MECHANICAL VIBRATIONS

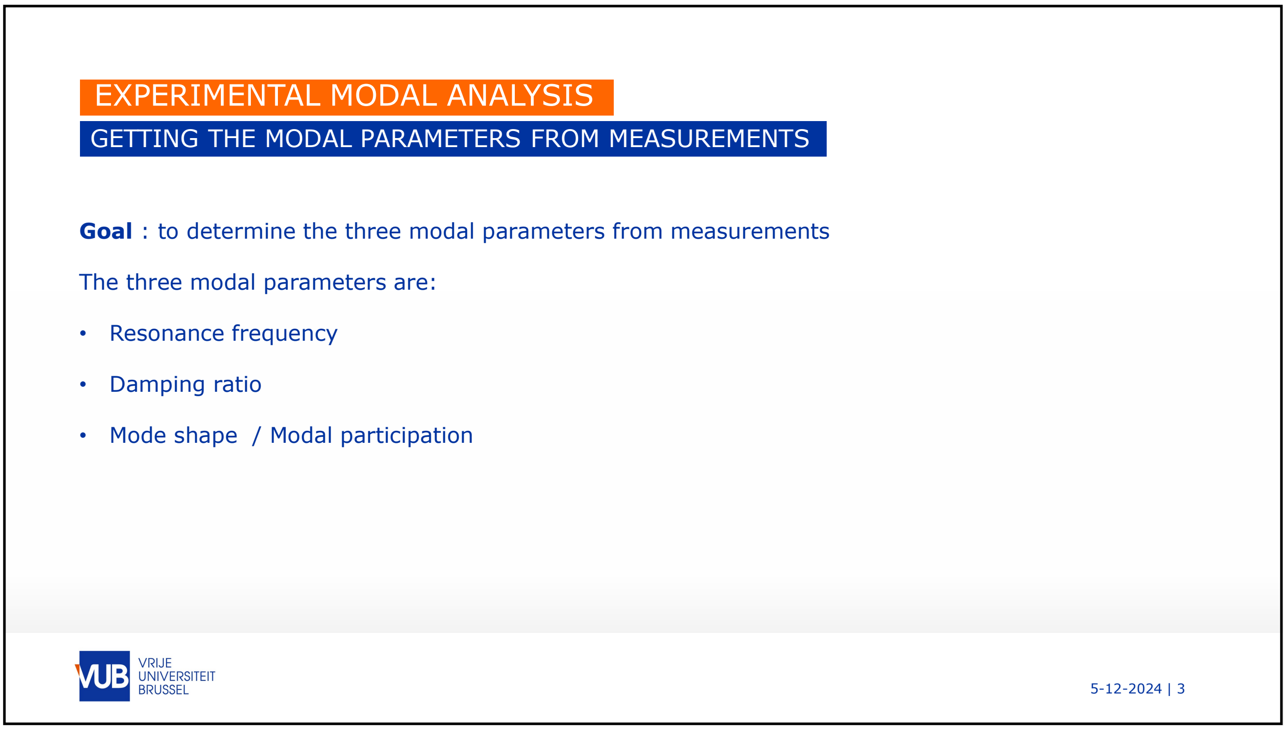
PARAMETRIC MODAL PARAMETER ESTIMATION

EXPERIMENTAL MODAL ANALYSIS

GETTING THE MODAL PARAMETERS FROM MEASUREMENTS

Goal : to determine the three modal parameters from measurements

The three modal parameters are:



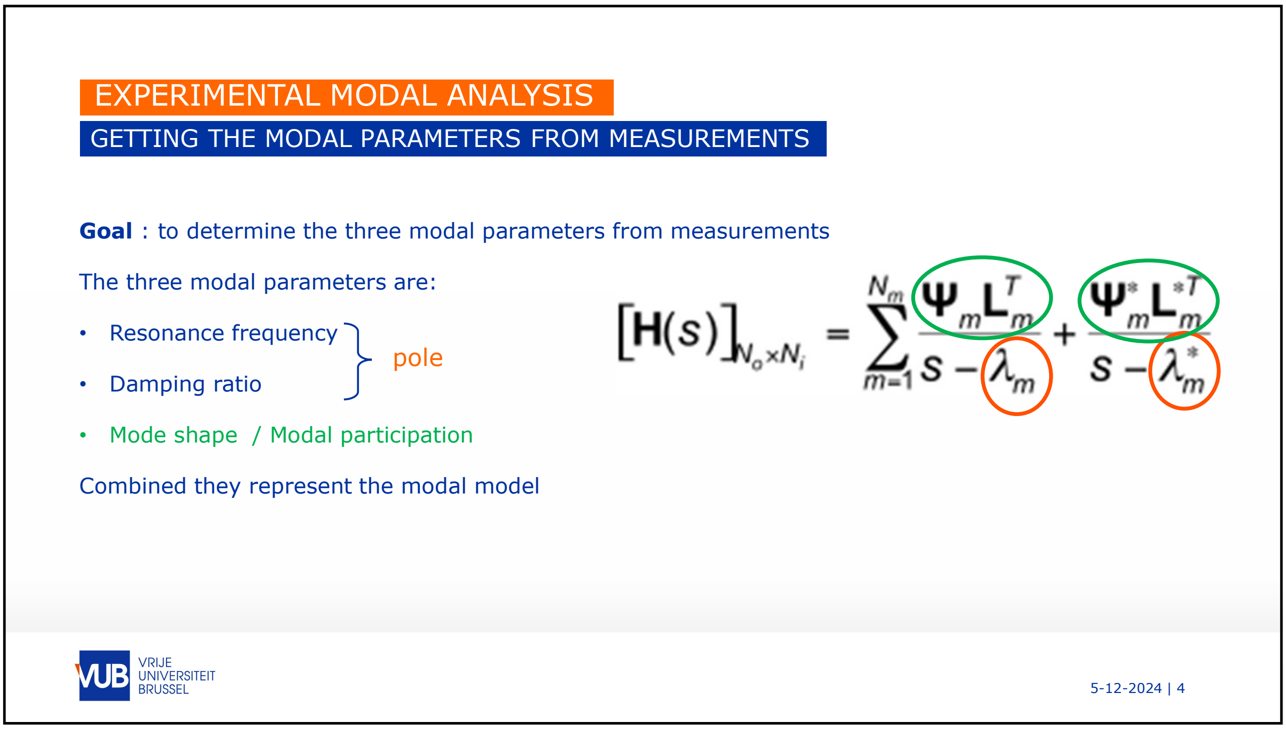
EXPERIMENTAL MODAL ANALYSIS

GETTING THE MODAL PARAMETERS FROM MEASUREMENTS

Goal : to determine the three modal parameters from measurements

The three modal parameters are:

- Resonance frequency
- Damping ratio
- Mode shape / Modal participation



EXPERIMENTAL MODAL ANALYSIS

GETTING THE MODAL PARAMETERS FROM MEASUREMENTS

Goal : to determine the three modal parameters from measurements

The three modal parameters are:

- Resonance frequency
- Damping ratio

pole

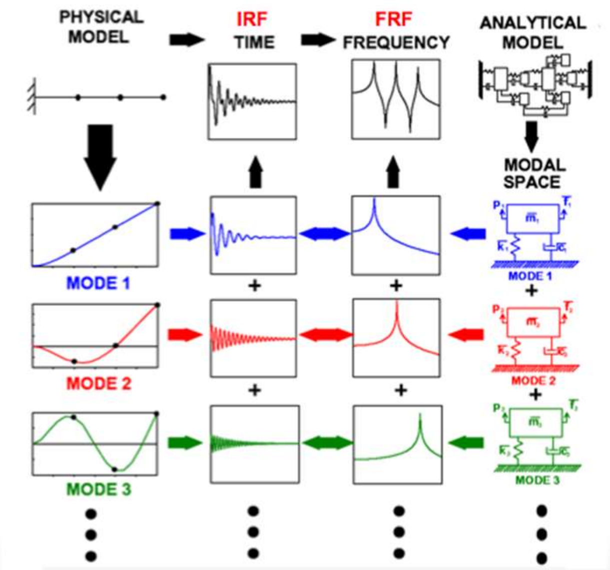
- Mode shape / Modal participation

$$[\mathbf{H}(s)]_{N_o \times N_i} = \sum_{m=1}^{N_m} \frac{\boxed{\mathbf{\Psi}_m \mathbf{L}_m^T}}{s - \boxed{\lambda_m}} + \frac{\boxed{\mathbf{\Psi}_m^* \mathbf{L}_m^{*T}}}{s - \boxed{\lambda_m^*}}$$

