

- Français
- English

Accès libre

Vous désirez en savoir plus sur la gestion de vos données de recherche ? Ce [tutoriel](#) est pour vous ! Les vidéos d'information sont disponibles [ici](#).

Modèles numériques

Aubes de la NASA

Pour citer ce catalogue: article de référence

×

```
@proceedings{10.1115/GT2023-101107,  
  author = {Kojtych, Solène and Batailly, Alain},  
  title = "{Open NASA blade models for nonlinear dynamics simulations}",  
  volume = {Volume 8A: Structures and Dynamics – Aerodynamics Excitation  
and Damping; Bearing and Seal Dynamics; Emerging Methods in Engineering  
Design, Analysis, and  
Additive Manufacturing; Fatigue, Fracture, and Life Prediction},  
  series = {Turbo Expo: Power for Land, Sea, and Air},  
  year = {2023},  
  month = {06},  
  doi = {10.1115/GT2023-101107},  
  url = {https://doi.org/10.1115/GT2023-101107},  
}
```

fichiers en libre accès

×

```
@softwareversion{kojtych:hal-03945336v1,  
  title = {{A catalogue of open NASA blade models}},  
  author = {Kojtych, Solène and Batailly, Alain},  
  url = {https://hal.science/hal-03945336},  
  note = {},  
  year = {2023},  
  month = Jan,  
  swhid =  
{swh:1:dir:1ac35a9ed33ed49c0f3297f08bb5bb0a6578da5c;origin=https://gitlab  
.lava.polymtl.ca/depots\_publics/modeles/catalogue\_aubes;visit=swh:1:snp  
:6d1d388afc61917fa91efdb52e5cc17c4ea3d0a5;anchor=swh:1:rev:108595ab39d617}
```

```
f73483ed578be9a657e3637c95},  
  verison = {1.0},  
  repository =  
{https://gitlab.lava.polymtl.ca/depots\_publics/modeles/catalogue\_aubes},  
  license = {https://spdx.org/licenses/GPL-3.0+},  
  keywords = {open blade model ; NASA blade ; blade catalogue ;  
mod{\`e}le d'aube ouvert ; aube de la NASA ; catalogue d'aubes},  
  hal_id = {hal-03945336},  
  hal_version = {v1},  
}
```



NASA rotor 74A3



NASA rotor 74A2



NASA rotor 74A1 



NASA rotor 68 



NASA rotor 67 



NASA rotor 66 



NASA rotor 65 



NASA rotor 55 



NASA rotor 54 



NASA rotor 53 



NASA rotor 52 



NASA rotor 51B 



NASA rotor 51A 



NASA rotor 38 

**NASA rotor 37** **NASA rotor 36** **NASA rotor 35** 



NASA rotor 28B 



NASA rotor 27A 



NASA rotor 26B 



NASA rotor 25A 



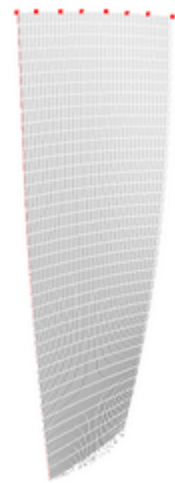
NASA rotor 24A 



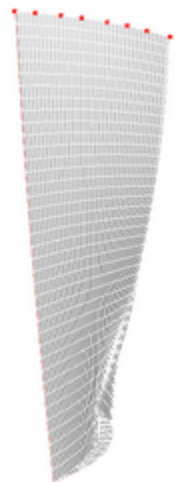
NASA rotor 23B 



NASA rotor 21 



NASA rotor 20 



NASA rotor 18 



NASA rotor 17 



NASA rotor 16 



NASA rotor 15 



NASA rotor 14 



NASA rotor 12 



NASA rotor 11 



NASA rotor 08 



NASA rotor 07 



NASA rotor 06 



NASA rotor 05 



NASA rotor 04 



NASA rotor 03 



NASA rotor 01

Contributions

- Ikram Beghdadi (hiver 2023) mise à jour des pages pour les rotors 1 à 74A3 (structure, liens, images, références bibliographiques, fréquences)
- Solène Kojtych (été 2022-aujourd'hui) supervision, relecture, corrections ponctuelles
- Antoine Lémery (été 2022) création du catalogue

Commentaires sur les maillages

Les maillages fournis (au format `.unv` et `.med`) contiennent des éléments 1D (poutres), des éléments 2D (triangles et quadrangles) et les éléments 3D (pentaèdres) effectivement utilisés pour les calculs éléments finis. La présence d'éléments 1D et 2D est due à la méthodologie de construction du maillage utilisée dans Salome: discrétisation des segments (donne des mailles 1D), puis discrétisation des faces (donne des mailles 2D) puis obtention des mailles 3D.

Commentaires sur le format unv

Structure générale

- le fichier est organisé en jeux de données (*dataset*) décrivant les noeuds, éléments, groupes et champs éventuels
- dans un jeu de données, chaque entité est décrite par une ou plusieurs lignes (*records*), chacune étant constitué d'un ou plusieurs champs (*fields*), séparés par des espaces
 - quelques jeux de données courants: 2411 (noeuds), 2412 (éléments), 2467 (groupes)
 - [syntaxe de référence pour tous les dataset](#) (seuls les jeux utilisés présentement sont affichés par défaut, utiliser la barre de recherche pour les autres)

fichier d'exemple : [maillage de l'aube initiale rotor 67](#)

- **I.13 à 16** : dataset [2420](#) ⇒ définition du repère

Global Cartesian Coordinate System

1.0000000000000000E+0 (axe x)	0.0000000000000000E+0	0.0000000000000000E+0
0.0000000000000000E+0 (axe y)	1.0000000000000000E+0	0.0000000000000000E+0
0.0000000000000000E+0 (axe z)	0.0000000000000000E+0	1.0000000000000000E+0
0.0000000000000000E+0 (origine)	0.0000000000000000E+0	0.0000000000000000E+0

- **I.20 à 40417** : dataset [2411](#) ⇒ **définition des noeuds**
 - par exemple **I.20 à 21** :

