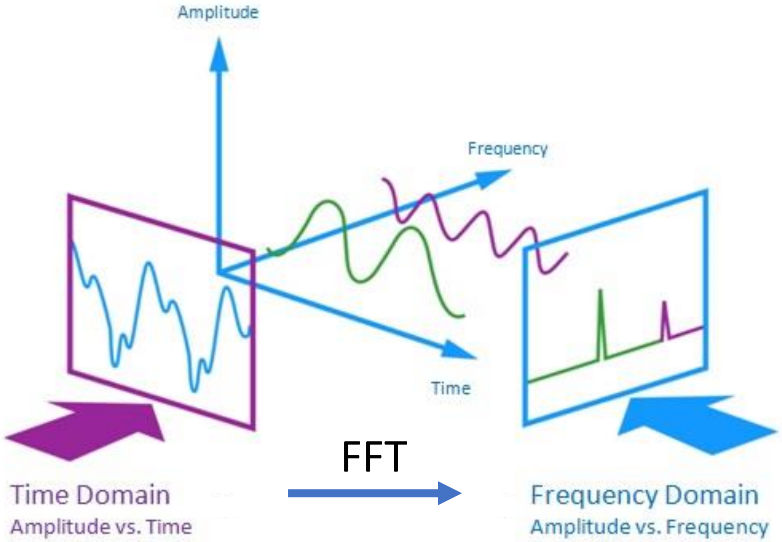
Time Series Forecasting for Structures Subjected to Nonstationary Inputs

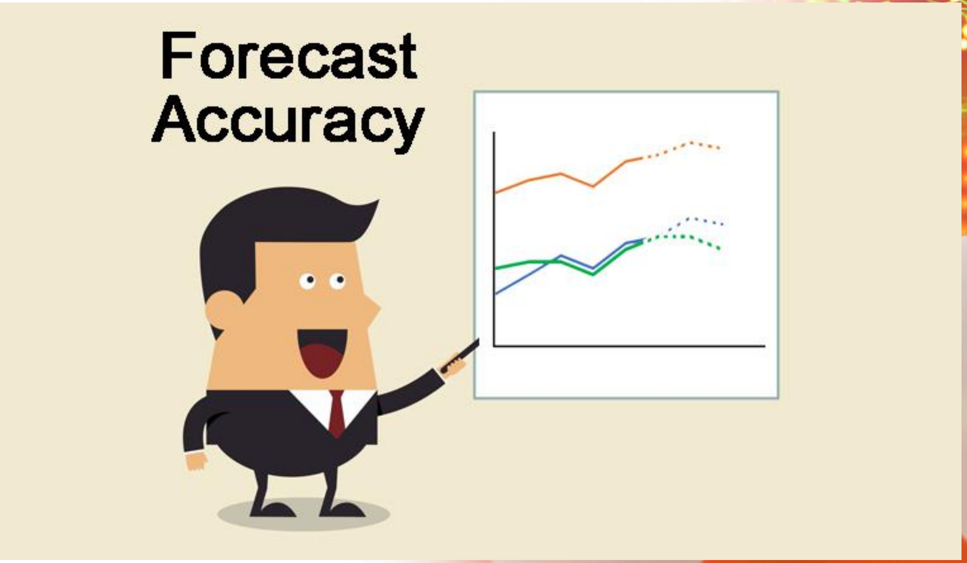
Puja Chowdhury[1], Philip Conrad[2], Jason D. Bakos[2], and Austin Downey[1,3]

[1]Department of Mechanical Engineering
[2]Department of Computer Science and Engineering
[3]Department of Civil and Environmental Engineering
University of South Carolina, Columbia, SC

TABLE OF CONTENTS:

- ❑ Introduction
- ❑ Methodology
- ❑ Experimental Setup
- ❑ Results and Discussion
- ❑ Conclusion