Combined they represent the modal model

EXPERIMENTAL MODAL ANALYSIS

THE MODAL MODEL



In words:

All dynamics of a MDOF system can be explained as the summed response of SDOF systems

$$[\mathbf{H}(s)]_{N_o \times N_i} = \sum_{m=1}^{N_m} \frac{\mathbf{\Psi}_m \mathbf{L}_m^T}{s-\lambda_m} + \frac{\mathbf{\Psi}_m^* \mathbf{L}_m^{*T}}{s-\lambda_m^*}$$

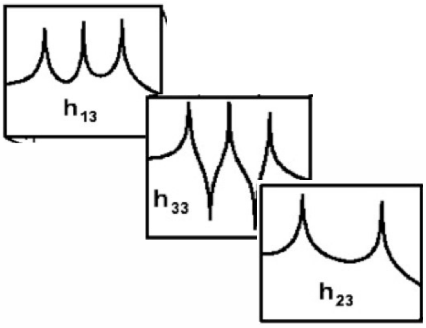
EXPERIMENTAL MODAL ANALYSIS

GETTING THE MODAL PARAMETERS FROM MEASUREMENTS

From our experimental work and **non-parametric** processing we obtained estimates of the inputs, outputs and ultimately the **transfer function** $H(\omega)$.

But what we have at this stage is just data, a complex-valued $N_o \times N_i$ matrix for the transfer function at every value of ω.

To translate these results into the modal model we'll do **parametric modal parameter estimation** (mpe) to estimate the values of the modal parameters.



$$[\mathbf{H}(s)]_{N_o \times N_i} = \sum_{m=1}^{N_m} \frac{\mathbf{\Psi}_m \mathbf{L}_m^T}{s-\lambda_m} + \frac{\mathbf{\Psi}_m^* \mathbf{L}_m^{*T}}{s-\lambda_m^*}$$

EXPERIMENTAL MODAL ANALYSIS

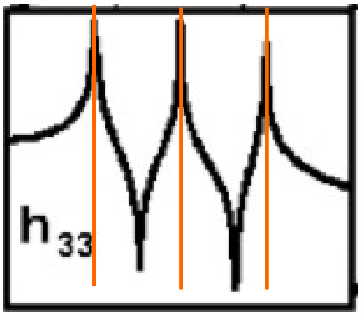
GETTING THE MODAL PARAMETERS FROM MEASUREMENTS

Peak picking

Selecting the highest peaks in the FRF and refer to them as resonance frequencies.

Limitations

- Results are limited to the frequency resolution
- Typically only a single FRF at a time (issue with nodal points)
- Does not provide any information on damping



**OK for a quick and dirty preliminary assessment…
but still used too much as the final result**

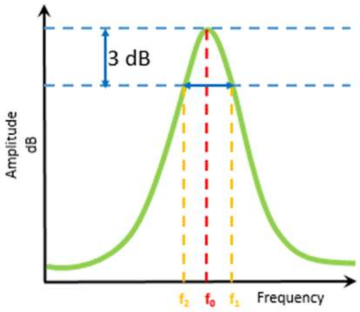
EXPERIMENTAL MODAL ANALYSIS

GETTING THE MODAL PARAMETERS FROM MEASUREMENTS

Peak picking (with damping)

Calculate the damping from the FRF using the 3dB method

Unreliable at best, do not use



$$Q = \frac{f_0}{f_2 - f_1}$$

EXPERIMENTAL MODAL ANALYSIS

Getting the modal parameters from measurements

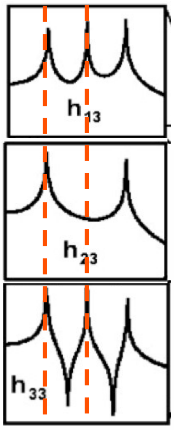
Peak picking

Selecting the peaks across the different rows/columns of our experiment allows to the Operational Deflection shapes.

But this ignores the orthogonality!

$$\psi_1 = \begin{Bmatrix}1\\1\\1\end{Bmatrix}$$

$$\psi_2 = \begin{Bmatrix}1\\0\\1\end{Bmatrix}$$



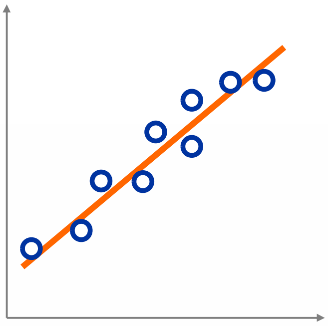
Note;
For simplicity I'm ignoring phase information here

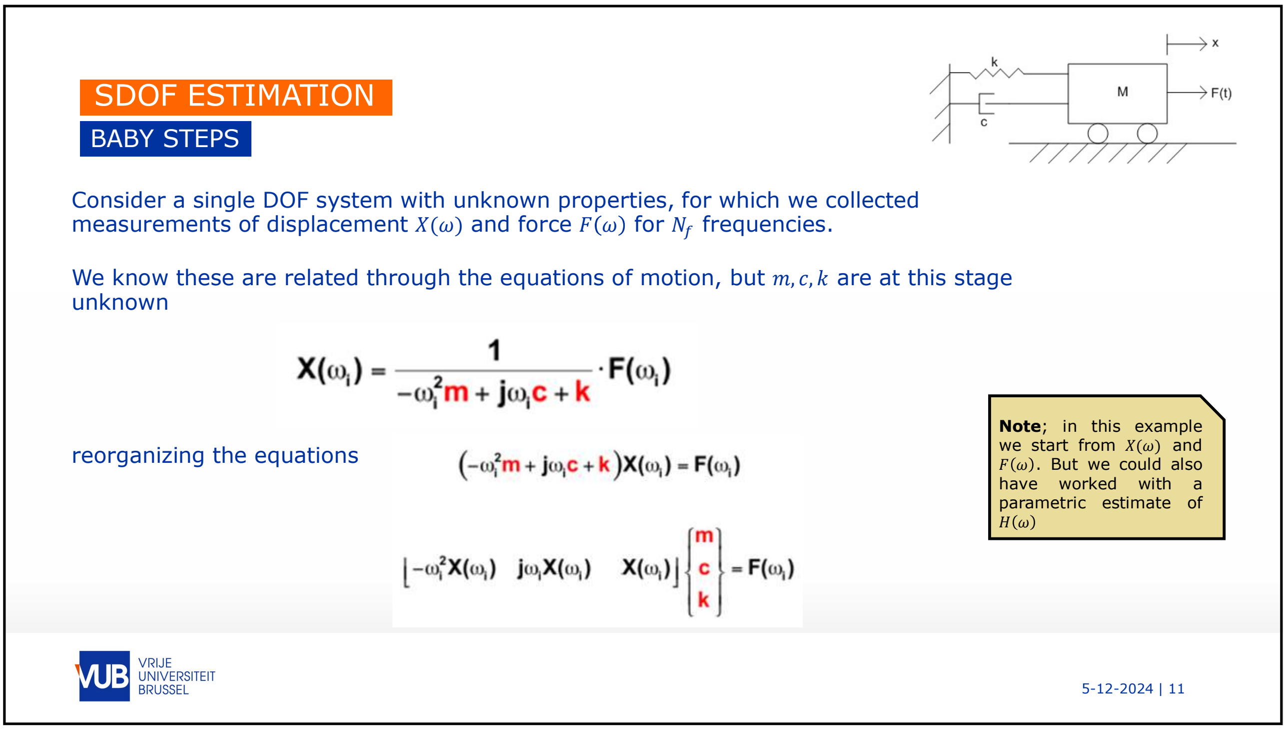
EXPERIMENTAL MODAL ANALYSIS

PREFERRED STRATEGY

The problem posed is similar to **curve-fitting** :

- We have collected data; samples; over the entire spectrum.
  - These samples will always be noisy to some degree (even after averaging)
- We have a parametric model that (in theory) should match the 'noise-free' data
  - E.g. in a simple linear regression $y = \boldsymbol{a}x + \boldsymbol{b}$
- We need to solve for the parameters; (a, b); that best fit the data





SDOF ESTIMATION

BABY STEPS

Consider a single DOF system with unknown properties, for which we collected measurements of displacement $X(\omega)$ and force $F(\omega)$ for N_f frequencies.