1 (id noeud)	1 (id du repère de description des coordonnées)	1 (id du repère utilisé pour les déplacements)	11 (couleur)
2.4451824556310103E-01 (coordonnées)	-3.8067500825705453E-02	6.7879268792358172E-02	

- **I.40421 à 65996** : dataset [2412](#) ⇒ **définition des éléments**
 - élément de poutre (par exemple **I.40421 à 40423**) :

1 (id élément)	22 (type élément, ici poutre quadratique)
2 (id de la table de propriétés physiques)	1 (id de la table de propriétés matériau)
7 (couleur)	3 (nombre de noeuds de l'élément)
0	1
1 (orientation poutre, indice section transverse début poutre, indice section transverse fin de poutre)	
1	4681
3 (id des 3 noeuds de l'élément)	

- élément autre que des poutres, par exemple triangle (**I.41105 - 41106**)

229 (id élément)	42 (type élément, ici triangle quadratique)
2 (id de la table de propriétés physiques)	1 (id de la table de propriétés matériau)
7 (couleur)	6 (nombre de noeuds de l'élément)
117	4797
118	7926
224	7927 (id des noeuds de l'élément)

- **I.66000 à 66620** : dataset [2467](#) ⇒ **définition des groupes de noeuds et d'éléments**

Références

- [description sommaire du format](#)
- [syntaxe de référence officielle pour tous les jeux de données](#)

Ajout d'un nouveau modèle

[procédure](#)

Documents techniques



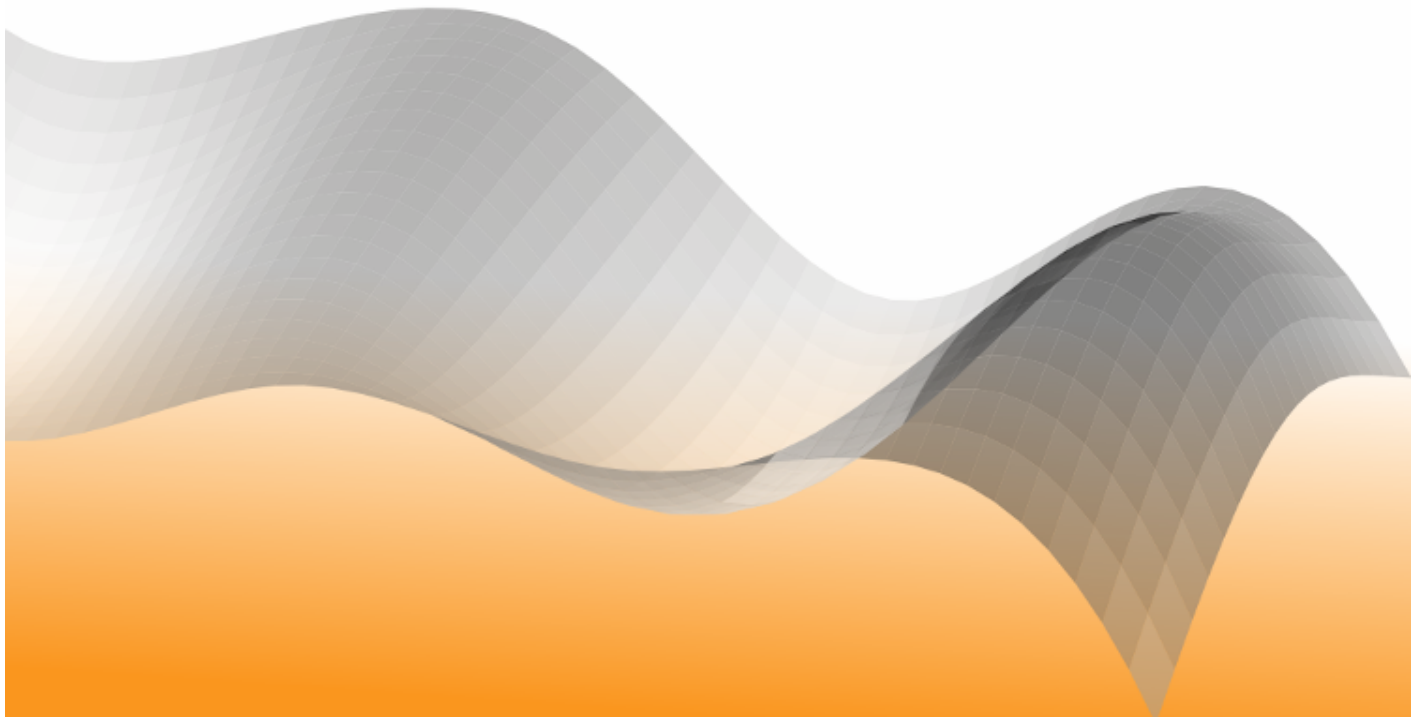
date : **November 20, 2020**

authors : **Quentin Agrapart** (quentin.agrapart@polymtl.ca)
Alain Batailly (alain.batailly@polymtl.ca)

online resources :

wiki: <https://lava-wiki.meca.polymtl.ca/public/notespline/accueil>

Cubic and bicubic spline interpolation in Python



• **HAL**



Q. Agrapart, *Interpolation par courbes splines cubiques et bi-cubiques avec Python*, 2020.



NOTE TECHNIQUE

MESURES D'ACCÉLÉRATIONS ET SYSTÈME D'ACQUISITION PAR MICROCONTRÔLEUR ARDUINO UNO

Solène KOJTYCH

Laboratoire d'Analyse Vibratoire et Acoustique (LAVA)

Département de Génie Mécanique, École Polytechnique de Montréal

- HAL

- 

S. Kojtych, *Mesures d'accélérations et système d'acquisition par microcontrôleur Arduino Uno*, 2019.

Signal analysis techniques based on empirical mode decomposition

Nicolas Di Palma¹, Alain Batailly^{2,*}, Mathias Legrand¹

Abstract

This document aims at presenting, assessing and comparing several signal analysis techniques relying on the empirical mode decomposition. This adaptative method, developed by Huang, is frequently used for the analysis of nonlinear signals, *i.e.* signals stemming from nonlinear systems. One weakness of the method lies in its lack of theoretical fundation, though it should be mentioned that some authors attempted to make a more mathematical presentation of the method. Nonetheless, the EMD has been applied for a large variety of applications—such as industrial machines monitoring and medical signals analysis to name a few. The EMD is the cornerstone of increasingly popular signal analysis techniques such as the local mean decomposition and the Hilbert-Huang Transform. Theses methods are detailed in this note along with the associated Python implementation that is freely downloadable with this document.