Introduction

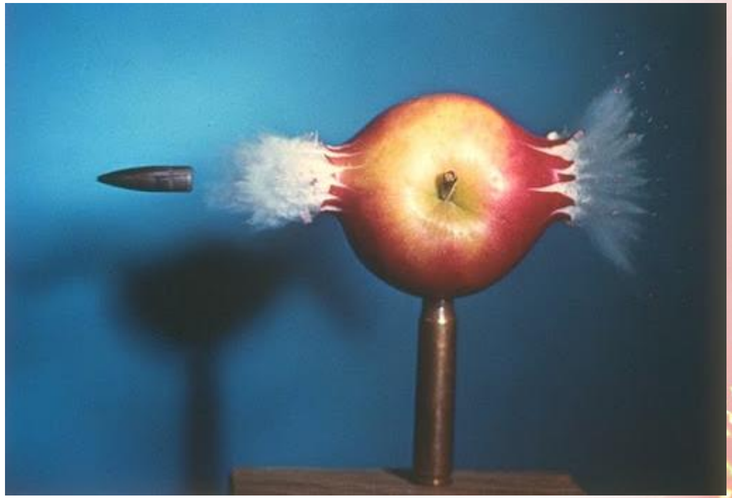
HIGH-RATE DYNAMICS

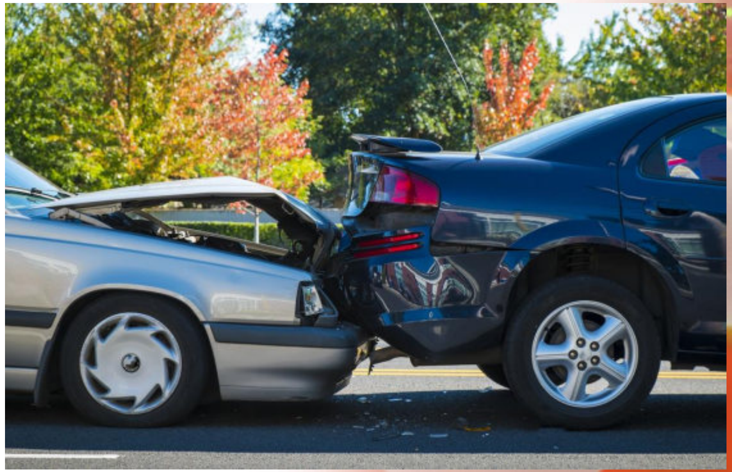
❑ Description of High-rate dynamics:

- high-rate (< 100 ms)
- high-amplitude (acceleration > 100 *g*)
- such as a blast or an impact

❑ The high-rate dynamics are subjected to

- large uncertainties in external loads;
- high levels of nonstationarities and heavy disturbances, and
- the generation of unmodeled dynamics from changes in system configuration





https://lh3.googleusercontent.com/proxy/w-IFlU4KzYIqsSI-zq860V8NNiweX6sfAilLLNxztT_6yCu7V3eb8RgnrQFboqnRllaCkpX0ULMcYiKRHHAi7jBClOphkYSjpBnPxEZqz0Y1oBP6PB58
https://injuredcalltoday.com/wp-content/uploads/2020/02/Car-Accidents-Statistics-Most-Dangereous-Places-to-Drive-in-America.jpg

Structures Experiencing High-Rate Dynamics

Hypersonic vehicles

Space launch system

Blast seat energy absorbers

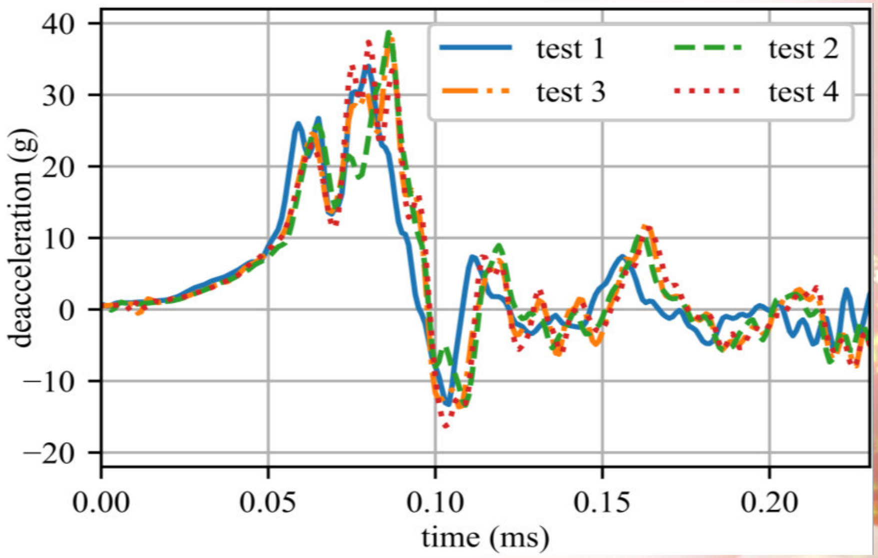
Ballistics packages

Vehicle collision

Blast protection damper

HIGH-RATE DYNAMICS (continued)

- Goals
  - Application: Real-time decision making for structures
  - Required Technologies:
    - low-latency model updating
    - system state prognostics in real time
- Challenges:
  - Computing power is limited
    - memory, available energy, processors
  - Unknown sources of the inputs (forces, location)
  - Inability to calculate fault scenarios in advance.
  - Rare and extreme situations

