

Article

Dynamics-Guided Support Vector Machines for Response Analysis of Steel Frame Under Sine Wave Excitation

Yao Wang * and Huaiman Li

School of Civil Engineering and Architecture, Three Gorges University, Yichang 443005, China;
202308590021032@ctgu.edu.cn

* Correspondence: 202208140021036@ctgu.edu.cn

Abstract: Classical explicit and implicit time integration methods, such as the central difference method and Newmark method, are widely used for dynamic response analysis of systems. However, their computational accuracy and stability are highly sensitive to the time step size. To address this issue, a novel dynamics-guided support vector machine (DG-SVM) method is proposed, which embeds an optimization process to reduce dependence on the time step size. Unlike traditional machine learning approaches, the DG-SVM model incorporates initial conditions and dynamic equilibrium equations at each time step as physical constraints, ensuring that inertial forces, damping forces, resistance forces, and external dynamics satisfy equilibrium without relying on system dynamic response data. Furthermore, a solution algorithm combining DG-SVM with static condensation and mode decomposition methods is developed to enhance computational efficiency for the analysis of multi-degree-of-freedom systems. The superior accuracy and reliability of the proposed method are validated using a three-story steel frame structure subjected to sinusoidal excitation, where the numerical results obtained by DG-SVM are compared with those computed from classical integration methods, with analytical solutions serving as benchmarks.

Keywords: dynamic response; steel frame; support vector machines; static condensation method; mode decomposition method



Academic Editor: Shaohong Cheng

Received: 29 March 2025

Revised: 18 April 2025

Accepted: 20 April 2025

Published: 22 April 2025

Citation: Wang, Y.; Li, H. Dynamics-Guided Support Vector Machines for Response Analysis of Steel Frame Under Sine Wave Excitation. *Buildings* **2025**, *15*, 1399. <https://doi.org/10.3390/buildings15091399>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Classical explicit and implicit time integration schemes, such as the central difference method and the Newmark method, are widely employed in the dynamic response analysis of complex systems. The mathematical frameworks underlying these schemes are rigorously constructed upon fundamental physical principles and theoretical approximations [1]. Such theoretical approximations confer unique advantages to these schemes in resolving specific structural dynamics challenges, albeit with inherent limitations stemming from stringent stability and accuracy requirements. In recent years, extensive research efforts have been dedicated to conducting comprehensive investigations into the stability and accuracy characteristics of classical time integration schemes [2–9]. Hassan [10] conducted an all-encompassing numerical experiment. The purpose was to assess the numerical errors of the most widely adopted schemes in structural dynamics applications and to investigate the suitability of each method under diverse excitation scenarios, damping conditions, and varying ranges of natural periods. The research revealed that neglecting the range of the system's natural periods and simply setting the time step as a constant fraction of the natural period could trigger substantial numerical errors. Different numerical methods

demonstrated distinct response characteristics in response to variations in the time step. In essence, the impact of the time step on different numerical methods varied significantly. Öztürk et al. [11] conducted a study on a single-degree-of-freedom cantilever column system to examine its behavior under pulse loads and earthquake records. Employing the α -operator splitting method and the central difference method, the research explored the impacts of various time integration schemes and different time step intervals on the outcomes. Through an analysis of the experimental data, it was revealed that with the central difference method, as the time step interval grew, the system's period decreased, the maximum displacement increased, and the method's stability worsened. Conversely, the α -operator splitting method exhibited adequate sensitivity in the pseudo-dynamic simulation algorithm. Bathe and Wilson [2], aiming to enhance stability and accuracy, optimized the direct integration method and formulated a criterion for time step selection. Through an assessment of the responses of undamped systems, they identified an approach to quantify numerical integration errors. The research indicated that when the ratio of the time step Δt to the period T ($\Delta t/T$) was less than approximately 0.01, the integration was relatively precise, and errors grew as this ratio increased. Under specific parameter configurations, the Newmark method exhibited the highest level of accuracy. Classical explicit and implicit time integration schemes have been extensively applied in structural dynamic analysis; however, their inherent limitations remain pronounced. Explicit schemes, such as the central difference method, are favored for their straightforward computational process and non-iterative nature; however, their stringent stability criteria typically necessitate smaller time steps to maintain numerical stability, thereby compromising computational efficiency. Furthermore, explicit schemes often demand supplementary numerical techniques to preserve solution accuracy when addressing nonlinear dynamics. Implicit schemes, such as the Newmark method, are characterized by superior numerical stability, the ability to accommodate larger time steps, and robustness in handling nonlinear dynamics; however, their computational complexity necessitates solving large-scale nonlinear systems, leading to elevated computational costs. Additionally, implicit schemes may induce numerical dissipation under specific conditions, thereby impairing the precision of high-frequency response predictions. Consequently, classical explicit and implicit time integration schemes entail inherent trade-offs between computational efficiency, numerical stability, and solution accuracy, posing difficulties in simultaneously addressing the multifaceted requirements of complex dynamic systems.

With the advancement of machine learning (ML) technologies, their advantages in addressing nonlinear high-dimensional data fitting have been increasingly applied to structural dynamic response prediction in recent years [12–15]. Currently, the development of various ML algorithms, including neural networks and support vector machines (SVMs), has addressed the limitations of classical time integration methods in terms of computational accuracy and stability. Although neural networks demonstrate strong nonlinear fitting capabilities in structural dynamic response prediction, their parameter-tuning processes are often complex and sensitive to initial conditions, which can easily lead to local optima [16–21]. Support vector machines (SVMs) exhibit a strong theoretical foundation and excellent performance in structural dynamic response analysis. However, SVMs require solving a quadratic programming problem, leading to an inefficient training process [22]. Least Squares Support Vector Machines (LSSVMs), as an improvement over SVM, aim to address the aforementioned limitations. By transforming the traditional quadratic programming problem into a system of linear equations, LSSVMs significantly reduce the computational complexity, making it more efficient for handling large-scale data [23]. Zhang et al. [24] developed a physics-informed convolutional neural network (PhyCNN) based on deep learning for data-driven structural seismic response analysis.

The PhyCNN is trained using limited seismic input–output data and physical constraints to construct a surrogate model, thereby enhancing model stability and reliability while reducing the need for large-scale training datasets. Kazemi et al. [25] proposed an ML model for predicting the seismic response of RC frame structures and evaluating their seismic performance. The model generated 92,400 data points through Incremental Dynamic Analyses (IDAs), covering 165 RC frame structures with varying numbers of stories and spans, while incorporating key structural features as input variables. To validate the model's generality and accuracy, a five-story RC frame structure was tested. The results demonstrated that the model accurately predicts the seismic performance of the five-story RC frame. Nguyen et al. [26] employed Artificial Neural Networks (ANNs) and XGB to develop an ML model for predicting the seismic response of steel frames. The model conducted 624 seismic response simulations on steel frames with 36 different structural features, using maximum inter-story drift and maximum top displacement as primary seismic response indicators. The results showed that the XGB model outperformed the ANN, with R^2 values of 0.975 and 0.962 for the test data, respectively. Shafghfard et al. [27] constructed a data-rich framework using 307 experimental data points from the literature between 2000 and 2022 to predict the compressive strength (CS) of steel-fiber-reinforced concrete (SFRC) under high temperatures. The study demonstrated that stacking techniques, as an accurate and adaptable attribute evaluation tool, can effectively predict the CS of SFRC at high temperatures. In the aforementioned studies, most ML models for structural dynamic response analysis typically rely on large quantities of high-quality structural dynamic response data as a training foundation. Although this data-driven approach enhances the model's predictive capability for specific issues to some extent, it inevitably introduces a high dependency on data scale and quality. This dependency directly limits the model's predictive performance and weakens its generalization ability, especially when the training data coverage is insufficient or data quality is biased, potentially leading to significant prediction deviations.