Keywords

Empirical mode decomposition; Local mean decomposition; Hilbert-Huang Transform; Edge effects; Signal analysis

1 - Structural Dynamics and Vibration Laboratory, McGill University, 817 Sherbrooke West, McConnell Engineering Bldg, Room 122, H3A 0C3, Montréal, Québec, Canada

2 - Laboratoire d'Acoustics and Vibration Analysis, École Polytechnique de Montréal, C.P. 6079, Succ. Centre-ville, Montréal, Québec, CANADA, H3C 3A7

* Corresponding author: Tel.: +1514 340 4711 - 3932. E-mail address: alain.batailly@polymtl.ca

Méthodes de traitement du signal par décomposition en modes empiriques

Nicolas Di Palma¹, Alain Batailly^{2,*}, Mathias Legrand¹

Résumé

Cet article présente, évalue et compare différentes techniques de traitement du signal reposant sur la décomposition en modes empiriques. Initialement introduite par Huang, la décomposition en modes empiriques est une méthode adaptative fréquemment utilisée pour l'analyse de réponses vibratoires de systèmes mécaniques non-linéaires. Malheureusement, l'assise théorique de la méthode est manquante même si quelques contributions abouties proposent une présentation mathématique. Elle a néanmoins été appliquée avec succès dans des domaines aussi variés que le diagnostic de machines industrielles ou l'analyse de signaux médicaux, et constitue le socle commun à plusieurs méthodes récentes de traitement du signal telles que la décomposition en moyenne locale et la méthode de Hilbert-Huang. Ces méthodes sont détaillées et leur mise en œuvre, disponible en annexe en langage Python, est évaluée pour quelques signaux académiques.

Mots-clés

Décomposition en mode empirique; Décomposition en moyenne locale; Transformée de Hilbert-Huang; Effets de bord; Traitement du signal

1 - Laboratoire de Dynamique des Structures et Vibrations, Université McGill, 817 Sherbrooke Ouest, Pavillon McConnell, Salle 122, H3A-0C3, Montréal, Québec, Canada

2 - Laboratoire d'Analyse Vibratoire et Acoustique, École Polytechnique de Montréal, C.P. 6079, Succ. Centre-ville, Montréal, Québec, CANADA, H3C 3A7

* Corresponding author: Tel.: +1514 340 4711 - 3932. E-mail address: alain.batailly@polymtl.ca

1. Introduction

Le développement de méthodes numériques pour le traitement de signaux non-linéaires (signaux issus de systèmes mécaniques gouvernés par des équations du mouvement non-linéaires) et non-stationnaires (signaux dépendant du temps) a été l'objet de nombreuses publications, notamment au cours des vingt dernières années. De tels signaux ne peuvent pas être analysés de façon optimale par les méthodes usuelles de traitement du signal (transformée de Fourier ou ondelettes) qui ont été développées pour des signaux linéaires. En effet, lorsque la transformée de Fourier est utilisée pour des signaux non-linéaires non-stationnaires, typiquement sous forme de spectrogramme, des limites de résolution apparaissent. D'autre part, la transformée en ondelettes présente l'inconvénient de devoir adapter les familles d'ondelettes au signal (densité du contenu fréquentiel, discontinuités éventuelles...).

Les méthodes adaptatives, dont il est question ici, offrent une perspective intéressante pour l'analyse de signaux non-linéaires et non-stationnaires. L'incarnation la plus connue est probablement la méthode dite de Hilbert-Huang [1]. Elle comprend deux étapes distinctes: (1) une première décomposition du signal en une somme de signaux appelés *modes empiriques*; on parle de *décomposition en modes empiriques* et (2) une transformée de Hilbert des modes empiriques afin d'obtenir les fréquences instantanées associées

• HAL

• 

N. Di Palma, *Méthodes de traitement du signal par décomposition en modes empiriques*, 2018.



Introduction to signal processing and frequency analysis with Fourier transforms

Solène Kojtych¹

Résumé

The aim of this technical note is to provide practical knowledge on signal processing based on Fourier transforms. Other decomposition bases for signal analysis as well as time-frequency analysis are out of the scope of this note. Theoretical concepts are introduced and illustrated with practical examples that are preferred to an exhaustive mathematical description for the sake of clarity. This way, the required mathematical background to read this technical note corresponds approximately to the one of a second-year student of an undergraduate science program.

First, different types of usual signals are introduced and both time and frequency analyses are presented. Then the continuous signal processing based on the continuous Fourier transform is introduced and generalized to discrete signals with the discrete-time Fourier transform. Afterwards the influence of sampling on the quality of the signal processing is discussed. Finally the digital signal processing is introduced thanks to the discrete Fourier transform.

Keywords

signal processing, Fourier transforms, frequency analysis

1 - Department of Mechanical Engineering, Polytechnique Montréal, P.O. Box 6079, Succ. Centre-Ville, Montréal, Qc, Canada H3C 3A7

Introduction au traitement du signal et à l'analyse fréquentielle par transformées de Fourier

Solène Kojtych¹

Résumé

Cette note technique a pour but de fournir des éléments de compréhension pour l'analyse de signaux par transformées de Fourier. Elle ne traite pas des autres bases de fonctions permettant de décomposer un signal ni de l'analyse temps-fréquence. Les concepts sont introduits progressivement et appuyés par des exemples facilitant la compréhension plutôt que par une analyse mathématique poussée. Les connaissances en mathématiques correspondant au début de cycle universitaire en sciences sont supposées connues.

Tout d'abord plusieurs types de signaux usuels sont présentés, puis les concepts d'analyse temporelle et fréquentielle sont introduits. L'analyse de signaux continus est abordée avec la transformée de Fourier continue, puis sa généralisation à l'analyse de signaux discrets, la transformée de Fourier à temps discret, est introduite. L'influence de l'échantillonnage sur la qualité de l'analyse est discutée, puis l'analyse de signaux discrets est abordée de manière pratique, avec la transformée de Fourier discrète.