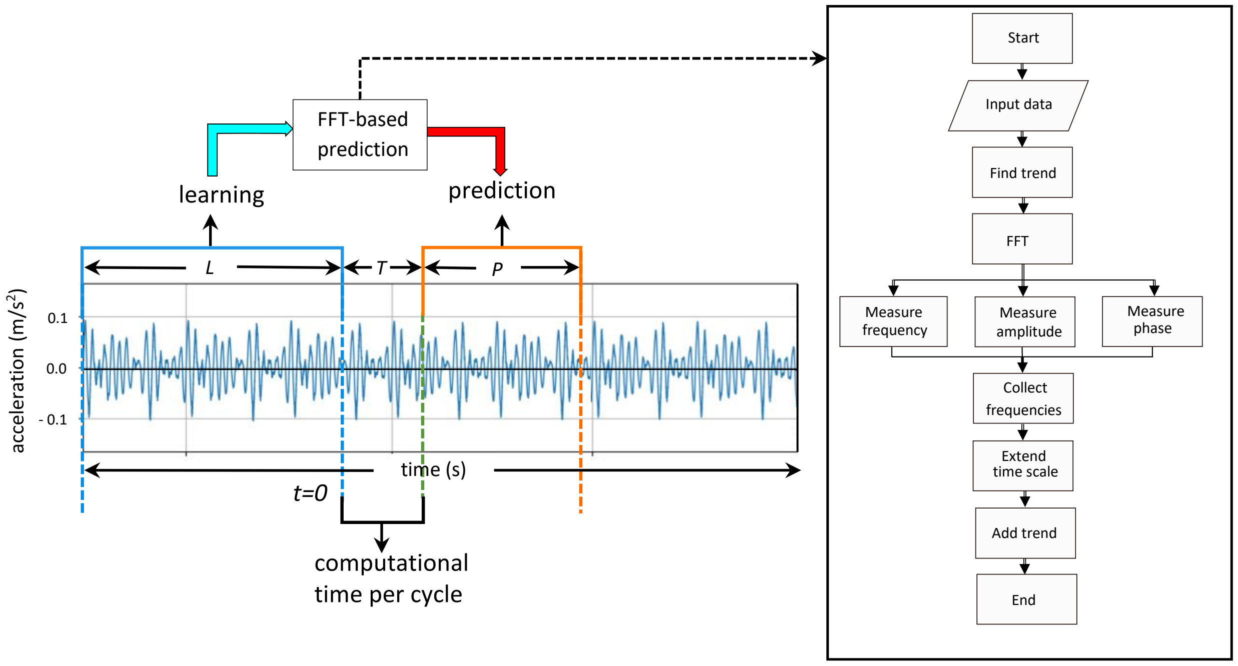


Methodology

THE MAIN AIM OF THIS WORK

- A numerical analysis for the real-time implementation
  - Fast Fourier Transform (FFT)-based approach
  - time series forecasting.
- The main contribution: FFT-based approach responds on a nonstationary event
  - before, during, and directly following the event
  - considering different learning window lengths and assumed computation times.
- Implementation of this preliminary time series forecasting work
  - Offline: using pre-recorded experimental data
  - The FFT-based approach is implemented in a rolling window configuration.

ALGORITHM



FFT-based prediction

Schematic Algorithm diagram of FFT-based time series prediction algorithms.

Collected frequencies

frequencies (Hz)						
20	-20	60	-60	70	-70	80
-80	100	-100	120	-120	140	-140
150	-150	160	-160	170	-170	180
-180	200	-200	220	-220	240	-240

Parameter values

learning length	computational time	prediction length
L (s)	T (s)	P (s)
0.1, 0.5, 1	0.01	1
0.1, 0.5, 1	0.1	1
0.1, 0.5, 1	0.5	1
0.1, 0.5, 1	1	1

PROBLEM STATEMENT

- The measured acceleration signal is

$$x_{\mathrm{v}} = (x_1,\ x_2,\ x_3,\ \dots\ , x_V)$$

- The rolling window is

$$x_{\mathrm{a}} = (x_{\mathrm{a}1},\ x_{\mathrm{a}2},\ x_{\mathrm{a}3},\ \dots\ , x_{\mathrm{a}N})$$

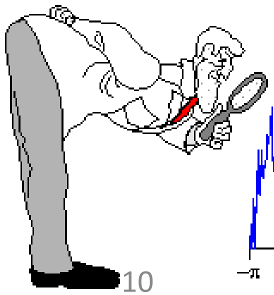
- A polynomial function is used for finding trend.