We know these are related through the equations of motion, but m, c, k are at this stage unknown

$$X(\omega_i) = \frac{1}{-\omega_i^2 m + j\omega_i c + k} \cdot F(\omega_i)$$

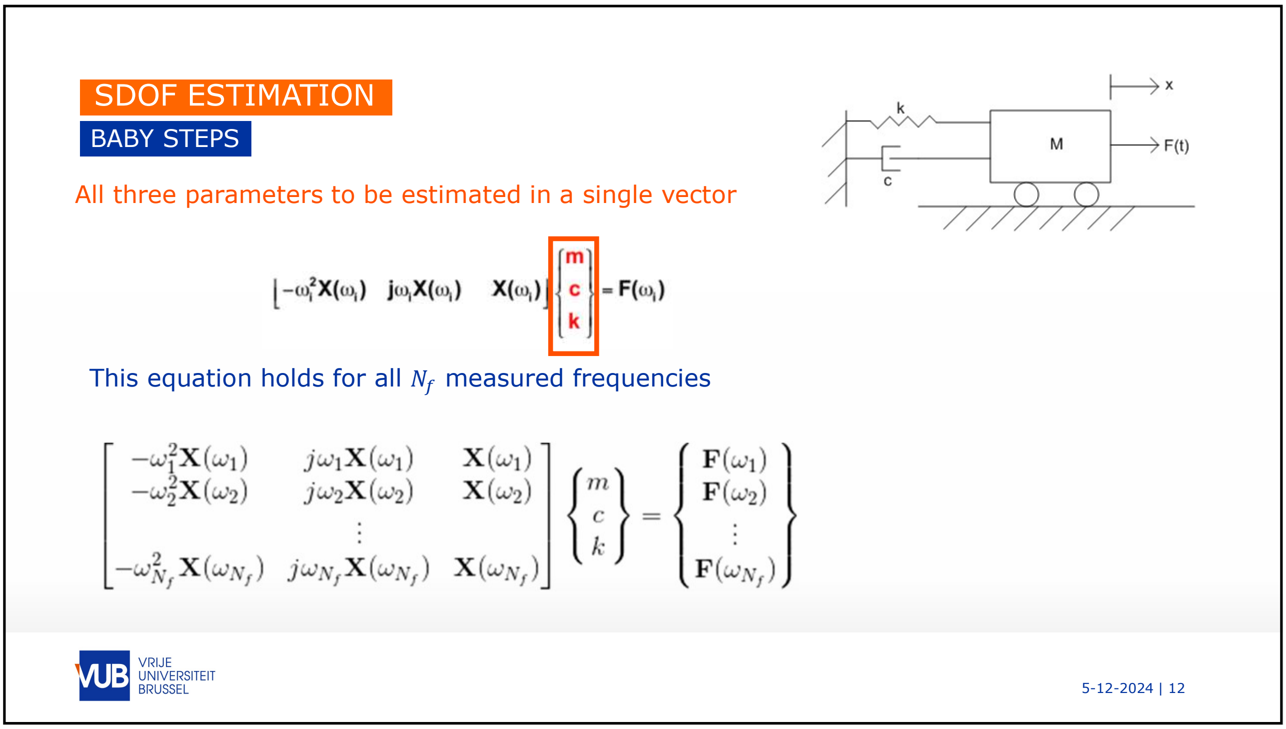
reorganizing the equations

$$\left(-\omega_i^2 m + j\omega_i c + k\right) X(\omega_i) = F(\omega_i)$$

$$\lfloor -\omega_i^2 X(\omega_i) \quad j\omega_i X(\omega_i) \quad X(\omega_i) \rfloor \begin{Bmatrix} m \\ c \\ k \end{Bmatrix} = F(\omega_i)$$

Note; in this example we start from $X(\omega)$ and $F(\omega)$. But we could also have worked with a parametric estimate of $H(\omega)$

VRIJE
UNIVERSITEIT
BRUSSEL



SDOF ESTIMATION

BABY STEPS

All three parameters to be estimated in a single vector

$$\lfloor -\omega_i^2 X(\omega_i) \quad j\omega_i X(\omega_i) \quad X(\omega_i) \rfloor \boxed{\begin{Bmatrix} m \\ c \\ k \end{Bmatrix}} = F(\omega_i)$$

This equation holds for all N_f measured frequencies

$$\begin{bmatrix} -\omega_1^2 \mathbf{X}(\omega_1) & j\omega_1 \mathbf{X}(\omega_1) & \mathbf{X}(\omega_1) \\ -\omega_2^2 \mathbf{X}(\omega_2) & j\omega_2 \mathbf{X}(\omega_2) & \mathbf{X}(\omega_2) \\ & \vdots & \\ -\omega_{N_f}^2 \mathbf{X}(\omega_{N_f}) & j\omega_{N_f} \mathbf{X}(\omega_{N_f}) & \mathbf{X}(\omega_{N_f}) \end{bmatrix} \begin{Bmatrix} m \\ c \\ k \end{Bmatrix} = \begin{Bmatrix} \mathbf{F}(\omega_1) \\ \mathbf{F}(\omega_2) \\ \vdots \\ \mathbf{F}(\omega_{N_f}) \end{Bmatrix}$$

VRIJE
UNIVERSITEIT
BRUSSEL

SDOF ESTIMATION

BABY STEPS

$$\begin{bmatrix} -\omega_1^2\mathbf{X}(\omega_1) & j\omega_1\mathbf{X}(\omega_1) & \mathbf{X}(\omega_1) \\ -\omega_2^2\mathbf{X}(\omega_2) & j\omega_2\mathbf{X}(\omega_2) & \mathbf{X}(\omega_2) \\ & \vdots & \\ -\omega_{N_f}^2\mathbf{X}(\omega_{N_f}) & j\omega_{N_f}\mathbf{X}(\omega_{N_f}) & \mathbf{X}(\omega_{N_f}) \end{bmatrix} \begin{Bmatrix} m \\ c \\ k \end{Bmatrix} = \begin{Bmatrix} \mathbf{F}(\omega_1) \\ \mathbf{F}(\omega_2) \\ \vdots \\ \mathbf{F}(\omega_{N_f}) \end{Bmatrix}$$

Overdetermined problem

- 3 unknown parameters ***m, c, k*** <<< N_f equations

Solve for the unknowns θ as to **minimize** the least squares error function LS(θ)

$$K_{(N_f\times3)}\theta_{(3\times1)} = \mathbf{F}_{(N_f\times1)}$$

$$\mathbf{e}_{(N_f\times1)} = K_{(N_f\times3)}\theta_{(3\times1)} - \mathbf{F}_{(N_f\times1)}$$

$$LS(\theta) = \mathbf{e}^T\mathbf{e}$$

$$\frac{\delta LS}{\delta\theta} = 0$$

SDOF ESTIMATION

BABY STEPS

$$\mathbf{e}_{(N_f\times1)} = K_{(N_f\times3)}\theta_{(3\times1)} - \mathbf{F}_{(N_f\times1)}$$

$$LS(\theta) = \mathbf{e}^T\mathbf{e}$$

$$\frac{\delta LS}{\delta\theta} = 0$$

$$2\frac{\delta\mathbf{e}^T}{\delta\theta}\mathbf{e} = 0$$

$$2K^T\cdot(K\theta_{\mathbf{LS}} - \mathbf{F}) = 0$$

$$\boxed{\begin{aligned}\theta_{\mathbf{LS}} &= \left(K^TK\right)^{-1}K^T\mathbf{F} \\ &= K \setminus \mathbf{F}\end{aligned}}$$

The values of θ found by solving the bottom equation represent the least squares estimate of the problem posed,

They are the set of parameters that best fit with the data contained in the measurements

We found **m,c,k**

SDOF ESTIMATION

BABY STEPS

We found **m,c,k** how to get the modal parameters?

$$\mathbf{X}(\omega_i) = \frac{\mathbf{1}}{-\omega_i^2\mathbf{m} + \mathbf{j}\omega_i\mathbf{c} + \mathbf{k}} \cdot \mathbf{F}(\omega_i)$$

The system poles (λ) are the roots of the denominator, so just use the found *m,c,k* values to calculate the roots.