Mots-clés

traitement du signal, transformées de Fourier, analyse fréquentielle

• HAL



S. Kojtych, *Analyse de signal par transformées de Fourier*, 2017.

How to extract structural matrices (mass, stiffness...) from Ansys and import them into Matlab. Obtention of elementary cyclic symmetric matrices from an elementary sector in Ansys.

Alain Batailly*

Abstract

The purpose of this note is to explain how to extract stiffness and mass matrices from Ansys. This can be useful for specific calculations with other codes such as Matlab. Section one focuses on structures featuring cyclic symmetric where the stiffness and mass matrices of only one elementary sector allow to obtain the global matrices. Section two presents the Ansys commands to obtain the binary files (*.emat* and *.full* files) containing the matrices. Section three is devoted to the interpretation of these files.

Keywords

Cyclic symmetry, Ansys, Matlab.

1 - Structural Dynamics and Vibration Laboratory, McGill University, 817 Sherbrooke West, McConnell Engineering Bldg, Room 122, H3A 0G3, Montréal, Québec, Canada

* Corresponding author: Tel.: +1 514 398 5321; fax: +1 514 398 7365. E-mail address: alain.batailly@mcgill.ca

Comment extraire les matrices structurelles (masse, raideur...) d'Ansys pour les importer dans Matlab. Obtention des matrices cycliques élémentaires à partir d'un secteur élémentaire lu dans Ansys.

Alain Batailly*

Résumé

L'objectif de cette note technique est d'expliquer la procédure permettant d'extraire les matrices masse et raideur d'un modèle éléments finis créé avec le logiciel Ansys. Cette procédure peut être utile lorsque des manipulations sur les matrices structurelles sont requises (par exemple sous Matlab ou Python) en dehors du cadre défini par Ansys. La première section porte sur les structures à symétrie cyclique pour lesquelles les matrices masse et raideur du secteur élémentaire permettent de reconstituer les matrices masse et raideur de la structure globale. La deuxième section porte quant à elle sur les commandes Ansys permettant d'obtenir les fichiers binaires (*.emat* et *.full*) qui contiennent ces matrices. La dernière section est dédiée à l'interprétation de ces fichiers.

Mots-clés

Symétrie cyclique, Ansys, Matlab.

- HAL

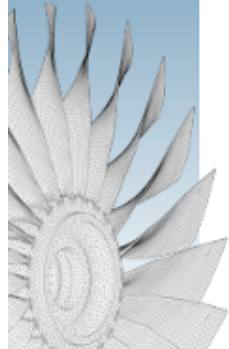
- 

A. Batailly, *Extraction de matrices masse et raideur depuis Ansys*, 2015.

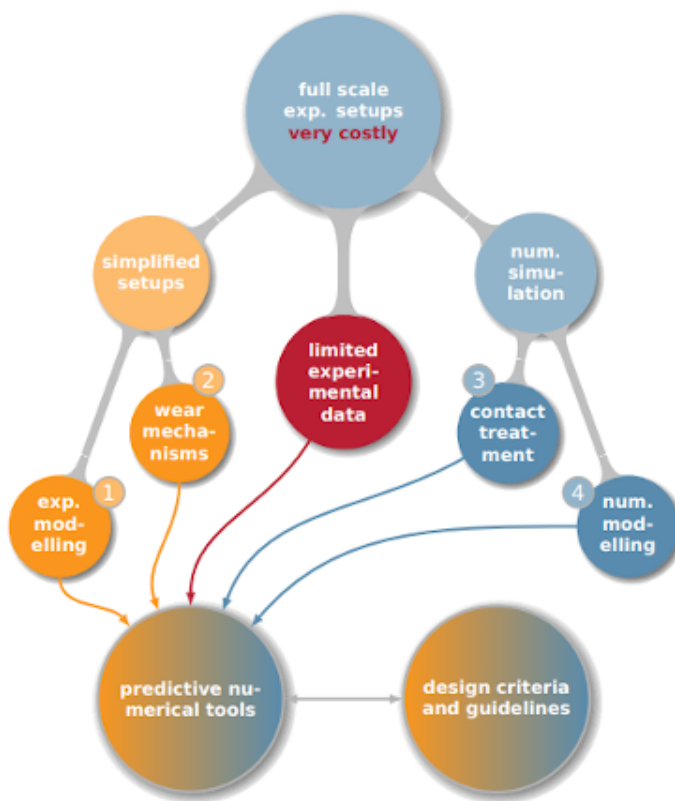
ASME Turbo Expo 2020
+

I - Rotor/stator interactions
Overview
Blade/casing interface
Problem-solving strategy

II - Research investigations
III - Industrial applications
IV - Going forward...
References



Understanding and predicting blade/casing contacts




• HAL



A. Batailly, *Interactions aubes/carter dans les turbomachines*, 2020.

Codes

Retrouvez tous les codes du LAVA disponibles en libre accès sur sa [plateforme Gitlab publique](#).

nom	accès	GitLab	auteur	année
Py2tikz	page wiki		Thibaut Vadcard	2021

Open access

Do you wish to know more about research data management ? This [tutorial](#) is for you ! Informational videos are available [here](#).

