POLITECNICO DI TORINO

FACOLTÀ DI INGEGNERIA MECHANICA

TESI DI LAUREA MAGISTRALE

AN EFFICIENT SIMULATION STRATEGY FOR THE DESIGN OF COMPOSITE ACOUSTIC METAMATERIALS

Zhijun He

Supervisors:
Prof. Erasmo Carrera
Prof. Maria Cinefra
Prof. Marco Petrolo

March 2023

Acknowledgement

I want to thank Prof. Erasmo Carrera, from Politecnico di Torino, for the opportunities he offered me to overcome practical challenges in complex simulation environments and advanced material designs. I also want to thank Prof. Maria Cinefra for her guidance and suggestions in my study field. I want to thank Prof. Marco Petrolo for his help and suggestions for accomplishing my thesis.

Abstract

Acoustic metamaterials are artificial structures that apply in fluid-structure systems to solve vibro-acoustic problems. In general, acoustic metamaterials are substructure-compounded systems. They have unconventional effective properties independent of their chemical or physical natures. In history, acoustic metamaterial designs started around 2000. So far, the effective methods to test acoustic metamaterials are experiments and simulations. However, practical experiments are relatively expensive and hard to extend. Thus, simulations are the promising choice left. Still, no ready-made simulation software is dedicated to the tests of acoustic metamaterials. FEM software is a reasonable choice since substructure-compounded acoustic metamaterials are analogous to mechanical systems. Then, challenges come from two main aspects: the way to process massive intermediate data and the effective mathematical model to build a vibro-acoustic environment. Therefore, simulation strategy designs considering efficiency and accuracy are needed. At the same time, the simulation strategy design and the acoustic metamaterial design are mutually dependent.

The simulation strategy designs start with choosing the proper simulation software. And this paper chooses MSC Nastran 2020, MSC Actran 2020, and Abaqus 2020 (for convenience, they are called Nastran, Actran, and Abaqus in the following contents) as candidate software due to their distinctive characteristics. Specifically, Nastran is an expert in structure analyses but not competent in acoustic analyses. In contrast, Actran is an expert in acoustic analyses but not so efficient in structure analyses. In comparison, Abaqus is a Multiphysics analyser but not an expert in fluid or structural fields. And Actran's data management capability is weak than Nastran and Abaqus. Finally, it is convinced that Nastran's "ACMS Method" is the best in eigenvalue extraction analyses. Then, Nastran's modal-space-based "Ordinary Method" is the best in steady-state-dynamic analyses of frequency-dependent models. At the same time, Nastran's modal-space-based "ACMS Method" is the best in steady-state-dynamic analyses of non-frequency-dependent models. After that, Nastran's eigenvalue extraction analyses with the "ACMS Method" cooperate with Actran's modal-space-based analyses with the diffuse incident field is the best in vibro-acoustic analyses.

The acoustic metamaterial designs start with a panel-like periodic structure. And its substructures (meta-atoms) are in a centre-mass-intensive form that consists of an aluminium cylindrical core and a melamine cubic base. The aluminium core is isotropic and non-frequency-dependent, while the melamine base is orthotropic and frequency-dependent. Then, based on the basic configuration, the size-controlled panels, the constrained-lateral-boundary-controlled panels, the aluminium-inclusion-controlled panels, and the layers-controlled panels are four derivatives that this paper sets as the preliminary acoustic metamaterial designs. The simulations' task in this paper is to verify their low-frequency sensibility and soundproof capability. Finally, it is convinced that acoustic metamaterials are sensible in the low-frequency range (0–220Hz), and the sensibility is adjustable through the control parameters. At the same time, acoustic metamaterials' soundproof capability is weaker than pure melamine foam and aluminium alloy. Fortunately, increasing the panel's constrained lateral boundaries can significantly increase the material's soundproof capability.

Contents

List of Figures

List of Tables

Symbols

θ_{phase}	The phase lag of responses
ω	Angular frequency
ω_n	Natural frequency
ω_p	The frequency of response peak
f	Temporal frequency
f_k	The property-altering frequency at the kth time
f_i	The excitation frequency at the ith time
$TR(f)$ from [33]	The scaling factors of elasticity (temporal-frequency-dependent, Nastran)
$TI(f)$ from [33]	The scaling factors of anelasticity (temporal-frequency-dependent, Nastran)
ρ	General mass density
ρ_{eff}	Effective mass density
ρ_{AL}	The mass density of aluminium alloy (AL 6061)
ρ_x	X-axis mass density of melamine foam
ρ_y	Y-axis mass density of melamine foam
ρ_z	Z-axis mass density of melamine foam
ρ_{MELA}	The average mass density of melamine foam

ρ_f	Fluid mass density
ρ_s	Structure mass density
$[G_{ot}]$ from [20]	Boundary-DOFs-transformation matrix (Nastran)
$[G_{oq}]$ from [20]	Dynamic transformation matrix (Nastran)
$[G_{oa}]$ from [20]	Synthesised transformation matrix (Nastran)
K	A general bulk modulus
K_{eff}	Effective bulk modulus
K_{AL}	The bulk modulus of aluminium alloy (AL 6061)
K_f	Fluid bulk modulus
$[K_{ij}]_{MELA}$	Melamine foam's material elastic matrix (orthotropic elasticity)
$[K_{ff}]$ from [20]	Unconstrained-DOFs stiffness matrix (Nastran)
$[K_{oo}]$ from [20]	Inner-DOFs stiffness matrix (Nastran)
$[\bar{K}_{tt}]$ from [20]	Outer-DOFs stiffness matrix (Nastran)
$[K_{ot}]$ from [20]	Intersecting-DOFs stiffness matrix (Nastran)
$[K_{tt}]$ from [20]	Condensed-boundary-DOFs stiffness matrix (Nastran)
$[K_{qq}]$ from [20]	Component-modes-DOFs stiffness matrix (Nastran)
$[K_{aa}]$ from [20]	Analysis-DOFs stiffness matrix (Nastran)
$[K]_G$	The general stiffness matrix of s whole system
$[K]_{sd}$	The general structural damping matrix of a whole system
$[K]_{mat}$	The stiffness matrix of a given material
$[K_{dd}]_V$ from [33]	Dynamic stiffness matrix (whole structure, Nastran)

$[K_{dd}^{1}]_{V}$ from [33]	Elastic stiffness matrix (whole part, Nastran)
$[K_{dd}^{4}]_{V}$ from [33]	Structural damping matrix (anelastic part, Nastran)
$[K_{dd}^{4}]_{mode}$ from [33]	Modal structural damping matrix (anelastic part, Nastran)
$[K]_{AL}$	The stiffness matrix of aluminium alloy
$[K]_{MELA}$	The stiffness matrix of melamine foam
$[K]_{e}$ from [36]	The mass matrix of given elements (Abaqus)
$[K]_{matrix}$ from [36]	An input stiffness matrix (Abaqus)
$[K_{\mathrm{f}}]^{PQ}$ from [75]	Fluid elasticity matrix (inner domain, Abaqus)
$[K_{s}]^{NM}$ from [75]	Structure stiffness matrix (inner domain, Abaqus)
M	A general mass
$[M_{ff}]$ from [20]	Unconstrained-DOFs mass matrix (Nastran)
$[M_{oo}]$ from [20]	Inner-DOFs mass matrix (Nastran)
$[\bar{M}_{tt}]$ from [20]	Outer-DOFs mass matrix (Nastran)
$[M_{ot}]$ from [20]	Intersecting-DOFs mass matrix (Nastran)
$[M_{tt}]$ from [20]	Condensed-boundary-DOFs mass matrix (Nastran)
$[M_{qq}]$ from [20]	Component-modes-DOFs mass matrix (Nastran)
$[M_{aa}]$ from [20]	Analysis-DOFs mass matrix (Nastran)
$[M_{tq}]$ from [20]	Coupling-DOFs mass matrix (Nastran)
$[M]_{G}$	The general mass matrix of a whole system
$[M]_{e}$ from [36]	The mass matrix of given elements (Abaqus)
$[M]_{matrix}$ from [36]	An input mass matrix (Abaqus)

$[M_{\mathrm{f}}]^{PQ}$ from [75]	Fluid inertia matrix (inner domain, Abaqus)
$[M_{\mathrm{fr}\cup\mathrm{fi}}]^{PQ}$ from [75]	Fluid inertia matrix (reactive and radiating boundary, Abaqus)
$[M_{\mathrm{s}}]^{NM}$ from [75]	Structure inertia matrix (inner domain, Abaqus)
η	A general structural damping factor
η_{AL}	Structural damping of aluminium alloy (AL 6061)
$\eta_{MELA}(\omega)$	Structural damping of melamine foam (angular-frequency-dependent)
η_{mat}	Material structural damping factor
η_{REF} from [33]	Reference structural damping factor (Nastran)
η_G	Global structural damping factor
η^0_{MELA}	Structural damping of melamine foam (at reference state)
C	A general viscous damping
C_{cr}	Critical damping factor
C_{eq}	Equivalent damping factor
C_{mat}	The viscous damping factor of a given material
$C_{cr,\ mat}$	The critical damping of a given material
$[C]_{vd}$	The general viscous damping matrix of a whole system
$[C_{\mathrm{f}}]^{PQ}$ from [75]	Fluid damping matrix (volumetric drag, inner domain, Abaqus)
$[C_{\mathrm{fr}\cup\mathrm{fi}}]^{PQ}$ from [75]	Fluid damping matrix (reactive and radiating boundary, Abaqus)
$[C_{\mathrm{f}(\eta_f)}]^{PQ}$ from [75]	Fluid damping matrix (global acoustic structural damping, Abaqus)
$[C_{s(\mathrm{m})}]^{NM}$ from [75]	Structure viscous damping matrix (mass-proportional type, inner domain, Abaqus)

$[C_{s(\mathrm{k})}]^{NM}$ from [75]	Structure viscous damping matrix (stiffness-proportional type, inner domain, Abaqus)
$[C_{s(\eta_S)}]^{NM}$ from [75]	Structure structural damping matrix (inner domain, Abaqus)
$[S_{\mathrm{fs}}]^{PM}$ from [75]	Fluid-structure interface matrix (Abaqus)
$[S_{\mathrm{fs}}]^{NQ}$ from [75]	Structure-fluid interface matrix (Abaqus)
ζ	Critical damping ratio
ζ_{eq}	The equivalent critical damping ratio
ζ_i^M from [36]	The equivalent critical damping ratio at ith mode (mass-matrix weighted average, Abaqus)
ζ_{mat} from [36]	The critical damping ratio of a given material
ζ_e^M from [36]	The critical damping ratio of given elements (mass-matrix weighted average, Abaqus)
ζ_{matrix}^M from [36]	The critical damping ratio of an input mass matrix (Abaqus)
ζ_i^K from [36]	The equivalent critical damping ratio at ith mode (stiffness-matrix weighted average, Abaqus)
ζ_e^K from [36]	The critical damping ratio of given elements (stiffness-matrix weighted average, Abaqus)
ζ_{matrix}^K from [36]	The critical damping ratio of an input stiffness matrix (Abaqus)
Q	Quality factor
E_1 from [11]	X-axis Yong's modulus
E_2 from [11]	Y- axis Yong's modulus
E_3 from [11]	Z- axis Yong's modulus
$\tilde{E}_1(\omega)$	X-direction Yong's modulus (angular-frequency-dependent, complex)
$\tilde{E}_2(\omega)$	Y-direction Yong's modulus (angular-frequency-dependent, complex)
$\tilde{E}_3(\omega)$	Z-direction Yong's modulus (angular-frequency-dependent, complex)

$\tilde{E}_{mat}$	The elastic modulus of a given material (complex, Actran)
E_{mat}	The elastic modulus of a given material (real, Actran)
G_{23} from [11]	YZ-plane Shear modulus
G_{31} from [11]	ZX-plane Shear modulus
G_{12} from [11]	XY-plane Shear modulus
$\tilde{G}_{23}(\omega)$	YZ-plane Shear modulus (angular-frequency-dependent, complex)
$\tilde{G}_{31}(\omega)$	ZX-plane Shear modulus (angular-frequency-dependent, complex)
$\tilde{G}_{12}(\omega)$	XY-plane Shear modulus (angular-frequency-dependent, complex)
G_{mat}	Material shear modulus (Nastran)
G_{REF}	Reference shear modulus (Nastran)
$G(f)$ from [33]	Shear modulus (complex, temporal-frequency-dependent, Nastran)
$G'(f)$ from [33]	Shear storage modulus (temporal-frequency-dependent, Nastran)
$G''(f)$ from [33]	Shear loss modulus (temporal-frequency-dependent, Nastran)
ν_{32} from [11]	ZY-related Poisson's ratio
ν_{13} from [11]	XZ-related Poisson's ratio
ν_{21} from [11]	XY-related Poisson's ratio
ν_{mat}	The Poisson's ratio of a given material
β from [11]	Relaxation frequency
b from [11]	Real scalar constant

α from [11]	Fractional derivative order
$[\Phi]$	Eigenmatrix
$[\Phi_o]$	Component eigenmodes matrix (Nastran)
$\bar{\boldsymbol{x}}$	A general position vector
$\bar{\boldsymbol{x}}_s$ from [76]	The position vector of the source point (Abaqus)
$\bar{\boldsymbol{x}}_0$ from [76]	The position vector of the standoff point (Abaqus)
$\bar{\boldsymbol{x}}_j$ from [76]	The position vector of the loaded point (Abaqus)
τ_j from [76]	Retarded time of the loading point regarding the standoff point (Abaqus)
R_j from [76]	The distance between the source point and the loaded point (Abaqus)
R_0 from [76]	The distance between the source point and the standoff point (Abaqus)
$\tilde{p}_t$ from [76]	The amplitude of incident pressure at the standoff point (Abaqus)
$p_t(\tau_j)$ from [76]	Time history function of the loaded point (Abaqus)
$p_{\bar{x}}(\bar{\boldsymbol{x}}_j)$ from [76]	The spatial decay function of the incident wave (Abaqus)
$p_f(\bar{\boldsymbol{x}},t)$ from [76]	Harmonic fluid pressure (a function of position vector and time)
$p_I(\bar{\boldsymbol{x}}_j,t)$ from [76]	The incident wave function on a loaded point (Abaqus)
$\tilde{P}_f(\bar{\boldsymbol{x}})$ from [76]	Fluid pressure amplitude (complex, a function of position vector)
$\hat{P}_f(\bar{\boldsymbol{x}})$	The modulus of $\tilde{P}_f(\bar{\boldsymbol{x}})$
$\tilde{P}_{f,I}(\bar{\boldsymbol{x}})$ from [77]	Incident wave amplitude (complex, a function of position vector, Actran)
$\tilde{P}_{f,S}(\bar{\boldsymbol{x}})$ from [77]	Scattered wave amplitude (complex, a function of position vector, Actran)
$\hat{P}_{fs}$	The scattered acoustic pressure amplitude at the response point

$\hat{P}_{fi}$	The incident acoustic pressure amplitude at the response point
$\hat{P}_{REF}$	Reference pressure amplitude
$\{\tilde{p}_f\}^P$ from [75]	Fluid loading vector (concentrated pressure-conjugate load, inner domain, Abaqus)
$\{\tilde{p}_f\}^Q$ from [75]	Fluid loading vector (boundary domain, Abaqus)
$\{\tilde{P}_f\}_I^Q$ from [75]	Incident fluid loading vector (boundary domain, Abaqus)
$\{\tilde{P}_f\}_S^Q$ from [75]	Scattered fluid loading vector (boundary domain, Abaqus)
$\{\tilde{p}_s\}^N$ from [75]	Structural loading vector (direct boundary traction to structure, complex, Abaqus)
$v_f(\bar{\boldsymbol{x}},t)$ from [64]	Harmonic fluid velocity (a function of position vector and time)
$\tilde{V}_f(\bar{\boldsymbol{x}})$ from [68]	Fluid velocity amplitude (complex, a function of position vector)
$\hat{V}_f(\bar{\boldsymbol{x}})$ from [68]	The modulus of $\tilde{V}_f(\bar{\boldsymbol{x}})$
$\tilde{V}_0$ from [69]	The velocity amplitude of pulsating sphere (complex)
$u_s(\bar{\boldsymbol{x}},t)$ from [75]	Harmonic structure displacement (a function of position vector and time)
$u_f(\bar{\boldsymbol{x}},t)$ from [75]	Harmonic fluid displacement (a function of position vector and time)
$\tilde{U}_f(\bar{\boldsymbol{x}})$ from [75]	Fluid displacement amplitude (complex, a function of position vector)
$\tilde{U}_s(\bar{\boldsymbol{x}})$ from [75]	Structure displacement amplitude (complex, a function of position vector)
$\{\tilde{U}_s\}^M$ from [75]	Structural displacement amplitude vector (inner domain, Abaqus)
$z(\bar{\boldsymbol{x}})$ from [65]	Acoustic impedance (a function of position vector)
$t(\bar{\boldsymbol{x}})$ from [75]	Surface traction (structural boundary, Abaqus)
$\tilde{\psi}_f(\bar{\boldsymbol{x}})$	Fluid velocity potential (a function of the position vector, Actran)

θ_i from [75]	Physical fields (Abaqus)
$\gamma(\bar{\boldsymbol{x}},\theta_i)$ from [75]	Volumetric drag (a function of the position vector and physical fields, Abaqus)
$1/k_1$ from [75]	Fluid pressure-acceleration relation factor (Abaqus)
$1/c_1$ from [75]	Fluid pressure-velocity relation factor (Abaqus)
$1/a_1$ from [75]	Fluid pressure-displacement relation factor (Abaqus)
$\widetilde{T}(\bar{\boldsymbol{x}})$ from [75]	A general boundary traction amplitude (complex, a function of position vector, Abaqus)
$\widetilde{T}_{\mathrm{ft}}(\bar{\boldsymbol{x}})$ from [75]	Boundary traction amplitude (complex, enforced boundary, a function of position vector, Abaqus)
$\widetilde{T}_{\mathrm{fr}}(\bar{\boldsymbol{x}})$ from [75]	Boundary traction amplitude (complex, reactive boundary, a function of position vector, Abaqus)
$\widetilde{T}_{\mathrm{fi}}(\bar{\boldsymbol{x}})$ from [75]	Boundary traction amplitude (complex, radiating boundary, a function of position vector, Abaqus)
$\widetilde{T}_{\mathrm{fs}}(\bar{\boldsymbol{x}})$ from [75]	Boundary traction amplitude (complex, fluid-structure interface, a function of position vector, Abaqus)
$\widetilde{T}_{\mathrm{frs}}(\bar{\boldsymbol{x}})$ from [75]	Boundary traction amplitude (complex, mixed boundary, a function of position vector, Abaqus)
$\left\{\widetilde{T}_s\right\}_I^M$ from [75]	Incident structural traction vector (inner domain, Abaqus)
$\left\{\widetilde{T}_s\right\}_S^M$ from [75]	Scattered structural traction vector (inner domain, Abaqus)
c from [66]	Sound speed
k from [67]	Wave number
λ from [67]	Wavelength
r from [68]	The radius of a spherical wave
$\widetilde{A}$ from [68]	Source pressure amplitude (complex)
$\hat{A}$ from [68]	Modulus of $\widetilde{A}$
$\widetilde{A}^H$ from [68]	Complex conjugate of $\widetilde{A}$

A_n from [77]	Boundary admittance (Actran)
$\vec{M}$ from [77]	Unifor mean flow velocity vector (Actran)
$\tilde{Q}$ from [69]	The source strength of pulsating sphere (complex)
$\hat{Q}$ from [69]	The modulus of $\tilde{Q}$
$\tilde{Q}^H$ from [69]	The complex conjugate of $\tilde{Q}$
I from [68]	Soundwave intensity
$\bar{\boldsymbol{n}}^-$ from [75]	Inward-direction vector (normal to fluid boundary)
$\Delta\boldsymbol{\varepsilon}$ from [75]	Strain tensor variation (Abaqus)
$\boldsymbol{\sigma}_0$ from [75]	Stress tensor (base state, Abaqus)
$\delta\boldsymbol{u}_S$ from [75]	Virtual displacement vector (structure, Abaqus)
$\delta\boldsymbol{\varepsilon}$ from [75]	Virtual strain tensor (Abaqus)
δp from [75]	Virtual acoustic pressure (Abaqus)
α_c from [75]	Mass-proportional damping factor (Abaqus)
β_c from [75]	Stiffness-proportional damping factor (Abaqus)
$\boldsymbol{D}^{el}$ from [75]	Material elasticity matrix (Abaqus)
H^P from [75]	The interpolation function of fluid elements (inner DOFs, Abaqus)
H^Q from [75]	The interpolation function of fluid elements (outer DOFs, Abaqus)
$\boldsymbol{N}^N$ from [75]	The interpolation function of structural elements (inner DOFs, Abaqus)
$\boldsymbol{N}^M$ from [75]	The interpolation function of structural elements (outer DOFs, Abaqus)

$\boldsymbol{\beta}^N$ from [75]	The interpolation function of strain tensor (body DOFs, Abaqus)
$\partial\boldsymbol{\beta}^N/\partial\{\tilde{U}_s\}^M$ from [75]	The interpolation function of stress tensor (body to surface DOFs, Abaqus)
$[A]$ from [29]	A general matrix (Nastran)
$[L]$ from [29]	Lower-triangular matrix (Nastran)
$[U]$ from [29]	Upper-triangular matrix (Nastran)
$[D]$ from [29]	Diagonal matrix (Nastran)
$[PP]$ from [30]	Precondition matrix (Nastran)
$\{x\}$ from [30]	A general response vector (Nastran)
$\{b\}$ from [30]	A general load vector (Nastran)
$\{\mathrm{y}\}$ from [30]	Intermediate results vector (Nastran)
$\{res\}$ from [30]	Residual vector (Nastran)
$\{pp\}$ from [30]	Iteration direction vector (Nastran)
$Err(x)$ from [30]	Iteration error function (Nastran)
$\varepsilon_{\text{thresh}}$ from [30]	Iteration threshold (Nastran)
$\alpha\alpha$ from [30]	Iteration distance (Nastran)
$\beta\beta$ from [30]	Iteration intermediate parameter (Nastran)
$f_{\text{EXP}}\,(\text{Hz})$	Experimental eigenfrequency
$f_{\text{OD}}\,(\text{Hz})$	Numerical eigenfrequency from the “Ordinary Method”, Nastran, Abaqus, Actran
$f_{\text{ACMS}}\,(\text{Hz})$	Numerical eigenfrequency from the “ACMS Method”, Nastran
$f_{\text{SESET}}\,(\text{Hz})$	Numerical eigenfrequency from the “List Superelement Method”, Nastran

$f_{\mathrm{CSUPER}}\,(\mathrm{Hz})$	Numerical eigenfrequency from the “Image Superelement Method”, Nastran
$f_{\mathrm{AMS}}\,(\mathrm{Hz})$	Numerical eigenfrequency from the “AMS Method”, Abaqus
$\mathrm{ERR}_{\mathrm{OD}}\,(\%)$	Relative error of the “Ordinary Method”, Nastran, Abaqus, Actran
$\mathrm{ERR}_{\mathrm{ACMS}}\,(\%)$	Relative error of the “ACMS Method”, Nastran
$\mathrm{ERR}_{\mathrm{SESET}}\,(\%)$	Relative error of the “List Superelement Method”, Nastran
$\mathrm{ERR}_{\mathrm{CSUPER}}\,(\%)$	Relative error of the “Image Superelement Method”, Nastran
$\mathrm{ERR}_{\mathrm{AMS}}\,(\%)$	Relative error of the “AMS Method”, Abaqus

Abbreviations

FEM	Finite element method
DOFs	Degrees of freedom
DOF	Degree of freedom
ACMS	Automated component modes synthesis (Nastran)
AMS	Automatic multi-level substructure (Abaqus)
DMAP	Direct matrix abstracting program (Nastran)
NDDL	Nastran data definition language (Nastran)
ADF	Acoustic double fishnets
RAM	Random access memory
DBsets	Database sets
GUI	Graphical user interface
CAD	Computer-aided design
CMS	Component modes synthesis
FRF	Frequency response functions
M. cfFreq	“Classic-Frequency-Dependent Material” pattern (Nastran)
M. veFreq	“Viscoelastic Material” pattern (Nastran)
M. NFreq	Non-frequency-dependent material
M. Freq	Frequency-dependent material
M. MFreq	Modal-frequency-dependent material
PS	Physical space
PSFRF	Physical-space-based frequency response functions
MS	Modal space
MSFRF	Modal-space-based frequency response functions
MSFRF_ACMS	Modal-space-based frequency response functions, “ACMS Method”.
MSFRF_AMS	Modal-space-based frequency response functions, “AMS Method”.

PSFRF_OD	Physical-space-based frequency response functions, “Ordinary Method”.
MSFRF_OD	Modal-space-based frequency response functions, “Ordinary Method”.
Pxxx (such as P25)	Panel xxx (xxx can be 25, 100, 225, or 400, which means the number of cells of the panel)
Pxxf_PSFRF_OD	Panel xxx, frequency-dependent material, physical-space-based frequency response functions, “Ordinary Method”, Actran.
Pxxxcf_PSFRF_OD	Panel xxx, “Classic-Frequency-Dependent Material” pattern, physical-space-based frequency response functions, “Ordinary Method”, Nastran.
Pxxxve_PSFRF_OD	Panel xxx, “Viscoelastic Material” pattern, physical-space-based frequency response functions, “Ordinary Method”, Nastran.
Pxxxcf_MSFRF_OD	Panel xxx, “Classic-Frequency-Dependent Material” pattern, modal-space-based frequency response functions, “Ordinary Method”, Nastran.
Pxxxcf_MSFRF_ACMS	Panel xxx, “Classic-Frequency-Dependent Material” pattern, modal-space-based frequency response functions, “ACMS Method”, Nastran.
Pxxxve_PSFRF_SESET	Panel xxx, “Viscoelastic Material” pattern, physical-space-based frequency response functions, “List Superelement Method”, Nastran.
Pxxxve_PSFRF_CSUPER	Panel xxx, “Viscoelastic Material” pattern, physical-space-based frequency response functions, “Image Superelement Method”, Nastran.
PxxxMSFRF_OD	Panel xxx, non-frequency-dependent material, modal-space-based frequency response functions, “Ordinary Method”, Nastran, Abaqus.
PxxxMSFRF_ACMS	Panel xxx, non-frequency-dependent material, modal-space-based frequency response functions, “ACMS Method”, Nastran.
PxxxMSFRF_AMS	Panel xxx, non-frequency-dependent material, modal-space-based frequency response functions, “AMS Method”, Abaqus.