E.g. in MATLAB and Python using the `roots` command

From the poles the resonance frequency and damping ratios are calculated

$$f_{res} = \frac{abs(\lambda)}{2\pi}$$

$$\xi = \frac{-real(\lambda)}{\|\lambda\|}$$

SDOF ESTIMATION

ADDING A FINAL CONSTRAINT

In fact, the poles of the previous approach will be two values of λ that solve the equation

$$m\lambda^2 + c\lambda + k = 0$$

There is no guarantee that these two λ are eachothers complex conjugate, while this is to be expected from the modal model!

An additional constraint is necessary to force this on the least squares solution.

$$\begin{bmatrix} real(K) \\ imag(K) \end{bmatrix} \theta = \begin{Bmatrix} real(\mathbf{F}) \\ imag(\mathbf{F}) \end{Bmatrix}$$

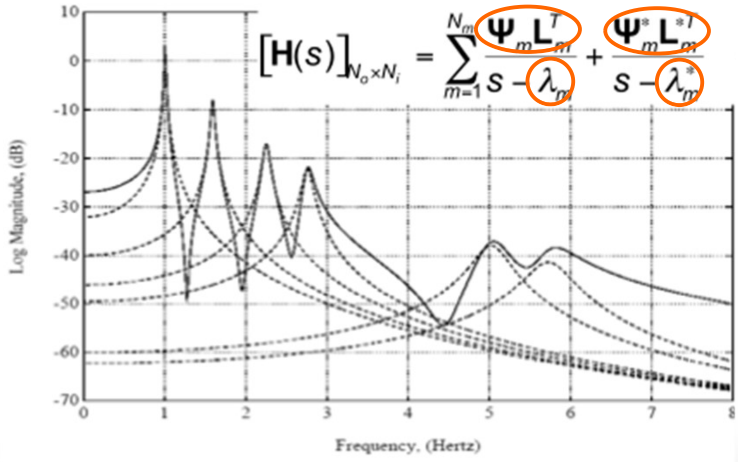
EXPERIMENTAL MODAL ANALYSIS

GOING TO MDOF

But how do expand this strategy to an MDOF system. Knowing $H(\omega)$ is a matrix and we need to determine the modal parameters, **including the mode shapes.**

But the modal model is non linear in the (modal) parameters

To directly fit the modal model to the FRF would imply a (recursive) non-linear optimisation



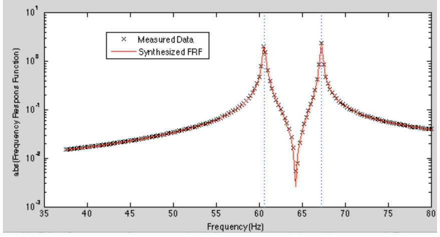
$$[H(s)]_{N_o \times N_i} = \sum_{m=1}^{N_m} \frac{\Psi_m L_m^T}{s-\lambda_m} + \frac{\Psi_m^* L_m^{*T}}{s-\lambda_m^*}$$

EXTENDING TO MULTIPLE MODES

ADDING ZEROS

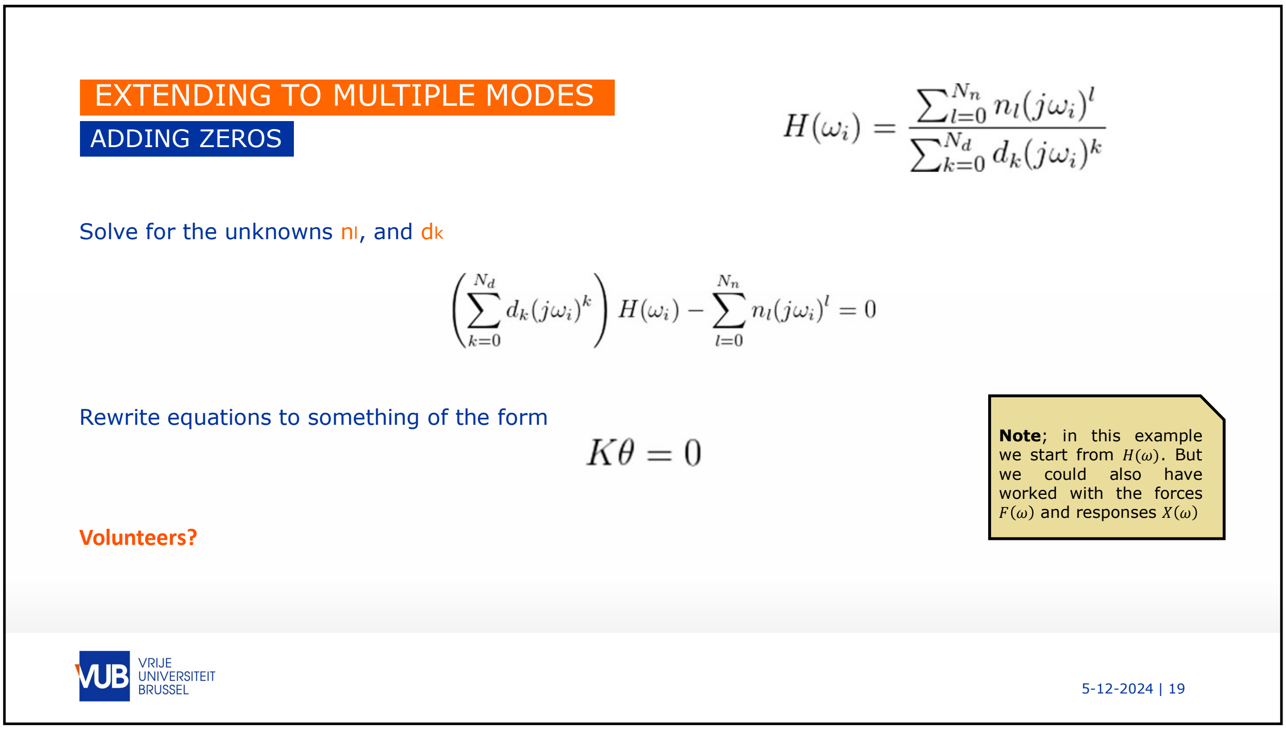
Two things change when more modes are considered:

- The model order has to be increased to allow for a larger number of modes
- Zeros need to be considered



Previous single degree of freedom model is expanded to a MDOF nominator-denominator model in which the model orders (N_n, N_d) exceed 2

$$H(\omega_i) = \frac{\sum_{l=0}^{N_n} n_l (j\omega_i)^l}{\sum_{k=0}^{N_d} d_k (j\omega_i)^k}$$



EXTENDING TO MULTIPLE MODES

ADDING ZEROS

$$H(\omega_i) = \frac{\sum_{l=0}^{N_n} n_l (j\omega_i)^l}{\sum_{k=0}^{N_d} d_k (j\omega_i)^k}$$

