Unsteady Aerodynamic Force Sensing from Measured Strain

Prepared For:
30th Congress of the International Council of the Aeronautical Science
Daejeon, South Korea, September 25-30, 2016

Prepared By:

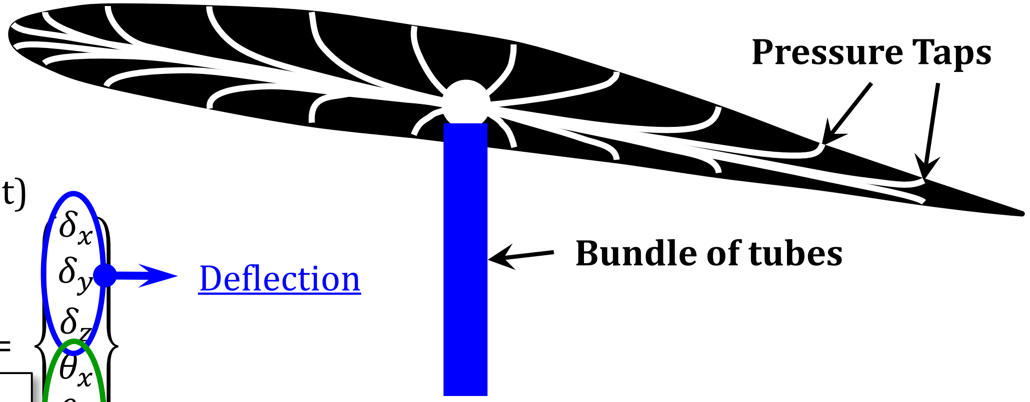
Chan-gi Pak, Ph.D.
Structural Dynamics Group, Aerostructures Branch (Code RS)
NASA Armstrong Flight Research Center

Overview

What the technology does

Problem Statement

- To improve **fuel efficiency** for an aircraft
  - **Reducing weight or drag**
    - **Similar effect** on fuel savings
  - Multidisciplinary design optimization (design phase) or active control (during flight)
- Real-time measurement of **structural responses and loads** during flight are **critical data**.
  - Active flexible motion control
  - Active induced drag control



Complete degrees of freedom

$$q(t) = \begin{Bmatrix} \delta_x \\ \delta_y \\ \delta_z \\ \theta_x \\ \theta_y \\ \theta_z \end{Bmatrix}$$

$\delta_x, \delta_y, \delta_z$ → Deflection

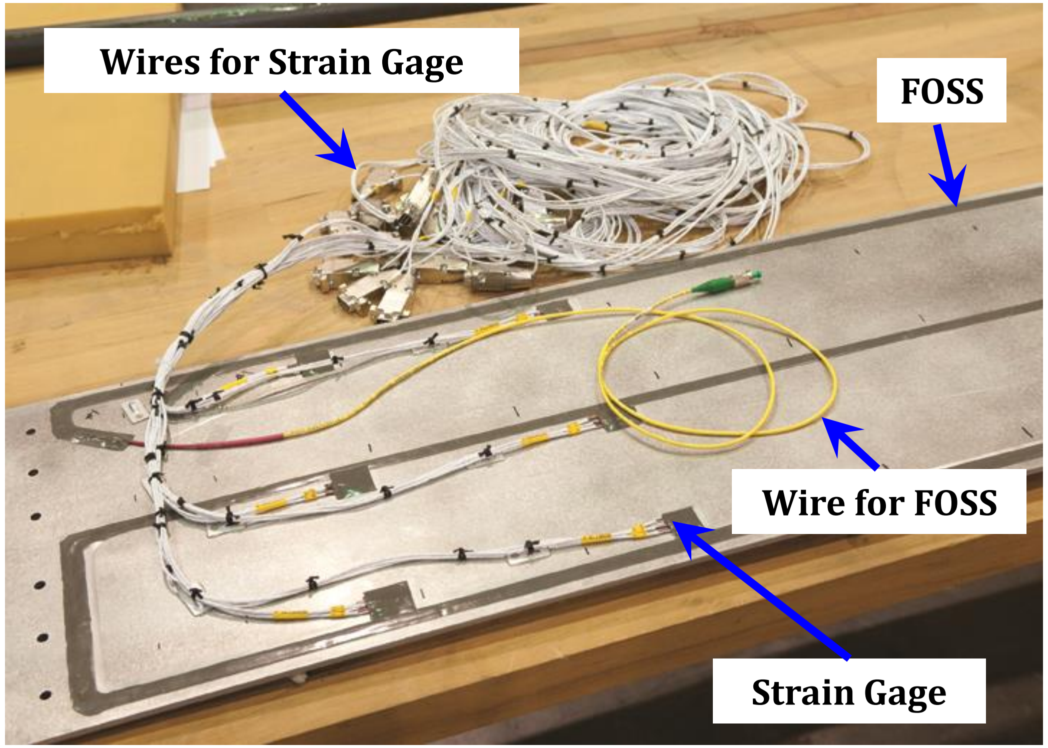
$\theta_x, \theta_y, \theta_z$ → Slope (angle)

Objective

- Compute unsteady aerodynamic loads from unsteady strain measurements
- Structural responses (complete degrees of freedom) are essential quantities for load computations during flight.
  - Loads can be computed from the following governing equations of motion.

$$[\mathbf{M}]\{\ddot{q}(t)\} + [\mathbf{G}]\{\dot{q}(t)\} + [\mathbf{K}]\{q(t)\} = \{Q_a(Mach, \{q(t)\}, \{\dot{q}(t)\}, \{\ddot{q}(t)\})\}$$

    - Internal Loads: using finite element structure model
      - ✓ $[\mathbf{M}]\{\ddot{q}(t)\}$, $[\mathbf{G}]\{\dot{q}(t)\}$, $[\mathbf{K}]\{q(t)\}$: Inertia, damping, and elastic loads
    - External Load: using unsteady aerodynamic model
      - ✓ $\{Q_a(Mach, \{q(t)\}, \{\dot{q}(t)\}, \{\ddot{q}(t)\})\}$: Aerodynamic load



Issue

- Traditionally, lift load over the wing are measured using a pressure gauge.
  - This conventional pressure gauge with associated piping and cabling would create **weight and space limitation** issues and pressure data will be available only at discrete gauge location. Therefore, a **new innovation** is needed.
    - **Fiber optic strain sensor (FOSS)** is an ideal choice for aerospace applications.

Previous technologies

- Liu, T., Barrows, D. A., Burner, A. W., and Rhew, R. D., "Determining Aerodynamic Loads Based on Optical Deformation Measurements," *AIAA Journal,* Vol.40, No.6, June **2002**, pp.1105-1112

  - **NASA LRC;** Application is limited for **"beam"; static deflection & aerodynamic loads**

- Igawa, H. et al., "Measurement of Distributed Strain and **Load** Identification Using 1500 mm Gauge Length FBG and Optical Frequency Domain Reflectometry," 20th International Conference on Optical Fibre Sensors, **2009**

  - **JAXA; using inverse analysis. "Beam" application only; static deflection & loads**

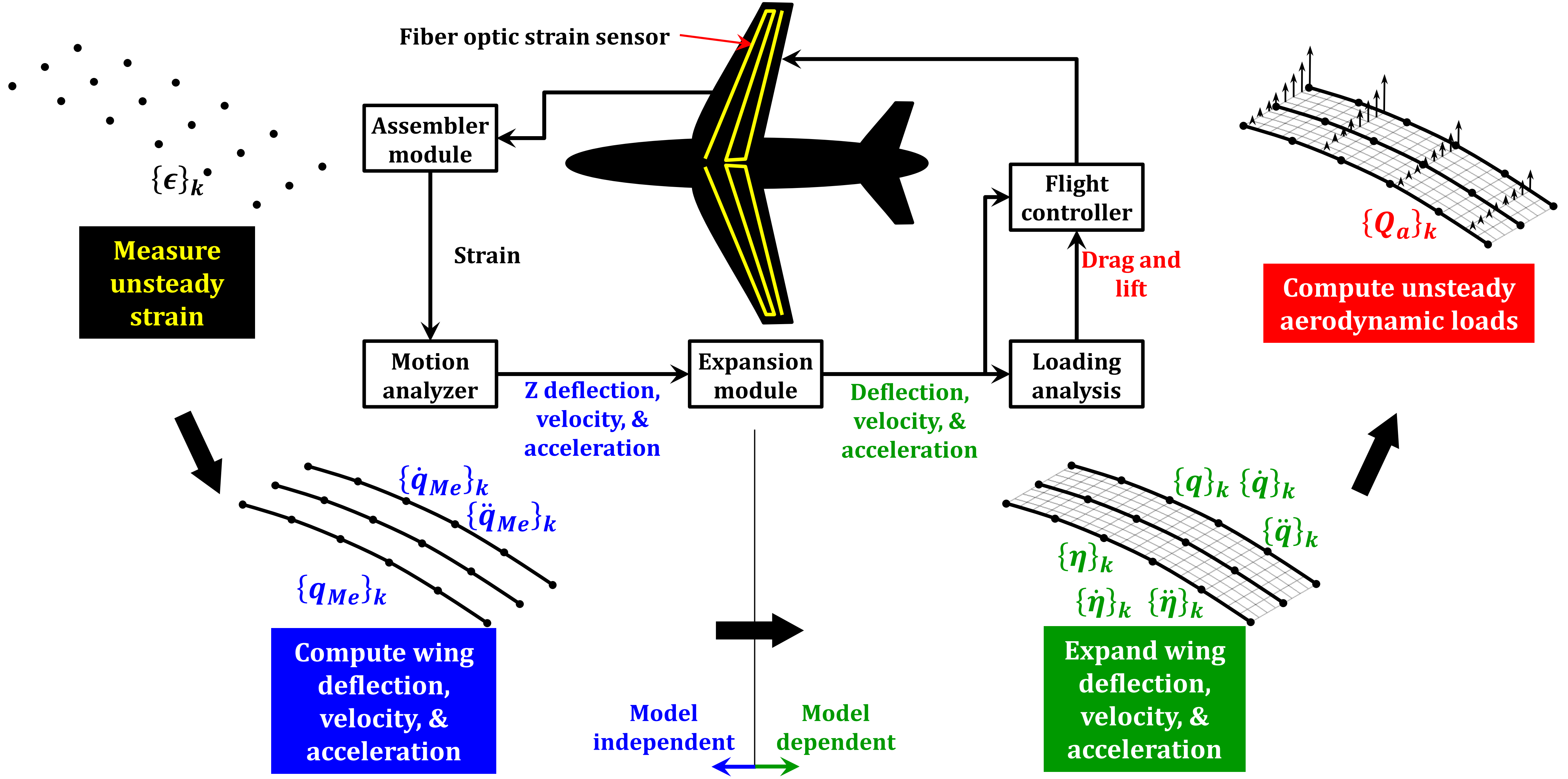
- Richards, L. and Ko, W. , "Process for using surface strain measurements to obtain operational **loads** for complex structures," US Patent #7715994, May 11, **2010**

  - **NASA AFRC; "sectional" bending moment, torsional moment, and shear force** along the **"beam".**

- Carpenter, T.J. and Albertani, R., "**Aerodynamic Load** Estimation from Virtual Strain Sensors for a Pliant **Membrane Wing**," *AIAA Journal,* Vol.53, No.8, August **2015**, pp.2069-2079

  - **Oregon State University; Aerodynamic loads** are estimated from measured strain using virtual strain sensor technique.

Steps used to compute aerodynamic load from measured strain



Z deflection, velocity, & acceleration along each fiber are model independent quantities

Technical features of two-step approach : Deflection Computation

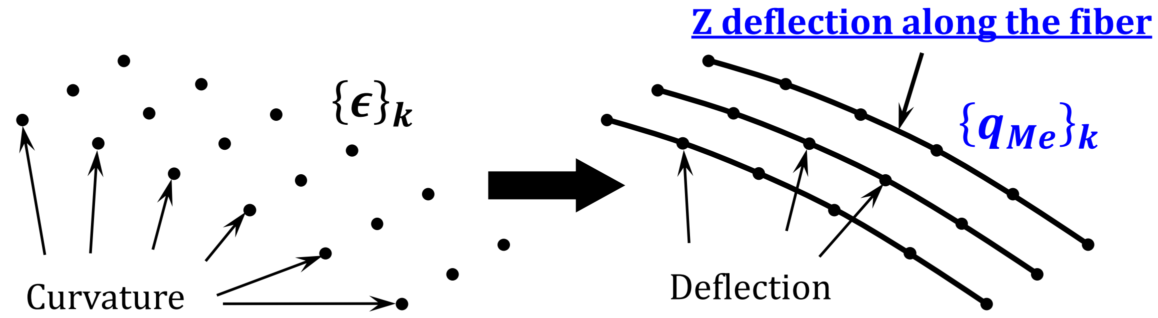
- First Step of two-step approach
  - Use **piecewise least-squares method** to **minimize noise** in the measured strain data (strain/offset): **re-generate** strain data
  - Obtain **cubic spline** (Akima spline) function using re-generated strain data points (assume small motion):

$$\frac{d^2\delta_k}{ds^2} = -\epsilon_k(s)/c(s)$$

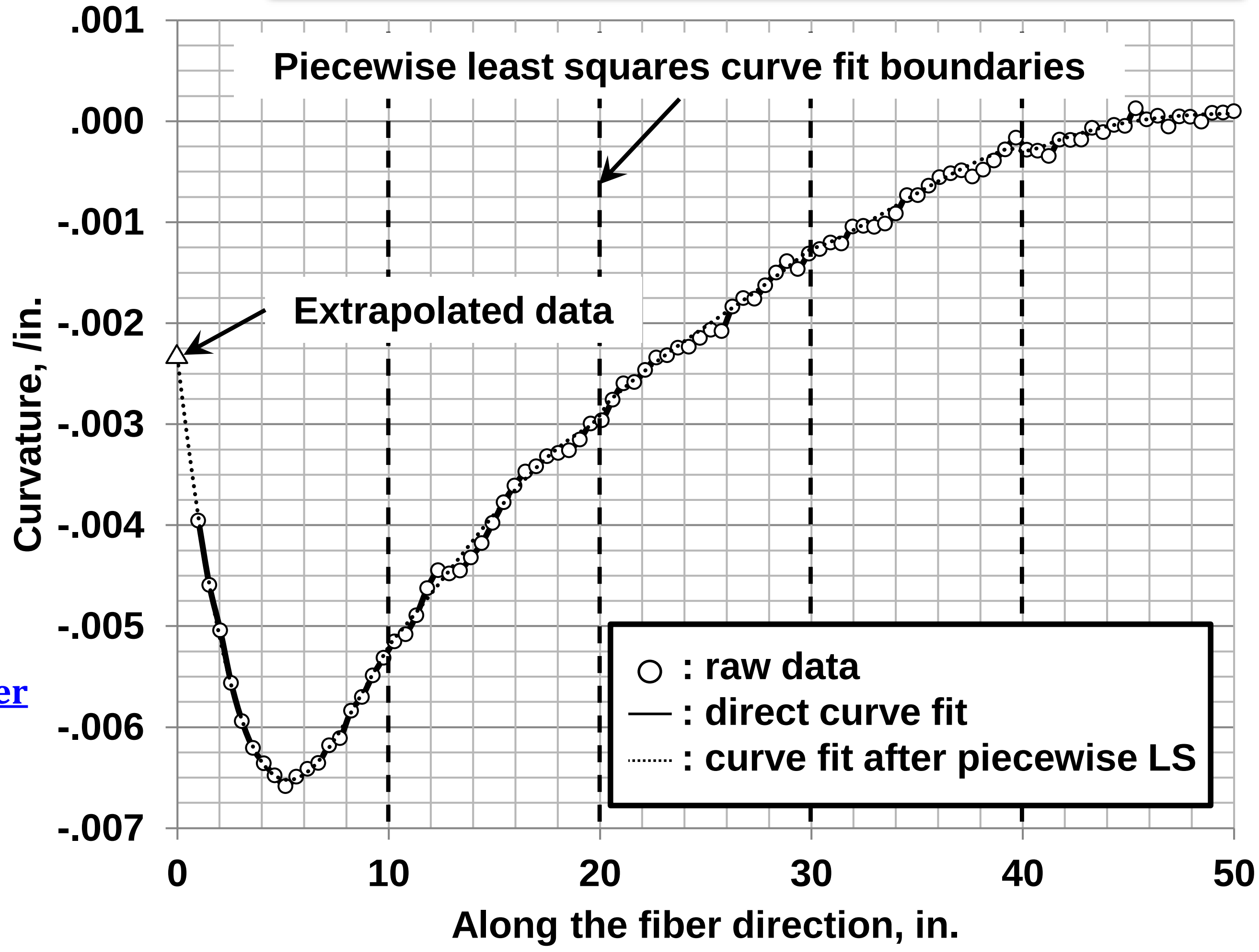
  - **Integrate fitted spline function** to get slope data:

$$\frac{d\delta_k}{ds} = \theta_k\ (s)$$

  - Obtain cubic spline (Akima spline) function using computed slope data
  - **Integrate fitted spline function** to get deflection data: $\delta_k(s)$