PxxxMSFRF_SESET	Panel xxx, non-frequency-dependent material, modal-space-based frequency response functions, “List Superelement Method”, Nastran.
PxxxMSFRF_CSUPER	Panel xxx, non-frequency-dependent material, modal-space-based frequency response functions, “Image Superelement Method”, Nastran.
PxxxPSFRF_OD	Panel xxx, non-frequency-dependent material, physical-space-based frequency response functions, “Ordinary Method”, Nastran, Abaqus.
PxxxPSFRF_SESET	Panel xxx, non-frequency-dependent material, physical-space-based frequency response functions, “List Superelement Method”, Nastran.
PxxxPSFRF_CSUPER	Panel xxx, non-frequency-dependent material, physical-space-based frequency response functions, “Image Superelement Method”, Nastran.
Nastran_PxxxPSFRF_OD	Panel xxx, non-frequency-dependent material, physical-space-based frequency response functions, “Ordinary Method”, Nastran.
Abaqus_PxxxPSFRF_OD	Panel xxx, non-frequency-dependent material, physical-space-based frequency response functions, “Ordinary Method”, Abaqus.
Nastran_Pxxxcf_PSFRF_OD	Panel xxx, “Classic-Frequency-Dependent Material” pattern, physical-space-based frequency response functions, “Ordinary Method”, Nastran.
Actran_Pxxxf_PSFRF_OD	Panel xxx, frequency-dependent material, physical-space-based frequency response functions, “Ordinary Method”, Actran.
P. Melamine	Panel with pure melamine foam
P. AL4	Metamaterial panel, the inner diameter of the aluminium bar is 4mm.
P. AL8	Metamaterial panel, the inner diameter of the aluminium bar is 8mm.
P. AL12	Metamaterial panel, the inner diameter of the aluminium bar is 12mm.

P. AL16	Metamaterial panel, the inner diameter of the aluminium bar is 16mm.
FP	Fluid pressure
SPL	Sound pressure level
STL	Sound transmission loss
1 Layer P400 AL8mm	1 Layer panel with 400 cells, first constraint condition, the inner diameter of the aluminium bar is 8mm.
1 Layer P400.CONS2 AL8mm	1 Layer panel with 400 cells, second constraint condition, the inner diameter of the aluminium bar is 8mm.
1 Layer P400.CONS3 AL8mm	1 Layer panel with 400 cells, third constraint condition, the inner diameter of the aluminium bar is 8mm.
1 Layer P400.CONS4 AL8mm	1 Layer panel with 400 cells, fourth constraint condition, the inner diameter of the aluminium bar is 8mm.
1 Layer Melamine	1 Layer panel, first constraint condition, consisting of pure melamine foam.
1 Layer Melamine CONS2	1 Layer panel with 400 cells, second constraint condition, consisting of pure melamine foam.
1 Layer Melamine CONS3	1 Layer panel with 400 cells, third constraint condition, consisting of pure melamine foam.
1 Layer Melamine CONS4	1 Layer panel with 400 cells, fourth constraint condition, consisting of pure melamine foam.
1 Layer P400 AL4mm	1 Layer panel with 400 cells, first constraint condition, the inner diameter of the aluminium bar is 4mm.
1 Layer P400 AL12mm	1 Layer panel with 400 cells, first constraint condition, the inner diameter of the aluminium bar is 12mm.
1 Layer P400 AL16mm	1 Layer panel with 400 cells, first constraint condition, the inner diameter of the aluminium bar is 16mm.
1 Layer Aluminium	1 Layer panel, first constraint condition, consisting of pure aluminium alloy.
2 Layers P400 AL8mm	2 Layers panel with 400 cells, first constraint condition, the inner diameter of the aluminium bar is 8mm.
3 Layers P400 AL8mm	3 Layers panel with 400 cells, first constraint condition, the inner diameter of the aluminium bar is 8mm.

Chapter 1

Paper Tours

1.1 Theory Tours

1.1.1 Theory Tour I

The first theoretical content introduces the fundamental concepts about potential tasks and available tools, and chapter **2** gives correlated elaborations. Specifically, section **2.1** introduces the current situation and tendency of acoustic metamaterial development through several representatives. The content of this part is mainly about the effective mass density and bulk modulus of these acoustic metamaterials, which are dynamically changed. Based on simulation and experiment results, negative dynamic mass density and bulk modulus are correlated with resonance-antiresonance phenomena. Thus, studying the resonance-antiresonance effects is a good way to prompt acoustic metamaterial design. Therefore, section **2.2** discusses the resonance-antiresonance issues about the acoustic metamaterial that periodically consists of meta-toms. It is supposed that controlling the gradient in a meta-atom and the gradient between meta-atoms can control the compounded mechanical properties of the acoustic metamaterial. In this way, rebuilding the compounded mass density and bulk modulus can change the values of natural frequency, and reconstructing the structural gradients can change the emergence and distribution of natural frequencies. If we want to enhance resonance-antiresonance effects at a low-frequency range, we can build the high value of compounded mass density and the low value of bulk modulus. Therefore, the meta-atom introduced in this paper consists of melamine foam and aluminium alloy. The melamine foam is light and soft, while the aluminium alloy is heavy and stiff. Thus, their compounding forms may enhance resonance-antiresonance effects at a low-frequency range.

For studying resonance-antiresonance phenomena, eigenvalue extraction analyses and steady-state dynamic analyses are two necessary procedures. These two procedures can construct a primary concept of the acoustic metamaterial's dynamic properties. After that, steady-state vibro-acoustic analyses can test the acoustic performance of the acoustic

metamaterial. These three analysis types are methods in this paper to verify the designs of acoustic metamaterials. For executing simulations, FEM software is the best choice. Section **2.3** briefly elaborates on simulation strategy designs and introduces three available FEM software. In addition, FEM software may be the only choice for supporting mechanical simulations, and section **2.4** gives the reason.

Section **2.4** indicates that simulation software must be able to manage the intermediate data automatically. And the intermediate data control requires the solving system to have a predefined logical layout. Under the predefined logical layout, the lifetime of every intermediate data is known. An intermediate data may experience the state of generation, dispatch, storage, and elimination in its lifetime. And proper intermediate data flow must be controlled to match the solution sequences in the simulation process. In simulation software, people cannot write their algorithm but customize the solution sequences that the software completely predefined. In comparison, people can write their algorithm through mathematical software or self-build codes, but these two tools cannot automatically process the intermediate data. In a word, the FEM software is indispensable in simulations.

Modern FEM software has its specific inner contents and outer configurations. In the first place, section **2.5** uses Nastran as an example to introduce the inner contents of FEM software. These inner contents include functional modules, a database, and an executive system. And these three contents are the main parts of a software's solving system. Thus, the core of FEM software is always called the solver. At the same time, section **2.6** introduces the normal configuration of modern FEM software, and section **2.7** introduces the configuration instances of Nastran, Abaqus and Actran.

1.1.2 Theory Tour II

The second theoretical content elaborates on available simulation strategies and their solving efficiencies based on characteristics of specific FEM software. This part includes three chapters: **3**, **4**, and **5**.

Chapter **3** stretches the arguments of available simulation strategies and their efficiencies in Nastran. And these arguments gradually emerge with the introduction of simulation procedures in Nastran. Since superelement correlated techniques are the forte of Nastran in processing massive DOFs and periodic structures, the theory of the superelement method and how it works are inserted in the introduction of Nastran.

Learning introductions in sections **3.1** and **3.2**, we know that users can write their simulation plan into an input file and use it to drive simulation practices through Nastran's solving system. The input file can be subdivided into five subsections. Once the input file activates the solving system, the lifetime of a simulation practice in Nastran can be subdivided into five stages; the first four are the simulation stages, and the last is the postprocessing stage. In the four simulation stages, "Phase 0" is the data-input stage, "Phase I" and "Phase II" are the solving stages, and "Phase III" is the data-output stage. And these four stages are introduced in sections **3.3**, **3.4**, **3.7**, and **3.9**, respectively.

For a simulation practice, the two solving stages are the core. In detail, "Phase I" is a model discretization stage; finite element matrices are generated in this stage, and they will be assembled into global matrices. "Phase I" transforms models' geometry and mechanical information into numerical data. Thus, once the model's geometry and mechanical property

change, “Phase I” will be reactivated to update the model’s numerical data. In “Phase I”, superelement techniques can be compounded with normal solution sequences. We can see that section **3.5** conceptually introduces some applicable superelement techniques in Nastran. Then, section **3.6** specifically elaborates on fundamental theories and executing procedures of the superelement method.

As the second solving stage, “Phase II” can solve matrix equations in physical or modal space, and the matrices in matrix equations come from “Phase I”. The matrices may be reduced if “Phase I” adopts superelement techniques or non-reduced if “Phase I” does not. In modal space, if eigenmodes can uncouple matrices, the matrix equation becomes independent algebraic equations. And the solving process is a simple algebraic calculation. Otherwise, if the matrices are uncoupled, the matrix equations can be solved by specific solvers. Section **3.8** introduces two solvers in Nastran that can solve the uncoupled matrix equation: the “Direct Solver” and the “Iterative Solver”. Based on the fundamental theories of these two solvers, we can learn their pros and cons. Finally, the “Direct Solver” is the choice for Nastran’s simulation in this paper.

After “Phase II” calculations, the system generates solutions of the matrix equations and sends them into “Phase III”. These solutions can be modal results (modal-space-based calculation) or physical results (physical-space-based calculation). And these results may belong to reduced models (with superelement techniques) or non-reduced models (without superelement techniques). If the solutions are modal results, they must be transformed into physical results before further manipulations. If the solutions belong to reduced models, they must be transformed into the ones of non-reduced models before further manipulations. Then, according to requirements, users can customize the output data. The customization includes choices of physical parameters (such as displacements, stress, heat transfers, etc.) and controls of output objects (such as grid output, element output, component output, etc.). And getting these output data requires further calculations based on “Phase II” results. Section **3.9** gives correlated descriptions of manipulations in “Phase III”.

Considering the rules that Nastran executes a simulation, the simulation strategy design mainly concerns the cross effects in “Phase I” and “Phase II”. We must find the best trade-off between cost and saving for an efficient simulation. Then, how to choose proper methods depends on the acoustic metamaterial’s characteristics. Contents in section **3.10** are arguments for simulation strategies and their theoretical efficiencies. We know from the arguments that Nastran has many available methods to treat the acoustic metamaterial in eigenvalue extraction and steady-state dynamic analyses. However, Nastran is not competent in conducting steady-state vibro-acoustic analyses because it cannot model diffuse incident fields. And section **3.11** gives correlated arguments.

Chapter **4** stretches the arguments of available simulation strategies and their efficiencies in Abaqus. Analogous to Nastran, Abaqus conducts simulation practice based on a user-defined input file. The input file contains two parts: the model and history data. Section **4.1** gives correlated descriptions.

Simulations in Abaqus can be executed in physical or modal space, and these two simulation spaces represent two technical routes. Based on technical routes, users can choose proper software architectures. Abaqus developed two software architectures: the traditional architecture and the SIM architecture. Once the modal space is chosen, one of the two software architectures can be a candidate. Otherwise, traditional architecture is the only choice. Section

4.2 briefly introduces the two software architectures and how they influence the available technical routes. Analogous to Nastran, Abaqus can solve matrix equations through the direct sparse solver or the iterative solver, and which solver is used requires users to make a choice.

Subsequently, section **4.3** gives arguments for simulation strategies and their theoretical efficiencies. And this section uses Nastran's simulation strategies as references to talk about availabilities that adopt analogous simulation strategies in Abaqus. After theoretical deductions, Abaqus is not as competent as Nastran. Fortunately, Abaqus can model diffuse incident fields to execute steady-state vibro-acoustic analyses. And this is a necessary condition to test the metamaterial's acoustic performance. Section **4.4** gives correlated arguments.

Chapter **5** stretches the arguments of available simulation strategies and their efficiencies in Actran. Analogous to Nastran and Abaqus, Actran executes simulation practices based on a user-defined input file. Actran's input file contains eight "Data Groups" as subsections. Section **5.1** gives correlated descriptions.

Simulations in Actran can be executed in physical or modal space, and these two simulation spaces represent two technical routes. When choosing different technical routes in Actran, users do not need to consider software architecture but think about how to choose a proper solver. Actran supplies many solvers and lets users judge which solver is proper for corresponding simulation scenarios. Solvers of modal-space-based analyses are different from the ones of physical-space-based analyses. Section **5.2** introduces Actran's technical routes and corresponding solvers.

Subsequently, section **5.3** gives arguments for simulation strategies and their theoretical efficiencies. Then, section **5.4** introduces the capability of Actran to treat acoustic problems and simulate diffuse incident fields. Theoretically, Actran is an expert in acoustic analyses, but its solving system is not as competent as Nastran or Abaqus. Thus, Actran cannot supply very efficient simulation strategies to treat the acoustic metamaterial samples. For example, a modal-space-based analysis needs two calculation cycles in Actran but only one in Nastran or Abaqus. Actran must execute a prior eigenvalue extraction analysis in the first cycle and send the extracted eigenmodes to the modal-space-based analysis in the second cycle. The two-cycle computation of modal-space-based analysis in Actran is a deficiency. However, this deficiency brings an unexpected advantage to Actran. The eigenmodes can be computed through other software, such as Nastran. And Nastran has a very efficient method to execute eigenvalue extraction analyses. Thus, this is a window for joint simulations between Actran and Nastran. Then, the joint simulations can increase the capability of the two software in executing vibro-acoustic analyses.

1.1.3 Theory Tour III

The third theoretical contents introduce numerical damping models and the setting methods in different application scenarios with different software. This part includes four chapters: **6**, **7**, **8**, and **9**. Specifically, chapter **6** introduces fundamental theories of damping. It is a pre-season chapter of the other three chapters. Then, from chapters **7** to **9**, they introduce numerical damping models and usage methods. To some extent, chapters **7**, **8**, and **9** are theory compensations of chapters **3**, **4**, and **5**, respectively.

As introduced before, acoustic metamaterial consists of aluminium alloy and melamine foam. Section **2.2** shows that these two constituent materials have small damping effects. In

addition, melamine foam is a frequency-dependent "Viscoelastic Material"; its anelasticity changes significantly with frequency. It is worth considering the damping effects of aluminium alloy and melamine foam, especially the damping's frequency dependency. In dynamic analyses, damping effects are crucially important to dynamic responses; even the damping value is small.

Chapter **7** elaborates on numerical models of structural damping in Nastran. The content in this chapter mainly talks about damping's setting methods and application scenarios. Specifically, section **7.1** briefly introduces the classifications of damping in Nastran. This section gives gross concepts of damping models. Then, the following sections discuss damping model usage in correlated simulation scenarios. Section **7.2** introduces the usage of material structural damping in physical-space-based dynamic analyses, and the material structural damping can be set as a non-frequency-dependent or frequency-dependent parameter. The damping's frequency dependency can be set through the "Classic Frequency-Dependent Material" or "Viscoelastic Material" pattern. And these two material patterns have their pros and cons. Then, section **7.3** introduces the usage of material structural damping in modal-space-based dynamic analyses. In this case, material structural damping will be transformed into modal structural damping. And the damping can be a non-frequency-dependent or frequency-dependent parameter. However, the damping's frequency dependency can only be set through "Classic Frequency-Dependent Material" patterns. As an extension of section **7.3**, section **7.4** talks about extracting modal damping factors through Nastran's DMAP using the "Composite Modal Damping" method. Computations with the "Composite Modal Damping" method can treat damping models and get approximate results. For frequency-dependent damping models, further approximations are needed. Finally, section **7.5** lists Nastran's specific simulation practices to that corresponding damping models are applied.

Chapter **8** elaborates on numerical models of structural damping in Abaqus. Since Abaqus cannot model frequency-dependent orthotropic material, this chapter's structural damping is considered a non-frequency-dependent parameter. Sections **8.1**, **8.2**, and **8.3** are analogous to sections **7.1**, **7.2**, and **7.3**, respectively, except for the simulation platform and the frequency-dependent damping models. Then, section **8.4** introduces the "Composite Modal Damping" models and their setting method. After that, section **8.5** gives an example of how to set the direct-input modal damping. Finally, section **8.6** lists Abaqus's specific simulation practices to that corresponding damping models are applied.

Chapter **9** elaborates on numerical models of structural damping in Actran. Like Nastran, Actran has predefined material patterns, and properties in the material patterns can be non-frequency-dependent or frequency-dependent parameters. However, the difference is that Actran's material pattern can define complex properties, but Nastran's material pattern cannot. In comparison, Abaqus's material properties are free-state parameters; Abaqus does not have predefined material patterns. Section **9.1** describes how to set complex elasticity to non-frequency-dependent and frequency-dependent models in physical space. After that, section **9.2** gives examples of how to set material structural damping to non-frequency-dependent and frequency-dependent models in physical space. At the same time, it is worth noting that the effects of complex elasticity and material structural damping are equivalent. Finally, we can find that the physical-space-based analyses in Actran are analogous to those in Nastran and Abaqus. However, it is another situation of modal-space-based analyses. A modal-space-based analysis needs two calculation cycles in Actran, while only one in Nastran and Abaqus. Thus, Actran cannot directly transform physical damping into modal damping. The way to add

damping effects to modal-space-based analyses is to input modal damping factors directly. Section **9.3** gives an example of how to set direct-input modal structural damping. Finally, section **9.4** lists Actran's specific simulation practices to that corresponding damping models are applied.

1.1.4 Theory Tour IV

The fourth theoretical contents introduce fundamental theories and numerical models of acoustic fields. This part includes four chapters: **10**, **11**, **12**, and **13**. As one of these four chapters, chapter **10** introduces acoustic governing equations, acoustic sources, and crucial factors of acoustic simulations in different FEM software. It is a pre-season chapter of the other three chapters. Then, from chapters **11** to **13**, they introduce acoustic theories in Abaqus, Actran, and Nastran, respectively.

As we know, Abaqus, Actran and Nastran are three candidates that this paper uses to make simulation strategy designs. Based on a study of software theory handbooks, Abaqus gives a relatively comprehensive explanation of the acoustic theory. In comparison, Actran's acoustic theory introduction is moderate, while Nastran's is limited. Thus, Abaqus's acoustic theory system is the reference of this paper.

Chapter **11** elaborates on Abaqus's acoustic theory background. Section **11.1** introduces the original form of the acoustic filed governing equation and the general expression of boundary tractions in Abaqus. And the derivative of this governing equation is dependent on specific boundary conditions. Then, section **11.2** introduces the mathematical expressions of specific boundary conditions. These specific boundary conditions are crucially important in influencing the force and response of acoustic domains. Based on the mathematical expressions of the governing equation and boundary conditions, a FEM discretization can transform acoustic and structural information into numerical models. Section **11.3** introduces the FEM discretization process, and the products of this discretization are matrix equations. The software's solver can solve these matrix equations. Then, the solving method of a fluid-structure system in steady-state dynamic analyses is based on scattered wave theory. Section **11.4** gives corresponding explanations of how scattered wave theory works. Finally, section **11.5** introduces the acoustic loading system in Abaqus.

Chapter **12** elaborates on Actran's acoustic theory background, and the content in this part is elegant. Sections **12.1**, **12.2**, and **12.3** discuss the Helmholtz Equation and its derivatives. These derivatives are governing equations for different simulation scenarios. Then chapter **13** elaborates on Nastran's acoustic theory background, and the content in this part is limited. Nastran is not an expert in acoustic simulations. It has partial capabilities to solve acoustic problems.

1.2 Simulation Series Tours

The simulation series includes four chapters: **14**, **15**, **16**, and **17**. The task level of simulations in these four chapters is upgraded. At the first task level, the accuracy of different simulation strategies is the main topic. This part relates to chapters **14** and **15**. At the second task level, the efficiency of different simulation strategies is the main topic. This part relates to chapters **15** and **17**. And the simulations in the first and second task levels are in concert with the theoretical

arguments of simulation strategies. Then, it is easy to find the best simulation strategies at the trade-off point considering proper accuracy and efficiency. At this moment, the simulation strategy designs are accomplished. At the third task level, the best simulation strategies are used in corresponding simulation tasks to verify the metamaterial design. This part relates to chapters **15**, **16**, and **17**.

1.2.1 Simulations I

Chapter **14** elaborates on the practical accuracy benchmarks of different numerical methods in Nastran, Abaqus and Actran. The simulation tasks are the eigenvalue extraction analyses of two perforated PVC panels. As for the testing samples, one PVC panel is perforated with 300 holes, and another is perforated with 600 holes. And their eigenvalues have been tested in experiments. Then, the eigenvalues from experiments are the practical reference of simulation results. Section **14.1** lists the simulation results of eigenvalue extraction analyses through the four numerical methods in Nastran. And these four numerical methods are testing objects in this section. The four numerical methods are the "Ordinary Method", the "ACMS Method", the "List Superelement Method", and the "Image Superelement Method". In these four methods, the "ACMS Method", the "List Superelement Method", and the "Image Superelement Method" are three derivatives of superelement techniques in Nastran. The "ACMS Method" needs no extra manipulations to the original model. While the "List Superelement Method" and "Image Superelement Method" require users to partition the original model to define superelements and residual structures before making solutions. Then, section **14.2** lists the simulation results of eigenvalue extraction analyses through the two numerical methods in Abaqus. And these two numerical methods are testing objects in this section. The "SIM-Architecture-Based Ordinary Method" and the "SIM-Architecture-Based AMS Method" are the two numerical methods. It is worth noting that the "SIM-Architecture-Based AMS Method" is a derivative of superelement techniques in Abaqus, and this method needs no extra manipulations to the original model. After that, section **14.3** lists the simulation results of eigenvalue extraction analyses through the "Ordinary Method" in Actran. The "Ordinary Method" is the only testing object in this section. Finally, section **14.4** elaborates on the practical accuracy assessments of the simulation strategies in the three software.

1.2.2 Simulations II

Chapter **15** is more complex than the other three chapters because it must clarify the implicit relations between simulation strategy designs and acoustic metamaterial designs. The simulation tasks in this chapter include two types: eigenvalue extraction analyses and steady-state dynamic analyses. Consequently, the eigenvalue extraction analyses give the distribution of the eigenfrequencies. And the steady-state dynamic analyses give the frequency response functions of acoustic metamaterial samples in the predefined frequency range. Furthermore, it is worth noting that the distributed eigenfrequencies are in concert with the resonance and anti-resonance points of frequency response functions. And the extracted eigenmodes are preconditions of modal-space-based steady-state dynamic analyses.

On the one hand, considering simulation strategy designs, the eigenvalue extraction and steady-state dynamic analyses are the touchstones of steady-state vibro-acoustic analyses. And the indicators of simulation-strategy performance are relative accuracy and efficiency. As one

of the indicators, relative accuracy means the reference data comes from inner simulations instead of practical experiments, and the inner simulations are driven by the most accurate numerical method. As another indicator, the efficiency is measured by time consumption in treating given models with different simulation strategies, and the time consumption is positively associated with the number of DOFs. Then, it is necessary to set a control group in which the testing samples have a different number of DOFs. This paper's control group of testing samples are the four size-controlled acoustic metamaterial panels. They contain a different number of meta-atoms. On the other hand, considering acoustic metamaterial designs, how the number of meta-atoms (or the size of panels) influence dynamic performance can be verified through eigenvalue extraction and steady-state dynamic analyses.

Section **15.1** elaborates on the relative accuracy and efficiency of available simulation strategies in Nastran. The candidate numerical methods in this section are the "Ordinary Method", the "ACMS Method", the "List Superelement Method", and the "Image Superelement Method". In the beginning, subsections **15.1.1** and **15.1.2** describe the partitioning operations of the original model to match the set requirements of the "List Superelement Method" and the "Image Superelement Method", respectively. These two numerical methods will be used in eigenvalue extraction and steady-state dynamic analyses.

The first part of this section is the eigenvalue extraction analysis. Except for the testing panels, these eigenvalue extraction analyses are analogous to those in section **14.1**. The eigenvalues extracted by the "Ordinary Method" is the reference because the "Ordinary Method" is theoretically the most accurate numerical method. Then, the results are illustrated in subsection **15.1.3**.

The second part of this section is the steady-state dynamic analyses of the four testing panels. In this part, the testing objects are the joint simulation strategies considering three factors. The first factor is the four numerical methods, the second is the three material patterns, and the third is the two simulation spaces. Specifically, the four numerical methods in steady-state dynamic analyses are the same as in eigenvalue extraction analyses. Then, the three material patterns are the "Classic Frequency-Dependent Material" pattern, the "Viscoelastic Material" pattern, and the "Non-Frequency-Dependent Material" pattern. At the same time, the two simulation spaces are the physical and modal spaces. Subsection **15.1.4** lists the matching conditions of joint works between the numerical methods, the material patterns, and the simulation spaces. The matching conditions can guide how to make simulation strategies correctly. Then, it is reasonable to fix one factor and let the other two be the control parameters to assess which simulation strategies are the best.

In the first place, when fixing the numerical methods, the control parameters are the two simulation spaces and the three material patterns. In subsection **15.1.5**, the numerical method is the "Ordinary Method" because this one is theoretically the most accurate. Then, the frequency response functions show the influence of the two simulation spaces and the three material patterns on the frequency response functions and the relative accuracies.

In the second place, when fixing the material patterns, the control parameters are the numerical methods and the simulation spaces. In subsection **15.1.6**, the "Classic Frequency-Dependent Material" and the "Viscoelastic Material" patterns correspond with the frequency-dependent melamine foam. Then, the frequency response functions show the influence of the two simulation spaces and the four numerical methods on the frequency response functions and the relative accuracies.

In the third place, when fixing the simulation space and the material pattern, the control parameter is the numerical method. In subsection **15.1.7**, the simulations in physical and modal space are independent, and the material pattern is the "Non-Frequency-Dependent Material". Then, the frequency response functions show the influence of the four numerical methods on the frequency response functions and the relative accuracies. Finally, subsection **15.1.8** summarizes the efficiencies of numerical methods in eigenvalue extraction analyses and the efficiencies of simulation strategies in steady-state dynamic analyses. At the same time, this subsection comments on the accuracy grade of numerical methods in eigenvalue extraction analyses and simulation strategies in steady-state dynamic analyses.

Section **15.2** elaborates on the relative accuracy and efficiency of available simulation strategies in Abaqus. The candidate numerical methods in this section are the "Ordinary Method" and the "AMS Method". As the first part of this section, subsection **15.2.1** illustrates the results of eigenvalue extraction analyses of the four testing panels. And the results from the "Ordinary Method" is the reference. Then, before discussing the joint simulation strategies in steady-state dynamic analyses, subsection **15.2.2** lists the matching conditions of the two numerical methods and the two simulation spaces. And the matching conditions can guide how to make simulation strategies correctly. It is worth noting that the material pattern in this section is non-frequency-dependent since Abaqus cannot model frequency-dependent orthotropic materials. Thus, the frequency response functions in subsection **15.2.3** show the influence of the two simulation spaces and the two numerical methods on the frequency response functions and the relative accuracies. Finally, subsection **15.2.4** summarizes the efficiencies of numerical methods in eigenvalue extraction analyses and the efficiencies of simulation strategies in steady-state dynamic analyses. At the same time, this subsection comments on the accuracy grade of numerical methods in eigenvalue extraction analyses and simulation strategies in steady-state dynamic analyses.

Section **15.3** elaborates on the relative accuracy and efficiency of available simulation strategies in Actran. The solely numerical method in this section is the "Ordinary Method" because Actran cannot supply proper superelement correlated techniques. As the first part of this section, subsection **15.3.1** illustrates the results of eigenvalue extraction analyses of the four testing panels. Then, subsection **15.3.2** lists the sole simulation strategy: the physical-space-based analysis with the "Ordinary Method", and the object is the normal frequency-dependent model in Actran. After that, subsection **15.3.3** shows the frequency response functions of frequency-dependent models through the physical-space-based steady-state dynamic analyses with the "Ordinary Method". Finally, subsection **15.3.4** lists the number of extracted eigenfrequencies, the time consumption of the eigenvalue extraction and steady-state dynamic analyses.