Solve for the unknowns nl, and dk

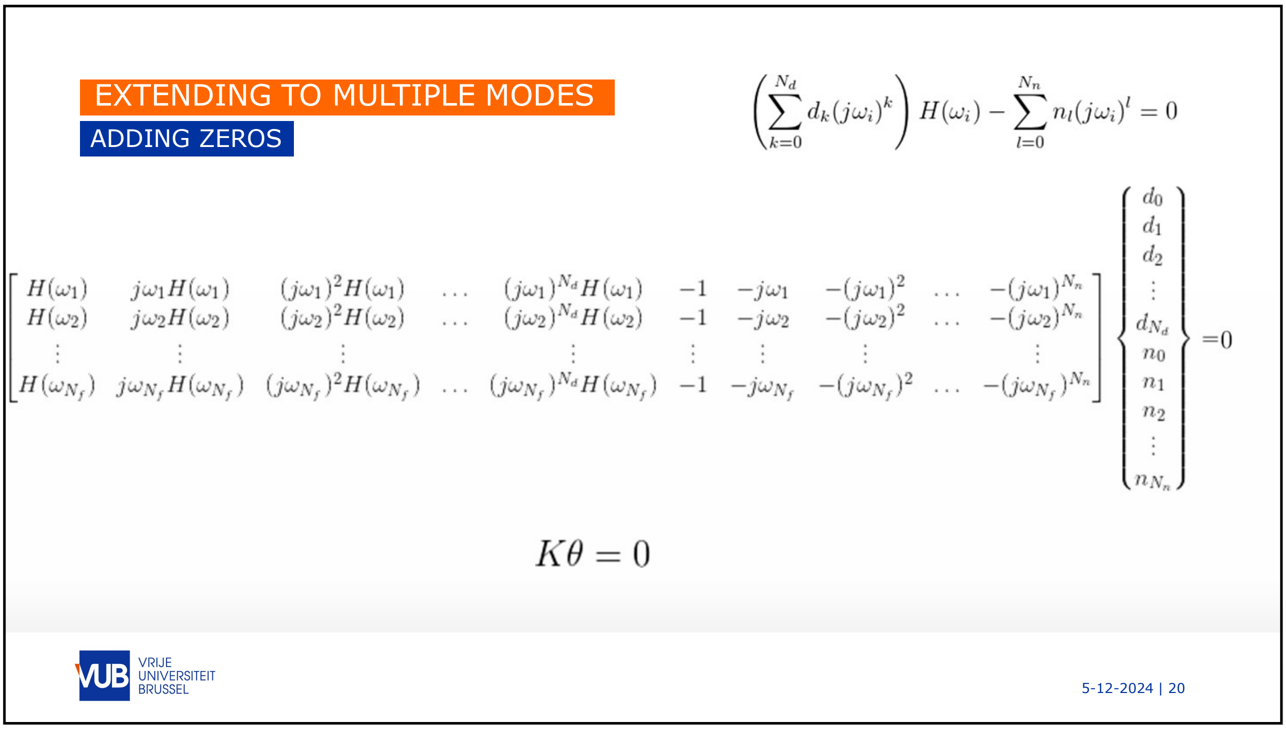
$$\left(\sum_{k=0}^{N_d} d_k (j\omega_i)^k \right) H(\omega_i) - \sum_{l=0}^{N_n} n_l (j\omega_i)^l = 0$$

Rewrite equations to something of the form

$$K\theta = 0$$

Note; in this example we start from $H(\omega)$. But we could also have worked with the forces $F(\omega)$ and responses $X(\omega)$

Volunteers?



EXTENDING TO MULTIPLE MODES

ADDING ZEROS

$$\left(\sum_{k=0}^{N_d} d_k (j\omega_i)^k \right) H(\omega_i) - \sum_{l=0}^{N_n} n_l (j\omega_i)^l = 0$$

$$\begin{bmatrix} H(\omega_1) & j\omega_1 H(\omega_1) & (j\omega_1)^2 H(\omega_1) & \dots & (j\omega_1)^{N_d} H(\omega_1) & -1 & -j\omega_1 & -(j\omega_1)^2 & \dots & -(j\omega_1)^{N_n} \\ H(\omega_2) & j\omega_2 H(\omega_2) & (j\omega_2)^2 H(\omega_2) & \dots & (j\omega_2)^{N_d} H(\omega_2) & -1 & -j\omega_2 & -(j\omega_2)^2 & \dots & -(j\omega_2)^{N_n} \\ \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & \vdots & & \vdots \\ H(\omega_{N_f}) & j\omega_{N_f} H(\omega_{N_f}) & (j\omega_{N_f})^2 H(\omega_{N_f}) & \dots & (j\omega_{N_f})^{N_d} H(\omega_{N_f}) & -1 & -j\omega_{N_f} & -(j\omega_{N_f})^2 & \dots & -(j\omega_{N_f})^{N_n} \end{bmatrix} \left\{ \begin{matrix} d_0 \\ d_1 \\ d_2 \\ \vdots \\ d_{N_d} \\ n_0 \\ n_1 \\ n_2 \\ \vdots \\ n_{N_n} \end{matrix} \right\} = 0$$

$$K\theta = 0$$

EXTENDING TO MULTIPLE MODES

APPLYING CONSTRAINT

Next it remains to solve $K\theta = 0$

But, an infinite number of solutions exist (i.e. any multiple of θ) !

To solve for θ the number of solutions has to be constrained to allow only a single least squares solution.

Possible constraints are :

- $d_{nd} = 1$: Most common assumption
- $|\theta| = 1$: Total least squares solution

EXTENDING TO MULTIPLE MODES

APPLYING CONSTRAINT

$$\begin{bmatrix} H(\omega_1) & j\omega_1 H(\omega_1) & (j\omega_1)^2 H(\omega_1) & \dots & (j\omega_1)^{N_d} H(\omega_1) & -1 & -j\omega_1 & -(j\omega_1)^2 & \dots & -(j\omega_1)^{N_n} \\ H(\omega_2) & j\omega_2 H(\omega_2) & (j\omega_2)^2 H(\omega_2) & \dots & (j\omega_2)^{N_d} H(\omega_2) & -1 & -j\omega_2 & -(j\omega_2)^2 & \dots & -(j\omega_2)^{N_n} \\ \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & \vdots & & \vdots \\ H(\omega_{N_f}) & j\omega_{N_f} H(\omega_{N_f}) & (j\omega_{N_f})^2 H(\omega_{N_f}) & \dots & (j\omega_{N_f})^{N_d} H(\omega_{N_f}) & -1 & -j\omega_{N_f} & -(j\omega_{N_f})^2 & \dots & -(j\omega_{N_f})^{N_n} \end{bmatrix} \begin{Bmatrix} d_0 \\ d_1 \\ d_2 \\ \vdots \\ \mathbf{1} \\ n_0 \\ n_1 \\ n_2 \\ \vdots \\ n_{N_n} \end{Bmatrix} = 0$$

$$\begin{bmatrix} \begin{matrix} H(\omega_1) & j\omega_1 H(\omega_1) & (j\omega_1)^2 H(\omega_1) & \dots \\ H(\omega_2) & j\omega_2 H(\omega_2) & (j\omega_2)^2 H(\omega_2) & \dots \\ \vdots & \vdots & \vdots & \\ H(\omega_{N_f}) & j\omega_{N_f} H(\omega_{N_f}) & (j\omega_{N_f})^2 H(\omega_{N_f}) & \dots \end{matrix} & \begin{matrix} -1 & -j\omega_1 & -(j\omega_1)^2 & \dots & -(j\omega_1)^{N_n} \\ -1 & -j\omega_2 & -(j\omega_2)^2 & \dots & -(j\omega_2)^{N_n} \\ \vdots & \vdots & \vdots & & \vdots \\ -1 & -j\omega_{N_f} & -(j\omega_{N_f})^2 & \dots & -(j\omega_{N_f})^{N_n} \end{matrix} \end{bmatrix} \begin{Bmatrix} d_o \\ d_1 \\ d_2 \\ \vdots \\ n_0 \\ n_1 \\ n_2 \\ \vdots \\ n_{n_n} \end{Bmatrix} = - \begin{Bmatrix} (j\omega_1)^{n_d} H(\omega_1) \\ (j\omega_2)^{n_d} H(\omega_2) \\ \vdots \\ (j\omega_f)^{n_d} H(\omega_f) \end{Bmatrix}$$

And we can solve again using $\theta = K\backslash Y$

EXTENDING TO MULTIPLE MODES

SOLVING FOR THE MODAL PARAMETERS

Again the poles are found as the roots of the denominator polynomial :

$$\sum_{k=0}^{N_d} d_k (j\omega_i)^k \quad = 0$$

This can be done e.g. using the `roots` command in MATLAB or numpy

---

EXTENDING TO MULTIPLE MODES

MODEL ORDERS

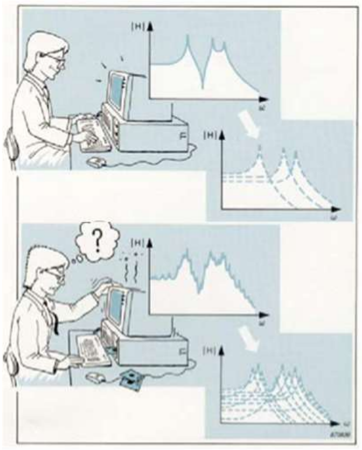
The model orders (N_n and N_d) are settings by the user.
There is really no upper bound,…

With increasing model orders the model fit will get better and the LS error will reduce….