As data-driven models become increasingly reliant on high-quality data, the academic community has shifted considerable attention toward the study of physics-driven machine learning models [28,29]. These models integrate physical constraints (e.g., governing equations and boundary conditions) directly into machine learning frameworks, effectively enhancing the efficiency and accuracy of numerical computations and predictions under data-scarce or low-quality conditions. This approach not only reduces reliance on large-scale, high-fidelity data but also provides a novel pathway for simulating and predicting complex physical systems. Yu et al. [30] proposed a novel physics-informed machine learning framework that integrates recurrent neural networks (RNNs), Multilayer Perceptrons (MLPs), and physical prior knowledge to construct a dual-layer model combining data-driven learning capabilities with physical constraints. Numerical experiments demonstrated that, compared to purely data-driven methods, the hybrid model maintained high prediction accuracy even in data-scarce scenarios, reducing training costs by approximately 40%. Eshkevari et al. [31] proposed a physics-informed recurrent neural network model for predicting the dynamic response of multi-degree-of-freedom systems. The model can estimate system displacement, velocity, acceleration, and internal forces. Compared to existing models, it requires less training data while achieving higher accuracy. In existing studies, physics-driven machine learning models, driven by core physical principles, exhibit significant advantages such as independence from data, high computational accuracy, efficient training, strong robustness, and superior physical interpretability. These models effectively address the data scarcity challenges faced by traditional data-driven methods in high-dimensional and complex physical scenarios, demonstrating broad application potential and research value.

In summary, traditional numerical integration methods in dynamic response analysis are highly sensitive to the selection of time steps in terms of computational accuracy and numerical stability. While shorter time steps can improve computational accuracy, they significantly increase computational costs. Conversely, longer time steps may lead to numerical instability or inaccuracies. Additionally, machine learning methods for seismic response prediction heavily rely on large-scale, high-quality data, and their model-training processes often require substantial computational resources, further increasing practical application costs. Therefore, the limitations of traditional time integration methods and machine learning approaches in structural dynamic response analysis urgently need to be addressed. To address these challenges, this study proposes a novel dynamics-guided support vector machine (DG-SVM) method to mitigate the sensitivity of computational accuracy and stability to time steps in classical explicit and implicit time integration methods for structural dynamic response analysis, while overcoming the limitations of machine learning methods that heavily depend on structural dynamic response data and exhibit limited generalization capabilities. DG-SVM is a novel computational method whose core concept involves using the objective function of the support vector machine (SVM) model as the optimization target and incorporating the system's motion equations and initial conditions under dynamic loading to construct a physics-constrained optimization model. Through its embedded optimization process, DG-SVM effectively reduces the sensitivity of computational accuracy and stability to time steps. Furthermore, DG-SVM directly incorporates the system's initial conditions and dynamic equilibrium equations at each time step as physical constraints, ensuring the balance of inertial forces, damping forces, resistance forces, and external dynamic forces, thereby eliminating the need for any system dynamic response data. This study examines a three-story steel frame structure subjected to sinusoidal excitation and uses the corresponding analytical solution as a benchmark to investigate the sensitivity of the DG-SVM method, the central difference method, and the Newmark method (including the average acceleration method and linear acceleration method) under varying time steps, while evaluating the performance of the DG-SVM model. By comparing the computational accuracy of the DG-SVM method with the central difference method and the Newmark methods, the accuracy and reliability of the DG-SVM model in dynamic response prediction were validated. The results indicate that DG-SVM, as a dynamics-driven computational method, effectively overcomes the limitations of traditional time integration methods and machine learning approaches, providing an efficient and stable solution for structural dynamic response analysis. It should be noted that traditional ML-based pure data-driven methods are not considered in this study, as they require extensive dynamic response data to train models, which are outside the scope of this paper that only introduces a data-free ML-based method (i.e., DG-SVM). Table 1 presents a summary of the existing research.

Table 1. Summary of the state of the art.

Independent Variables	Target Variable	Input Data Sample Size	Train/Test	Model	Metrics	Reference
Ground acceleration (PGA, preprocessed acceleration time series)	Displacement	23 (from CESMD)	Training: 48%; validation: 17%; testing: 35%	Modified PhyCNN	Prediction error: 5%, confidence intervals: 3rd floor: 97%; roof: 93%	Zhang et al. [24]
Displacement, velocity, acceleration, restoring force	Displacement, velocity, restoring force	100 (from PEER motion database)	Training: 10% testing: 90%	PhyCNN; CNN	Correlation coefficients: PhyCNN: 0.95, 0.92, 0.87, 0.61; CNN: 0.60, 0.72, 0.66, 0.37	

Table 1. Cont.

Independent Variables	Target Variable	Input Data Sample Size	Train/ Test	Model	Metrics	Reference
Weight, aspect ratio, reinforcement ratio for beams and columns, story number, bay length, total height, fundamental period, record direction RSN number, spectral acceleration	Maximum inter-story drift ratio; spectral acceleration of M-IDA curves	92,400 (from IDA)	Training: 70%; validation: 15%; testing: 15%	ANNs; XGBoost	IDRmax prediction: ANNs: 95.7%	Kazemi et al. [25]
Ground motion intensity (e.g., PGA, PGV, PGD), earthquake and soil characteristics (e.g., magnitude, soil type), structural geometric configurations (e.g., number of stories N_s , number of bays N_b), 5% critical damped spectral accelerations at the first three natural periods	Maximum top drift; maximum inter-story drift;	22,464 (from nonlinear dynamic analysis)	Training: 70%; validation: 15%; testing: 15%	ANN; XGBoost	XGBoost $R^2 = 0.9865$ ANN $R^2 = 0.962$	Nguyen et al. [26]
Seismic wave amplitude range, physical information (e.g., structural mass, stiffness, etc.)	Acceleration, velocity, displacement response time histories of structural nodes	200 (from PEER motion database)	Training: 70%; validation: 15%; testing: 15%	Phy-Seisformer	MAE = 0.7942 MSE = 1.7645 R = 0.7714	Zhou et al. [28]
Nonlinear state variables, time step size, physical information (mass, damping, stiffness matrices)	Displacement, velocity, restoring force	20	Training: 80%; testing: 20%	Residual network		Guo et al. [19]

2. Mathematical Model Development of the Dynamics-Guided Support Vector Machine (DG-SVM)

Support vector machines (SVMs), originally proposed by Vapnik et al. [32] in the early 1990s based on statistical learning theory, represent a foundational advancement in machine learning. Its core advantages lie in efficiently handling high-dimensional feature space data while exhibiting stability and maintaining an excellent generalization performance even with limited sample sizes. SVMs construct predictive models by solving a complex quadratic programming problem, making the computational process relatively intricate. To circumvent the need for solving complex quadratic programming problems, an alternative mathematical model of SVM, known as Least Squares SVM (LS-SVM), replaces the inequality constraints with equality constraints and utilizes the L2 norm as the loss function [33]. This modification reduces the training process to solving a linear programming problem, significantly simplifying the training procedure of SVMs. Given a set of training samples $\{(x_i, y_i)\}_{i=1}^N$, where $x_i \in R^p$ denotes the input variable and $y_i \in R$ denotes the output variable, LS-SVM derives the model parameters $\omega \in R^h$ and $b \in R$ by solving the following optimization problem, enabling the construction of the prediction model $\hat{y}(x) = \omega^T \varphi(x) + b$:

$$\min_{\omega, b, e_k} J(\omega, e_i) = \frac{1}{2} \omega^T \omega + \frac{1}{2} \gamma \sum_{k=1}^N e_k^2 \quad (1)$$

$$\text{s.t. } y(x_k) = \omega^T \varphi(x_k) + b + e_k, k = 1, \dots, N.$$

In the equation, $\varphi(x_k) = [\varphi_1(x_k), \varphi_2(x_k), \dots, \varphi_h(x_k)] \in R^h$, where $\varphi(\cdot) : R^p \rightarrow R^h$ denotes a mapping function projecting the p -dimensional space onto a high-dimensional Hilbert space of h dimensions. Here, e_k represents the error variable, and γ denotes the regularization parameter.

According to the principles of structural dynamics, the motion equation for a single-degree-of-freedom (SDOF) system subjected to a dynamic load $p(t)$ at time t can be expressed as

$$m\ddot{u}(t) + c\dot{u}(t) + ku(t) = p(t) \quad (2)$$

s.t. $u(t_0) = 0, \dot{u}(t_0) = 0$.

In the equation, m denotes the mass of the single-degree-of-freedom (SDOF) system, c is the damping coefficient, k represents the stiffness coefficient, $u(t)$ describes the system's displacement at time t , $\dot{u}(t)$ corresponds to the velocity at time t , and $\ddot{u}(t)$ indicates the acceleration at time t .