Section **15.4** elaborates on the relative accuracy and efficiency of available simulation strategies between Nastran, Abaqus, and Actran. The first part discusses the eigenvalue extraction analyses, and Nastran's "ACMS Method" is the most efficient. The second part talks about steady-state dynamic analyses. When the analysing objects are non-frequency-dependent models, the baseline simulation strategy of Nastran and Abaqus is the physical-space-based analyses with the "Ordinary Method". At the same time, when the analysing objects are frequency-dependent models, the baseline simulation strategy of Nastran and Actran is also the physical-space-based analyses with the "Ordinary Method". It is worth noting that the baseline simulation strategy is the most accurate in its software platform, and the discrepancies of the results from the baseline simulation strategies in different software mean that the baseline

accuracy in different software is different. As for efficiency, modal-space-based analyses are always higher than the physical-space-based analyses in treating frequency-dependent or non-frequency-dependent models.

1.2.3 Simulations III

Chapter **16** purely elaborates on metamaterial design. The topic in this chapter is the eigenfrequencies distribution of metamaterial panels. The panels' size is fixed, and the control parameter is aluminium alloy inclusions. The content in this chapter is very simple. It is the extension of the eigenvalue extraction analyses of chapter **15**. And the numerical method adopted in this chapter is Nastran's "ACMS Method".

1.2.4 Simulations IV

Chapter **17** holds the ultimate purpose of this paper: finding the best simulation strategy to verify the acoustic metamaterial design. The first part of this chapter is technical preparations for finding the best simulation strategy for steady-state vibro-acoustic analyses, and this part includes sections **17.1** and **17.2**. Specifically, section **17.1** theoretically elaborates on available simulation strategies from Nastran, Abaqus, and Actran for executing the steady-state vibro-acoustic analyses. Then, section **17.2** arranges corresponding tests to verify which simulation strategy is the best. Finally, Nastran cooperates Actran is the best choice. In detail, Nastran executes eigenvalue extraction analyses through the "ACMS Method" and generates eigenmodes and modal damping as the first step. Then, Actran uses the eigenmodes and modal damping from Nastran to execute modal-space-based steady-state vibro-acoustic analyses as the second step. All simulations in section **17.3** adopt this simulation strategy.

In the second part, the main content of this chapter is steady-state vibro-acoustic analyses around different types of metamaterial samples. Section **17.3** contains three control groups of acoustic metamaterial samples and one compensation group of pure melamine foam with different control parameters. As already explained, eigenvalues extraction and steady-state dynamic analyses in chapters **15** and **16** are touchstones of steady-state vibro-acoustic analyses.

Simulations in subsection **17.3.1** process the first control group of metamaterial samples, and the control parameter of this group is the panels' lateral-boundary-constrained conditions. The simulations in the first control group are inspired by the tests of metamaterial samples with different sizes in chapter **15**. The reason is that the lateral-boundary constraints are analogous to subdivisions of the panel, and different lateral-boundary-constrained conditions transform the panel into subparts with different sizes. Then, as compensation for the first control group, simulations in subsection **17.3.2** process the samples of pure melamine foam with the same constraint conditions as in subsection **17.3.1**. The sample of the compensation group is the melamine foam with different constraint conditions. And the test of the compensation group can isolate the effects of the constraint conditions and aluminium bar inclusions.

Subsequently, simulations in subsection **17.3.3** process the second control group of metamaterial samples, and the control parameter of this group is the aluminium inclusions. The simulations in the second control group are inspired by the tests of metamaterial samples with different aluminium bar inclusions in chapter **16**. We know that the aluminium bar inclusions influence the distribution of the eigenfrequencies in the given frequency range. And the

eigenfrequencies dominate the resonance-antiresonance effects. Thus, metamaterial samples with different aluminium bar inclusions have different dynamic performances.

Lastly, simulations in subsection **17.3.4** process the third control group of metamaterial samples, and the control parameter of this group is the number of panel layers. In this control group, the total thickness of intermediate samples is fixed, but the panel layers of intermediate samples are changed; from one layer to three layers. In this case, different layers can interfere with each other. The reason is that acoustic waves reaching different layers have phase lags due to the gap between layers. These phase lags of acoustic waves introduce staggers to resonance and anti-resonance of different layers.

In the third part, section **17.4** elaborates on theoretical explanations of the results from the vibro-acoustic analyses in section **17.3**. Specifically, subsection **17.4.1** lists the time consumption of the modal-space-based vibro-acoustic analyses of fourteen fluid-structural systems. And these time consumptions consist of two parts. One relates to eigenvalue extraction analyses, and another relates to vibro-acoustic analyses in modal space. Then, subsection **17.4.2** elaborates on eigenfrequencies and their distributions of the fourteen fluid-structure systems. This part is the forecasting indicator of the system's dynamic performance. Subsequently, subsection **17.4.3** elaborates on the low-frequency sensibilities and soundproof capabilities of the metamaterial samples based on the frequency response functions from section **17.3**. Then, as the theoretical extensions, **17.4.4** and **17.4.5** give reasonable explanations of low-frequency sensibility and soundproof capability theories.

Chapter 2

Metamaterial and Software

2.1 Brief Introductions of Existing Acoustic Metamaterials

In a sound transmission process, wave propagations are controlled by the media's mass density ρ and bulk modulus K, including acoustic and structural media. The values of these two parameters are naturally positive. However, in a dynamic process, the motivation in acoustic media may be in phase or out-off phase regarding the motivation in structural media. When the motivation of structural media is out of phase, it will reject acoustic wave transduce, and this phenomenon can be seen as acoustic wave reflection. The key point is to control structural media's motivation artificially. Fortunately, breakthroughs may happen in a new category of materials. They are conceptually distinguished from traditional acoustic materials and are called acoustic metamaterials. Metamaterials are artificial structures, typically periodic (but not necessarily so), composed of small meta-atoms that, in bulk, behave like a continuous material with unconventional effective properties. By designing and engineering artificial materials with more complex properties, unprecedented functionalities can be obtained. Through years of development, acoustic metamaterials can dynamically control the effective mass density ρ_{eff} and bulk modulus K_{eff}. These two parameters may be equivalent to zero or negative at specific vibration frequencies or in frequency bands. The upper arguments are based on [1]–[3].

Seven representative acoustic metamaterials with specific dynamic performances have been developed. They are classified into three groups regarding their effective mass density and bulk modulus. Metamaterials in the first group have negative effective mass density; they are introduced in subsections **2.1.1**, **2.1.2**, and **2.1.3**. The metamaterial introduced in subsection **2.1.1** is called the local resonator. The metamaterials introduced in subsections **2.1.2** and **2.1.3** can be called cavity-membrane resonators. Then, metamaterials in the second group have effective negative modulus; they are introduced in subsections **2.1.4** and **2.1.5**. And these two metamaterials are in cavity-structure configurations with specific layouts of a series of resonators. Finally, metamaterials in the third group simultaneously have negative effective mass density and bulk modulus, introduced in subsections **2.1.6** and **2.1.7**.

2.1.1 Locally Resonant Sonic Material

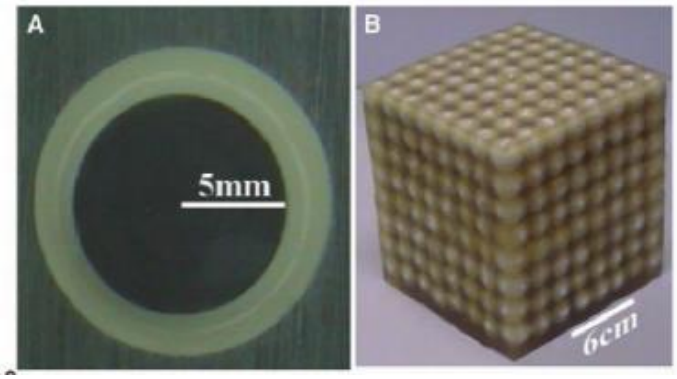


Figure 2.1: The locally resonant sonic material [4].

One of the popular configurations of acoustic metamaterials is an artificial periodic structure that comprises resonant units as meta-atoms. The study of this material configuration started in 2000. It is a new artificially compounding structure: an epoxy cube containing well-arranged lead balls coated with silicone rubber. **Figure 2.1** shows the specific configuration. This material is called "Locally Resonant Sonic Material". When these inertial metamaterials are excited at the natural frequency, the core strongly oscillates out of phase with the driving force, giving rise to a negative mass density. However, the obvious drawback of the locally resonant sonic material is that it can function well only in a limited range of frequencies. This metamaterial introduced here is based on [4].

2.1.2 Membrane with an Additional Central Mass

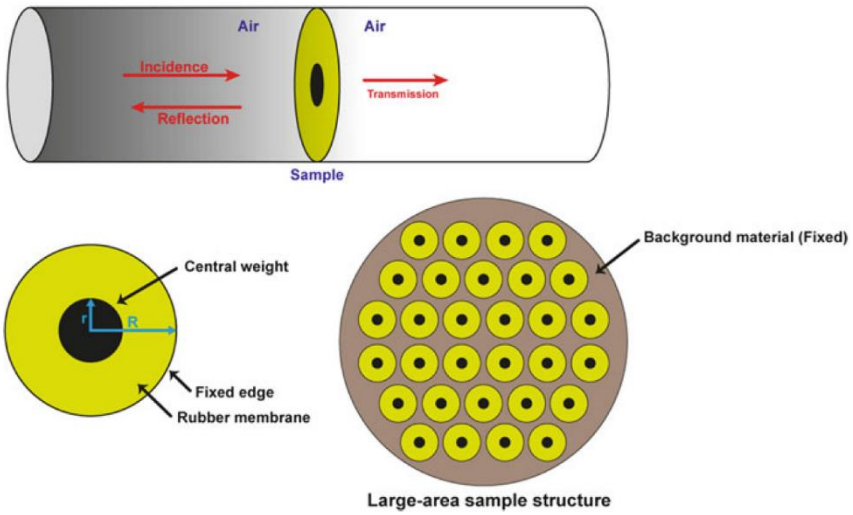


Figure 2.2: Membrane-type acoustic metamaterial with additional centre mass [2].

The first cavity-membrane resonator uses a rigid grid tube to fix the edge of a slightly stretched rubber membrane, and a small weight attaches to the centre of the membrane. **Figure 2.2** shows the specific configuration. Adjusting the value of the central weight with this configuration can tune the oscillating frequency. Then, stacking many layers of the membrane can broaden the frequency range. This scheme mitigates noise by resonance-antiresonance effects, and the negative effective mass density is detected at the antiresonance points. This metamaterial introduced here is based on [2].

2.1.3 Thin Membrane without an Additional Central Mass

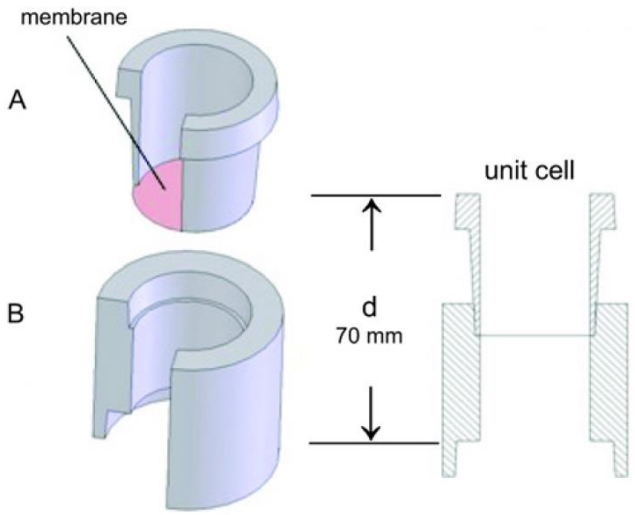


Figure 2.3: Membrane-type acoustic metamaterial without additional centre mass [5].

The second cavity-membrane resonator uses a rigid circle tube to fix the edge of a very thin membrane without additional central weight. **Figure 2.3** shows the specific configuration. The membrane is very slight, and its on-site tension is measured in the experiment. Concatenating several circle tubes, confined spaces with air separated by membranes appear. The air in a unit tube functions as a mass, while the membrane functions as a spring. It is like a resonator. Due to the special arrangement, this configuration can block acoustic waves below a critical frequency. Under this frequency, the effective mass density is negative. This metamaterial introduced here is based on [5].

2.1.4 Waveguide with Helmholtz Resonators

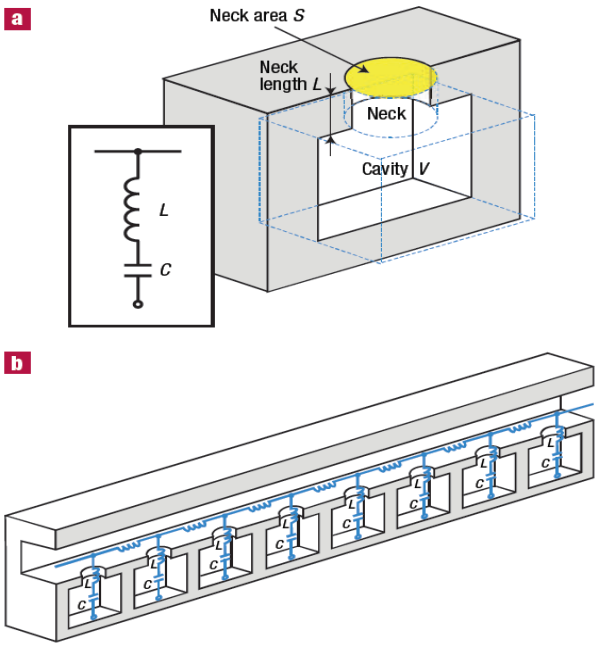


Figure 2.4: Helmholtz-resonator-compounded metamaterial [6].

The main configuration of this metamaterial is a waveguide tube that connects Helmholtz resonators with narrow channels. **Figure 2.4** shows the specific configuration. As the meta-atom, a Helmholtz resonator is a closed cavity with a narrow open channel (**Figure 2.4** (a)). Then, this ultrasonic metamaterial consists of an array of subwavelength Helmholtz resonators. And these Helmholtz resonators have designed acoustic inductance and capacitance. This metamaterial has an effective dynamic modulus with negative values near the resonance frequency. In addition, changing the cavity volume changes the resonance frequencies. This metamaterial introduced here is based on [6].

2.1.5 A Waveguide with Side Holes

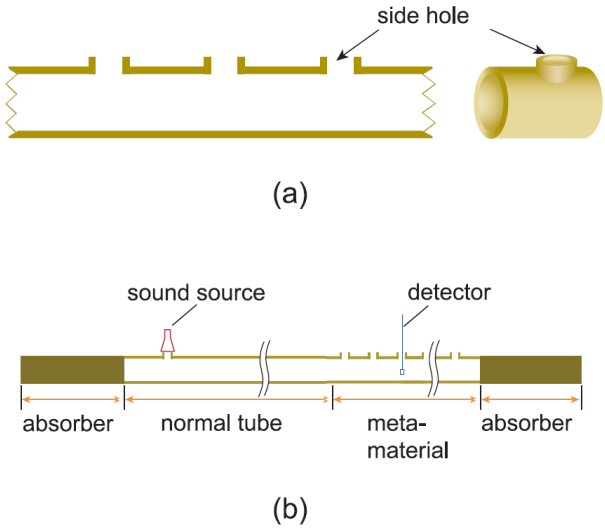


Figure 2.5: Side-hole-compounded metamaterial [7].

The metamaterial is fabricated as a waveguide with side holes, as shown in **Figure 2.5**. This metamaterial exhibits an effective negative modulus in the frequency range from zero to a cut-off frequency. And the frequency characteristics are the same as that of the metallic permittivity. This metamaterial introduced here is based on [7].

2.1.6 Cavity-Membrane Tubes with Side Holes

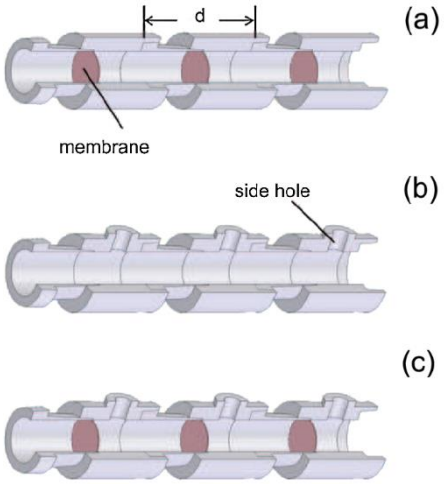


Figure 2.6: Hole-membrane-compounded metamaterial [8].

Based on the resonate units, a new metamaterial with negative effective density and modulus has been found. The meta-atom of this metamaterial is a side-hole tube with a central membrane (**Figure 2.6** (c)). It seems like a combination of a side-hole tube and a central-membrane tube in configuration, and their effective properties are complementary. **Figure 2.6** shows the specific configuration. This metamaterial introduced here is based on [8].

2.1.7 Multi-Dimensional Metamaterials

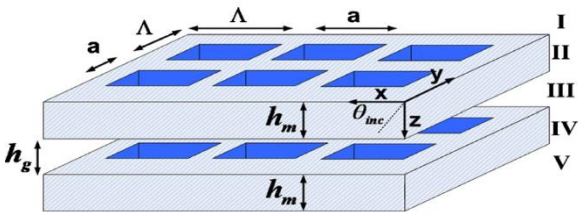


Figure 2.7: Acoustic-double-fishnet metamaterial [9].

The metamaterials introduced in subsections **2.1.2** to **2.1.6** can be called one-dimensional acoustic impeding metamaterials. They can effectively impede the axial incoming waves but weakly mitigate the incoming waves with off-normal incident angles. Besides, a new type of metamaterials called acoustic double fishnets (ADF) emerged to conquer the degradation of noise mitigation due to off-normal incident angles. **Figure 2.7** shows the specific configuration. And this type of metamaterial has negative effective density and modulus. This metamaterial introduced here is based on [9].

2.2 Brief Introductions of Acoustic Metamaterial Designs

According to the upper introduction about representative acoustic metamaterials, resonance-antiresonance effects are the main mechanism they rely on to attenuate sound propagation. Except for the "Locally Resonate Sonic Material", resonance effects of introduced metamaterials are dominated by resonant units (meta-atoms). These resonant units consist of solid structures with specific shapes, and their properties are affected by cavity-membrane or cavity-structure interactions. For these resonant units, fluid material properties, cavity-structure interactions, or cavity-membrane interactions highly participate and intensify resonance-antiresonance effects. In contrast, structural material properties contribute little to resonance-antiresonance effects. As we know, resonance intensifies vibrations while antiresonance diminishes vibrations. In acoustic wave propagations, the resonance effect is an impeller, while the antiresonance effect is a defender. Supposing people want to intensify acoustic wave propagations, they must try to prompt the resonance and hide the antiresonance. Then, supposing people want to diminish acoustic wave propagations, they must try to prompt the antiresonance and hide the resonance. However, isolating resonance and antiresonance is difficult. And the upper-introduced acoustic metamaterials are hard to extend due to their specific configurations. Thus, simplifying the configuration is one thing that needs to be

considered in new acoustic metamaterial design. And solely considering solid structure is a promising way.

For a solid structure, the resonant frequencies correspond to eigenfrequencies, and damping effects can shift resonant points or even diminish resonance. Since the damping of solid materials is always slight, it can moderately affect resonance and slightly consume energy. Then, an effective route for controlling the resonance is to control eigenfrequencies distribution. The eigenfrequencies distribution correlates with the structures' material property and geometry shape. The way to control the range of eigenfrequencies is to control the mass density and elastic moduli. Then, composite materials and constraint conditions are effective since they can adjust the original material's mechanical properties. At the same time, the way to intensify the emergence of eigenfrequencies is to complicate the material's inner structure. Then, periodic configuration based on unit cells is effective since it can increase the gradient of the inner complexity of the global structure. Furthermore, a substructure-compounded global structure may have unprecedented features once the substructure size is lower than the subwavelength threshold (at least ten times lower than the wavelength) [3].

2.2.1 Basic Configuration

A basic configuration of acoustic metamaterial samples is a panel consisting of periodic meta-atoms with a centre-mass-intensive form (see **Figure 2.8**). A meta-atom is a melamine cube that wraps an aluminium alloy bar. Thus, the melamine foam is the seat, while the aluminium bar is the core. And this configuration is simple and extensible. Based on superficial judgements, the centre-mass-intensive form increases low-frequency oscillations by contrasting weak surrounding elasticity and heavy centre mass.

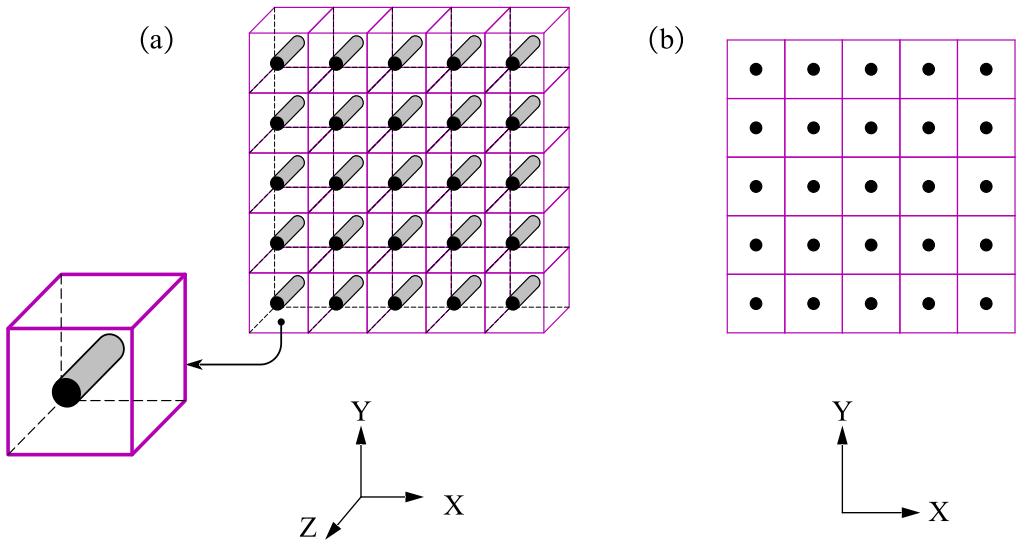


Figure 2.8: Basic configuration of acoustic metamaterials that this paper studies.

2.2.2 Compounding Material–Aluminium Alloy

More specifically, the core material adopted here is the aluminium alloy "AL 6061" with the theoretical elastic modulus $K_{AL} = 68.9$ GPa , the approximate structural damping factor $\eta_{AL} = 0.0023$, and the theoretical mass density $\rho_{AL} = 2700$ kg/m^3 . This material can be seen as an isotropic material. And these parameters introduced here are based on [10].

2.2.3 Compounding Material–Melamine Foam

The surrounding material, melamine foam, has a more sophisticated inner structure and mechanical properties. It is an orthotropic material with linear viscoelasticity. Its elastic moduli and mass density are estimated parameters, and the following data or expressions are extracted from the estimated parameters based on [11]. Then, we can learn that the melamine foam's elastic and anelastic properties are frequency-dependent, and the anelasticity is also called structural damping. Anelasticity is the elasticity times a scaling factor, and these two parameters can be consolidated into a complex form. The frequency-dependent complex elastic matrix can be expressed as follow:

$$[K_{ij}]_{MELA} = [H_{ij}] = [C_{ij}] \cdot \left(1 + \frac{b(i\omega/\beta)^{\alpha}}{1+(i\omega/\beta)^{\alpha}}\right) \tag{1.1}$$

The expression and the coordinate system of the inverse compliance matrix $[C_{ij}]$ are illustrated in **Figure 2.9**:

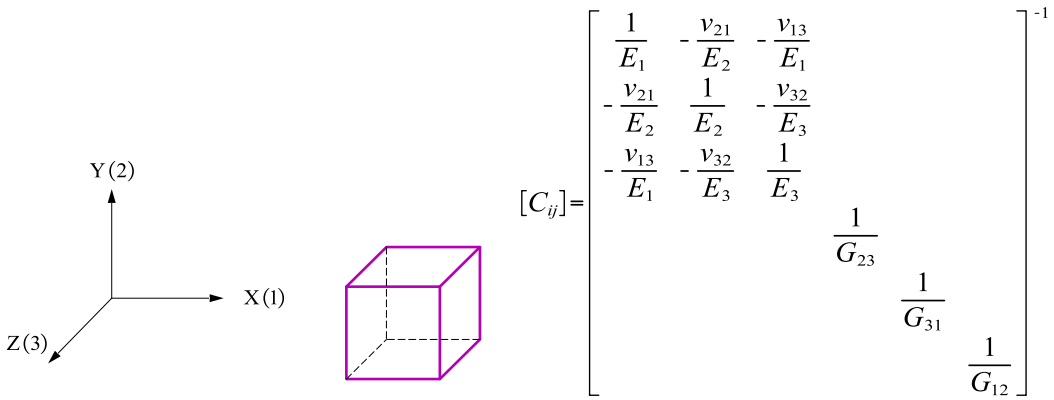


Figure 2.9: The inverse compliance matrix of the melamine foam.

The upper expression is an inverse compliance matrix with a given coordinate system. And the subscripts $i,\ j = 1, 2, 3, 4, 5, 6$ denote degrees of freedom. The other parameters include the angular frequency ω , the relaxation frequency β , the real scalar constant b , and

the fractional derivative order α. The estimated values of parameters, elastic moduli and Poisson's ratios are listed in **Table 2.1**:

Table 2.1: Estimated mechanical properties of melamine foam.

	E_1	E_2	E_3	G_{23}	G_{31}	G_{12}	v_{21}	v_{13}	v_{32}	β	α	b
Values	448	211	170	104	124	101	0.445	-0.514	0.433	813	0.333	0.296
Unit	kPa	kPa	kPa	kPa	kPa	kPa				krad/s		

Based on the estimated values and the frequency-dependent function, we can get the expression of frequency-dependent moduli:

$$\begin{aligned}&\left[\tilde{E}_1(\omega), \tilde{E}_2(\omega), \tilde{E}_3(\omega), \tilde{G}_{23}(\omega), \tilde{G}_{31}(\omega), \tilde{G}_{12}(\omega)\right] \\ &= [E_1, E_2, E_3, G_{23}, G_{31}, G_{12}] \cdot \left(1 + \frac{b(i\omega/\beta)^{\alpha}}{1+(i\omega/\beta)^{\alpha}}\right)\end{aligned} \tag{1.2}$$

In the upper equation, $\tilde{E}_1(\omega)$, $\tilde{E}_2(\omega)$, $\tilde{E}_3(\omega)$, $\tilde{G}_{23}(\omega)$, $\tilde{G}_{31}(\omega)$, and $\tilde{G}_{12}(\omega)$ are frequency-dependent complex moduli. The unit or angular frequency ω is rad/s, and it is easy to convert to the frequency f with the unit of Hz through the formula $f = 2\pi\omega$. Based on equation **(1.2)**, we can get the frequency-dependent values of the complex moduli of the melamine foam in **Figure 2.10** with a frequency range of 0–10^{10} Hz, and **Figure 2.11** with a frequency range of $0-300$ Hz. In **Figure 2.10**, subplot (a) shows the values of the elastic moduli smoothly increase with the frequency increasing. In subplot (b), we can see that the changes in anelastic moduli present a parabola shape. And the anelastic moduli reach the highest value at the frequency of around 10^5 Hz. Since the upper limit of the testing frequency range in chapters **15** and **17** is 220 Hz, **Figure 2.11**, with a frequency range of $0-300$ Hz, is proper to display the property changes of the melamine foam. In **Figure 2.11**, subplot (a) shows that the elastic moduli values almost have no change under the frequency of 220 Hz. Subplot (b) shows the values of the anelastic moduli continuously increase with the frequency values from 0 Hz to 220 Hz.