Numerical models

NASA blades

To cite this catalogue: reference article

×

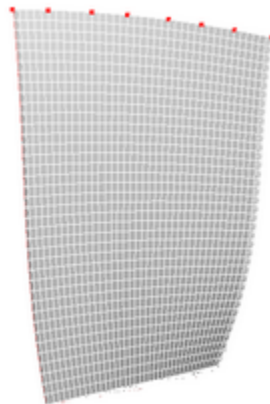
```
@proceedings{10.1115/GT2023-101107,
  author = {Kojtych, Solène and Batailly, Alain},
  title = "{Open NASA blade models for nonlinear dynamics simulations}",
  volume = {Volume 8A: Structures and Dynamics – Aerodynamics Excitation and Damping; Bearing and Seal Dynamics; Emerging Methods in Engineering Design, Analysis, and Additive Manufacturing; Fatigue, Fracture, and Life Prediction},
  series = {Turbo Expo: Power for Land, Sea, and Air},
  year = {2023},
  month = {06},
  doi = {10.1115/GT2023-101107},
  url = {https://doi.org/10.1115/GT2023-101107},
}
```

files in open access

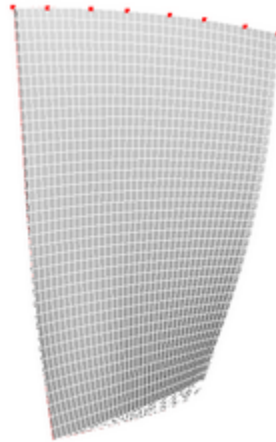
×

```
@softwareversion{kojtych:hal-03945336v1,
  title = {{A catalogue of open NASA blade models}},
```

```
author = {Kojtych, Sol{\`e}ne and Batailly, Alain},
url = {https://hal.science/hal-03945336},
note = {},
year = {2023},
month = Jan,
swhid =
{swh:1:dir:1ac35a9ed33ed49c0f3297f08bb5bb0a6578da5c;origin=https://gitlab
.lava.polymtl.ca/depots\_publics/modeles/catalogue\_aubes;visit=swh:1:snp
:6d1d388afc61917fa91efdb52e5cc17c4ea3d0a5;anchor=swh:1:rev:108595ab39d617
f73483ed578be9a657e3637c95},
verison = {1.0},
repository =
{https://gitlab.lava.polymtl.ca/depots\_publics/modeles/catalogue\_aubes},
license = {https://spdx.org/licenses/GPL-3.0+},
keywords = {open blade model ; NASA blade ; blade catalogue ;
mod{\`e}le d'aube ouvert ; aube de la NASA ; catalogue d'aubes},
hal_id = {hal-03945336},
hal_version = {v1},
}
```



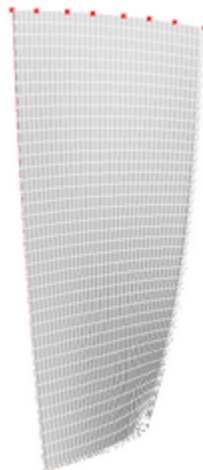
NASA rotor 74A3



NASA rotor 74A2 



NASA rotor 74A1 



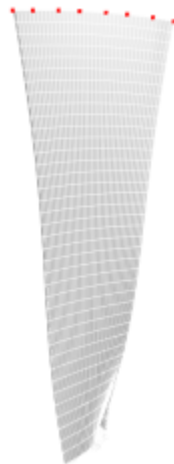
NASA rotor 68 



NASA rotor 67 



NASA rotor 66 



NASA rotor 65 



NASA rotor 55 



NASA rotor 54 



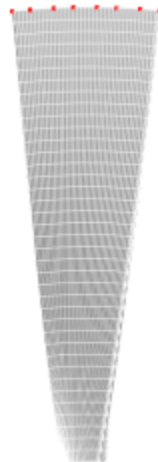
NASA rotor 53 



NASA rotor 52 



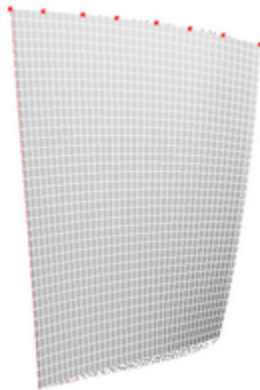
NASA rotor 51B 



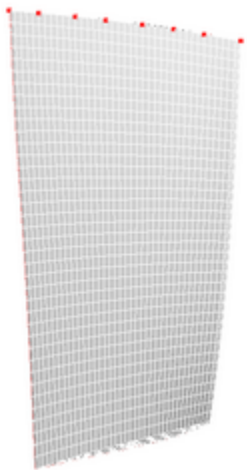
NASA rotor 51A 



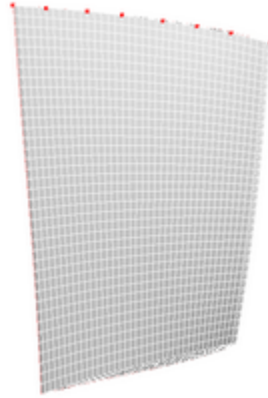
NASA rotor 38 



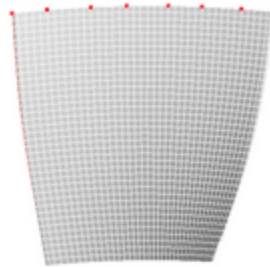
NASA rotor 37 



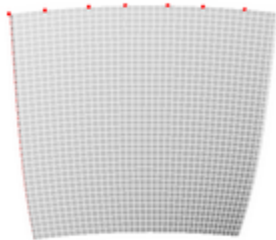
NASA rotor 36 



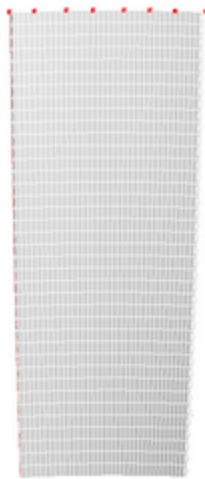
NASA rotor 35 

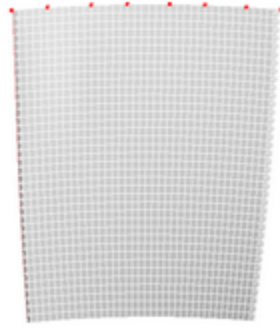


NASA rotor 28B 

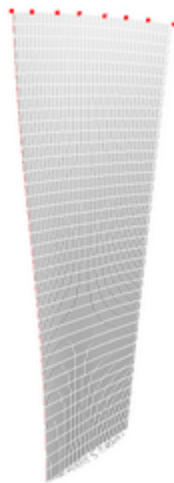


NASA rotor 27A 

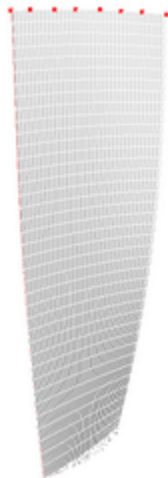
**NASA rotor 26B** **NASA rotor 25A** **NASA rotor 24A** 