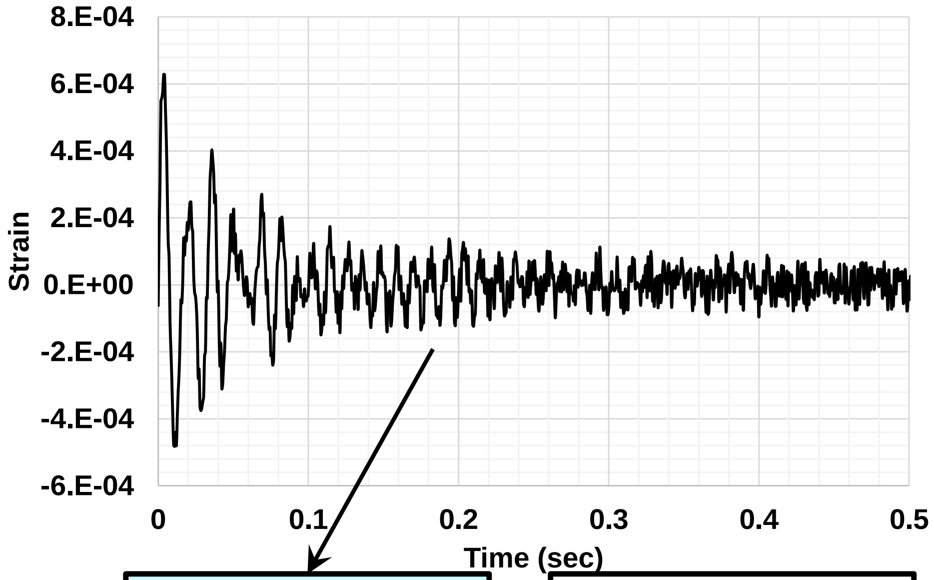


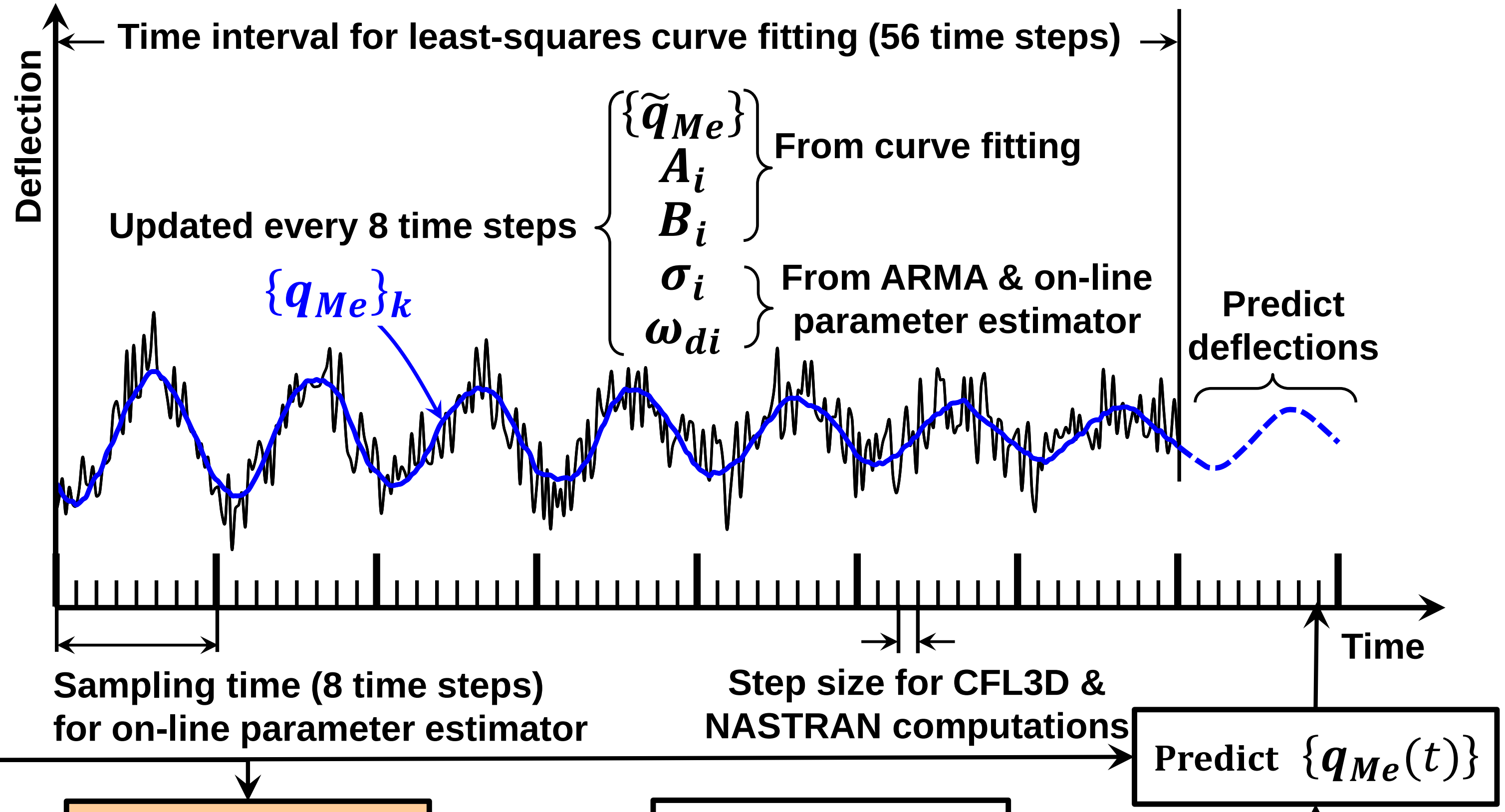
#1 Least squares fitting with respect to spatial coordinates using piecewise polynomial functions



A measured strain is fitted using a piecewise least-squares method together with the cubic spline technique.

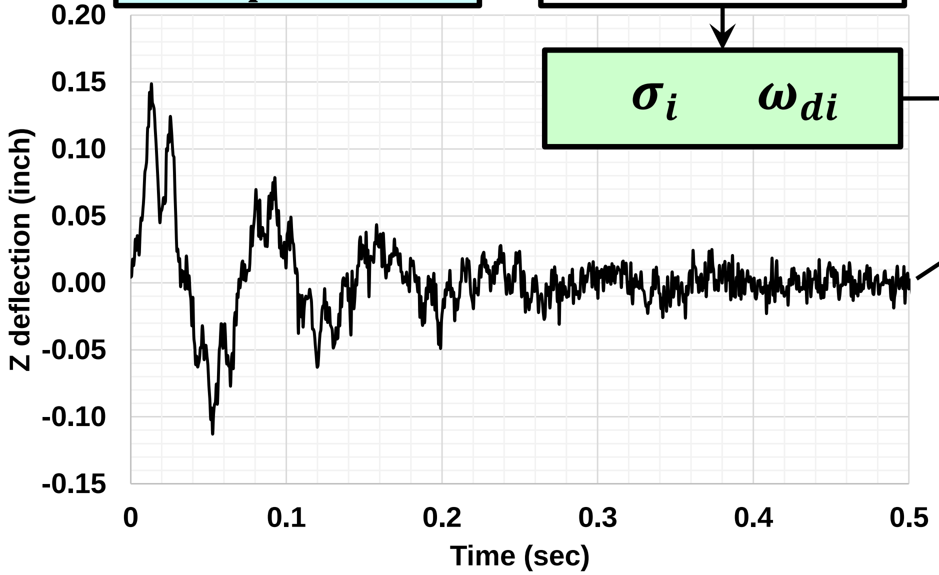
Technical features of new technology: Velocity & Acceleration Computation





Sine Butterworth low pass filter → On-line parameter estimator → σ_i ω_{di}

Least-squares curve fitting → $\{\tilde{q}_{Me}\}$ A_i B_i



$$\{q_{Me}(t)\} = \{\tilde{q}_{Me}\} + \sum_{i=1}^{nm} e^{-\sigma_i t}\{A_i cos(\omega_{di}t) + B_i sin(\omega_{di}t)\}$$

$$\{\dot{q}_{Me}(t)\} = \frac{d}{dt}\{q_{Me}(t)\}$$

$$\{\ddot{q}_{Me}(t)\} = \frac{d^2}{dt^2}\{q_{Me}(t)\}$$

#2 Least squares fitting with respect to time using sine, cosine, & exponential functions

$\{\dot{q}_{Me}\}_k$ $\{\ddot{q}_{Me}\}_k$

Use low pass filter, ARMA model, on-line parameter estimator, and least-squares curve fitting method to obtain velocity and acceleration.

Technical features of expanding procedure

- Second step of two step approach: Based on General Transformation
  - Definition of the generalized coordinates vector $\{\boldsymbol{q}\}_k$ and the othonormalized coordinates vector $\{\boldsymbol{\eta}\}_k$ at discrete time k

$$\{\boldsymbol{q}\}_k = \begin{Bmatrix} \boldsymbol{q_M} \\ \boldsymbol{q_S} \end{Bmatrix}_k = [\boldsymbol{\Phi}]\{\boldsymbol{\eta}\}_k = \begin{bmatrix} \boldsymbol{\Phi_M} \\ \boldsymbol{\Phi_S} \end{bmatrix} \{\boldsymbol{\eta}\}_k$$

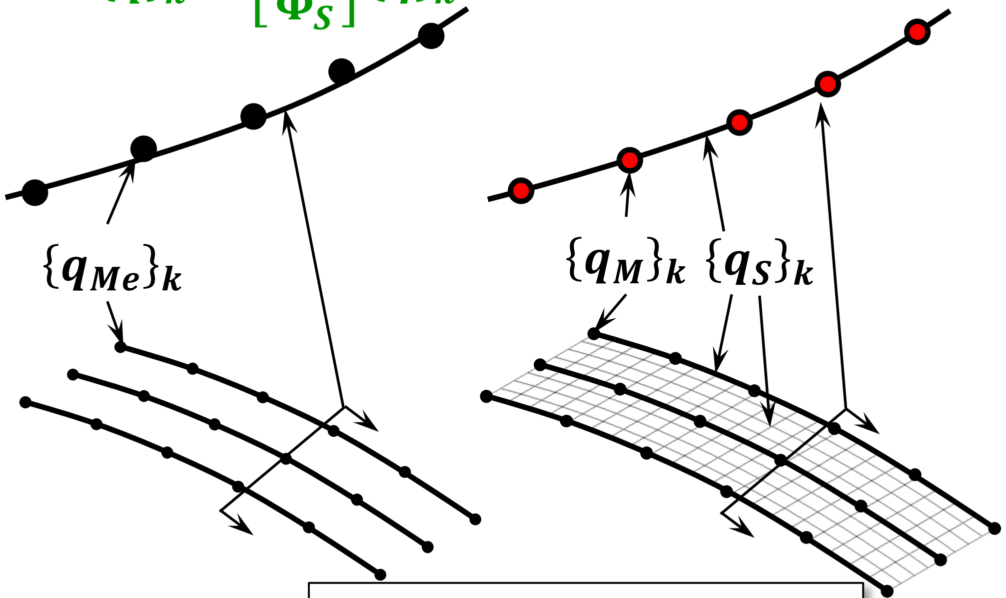
  - For all model reduction/expansion techniques, there is a relationship between the **master (measured or tested) degrees of freedom** and the **slave (deleted or omitted) degrees of freedom** which can be written in general terms as $\{\boldsymbol{q_M}\}_k = [\boldsymbol{\Phi_M}]\{\boldsymbol{\eta}\}_k \quad \{\boldsymbol{q_S}\}_k = [\boldsymbol{\Phi_S}]\{\boldsymbol{\eta}\}_k$
  - Changing master DOF at discrete time k $\{\boldsymbol{q_M}\}_k$ to the corresponding measured values $\{\boldsymbol{q_{Me}}\}_k$

$$\{\boldsymbol{q_{Me}}\}_k = [\boldsymbol{\Phi_M}]\{\boldsymbol{\eta}\}_k$$
$$[\boldsymbol{\Phi_M}]^T\{\boldsymbol{q_{Me}}\}_k = [\boldsymbol{\Phi_M}]^T[\boldsymbol{\Phi_M}]\{\boldsymbol{\eta}\}_k$$
$$\{\boldsymbol{\eta}\}_k = \left([\boldsymbol{\Phi_M}]^T[\boldsymbol{\Phi_M}]\right)^{-1}[\boldsymbol{\Phi_M}]^T\{\boldsymbol{q_{Me}}\}_k$$

- Expansion of displacement using SEREP: kinds of least-squares surface fitting; most accurate reduction-expansion technique
  - $\{\boldsymbol{q_{Me}}\}_k$: master DOF at discrete time k; **deflection** along the fiber "**computed from the first step**"
  - $\{\boldsymbol{q_M}\}_k = [\boldsymbol{\Phi_M}]\left([\boldsymbol{\Phi_M}]^T[\boldsymbol{\Phi_M}]\right)^{-1}[\boldsymbol{\Phi_M}]^T\{\boldsymbol{q_{Me}}\}_k$: smoothed master DOF
  - $\{\boldsymbol{q_S}\}_k = [\boldsymbol{\Phi_S}]\left([\boldsymbol{\Phi_M}]^T[\boldsymbol{\Phi_M}]\right)^{-1}[\boldsymbol{\Phi_M}]^T\{\boldsymbol{q_{Me}}\}_k$: deflection and slope all over the structure

$$\{\boldsymbol{\eta}\}_k = \left([\boldsymbol{\Phi_M}]^T[\boldsymbol{\Phi_M}]\right)^{-1}[\boldsymbol{\Phi_M}]^T\{\boldsymbol{q_{Me}}\}_k$$
$$\{\dot{\boldsymbol{\eta}}\}_k = \left([\boldsymbol{\Phi_M}]^T[\boldsymbol{\Phi_M}]\right)^{-1}[\boldsymbol{\Phi_M}]^T\{\dot{\boldsymbol{q}}_{\boldsymbol{Me}}\}_k$$
$$\{\ddot{\boldsymbol{\eta}}\}_k = \left([\boldsymbol{\Phi_M}]^T[\boldsymbol{\Phi_M}]\right)^{-1}[\boldsymbol{\Phi_M}]^T\{\ddot{\boldsymbol{q}}_{\boldsymbol{Me}}\}_k$$
$$\{\boldsymbol{q}\}_k = \begin{bmatrix} \boldsymbol{\Phi_M} \\ \boldsymbol{\Phi_S} \end{bmatrix}\{\boldsymbol{\eta}\}_k$$
$$\{\dot{\boldsymbol{q}}\}_k = \begin{bmatrix} \boldsymbol{\Phi_M} \\ \boldsymbol{\Phi_S} \end{bmatrix}\{\dot{\boldsymbol{\eta}}\}_k$$
$$\{\ddot{\boldsymbol{q}}\}_k = \begin{bmatrix} \boldsymbol{\Phi_M} \\ \boldsymbol{\Phi_S} \end{bmatrix}\{\ddot{\boldsymbol{\eta}}\}_k$$

Z motion along the fiber



#3 Least-squares "surface" fitting using basis functions

System Equivalent Reduction and Expansion Process

Technical features of New Technology: Unsteady Aerodynamic Loads

$$\{\eta\}_k = \left([\Phi_M]^T[\Phi_M]\right)^{-1}[\Phi_M]^T\{q_{Me}\}_k$$

$$\{\dot{\eta}\}_k = \left([\Phi_M]^T[\Phi_M]\right)^{-1}[\Phi_M]^T\{\dot{q}_{Me}\}_k$$

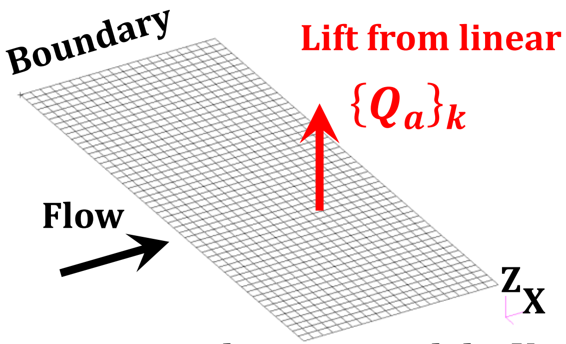
$$\{\ddot{\eta}\}_k = \left([\Phi_M]^T[\Phi_M]\right)^{-1}[\Phi_M]^T\{\ddot{q}_{Me}\}_k$$

$$[\mathbf{M}]\{\ddot{q}(t)\} + [\mathbf{G}]\{\dot{q}(t)\} + [\mathbf{K}]\{q(t)\} = \{Q_a(t)\}$$

$$s^2\,[\mathbf{M}][\Phi]\{\eta(s)\} + s[\mathbf{G}][\Phi]\{\eta(s)\} + [\mathbf{K}][\Phi]\{\eta(s)\} = \{Q_a(s)\}$$

$$s^2[\Phi]^T[\mathbf{M}][\Phi]\{\eta(s)\} + s[\Phi]^T[\mathbf{G}][\Phi]\{\eta(s)\} + [\Phi]^T[\mathbf{K}][\Phi]\{\eta(s)\} = [\Phi]^T\{Q_a(s)\} = \{N(s)\} = q_D[\mathbf{A}(s)]\{\eta(s)\}$$

Lift from linear panel code



aerodynamic model

- Rational function approximation: Select Roger's Approximation

Modal Aerodynamic Influence Coefficient Matrix →

$$[\mathbf{A}(s)] = [\mathbf{D}_0] + s[\mathbf{D}_1] + s^2[\mathbf{D}_2] + \sum_{j=1}^{LT}\frac{s[\mathbf{C}_j]}{s+\Omega_j}$$

- Time marching algorithm:

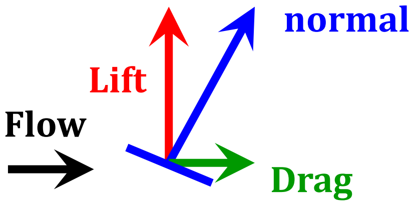
$$\{N\}_k = q_D\left([\mathbf{D}_0]\{\eta\}_k + [\mathbf{D}_1]\{\dot{\eta}\}_k + [\mathbf{D}_2]\{\ddot{\eta}\}_k + [\mathbf{C}]\{x\}_k\right)$$

$$\{x\}_k = [\mathbf{E}]\{x\}_{k-1} + [\theta][\mathbf{B}]\frac{\{\dot{\eta}\}_k + \{\dot{\eta}\}_{k-1}}{2}$$

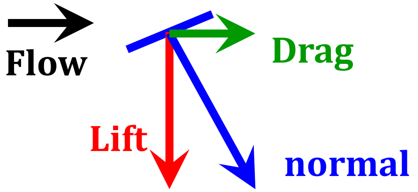
$$[\Phi]^T\{Q_a\}_k = \{N\}_k$$

$$\{Q_a\}_k = \left([\Phi]^T\right)^{-1}\{N\}_k$$

A rectangular matrix $[\Phi]^T$ can be inverted using a singular value decomposition technique.