The equation of motion (2) can be further simplified, resulting in the derivation of the control equation expressed as follows:

$$\ddot{u}(t) + 2\zeta\omega_n\dot{u}(t) + \omega_n^2u(t) = p(t)/m \quad (3)$$

In the equation, ζ represents the damping ratio of the system, while ω_n denotes the natural frequency of the system.

Based on the LS-SVM prediction model $\hat{y}(x) = \mathbf{w}^T\boldsymbol{\varphi}(x) + b$, the input variable x is considered the time variable t , and the output variable $y(x)$ is interpreted as the system displacement $u(t)$ at time t . Consequently, the displacement response prediction model for the SDOF system at time t is formulated as $\hat{u}(t) = \mathbf{w}^T\boldsymbol{\varphi}(t) + b$. Under the assumption that the high-dimensional feature mapping function $\boldsymbol{\varphi}(\cdot)$ is sufficiently differentiable, the first-order derivative of the displacement prediction model with respect to the time variable t yields the velocity response prediction model for the SDOF system at time t , expressed as $d\hat{u}(t)/dt = \mathbf{w}^T\dot{\boldsymbol{\varphi}}(t)$. Similarly, the second-order derivative leads to the acceleration response prediction model, given by $d^2\hat{u}(t)/dt^2 = \mathbf{w}^T\ddot{\boldsymbol{\varphi}}(t)$. Building upon this framework, given the external dynamic load $\{(t_i, p(t_i))\}_{i=0}^N$, the acceleration, velocity, and displacement response prediction models of the SDOF system at time t are substituted into Equation (3) alongside the initial conditions of the SDOF system. These models are subsequently integrated as constraints into the objective function $J(\mathbf{w}, e_i)$ of the LS-SVM, resulting in the formulation of the dynamics-guided support vector machine (PGSVM-SRPM) through the following optimization model:

$$\begin{aligned} \min_{\mathbf{w}, b, e_i} J(\mathbf{w}, e_i) &= \frac{1}{2}\mathbf{w}^T\mathbf{w} + \frac{1}{2}\gamma\sum_{i=1}^N e_i^2 \\ \text{s.t.} \quad &\begin{cases} \mathbf{w}^T\ddot{\boldsymbol{\varphi}}(t_i) + 2\zeta_n\omega_n\mathbf{w}^T\dot{\boldsymbol{\varphi}}(t_i) + \omega_n^2(\mathbf{w}^T\boldsymbol{\varphi}(t_i) + b) \\ \quad = (p(t_i)/m) + e_i, i=1, \dots, N \\ \mathbf{w}^T\boldsymbol{\varphi}(t_0) + b = u(t_0) \\ \mathbf{w}^T\dot{\boldsymbol{\varphi}}(t_0) = \dot{u}(t_0) \end{cases} \end{aligned} \quad (4)$$

The parameters $\mathbf{w}$ and b of the prediction model are determined using the Lagrange multiplier method, wherein the Lagrangian function is formulated to address optimization model (4), yielding the following equations:

$$\begin{aligned} L(\mathbf{w}, b, e_i, \alpha_i, \beta_1, \beta_2) &= J(\mathbf{w}, e_i) - \sum_{i=1}^N \alpha_i (\mathbf{w}^T\ddot{\boldsymbol{\varphi}}(t_i) + 2\zeta_n\omega_n\mathbf{w}^T\dot{\boldsymbol{\varphi}}(t_i) + \omega_n^2(\mathbf{w}^T\boldsymbol{\varphi}(t_i) + b) \\ &\quad - p(t_i)/m - e_i) - \beta_1 (\mathbf{w}^T\boldsymbol{\varphi}(t_0) + b - u(t_0)) - \beta_2 (\mathbf{w}^T\dot{\boldsymbol{\varphi}}(t_0) - \dot{u}(t_0)) \end{aligned} \quad (5)$$

In the equation, α_i , β_1 , and β_2 are Lagrange multipliers.

Based on the Karush–Kuhn–Tucker (KKT) conditions, the linear equation system, comprising Equations (6)–(11), is formulated as follows:

$$\frac{\partial L}{\partial \mathbf{w}} = 0 \rightarrow \mathbf{w} = \sum_{i=1}^{n_g} \alpha_i (\ddot{\boldsymbol{\varphi}}(t_i) + 2\zeta_n\omega_n\dot{\boldsymbol{\varphi}}(t_i) + \omega_n^2\boldsymbol{\varphi}(t_i)) + \beta_1\boldsymbol{\varphi}(t_0) + \beta_2\dot{\boldsymbol{\varphi}}(t_0) \quad (6)$$

$$\frac{\partial L}{\partial b} = 0 \rightarrow \sum_{i=1}^{n_g} \alpha_i \omega_n^2 + \beta_1 = 0 \quad (7)$$

$$\frac{\partial L}{\partial e_i} = 0 \rightarrow e_i = \frac{\alpha_i}{r} \quad (8)$$

$$\frac{\partial L}{\partial \alpha_i} = 0 \rightarrow \mathbf{w}^T \ddot{\boldsymbol{\varphi}}(t_i) + 2\zeta_n \omega_n \mathbf{w}^T \dot{\boldsymbol{\varphi}}(t_i) + \omega_n^2 (\mathbf{w}^T \boldsymbol{\varphi}(t_i) + b) - p(t_i)/m - e_i = 0 \quad (9)$$

$$\frac{\partial L}{\partial \beta_1} = 0 \rightarrow \mathbf{w}^T \boldsymbol{\varphi}(t_0) + b = u(t_0) \quad (10)$$

$$\frac{\partial L}{\partial \beta_2} = 0 \rightarrow \mathbf{w}^T \dot{\boldsymbol{\varphi}}(t_0) = \dot{u}(t_0) \quad (11)$$

By removing $\mathbf{w}$ and b from the linear equation system and employing the kernel method $K(t_i, t_j) = \boldsymbol{\varphi}^T(t_i)\boldsymbol{\varphi}(t_j)$, which incorporates its first- and second-order derivative forms to characterize the inner product between the high-dimensional feature mapping function and its derivatives, the matrix equation is expressed as follows:

$$\begin{bmatrix} H_{1,1} + \frac{1}{\gamma} & \cdots & H_{1,N} & B_{1,1} & D_{1,1} & \omega_n^2 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ H_{N,1} & \cdots & H_{N,N} + \frac{1}{\gamma} & B_{N,1} & D_{N,1} & \omega_n^2 \\ T_{1,1} & \cdots & T_{1,N} & [K_0^0] & [K_1^0] & 1 \\ G_{1,1} & \cdots & G_{1,N} & [K_0^1] & [K_1^1] & 0 \\ \omega_n^2 & \cdots & \omega_n^2 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_N \\ \beta_1 \\ \beta_2 \\ b \end{bmatrix} = \begin{bmatrix} p(t_1)/m \\ \vdots \\ p(t_N)/m \\ \mathbf{u}(t_0) \\ \dot{\mathbf{u}}(t_0) \\ 0 \end{bmatrix} \quad (12)$$

In this equation, $H_{i,j}$, $B_{i,1}$, $D_{i,1}$, 1 , $T_{1,j}$, j , and $G_{1,j}$ are defined as follows:

$$H_{i,j} = [K_2^2]_{t_i,t_j} + 2\zeta_n \omega_n ([K_1^2]_{t_i,t_j} + [K_2^1]_{t_i,t_j} + 2\zeta_n \omega_n [K_1^1]_{t_i,t_j}) + \omega_n^2 ([K_0^2]_{t_i,t_j} + [K_2^0]_{t_i,t_j} + \omega_n^2 [K_0^0]_{t_i,t_j}) + 2\zeta_n \omega_n^3 ([K_0^1]_{t_i,t_j} + [K_1^1]_{t_i,t_j}), i, j = 1, \dots, N \quad (13)$$

$$B_{i,1} = [K_0^2]_{t_i,t_0} + 2\zeta_n \omega_n [K_0^1]_{t_i,t_0} + \omega_n^2 [K_0^0]_{t_i,t_0}, i, j = 1, \dots, N \quad (14)$$

$$D_{i,1} = [K_1^2]_{t_i,t_0} + 2\zeta_n \omega_n [K_1^1]_{t_i,t_0} + \omega_n^2 [K_1^0]_{t_i,t_0}, i, j = 1, \dots, N \quad (15)$$

$$T_{1,j} = [K_2^0]_{t_0,t_j} + 2\zeta_n \omega_n [K_1^0]_{t_0,t_j} + \omega_n^2 [K_0^0]_{t_0,t_j}, i, j = 1, \dots, N \quad (16)$$

$$G_{1,j} = [K_1^1]_{t_0,t_j} + 2\zeta_n \omega_n [K_1^1]_{t_0,t_j} + \omega_n^2 [K_0^1]_{t_0,t_j}, i, j = 1, \dots, N \quad (17)$$