NASA rotor 23B 



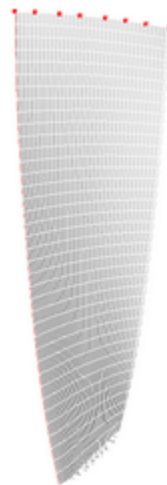
NASA rotor 21 



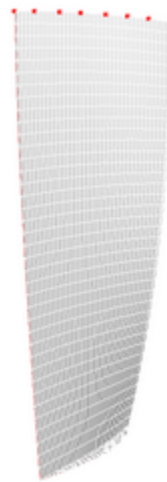
NASA rotor 20 



NASA rotor 18 



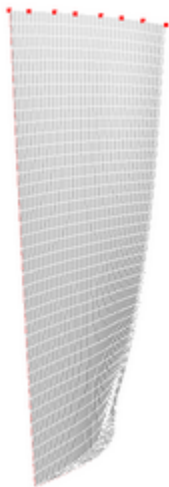
NASA rotor 17 



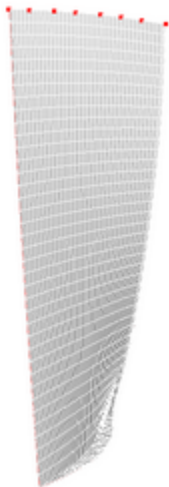
NASA rotor 16 



NASA rotor 15 



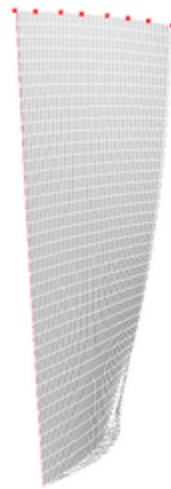
NASA rotor 14 



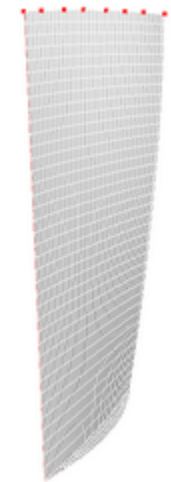
NASA rotor 12 



NASA rotor 11 



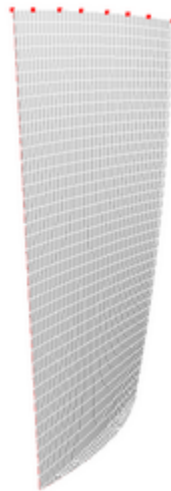
NASA rotor 08 



NASA rotor 07 



NASA rotor 06 



NASA rotor 05 



NASA rotor 04 



NASA rotor 03



NASA rotor 01

Contributions

- Ikram Beghdadi (winter 2023) update pages of rotor 1 to 74A3 (structure, links, images, references, frequencies)
- Solène Kojtych (summer 2022-today) supervision, review, minor corrections
- Antoine Lémery (summer 2023) catalogue creation

Remarks on the mesh files

The provided mesh files (.unv and .med) contain not only 3D elements (pentahedron) used for finite element computations but also 2D elements (triangles and quadrangles) and 1D elements (beam). This is a consequence of the meshing methodology used in Salome: first edges are discretized (1d elements are computed), then faces are generated (2D elements) and finally 3D elements.

Comments on the .unv format

General structure

- the unv file is split in different *dataset* describing nodes, elements, groups and field
- in a data set, each object is described by one or several lines, called *records*. Each line contains one or several *fields*, separated by spaces
 - some common datasets are: 2411 (nodes), 2412 (elemnts), 2467 (groups)
 - [official syntax for the different dataset](#) (only currently used dataset are presented on the main page, use the research tool for the others)

file example : [initial rotor 67 blade](#)

- [I.13 to 16](#) : dataset [2420](#) ⇒ coordinate system definition

```
Global Cartesian Coordinate System
  1.0000000000000000E+0  0.0000000000000000E+0  0.0000000000000000E+0
(x axis)
  0.0000000000000000E+0  1.0000000000000000E+0  0.0000000000000000E+0
(y axis)
  0.0000000000000000E+0  0.0000000000000000E+0  1.0000000000000000E+0
(z axis)
  0.0000000000000000E+0  0.0000000000000000E+0  0.0000000000000000E+0
(origin)
```

- [I.20 to 40417](#) : dataset [2411](#) ⇒ **node definition**
 - for instance [I.20 to 21](#) :

```
1 (node id)          1 (id of the physical coordinate system)  1 (id of the
displacement coordinate system)          11 (color)
  2.4451824556310103E-01  -3.8067500825705453E-02  6.7879268792358172E-02
(coordinates)
```

- [I.40421 to 65996](#) : dataset [2412](#) ⇒ **elements definition**
 - beam element (for instance [I.40421 to 40423](#)) :

```
1 (element id)      22 (element type, here a quadratic beam)
2 (physical property table id)  1 (material property table id)  7
(color)             3 (number of nodes on element)
  0           1           1 (beam orientation, beam fore-end cross section
id, beam aft-end cross section id)
  1           4681          3 (3 node labels defining element)
```

- non-beam elements, for instance triangle ([I.41105 to 41106](#))

```
229 (element id)    42 (element type, here a quadratic triangle)
2 (physical property table id)  1 (material property table id)  7
(color)             6 (number of nodes on element)
```

117	4797	118	7926	224	7927 (node labels defining element)
-----	------	-----	------	-----	--

- **I.66000 to 66620** : dataset 2467 ⇒ **definition of groups of nodes or elements**

References

- [high level description of the unv format](#)
- [official syntax reference for all datasets](#)

Adding a new model to the catalogue

[how to](#)

Technical documents



date : **November 20, 2020**

authors : **Quentin Agrapart** (quentin.agrapart@polymtl.ca)
Alain Batailly (alain.batailly@polymtl.ca)

online resources :

wiki: <https://lava-wiki.meca.polymtl.ca/public/notespline/accueil>

Cubic and bicubic spline interpolation in Python



• [HAL](#)