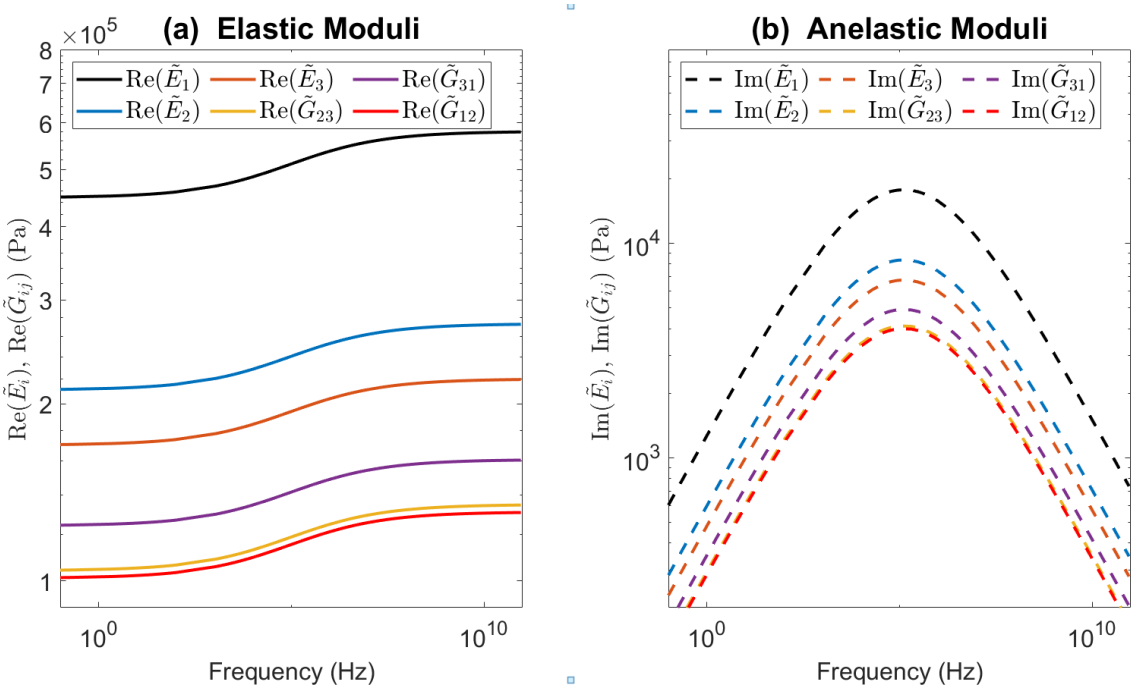


Figure 2.10: Melamine foam's elastic and anelastic moduli in the range of 0–10^{10} Hz by Matlab.

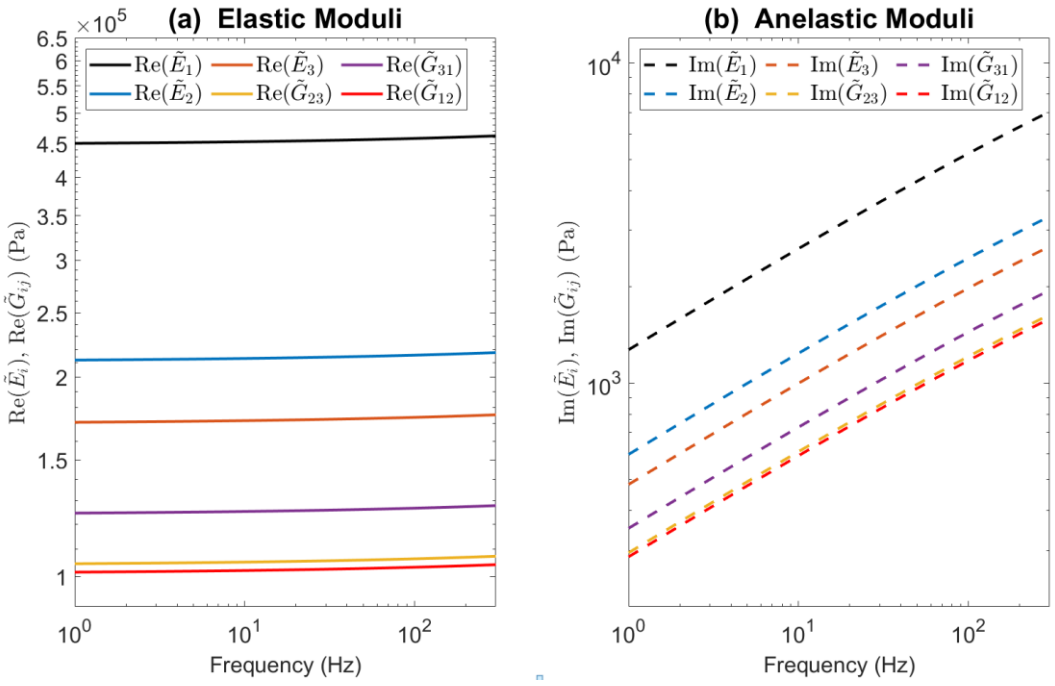


Figure 2.11: Melamine foam's elastic and anelastic moduli in the range of 0–300 Hz by Matlab.

We know that the loss factor is the ratio of the anelastic part divided by the elastic part. Thus, based on equation **(1.1)**, we can get the melamine foam's loss factor (structural damping factor). It is expressed as:

$$\eta_{MELA}(\omega) = \frac{|\mathrm{Im}(H_{ij})|}{|\mathrm{Re}(H_{ij})|} = \frac{\left|\mathrm{Im}\left(1+\frac{b(i\omega/\beta)^{\alpha}}{1+(i\omega/\beta)^{\alpha}}\right)\right|}{\left|\mathrm{Re}\left(1+\frac{b(i\omega/\beta)^{\alpha}}{1+(i\omega/\beta)^{\alpha}}\right)\right|} \tag{1.3}$$

In the upper equation $\eta_{MELA}(\omega)$ is the frequency-dependent structural damping factor of melamine foams. The unit or angular frequency ω is rad/s, and it is easy to convert to the cyclic frequency f with the unit of Hz through the formula $f = 2\pi\omega$. Based on equation **(1.3)**, we can get the frequency-dependent values of the structural damping factor of the melamine foam in **Figure 2.12** with a frequency range of $0-10^{10}$ Hz, and **Figure 2.13** with a frequency range of $0-300$ Hz. **Figure 2.12** shows the values of the structural damping factor change from increasing to decreasing with the frequency increasing. The highest value of the structural damping factor is $\eta_{MELA}=0.0347$ at the frequency of around 9×10^{4} Hz. Since the upper limit of the testing frequency range in chapters **15** and **17** is 220 Hz, **Figure 2.13**, with a frequency range of $0-300$ Hz, is proper to display the changes in the structural damping factor of the melamine foam. **Figure 2.13** shows the values of the anelastic moduli continuously increase with the frequency increases. For example, the value of structural damping is $\eta_{MELA}=0.0013$ at 10^{-2} Hz, $\eta_{MELA}=0.0028$ at 1 Hz, and $\eta_{MELA}=0.0137$ at 220 Hz.

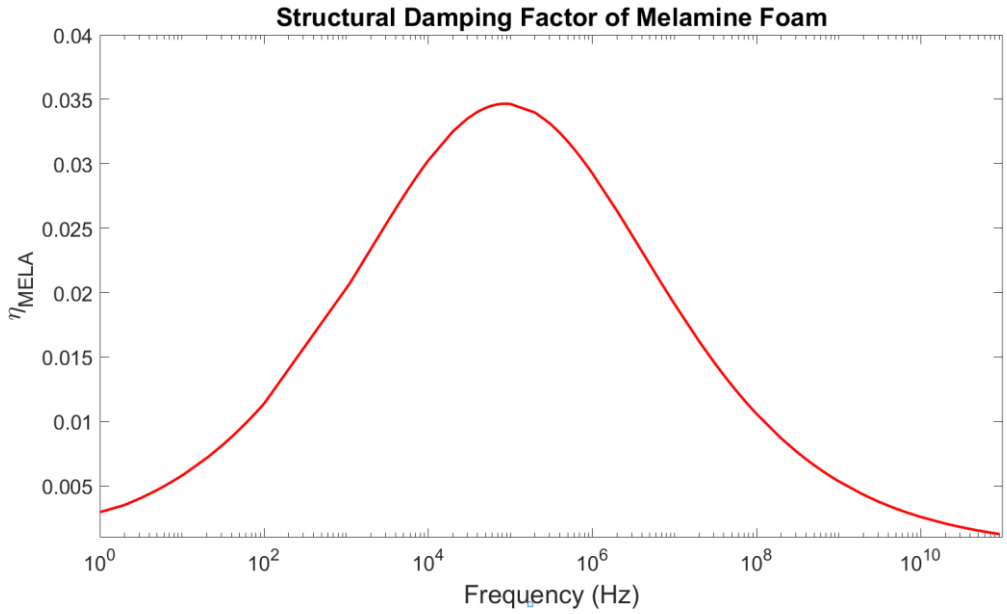


Figure 2.12: Melamine foam's structural damping factor in the range of 0–10^{10} Hz by Matlab.

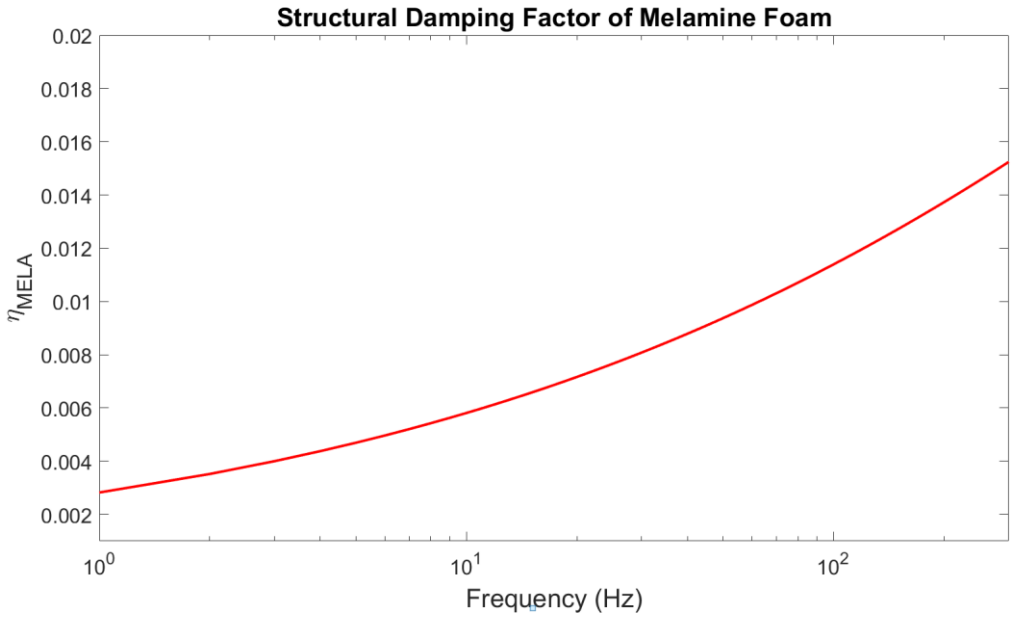


Figure 2.13: Melamine foam's structural damping factor in the range of 0–300 Hz by Matlab.

As for the mass density, considering the reference coordinate system, there are experimentally tested values, the $\rho_x = 9.1\ \mathrm{kg \cdot m^{-3}}$, $\rho_y = 10.2\ \mathrm{kg \cdot m^{-3}}$ and $\rho_z = 9.5\ \mathrm{kg \cdot m^{-3}}$ are in the x, y, and z-axis, respectively. For simplicity, this paper's simulations adopt the average value of these mass densities:

$$\rho_{MELA} = \frac{\rho_x + \rho_y + \rho_z}{3} = 9.6\ \ \mathrm{kg \cdot m^{-3}} \tag{1.4}$$

2.3 Brief Introductions of Simulation Strategy Designs

Designing artificial structures to let them be equipped with unprecedented properties is not an easy job. As described before, inducing eigenfrequencies distribution in the desired frequency range needs proper material compositions and well-designed configurations. The most economic and convenient tool for assisting in metamaterial designs is the FEM software.

FEM software can process eigenvalue extraction analyses, steady-state dynamic analyses, steady-state vibro-acoustic analyses, etc. Commercial software is the main choice for processing sophisticated mechanical simulations. Conceptual designs, prototypes, and verifications always rely on commercial software solution schemes. The reason is that commercial software supplies sufficient mathematical tools and powerful simulation environments. Thus, researchers in different disciplines can release from overloaded mathematical work and coding work to their main subject areas. Experienced corporations got their non-replicable and powerful simulation software through long-term development, and each improvement is difficult. Amon the simulation software, several of them can be candidates for acoustic metamaterial testing, and they may supply several solution schemes for supporting assigned simulation practices. Therefore, choosing proper software and finding efficient simulation strategies is important.

In a preview, this paper concerns two aspects. One is the simulation strategies design, and another is the metamaterial design. As introduced before, metamaterial compositions and configurations correlate highly with the eigenfrequencies distribution. Thus, these two variables are crucial to the dynamic performance of metamaterial. Then, efficient and reliable simulation practices are needed to verify different metamaterial designs. And good simulation practices always need proper simulation strategy designs.

A proper simulation strategy must consider the composite material properties, the relation between substructures and the global structure, the number of DOFs, the fluid-structure interactions, the loading conditions, etc. Furthermore, a good simulation strategy must balance computation accuracy and efficiency. And the testing objects must be representative.

A simulation strategy design starts with choosing the proper simulation software. The reason is that different simulation software has different adaptive fields and upper limits. Thus, a matching process is needed to assess the adaptability of the candidate software before using them. A matching process is to test all available simulation methods and choose the best one. And the method arrangements in or between simulation software can be called simulation strategy designs. Then, control groups of tests are needed to rank the software performances and choose the best simulation strategy (see chapters **14**, **15**, **16**, and section **17.2**).

2.3.1 The Necessity of Simulation Strategy Designs

In general, the calculation efficiency, the hardware requirements, and the computation success rate are highly influenced by the task burden, the number of DOFs, the complexity of the material properties, etc. For example, a dynamic analysis with considerable excitation frequencies to a model with massive DOFs will introduce a big challenge to the hardware capability, even worse, encounter computation failure due to the lack of memory capacity. Furthermore, if the model has frequency-dependent properties, the property-altering mechanism will induce model-rebuilding operations. Thus, multi-time model rebuilding operations will consume enormous computation time. Consequently, a simulation strategy design must adapt different simulation tasks and match diversified model characteristics.

On the one hand, properly modelling the consisting materials is very important. The metamaterial consists of aluminium alloy and melamine foam. The aluminium alloy is elastic and slightly damped, and its inner structure is isotropic. While the melamine foam is viscoelastic, its inner structure is orthotropic. There is no challenge to defining and simulating the aluminium alloy but the melamine foam. The viscoelastic melamine foam is a touchstone because FEM software must update its properties by reconstructing the finite element model. These actions will generate massive intermediate data. And processing massive intermediate data needs a powerful executing system based on advanced software architecture. Thus, the frequency-dependent viscoelasticity can verify the software's intermediate data processing capability.

On the other hand, properly utilising the periodic structure may increase simulation efficiency. The most prominent characteristic of the periodic structure is its congruent unit cells. Thus, copying the properties of one unit cell to others can be a shortcut. Furthermore, simulation design will be more flexible if each unit cell can be defined, managed, and processed independently. Fortunately, superelement techniques may be promising in treating the periodic structure. However, independent operations of each unit cell may induce many more extra

procedures than a single operation to the whole model. Thus, superelement techniques may be deficient.

2.3.2 A Glance of Available Software

This paper adopts three available software: Nastran, Abaqus, and Actran. They have similarities but are experts in different fields.

Nastran is an expert structure analyser due to its efficient solving system. It has advanced model processing techniques, such as domain solver of parallel calculation, superelement techniques, efficient modal-space-based dynamic analyses, “Viscoelastic Material” pattern (efficient frequency-dependent material pattern), DMAP (direct matrix abstracting program) operation to manipulate functional modules directly, etc. However, Nastran is not competent in acoustic analysis.

Abaqus is an advanced Multiphysics analyser, an expert in modelling complex boundary conditions and interactions. Furthermore, Abaqus has an efficient way of modelling acoustic sources (the source-standoff points system). In a word, Abaqus is competent in constructing sophisticated simulation environments. However, Abaqus is not as good as Nastran at treating structural models. Such as defining the frequency-dependent orthotropic material, Nastran has two technique routes to do this job, but Abaqus has no technique supports.

Actran is dedicated to processing acoustic problems, but its solving system is weaker than Nastran and Abaqus. Actran is not good at processing models with massive DOFs. It cannot manipulate intermediate data as efficiently as Nastran or Abaqus. For example, Nastran or Abaqus can execute modal-space-based dynamic analyses in one computation cycle, but Actran needs two independent computation cycles to do the same job. Actran extracts eigenvalues in the first cycle; the extracted eigenvalues and eigenmodes are stored in an external file once the first cycle finishes. Then, Actran starts a new computation cycle based on the extracted eigenvalues and eigenmodes to do the modal-space-based calculations. In addition, Actran does not automatically integrate kernel solvers, so users must manually choose proper solver schemes before starting simulations.

2.4 Scientific Computations VS Simulations

2.4.1 Numerical Data Storage and Dispatch

It is necessary to note that the theoretical reference of this subsection comes from [12]– [15]. Then, we can learn that once a physical model is transformed into numerical data, the mechanical simulation process is controlled by the logic of specific algorithms. In theory, an algorithm that drives calculations is a task-oriented logical layout. And this logical layout compounds a series of functional modules. These functional modules work together one by one to process data. Thus, a simulation process accompanies continuous data flow. The data flow can be roughly subdivided into four states. Data is in processing, storage, dispatch, and deletion. Based on these four states, a data management system is always needed to match the data flow with the solution sequence. In simulations, due to data management, a numerical computation must consider the capability of devices, especially the storage compacity, including the capacities of in-core memory (RAM) and out-of-core memory (hard disk).

The simulation process generates intermediate data besides the input and output data. The input and output data are much smaller than the intermediate data. Especially in dynamic analyses, the amount of intermediate data may be several orders larger than the input and output data.

The input data are written in an input file, including solution driving commands, output requirements declaration, physical models' description (geometric and mechanical information of the model), etc. Similarly, the output data are written in an output file containing the simulation results. The input and output data are permanent, and their files are generally stored on a hard disk.

In comparison, the intermediate data are temporary. In a step-by-step solution sequence, the intermediate data come from upstream calculations and go into downstream calculations. They are dispatched or stored temporarily in scratch files. Once the mission of intermediate data is finished, it will be erased to make place for new incoming data for other calculation tasks. Thus, every intermediate data has a given lifetime.

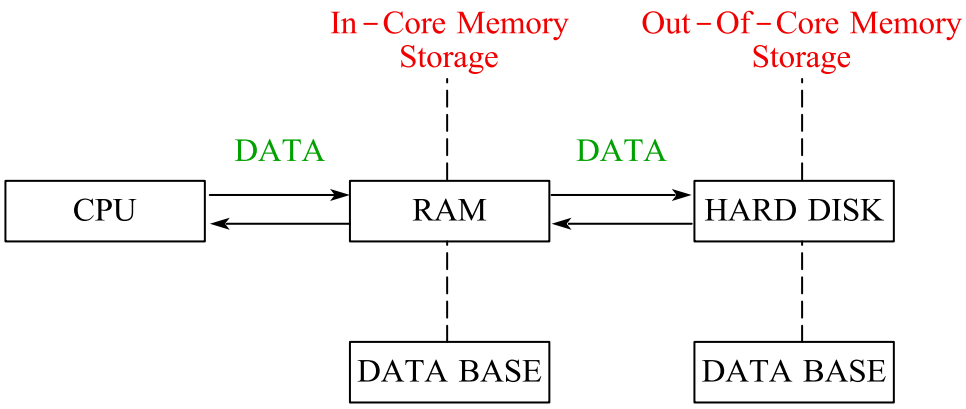


Figure 2.14: Layout of the intermediate data processing.

Consequently, a simulation task must run with proper working memory for efficient data storage and dispatch. In a computer, the working memory is correlated with the storage units. The storage units mainly indicate the RAM (random process memory or in-core memory) and hard disk (out-of-core memory). If the space of RAM is big enough to be the working memory, the simulation will adopt the in-core calculation; while the space of RAM is not enough, the simulation will adopt a merged-core calculation (RAM pluses hard disk). The temporarily stored data may be in the RAM or hard disk, and the temporarily dispatched data may be between the CPU and RAM or between the RAM and hard disk. **Figure 2.14** gives the intermediate data processing schematic layout.

2.4.2 From Self-Build Code to FEM Simulation Software

FEM simulation software is a very complex product. It needs support from various fields, which include physics, mathematics, computer science, software engineering, industrial applications, etc. And every field must experience iterations and upgrades for decades to become mature and practical.

Some simple mechanical computations can be accomplished by self-build codes based on the people's knowledge level, including expertise in specific fields, general mathematical knowledge, general programming skills, limited knowledge of software engineering, etc. When the complexity of computations increases, people must rely on highly integrated mathematical software developed by professional groups. With the help of mathematical software, people's moderate mathematical knowledge and programming skills constrain little in building their concepts and ideas.

Then, when complex computations upgrade into industrial-grade simulations, people must use powerful simulation software developed by world-class companies. Thus, people can overcome barriers of different disciplines, use the virtual environment to model and verify their designs, release themselves from building complex algorithms, and avoid directly processing the intermediate data.

2.4.3 Self-Build Codes

The self-build codes can be used in simple computation tasks. Developing general mathematical models or even simulation environments is far more a personal job. The reason is that people from different disciplines are always experts in limited fields. Developing mathematical or simulation software is years of work by professional groups or companies. Only several worldwide top companies can develop and upgrade FEM simulation software.

Self-build codes aim to execute singular tasks at independent states instead of in given processes. Computations in given processes will generate massive intermediate data, especially dynamic analyses. The massive intermediate data needs data flow control by a well-developed solving system. The well-developed solving system is far more complex than the singular-task algorithm. For writing self-build code, developers may use programming langue such as C++, Fortran, Python, etc. At this moment, people are source-code builders. They must build every fundamental logic, mathematical model, and algorithm by themselves.

2.4.4 Mathematical Software

The mathematical software contains highly integrated mathematical models. And users only need to build their algorithm using proper mathematical tools. This kind of software developed by commercial companies aims to release users from heavy mathematical work and then dedicate them to the field of their forte.

Such as Matlab supplies a comprehensive mathematical platform with plenty of "toolboxes" (integrations of mathematical models in given fields). Users can use it to accomplish various numerical computations. However, Matlab is a powerful mathematical tool rather than a virtual physical environment. Its primary aim is to help users verify their concepts, ideas, and thinking. Through Matlab, users can get intuitive responses from numerical solutions. Matlab cannot automatically execute solution sequences based on predefined logical layouts. Users must design logical layouts for different computation tasks by themselves. Computations in Matlab are mainly based on matrices, and the process can be executed in RAM only. Matlab cannot automatically manage the intermediate data. The reason is that the management of intermediate data needs a predefined logical layout, and Matlab does not have a predefined logical layout.

2.4.5 Industrial-Grade FEM Simulation Software

Industrial-grade FEM simulation software is an application-scenario-oriented system. For example, Nastran is an expert in structural analyses. And Actran is an expert in acoustic analyses. At the same time, Abaqus is a Multiphysics analyser and good at nonlinear analyses. Simulation systems have their specific simulation environments. And the simulation environments have corresponding predefined logical layouts. In a word, simulation systems can calculate what they already know, and they cannot calculate what they don't know. Users can customise predefined solution sequences rather than write algorithms.

Logical layouts are architectures of algorithms. The predefined logical layouts make every intermediate data traceable. Thus, predefined logical layouts are preconditions of automatic data flow control mechanisms. The automatic data flow control mechanisms are necessary functions for applicable simulation software. The reason is that different simulation tasks will generate intermediate data with different scales, and intermediate data management must match the data flow to the hardware storage capacity.

2.5 Concepts of Solving System Based on Nastran

The core of simulation software is the solving system. Like Nastran, its solving system consists of functional modules, a database, and an executive system [16].

2.5.1 The Functional Modules

The functional modules are enveloped mathematical models, which are source-code-level programs. A functional module can be seen as a task-oriented solving unit with specific capabilities for treating corresponding subtasks. For simulation works, control statements in DMAP (direct matrix abstraction program) perform logical branching or looping operations. Then, corresponding functional modules from upstream to downstream are executed sequentially. Thus, functional modules communicate and cooperate to accomplish simulation practices. Nastran's executive system manages communications and cooperation between functional modules via input data blocks, output data blocks, and parameters. Data blocks and parameters can be in scratch or permanent data files as part of the database. In scratch files, they are intermediate data, while in permanent files, they are initial input information or final output results. Data blocks are subdivided into input and output types. They have two distinct forms: matrices and tables. The matrices mainly participate in computing operations, while the tables mainly participate in recording or controlling operations. Data blocks with given names have a header and trailer for defining their characteristics. Parameters are scalar items that specify system characteristics, operations, or controls. There are two types of parameters, the input and output, which work with functional modules. In between functional modules, input parameters affect the internal operation, and output parameters control DMAP logic to convey information upstream to downstream. By the way, these upper arguments are based on [16].

2.5.2 The Database

It is necessary to note that the arguments in this subsection are based on [12], [14], [16]. Then, we can learn that the database is a data pool with given templates containing various data forms, including data blocks, parameters, scratch files, DMAP source files, and DMAP object files required by the software. Thus, during the runtime, the program will temporarily store and retrieve data to cooperate with the solution logic. A database is divided into DBsets (database sets). They are classified into two types: the permanent DBsets and the scratch DBsets. Permanent DBsets may be saved at the end of the run and reused in a restart run. Scratch DBsets are automatically deleted at the end of the run. According to the job processing logic, three databases are emerging for corresponding purposes: the primary database, the delivery database, and the located database.

The set of files forming the database for the current job is called the primary database. This database contains five physical files: MASTER, DBALL, SCRATCH, SCRATCH300, and OBJSCR. The MASTER and DBALL are permanent DBsets, the SCRATCH and SCRATCH300 are scratch DBsets, and the OBJSCR is a temporary DBsets.

- MASTER DBset: a directory of database that contains all the names of DBsets, DBset members and their physical file names, and a directory of projects, versions, data

blocks, parameters, DMAP source, and object files. It also contains the NDDL scheme to describe the database.

- DBALL DBset: a file containing all the DMAP data blocks which may be saved permanently for reuse in a subsequent run.
- SCRATCH DBset: a file temporarily contains all scratch DMAP data blocks.
- SCRATCH300 DBset: a file that temporarily contains internal scratch DMAP functional modules.
- OBJSCR DBset: a temporary file for compilation of DMAP source.