Side view of the unsteady wing motion



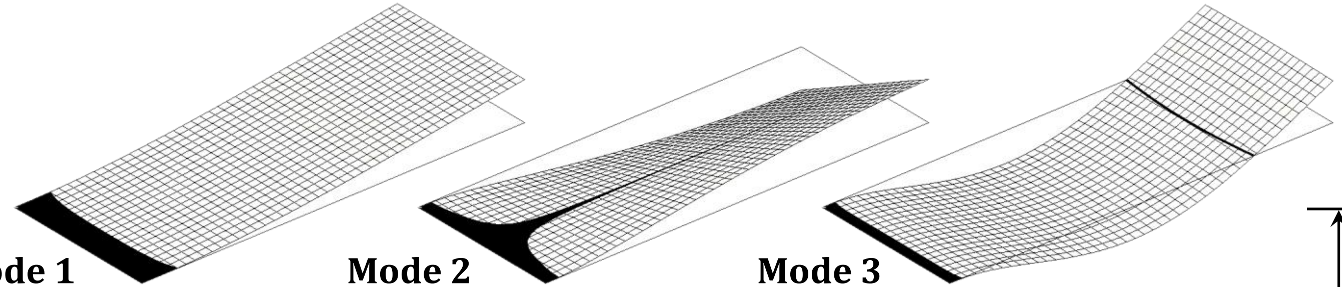
$$[\mathbf{E}] = e^{[\mathbf{A}]T_a} \quad [\theta] = \int_0^{T_a} e^{[\mathbf{A}](T_a-\tau)}d\tau \quad [\mathbf{C}] = [\mathbf{C}_1\ \mathbf{C}_2\ \ldots \mathbf{C}_{LT}] \quad [\mathbf{A}] = \begin{bmatrix} -\Omega_1\mathbf{I} & 0 & \ldots & 0 \\ 0 & -\Omega_2\mathbf{I} & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & -\Omega_{LT}\mathbf{I} \end{bmatrix} \quad [\mathbf{B}] = \begin{bmatrix}\mathbf{I}\\ \mathbf{I}\\ \vdots \\ \mathbf{I}\end{bmatrix} \quad \{x\}_k = \begin{Bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{LT}\end{Bmatrix}_k$$

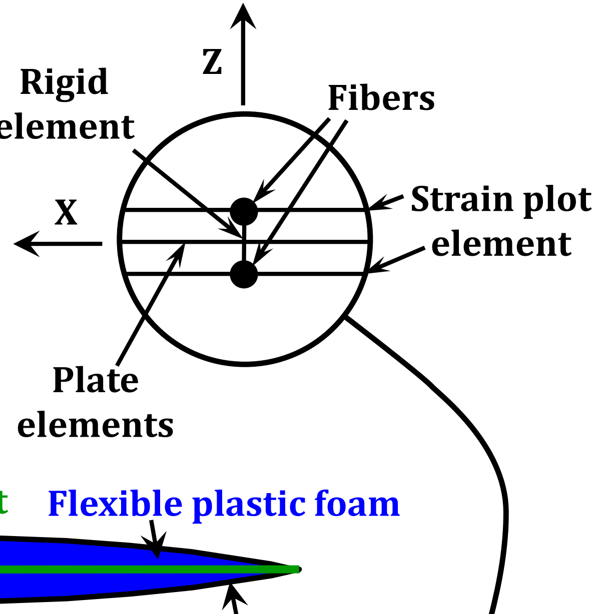
Computational Validation

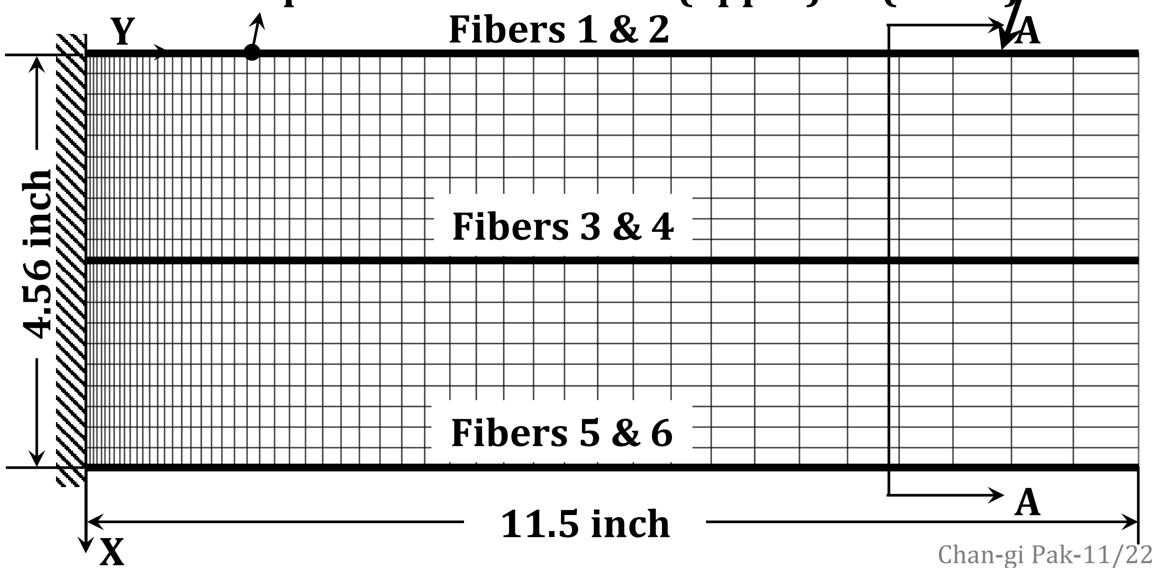
Cantilevered rectangular wing model

Structural Model & Results from Modal Analysis

- Configuration of a wind tunnel test article
  - Has **aluminum insert** (thickness = 0.065 in) covered with **6% circular arc** cross-sectional shape (**plastic foam**)
  - lumped mass weight are computed based on 6% circular-arc cross sectional shape.
    - Use structural dynamic model tuning technique
    - Chan-gi Pak and Samson Truong, "Creating a Test-Validated Finite-Element Model of the X-56A Aircraft Structure," *Journal of Aircraft*, Vol. 52, No. 5, pp. 1644-1667, 2015. doi: http://arc.aiaa.org/doi/abs/10.2514/1.C033043
  - 300 beam elements for fictitious FOSS (50 per each fiber). Zero stiffness and zero weight.
- Modal analysis
  - NASTRAN sol. 103





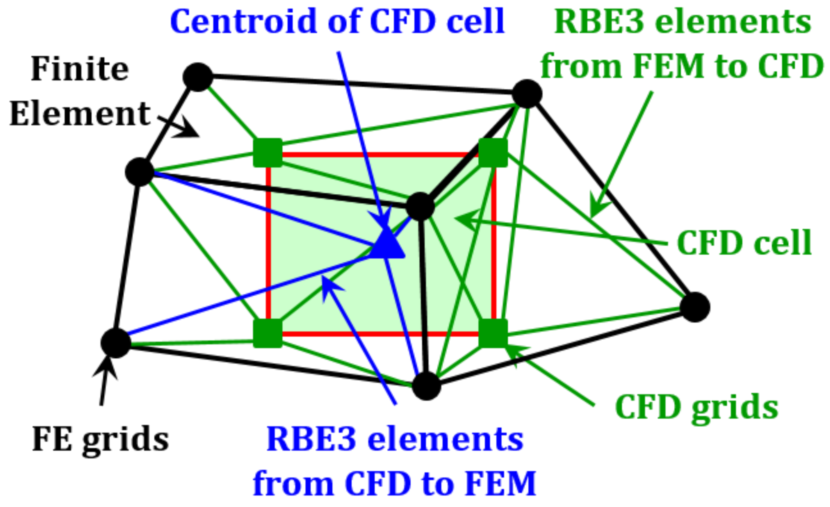


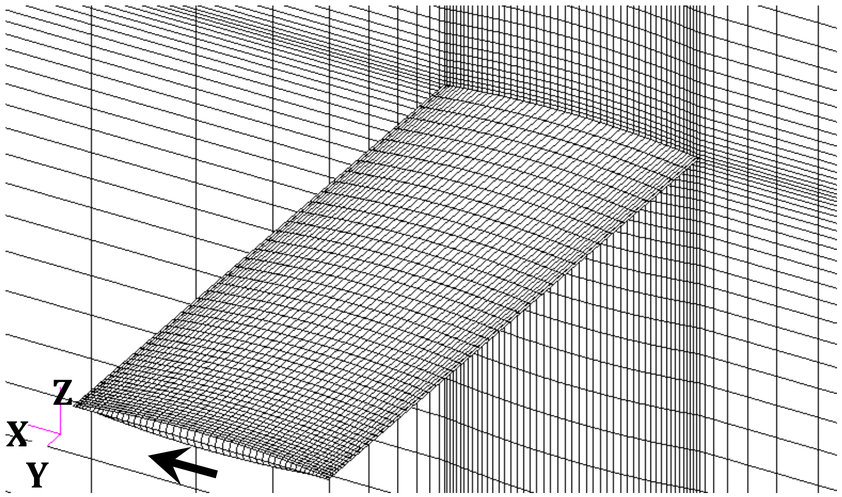
Measured and computed natural frequencies

Mode	Measured (Hz)	Computed (Hz)	% Error
1	14.29	14.29	0.0
2	80.41	80.17	-0.3
3	89.80	89.04	-0.8

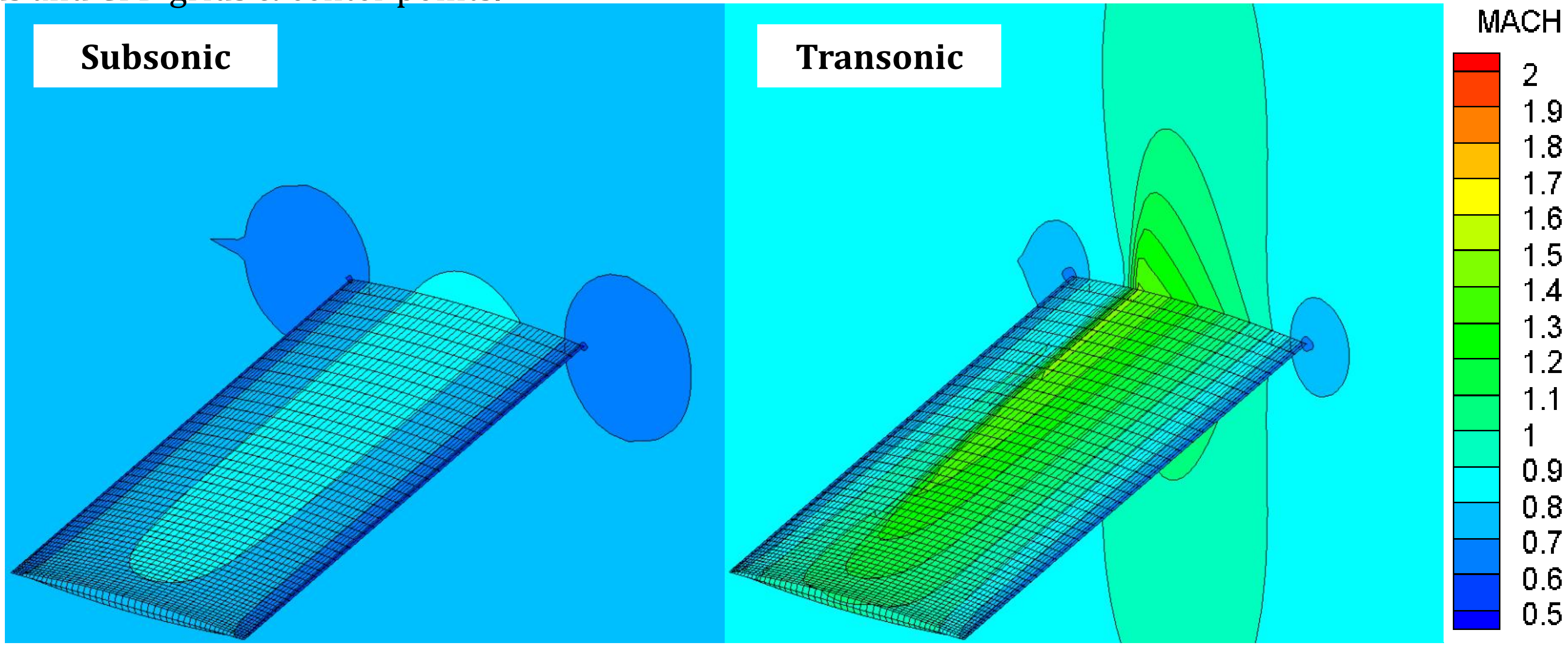
CFL3D Model & Aeroelastic Analysis using CFL3D/NASTRAN

- ❑ **CFL3D code** is used to generate unsteady aerodynamic loads.
  - ❖ **Compute aerodynamic load vector** at structural grid points.
  - ❖ The CFD grid is a multi-block (97 × 73 × 57) grid with H-H topology.
  - ❖ M=0.714 selected. Delta t = 0.000060515 sec. 10240 time steps
  - ❖ The first three flexible modes are used.
  - ❖ Computes **deflections** and **velocities**. (compare with NASTRAN results)
- ❑ **MSC/NASTRAN sol 112: to compute unsteady strain**
  - ❖ Modal transient response analysis with 1024 time steps, Delta t = 0.00060515 sec.
  - ❖ Force cards are obtained from CFL3D code. Available @ CFD center points.
  - ❖ **Computes strain** (assume measured value), deflection, velocity, & **acceleration** (**target**)
- ❑ **Splines between CFL3D and NASTRAN**
  - ❖ **Develop new approach.**
  - ❖ Use interpolation element, RBE3, between FE grids and CFD grids & center points.
    - ➢ CFD grids: pressure
    - ➢ CFD center points: aerodynamic load vector