Q. Agrapart, *Cubic and bicubic B-spline interpolation with Python*, 2020.



NOTE TECHNIQUE

MESURES D'ACCÉLÉRATIONS ET SYSTÈME D'ACQUISITION PAR MICROCONTRÔLEUR ARDUINO UNO

Solène KOJTYCH

Laboratoire d'Analyse Vibratoire et Acoustique (LAVA)
Département de Génie Mécanique, École Polytechnique de Montréal

• HAL



S. Kojtych, *Acceleration measurements and acquisition system with Arduino Uno* (in French), 2019.

Signal analysis techniques based on empirical mode decomposition

Nicolas Di Palma¹, Alain Batailly^{2,*}, Mathias Legrand¹

Abstract

This document aims at presenting, assessing and comparing several signal analysis techniques relying on the empirical mode decomposition. This adaptative method, developed by Huang, is frequently used for the analysis of nonlinear signals, *i.e.* signals stemming from nonlinear systems. One weakness of the method lies in its lack of theoretical foundation, though it should be mentioned that some authors attempted to make a more mathematical presentation of the method. Nonetheless, the EMD has been applied for a large variety of applications—such as industrial machines monitoring and medical signals analysis to name a few. The EMD is the cornerstone of increasingly popular signal analysis techniques such as the local mean decomposition and the Hilbert-Huang Transform. These methods are detailed in this note along with the associated Python implementation that is freely downloadable with this document.

Keywords

Empirical mode decomposition; Local mean decomposition; Hilbert-Huang Transform; Edge effects; Signal analysis

¹ - Structural Dynamics and Vibration Laboratory, McGill University, 817 Sherbrooke West, McConnell Engineering Bldg, Room 122, H3A 0C3, Montréal, Québec, Canada

² - Laboratory of Acoustics and Vibration Analysis, École Polytechnique de Montréal, C.P. 6079, Succ. Centre-ville, Montréal, Québec, CANADA, H3C 3A7

* Corresponding author: Tel.: +1 514 340 4711 - 3932. E-mail address: alain.batailly@polymtl.ca

Méthodes de traitement du signal par décomposition en modes empiriques

Nicolas Di Palma¹, Alain Batailly^{2,*}, Mathias Legrand¹

Résumé

Cet article présente, évalue et compare différentes techniques de traitement du signal reposant sur la décomposition en modes empiriques. Initialement introduite par Huang, la décomposition en modes empiriques est une méthode adaptative fréquemment utilisée pour l'analyse de réponses vibratoires de systèmes mécaniques non-linéaires. Malheureusement, l'assise théorique de la méthode est manquante même si quelques contributions abouties proposent une présentation mathématique. Elle a néanmoins été appliquée avec succès dans des domaines aussi variés que le diagnostic de machines industrielles ou l'analyse de signaux médicaux, et constitue le socle commun à plusieurs méthodes récentes de traitement du signal telles que la décomposition en moyenne locale et la méthode de Hilbert-Huang. Ces méthodes sont détaillées et leur mise en œuvre, disponible en annexe en langage Python, est évaluée pour quelques signaux académiques.

Mots-clés

Décomposition en mode empirique; Décomposition en moyenne locale; Transformée de Hilbert-Huang; Effets de bord; Traitement du signal

¹ - Laboratoire de Dynamique des Structures et Vibrations, Université McGill, 817 Sherbrooke Ouest, Pavillon McConnell, Salle 122, H3A-0C3, Montréal, Québec, Canada

² - Laboratoire d'Analyse Vibratoire et Acoustique, École Polytechnique de Montréal, C.P. 6079, Succ. Centre-ville, Montréal, Québec, CANADA, H3C 3A7

* Corresponding author: Tel.: +1 514 340 4711 - 3932. E-mail address: alain.batailly@polymtl.ca

1. Introduction

Le développement de méthodes numériques pour le traitement de signaux non-linéaires (signaux issus de systèmes mécaniques gouvernés par des équations du mouvement non-linéaires) et non-stationnaires (signaux dépendant du temps) a été l'objet de nombreuses publications, notamment au cours des vingt dernières années. De tels signaux ne peuvent pas être analysés de façon optimale par les méthodes usuelles de traitement du signal (transformée de Fourier ou ondelettes) qui ont été développées pour des signaux linéaires. En effet, lorsque la transformée de Fourier est utilisée pour des signaux non-linéaires non-stationnaires, typiquement sous forme de spectrogramme, des limites de résolution apparaissent. D'autre part, la transformée en ondelettes présente l'inconvénient de devoir adapter les familles d'ondelettes au signal (densité du contenu fréquentiel, discontinuités éventuelles...).

Les méthodes adaptatives, dont il est question ici, offrent une perspective intéressante pour l'analyse de signaux non-linéaires et non-stationnaires. L'incarnation la plus connue est probablement la méthode dite de Hilbert-Huang [1]. Elle comprend deux étapes distinctes: (1) une première décomposition du signal en une somme de signaux appelés *modes empiriques*; on parle de *décomposition en modes empiriques* et (2) une transformée de Hilbert des modes empiriques afin d'obtenir les fréquences instantanées associées

- HAL

- 

N. Di Palma, *Empirical Mode Decomposition-based signal analysis* (in French), 2018.



Introduction to signal processing and frequency analysis with Fourier transforms

Solène Kojtych¹

Résumé

The aim of this technical note is to provide practical knowledge on signal processing based on Fourier transforms. Other decomposition bases for signal analysis as well as time-frequency analysis are out of the scope of this note. Theoretical concepts are introduced and illustrated with practical examples that are preferred to an exhaustive mathematical description for the sake of clarity. This way, the required mathematical background to read this technical note corresponds approximately to the one of a second-year student of an undergraduate science program.

First, different types of usual signals are introduced and both time and frequency analyses are presented. Then the continuous signal processing based on the continuous Fourier transform is introduced and generalized to discrete signals with the discrete-time Fourier transform. Afterwards the influence of sampling on the quality of the signal processing is discussed. Finally the digital signal processing is introduced thanks to the discrete Fourier transform.