In the equation, the first- and second-order derivatives of the kernel function characterize the inner product relationships between the high-dimensional feature mapping function and its derivatives, as detailed in Equation (18). This study employs the Gaussian kernel function, whose formulation is presented in Equation (19):

$$[K_s^r]_{t_i,t_j} = \frac{\partial^{r+s} K(t_i, t_j)}{\partial (t_i)^r \partial (t_j)^s} = (\boldsymbol{\varphi}^{(s)}(t_j))^T \boldsymbol{\varphi}^{(r)}(t_i) \quad (18)$$

$$K(t_i, t_j) = \exp(-\frac{\|t_i - t_j\|^2}{2\sigma^2}) \quad (19)$$

The coefficients $\{\alpha_i\}_{i=1}^N$, β_1 , β_2 , and b are derived through the solution to matrix Equation (12). By substituting Equation (6) into the system's displacement response pre-

diction model $\hat{u}(t) = w^T \varphi(t) + b$, the dual form for calculating the dynamic response is established, with the explicit expression given as follows:

$$\hat{u}(t) = \sum_{i=1}^N \alpha_i \left([K_2^0]_{t,t_i} + 2\zeta_n \omega_n [K_1^0]_{t,t_i} + \omega_n^2 [K_0^0]_{t,t_i} \right) + \beta_1 [K_0^0]_{t,t_0} + \beta_2 [K_1^0]_{t,t_0} + b \quad (20)$$

The velocity and acceleration response prediction models of the system are derived by computing the first- and second-order derivatives of Equation (20) with respect to the time variable, as illustrated below:

$$\frac{d}{dt} \hat{u}(t) = \sum_{i=1}^N \alpha_i \left([K_2^1]_{t,t_i} + 2\zeta_n \omega_n [K_1^1]_{t,t_i} + \omega_n^2 [K_0^1]_{t,t_i} \right) + \beta_1 [K_0^1]_{t,t_0} + \beta_2 [K_1^1]_{t,t_0} \quad (21)$$

$$\frac{d^2}{dt^2} \hat{u}(t) = \sum_{i=1}^N \alpha_i \left([K_2^2]_{t,t_i} + 2\zeta_n \omega_n [K_1^2]_{t,t_i} + \omega_n^2 [K_0^2]_{t,t_i} \right) + \beta_1 [K_0^2]_{t,t_0} + \beta_2 [K_1^2]_{t,t_0} \quad (22)$$

Based on the aforementioned theory, it is evident that, for a given SDOF system with parameters such as mass m , damping coefficient c , and stiffness k , the DG-SVM training process solely relies on the external dynamic load $p(t)$ and does not require any system dynamic response data (such as displacement $u(t)$, velocity $\dot{u}(t)$, or acceleration $\ddot{u}(t)$). Consequently, in contrast to data-driven machine learning (ML) approaches, the DG-SVM model does not rely on any system's dynamic response data. Moreover, the DG-SVM model constitutes an optimization model (as expressed in Equation (4)), and the displacement $u(t)$ (Equation (20)), velocity $\dot{u}(t)$ (Equation (21)), and acceleration $\ddot{u}(t)$ (Equation (22)) prediction models derived from it represent optimal hyperplanes in a high-dimensional space, with the training process fundamentally serving as a parameter-optimization procedure. As a result, the size of the time step exerts a minimal effect on its computational accuracy. Furthermore, during the derivation of the DG-SVM model, no specific form for the external dynamic load $p(t)$ is required. Moreover, as demonstrated by Equation (12), provided that the values of $p(t)$ at discrete or continuous time steps are known, the DG-SVM model can still be solved. Hence, the DG-SVM model is applicable to any type of external excitation load that varies over time.

The static condensation method is employed to condense out the secondary degrees of freedom (DOFs) that have no mass, while retaining the primary DOFs that consider mass, resulting in a condensed equation of motion that involves only the primary DOFs. Then, the mode decomposition is applied to decouple the MDOF system into multiple SDOF systems. Each SDOF system can be solved using the DG-SVM model established in Section 2. Compared to directly solving the equations of motion for n DOFs, the method proposed in this study requires solving only multiple SDOF systems using the DG-SVM model. Therefore, the proposed DG-SVM model, combined with the static condensation method and the mode decomposition method, significantly enhances the computational efficiency of dynamic response analysis for MDOF systems.

Table 2 presents the inputs and outputs of the DG-SVM model. Specifically, the table details the key input parameters required by the model, including the mass matrix, damping matrix, and stiffness matrix, which describe the physical characteristics of the structure. Additionally, the table lists the model's output results, namely displacement, velocity, and acceleration, which reflect the dynamic responses of the system under dynamic loading. This table aims to provide readers with a more intuitive understanding of the construction logic and functional features of the DG-SVM model.

Table 2. Inputs and outputs of the DG-SVM model.

Category	Details
Inputs	Mass matrix Damping matrix Stiffness matrix Time Dynamic loads
Outputs	Displacement Velocity Acceleration

Algorithm 1 outlines the detailed implementation process of the proposed DG-SVM for a dynamic response analysis of structural systems.

Algorithm 1. Dynamic response prediction using DGSVM

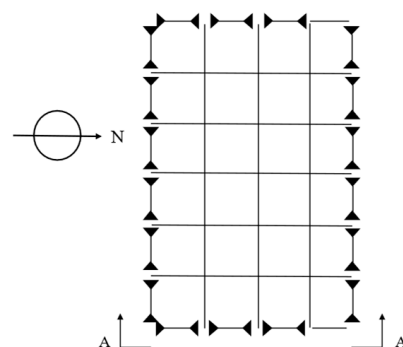
Inputs: system parameters (m, c, k); dynamic loads ($t, P(t)$); hyper-parameters (γ, σ_2); initial conditions ($u(t_0), \dot{u}(t_0)$)

- Calculate the H, B, D, T, G and I using the Equations (13)–(17);
- From the matrix Equation (12) based on $u(t_0), \dot{u}(t_0), H, B, D, T, G$ and $P(t)$;
- Solve the Equation (12) to obtain model parameters α, β_1, β_2 and b ;
- Compute the displacement response $\hat{u}(t)$, velocity response $\hat{\dot{u}}(t)$, and acceleration response $\hat{\ddot{u}}(t)$ using Equations (20)–(22);
- Record the computed dynamic response results $\hat{u}(t), \hat{\dot{u}}(t)$, and $\hat{\ddot{u}}(t)$;

Outputs: displacement ($\hat{u}(t)$), velocity ($\hat{\dot{u}}(t)$), acceleration ($\hat{\ddot{u}}(t)$)

3. Numerical Examples

To validate the predictive performance of the DG-SVM model, this study selects a three-story steel structure building designed as part of the SAC project in Los Angeles, California [34]. The building's floor plan is shown in Figure 1. The building has a total floor area of 6019.65 m², with each standard floor measuring 36.58 m in length, 54.87 m in width, and 3.96 m in height. The structural height of the main framework is 11.89 m. The structure has spans of 9.15 m in both principal directions, with six spans in the east–west direction and four spans in the north–south (N–S) direction. The elevation view of the framework is illustrated in Figure 2. It should be noted that the left corner column in this building may experience torsional response under a bidirectional loading as indicated in [35]. However, we only consider a unidirectional dynamic loading case, with one frame depicted in Figure 2 serving as the numerical example, as illustrated in [34].