But a lot of numerical modes (without physical meaning) will appear as the model cannot distinguish between physical modes and noise.

Always keep in mind that the curve fitter only aims to minimize the LS cost function, not to improve the "physicality" of the result.

There is thus a tradeof for higher model orders.



STABILIZATION DIAGRAM

A TOOL TO DETERMINE STRUCTURAL MODES

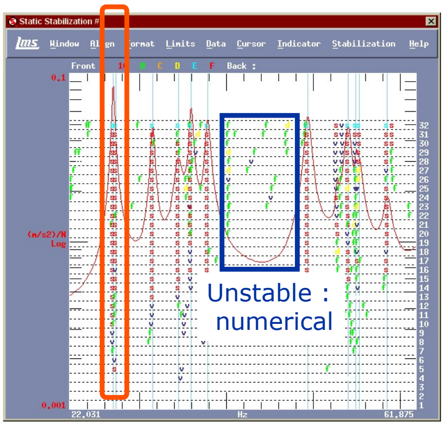
Typically a so called stabilization chart is used to distinguish between structural and numerical modes.

Basic assumption :

Structural models will appear in models of different model order while numerical modes will vary for different model orders.

Thus by calculating the poles for different model orders , structural poles will appear in all model orders. They are "stable" (s).

Rule of thumb : choose the lowest model order where all modes of interest are stable



Stable (s)

Unstable : numerical

Different model orders

Frequency

NUMERICAL UNBALANCE

USING ORTHOGONAL BASE FUNCTIONS

One of the common issues when implementing the frequency domain estimators is a numerical unbalance at high model orders.

$$\begin{matrix} -1 & -j\omega_1 & -(j\omega_1)^2 & \dots & -(j\omega_1)^{N_n} \\ -1 & -j\omega_2 & -(j\omega_2)^2 & \dots & -(j\omega_2)^{N_n} \\ \vdots & \vdots & \vdots & & \vdots \\ -1 & -j\omega_{N_f} & -(j\omega_{N_f})^2 & \dots & -(j\omega_{N_f})^{N_n} \end{matrix}$$

For high model orders the values in the first column and the last column differ in several orders of magnitude

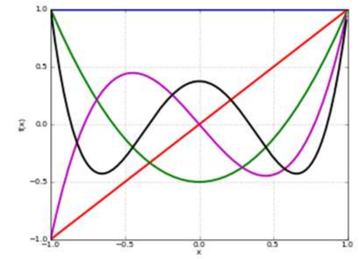
e.g. for model order 64, and the row with ω=10

- First column : -1
- Last column : 10^64

This unbalance leeds to numerical issues and ultimately erroneous results (e.g. a poor fit at low frequencies).

NUMERICAL UNBALANCE

USING ORTHOGONAL BASE FUNCTIONS



To avoid these issues a first step is to normalize the frequency band (e.g. map all frequencies from 0 to 1).

But even better is the use of orthogonal base functions, which improve the conditioning of the problem altogether :

$$H(\omega_i) = \frac{\sum_{l=0}^{N_n} n_l \Omega_l(\omega_i)}{\sum_{k=0}^{N_d} d_k \Omega_k(\omega_i)}$$

$$\Omega_j(\omega_f) = \begin{cases} (i\omega_f)^j \\ \exp(-i\omega_f \cdot j) \\ \text{Forsythe polynomials, ...} \end{cases}$$

recommended

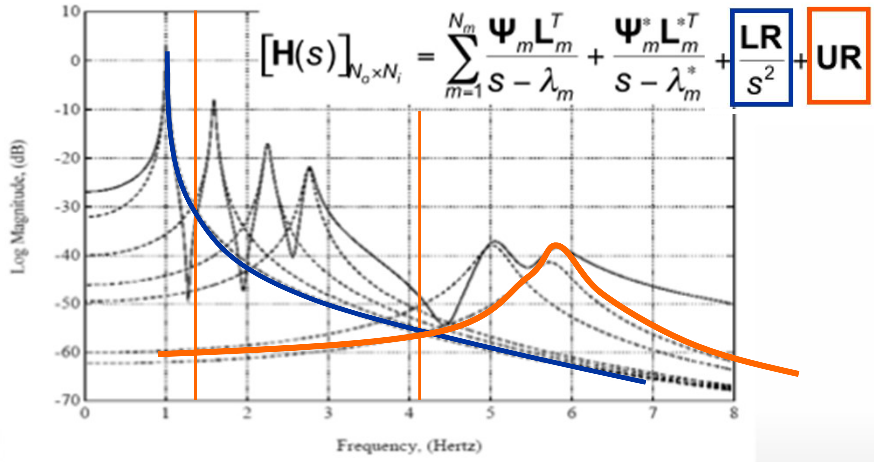
Note that if another polynomial is used, the found roots of the denominator have to be mapped back the frequency domain

VRIJE UNIVERSITEIT BRUSSEL

UPPER AND LOWER RESIDUALS

AN ADDITION WHEN PROCESSING A LIMITED BANDWIDTH

In general only a limited bandwidth is curve fitted. This implies some modes are out of this frequency band. Nonetheless these modes have an influence in the band of interest.



$$[H(s)]_{N_o \times N_i} = \sum_{m=1}^{N_m} \frac{\Psi_m L_m^T}{s-\lambda_m} + \frac{\Psi_m^* L_m^{*T}}{s-\lambda_m^*} + \boxed{\frac{LR}{s^2}} + \boxed{UR}$$

These residual terms can be added to the numerator nominator model as two additional unknowns

VRIJE UNIVERSITEIT BRUSSEL

FROM SDOF TO MDOF

PROCESSING THE WHOLE FRF

Up to now we were not able to determine the mode shapes. To do so the curve fitting algorithm has to be extended to process the entire FRF.

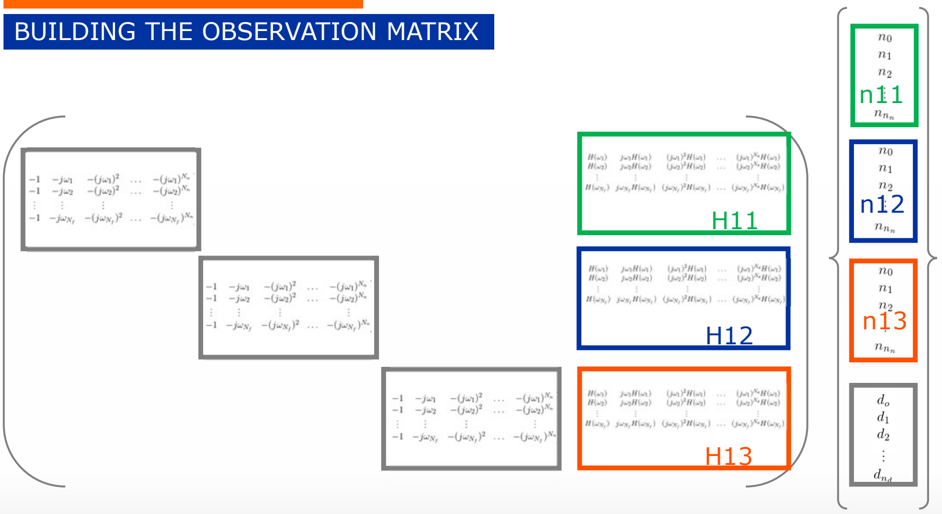
$$H_{11}(\omega) = \frac{\sum_{l=0}^{N_n} n_{11,l}\Omega_l(\omega)}{\sum_{k=0}^{N_d} d_k\Omega_k(\omega)} \quad H_{12}(\omega) = \frac{\sum_{l=0}^{N_n} n_{12,l}\Omega_l(\omega)}{\sum_{k=0}^{N_d} d_k\Omega_k(\omega)} \qquad H_{ij}(\omega) = \frac{\sum_{l=0}^{N_n} n_{ij,l}\Omega_l(\omega)}{\sum_{k=0}^{N_d} d_k\Omega_k(\omega)}$$