The format or internal layout of a database is called schema, and the primary database gets its schema from the delivery database. The delivery database includes three types: MASTER, MSCOBJ, and MSCSOU. In addition to the primary database, this database is automatically assigned to execute a solution sequence. A located database is remote to the primary database and typically contains data from a previous job. This kind of database is commonly present in a read-only restart run.

2.5.3 The Executive System

The executive system is an interactive platform between the computer operating system and the simulation environment. The executive system contains the logical architecture of the software. Thus, it can manage the database to build a tunnel of communication and cooperation between different functional modules [16].

2.6 The Configuration of Modern FEM Simulation Software

Competitive FEM simulation software needs long-term development by outstanding scientists and engineers. Only several worldwide top software companies hold the key to powerful solving techniques and efficient software architectures. A classic configuration of modern simulation software consists of integrated solvers and a GUI (graphical user interface) workbench. In software, the integrated solvers are the core, and the GUI workbench is the window. Historically, the GUI workbench was born many years after the integrated solvers. In the early stage, engineers must write commands or statements by hand instead of using icons in the GUI workbench. The GUI workbench brings convenience and increases efficiency in operations.

The integrated solvers can also be called the solving system. As introduced before, the solving system may consist of functional modules, a database, and an executive system. The contents of the functional modules and the executive system are source-code-level programs. They are written in a programming language with specific algorithms. One thing to be noted is that the source-code-level programs are not touchable for users. The arguments in this paragraph are based on [14] and [16].

The GUI workbench is an assistant for users. It consists of pre-processors, post-processors, and CAD modules. Thus, it can supply users with a visible and intuitive environment for manipulating geometry models, setting mechanical properties, setting analysis workflows, setting output requirements, demonstrating results extraction, etc. The GUI workbench is a

graphical bridge between the user and the solving system since users can use the GUI workbench to transform abstract concepts and mindsets into specific command instructions.

A simulation practice starts by sending an input file to the solving system. The input file contains two kinds of information. One is the geometrical data and mechanical properties of physical models; the other is the commands or statements of simulation instructions. Users can write an input file through a GUI workbench by clicking icons and filling in drop-down menus. This way is convenient for users, but a compact GUI workbench sacrifices the completeness of solving-system functionalities.

Moreover, users can directly edit an input file by manually writing bulk data and statement instructions. In this way, users can invoke all solving-system functionalities. However, directly editing an input file requires users to know the solving system well and skillfully express their ideas using solving-system languages (statements and data templates).

2.7 Brief Introductions of Software Adopted in This Paper

2.7.1 Brief Introductions of Nastran

Nastran is an expert in structural mechanics and is a powerful, efficient solving system. It can also do aeroelastic and thermal analysis but is not a typical Multiphysics solving system. Nastran always cooperates with Patran since Patran is a GUI workbench (see **Figure 2.15**).

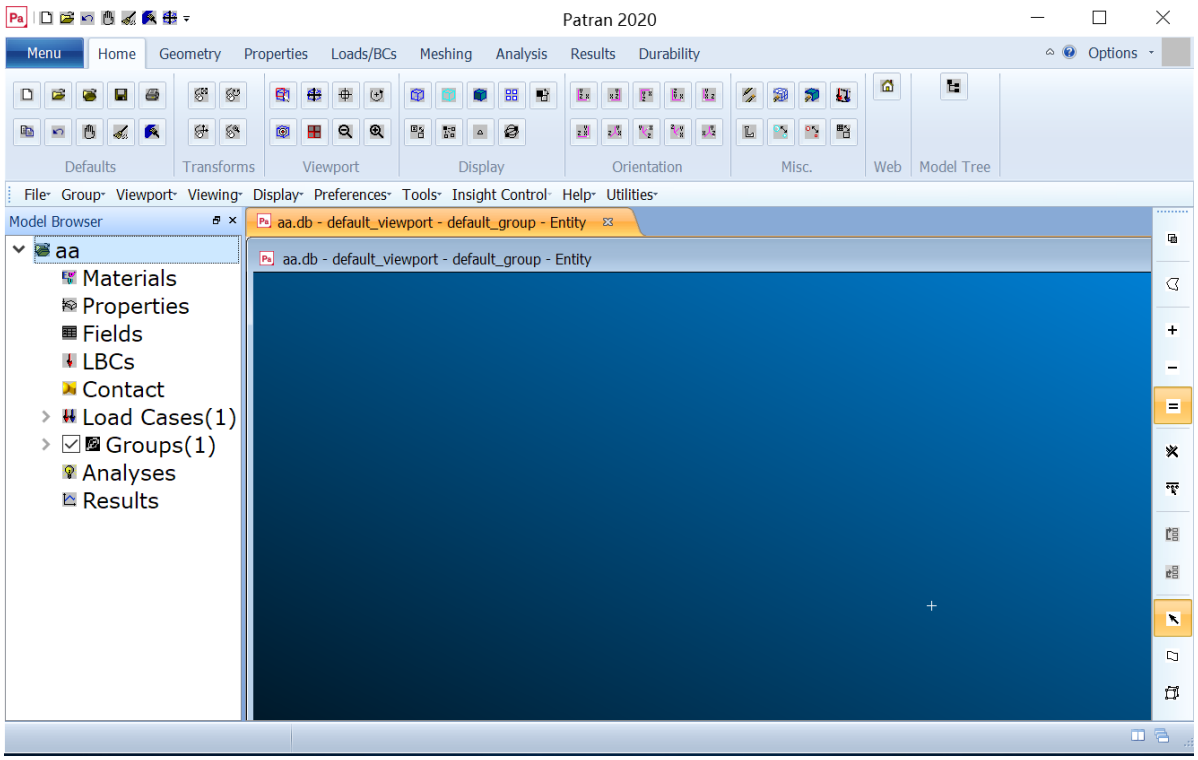


Figure 2.15: The interface of Patran GUI workbench (Version 2020).

As described before, Nastran consists of functional modules, the database, and the executive system. Compared with other solving systems, Nastran has a unique architecture. Users can touch independent functional modules and use DMAP (direct matrix abstraction program) to customise solution sequences. A customising operation includes changing the logic layout of corresponding functional modules and modifying data in the database through NDDL (Nastran data definition language). DMAP is a tool that can help users flexibly design their simulation schema or even accomplish fragmentary calculations. When Nastran executes simulation tasks written through DMAP, the Nastran executive system automatically controls the calculation process, which means the data-flow-control mechanisms automatically participate in the simulation process. Thus, through DMAP, users can process huge-size matrices and do simple matrix-based calculations without worrying about the RAM (random access memory) capacity. DMAP tasks can be automatically executed through in-core and out-

of-core calculations. In comparison, Matlab cannot process matrices or do matrix-based calculations once the data size exceeds the upper limit of RAM. The reason is that Matlab does not have a solving system; thus, it naturally cannot automatically manage the intermediate data through a data-flow-control mechanism.

In common cases, default solution sequences can satisfy most simulation requirements. A default solution sequence means that the software predefines the logical layout of functional modules and corresponding data blocks. Users can use specific statement entries to call the solver to execute default solution sequences, such as "SOL 108" for direct frequency response analysis and "SOL 111" for modal-space-based frequency response analysis. By the way, the description in this paragraph is based on [12].

2.7.2 Brief Introductions of Abaqus

Abaqus is advanced Multiphysics simulation software, good at nonlinear analysis. Its GUI (graphical user interface) workbench is called "Abaqus/CAE" (see **Figure 2.16**), like Patran. However, the solving system of Abaqus is different from Nastran.

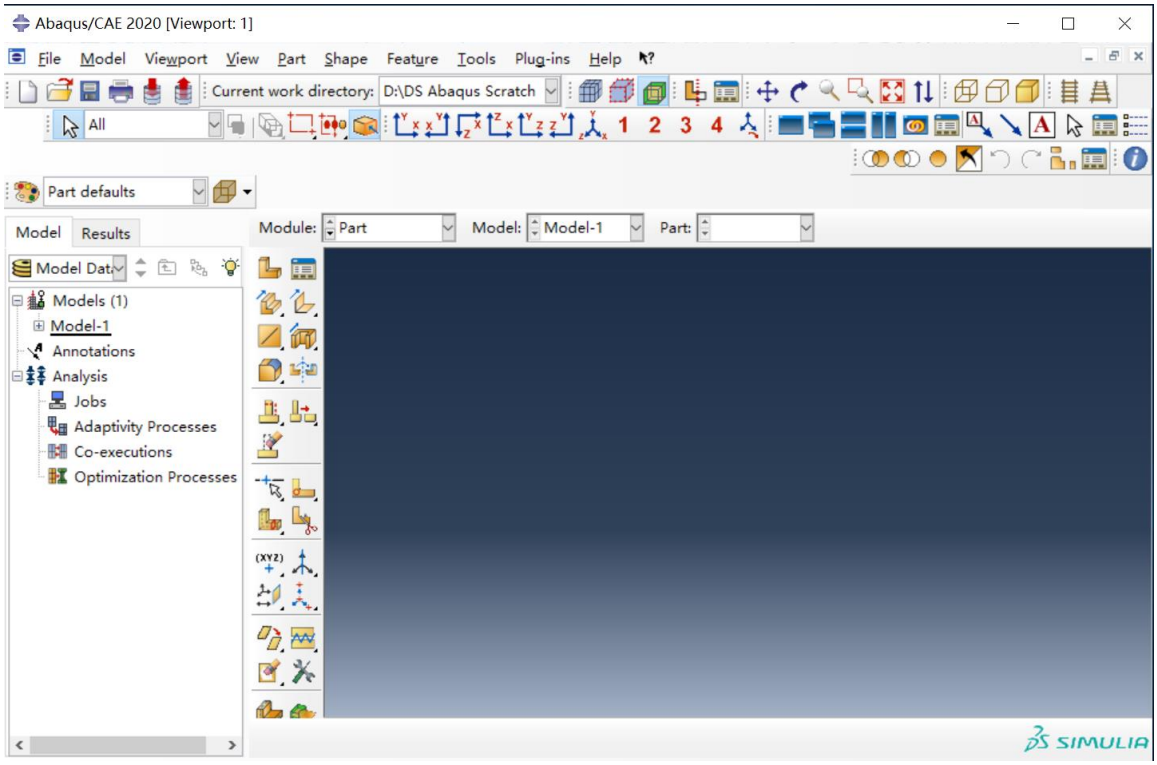


Figure 2.16: The interface of the "Abaqus/CAE" GUI workbench (Version 2020).

The Abaqus solving system contains three different integrated solvers and a co-simulation engine. The three different integrated solvers of Abaqus include "Abaqus/Standard", "Abaqus/Explicit", and "Abaqus/CFD". At the same time, the co-simulation engine is automatically activated when a cross-environment-solving job is acquired. "Abaqus/Standard"

is an integrated system for various simulation practices, including static and dynamic "stress/displacement" analysis, heat transfer and thermal stress analysis, electromagnetic analysis, acoustic and shock analysis, etc. "Abaqus/Standard" contains corresponding solvers for solving linear or nonlinear problems. It uses the implicit time integration method for nonlinear dynamic response analysis. "Abaqus/Explicit" is based on an explicit direct-integration procedure to solve dynamic problems, such as nonlinear dynamic response analysis. The integrated solver adopts an explicit time integration method. It can also solve heat transfer, acoustic, and annealing problems. "Abaqus/CFD" is used in fluid dynamic analysis with incompressible flow using the second-order projection method. By the way, the introductions in this paragraph are based on [17].

Simulations of Abaqus start from an input file. As introduced before, writing an input file can be done through a GUI workbench or by hand. Through "Abaqus/CAE", users can utilise part of the functionalities of the Abaqus solving system. In contrast, users must write the input file by hand for full functionalities.

2.7.3 Brief Introductions of Actran

Actran is software focused on acoustic analysis. Its solving system is called "Actran", and the GUI workbench is called "Actran VI" (see **Figure 2.17**). "Actran" has an access port to the solving system. It receives a user-defined input file and activates a simulation process. On the access port, users can adjust the working state of the solving system by specific commands. "Actran VI" is analogous to "Abaqus/CAE" and "Patran".

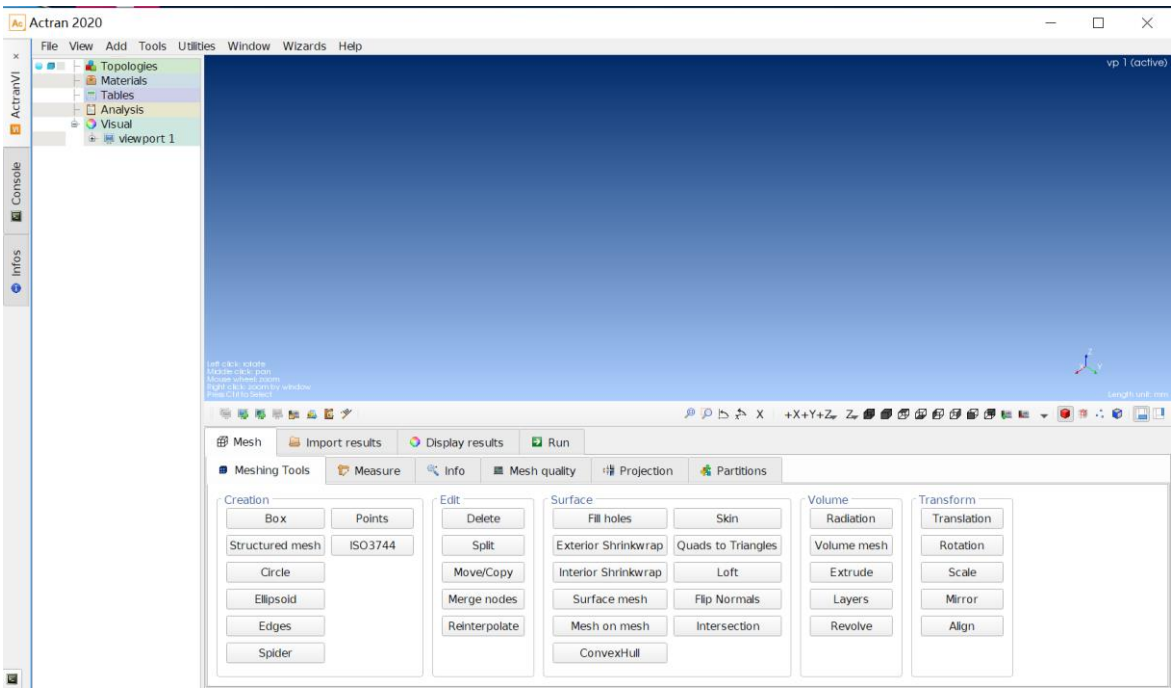


Figure 2.17: Interface of "Actran VI" GUI workbench (Version 2020).

Chapter 3

Simulations in Nastran

3.1 Driving Nastran Through an Input File

An input file of Nastran consists of five subsections, and they are listed in sequence. The input file structure is displayed in **Figure 3.1**. We can see the first subpart is the "Nastran Statement section". The "Nastran Statement Section" is used to adjust the simulating circumstances. In most simulation cases, there is no need to change the default. Therefore, the "Nastran Statement Section" is an optional part. After the first subpart, the second subpart is the "File Management section". The "File Management Section" is used to attach and initialise database or FORTRAN files. This section is optional and only needed for specific file assignments. Below the two optional sections are three required sections. The third subpart, the "Executive Control Section", is the simulation driving section. In the "Executive Control Section", users can call endogenous solution sequences or customise solution sequences through DMAP. Statements in this section may also call for diagnostics. This section is a required section of the input file. Then the fourth subpart is the "Case-Control Section". The "Case-Control Section" helps users customise requirements and manage solution processes, is a required section. Statements in this section can do the following things:

- select loads and constraints,
- request printing, plotting, or punching input and output data,
- set subcases for multiple loading analyses.

As the last subpart, the "Bulk Data Section" contains bulk data entries that specify model geometry, element connectivity, material properties, constraints, boundary conditions and loads, etc. "Bulk Data Section" is the biggest part of an input file and is a required section. By the way, the elaborations in this section are based on [18].

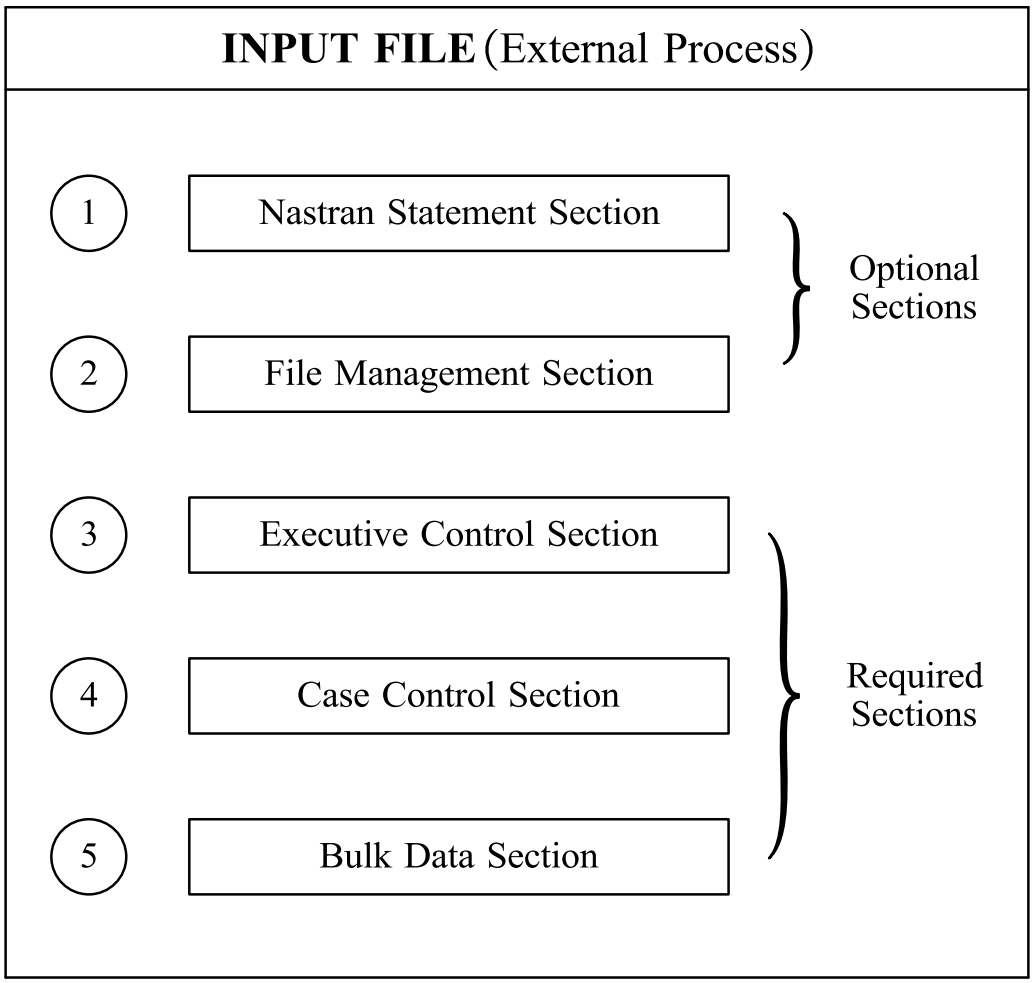


Figure 3.1: The configuration and content of Nastran's input file.

3.2 A Glance of Nastran's Simulations

Nastran collects data, receives statements from a user-defined input file, and translates them into an inner instruction. Then, the executive system automatically arranges corresponding DMAP functional modules. A functional module is a collection of FORTRAN subroutines designed to process a specific task, such as establishing finite-element models, assembly matrices, applying constraints, solving matrix equations, converting data forms, printing solutions, etc. A simulation practice needs cooperative works of massive functional modules.

Each type of analysis available in Nastran can be executed by a specific logical layout of functional modules through the DMAP algorithm. The logical layout of functional modules is called the solution sequence. Nastran supplies endogenous solution sequences for scenario-based applications, such as SOL 108 for physical-space-based steady-state dynamic analyses, SOL 111 for modal-space-based steady-state dynamic analyses, and SOL 103 for eigenvalue extraction analyses. Furthermore, users can customise their logical layout directly using the functional modules through DMAP. By the way, the arguments of these upper two paragraphs are based on [12] and [16].

3.2.1 A Glance of Superelement Techniques

An important step of a simulation is solving matrix equations. Smaller matrices can efficiently reduce the costs of solving matrix equations. Therefore, some simulation software has developed techniques to reduce the matrix size. Matrix-reduction techniques are based on structure subdivisions. And geometrically subdivided structures can be seen as substructures. After condensations, substructures are classified into condensed parts and residual parts. As a result, the condensed structures connect the residual structures with user-assigned boundaries. Then, the condensed structures can be called superelements because their inner DOFs are condensed into boundary DOFs, like a DOFs superposition. Substructures are geometrically subdivided sections of a discretised model. Users can independently assign specific operations, such as giving different definitions, executing controlled calculations and analyses, or requiring specific outputs from each substructure. Substructures in Nastran are always called superelements. They are independent domains in domain-controlled operations. Lastly, the arguments in this paragraph are based on [19].

Besides their independency, superelements are always used in static and dynamic reductions as a cost-saving tactic. The static reduction mainly indicates the Guyan reduction, while the dynamic reduction generally means the Guyan reduction coupling with CMS (Component Modes Synthesis). The reduction is an extra cost that generates smaller matrices. The smaller matrices are savings in succeeding calculations. Thus, superelement techniques are trade-off tactics that users must cleverly control the balance. For more information about the fundamental theories of superelement techniques, please see sections **3.5** and **3.6**. Lastly, the arguments in this paragraph are based on [20].

3.2.2 Solution Sequences with or without Superelements

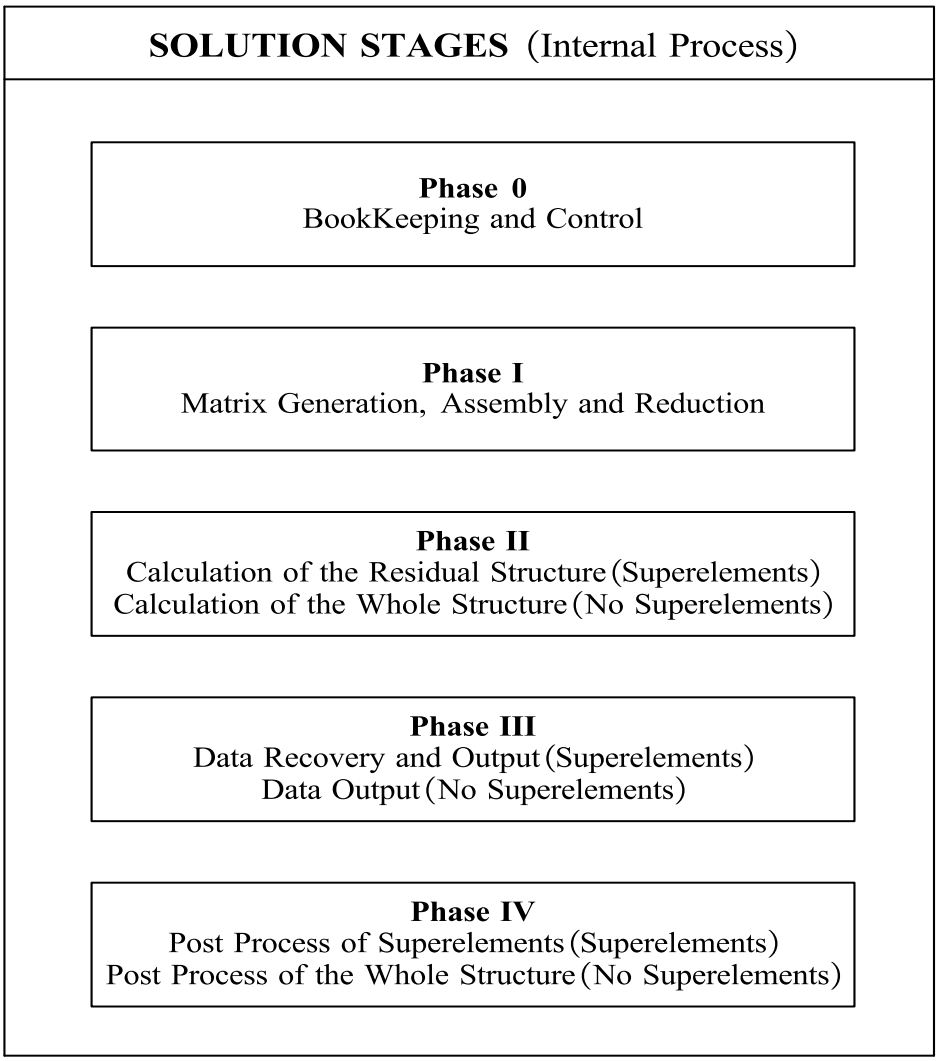


Figure 3.2: Solution stages of a complete simulation in Nastran.

The solution sequences of a simulation are generally classified into five stages. These five stages are “Phase 0” to “Phase IV”, shown in **Figure 3.2**. When adopting superelement techniques, a superelement generation is a succeeding procedure after the model discretisation. The superelement generation procedure is inserted between “Phase I” and “Phase II”. Since each superelement is independent, the superelements generation is a procedure that repeats multiple times of the single superelement generation. And the repeating loop is inserted between “Phase I” and “Phase II”. In dynamic reductions, the superelement generation includes three aspects. They are matrix condensations, component modes calculations, and residual structure assemblies. After that, in “Phase II”, the calculating objects are matrix equations of the reduced residual structure. Finally, based on the recorded information of superelement generations, through an inverse loop of the superelement generation process, the system will

automatically execute the data recovery of each superelement from the residual structure. The data recovery will happen in "Phase III", independent of each superelement. The superelement generation is a step before solving matrix equations. Nastran has an integrated architecture to process models with or without superelements. In general, a simulation process in Nastran contains many procedures. Without using superelement techniques, a simulating process includes model discretisation, the element matrices assembly, the matrix equations calculation, the results extraction, etc. The arguments in this subsection are based on [12] and [20].

3.2.3 A Glance of Simulation Phases

"Phase 0" is the preparation stage. This stage starts with reading an input file and then generating internal instructions. The internal instructions include bookkeeping solution sequences, classifying definitions and requirements, rearranging bulk data in internal conventions, preparing internal data blocks, listing commands or statements, etc. "Phase I" starts with a model discretisation procedure to generate element matrices. Then, if the simulation does not adopt the superelement technique, the element matrices will be assembled into global matrices of the non-condensed model. And if the simulation adopts the superelement technique, the first step is to assemble the element matrices into global matrices of each superelement and residual structure. Then, the second step is to reduce the global matrices of each superelement. After that, the third step is to assemble the reduced superelement matrices with the residual structure matrices to generate the global matrices of the condensed model. "Phase II" is the solving stage of matrix equations. If there are no superelements, the solving object is the non-condensed model. If there are superelements, the solving object is the condensed model. "Phase III" is automatically activated when adopting superelement techniques. This stage uses a loop to recover data of each assigned superelement from the condensed model. "Phase IV" is always used for post-processing. Post-processing indicates extracting data or plotting results. For a model with superelements, the post-processing of each superelement must have the data recovery process in "Phase III". By the way, the introductions in this subsection are based on [12].