**Figure 1.** Building plan of a three-story steel frame structure.

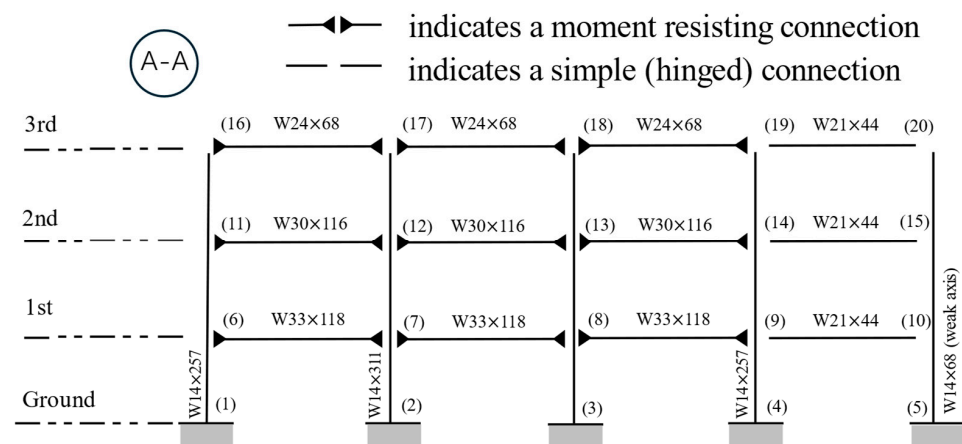


Figure 2. Building elevation of a three-story steel frame structure.

Figure 2 shows a three-story, four-span steel framework, where the columns are constructed using wide-flange steel with a yield strength of 345 MPa (50 ksi), and the beams are made of wide-flange steel with a yield strength of 248 MPa (36 ksi). The specific dimensions and types of beams and columns are detailed in Figure 2. Additionally, when constructing the finite element model of the steel framework, both the beams and columns are modeled using beam elements. The steel framework consists of 20 nodes, each with three degrees of freedom (DOFs). After excluding five boundary nodes, the total number of DOFs for the framework is 45. The mass matrix of the framework is established using the lumped mass method, where each node has mass only in the horizontal degree of freedom, while the vertical and rotational degrees of freedom have no mass. Consequently, as indicated in Section 2, the framework model has 15 primary degrees of freedom (DOFs) and 30 secondary DOFs. Subsequently, the static condensation method is applied to simplify these matrices into condensed equations of motion involving only the primary DOFs. Next, the condensed equations of motion are decomposed into 15 single-degree-of-freedom (SDOF) systems using the mode decomposition method. Finally, the DG-SVM model is employed to solve the governing equations of these 15 SDOF systems. All numerical simulations were implemented using Python 3.6.8.

3.1. Predictive Performance Quantification Metrics

For machine learning predictive models, performance evaluation metrics serve as a means to quantify the predictive capability of the model. In this study, the coefficient of determination (R^2), root mean square error (RMSE), and mean absolute error (MAE) are adopted to quantify the predictive performance of the proposed DG-SVM model. The corresponding calculation formulas are provided in Equations (23)–(25):

$$R^2 = 1 - \frac{\sum_{i=1}^N (y_i - \hat{y}_i)^2}{\sum_{i=1}^N (y_i - \bar{y})^2} \quad (23)$$

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2} \quad (24)$$

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i| \quad (25)$$

3.2. Sinusoidal Wave Excitation

In this study, a three-story steel frame structure subjected to sinusoidal excitation is used as an example. The corresponding analytical solution serves as the benchmark and is compared with the numerical solutions obtained from the proposed DG-SVM model,

the central difference method, the Newmark linear acceleration method, and the average acceleration method. This comparison aims to validate the accuracy of the DG-SVM model and assess the sensitivity of these four methods to different time step sizes. The sinusoidal excitation function is given as $p = 44.48\sin(\pi t/0.6)$.

Figures 3–5 illustrate the comparative time history responses of displacement, velocity, and acceleration for each floor of the structure obtained using the Newmark average acceleration method and the DG-SVM model with a time step size of 0.1 s. Figure 6 presents the comparison of the maximum inter-story drift angles obtained by the four methods and the analytical solution at various time step sizes, including 0.01 s, 0.02 s, 0.03 s, 0.04 s, 0.05 s, 0.06 s, 0.07 s, 0.08 s, 0.09 s, and 0.1 s. Figure 7 provides a detailed comparison of the performance metrics, including R^2 , RMSE, MAE, and computational time, for the displacement response obtained by the four methods during the dynamic time history analysis at various time step sizes.

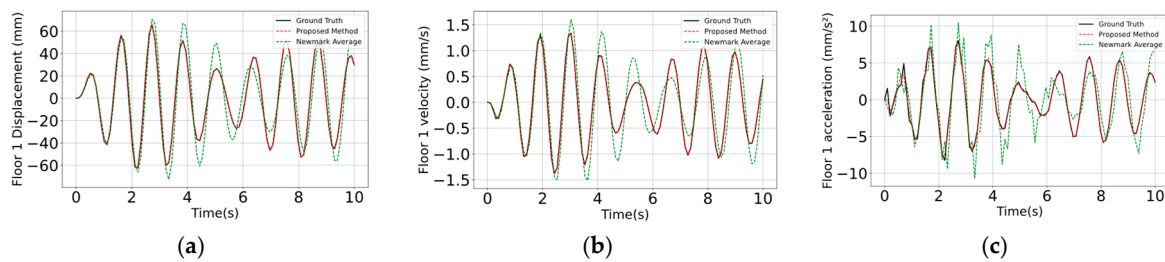


Figure 3. Comparison of displacement, velocity, and acceleration of the first floor structure: Newmark average acceleration method and DG-SVM method versus analytical solution. (a) Displacement; (b) velocity; (c) acceleration.

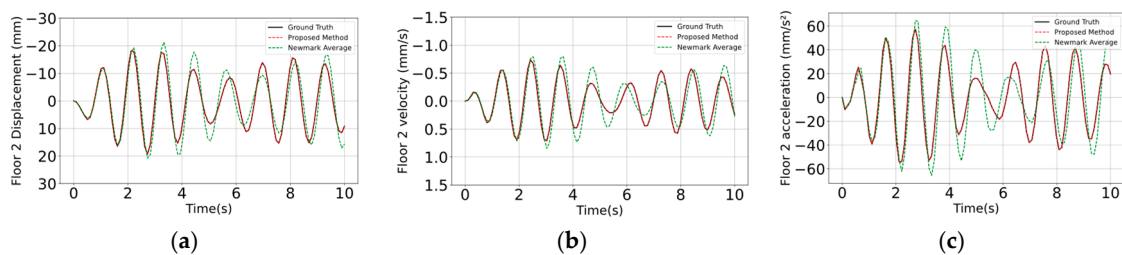


Figure 4. Comparison of displacement, velocity, and acceleration of the second floor structure: Newmark average acceleration method and DG-SVM method versus analytical solution. (a) Displacement; (b) velocity; (c) acceleration.

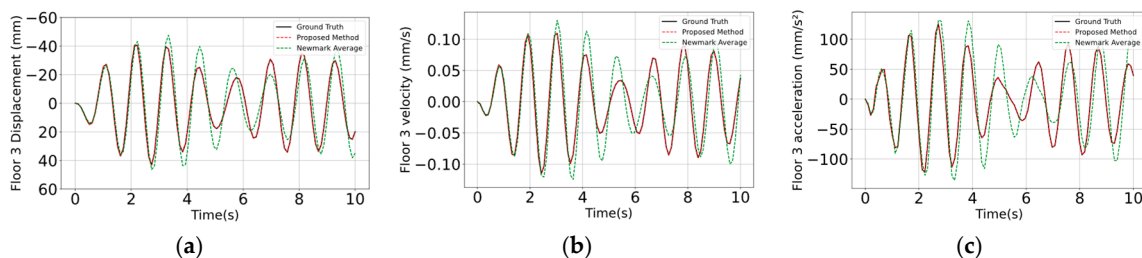


Figure 5. Comparison graphs of displacement, velocity, and acceleration of the third floor structure: Newmark average acceleration method and DG-SVM method versus analytical solution. (a) Displacement; (b) velocity; (c) acceleration.