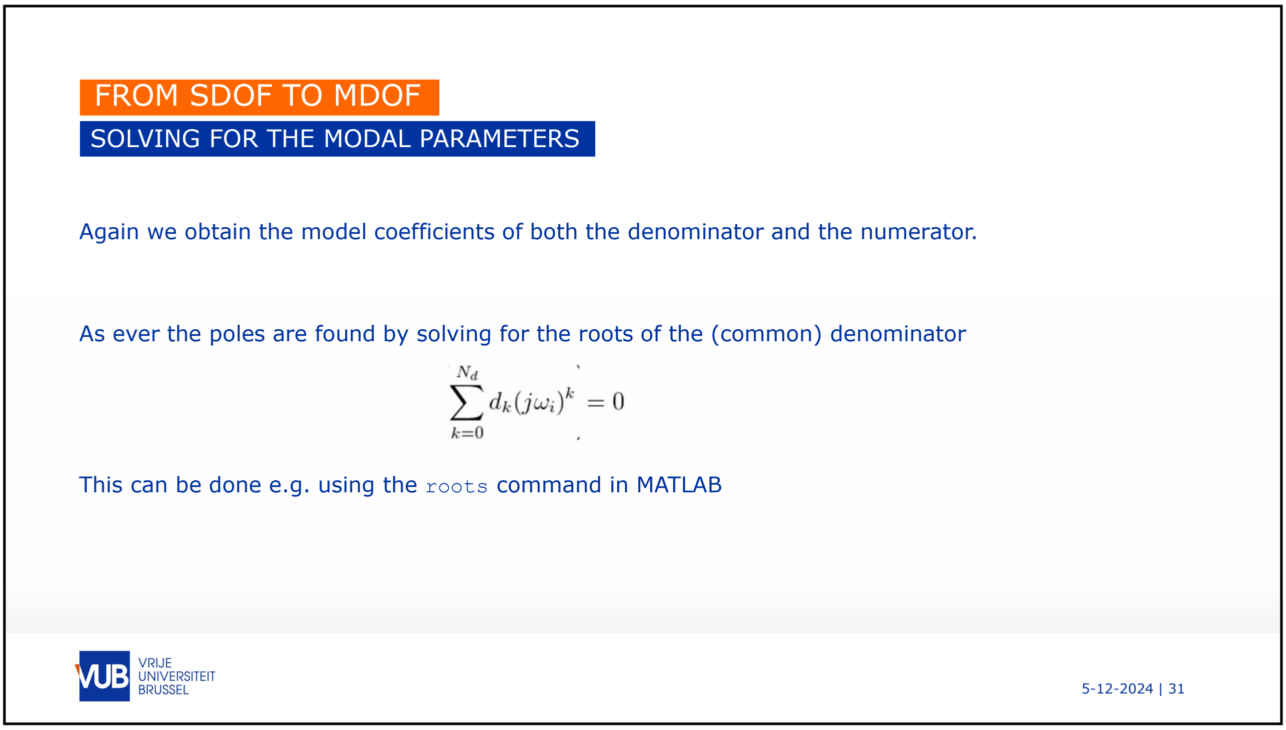
Consistent with the modal model, poles are global, the denominator is common to all equations.

-> **Common denominator model!**

FROM SDOF TO MDOF

BUILDING THE OBSERVATION MATRIX





FROM SDOF TO MDOF

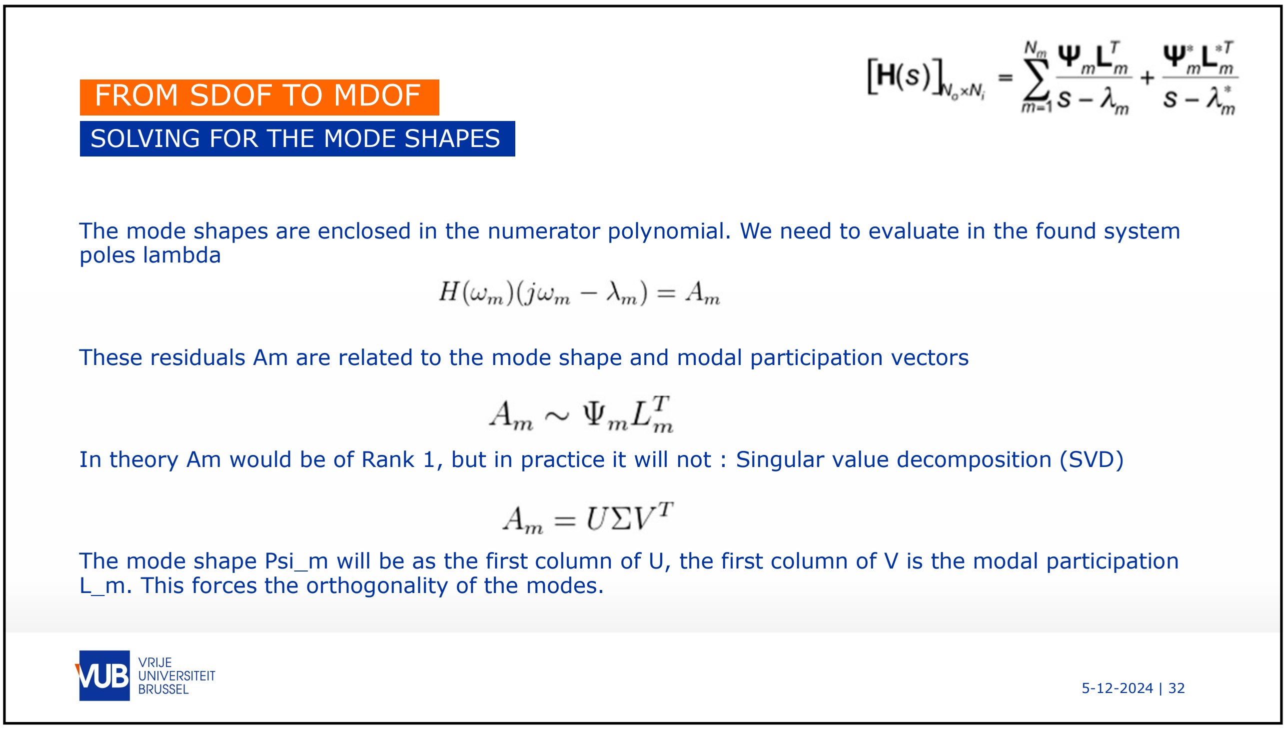
SOLVING FOR THE MODAL PARAMETERS

Again we obtain the model coefficients of both the denominator and the numerator.