3.3 The Simulation Phase 0

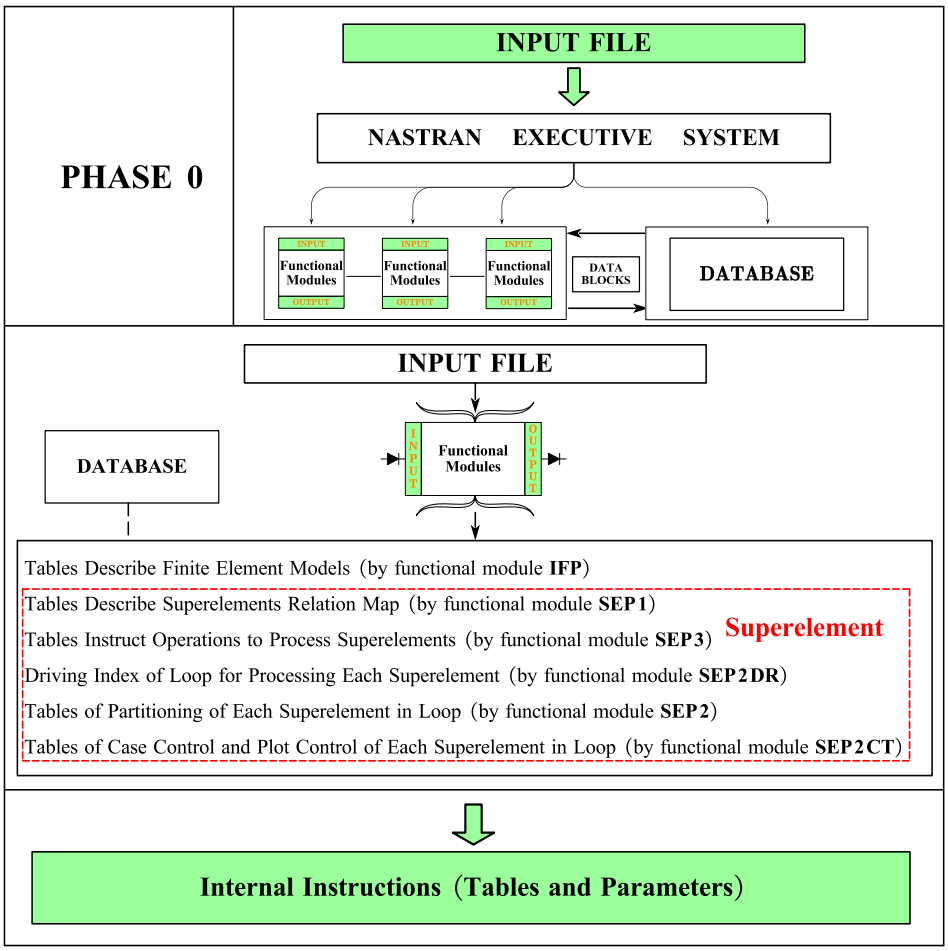


Figure 3.3: Nastran's "Phase 0".

Figure 3.3 is a simplified flow chart illustrating the "Phase 0" stage. When an input file comes in, the Nastran executive system starts driving functional modules, preparing the database, and controlling data flow through internal data blocks. In the simulation phase, the input is a user-defined file, and the outputs are internal instructions for the succeeding simulations. "Phase 0" is a translator that translates external input files to internal instructions. The instructions may be some internal tables or parameters. These tables or parameters are stored in specific internal data blocks as part of the database. The introductions in this paragraph are based on [12].

The frontal section of the following figure shows that the Nastran executive system arranges corresponding functional modules to process tasks of this simulation phase. And the executive system controls the logic of cooperative engagement between the functional modules. At the same time, the executive system prepares the corresponding database for the intermediate data. And the intermediate data is transferred through specific data blocks. The internal manipulations of intermediate data transfer and storage are called data flow control.

The middle section of the following figure shows part of the instruction contents and the name of functional modules. If the simulation adopts the superelement technique, the contents in the red-dashed box will be a part of the internal instructions.

3.4 The Simulation Phase I

3.4.1 PHASE I (Models without superelements, Physical Space)

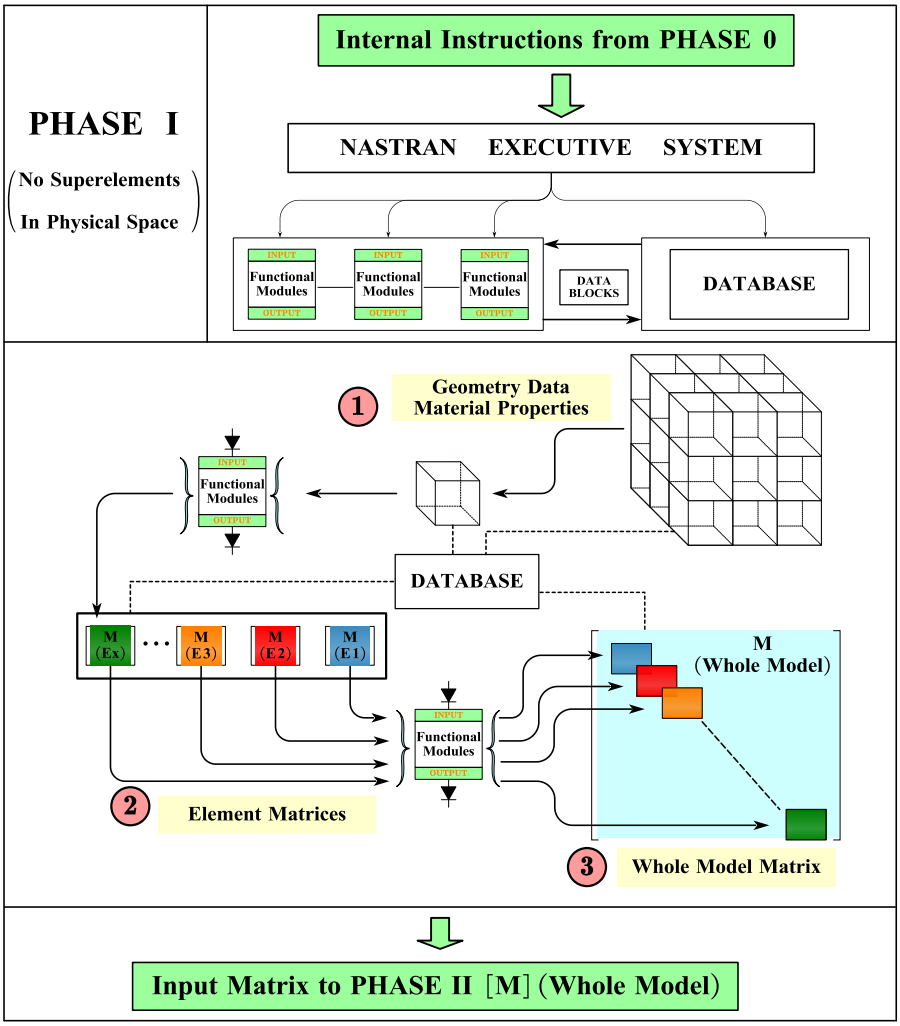


Figure 3.4: Nastran's "Phase I" (without superelements, physical space).

Figure 3.4 is a schematic diagram of Nastran's simulation at stage "Phase I" when models contain no superelements. And it is a physical-space-based simulation. The middle section of Figure 3.4 shows how discretised geometry data become a whole matrix of the non-condensed model. There are three data groups marked with numbers in red circles. The first one is the geometry and material information of the model, which will become finite element matrices

through specific functional modules. Thus, the finite element matrices become the second data group. In the next procedure, the finite element matrices will be assembled into a whole matrix of the non-condensed model. Thus, the whole matrix becomes the third data group. Finally, the generated whole matrix of the non-condensed model will be the input content of "Phase II". And the whole matrix may indicate the stiffness matrix, the mass matrix, the damping matrix, etc. The introductions in this paragraph are based on [12].

3.4.2 PHASE I (Models without Superelements) + PHASE II (Modal Space)

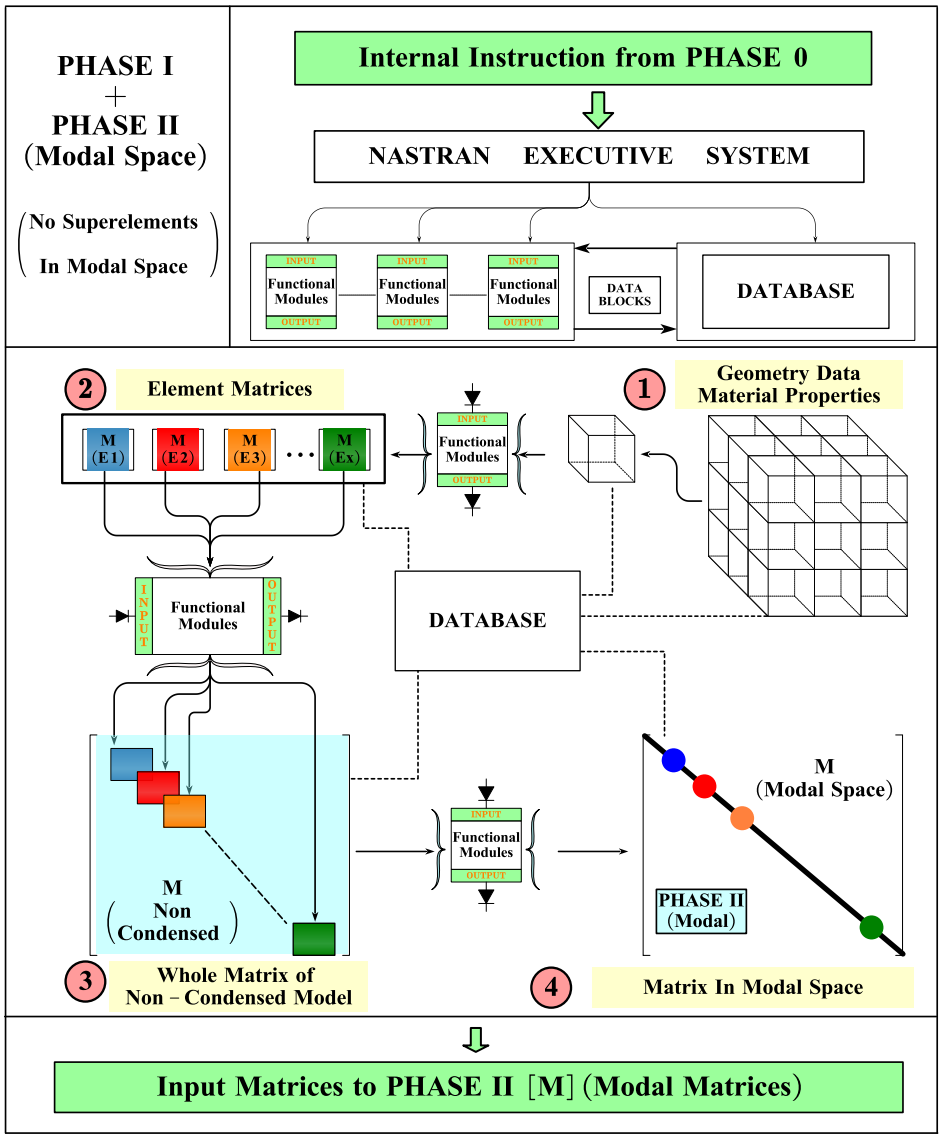


Figure 3.5: Nastran's "Phase I" (with superelements, modal space).

Figure 3.5 is a schematic diagram of Nastran's simulation at stage "Phase I" when models contain no superelements. And it is a modal-space-based simulation. This case adds a modal reduction procedure after assembling finite element matrices. The modal reduction procedure can be seen in the middle section of **Figure 3.5**, the fourth step marked in a red circle. This procedure needs a prior procedure for eigenvalue extraction in "Phase II". Outputs from this stage are modal matrices of the non-condensed model. The introductions in this paragraph are based on [12].

3.4.3 PHASE I (Models with superelements)

Figure 3.6 and **Figure 3.7** are schematic diagrams of Nastran's simulation at stage "Phase I" when models contain superelements. **Figure 3.6** corresponds to physical-space-based analyses, and **Figure 3.7** corresponds to modal-space-based analyses. The data processing in "Phase I" is the same. This data processing transforms the geometry and mechanical information into finite element matrices.

Figure 3.6 and **Figure 3.7** shows how specific functional modules transform the first data group (number 1 in a red circle) into the second data group (number 2 in a red circle). These finite element matrices are stored temporarily in the database and wait for the assembly procedure. When the simulation adopts the superelement technique, assemblies of finite element matrices will be executed in two cases. In the first case, the corresponding finite element matrices are assembled into the global matrix of the residual structure. Then, in the second case, the corresponding finite element matrices are assembled into the global matrices of each superelement. **Figure 3.6** and **Figure 3.7** show these two assembly cases. The first case generates the third data group (number 3 in a red circle), and the second case generates the fourth data group (number 4 in a red circle). Here is an eye-catching part in the fourth data group, highlighted with the red-dashed box, which tells that superelements processing needs a loop, and the loop is in a predefined sequence. Under the guidance of superelement defining tables from "Phase 0", each superelement is treated independently in "Phase I". The loop is a driver for repeating the same procedures that treat the superelement from the first to the last [20].

Superelement matrices generation conducted in "Phase I" consists of two steps. The first step is the assembly of the finite element matrices. And the second step is the superelement matrix condensation. The superelement matrix condensation in static analyses can be called static reductions, while it in dynamic analyses can be called dynamic reductions. And dynamic reduction, in essence, is the static reduction that pluses the component modes synthesis [20]. The reduced superelement matrices are temporarily stored in the database. Then, these reduced matrices will be assembled into the residual-structure matrix.

An advantage of the superelement technique is that it can provide reduced matrices. The reduced matrices can improve the efficiencies of matrix calculations in "Phase II". However, the matrix reduction of superelements incurs extra costs in "Phase I". Therefore, the superelement technique for users is a trade-off choice. Usages of the superelement technique in inappropriate situations may incur contrary effects. Such as model rebuilding operations due to frequency-dependent material. High costs in matrix reduction. Too many superelements are set.

Simulations in **Figure 3.6** are physical-space-based analyses. The objective matrices of the matrix-equation calculation in "Phase II" are reduced matrices in physical space by superelement reductions. And the reduced matrices are the fifth data group, marked with 5 in a red circle.

Simulations in **Figure 3.7** are modal-space-based analyses. The objective matrices of the eigenvalue extraction in "Phase II" are reduced matrices in physical space by superelement reductions. And the reduced matrices are the fifth data group, marked with 5 in a red circle. Then, the objective matrices of the matrix-equation calculation in "Phase II" are modal matrices by modal reductions. And the modal matrices are the sixth data group, marked with 6 in a red circle.

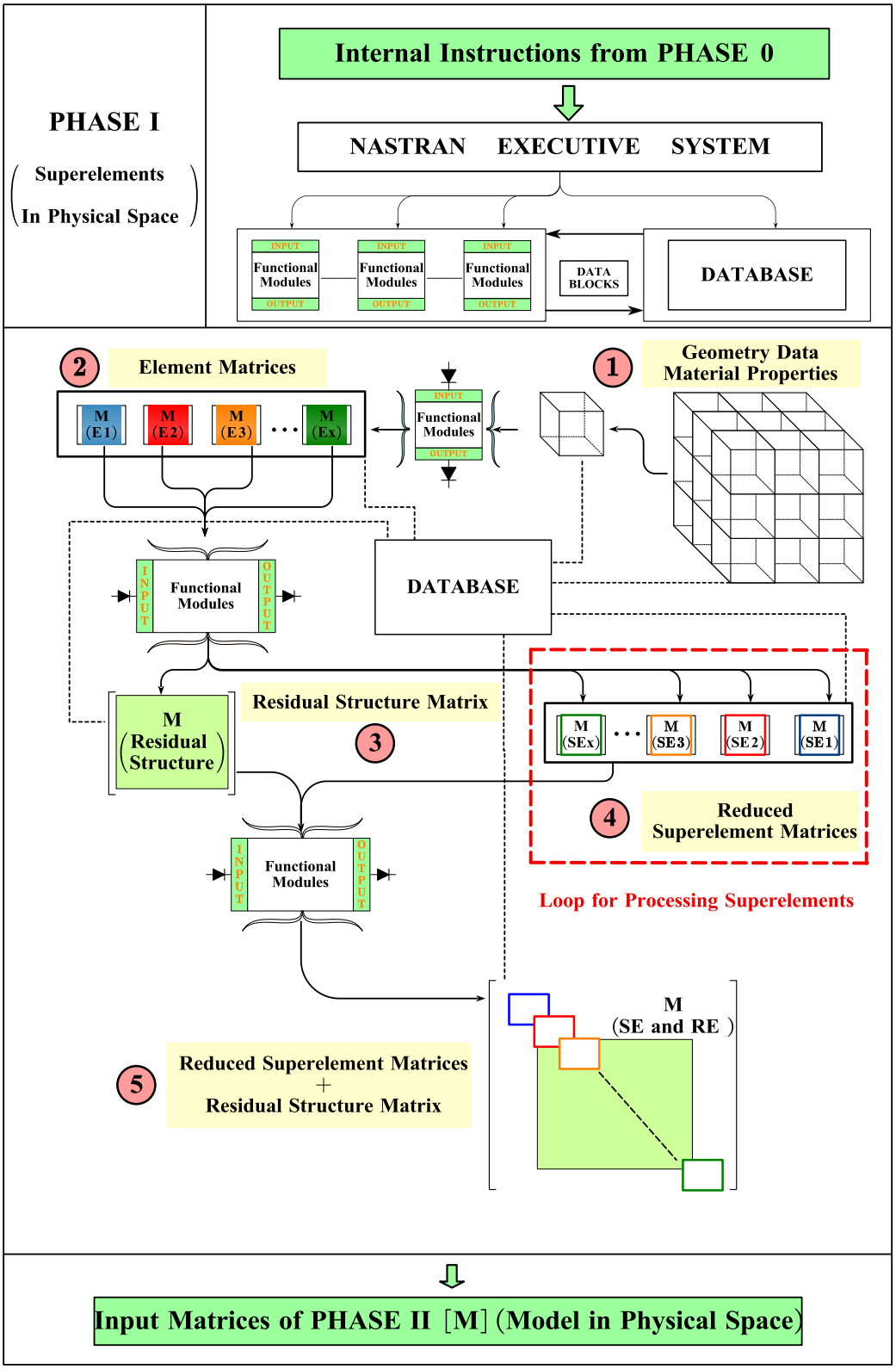


Figure 3.6: Nastran's "Phase I" (superelements, physical space).

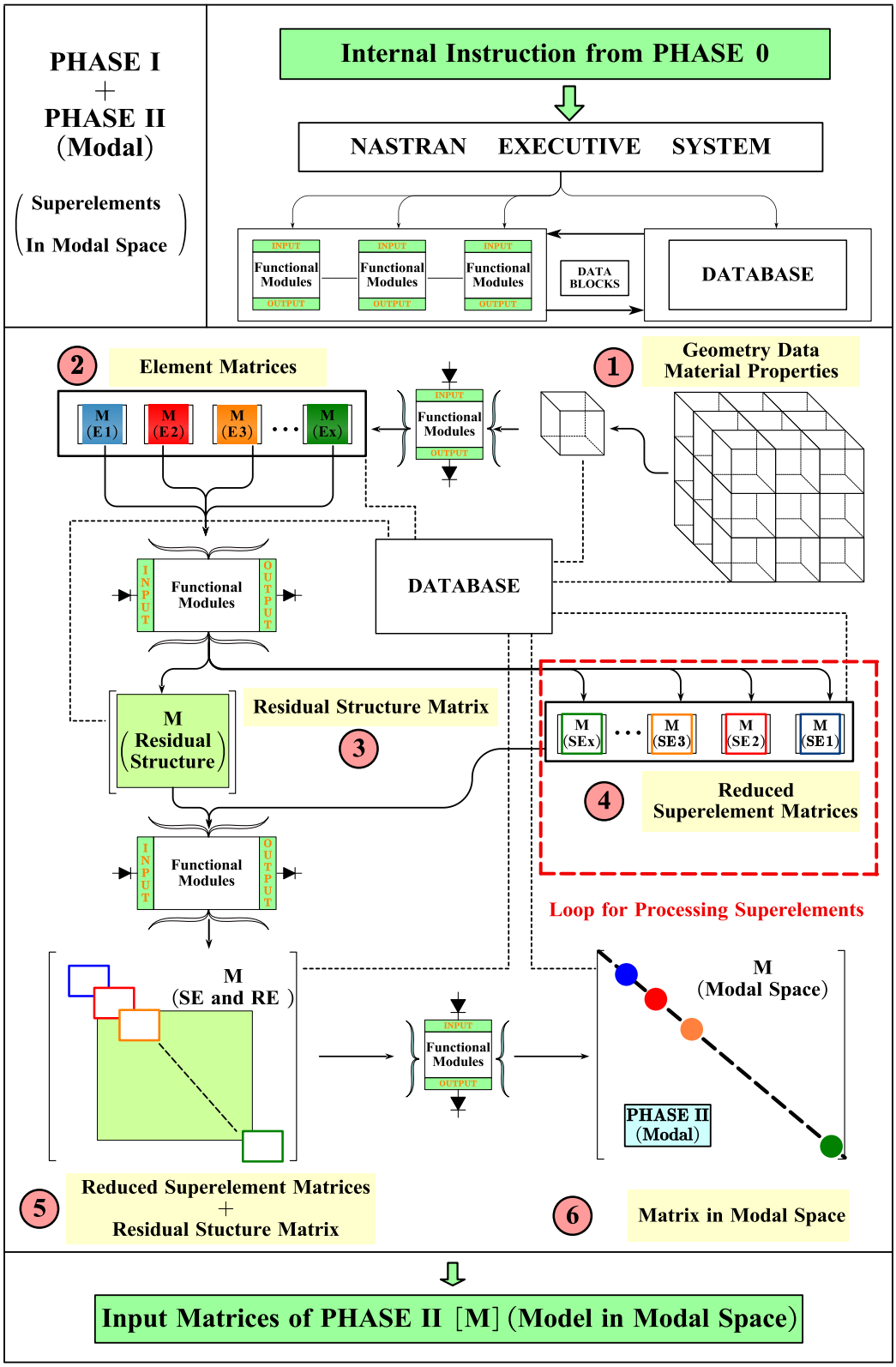


Figure 3.7: Nastran's "Phase I" (with superelements, modal space).

3.5 Brief Introductions of Superelement Techniques of Nastran

3.5.1 Classification of Superelement Techniques

Superelement techniques can increase the efficiency of simulations, especially when a model is processed multi times. Each superelement can be processed individually, so it is possible to apply local manipulations to assigned superelements instead of the whole model. Nastran supplies users with four kinds of superelement defining patterns: “List Superelement”, “Image Superelement”, “Part Superelement”, and “External Superelement” [21]. Moreover, Nastran has a high-performance solver called the “Domain Solver” that can execute parallel calculations. And this solver can drive a superelement-technique-based method called the “ACMS Method” (automated component modes synthesis method) that can efficiently execute the eigenvalue extraction analysis [19].

The “List Superelement Method” means listing superelement definitions in the main bulk-data section of a Nastran input file. Then, inner programs automatically execute superelement generations. The superelement generations do not need users to do external manipulations. By the way, the arguments in this paragraph are based on [21].

The “Image Superelement Method” means the superelement definition is like an image that copies the mechanical properties of some specifically assigned superelements. The specifically assigned superelements are called the primary superelements. They are mutually distinct and will be processed first. Then, other congruent superelements that copy the matrices of the corresponding primary superelements are called secondary superelements. The superelement generations do not need users to do external manipulations. By the way, the arguments in this paragraph are based on [22].

The “Part Superelement Method” means each superelement has its bulk-data section that contains its geometry information, mechanical properties, constraints conditions, and loading cases. And the “Part superelement Method” only executes model subdivisions but no reductions. The superelement generation and superelement usage are two independent steps. Thus, users must manually set and control the process. By the way, the arguments in this paragraph are based on [21].

The “External Superelement Method” is analogous to the “Part Superelement Method”. The superelement generation and superelement usage of the “External Superelement Method” are two independent steps. But the difference is that the “External Superelement Method” executes model subdivisions and reductions. Users must manually set and control the process. By the way, the arguments in this paragraph are based on [21].

3.5.2 Available Superelement Techniques

The metamaterial sample will be subdivided into many substructures, which need to be set as superelements (see subsection **2.2.1**). If too many superelements need to be set, manipulations of the “Part superelement Method” and the “External superelement Method” will incur practical difficulties. Thus, these two superelement techniques are indispensable in this paper. Considering the automatic superelement generations, the “List Superelement Method”, the

"Image Superelement Method", and the "Domain Slover" with "ACMS Method" are the superelement techniques this paper adopts for assessing the metamaterial samples.

In theory, the "Image Superelement Method" can save the cost of superelement processing compared to the "List Superelement Method". The reason is that when using the "Image Superelement Method", the secondary superelements will copy the matrices of the corresponding primary superelements. Thus, the secondary superelements do not need to be processed. However, it does not mean the "Image Superelement Method" is always more efficient than the "List Superelement Method". The reason is that the "Image Superelement Method" has a more complex data recovery process, and this process may influence efficiencies.

The "Domain Solver" with the "ACMS Method" is proper for the eigenvalue extraction and modal-space-based dynamic analyses. This method is a superelement-technique-based method. However, the advantage of this method is different from the conventional superelement method. Since the superelements come from a subdivision of the whole model, each superelement can be processed independently. Thus, the parallel calculation can be applied to the independent sub-models. That is to say, the cost saving of this method is not only the reduced matrices but also the optimised computing architecture. The arguments in this paragraph are based on [21] and [23].

3.5.3 Degrees-of-Freedom Sets

Discretised geometry models can be called "Mesh" because they are connected arrays of grids (or nodes). A discretised geometry model is a system consisting of grids and grids relations. If grids are geometrically connected, they are logically coupling with each other, and each two can be called a grid pair. In the matrices of a discretised model, the non-zero elements correspond to grid pairs. Then, it is necessary to note that this subsection's introductions are based on [24].

In FEM software, the DOFs need proper classifications to prompt the clear processing logic. The classifications of DOFs in Nastran can be called DOFs sets. These DOFs sets indicate boundary conditions, constraints, inner and outer DOFs of superelements, free body motions, scalar points, etc.

Figure 3.8 displays all kinds of DOFs sets in Nastran. Corresponding sections in the Nastran reference manual show specific meanings of these DOFs sets in detail. For models with superelement, some of them need more attention. The letter "g" represents all DOFs and can be subdivided into constrained DOFs and unconstrained DOFs. In constrained DOFs, the letter "m" represents multipoint constraints, and "s" represents single-point constraints. The letter "f" represents unconstrained (or free) DOFs. For the model with superelements, unconstrained DOFs include omitted DOFs, indicated by the letter "o", and analysis DOFs, indicated by the letter "a". The omitted DOFs mainly indicate the interior DOFs of a superelement because a static reduction condenses information of interior DOFs to boundary DOFs. The analysis DOFs, indicated by the letter "a", include component-modes DOFs, indicated by the letter "q", and physical boundary DOFs, indicated by the letter "t".