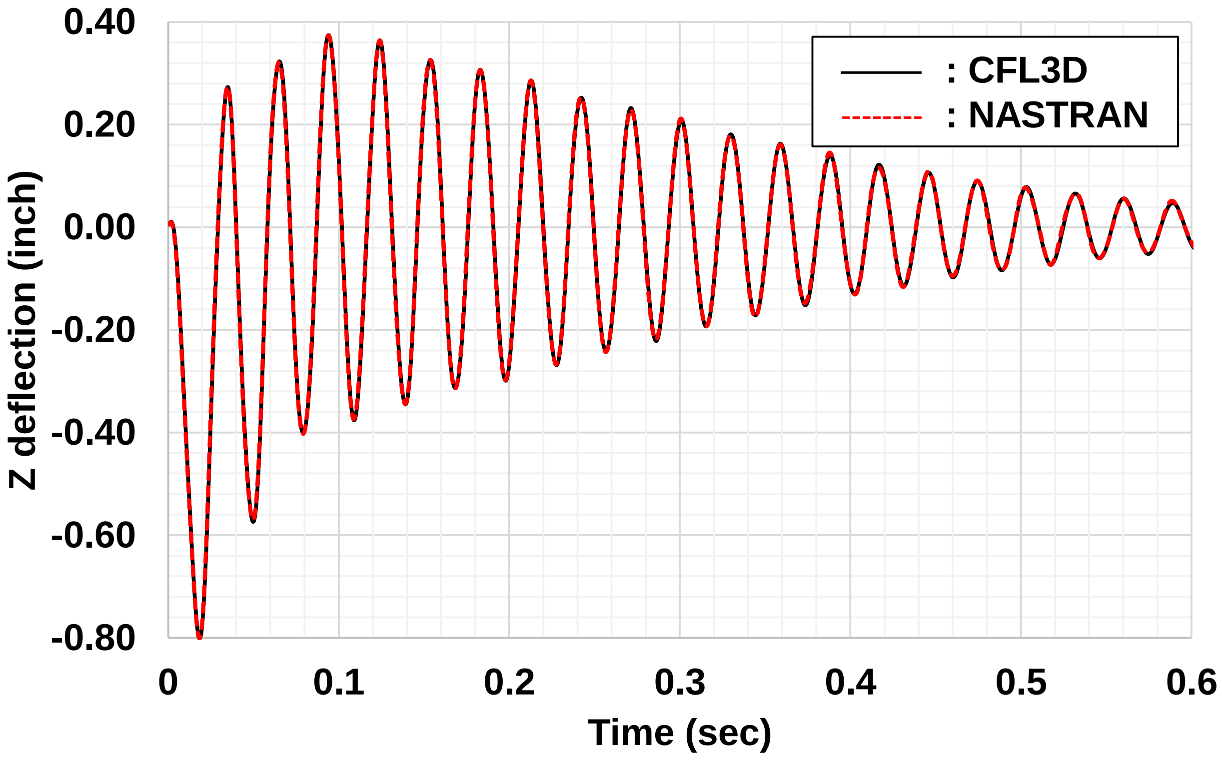
Flow direction



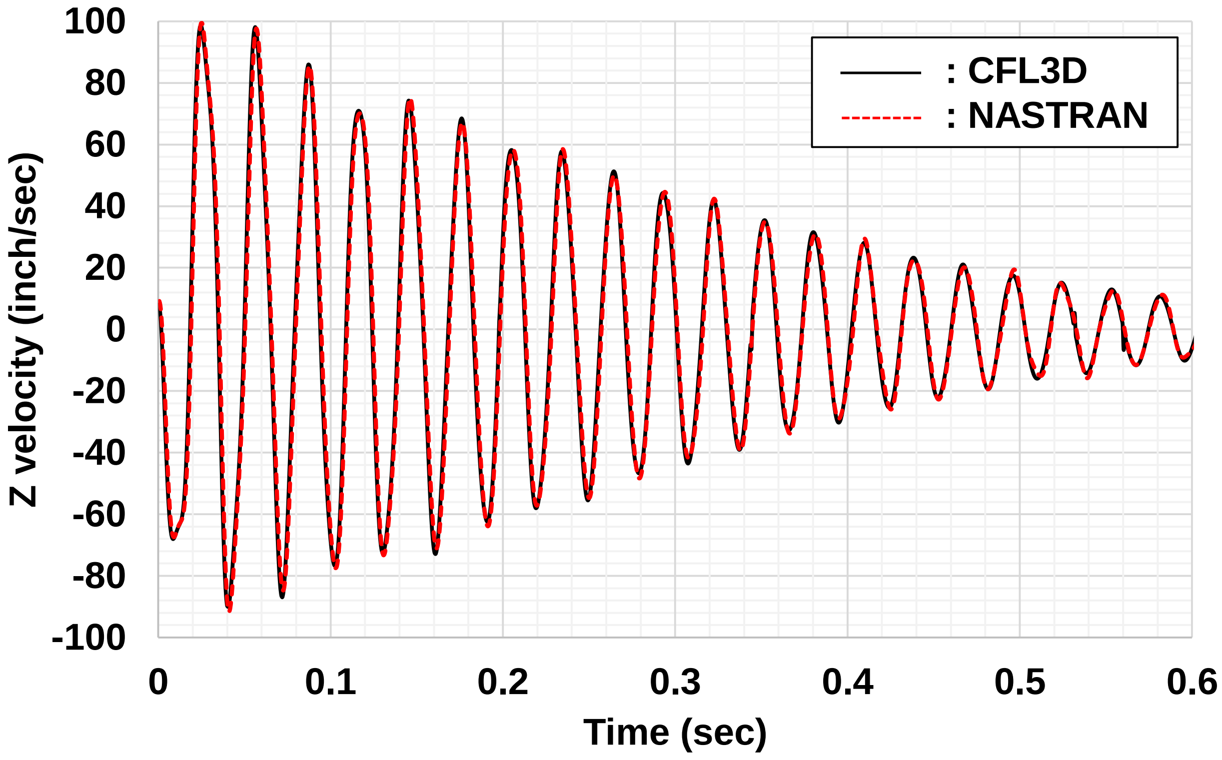
M=0.714 M=0.875

CFL3D vs. NASTRAN: deflection & velocity

$q_d = 1.455$



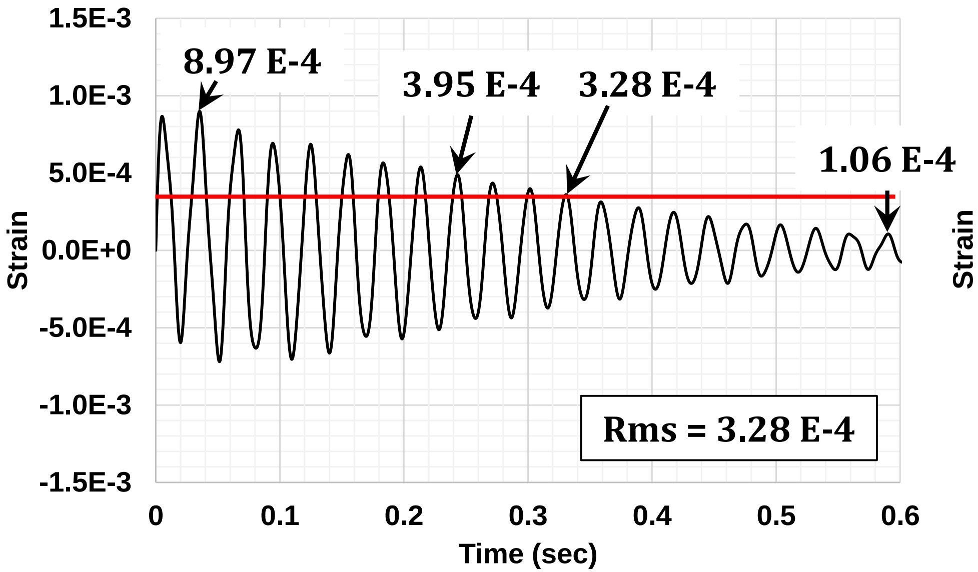
(a) Deflection



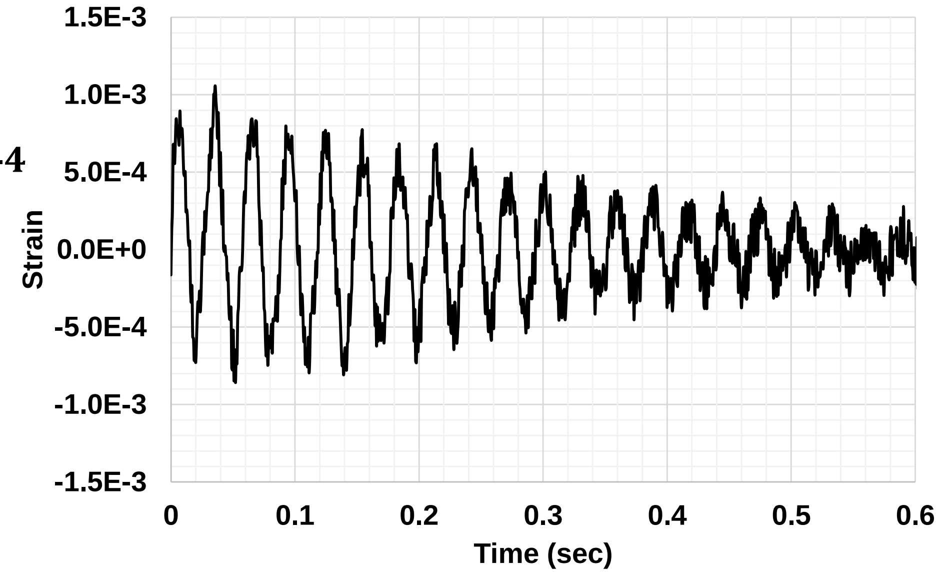
(b) Velocity

$q_{df} = 1.4561$: Dynamic pressure for wing flutter condition

Time Histories of Strain under Different Levels of Random White Noise

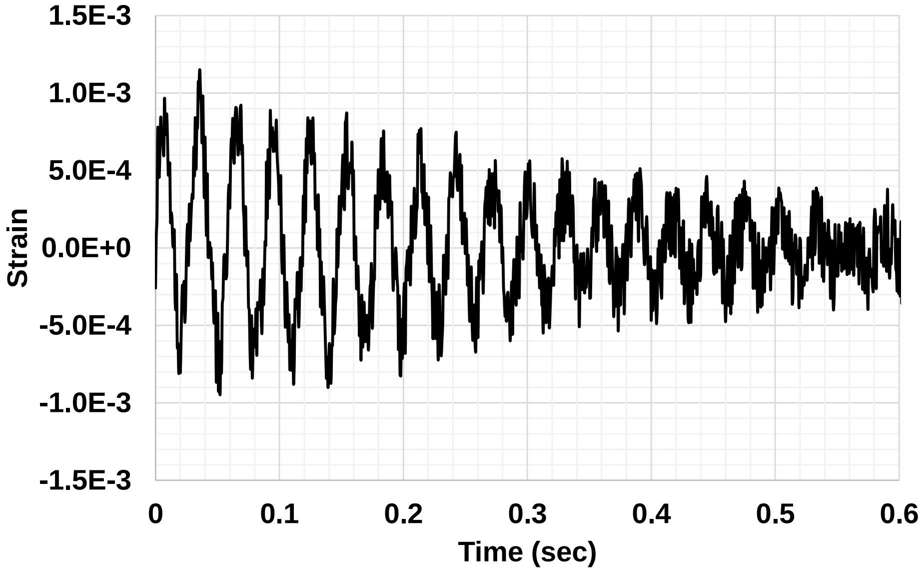


(a) Without noise

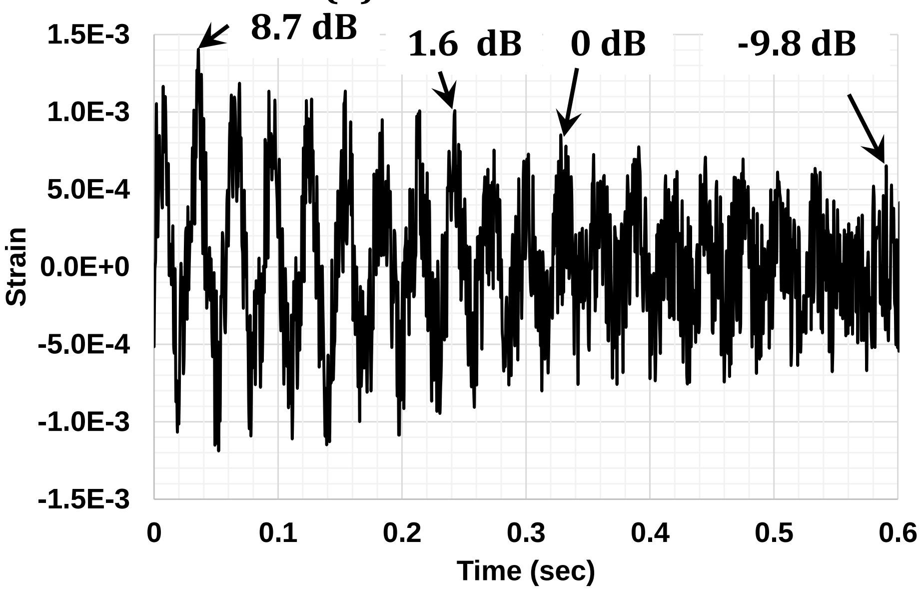


(b) SNR = 10 dB

- ❑ Strain is measured at the leading-edge of wing root section (upper surface).
- ❑ $SNR \equiv 20 \times log_{10} \frac{\epsilon_{rms}}{n_{rms}}$
  - ❖ ϵ_{rms} root-mean-squared level of strain
  - ❖ n_{rms} root-mean-squared level of noise
- ❑ SNR value is correct near 0.33 sec.



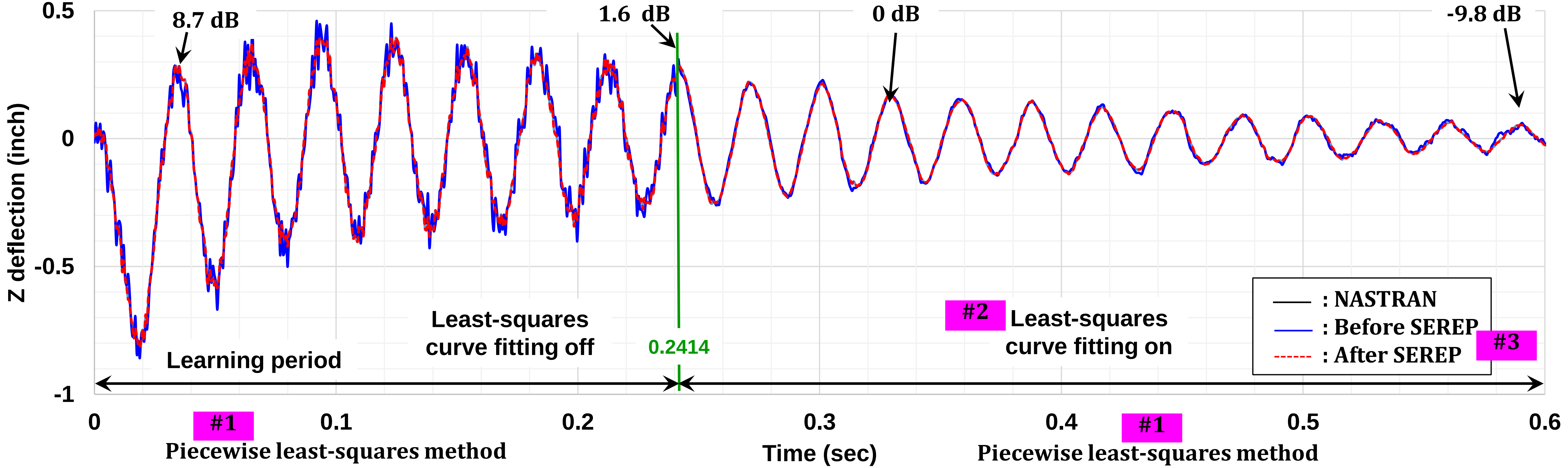
(c) SNR = 6 dB



(d) SNR = 0 dB

- ❑ $LSNR \equiv 20 \times log_{10} \frac{\epsilon_{max}}{n_{rms}}$
  - ❖ ϵ_{max} local maximum strain
  - ❖ n_{rms} root-mean-squared level of noise
- ❑ LSNR @ 0.035 sec
  - ❖ $20log_{10}(8.97/3.28) = 8.74$ dB
- ❑ LSNR @ 0.24 sec
  - ❖ $20log_{10}(3.95/3.28) = 1.61$ dB
- ❑ LSNR @ 0.33 sec
  - ❖ $20log_{10}(3.28/3.28) = 0$ dB
- ❑ LSNR @ 0.59 sec
  - ❖ $20log_{10}(1.06/3.28) = -9.83$ dB

