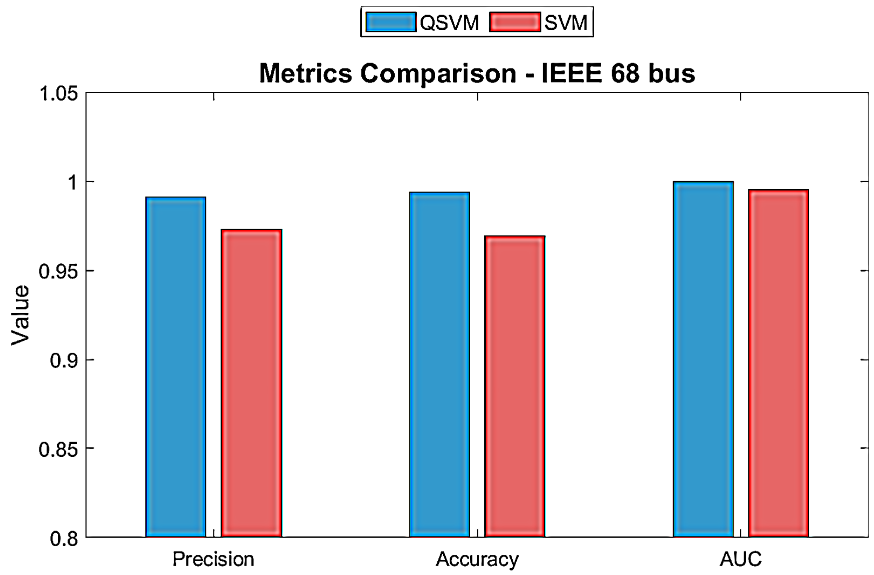


Figure 6. IEEE 68-bus metrics diagram.

The AQSVM remained superior to the classical SVM on the bigger IEEE 68-bus system, outperforming it in probabilistic calibration (Log Loss of 0.0138 vs. 0.1024) and accuracy (99.37% vs. 96.93%). These findings unequivocally show that, even in large-scale, very complex systems, quantum-enhanced machine learning models like AQSVM are not only competitive but also capable of outperforming conventional methods. The quantum model's improved capacity to better represent nonlinear patterns via quantum feature mapping is responsible for the improved performance. The increasing promise of quantum computing as a useful and potent instrument for AI-based evaluation tasks in power system applications is supported by these discoveries, which come as quantum hardware continues to advance.

Author Contributions

Conceptualization, Junior Morel Angouah Massaga, Claudette Christiane Koupna Eko, Patrick Nounamo Dabou, and Jacques Tagoudjeu; methodology, Junior Morel Angouah Massaga, Patrick Nounamo Dabou, and Jacques Tagoudjeu; software, Junior Morel Angouah Massaga; validation, Junior Morel Angouah Massaga, Claudette Christiane Koupna Eko, Patrick Nounamo Dabou, and Jacques Tagoudjeu; formal analysis, Junior Morel Angouah Massaga and Patrick Nounamo Dabou; investigation, Junior Morel Angouah Massaga and Claudette Christiane Koupna Eko; resources, Junior Morel Angouah Massaga and Jacques Tagoudjeu; data curation, Junior Morel Angouah Massaga; writing—original draft preparation, Junior Morel Angouah Massaga; writing—review and editing, Patrick Nounamo Dabou, Jacques Tagoudjeu, and Claudette Christiane Koupna Eko; visualization, Junior Morel Angouah Massaga and Claudette Christiane Koupna Eko; supervision, Patrick Nounamo Dabou and Jacques Tagoudjeu; project administration, Jacques Tagoudjeu; funding acquisition, Claudette Christiane Koupna Eko. All authors have read and agreed to the published version of the manuscript.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Dai, J.J. and Dourian, R. (2019) Standard Approach to Perform Power System Stability Studies in Oil and Gas Plants. Eaton Corporation, PCIC Europe Paper No. PCIC EUR19_04.