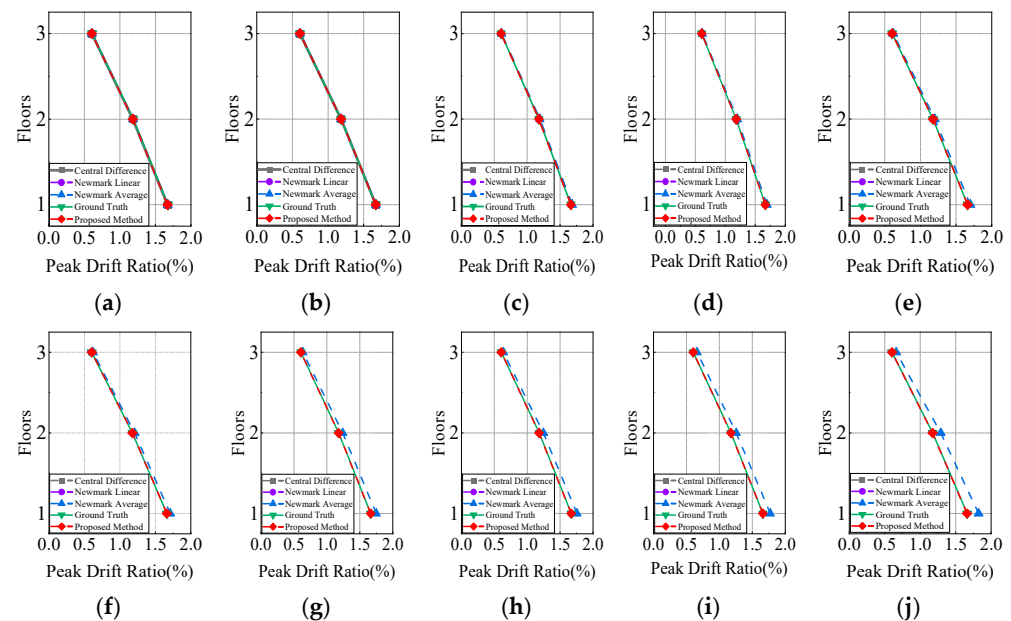


Figure 6. Comparison of peak drift ratio results of four methods under different time steps. (a) $\Delta t = 0.01$ s; (b) $\Delta t = 0.02$ s; (c) $\Delta t = 0.03$ s; (d) $\Delta t = 0.04$ s; (e) $\Delta t = 0.05$ s; (f) $\Delta t = 0.06$ s; (g) $\Delta t = 0.07$ s; (h) $\Delta t = 0.08$ s (i) $\Delta t = 0.09$ s (j) $\Delta t = 0.10$ s.

The analysis of Figures 3–5 indicates that, at a time step size of 0.1 s, the central difference method and the Newmark linear acceleration method fail to provide valid results, highlighting numerical stability issues that lead to divergence under larger time step sizes. In contrast, the DG-SVM method and the Newmark average acceleration method demonstrate good stability under the same time step size. Specifically, although the Newmark average acceleration method is unconditionally stable, its computed displacement, velocity, and acceleration time history responses exhibit significant deviations from the analytical solution, indicating poor computational accuracy. In comparison, the displacement, velocity, and acceleration time history responses obtained using the proposed DG-SVM model are nearly identical to the analytical solution, demonstrating that the model not only maintains numerical stability under larger time step sizes but also achieves a high computational accuracy.

The analysis of Figure 6 reveals that, when the time step size is less than or equal to 0.02 s, the DG-SVM method, Newmark linear acceleration method, Newmark average acceleration method, and central difference method produce results that are highly consistent with the analytical solution, indicating that these numerical methods can ensure high computational accuracy under smaller time step sizes. However, as the time step size increases, particularly when it reaches or exceeds 0.03 s, the performance of the Newmark linear acceleration method and the central difference method deteriorates significantly, failing to meet stability requirements and resulting in divergent solutions due to their conditional stability nature. In contrast, the Newmark average acceleration method continues to provide the maximum inter-story drift angles for each floor when the time step size increases to 0.03 s or above. However, as the time step size extends to 0.06 s, the deviation between its results and the analytical solution gradually increases, indicating that the computational accuracy of this method is affected by the time step size. In contrast, the proposed DG-SVM method exhibits robust stability and accuracy across various time step sizes, with its computational accuracy remaining unaffected by changes in the time step size. This demonstrates that the DG-SVM method is not only suitable for high-accuracy computations under small time step sizes but also capable of maintaining its computational accuracy and stability under larger time step sizes.

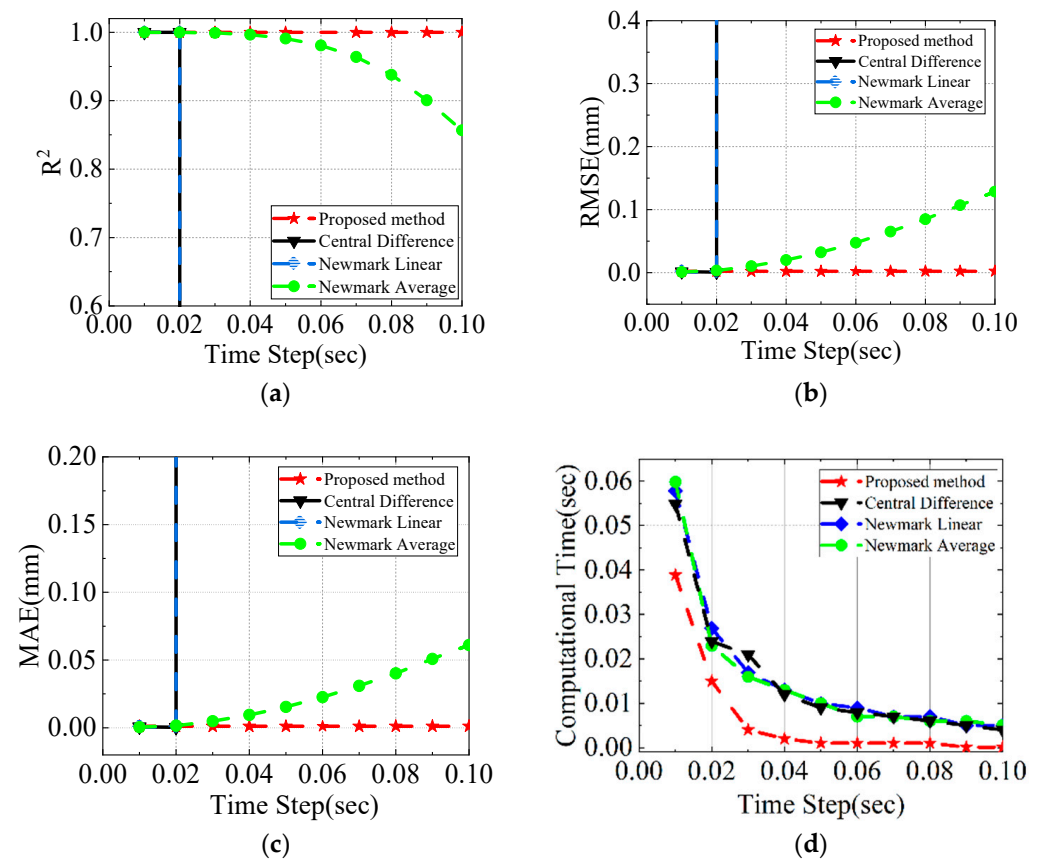


Figure 7. Comparison of calculation accuracy for displacement response results of four methods under different time steps. (a) R^2 ; (b) RMSE; (c) MAE; (d) time.