Time Histories of Z Deflection: SNR = 0 dB

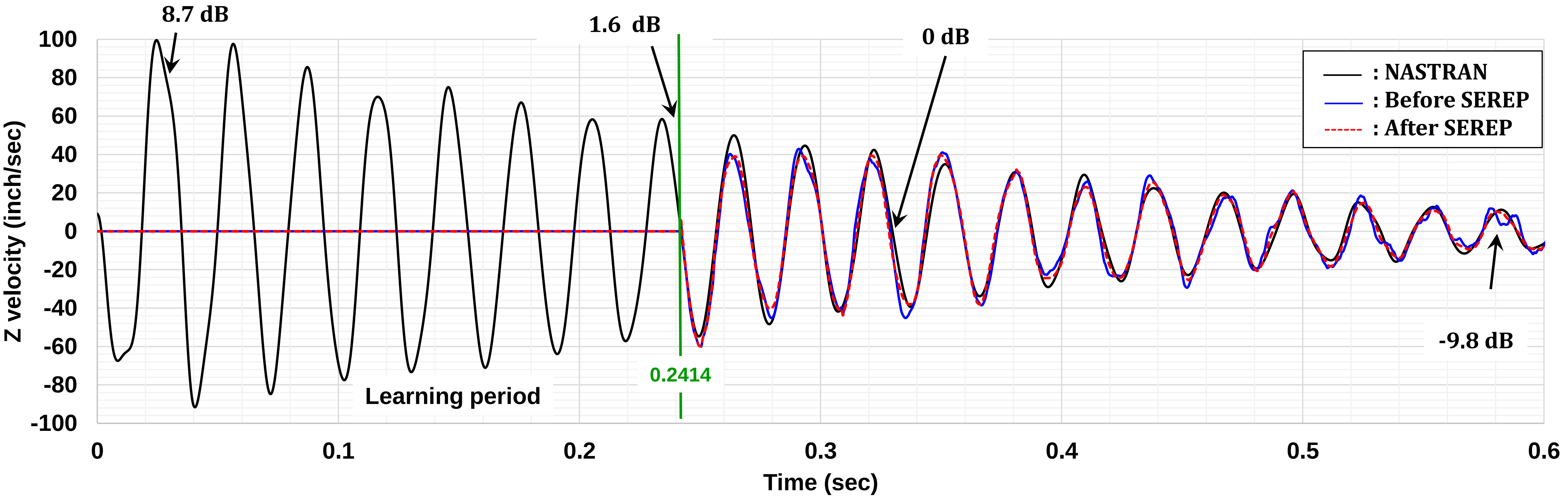


- Z deflection is computed at the leading-edge of wing tip section (upper surface).
- Time interval: 0 – 0.2414 sec
  - Learning period for on-line parameter estimator.
  - Effect of piecewise least squares method can be observed. (first step of two step approach)
- Time interval: 0.2141 sec – 0.6 sec
  - Least-squares curve fitting method is on.
  - Working even with **"SNR = 0 dB"**
- Effect of SEREP transformation can be observed.
  - SEREP transformation is a kind of least-squares surface fitting approach.
  - Noise in the signal after the first step of the two step approach is further filtered using SEREP transformation.

$$\{q_{Me}(t)\} = \{\tilde{q}_{Me}\} + \sum_{i=1}^{nm} e^{-\sigma_i t}\{A_i cos(\omega_{di} t) + B_i sin(\omega_{di} t)\}$$

$$\{q\}_k = \begin{bmatrix}\Phi_M \\ \Phi_S\end{bmatrix}\{\eta\}_k \qquad \{\eta\}_k = \left([\Phi_M]^T[\Phi_M]\right)^{-1}[\Phi_M]^T\{q_{Me}\}_k$$

Time Histories of Z Velocity: SNR = 0 dB



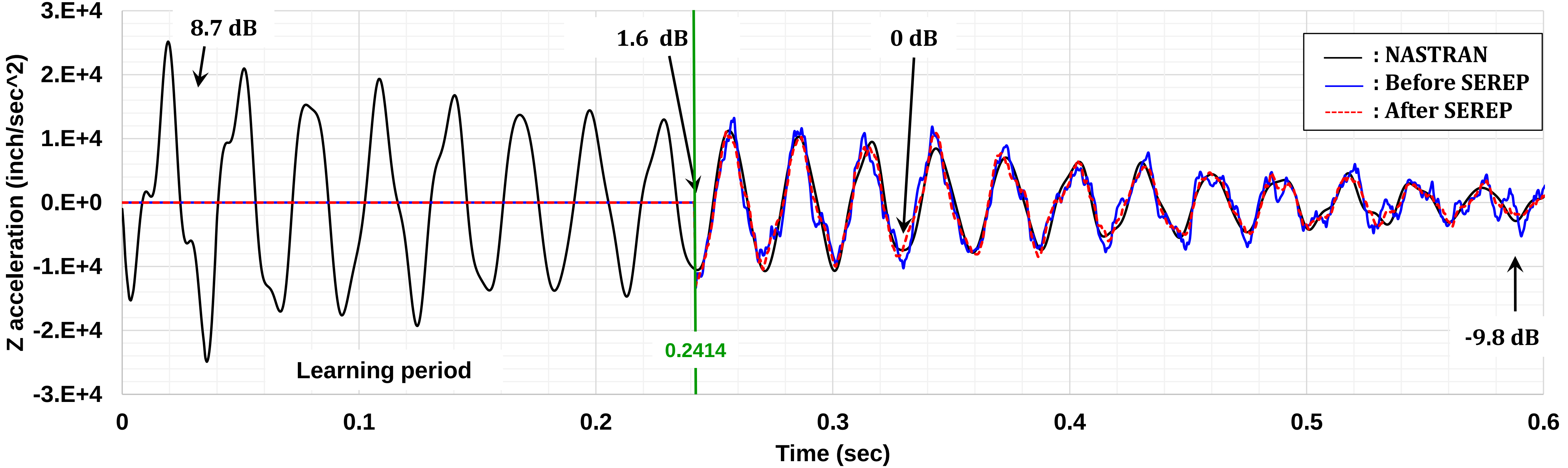
- Z velocity is computed at the leading-edge of wing tip section (upper surface).
- Time interval: 0 – 0.2414 sec
  - Learning period for on-line parameter estimator.
  - Velocities are not computed during this period.
- Time interval: 0.2141 sec – 0.6 sec
  - Least-squares curve fitting method is on.
  - Working even with **"SNR = 0 dB"**

$$\{\dot{q}_{Me}(t)\} = \frac{d}{dt}\{q_{Me}(t)\}$$

$$\{q_{Me}(t)\} = \{\tilde{q}_{Me}\} + \sum_{i=1}^{nm} e^{-\sigma_i t}\{A_i cos(\omega_{di} t) + B_i sin(\omega_{di} t)\}$$

$$\{\dot{q}\}_k = \begin{bmatrix}\Phi_M \\ \Phi_S\end{bmatrix}\{\dot{\eta}\}_k \qquad \{\dot{\eta}\}_k = \left([\Phi_M]^T[\Phi_M]\right)^{-1}[\Phi_M]^T\{\dot{q}_{Me}\}_k$$

Time Histories of Z Acceleration: SNR = 0 dB



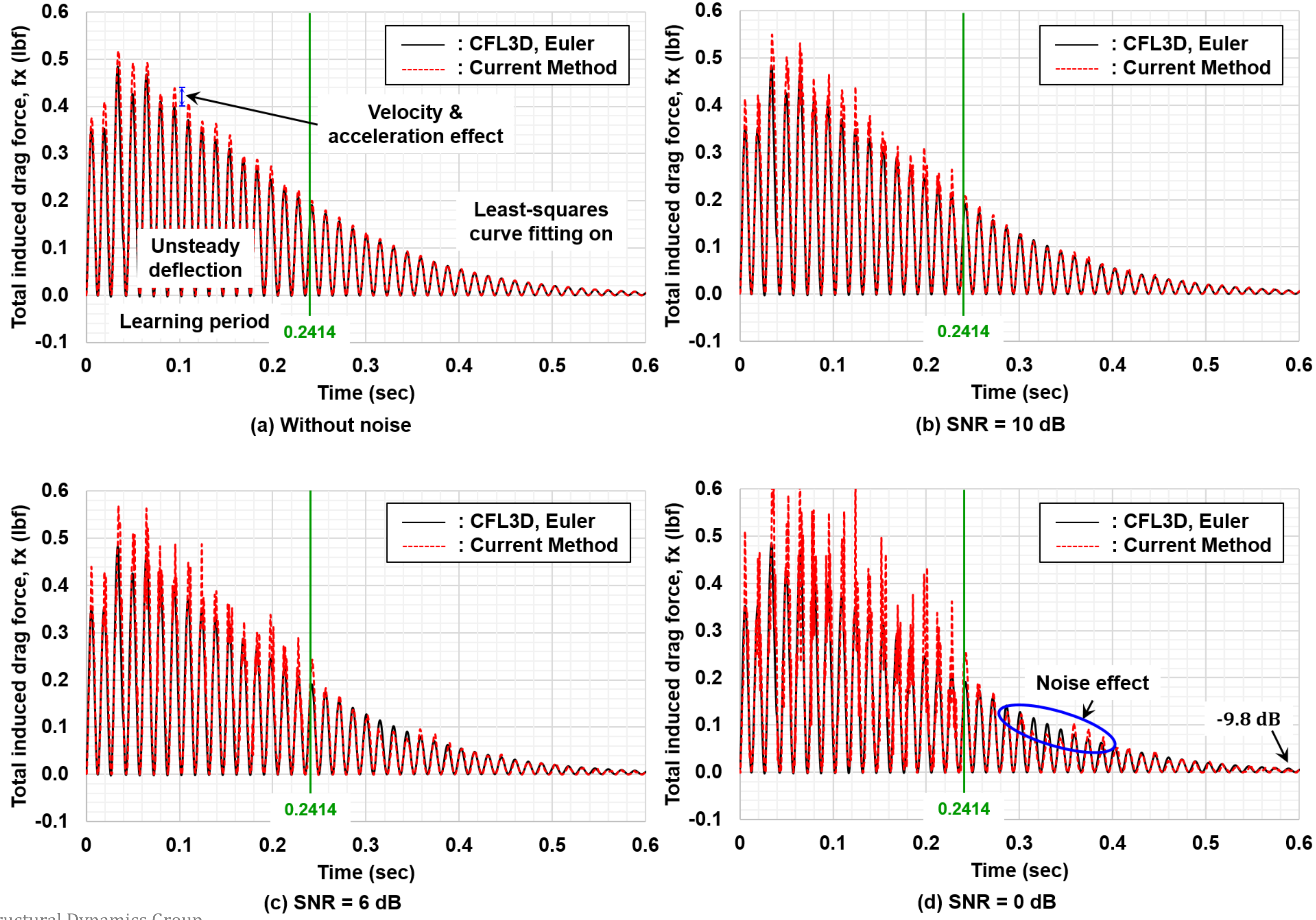
- Z acceleration is computed at the leading-edge of wing tip section (upper surface).
- Time interval: 0 – 0.2414 sec
  - Learning period for on-line parameter estimator.
  - Accelerations are not computed during this period.
- Time interval: 0.2141 sec – 0.6 sec
  - Least-squares curve fitting method is on.
  - Working even with **"SNR = 0 dB"**

$$\{\ddot{q}_{Me}(t)\} = \frac{d^2}{dt^2}\{q_{Me}(t)\}$$

$$\{q_{Me}(t)\} = \{\tilde{q}_{Me}\} + \sum_{i=1}^{nm} e^{-\sigma_i t}\{A_i cos(\omega_{di}t) + B_i sin(\omega_{di}t)\}$$

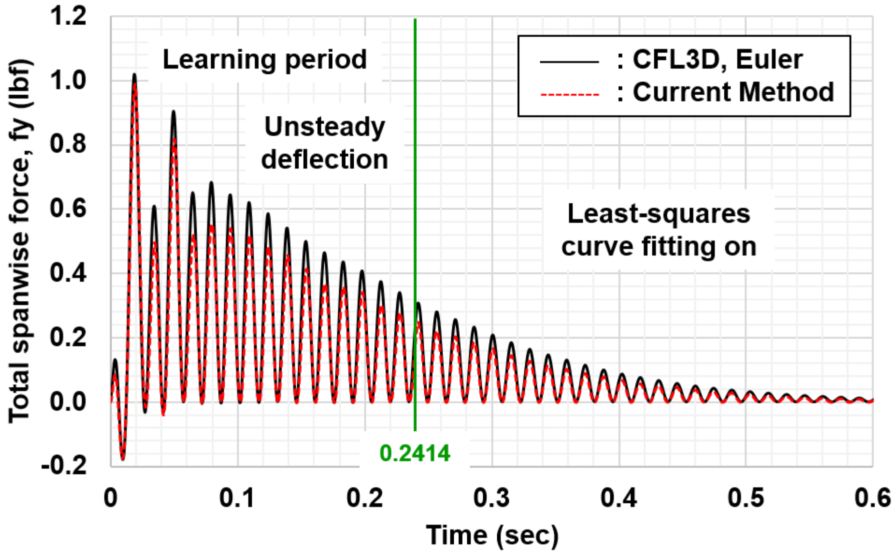
$$\{\ddot{q}\}_k = \begin{bmatrix}\Phi_M \\ \Phi_S\end{bmatrix}\{\ddot{\eta}\}_k \qquad \{\ddot{\eta}\}_k = \left([\Phi_M]^T[\Phi_M]\right)^{-1}[\Phi_M]^T\{\ddot{q}_{Me}\}_k$$

Time Histories of Total Induced Drag Load under Different Levels of Random White Noise

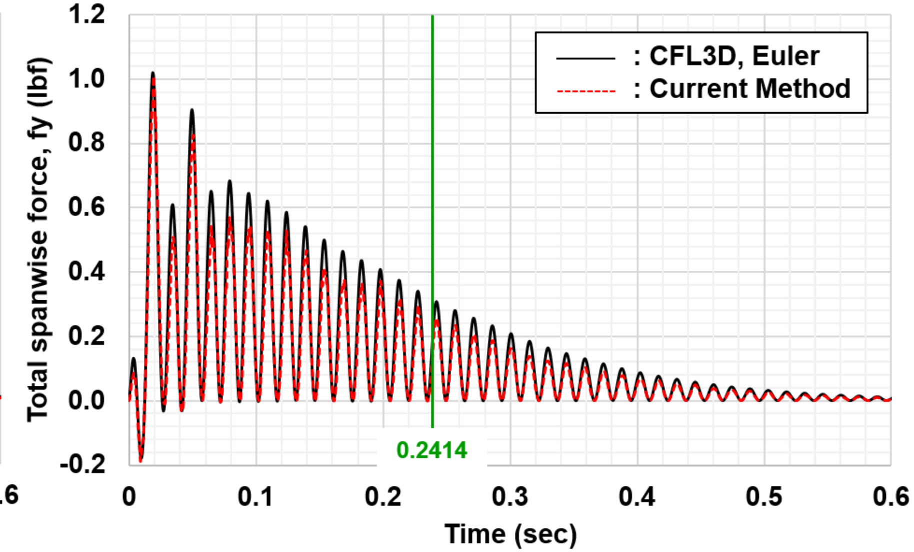


- Time interval: 0 – 0.2414 sec
  - Learning period for on-line parameter estimator.
  - Load computations are based on **wing deflection only**.
- Time interval: 0.2141 sec – 0.6 sec
  - **Least-squares curve fitting method is on.**
  - Big difference before and after the proposed method is on.
  - **Working even with "SNR = 0 dB"**
- CFL3D calculation
  - Subtracted 0.0353 (thickness effect)

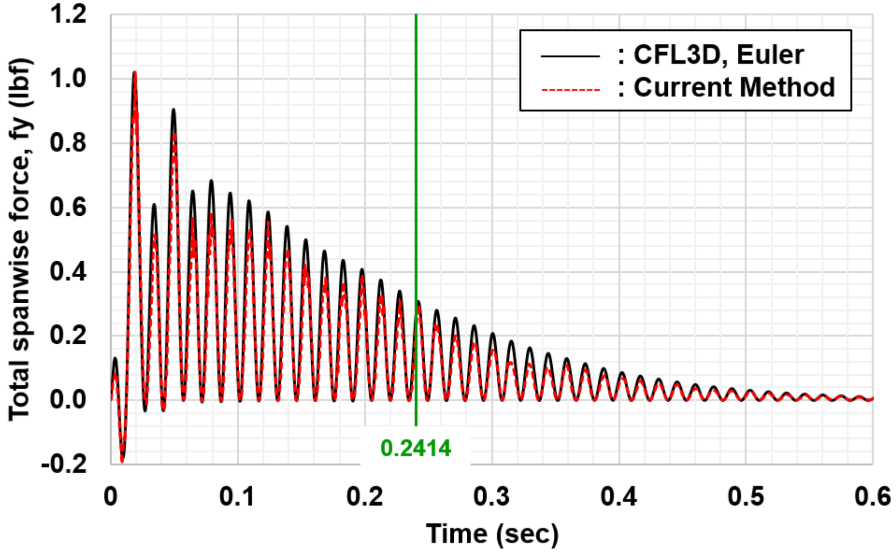
Time Histories of Total Spanwise Load under Different Levels of Random White Noise