[2] El Naqa, I. and Murphy, M.J. (2015) What Is Machine Learning? In: El Naqa, I., Li, R. and Murphy, M., Eds., *Machine Learning in Radiation Oncology*, Springer, 3-11. https://doi.org/10.1007/978-3-319-18305-3_1

[3] Dabou, R.T. and Kamwa, I. (2020) Rapid Design Method for Generating Power System Stability Databases in SPS for Machine Learning. 2020 *IEEE Canadian Conference on Electrical and Computer Engineering* (*CCECE*), London, 30 August-2 September 2020, 1-6. https://doi.org/10.1109/ccece47787.2020.9255746

[4] Liu, H. and Tang, W. (2023) Quantum Computing for Power Systems: Tutorial, Review, Challenges, and Prospects. *Electric Power Systems Research*, **223**, Article ID: 109530. https://doi.org/10.1016/j.epsr.2023.109530

[5] Hatziargyriou, N., Milanovic, J., Rahmann, C., Ajjarapu, V., Canizares, C., Erlich, I., *et al.* (2021) Definition and Classification of Power System Stability—Revisited & Extended. *IEEE Transactions on Power Systems*, **36**, 3271-3281. https://doi.org/10.1109/tpwrs.2020.3041774

[6] Liu, W., Kerekes, T., Dragicevic, T. and Teodorescu, R. (2023) Review of Grid Stability Assessment Based on AI and a New Concept of Converter-Dominated Power System State of Stability Assessment. *IEEE Journal of Emerging and Selected Topics in Industrial Electronics*, **4**, 928-938. https://doi.org/10.1109/jestie.2023.3236885

[7] Liu, Y., Sun, K., Yao, R. and Wang, B. (2019) Power System Time Domain Simulation Using a Differential Transformation Method. *IEEE Transactions on Power Systems*, **34**, 3739-3748. https://doi.org/10.1109/tpwrs.2019.2901654

[8] Zadkhast, P., Jatskevich, J. and Vaahedi, E. (2015) A Multi-Decomposition Approach for Accelerated Time-Domain Simulation of Transient Stability Problems. *IEEE Transactions on Power Systems*, **30**, 2301-2311. https://doi.org/10.1109/tpwrs.2014.2361529

[9] Bhui, P. and Senroy, N. (2016) Real Time Prediction and Control of Transient Stability Using Transient Energy Function. *IEEE Transactions on Power Systems*, **32**, 923-934. https://doi.org/10.1109/tpwrs.2016.2564444

[10] Xu, Y., Dong, Z.Y., Zhang, R., Xue, Y. and Hill, D.J. (2015) A Decomposition-Based Practical Approach to Transient Stability-Constrained Unit Commitment. *IEEE Transactions on Power Systems*, **30**, 1455-1464. https://doi.org/10.1109/tpwrs.2014.2350476

[11] Karami, A. and Esmaili, S.Z. (2013) Transient Stability Assessment of Power Systems Described with Detailed Models Using Neural Networks. *International Journal of Electrical Power & Energy Systems*, **45**, 279-292. https://doi.org/10.1016/j.ijepes.2012.08.071

[12] Huang, J., Guan, L., Su, Y., Yao, H., Guo, M. and Zhong, Z. (2020) Recurrent Graph Convolutional Network-Based Multi-Task Transient Stability Assessment Framework in Power System. *IEEE Access*, **8**, 93283-93296. https://doi.org/10.1109/access.2020.2991263

[13] Luo, Y., Lu, C., Zhu, L. and Song, J. (2021) Data-Driven Short-Term Voltage Stability

Assessment Based on Spatial-Temporal Graph Convolutional Network. *International Journal of Electrical Power & Energy Systems*, **130**, Article ID: 106753. https://doi.org/10.1016/j.ijepes.2020.106753

[14] Wang, Q., Pang, C. and Alnami, H. (2021) Transient Stability Assessment of a Power System Using Multi-Layer SVM Method. 2021 *IEEE Texas Power and Energy Conference* (*TPEC*), College Station, 2-5 February 2021, 1-5. https://doi.org/10.1109/tpec51183.2021.9384918

[15] Hu, W., Lu, Z., Wu, S., Zhang, W., Dong, Y., Yu, R., *et al.* (2019) Real-Time Transient Stability Assessment in Power System Based on Improved SVM. *Journal of Modern Power Systems and Clean Energy*, **7**, 26-37. https://doi.org/10.1007/s40565-018-0453-x

[16] Liu, X., Zhang, X., Chen, L., Xu, F. and Feng, C. (2020) Data-Driven Transient Stability Assessment Model Considering Network Topology Changes via Mahalanobis Kernel Regression and Ensemble Learning. *Journal of Modern Power Systems and Clean Energy*, **8**, 1080-1091. https://doi.org/10.35833/mpce.2020.000341