Keywords

signal processing, Fourier transforms, frequency analysis

1 - Department of Mechanical Engineering, Polytechnique Montréal, P.O. Box 6079, Succ. Centre-Ville, Montréal, Qc, Canada H3C 3A7

Introduction au traitement du signal et à l'analyse fréquentielle par transformées de Fourier

Solène Kojtych¹

Résumé

Cette note technique a pour but de fournir des éléments de compréhension pour l'analyse de signaux par transformées de Fourier. Elle ne traite pas des autres bases de fonctions permettant de décomposer un signal ni de l'analyse temps-fréquence. Les concepts sont introduits progressivement et appuyés par des exemples facilitant la compréhension plutôt que par une analyse mathématique poussée. Les connaissances en mathématiques correspondant au début de cycle universitaire en sciences sont supposées connues.

Tout d'abord plusieurs types de signaux usuels sont présentés, puis les concepts d'analyse temporelle et fréquentielle sont introduits. L'analyse de signaux continus est abordée avec la transformée de Fourier continue, puis sa généralisation à l'analyse de signaux discrets, la transformée de Fourier à temps discret, est introduite. L'influence de l'échantillonnage sur la qualité de l'analyse est discutée, puis l'analyse de signaux discrets est abordée de manière pratique, avec la transformée de Fourier discrète.

Mots-clés

traitement du signal, transformées de Fourier, analyse fréquentielle

- HAL

-

S. Kojtych, *Signal processing through Fourier transforms* (in French), 2017.

How to extract structural matrices (mass, stiffness. . .) from Ansys and import them into Matlab. Obtention of elementary cyclic symmetric matrices from an elementary sector in Ansys.

Alain Batailly*

Abstract

The purpose of this note is to explain how to extract stiffness and mass matrices from Ansys. This can be useful for specific calculations with other codes such as Matlab. Section one focuses on structures featuring cyclic symmetric where the stiffness and mass matrices of only one elementary sector allow to obtain the global matrices. Section two presents the Ansys commands to obtain the binary files (*.emat* and *.full* files) containing the matrices. Section three is devoted to the interpretation of these files.

Keywords

Cyclic symmetry, Ansys, Matlab.

1 - Structural Dynamics and Vibration Laboratory, McGill University, 817 Sherbrooke West, McConnell Engineering Bldg, Room 122, H3A 0G3, Montréal, Québec, Canada

* Corresponding author: Tel.: +1 514 398 5321; fax: +1 514 398 7365. E-mail address: alain.batailly@mcgill.ca

Comment extraire les matrices structurelles (masse, raideur. . .) d'Ansys pour les importer dans Matlab. Obtention des matrices cycliques élémentaires à partir d'un secteur élémentaire lu dans Ansys.

Alain Batailly*

Résumé

L'objectif de cette note technique est d'expliquer la procédure permettant d'extraire les matrices masse et raideur d'un modèle éléments finis créé avec le logiciel Ansys. Cette procédure peut être utile lorsque des manipulations sur les matrices structurelles sont requises (par exemple sous Matlab ou Python) en dehors du cadre défini par Ansys. La première section porte sur les structures à symétrie cyclique pour lesquelles les matrices masse et raideur du secteur élémentaire permettent de reconstituer les matrices masse et raideur de la structure globale. La deuxième section porte quant à elle sur les commandes Ansys permettant d'obtenir les fichiers binaires (*.emat* et *.full*) qui contiennent ces matrices. La dernière section est dédiée à l'interprétation de ces fichiers.

Mots-clés

Symétrie cyclique, Ansys, Matlab.

• HAL



A. Batailly, *How to extract structural matrices (mass, stiffness...) from Ansys*, 2015.

ASME Turbo Expo 2020

↑

I - Rotor/stator interactions

Overview

Blade/casing interface

Problem-solving strategy

II - Research investigations

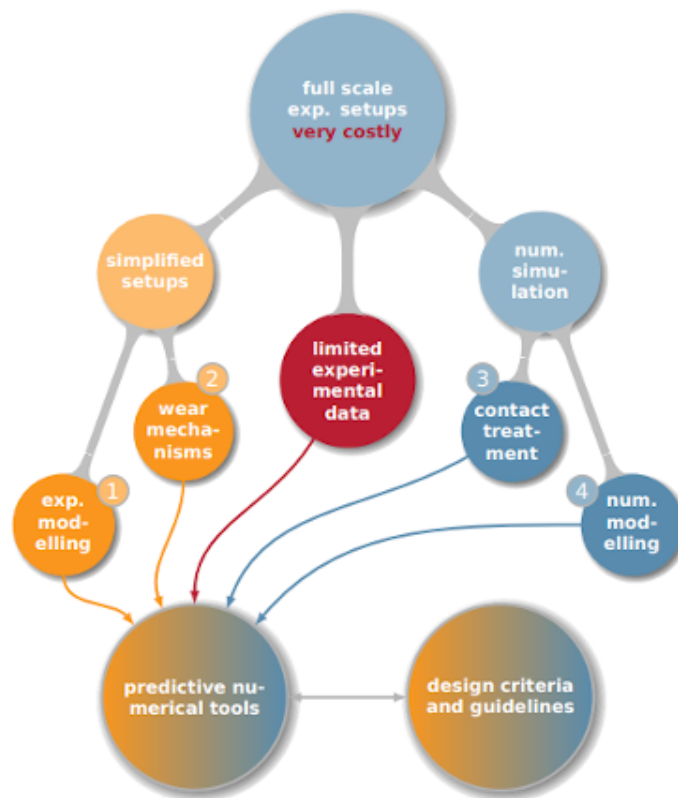
III - Industrial applications

IV - Going forward...

References



Understanding and predicting blade/casing contacts




- HAL

- 

A. Batailly, *Blade/casing contacts in turbomachinery*, 2020.

Codes

You may find all the open access codes and templates from the LAVA on its [public Gitlab platform](#).

name	access	GitLab	author	year
Py2tikz	wiki page		Thibaut Vadcard	2021

Document issu de la page wiki:

<https://lava-wiki.meca.polymtl.ca/public/accueil?rev=1699377196>

Dernière mise à jour: **2023/11/07 12:13**