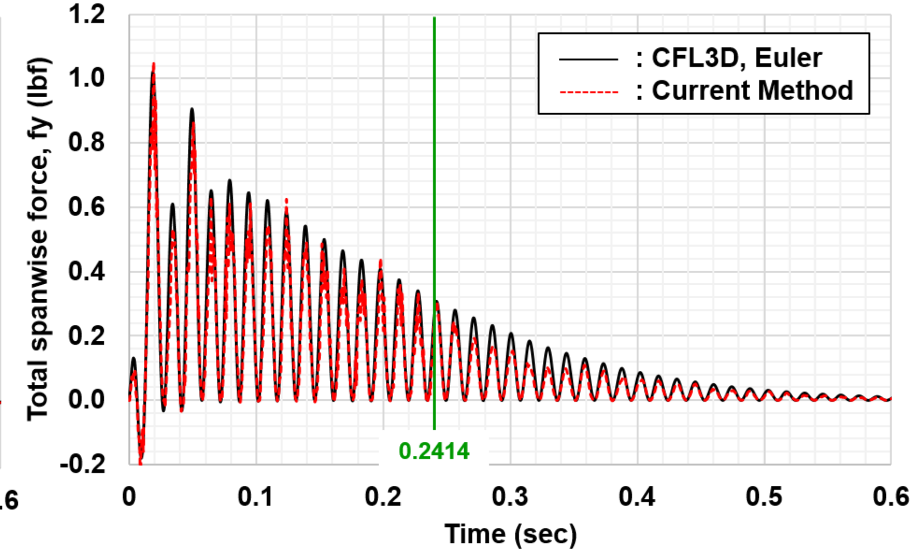
(a) Without noise



(b) SNR = 10 dB



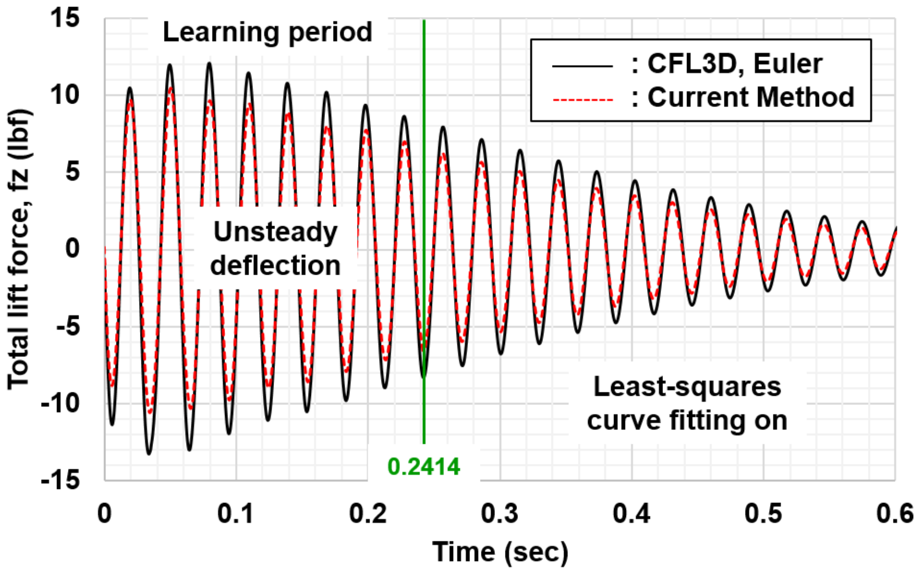
(c) SNR = 6 dB



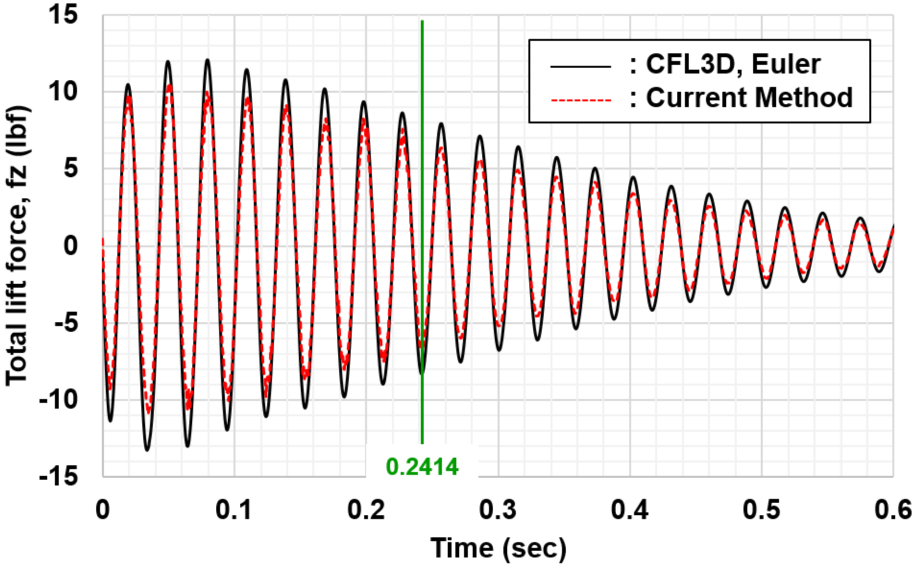
(d) SNR = 0 dB

- ❑ Time interval: 0 – 0.2414 sec
  - ❖ Learning period for on-line parameter estimator.
  - ❖ Load computations are based on wing deflection only.
- ❑ Time interval: 0.2141 sec – 0.6 sec
  - ❖ Least-squares curve fitting method is on.
  - ❖ Big difference before and after the proposed method is on.
  - ❖ Working even with **“SNR = 0 dB”**
- ❑ CFL3D calculation
  - ❖ Subtracted 0.0961 (thickness effect)

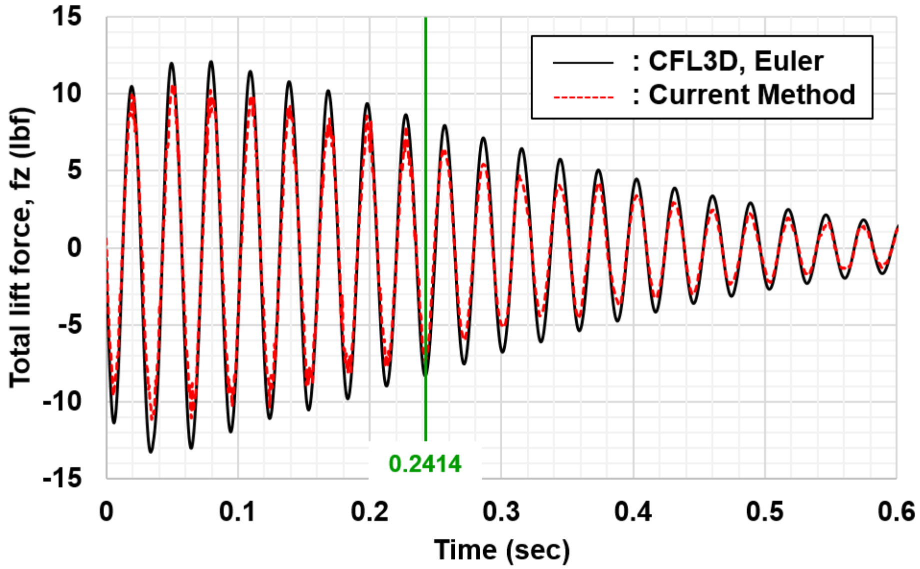
Time Histories of Total Lift Load under Different Levels of Random White Noise



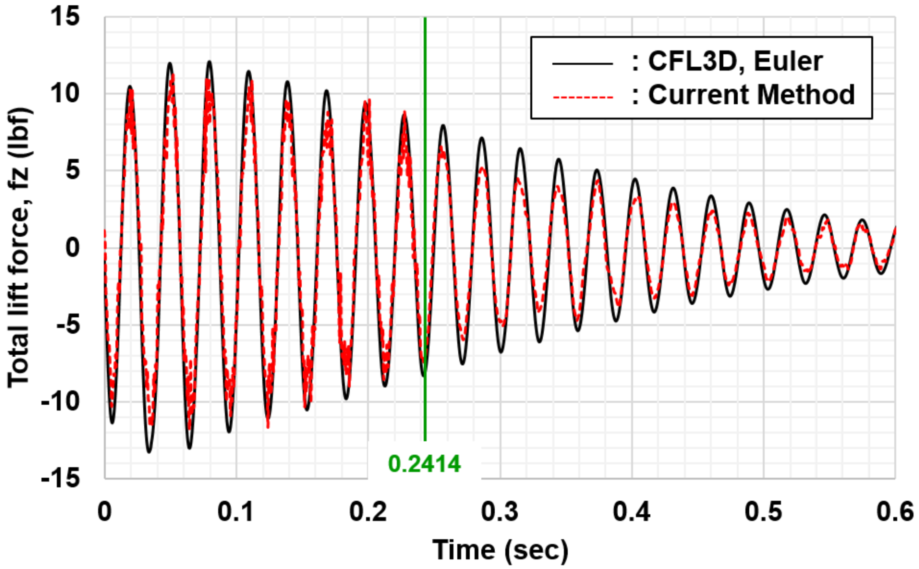
(a) Without noise



(b) SNR = 10 dB



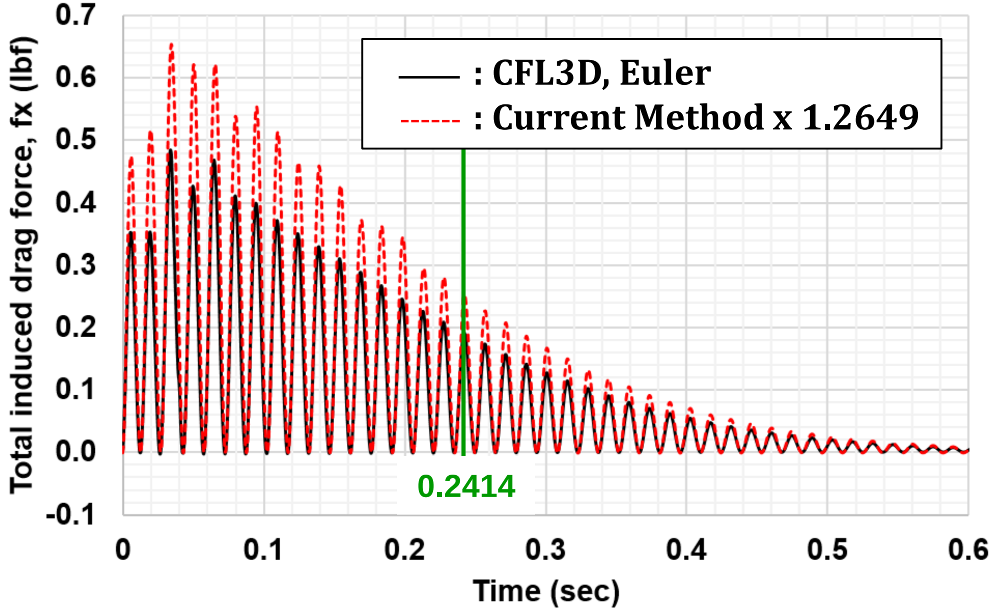
(c) SNR = 6 dB

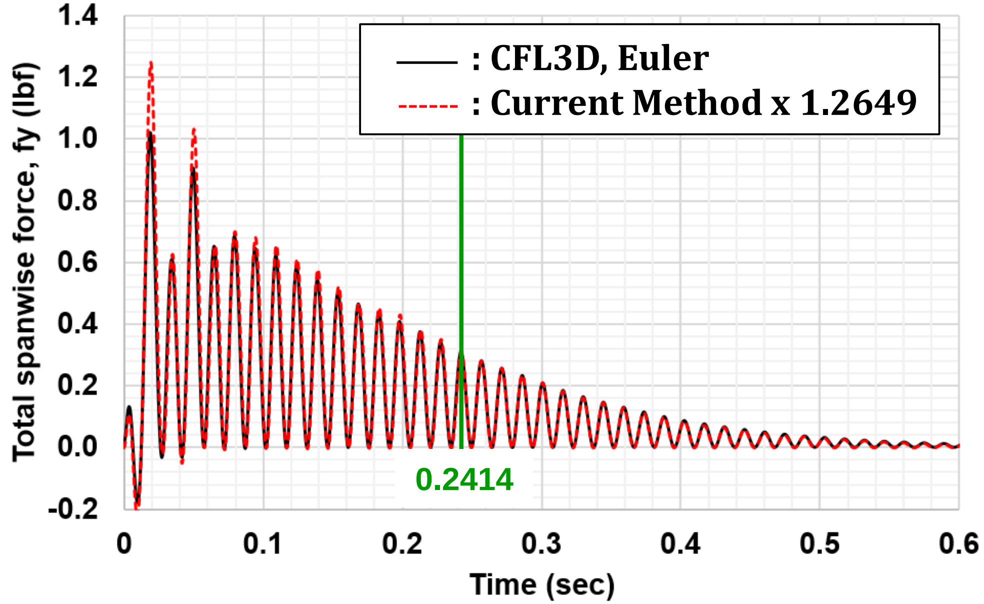


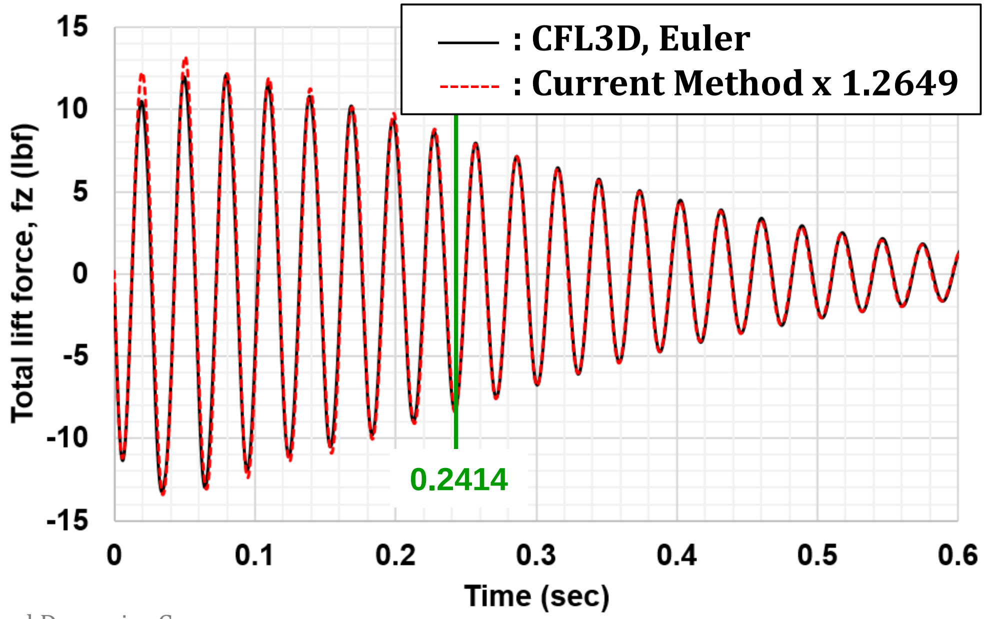
(d) SNR = 0 dB

- ❑ Time interval: 0 – 0.2414 sec
  - ❖ Learning period for on-line parameter estimator.
  - ❖ Load computations are based on wing deflection only.
- ❑ Time interval: 0.2141 sec – 0.6 sec
  - ❖ Least-squares curve fitting method is on.
  - ❖ Big difference before and after the proposed method is on.
  - ❖ Working even with **"SNR = 0 dB"**

Updating aerodynamic forces using scaling factor





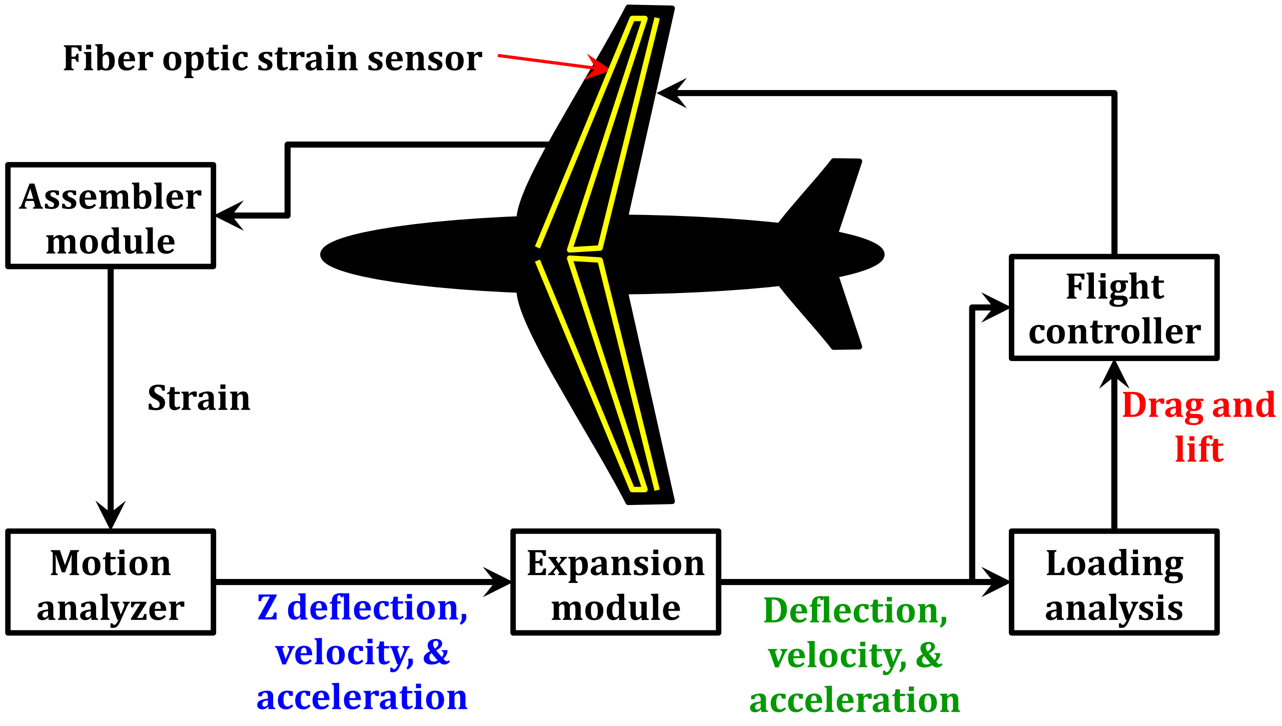


- Scaling factor = 1.2649
  - Pak, C.-g., "Unsteady Aerodynamic Model Tuning for Precise Flutter Prediction," *AIAA Journal of Aircraft*, Vol. 48, No. 6, 2011, pp. 2178 – 2184.
  - Scaling factors for the ATW2 wing were 1.2579 and 1.2719.
    - Scaling between flight test and ZAERO code based linear panel theory.
  - Use average of 1.2579 & 1.2719 for updating the unsteady aerodynamic forces.
    - Scaling between CFL3D code based Euler theory and ZAERO code based linear panel theory.