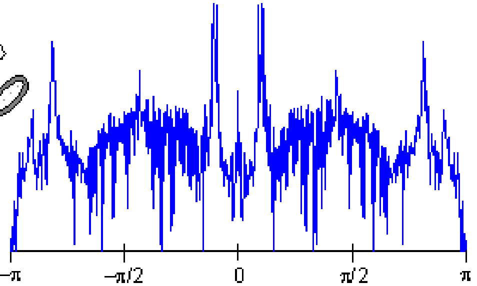
$$x_{\mathrm{trend}} = p(x) = c_0 + c_1 x + c_2 x^2 + \dots + c_q x^q$$

- After removing the trend, the new acceleration signal without trend is

$$x = x_{\mathrm{a}} - x_{\mathrm{trend}}$$

- As considered, the acceleration signal without the trend, $x = (x_1,\ x_2,\ x_3,\ \dots\ ,\ x_N)$





- Therefore, the discrete Fourier transform (DFT) of that series can be expressed as

$$X_k = \sum_{n=0}^{N-1} x_n e^{(-i2\pi(kn/N))} \text{ for } k = 0,\ \dots\ ,\ N$$

$$\omega = 2\pi/N = 2\pi f$$

$$(X_{\mathrm{amp}})_k = |X_k|$$

$$(X_{\mathrm{phase}})_k = X_k/|X_k|$$

- Similarly, the inverse DFT can be written as

$$x_n = 1/N \sum_{k=0}^{N-1} X_k e^{(i2\pi kn/N)} \text{ for } n = 0,\ \dots\ ,\ N$$

- A new series of *M* length where *M* > *N*. Using amplitude and phase information, the time series can be constructed and written as

$$x_m = \sum_{k=0}^{M-1} ((X_{\mathrm{amp}})_k \cos(2\pi(km/M)) + (X_{\mathrm{phase}})_k) \text{ for } m = 1,\ \dots\ ,\ M$$

- The time series with the trend information added back can be expressed as

$$x_{\mathrm{a\_new}} = x_m + x_{\mathrm{trend}}$$

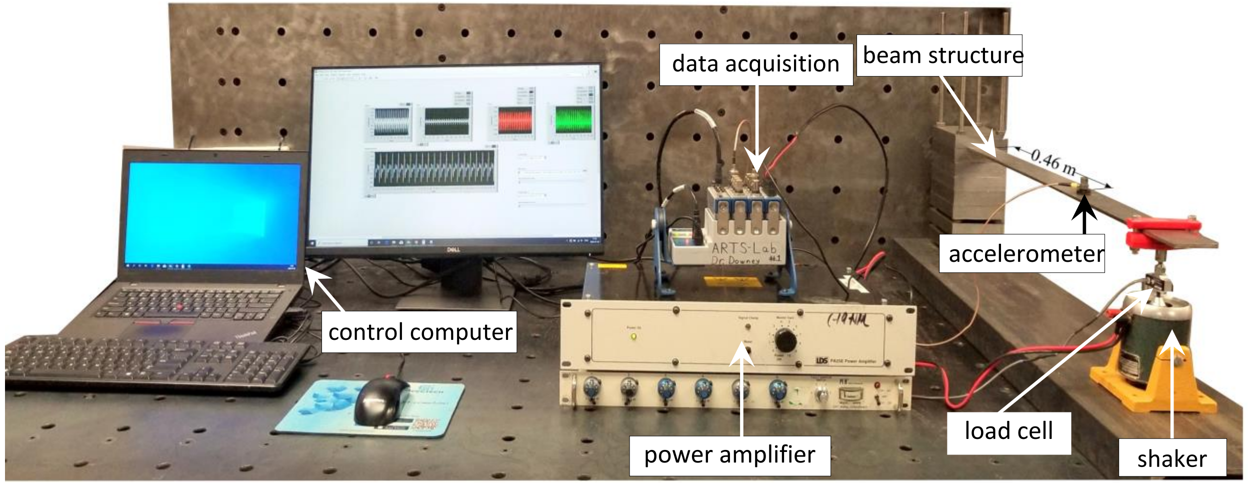
- **By applying the FFT-based time series forecasting method, a signal is generated that is *M* points long where *M* > *N*. The difference, (*M* - *N*) presents the length of the prediction horizon.**

- The predicted series would then be

$$x_{\mathrm{pred}} = (x_{\mathrm{a\_new}(N+1)},\ x_{\mathrm{a\_new}(N+2)},\ x_{\mathrm{a\_new}(N+3)},\ \dots\ ,\ x_{\mathrm{a\_new}(M)})$$

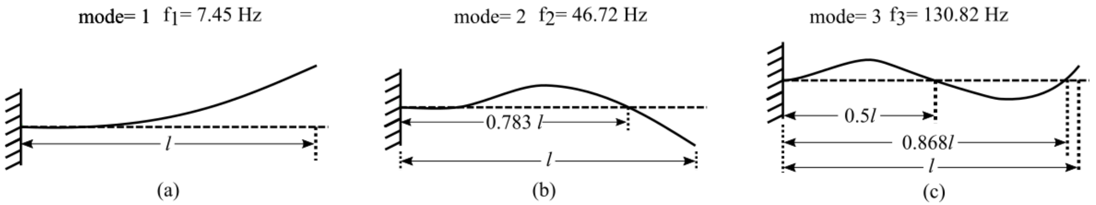
Experimental setup

EXPERIMENTAL SETUP



Experimental setup of a cantilever beam with key components and data acquisition setup.