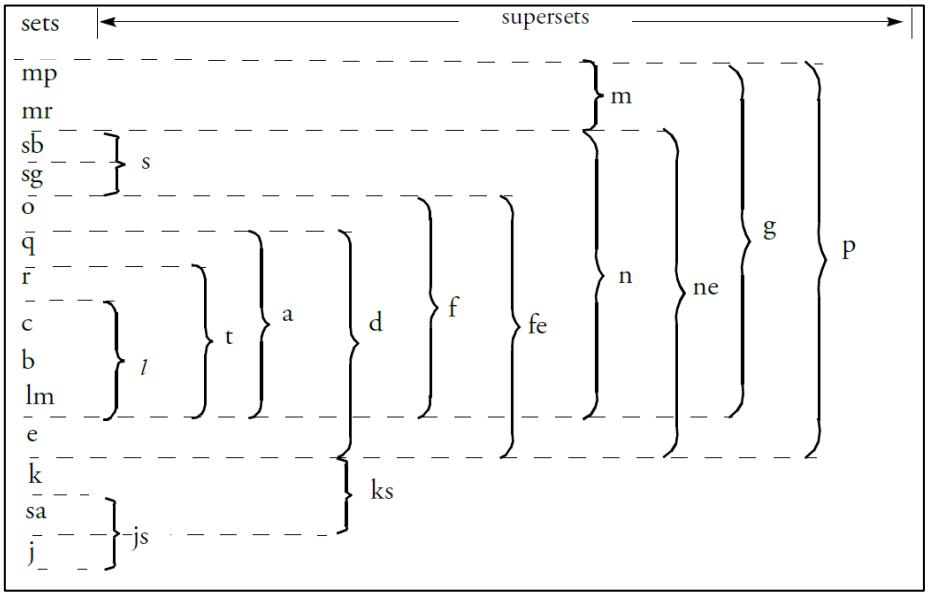


Figure 3.8: Schematic diagram of DOFs sets in Nastran [24].

3.6 The Processing of Superelements

3.6.1 Brief Introduction

It is necessary to note that the theoretical reference of subsections **3.6.2** to **3.6.6** is [20]. Based on reference [20], this paper quotes the formula and symbols; then reorganises the logic and expressions. After that, we can learn that static and dynamic analyses can use superelement techniques, but these two simulation routes undergo different processing schemes.

A static analysis concerns the stiffness matrix because the stiffness matrix is a medium between static loads and static responses. A static analysis commonly does not consider dynamic motions, such as accelerations and velocities, so it omits the effects of inertia and damping forces. The reduction in a static analysis treats the stiffness matrix, called the static reduction or Guyan reduction. A static reduction condenses matrices from the inner DOFs to the outer DOFs (boundary DOFs). Users define inner DOFs and outer DOFs through superelement definitions. As a result, it does not influence the accuracy of succeeding analyses. The reason is that no dynamic effects of the condensed inner DOFs need to be considered in the static process. The arguments in this paragraph are based on [19].

In comparison, a dynamic analysis must at least consider the inertia forces of the inner DOFs. A dynamic reduction is a static reduction plus a calculation of component modes. A static reduction cannot project the dynamic effects of the inner DOFs to the outer DOFs. Therefore, a dynamic reduction needs compensation for the dynamic effects of inner DOFs after Guyan reduction, and the approximated compensation is called component modes. The

component modes are modes of superelements, and they can be calculated in fixed-boundary or free-boundary conditions. The method that calculates component modes with a fixed-boundary condition is called the Craig-Bampton method, and this method is adopted in this paper. After a dynamic reduction, the analysis DOFs consist of the residual DOFs, the static condensed inner DOFs, and the compensated DOFs (superelement component modes). Dynamic reduction can also be called Component Modes Synthesis (CMS). And introductions in subsection **3.6.5** show the corresponding classification. Then, we can see that the analysis DOFs are indicated by the letter "a"; the component modes DOFs are indicated by the letter "q"; the residual DOFs are indicated by the letter "t". The arguments in this paragraph are based on [20].

The following section explains a dynamic reduction process, and this process includes the basic steps in "Phase I" for processing a superelement. The explanation consists of four parts to clearly show the logic of a dynamic reduction (Component Modes Synthesis by the Craig-Bampton method). For simplification, the processing objects of the dynamic reduction are stiffness and mass matrices. Damping matrices are not included here because of their complex definitions, but the processing of damping matrices is analogous to those of stiffness and mass matrices.

These four parts are the partitions of the whole model, the static reduction of superelement, the calculation of component modes, and the component modes synthesis. The processes of these four parts are illustrated in **Figure 3.9**, **Figure 3.10**, **Figure 3.11**, and **Figure 3.12**, respectively. The black boxes containing formulas in these figures connect the database with dashed lines. And it means that all data exchange between calculation steps must rely on the data-flow-control mechanism in Nastran.

3.6.2 Partition of the Whole Model

Partition of the whole model is the first step of setting superelements. This step depends on how users define the boundaries connecting each superelement and the residual structure. From the view of superelement, DOFs at the boundaries are the outer DOFs, and DOFs except boundaries are the inner DOFs.

The section marked with the number "1" in a red circle in **Figure 3.9** means the partition procedure is the first step of superelement processing. The sample model contains four superelements, the green dots represent the inner DOFs, and the black dots with solid lines represent the outer DOFs. As described before, the letter "f" represents unconstrained DOFs. The letter "o" represents omitted DOFs after condensation. The letter "t" represents boundary DOFs after a condensation, and the letter " $\bar{t}$ " represents original boundary DOFs before a condensation.

The subscripts in matrices or matrix subsections indicate the corresponding DOFs sets. The partition of the DOFs subdivides a superelement matrix into four parts. One is for the inner DOFs set, one for the outer DOFs set, and the other two are the intersecting DOFs sets, which are symmetric. Here use the stiffness matrix and mass matrix as examples. The unconstrained-DOFs stiffness matrix $[K_{ff}]$ is subdivided into an inner-DOFs part $[K_{oo}]$, an outer-DOFs part $[\bar{K}_{tt}]$, and an intersecting-DOFs part $[K_{ot}]$. The partition of the mass matrix is the same. The unconstrained-DOFs mass matrix $[M_{ff}]$ is subdivided into an inner DOFs

part $[M_{oo}]$ and an outer DOFs part $[\bar{M}_{tt}]$ before the reduction and intersecting-DOFs part $[M_{ot}]$.

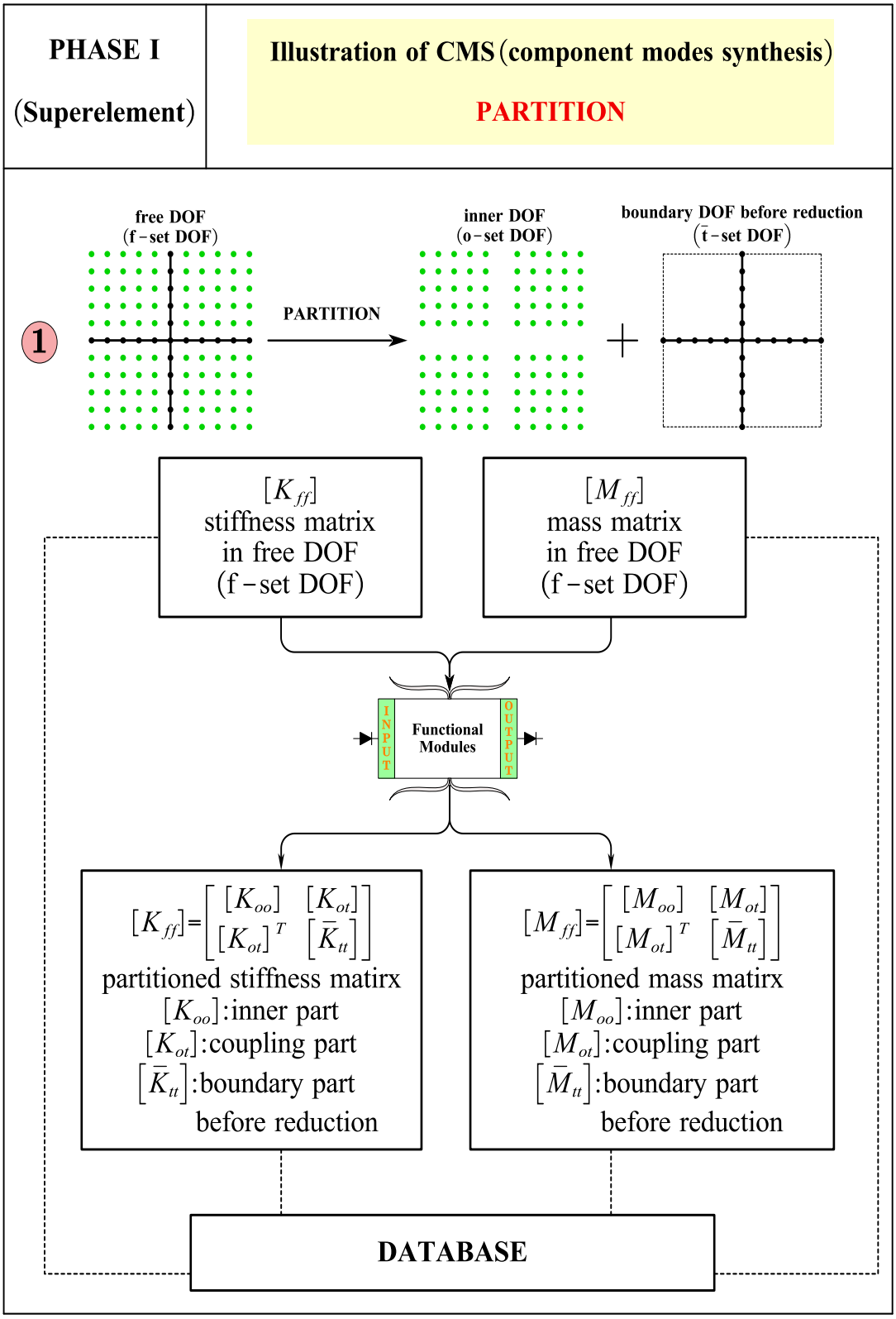


Figure 3.9: Matrix partition in defining superelements.

3.6.3 Static Reduction of Superelements

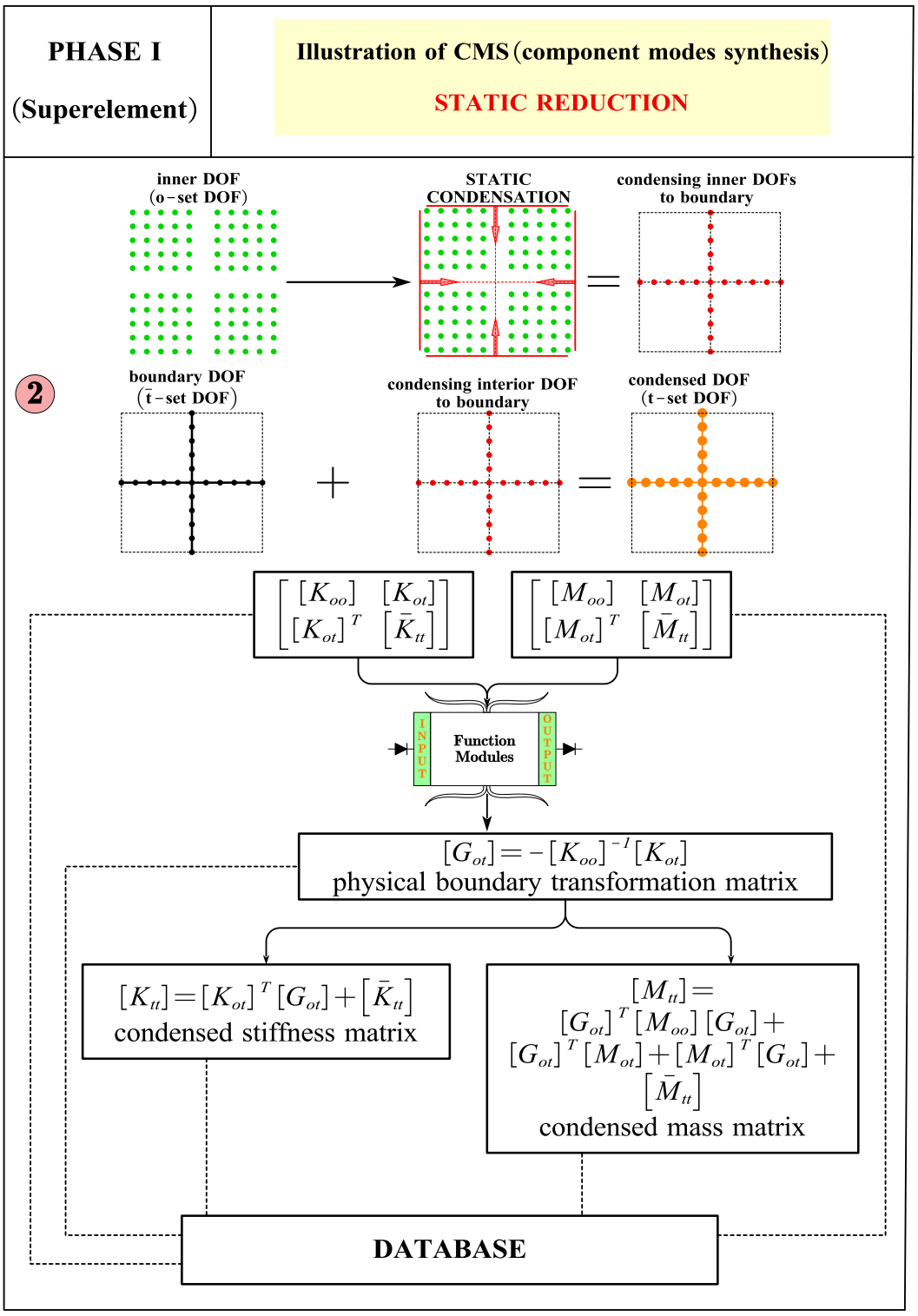


Figure 3.10: Static reduction in processing superelements.

The section marked with "2" in a red circle in **Figure 3.10** means the static reduction procedure is the second step of superelement processing. This procedure succeeds a partition procedure, and the first step is to process partitioned stiffness matrices. The matrices in **Figure 3.10** are original and transformed matrices of a superelement. Here is the definition of the boundary-DOFs transformation matrix,

$$[G_{ot}] = -[K_{oo}]^{-1}[K_{ot}] \tag{2.1}$$

In calculating the boundary-DOFs transformation matrix, the inverse of the inner-DOFs stiffness matrix $[K_{oo}]^{-1}$ may cause a massive calculating cost when the matrix size $[K_{oo}]$ is immense. Then, the condensation of the stiffness matrix can be expressed as:

$$[K_{tt}] = [K_{ot}]^{T}[G_{ot}] + \left[\bar{K}_{tt}\right] \tag{2.2}$$

The matrix $[K_{tt}]$ is the condensed boundary-DOFs stiffness matrix after the static condensation. Then, the condensation of a mass matrix can be expressed as:

$$[M_{tt}] = [G_{ot}]^{T}[M_{oo}][G_{ot}] + [G_{ot}]^{T}[M_{ot}] + [M_{ot}]^{T}[G_{ot}] + \left[\bar{M}_{tt}\right] \tag{2.3}$$

The matrix $[M_{tt}]$ is the condensed boundary-DOFs mass matrix after the static condensation. The structural or viscous damping matrices are reduced similarly. For simplicity, there is no more elaboration about the reduction process of structural or viscous damping matrices.

3.6.4 Calculation of Component Modes

The section marked with the number "3" in a red circle in **Figure 3.11** means the calculation of component modes is the third step of superelement processing, and the sample model contains four superelements. The component modes calculation demonstrated in the following content is about one superelement. If there are more than one superelements, repeat the calculation for each. The calculation of fixed-boundary modes (Craig-Bampton method) is an eigenvalue extraction of the inner DOFs. The eigenvalue extraction can be expressed as:

$$\left([K]_{oo} - \lambda_{\text{eig}}[M]_{oo}\right)\{\phi_o\} = \{0\} \tag{2.4}$$

In the upper equation, λ_{eig} is the eigenvalue and $\{\phi_o\}$ is the mode shape of inner-DOFs.

Since component modes compensate for the dynamic effects of inner DOFs, which are lost after the static condensation, the number of component modes extracted can influence the accuracy of dynamic analysis. The number of modes needed for a dynamic reduction is a trade-off between efficiency and accuracy. More modes mean higher accuracy but lower efficiency. The acquired component modes can be expressed as:

$$\begin{gathered}[\Phi_o] = [\{\phi_o\}_1, \{\phi_o\}_2, \{\phi_o\}_3, \cdots, \{\phi_o\}_n] \\ [G_{oq}] = [\Phi_o]\end{gathered} \tag{2.5}$$

In the upper equation, $[G_{oq}]$ can be called the dynamic transformation matrix. The subscript "n" is the number of acquired modes.

The next step is the modal-space projection of inner-DOFs to component-modes DOFs. $[K_{qq}]$ is the stiffness matrix of component-modes DOFs. $[M_{qq}]$ is the mass matrix of component-modes DOFs. The projection of damping matrices is similar.

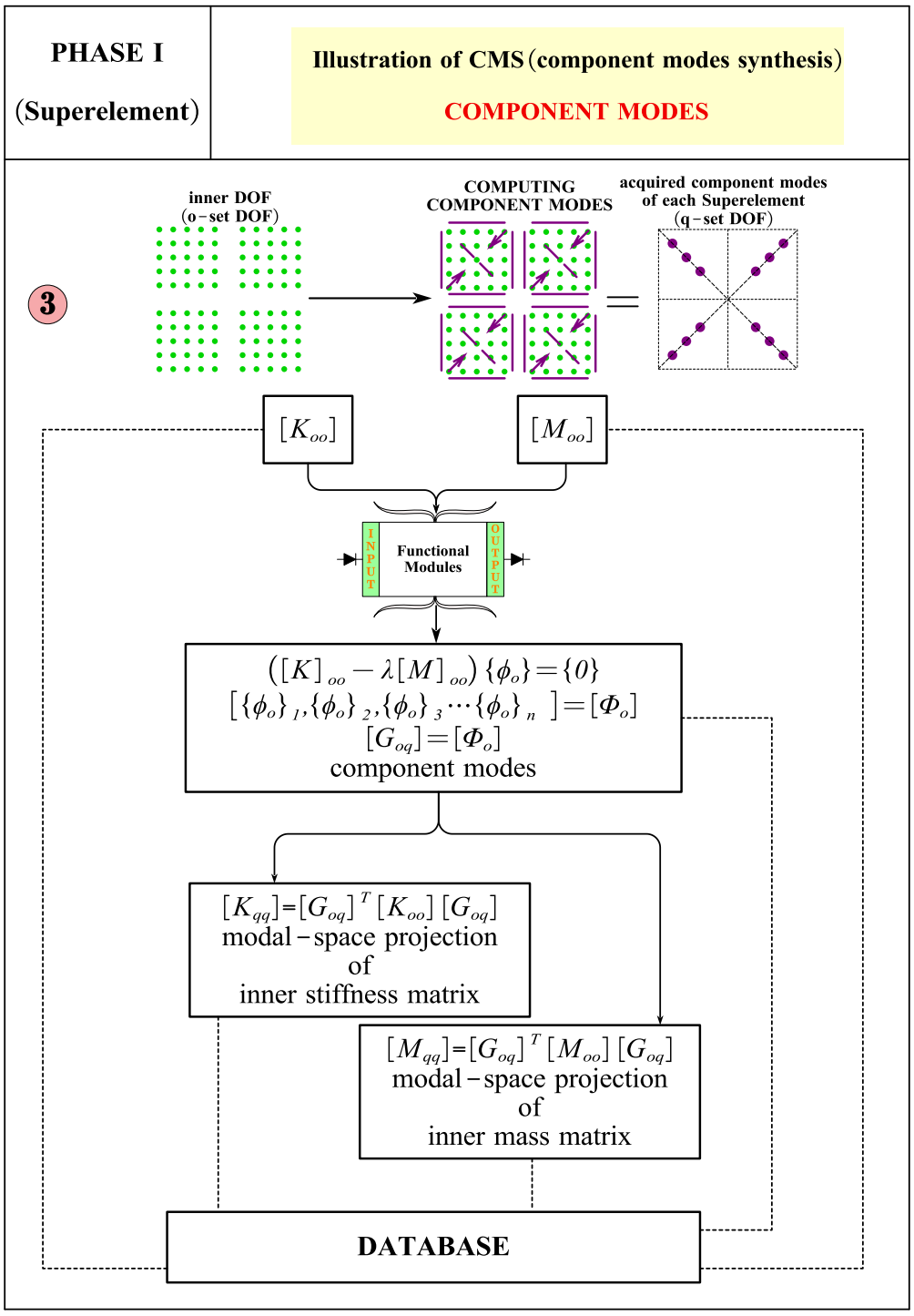


Figure 3.11: Computation of each superelement's component modes.

3.6.5 Component Modes Synthesis

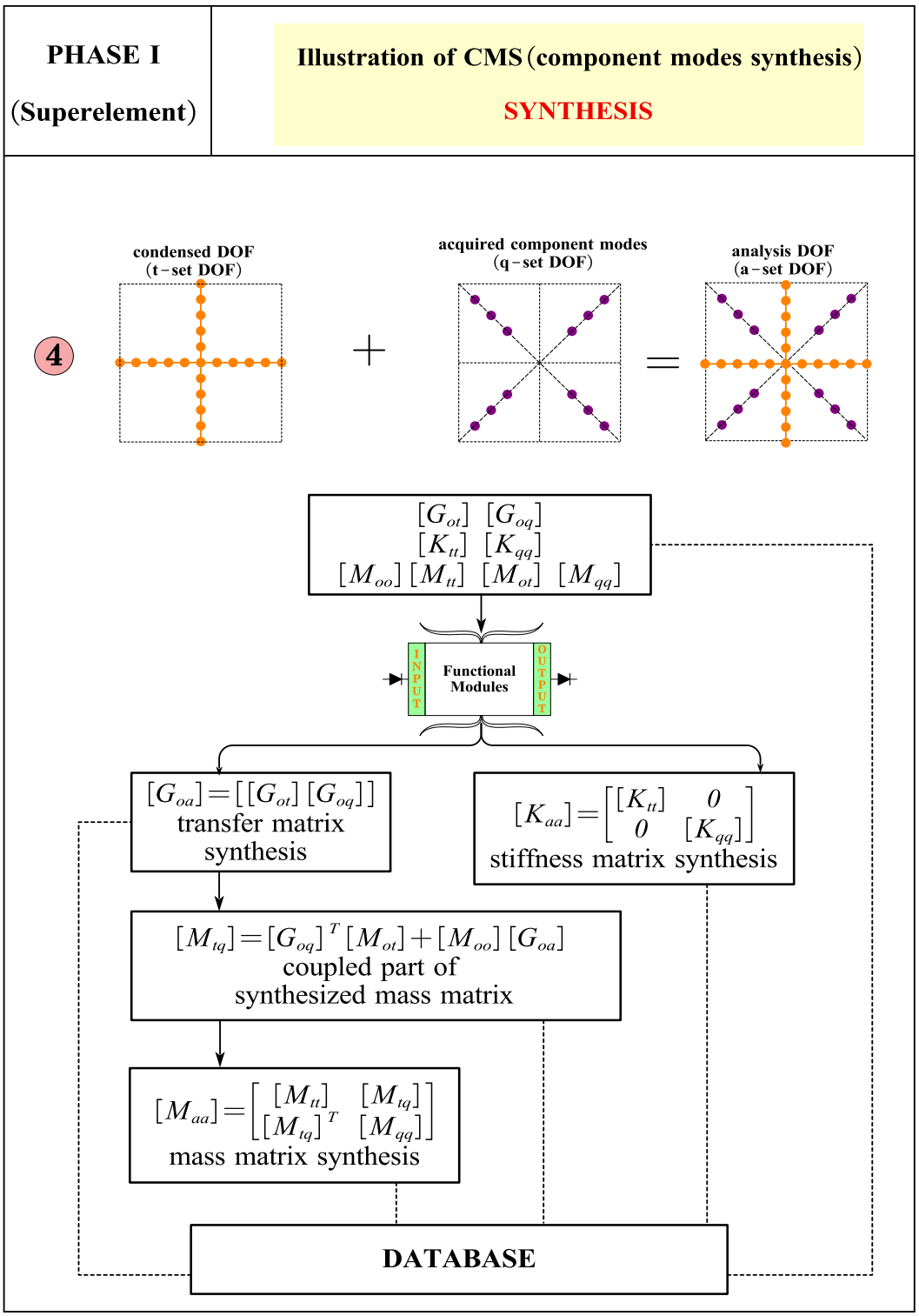


Figure 3.12: Component modes synthesis in superelements processing.

The section marked with the number “4” in a red circle in **Figure 3.12** means the component modes synthesis is the fourth step of superelement processing. The analysis DOFs, indicated by the letter “a”, consisting of the condensed-boundary DOFs, indicated in the letter “t”, and the component-modes DOFs, indicated in the letter “q”.

The total boundary transformation matrix, in definition, is the synthesis of the boundary-DOFs transformation matrix and the dynamic transformation matrix. Thus, the total boundary-DOFs transformation matrix can be expressed as:

$$[G_{oa}] = [[G_{ot}]\,[G_{oq}]] \tag{2.6}$$

The analysis-DOFs stiffness matrix can be expressed as:

$$[K_{aa}] = \begin{bmatrix} [K_{tt}] & 0 \\ 0 & [K_{qq}] \end{bmatrix} \tag{2.7}$$

The analysis-DOFs mass matrix can be expressed as:

$$[M_{aa}] = \begin{bmatrix} [M_{tt}] & [M_{tq}] \\ [M_{tq}]^{T} & [M_{qq}] \end{bmatrix} \tag{2.8}$$

In the upper equation, the part $[M_{tq}]$ is a coupling matrix between the condensed-boundary DOFs and the component-modes DOFs. This coupling matrix can be calculated as follows:

$$[M_{tq}] = [G_{oq}]^{T}\,[M_{ot}] + [M_{oo}]\,[G_{oa}] \tag{2.9}$$

3.6.6 Pros and Cons of Superelement Techniques

The core of processing superelements at “Phase I” is matrix reduction. A static analysis needs a static reduction, while a dynamic analysis needs a dynamic reduction. Subsection **3.6.1** introduces the difference between static and dynamic reductions. A static reduction for a static analysis brings exact solutions, while a dynamic reduction for a dynamic analysis brings approximated solutions.

The efficiency improvement in the matrix-equation solving process is the reduced matrices, which introduces the cost-saving term in “Phase II”. However, calculations of the inner-DOFs inverse stiffness matrix $[K_{oo}]^{-1}$ and component modes extractions are two extra costs of using superelement techniques. Therefore, superelement techniques are trade-off methods.

3.7 The Simulation Phase II

Calculations based on matrices start from “Phase II”. “Phase II” is a solving stage in Nastran. Matrices processed at this stage come from “Phase I”, and they are reduced if “Phase I” adopts superelement techniques or non-reduced if “Phase I” not adopts superelement techniques. A solving process of matrix equations may conduct in physical or modal space.