Conclusions

- **Unsteady aerodynamic loads are computed using simulated measured strain data.**
  - Unsteady structural deflections are computed using the **two-step approach**.
  - Unsteady velocities and accelerations are computed using the **ARMA model, on-line parameter estimator, low pass filter, and a least-squares curve fitting method** together with an **analytical derivatives with respect to time**.
  - The deflections, velocities, and accelerations at each sensor location is **independent of structural and aerodynamic models**.
  - **The distributed strain data together with the current proposed approaches can be used as a distributed deflection, velocity, and acceleration sensors.**
- **Induced drag loads, spanwise loads, and lift loads are obtained from the orthonormalized deflection, velocity, and acceleration** together with the following approaches.
  - The modal AIC matrices are fitted in Laplace-domain using **Roger's approximation**.
  - Laplace-domain aerodynamics are converted to the time-domain using **time-marching algorithm**.
  - Orthonormalized aerodynamic load vectors are transformed to the general coordinates using **pseudo matrix inversion based on singular value decomposition**.
  - Normal vectors to the oscillating wing surface are used to compute drag and spanwise loads.
  - **An active induced drag control system can be designed using these two computed aerodynamic loads, induced drag and lift, to improve the fuel efficiency of an aircraft.**
- **Interpolation elements (RBE3 in MSC/NASTRAN terminology) between structural FE grids and the CFD grids are successfully incorporated with the unsteady aeroelastic computation scheme.**
  - The numerical issues often associated with the **Harder and Desmarais surface splines technique** are bypassed through the use of the current technique with RBE3 elements.
- The deflection, velocity, and acceleration computation based on the proposed least-squares curve fitting method are validated with respect to the **unsteady strain with SNR of 10dB, 6dB, & 0dB (LSNR of 8.7dB to -9.8dB)**.
- **The most critical technology for the success of the proposed approach** is the **robust on-line parameter estimator** since the least-squares curve fitting method depends heavily on aeroelastic system frequencies and damping factors.

Questions ?