- This data is available in a public repository [1]. This paper used text_3.
- The mode shapes and natural frequencies for the first three modes of the cantilever were computed via **Euler's formula**

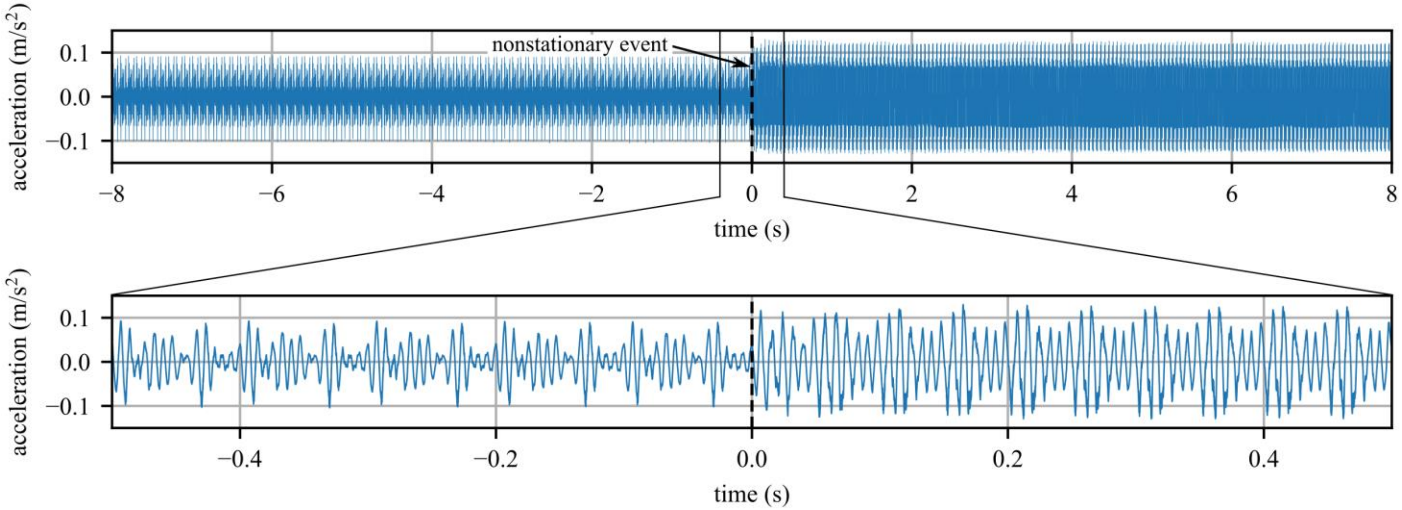


Mode shapes and frequencies for the cantilever beam setup showing: (a) mode shape 1; (b) mode shape 2, and; (c) mode shape 3.

[1] High-Rate-SHM-Working-Group. Dataset-4 univariate signal with nonstationarity. https://github.com/High-RateSHM-Working-Group/Dataset-4-Univariate-signal-withnon-stationarity

DATA STRUCTURE

- The structure's measured acceleration response for a composite sinusoidal input from the shaker.
- Two sine wave signals are concatenated together at *t*=0 where a 50% nonstationary is present. A 50% nonstationary event is introduced at 0 s, as measured by a 50% increase in the standard deviation of the signal.
- The first half of the composite signal is built from 100, 120, and 150 Hz frequencies while the second half signal consists of 100 and 120 Hz frequencies.



The full 16-second test is shown in the upper plot while the inset shows the 1 second around the nonstationary