The analysis of Figure 7 reveals that the central difference method and the Newmark linear acceleration method encounter stability issues when the time step size exceeds 0.02 s, rendering their coefficient of determination (R^2), root mean square error (RMSE), and mean absolute error (MAE) values unreasonable. Figure 7a illustrates that the R^2 value of the Newmark average acceleration method decreases as the time step size increases, reaching 0.8576 at a time step size of 0.1 s, whereas the R^2 value of the proposed DG-SVM method consistently remains at 0.9999. Figure 7b illustrates that the RMSE value of the Newmark average acceleration method increases as the time step size grows, reaching 0.1287 mm at a time step size of 0.1 s, whereas the RMSE value of the DG-SVM method consistently remains at 0.0023 mm, indicating that the difference between its numerical solution and the analytical solution is minimally affected by the time step size. Figure 7c illustrates that the MAE value of the Newmark average acceleration method similarly increases with the time step size, reaching 0.0611 mm at a time step size of 0.1 s, whereas the DG-SVM method maintains a high computational accuracy across different time step sizes, with its MAE value consistently stabilized at 0.0011 mm. The above results demonstrate that the computational accuracy of the DG-SVM model is less sensitive to the time step size, whereas the computational accuracy of the central difference method and the Newmark methods is significantly affected by the time step size. Figure 7d indicates that the computational time of both the classical methods and the DG-SVM model decreases exponentially with increasing time step size. Among the four methods, the DG-SVM model consistently exhibits the lowest computational time across different time step sizes, demonstrating its superior computational efficiency compared to the classical methods. Compared to the classical methods, the DG-SVM model achieves approximately 1.4 times computational acceleration at a time step of 0.01 s. When the time step size increases to 0.1 s, the DG-

SVM model achieves approximately 40 times computational acceleration. Considering that the computational accuracy of the DG-SVM model at larger time step sizes (e.g., 0.1 s) is comparable to that of the classical methods at smaller time step sizes (e.g., 0.01 s) (see Figure 7a–c), the DG-SVM model provides approximately 540 times computational acceleration under these conditions. The following tables provide a comparative analysis of displacement responses among four methods under varying time steps. Specifically, Table 3 presents the R^2 values, Table 4 shows the mean absolute error (MAE) values, and Table 5 provides the root mean square error (RMSE) values. These metrics are used to evaluate the accuracy and reliability of each method in predicting displacement responses.

Table 3. Comparison of R^2 values for displacement responses among four methods under varying time steps.

Time Step (s)	DG-SVM	Central Difference	Newmark Linear	Newmark Average
0.01	0.999955	0.999977	0.999977	0.99999
0.02	0.999955	0.999993	0.999993	0.99990
0.03	0.999955	NaN	NaN	0.99907
0.04	0.999956	NaN	NaN	0.99661
0.05	0.999956	NaN	NaN	0.99088
0.06	0.999955	NaN	NaN	0.98080
0.07	0.999955	NaN	NaN	0.96404
0.08	0.999947	NaN	NaN	0.93791
0.09	0.999944	NaN	NaN	0.90073
0.1	0.999942	NaN	NaN	0.85679

Table 4. Comparison of MAE values for displacement responses among four methods under varying time steps.

Time Step (s)	DG-SVM	Central Difference	Newmark Linear	Newmark Average
0.01	0.001107	0.000793	0.00079306	0.000490
0.02	0.001112	0.000406	0.00040692	0.001599
0.03	0.001107	NaN	NaN	0.004878
0.04	0.001106	NaN	NaN	0.009502
0.05	0.001106	NaN	NaN	0.015413
0.06	0.001111	NaN	NaN	0.022619
0.07	0.001122	NaN	NaN	0.030993
0.08	0.001162	NaN	NaN	0.040176
0.09	0.001181	NaN	NaN	0.050815
0.1	0.001190	NaN	NaN	0.061115

Table 5. Comparison of RMSE values for displacement responses among four methods under varying time steps.

Time Step (s)	DG-SVM	Central Difference	Newmark Linear	Newmark Average
0.01	0.0022	0.0016	0.0016	0.001
0.02	0.0022	0.0008	0.0008	0.0033
0.03	0.0022	NaN	NaN	0.0103
0.04	0.0022	NaN	NaN	0.02
0.05	0.0022	NaN	NaN	0.0324
0.06	0.0023	NaN	NaN	0.0476
0.07	0.0022	NaN	NaN	0.0652
0.08	0.0023	NaN	NaN	0.0848
0.09	0.0023	NaN	NaN	0.1071
0.1	0.0023	NaN	NaN	0.1287

4. Limitations and Future Work

Although the proposed DG-SVM model has shown a good performance in addressing the limitations of traditional explicit and implicit time integration methods whose computational accuracy and stability heavily rely on small time step sizes and machine learning approaches whose performance mainly depend on system dynamic response data, its ability is limited to linear dynamic response analysis. This is because the equation of motion considered in this study is linear, represented by its resisting force that is a linear function of displacement with a constant stiffness value as the slope. For nonlinear systems, the resisting force is computed by varying stiffness values when the system is in the nonlinear phase. Therefore, to consider the nonlinearity, the equation of motion in Equation (2) has to be re-written in a general form that the resisting force is a nonlinear function of displacement. The overall mathematical derivations regarding Equations (4)–(22) have to be changed based on the nonlinear equation of motion. This limitation will be addressed in our future work.

Additionally, this study only considers the sinusoidal excitation because of its analytical solution that is available. For more complicated random excitations that cannot be expressed by a continuous function, such as ground motions, there is no analytical solution provided. In this sense, if ground motions are employed as dynamic excitation, there is no analytical solution that serves as the benchmark to compare with numerical solutions computed by classical time integration methods. However, the proposed DG-SVM model still works for these complicated random excitations. To demonstrate this, the proposed DG-SVM model is utilized to compute the response spectrum for 20 ground motion records with a time step size of 0.05. The benchmark is obtained by the Newmark average acceleration method with a smaller time step size of 0.01. The comparison of results is presented below. By observation, the results obtained by the proposed DG-SVM model agree very well with those computed by the Newmark average acceleration method, demonstrating its practical application. The comparison in Figure 8 shows the ground motion response spectra obtained using the DG-SVM method and the Newmark average method, along with the actual seismic response.

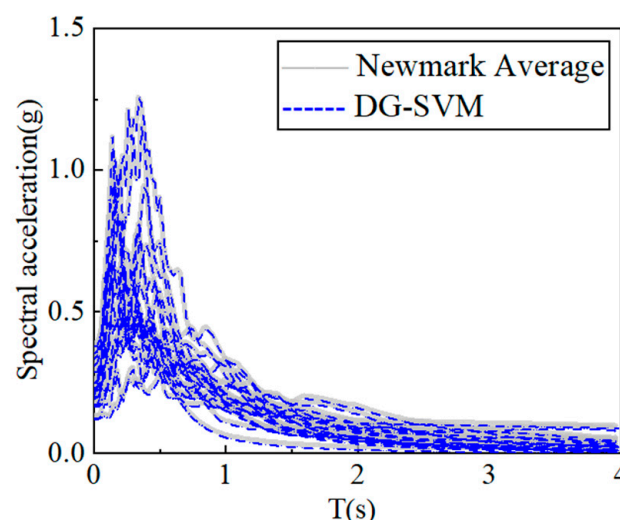


Figure 8. Comparison of Ground Motion Response Spectra: DG-SVM vs. Newmark average.

5. Conclusions

This study introduces a novel dynamics-guided support vector machine (DG-SVM) model for structural dynamic response analysis, addressing the limitations of traditional

explicit and implicit time integration methods and machine learning approaches. The key findings of this study are summarized as follows:

- (1) Low sensitivity to time step size: The DG-SVM model demonstrates low sensitivity to time step size in terms of computational accuracy. It provides high-accuracy results even with larger time step sizes, overcoming the limitations of classical central difference and Newmark methods, which exhibit poor precision and reliability under larger time step sizes.
- (2) Independence from structural dynamic response data: Unlike traditional machine learning methods that heavily rely on extensive structural dynamic response data, the DG-SVM model utilizes dynamic equilibrium equations and initial conditions as physical constraints. This eliminates the dependency on structural response data and significantly enhances the practicality and generalization capabilities of the proposed method.
- (3) High computational efficiency and applicability: the DG-SVM model achieves high computational accuracy and efficiency, particularly when applied to model order reduction for multi-degree-of-freedom (MDOF) systems with diagonal mass matrices and classical damping (e.g., Rayleigh damping).

This study demonstrates that the DG-SVM model effectively balances accuracy, stability, and computational efficiency, making it a promising tool for structural dynamic response analysis in practical engineering applications.