As ever the poles are found by solving for the roots of the (common) denominator

$$\sum_{k=0}^{N_d} d_k (j\omega_i)^k = 0$$

This can be done e.g. using the `roots` command in MATLAB



$$[\mathbf{H}(s)]_{N_o \times N_i} = \sum_{m=1}^{N_m} \frac{\mathbf{\Psi}_m \mathbf{L}_m^T}{s - \lambda_m} + \frac{\mathbf{\Psi}_m^* \mathbf{L}_m^{*T}}{s - \lambda_m^*}$$

FROM SDOF TO MDOF

SOLVING FOR THE MODE SHAPES

The mode shapes are enclosed in the numerator polynomial. We need to evaluate in the found system poles lambda

$$H(\omega_m)(j\omega_m - \lambda_m) = A_m$$

These residuals Am are related to the mode shape and modal participation vectors

$$A_m \sim \Psi_m L_m^T$$

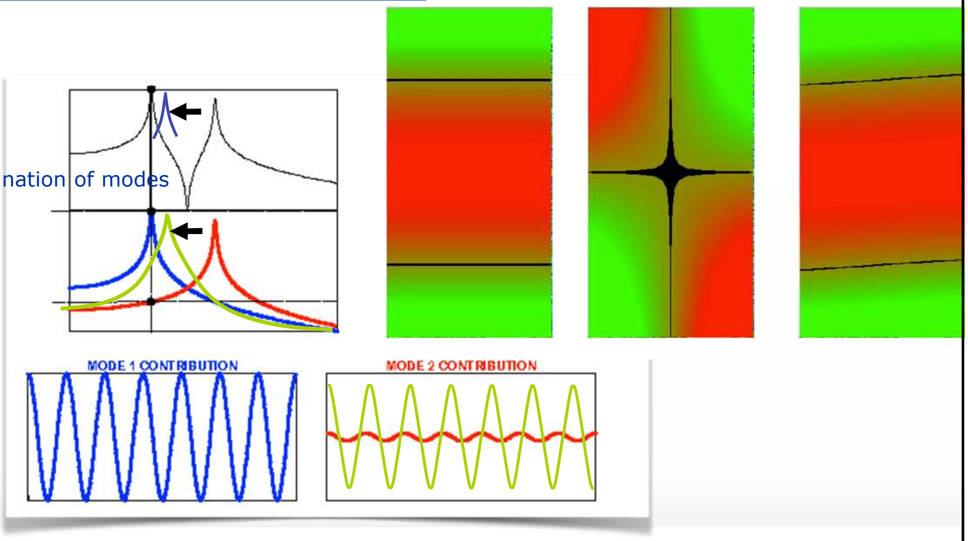
In theory Am would be of Rank 1, but in practice it will not : Singular value decomposition (SVD)

$$A_m = U \Sigma V^T$$

The mode shape Psi_m will be as the first column of U, the first column of V is the modal participation L_m. This forces the orthogonality of the modes.

FROM SDOF TO MDOF

The difference between ODS and modeshapes



Modes are orthogonal

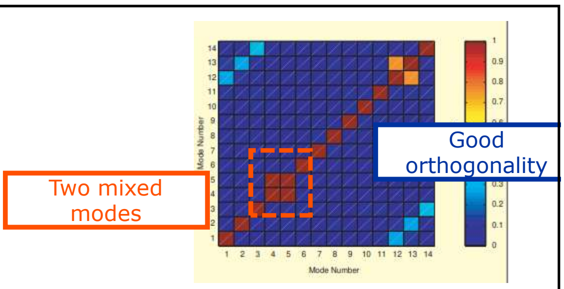
ODS are a linear combination of modes

$$ODS = \alpha\Psi_i + \beta\Psi_j$$

VRIJE UNIVERSITEIT BRUSSEL

MODAL ASSURANCE CRITERIUM

A WAY TO CHECK YOUR RESULTS



Check the orthogonality of the found modeshapes

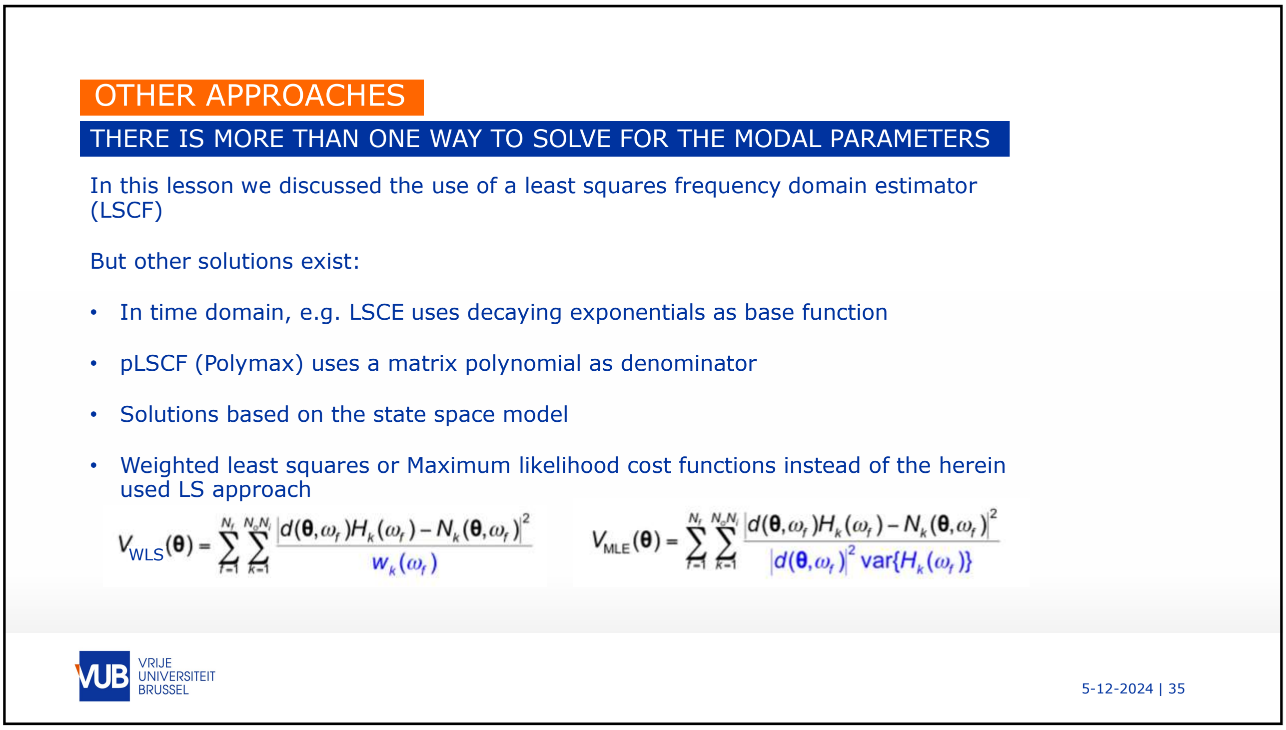
$$MAC(\Psi_i, \Psi_j) = \frac{\left\|\Psi_i^T \Psi_j^*\right\|^2}{(\Psi_i^T \Psi_i^*)(\Psi_j^T \Psi_j^*)}$$

MAC ranges from 1 to 0 with MAC(i,i) =1. And perfect orthogonality = 0

ODS result in MAC areas of higher MAC as ODS are the super positions of Mode shapes

$$MAC(ODS, \Psi_i) \sim \alpha \qquad MAC(ODS, \Psi_j) \sim \beta$$

VRIJE UNIVERSITEIT BRUSSEL



OTHER APPROACHES

THERE IS MORE THAN ONE WAY TO SOLVE FOR THE MODAL PARAMETERS

In this lesson we discussed the use of a least squares frequency domain estimator (LSCF)

But other solutions exist:

- In time domain, e.g. LSCE uses decaying exponentials as base function
- pLSCF (Polymax) uses a matrix polynomial as denominator
- Solutions based on the state space model
- Weighted least squares or Maximum likelihood cost functions instead of the herein used LS approach

$$V_{\mathrm{WLS}}(\boldsymbol{\theta}) = \sum_{f=1}^{N_f} \sum_{k=1}^{N_o N_i} \frac{\left| d(\boldsymbol{\theta},\omega_f) H_k(\omega_f) - N_k(\boldsymbol{\theta},\omega_f) \right|^2}{w_k(\omega_f)} \qquad V_{\mathrm{MLE}}(\boldsymbol{\theta}) = \sum_{f=1}^{N_f} \sum_{k=1}^{N_o N_i} \frac{\left| d(\boldsymbol{\theta},\omega_f) H_k(\omega_f) - N_k(\boldsymbol{\theta},\omega_f) \right|^2}{\left| d(\boldsymbol{\theta},\omega_f) \right|^2 \mathrm{var}\{H_k(\omega_f)\}}$$

VRIJE
UNIVERSITEIT
BRUSSEL