Results and discussion

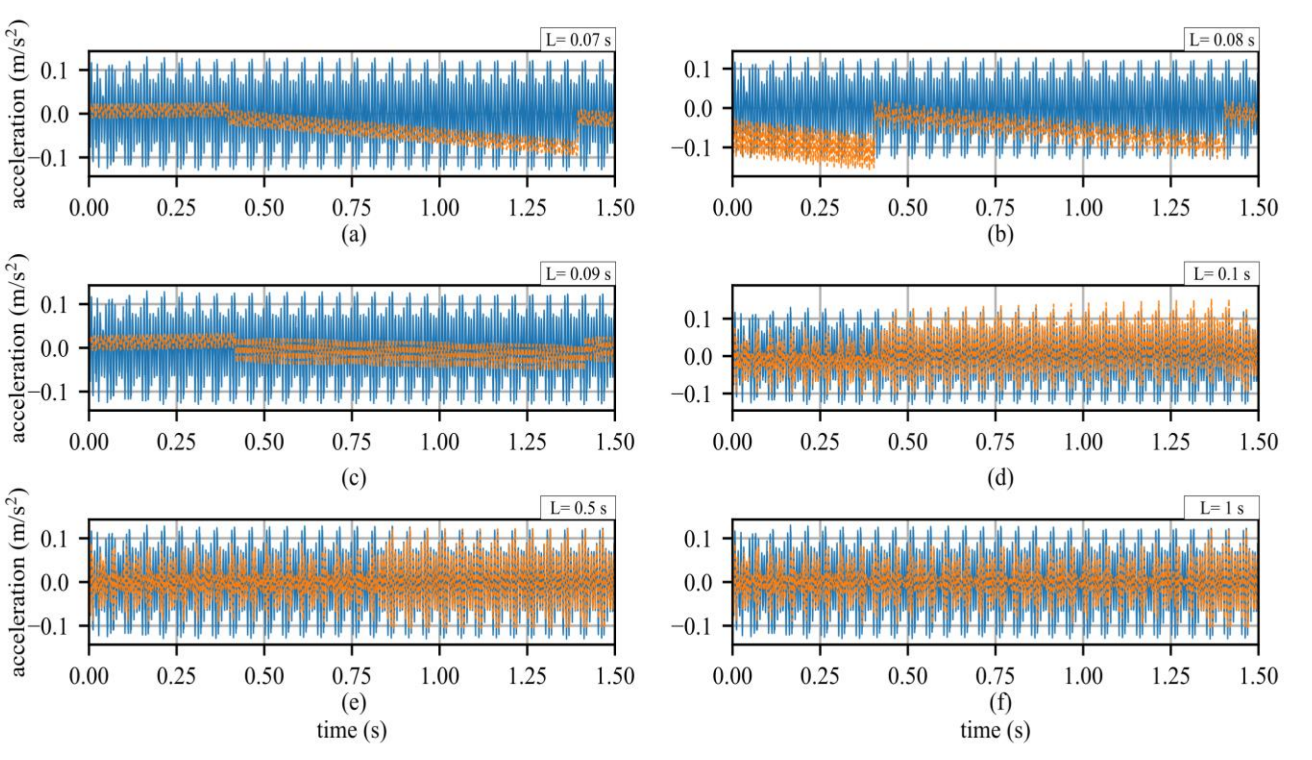
RESULTS

The effect of two parameters on the algorithm have been analyzed:

1. The length of the learning window
2. The computational time

The time series prediction for the FFT-based algorithm for the various learning window lengths considered.

- ❑ Applying the Nyquist Theorem, the minimum length of the learning window should be 0.1 s.