Author Contributions: Conceptualization, Y.W.; methodology, Y.W.; software, Y.W.; validation, Y.W.; formal analysis, H.L.; investigation, Y.W.; resources, H.L.; writing—original draft preparation, Y.W.; writing—review and editing, Y.W.; supervision, Y.W.; project administration, Y.W.; funding acquisition, Y.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data presented in this study are available upon request from the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Powell, M.J.D. *Approximation Theory and Methods*; Cambridge University Press: Cambridge, UK, 1981.
2. Bathe, K.J.; Wilson, E.L. Stability and accuracy analysis of direct integration methods. *Earthq. Eng. Struct. Dyn.* **1972**, *1*, 283–291. [[CrossRef](#)]
3. Chen, C.; Ricles, J.M. Development of direct integration algorithms for structural dynamics using discrete control theory. *J. Eng. Mech.* **2008**, *134*, 676–683. [[CrossRef](#)]
4. Fung, T.C. Stability and accuracy of differential quadrature method in solving dynamic problems. *Comput. Methods Appl. Mech. Eng.* **2002**, *191*, 1311–1331. [[CrossRef](#)]
5. Liang, X.; Mosalam, K.M. Lyapunov stability and accuracy of direct integration algorithms applied to nonlinear dynamic problems. *J. Eng. Mech.* **2016**, *142*, 04016022. [[CrossRef](#)]
6. Daniel, W.J.T. The subcycled Newmark algorithm. *Comput. Mech.* **1997**, *20*, 272–281. [[CrossRef](#)]
7. Ingles, J.; Darrozes, J.; Soula, J.C. Effects of the vertical component of ground shaking on earthquake-induced landslide displacements using generalized Newmark analysis. *Eng. Geol.* **2006**, *86*, 134–147. [[CrossRef](#)]
8. Sun, B.; Xuan, J.; Gao, L.; Wang, K.; Ou, J. Improved State-Space Approach Based on Lumped Mass Matrix for Transient Analysis of Large-Scale Locally Nonlinear Structures. *Buildings* **2024**, *14*, 2735. [[CrossRef](#)]
9. Wang, Y.; Zhao, D.; Li, H. Finite Element Model Updating Technique for Super High-Rise Building Based on Response Surface Method. *Buildings* **2025**, *15*, 126. [[CrossRef](#)]
10. Hassan, W.M. Numerical experiments on time step selection in linear dynamic analysis of structures. *J. Earthq. Eng.* **2021**, *25*, 1031–1077. [[CrossRef](#)]
11. Öztürk, S.; Doran, B.; Aksoylar, C. Effect of time step interval and time integration schemes in Pseudo-Dynamic analysis. *Structures* **2023**, *50*, 215–230. [[CrossRef](#)]

12. Oh, B.K.; Park, Y.; Park, H.S. Seismic response prediction method for building structures using convolutional neural network. *Struct. Control Health Monit.* **2020**, *27*, e2519. [[CrossRef](#)]
13. Alhalil, I.; Gullu, M.F. Predicting Main Characteristics of Reinforced Concrete Buildings Using Machine Learning. *Buildings* **2024**, *14*, 2967. [[CrossRef](#)]
14. Li, L.; Huang, X.; Chen, S.; Wu, T.; Mei, L.; Long, W.; Xiao, Y. Study on strategies for reducing training samples for accurate estimation of wind-induced structural response of LSTM networks. *J. Wind. Eng. Ind. Aerodyn.* **2023**, *238*, 105421. [[CrossRef](#)]
15. Kaffashchian, M.J.; Salkhordeh, M.; Karami Mohammadi, R. A boosted deep learning-based approach for near real-time response estimation of structures under ground motion excitations. *Struct. Infrastruct. Eng.* **2024**, *20*, 1–20. [[CrossRef](#)]
16. Hao, W.; Yin, H.; Liu, J.; Ma, R.; Zhao, D.; Lu, S. Seismic response prediction of soil tunnel structure based on CNN-LSTM model. *Structures* **2025**, *73*, 108381. [[CrossRef](#)]
17. Xu, Z.; Chen, J.; Shen, J.; Xiang, M. Recursive long short-term memory network for predicting nonlinear structural seismic response. *Eng. Struct.* **2022**, *250*, 113406. [[CrossRef](#)]
18. Wang, J.; Xie, Y.; Guo, T.; Du, Z. Predicting seismic interaction effect between soil and structure group using convolutional neural network. *Structures* **2023**, *50*, 1011–1020. [[CrossRef](#)]
19. Guo, J.; Enokida, R.; Li, D.; Ikago, K. Combination of physics-based and data-driven modeling for nonlinear structural seismic response prediction through deep residual learning. *Earthq. Eng. Struct. Dyn.* **2023**, *52*, 2429–2451. [[CrossRef](#)]
20. Peng, H.; Yan, J.; Yu, Y.; Luo, Y. Time series estimation based on deep Learning for structural dynamic nonlinear prediction. *Structures* **2021**, *29*, 1016–1031. [[CrossRef](#)]
21. Kim, T.; Kwon, O.S.; Song, J. Deep learning based seismic response prediction of hysteretic systems having degradation and pinching. *Earthq. Eng. Struct. Dyn.* **2023**, *52*, 2384–2406. [[CrossRef](#)]
22. Lin, K.Y.; Lin, T.K.; Lin, Y. Real-time seismic structural response prediction system based on support vector machine. *Earthq. Struct.* **2020**, *18*, 163–170.
23. Tezcan, J.; Marin-Artieda, C.C. Least-Square-Support-Vector-Machine-based approach to obtain displacement from measured acceleration. *Adv. Eng. Softw.* **2018**, *115*, 357–362. [[CrossRef](#)]
24. Zhang, R.; Liu, Y.; Sun, H. Physics-guided convolutional neural network (PhyCNN) for data-driven seismic response modeling. *Eng. Struct.* **2020**, *215*, 110704. [[CrossRef](#)]
25. Kazemi, F.; Asgarkhani, N.; Jankowski, R. Machine learning-based seismic response and performance assessment of reinforced concrete buildings. *Arch. Civ. Mech. Eng.* **2023**, *23*, 94. [[CrossRef](#)]
26. Nguyen, H.D.; Dao, N.D.; Shin, M. Prediction of seismic drift responses of planar steel moment frames using artificial neural network and extreme gradient boosting. *Eng. Struct.* **2021**, *242*, 112518. [[CrossRef](#)]
27. Shafighfard, T.; Bagherzadeh, F.; Rizi, R.A.; Yoo, D.-Y. Data-driven compressive strength prediction of steel fiber reinforced concrete (SFRC) subjected to elevated temperatures using stacked machine learning algorithms. *J. Mater. Res. Technol.* **2022**, *21*, 3777–3794. [[CrossRef](#)]
28. Zhou, Y.; Meng, S.; Lou, Y.; Kong, Q. Physics-informed deep learning-based real-time structural response prediction method. *Engineering* **2024**, *35*, 140–157. [[CrossRef](#)]
29. Luo, H.; Paal, S.G. A data-free, support vector machine-based physics-driven estimator for dynamic response computation. *Comput.-Aided Civ. Infrastruct. Eng.* **2023**, *38*, 26–48. [[CrossRef](#)]
30. Yu, Y.; Yao, H.; Liu, Y. Structural dynamics simulation using a novel physics-guided machine learning method. *Eng. Appl. Artif. Intell.* **2020**, *96*, 103947. [[CrossRef](#)]
31. Eshkevari, S.S.; Takáč, M.; Pakzad, S.N.; Jahani, M. DynNet: Physics-based neural architecture design for nonlinear structural response modeling and prediction. *Eng. Struct.* **2021**, *229*, 111582. [[CrossRef](#)]
32. Vapnik, V. *The Nature of Statistical Learning Theory*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 1999.
33. Suykens, J.A.K.; Vandewalle, J. Least squares support vector machine classifiers. *Neural Process. Lett.* **1999**, *9*, 293–300. [[CrossRef](#)]
34. Ohtori, Y.; Christenson, R.E.; Spencer, B.F., Jr.; Dyke, S.J. Benchmark control problems for seismically excited nonlinear buildings. *J. Eng. Mech.* **2004**, *130*, 366–385. [[CrossRef](#)]
35. Asgarkhani, N.; Kazemi, F.; Jankowski, R. Machine learning-based prediction of residual drift and seismic risk assessment of steel moment-resisting frames considering soil-structure interaction. *Comput. Struct.* **2023**, *289*, 107181. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.