[17] Sarajcev, P., Kunac, A., Petrovic, G. and Despalatovic, M. (2021) Power System Transient Stability Assessment Using Stacked Autoencoder and Voting Ensemble. *Energies*, **14**, Article 3148. https://doi.org/10.3390/en14113148

[18] Zhang, M., Li, J., Li, Y. and Xu, R. (2021) Deep Learning for Short-Term Voltage Stability Assessment of Power Systems. *IEEE Access*, **9**, 29711-29718. https://doi.org/10.1109/access.2021.3057659

[19] Azman, S.K., Isbeih, Y.J., Moursi, M.S.E. and Elbassioni, K. (2020) A Unified Online Deep Learning Prediction Model for Small Signal and Transient Stability. *IEEE Transactions on Power Systems*, **35**, 4585-4598. https://doi.org/10.1109/tpwrs.2020.2999102

[20] Zhou, Y. and Zhang, P. (2023) Noise-Resilient Quantum Machine Learning for Stability Assessment of Power Systems. *IEEE Transactions on Power Systems*, **38**, 475-487. https://doi.org/10.1109/tpwrs.2022.3160384

[21] Cheng, S., Yu, Z., Liu, Y. and Zuo, X. (2022) Power System Transient Stability Assessment Based on the Multiple Paralleled Convolutional Neural Network and Gated Recurrent Unit. *Protection and Control of Modern Power Systems*, **7**, Article No. 39. https://doi.org/10.1186/s41601-022-00260-z

[22] Zhan, X., Han, S., Rong, N., Liu, P. and Ao, W. (2022) A Two-Stage Transient Stability Prediction Method Using Convolutional Residual Memory Network and Gated Recurrent Unit. *International Journal of Electrical Power & Energy Systems*, **138**, Article ID: 107973. https://doi.org/10.1016/j.ijepes.2022.107973

[23] Wu, L., Kong, C., Hao, X. and Chen, W. (2020) A Short-Term Load Forecasting Method Based on GRU-CNN Hybrid Neural Network Model. *Mathematical Problems in Engineering*, **2020**, Article ID: 1428104. https://doi.org/10.1155/2020/1428104

[24] Zhang, W., Hu, W., Min, Y., Chen, L., Zheng, L. and Liu, X. (2015) A Novel Stability Classifier Based on Reformed Support Vector Machines for Online Stability Assessment. 2015 *IEEE PES Asia-Pacific Power and Energy Engineering Conference* (*APPEEC*), Brisbane, 15-18 November 2015, 1-5. https://doi.org/10.1109/appeec.2015.7380884

[25] Li, Y. and Yang, Z. (2017) Application of EOS-ELM with Binary Jaya-Based Feature Selection to Real-Time Transient Stability Assessment Using PMU Data. *IEEE Access*, **5**, 23092-23101. https://doi.org/10.1109/access.2017.2765626

[26] Havlíček, V., Córcoles, A.D., Temme, K., Harrow, A.W., Kandala, A., Chow, J.M., *et*

al. (2019) Supervised Learning with Quantum-Enhanced Feature Spaces. *Nature*, **567**, 209-212. https://doi.org/10.1038/s41586-019-0980-2

[27] Gentinetta, G., Thomsen, A., Sutter, D. and Woerner, S. (2024) The Complexity of Quantum Support Vector Machines. *Quantum*, **8**, 1225. https://doi.org/10.22331/q-2024-01-11-1225

[28] Willsch, D., Willsch, M., De Raedt, H. and Michielsen, K. (2020) Support Vector Machines on the D-Wave Quantum Annealer. *Computer Physics Communications*, **248**, Article ID: 107006. https://doi.org/10.1016/j.cpc.2019.107006

[29] Press, W.H., Teukolsky, S.A., Vetterling, W.T. and Flannery, B.P. (2007) Numerical Recipes: The Art of Scientific Computing. 3rd Edition, Cambridge University Press.

[30] Delilbasic, A., Cavallaro, G., Willsch, M., Melgani, F., Riedel, M. and Michielsen, K. (2021) Quantum Support Vector Machine Algorithms for Remote Sensing Data Classification. 2021 *IEEE International Geoscience and Remote Sensing Symposium IGARSS*, Brussels, 11-16 July 2021, 2608-2611. https://doi.org/10.1109/igarss47720.2021.9554802