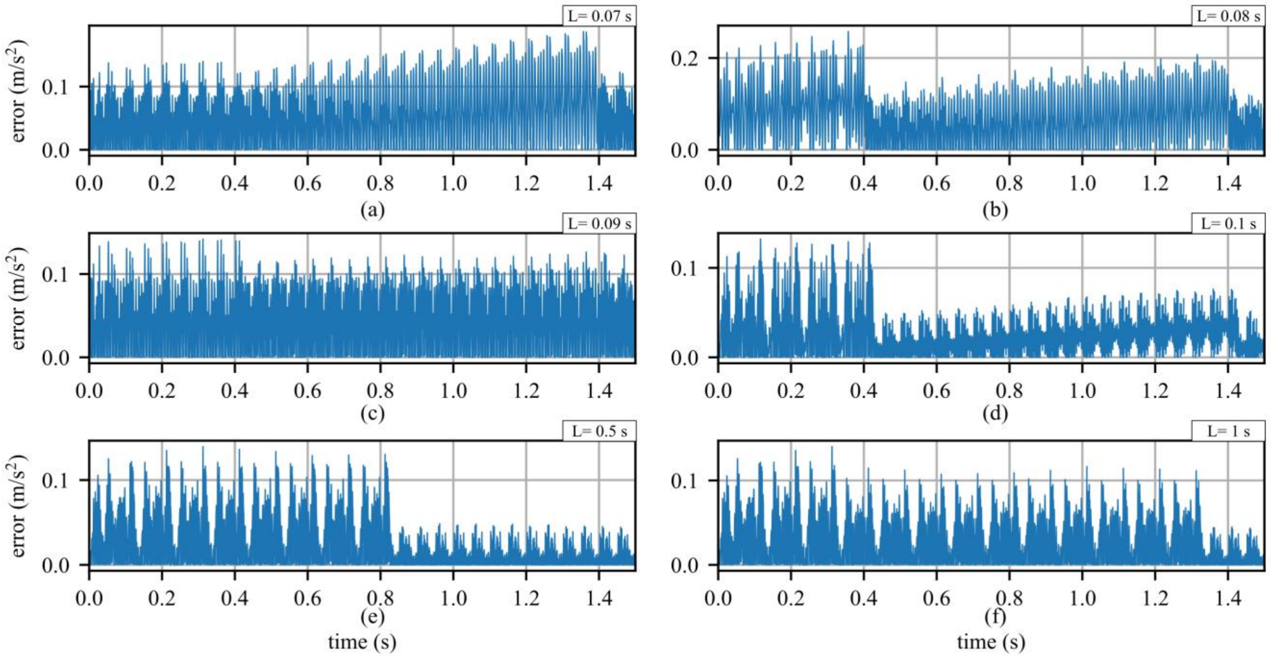


Time series prediction using various learning window lengths showing: (a) 0.07 s window length; (b) 0.08 s window length; (c) 0.09 s window length; (d) 0.1 s window length; (e) 0.5 s window length; and (f) 1 s window length.

RESULTS (continued)

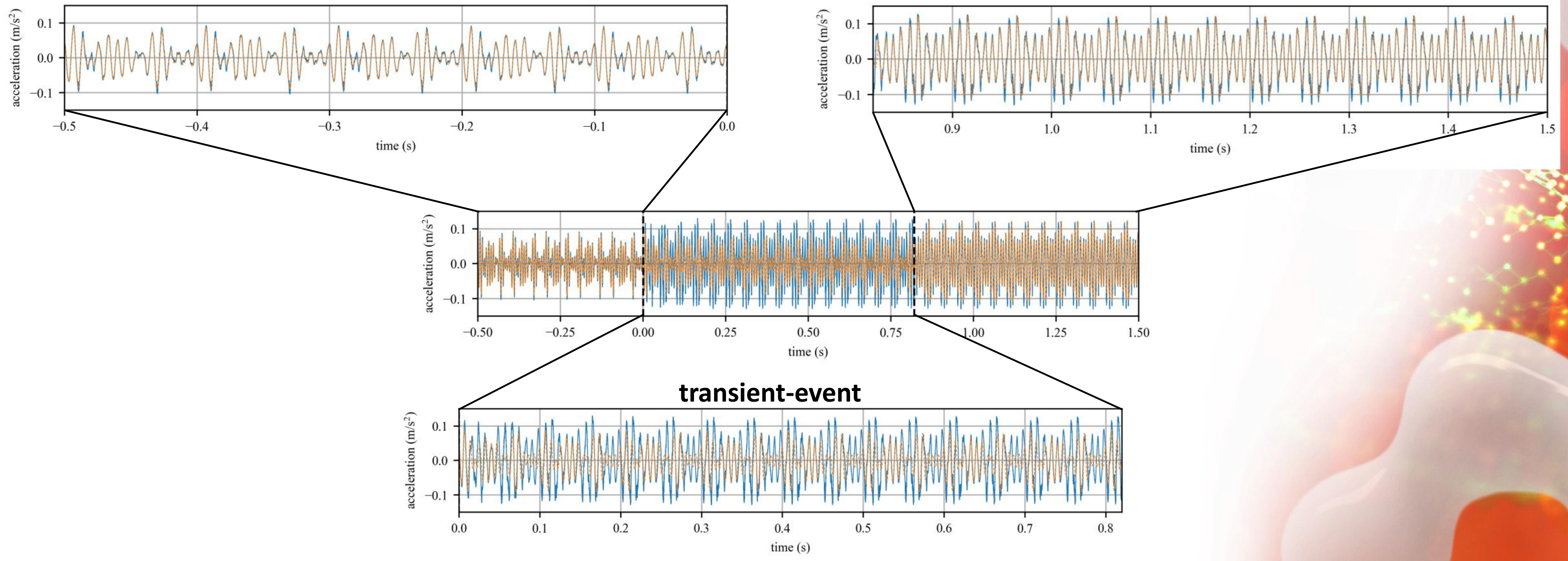
The instantaneous (i.e., point-by-point) error for the FFT based algorithm for various learning window lengths considered is shown in this figure.

- As the length of the learning window increases beyond 0.1 s, the quality of the reproduced signal improves.
- Finally, the three learning window lengths of 0.1 s (d), 0.5 s (e), and 1 s (f) are being considered for further analysis.