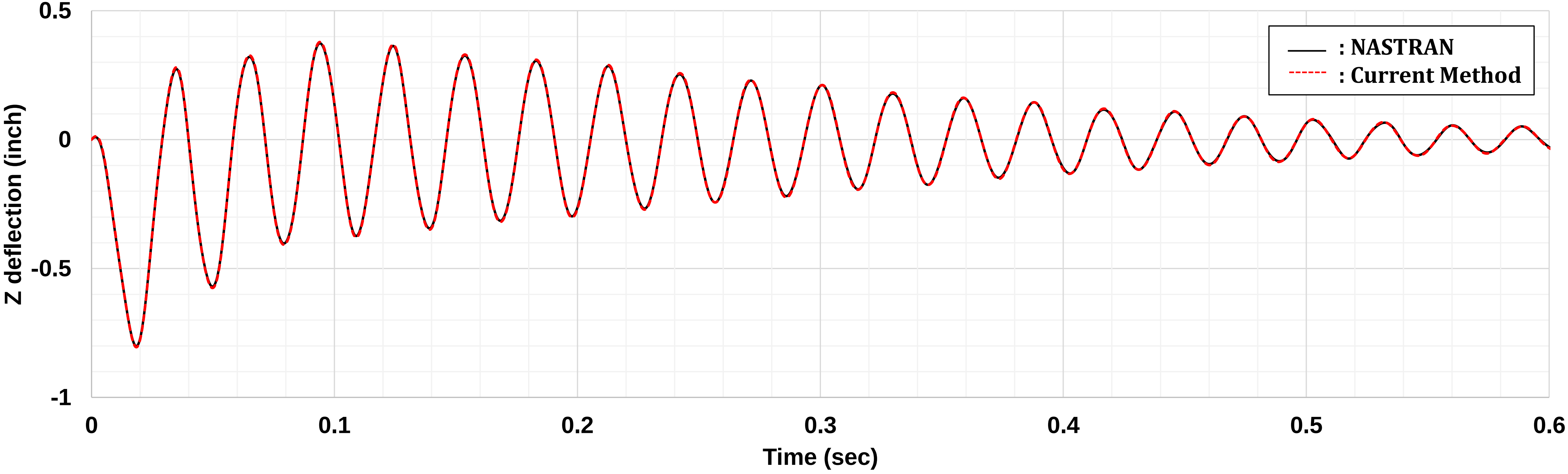


Unsteady Strain

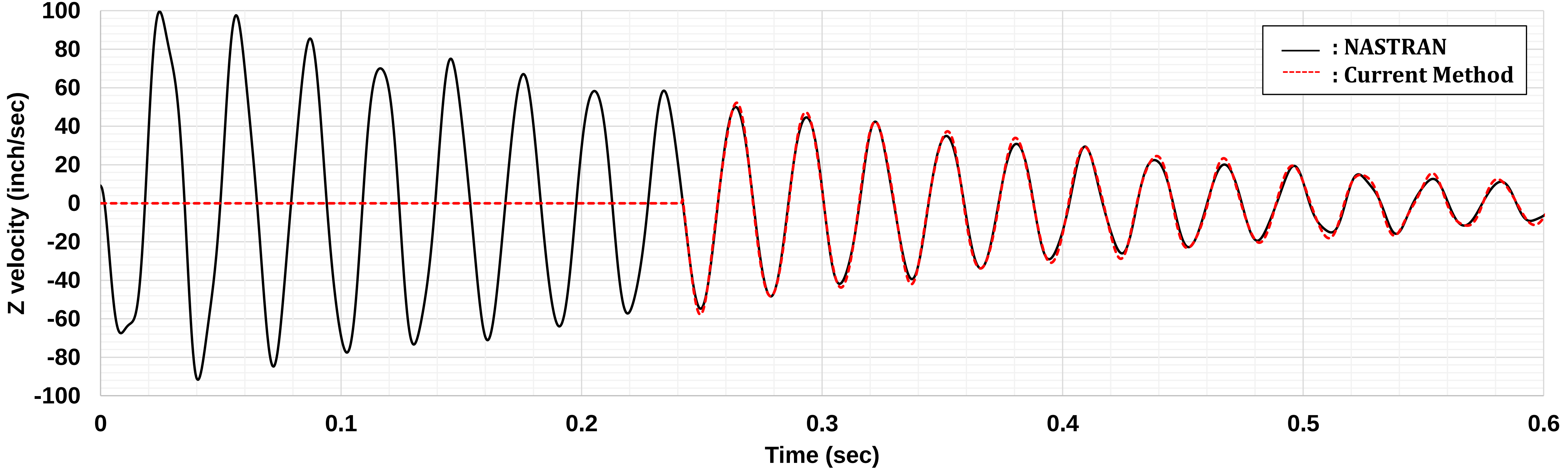
Unsteady Motion

Unsteady Aerodynamic Loads

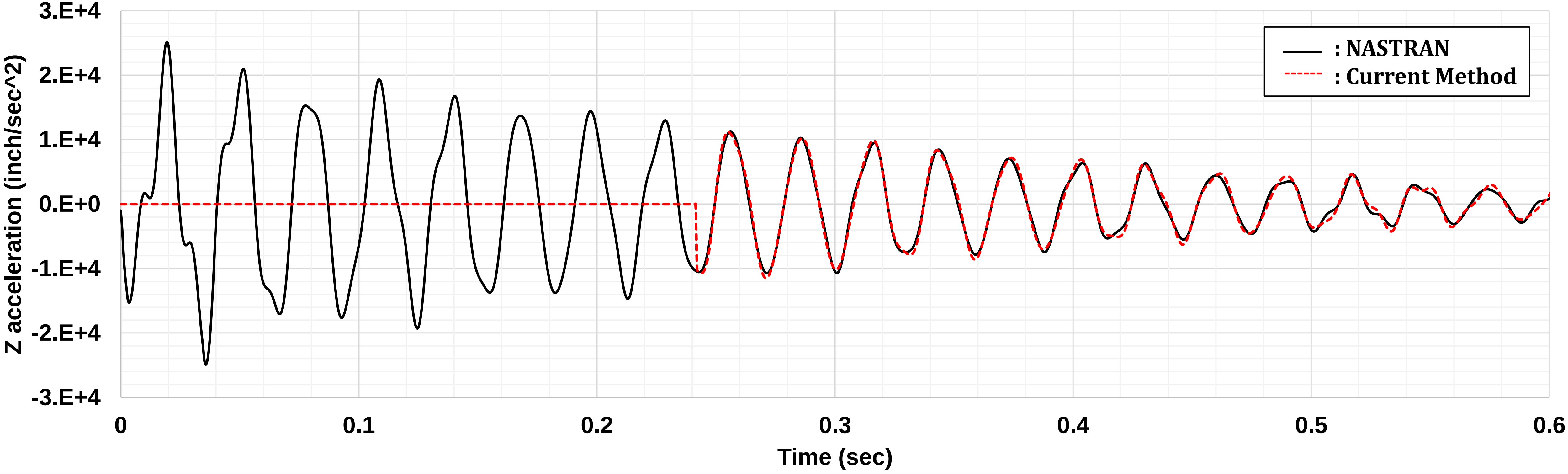
Time Histories of Z Deflection



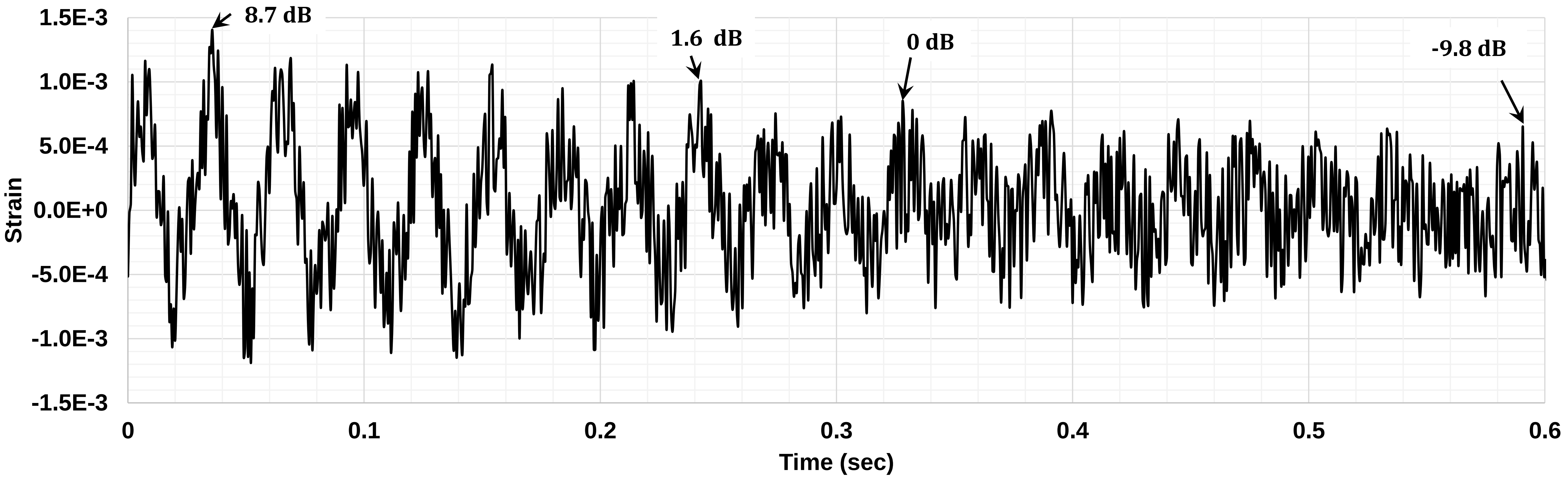
Time Histories of Z Velocity



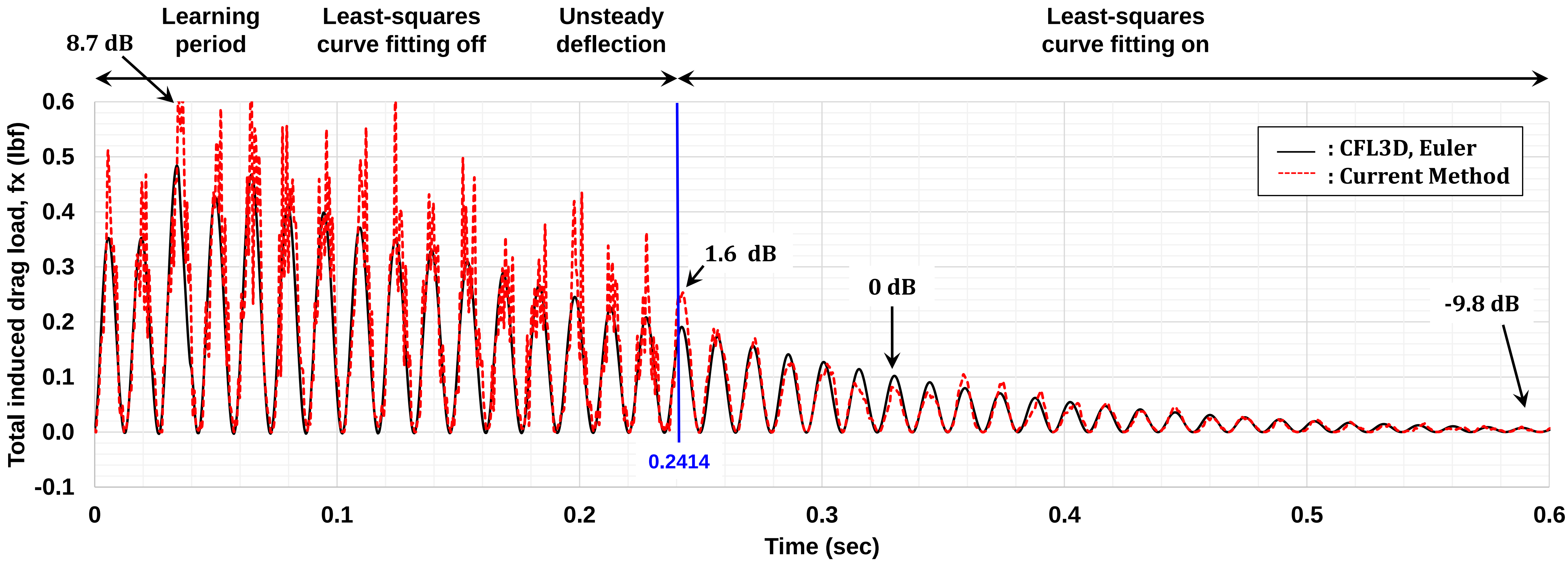
Time Histories of Z Acceleration



Time Histories of Strain under 0 dB Random White Noise

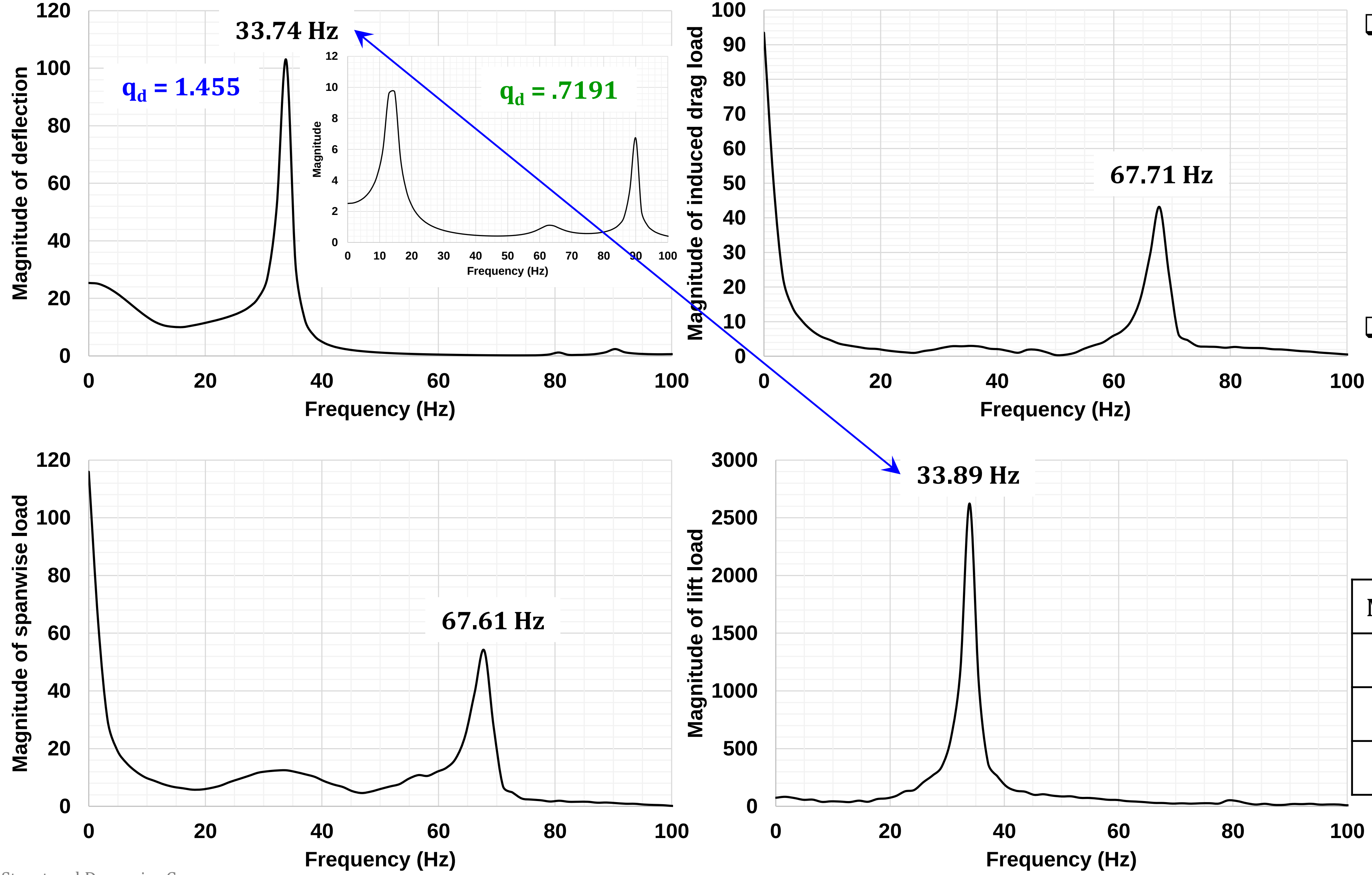


Time Histories of Total Induced Drag Load under 0 dB Random White Noise



- CFL3D calculation
  - Subtracted 0.0353 (thickness effect)

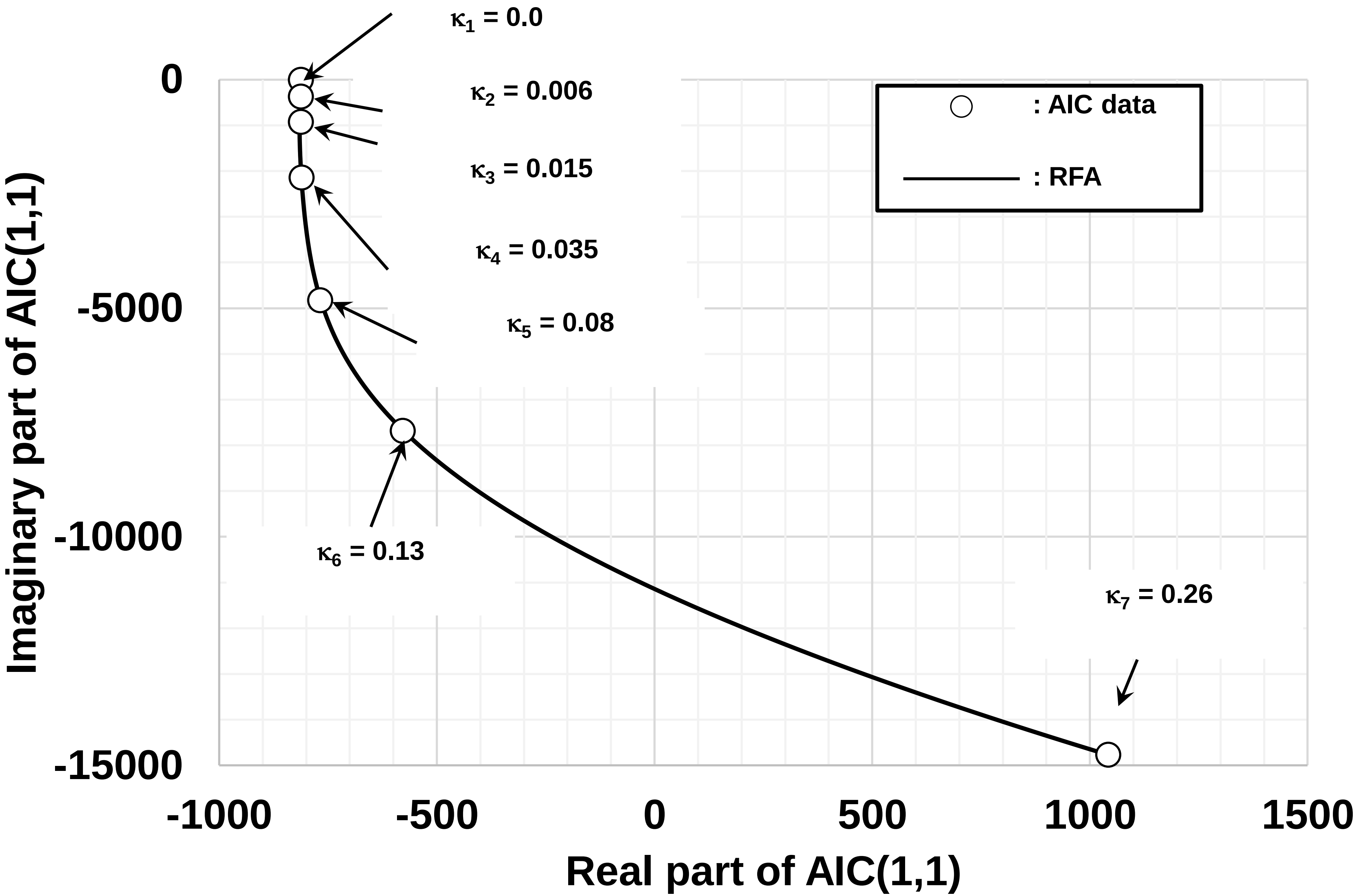
Aeroelastic System Frequencies



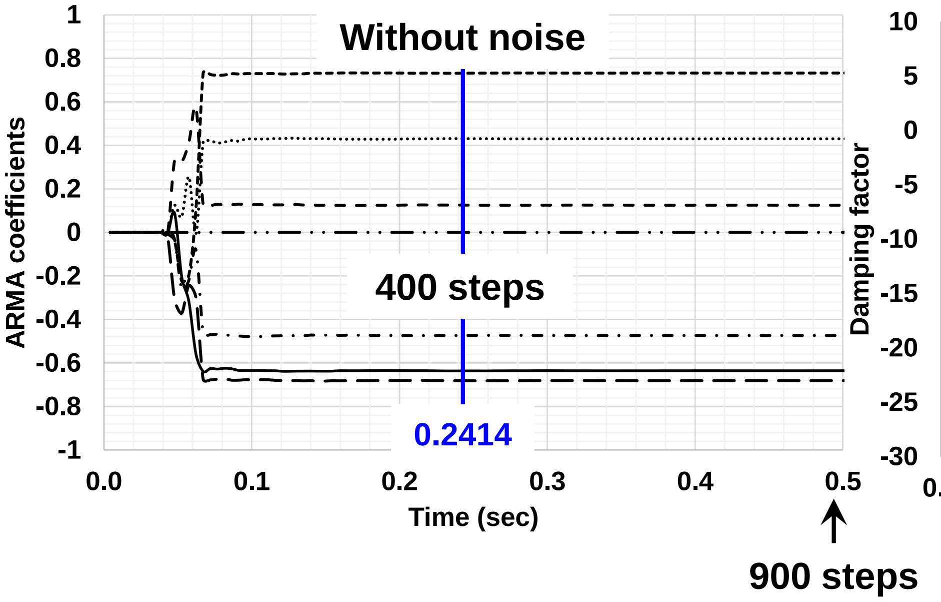
- From AIC computations
  - 0 Hz
  - 4.006Hz
  - 10.02Hz
  - 23.37Hz
  - 53.42Hz
  - 86.80Hz
  - 173.6Hz
- Aerodynamic lag terms
  - 11.81Hz
  - 47.22Hz
  - 106.2Hz
  - 188.9Hz

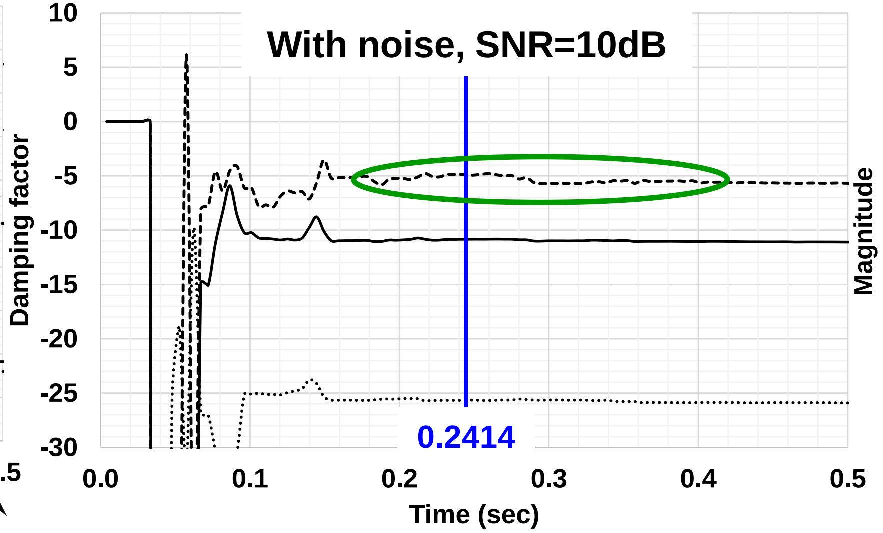
Mode	Natural frequency (Hz)
1	14.29
2	80.17
3	89.04

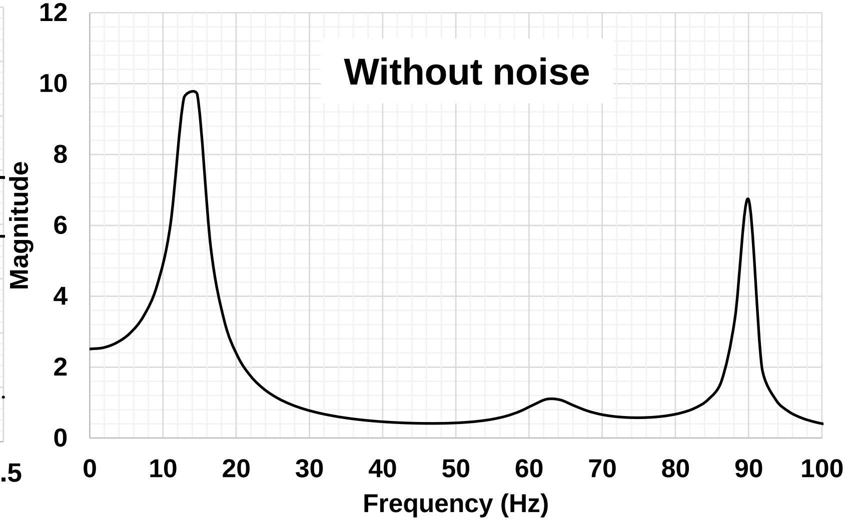
Roger's Approximation

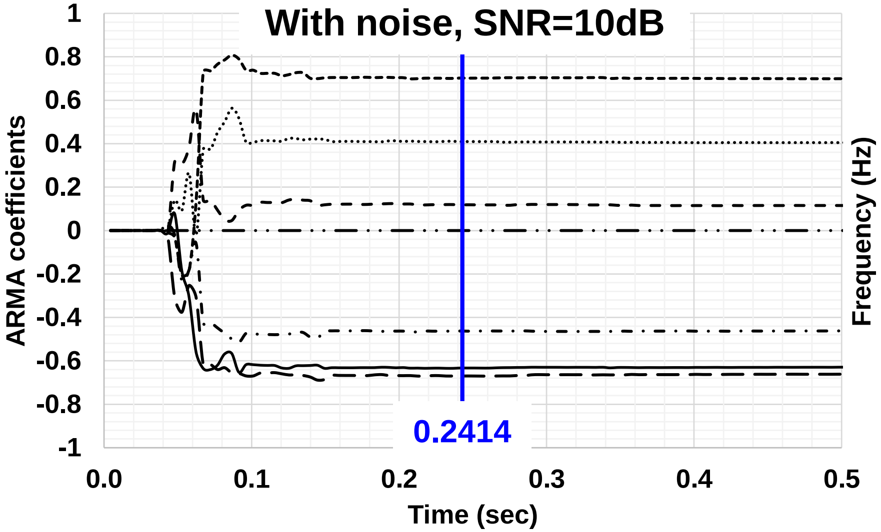


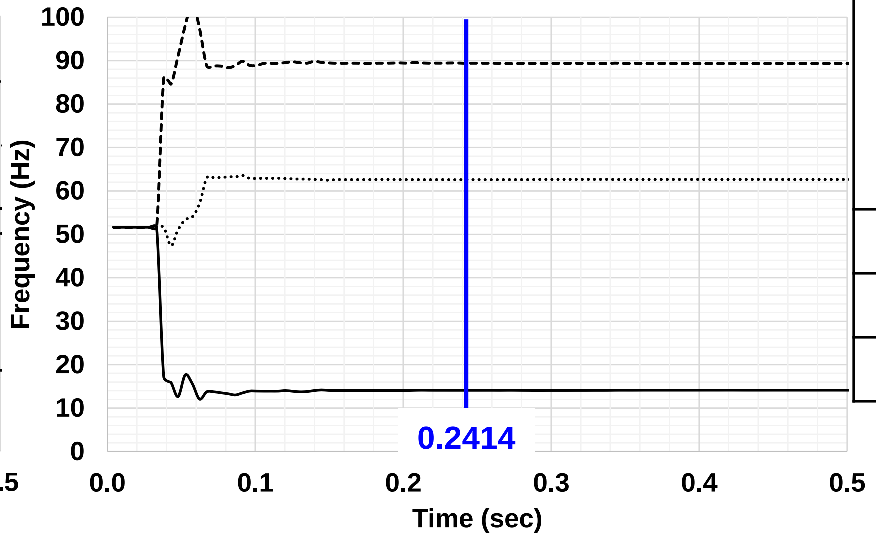
On-line parameter estimation with and without noise











First three damped aeroelastic frequencies

Mode	Without noise			With noise, SNR=10dB	
	FFT (Hz)	On-line parameter estimator			
		Damp. factor	Freq. (Hz)	Damp. factor	Freq. (Hz)
1	13.81	-10.02	13.88	-11.11	14.12
2	63.15	-25.46	62.73	-25.92	62.62
3	89.82	-4.187	89.44	-5.740	88.74

- $SNR \equiv 20 \times log_{10} \frac{\epsilon_{max}}{n_{rms}}$
  - ϵ_{max} maximum unsteady strain after 0.1 second.
  - n_{rms} root-mean-squared level of noise.

On-line parameter estimator is applied to the unsteady strain data