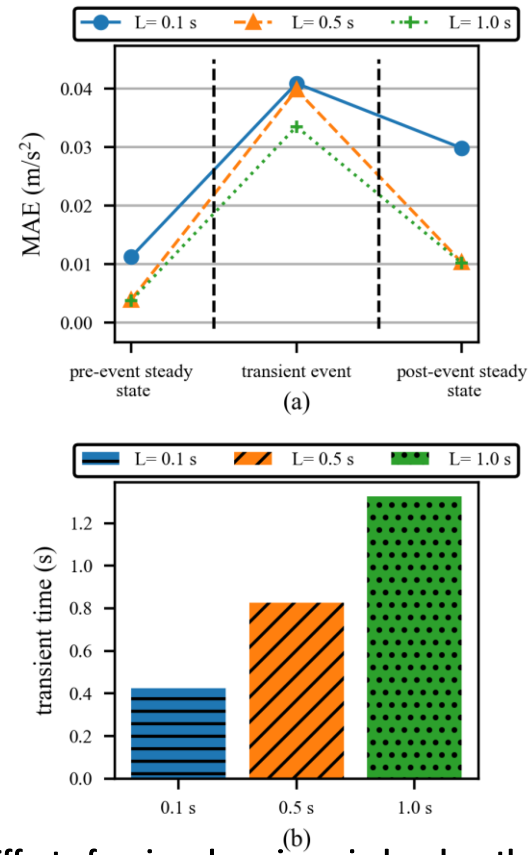
Calculated instantaneous error over for the experiment data with various learning window lengths showing: (a) 0.07 s window length; (b) 0.08 s window length; (c) 0.09 s window length; (d) 0.1 s window length; (e) 0.5 s window length; and (f) 1 s window length.

RESULTS (continued)



Time series prediction for 0.5 s learning window length in different states

THE LENGTH OF THE LEARNING WINDOW



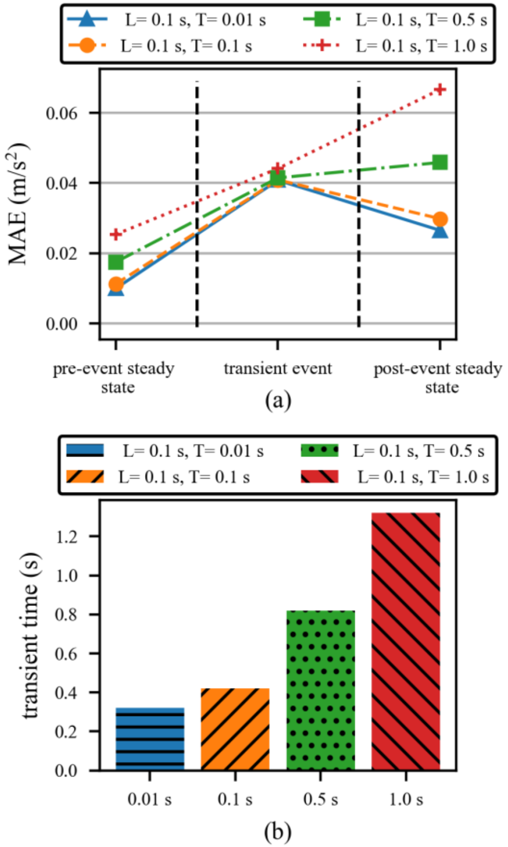
Effect of various learning window lengths (*L*) showing: (a) MAE in different states, and; (b) transient time.

Performance metrics for various learning window lengths

		learning window length		
		0.1 s	**0.5 s**	**1 s**
State MAE (m/s^2)	Pre-event steady state	0.0112	0.0039	0.0038
	Transient event	0.0409	0.0398	0.0335
	Post-event steady state	0.0298	0.0103	0.0102
Transient time (s)		0.42	0.82	1.32

- **The mean error and learning window length relationship are inversely proportional while the transient time and learning window length relationship are proportional.**

THE COMPUTATIONAL TIME



Effect of various computational time (*T*) in a specific learning window length (*L*) showing: (a) MAE in different states, and;(b) transient time.

Performance metrics for various computational times

		computational time			
		0.01 s	**0.1 s**	**0.5 s**	**1 s**
State MAE (m/s^2)	Pre-event steady state	0.0099	0.0112	0.0175	0.0254
	Transient event	0.0408	0.0409	0.0414	0.0441
	Post-event steady state	0.0265	0.0298	0.0459	0.0666
Transient time (s)		0.32	0.42	0.82	1.32

- **The computational time increases, the MAE and transient time increases. This is a proportional relationship.**

Conclusion

CONCLUSION

- This work presents a mathematical examination and exploratory outcomes for the continuous execution of a Fast Fourier Transform (FFT)-based methodology for time series forecasting.
- Learning window lengths are inversely proportional with mean error in different states and proportional with transient time.
- The relationship between computational time and mean error in different states, as well as transient time, is proportional.

FUTURE WORK

- In future work, the FFT-based rolling window prediction method will be implemented in hardware for real-time online time series forecasting.

ACKNOWLEDGMENT

This material is based upon work supported by the National Science Foundation under Grants 1850012 and 1937535.The support of the National Science Foundation is gratefully acknowledged. Any opinions, findings, and conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation. The authors would like to thank collaborators at Iowa State University for their contributions to this work.

NSF

IOWA STATE UNIVERSITY
OF SCIENCE AND TECHNOLOGY

UNIVERSITY OF
SOUTH CAROLINA

Thanks!