3.7.1 Modal-Space-Based Technical Route

A modal-space-based technical route needs a prior step to compute the eigenmodes as projecting basses, and this step is the real eigenvalue extraction (see subsection **3.7.3**). A modal-space-based projection can condense physical DOFs to modal DOFs, and the number of modal DOFs equals the number of selected eigenmodes. Objects in a modal-space-based projection include mass matrix, stiffness matrix, damping matrix, and loading vectors. These matrices or vectors can be non-reduced if “Phase I” does not adopt superelement techniques or reduced if “Phase I” adopts superelement techniques.

As for the modal-space-based projection, models without superelements undergo a one-stage DOFs condensation (see **Figure 3.5**). In comparison, models with superelements undergo a two-stage DOFs condensation. The first stage is a static or dynamic reduction at “Phase I” (static reduction for static analysis, dynamic reduction for dynamic analysis), and the second stage is a modal-space-based projection at “Phase II” (see **Figure 3.7**).

A modal-space-based technical route has some trade-off points regarding cost and saving in calculations. In a situation that models without superelements, the eigenmode extraction procedure is the extra cost before solving matrix equations. At the same time, saving is the modal matrix with a small number of modal DOFs. Therefore, there is a trade-off between the cost of the eigenmode extraction procedure and the saving in solving matrix equations with small modal DOFs.

In the situation that models with superelements, there are two trade-off points. Since a smaller number of physical DOFs bring less cost in the eigenmode extraction procedure. A noticeable decrease in physical DOFs can be accomplished by superelement reduction before the eigenmode extraction procedure. Still, the superelement reduction is a cost. Therefore, here is the first trade-off between the cost of superelement reduction and the saving in the eigenmode extraction procedure due to reduced physical DOFs. Then, the second trade-off is between the cost of the eigenmode extraction procedure and the saving in solving matrix equations with small modal DOFs.

Since finite-element models generate positive diagonal mass matrices and positive symmetric stiffness matrices, the extracted eigenmodes can condense and uncouple mass and stiffness matrices. However, the eigenmodes may not uncouple damping matrices. If damping matrices are mass-proportional or stiffness-proportional (Rayleigh damping or structural damping), they can be condensed and uncoupled. If not, damping matrices can only be condensed. The arguments in this paragraph are based on [25] and [26].

After a modal-space-based condensation, matrices may be uncoupled or coupled. The uncertainties may depend on the form of the damping matrix, the unsymmetric stiffness or the

unsymmetric mass matrix. Thus, a modal-space-based calculation at "Phase II" may solve an uncoupled system or coupled system with modal DOFs. An uncoupled system consists of linearly independent equations, and calculations about these independent equations are algebraic operations. In contrast, a coupled system must call specific solvers (see section **3.8**). Nastran automatically recognises and matches the correct solver for the coupled or uncoupled system. Users do not need to do extra settings.

3.7.2 Physical-Space-Based Technical Route

DOFs of finite-element models are always coupled. A physical-space-based technical route automatically calls the solvers to solve matrix equations with coupled DOFs (see section **3.8** for more details).

In physical-space-based technical routes, models without superelements undergo no DOFs condensations (see subsection **3.4.1** and **Figure 3.4**). In comparison, models with superelements undergo a one-stage DOFs condensation. It is a static or dynamic reduction at "Phase I" (see subsection **3.4.3** and **Figure 3.6**).

In the situation that models with superelements, there is one trade-off point. A smaller number of physical DOFs bring less cost in matrix-equation calculations. And a noticeable decrease in physical DOFs can be accomplished by superelement reduction. Still, the superelement reduction is a cost. Therefore, here is a trade-off between the cost of superelement reductions and the saving in matrix-equation calculations due to reduced physical DOFs.

3.7.3 Real Eigenvalue Extraction

One of the important analyses in simulation is real eigenvalue extraction. And it is a simulation procedure in "Phase II". As the products, the extracted eigenmodes are modal projection bases of modal-space-based analyses. In practice, the Lanczos method shows superior performance for models with enormous DOFs or massive eigenmodes that need to be calculated, and this method is the default method in Nastran. The cost of a real eigenvalue analysis depends on the size of matrices and the number of acquired eigenmodes. By the way, the arguments in this subsection are based on [27].

3.8 Solvers for Solving Coupled Matrix Equations

"Phase II" is a stage for solving matrix equations of different simulation tasks, such as static analyses, frequency response analyses, transient response analyses, buckling analyses, etc. Matrix equations can be solved through the direct or the iterative method. For simplification, a matrix equation [28] can be expressed as:

$$[A]\{x\}=\{b\} \tag{2.10}$$

In the upper equation, $[A]$ is a general matrix (real or complex), $\{x\}$ is a general responses vector, $\{b\}$ is a general loads vector.

3.8.1 Direct Solver

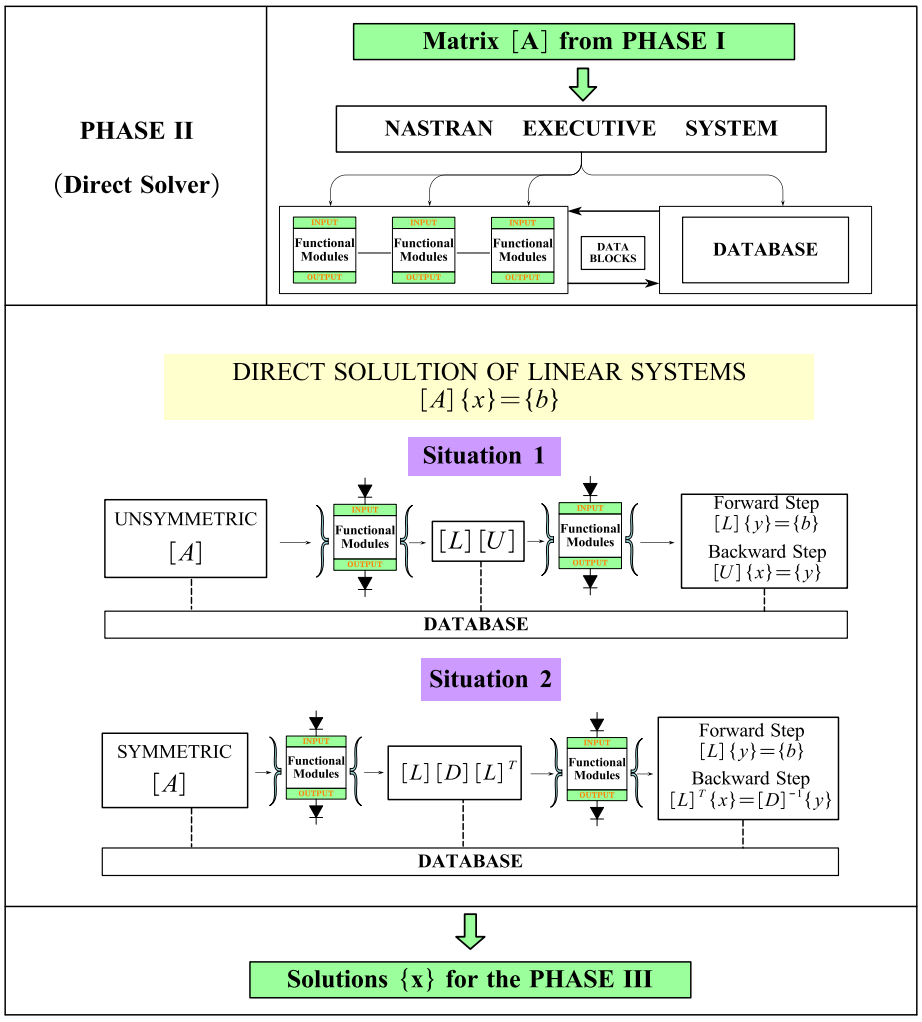


Figure 3.13: Solving logic of Nastran's direct solver.

It is necessary to note that the deductions in this subsection are based on [29]. Then, we can understand that the direct method relies on a matrix decomposition as the first step and follows a forward-backwards substitution as the second step. **Figure 3.13** illustrates the direct solution procedure in two different situations. The matrix $[A]$ is unsymmetric in "Situation 1". After decomposition, the matrix $[A]$ becomes a product of lower triangular matrix $[L]$ times an upper triangular matrix $[U]$,

$$[A]=[L][U] \tag{2.11}$$

The forward-backwards substitution displayed in "Situation 1" is the right-handed form (a left-handed form has equivalent effects, for simplicity, this paper does not stretch it). The forward step calculates the intermediate solutions by the lower triangular matrix $[L]$ and the load vector $\{b\}$, then stores the intermediate results in the vector $\{\mathrm{y}\}$. The backward step calculates the response vector $\{x\}$ through the intermediate vector $\{\mathrm{y}\}$ and the upper triangular matrix $[U]$. At the same time, the matrix $[A]$ is symmetric in "Situation 2". After decomposition, the matrix $[A]$ becomes a product of a lower triangular matrix $[L]$, the transposed lower triangular matrix $[L]^{\mathrm{T}}$, and a diagonal matrix $[D]$,

$$[A]=[L][D][L]^{\mathrm{T}} \tag{2.12}$$

The forward step is the same as in "Situation 1". The backward step calculates the responses vector $\{x\}$ through the intermediate vector $\{\mathrm{y}\}$, the inverse of the diagonal matrix $[D]^{-1}$, and the transposed lower triangular matrix $[L]^{\mathrm{T}}$.

The highest computing cost of the direct method happens at the matrix decomposition step. In comparison, the time needed for the forward-backwards substitution step is much lower, especially for models with immense DOFs. For more information about computing costs, the in-core memory requirements, and the out-core memory requirements, please see the related content in the reference guide.

3.8.2 Iterative Solver

It is necessary to note that the deductions in this subsection are based on [30]. Then, we can understand that the iterative method gives approximated solutions by using an iteration loop. This method does not need extra transformations to the original matrices. The iteration loop aims to find the minimum value of the error function by controlling the convergence criteria and convergence process. **Figure 3.14** illustrates the basic concepts of the iterative method adopted in Nastran. The error function is a gap between the left side and right side of the matrix equation,

$$Err(x)=\frac{1}{2}\{x\}^{\mathrm{T}}[A]\{x\}-\{x\}^{\mathrm{T}}\{b\} \tag{2.13}$$

The minimum value of the error function can be found by minimising the value of the gradient function,

$$\left\{\frac{dErr}{dx}\right\}=[A]\{x\}-\{b\}=-\{res\} \tag{2.14}$$

In the upper equation, $\{res\}$ is called the residual vector.

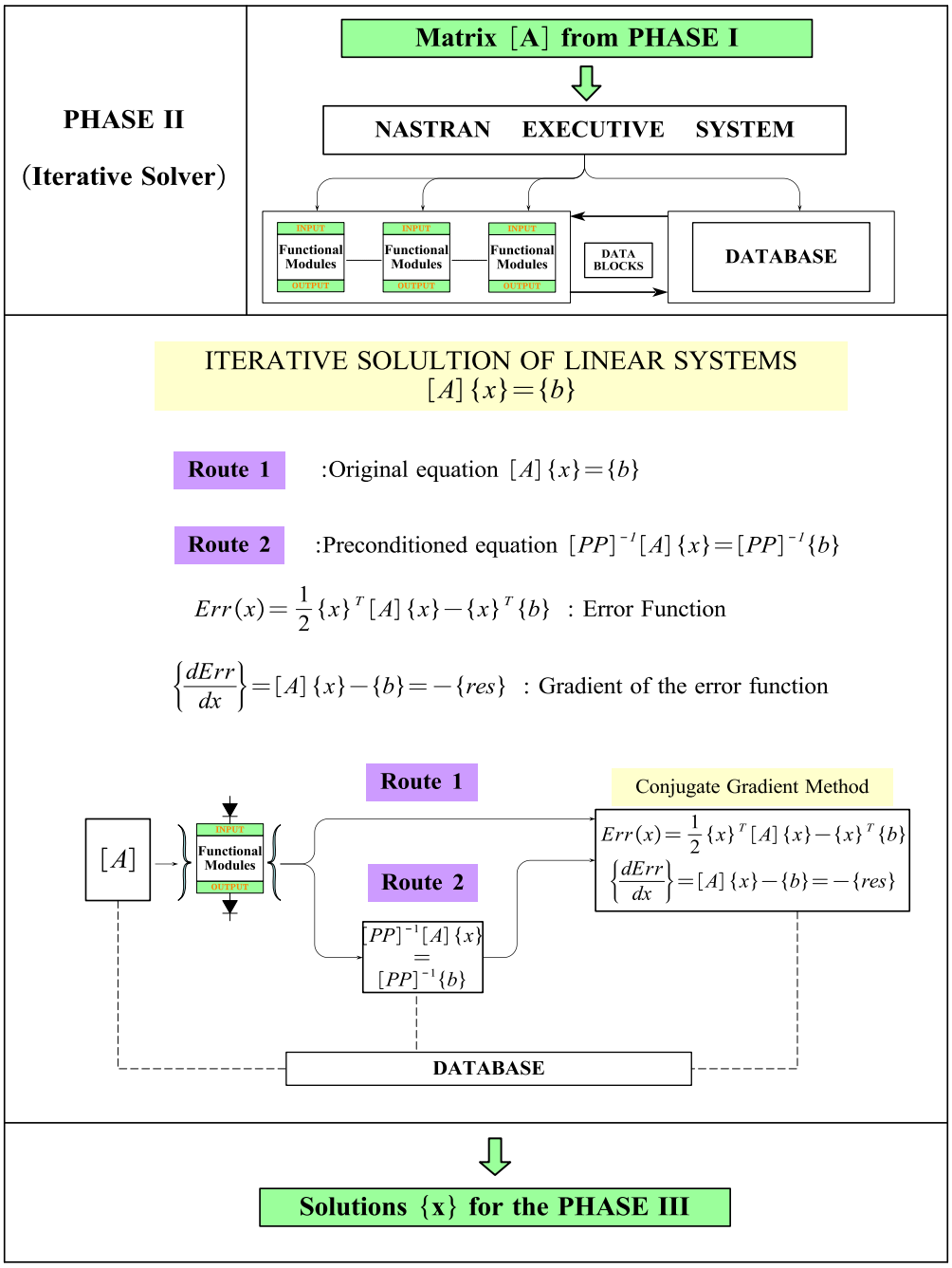


Figure 3.14: Solving logic of Nastran's iterative solver.

The iteration may process the original matrices or preconditioned matrices. In **Figure 3.14**, "Route 1" indicates the convergence process of the gradient function based on the original matrices; "Route 2" indicates the convergence process of the gradient function based on the preconditioned matrices. The matrix $[PP]$ is the preconditioner. The preconditioning for

matrices can reduce the number of iterations but increase the work in each iteration. There are many methods to set the preconditioner. This paper does not stretch the preconditioning methods. For further information, please see the reference guide.

Figure 3.15 illustrates the iteration process for the convergence of the gradient function based on original matrices. "Threshold 1" is an inequality that judges whether the ideal convergence criteria are achieved within a given number of iterations. This threshold is expressed as:

$$\frac{\|\{res\}_i\|}{\|\{b\}\|} < \varepsilon_{\text{thresh}} \tag{2.15}$$

In the upper inequation, $\{res\}_i$ denotes the residual vector after i-times iterations and $\varepsilon_{\text{thresh}}$ is a user-defined value of the upper limit. If the ideal convergence criteria cannot achieve after the maximum number of iterations, "Threshold 2" supplies accepted convergence criteria as an alternation. This threshold can be expressed as:

$$\frac{|\{x\}_i^T\{res\}_i|}{|\{b\}^T\{x\}_i|} < \varepsilon_{\text{thresh}} \tag{2.16}$$

In the upper inequation, $\{x\}_i$ denotes the guessed results after i-times iteration. The updating loop of the residual vector $\{res\}_i$ and the vector of guessed results $\{x\}_i$ are shown in **Figure 3.15**. An updating process needs two controlling parameters $\{pp\}$ and $\alpha\alpha$, and one intermediate parameter $\beta\beta$. $\{pp\}$ is the direction vector, and after i-times iteration, it can be expressed as:

$$\{pp\}_i = \{res\}_i + \beta\beta_i\{p\}_{i-1} \tag{2.17}$$

$\alpha\alpha$ is the distance parameter, and after i-times iteration, it can be expressed as:

$$\alpha\alpha_i = \frac{\{res\}_i^T\{res\}_i}{\{pp\}_i^T[A]\{pp\}_i} \tag{2.18}$$

$\beta\beta$ is the intermediate parameter for the calculation of the direction vector, and after i-times iteration, it can be expressed as:

$$\beta\beta_i = \frac{\{res\}_i^T\{res\}_i}{\{res\}_{i-1}^T\{res\}_{i-1}} \tag{2.19}$$

With the assistance of the two controlling parameters, the updated residual vector can be expressed as:

$$\{res\}_{i+1} = \{res\}_i - \alpha\alpha_i[A]\{p\}_i \tag{2.20}$$

And the updated results can be expressed as follows:

$$\{x\}_{i+1} = \{x\}_i + \alpha\alpha_i \{pp\}_i \tag{2.21}$$

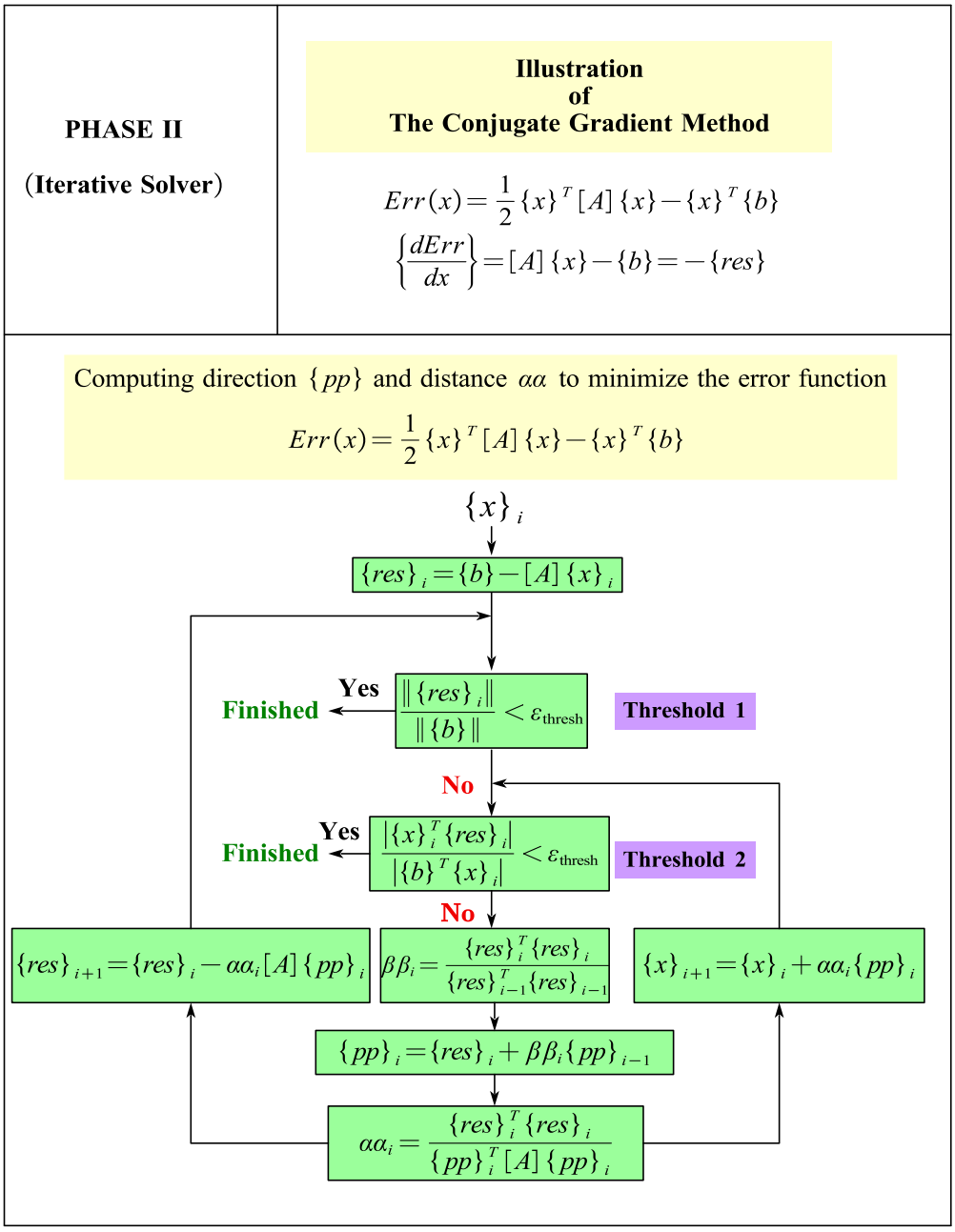


Figure 3.15: Convergence criteria of Nastran's iterative solver.

3.8.3 Direct Method vs Iterative Method

The direct method is the default method in Nastran for solving matrix equations. This method has a few limitations. One thing that needs to be noted is the matrix decomposition procedure.

It is the main cost in calculation through the direct method. By the way, the arguments in this paragraph are based on [29].

The iterative method is an alternative, but this method only supports several simulation schemes in Nastran. Such as the linear static analysis (SOL 101), the nonlinear static analysis (SOL 106), the physical-space-based steady-state dynamic analysis (SOL 108), and the modal-space-based steady-state dynamic analysis (SOL 111). The cost of the iterative method is hard to predict because of the unpredictable number of iterations. The settings and controls of the iterative method are more complex than the direct method. Lastly, the arguments in this paragraph are based on [30].

The iterative method is not the default method. Users can change the default setting through the Nastran input file. Adjustments at two different subsections of an input file are needed to activate and control the iterative method. The control of the iterative method must cooperate with the entry "SMETHOD" in the "Case-Control Section" and the entry "ITER" in the "Bulk Data Section" (see **Figure 3.1**). **Table 3.1** illustrates an example of the settings for using the iterative method. For more information about the meaning of these settings, please see the corresponding section in the reference guide [31] and [32].

Table 3.1: Setting for activating Nastran's iterative solver.

section		Entry
Nastran Statement section		
File Management section		
Executive Control Section		
Case-Control Section		SMETHOD = SID
Bulk Data Section	Parameters	
	Control	
	Bulk Data	ITER, SID , PRECOND = char, CONV = char, , MSGFLG = char, ITSEPS = real, ITSMAX = int , IPAD = int, IEXT = int, PREFONLY = int ("char": character type, "int": integer, "real": real number)

One of the destinations of this paper is to find an efficient method to analyse metamaterial. The best choice needs support from assessments of several testing simulations' performance. These testing simulations are independent analyses with different technical routes (see chapters **14**, **15**, **16**, and **17**). For reliable comparisons between these testing simulations, each step in these simulations must be predictable. Therefore, this paper uses the direct solver in "Phase II" to solve the matrix equations.

3.9 The Simulation Phase III

Operations in “Phase III” are succeeding processes for recovering and extracting the acquired data from “Phase II” solutions [12]. The acquired data outputs need specific statements in an input file’s “Case-Control Section” (see section **3.1**). Four cases need different operations in “Phase III” to extract the acquired results.

The first case is a physical-space-based simulation without superelement techniques. **Figure 3.16** illustrates the “Phase III” operations to extract acquired results directly, which are physical-DOFs results.

The second case is a modal-space-based simulation practice without superelement techniques. **Figure 3.17** illustrates the operations in “Phase III”. These operations include recovering data from modal DOFs to physical DOFs, extracting the acquired physical-DOFs results, or directly extracting the modal-DOFs results. The physical DOFs recovery from the modal DOFs is an inverse operation of modal-space-based condensation. By the way, the arguments in this paragraph are based on [26].

The third case is a physical-space-based simulation practice with superelement techniques. **Figure 3.18** illustrates the operations in “Phase III”. For models with superelements, the calculating object in “Phase II” is the reduced matrices. Thus, “Phase III” operations recover data from boundary DOFs to inner DOFs of acquired superelements. If more than one superelements need data recovery, a recovering loop is needed to accomplish this execution. The sequence of superelements data recovery in “Phase III” is inverse to superelements generation in “Phase I”. By the way, the arguments in this paragraph are based on [12].

The fourth case is a modal-space-based simulation practice with superelement techniques. **Figure 3.19** illustrates the operations in “Phase III”. The first step is to recover the physical results of the reduced model from the modal results. Then, the second step recovers data from boundary DOFs to inner DOFs of acquired superelements. If more than one superelements need data recovery, a recovering loop is needed to accomplish this execution. The sequence of superelements data recovery in “Phase III” is inverse to superelements generation in “Phase I”. By the way, the arguments in this paragraph are based on [12] and [26].

Output data from “Phase III” can be directly read or displayed in “Phase IV”. The cost of “Phase III” depends on the amount of output data acquired. More output needs more calculation time. As a post-process stage, “Phase IV” corresponds to the results display and presentation and shows structure deformation. “Phase IV” is not a typical simulation stage; therefore, this paper does not introduce more.

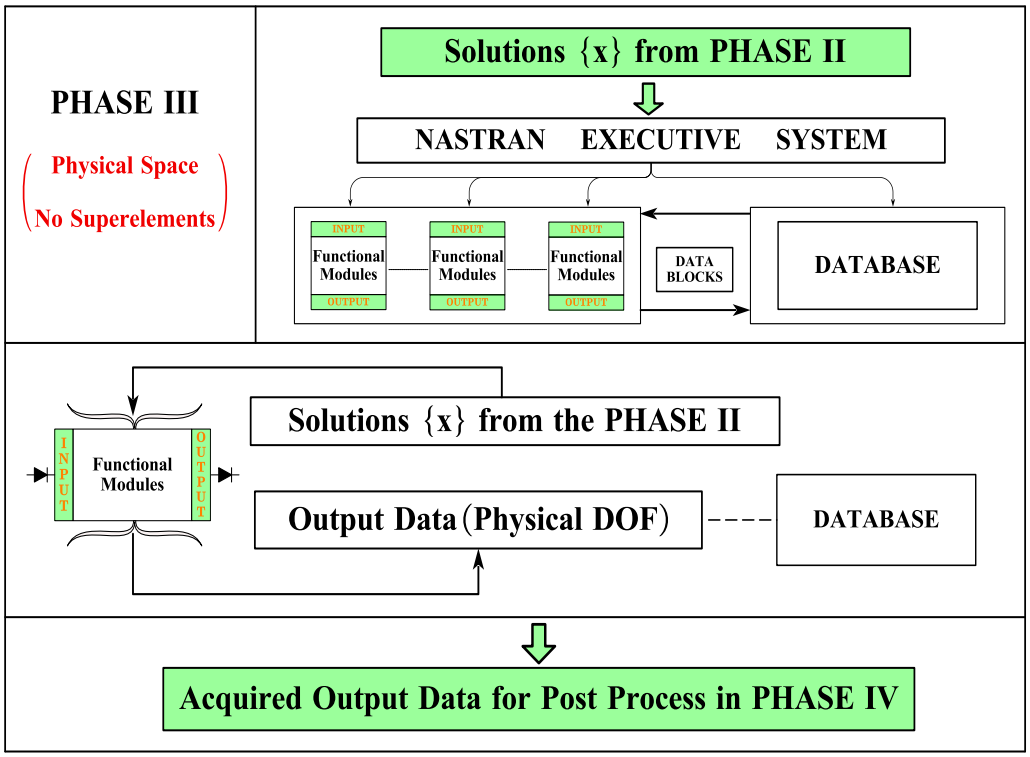


Figure 3.16: Physical data output from Nastran's "Phase III" (no data recovery).

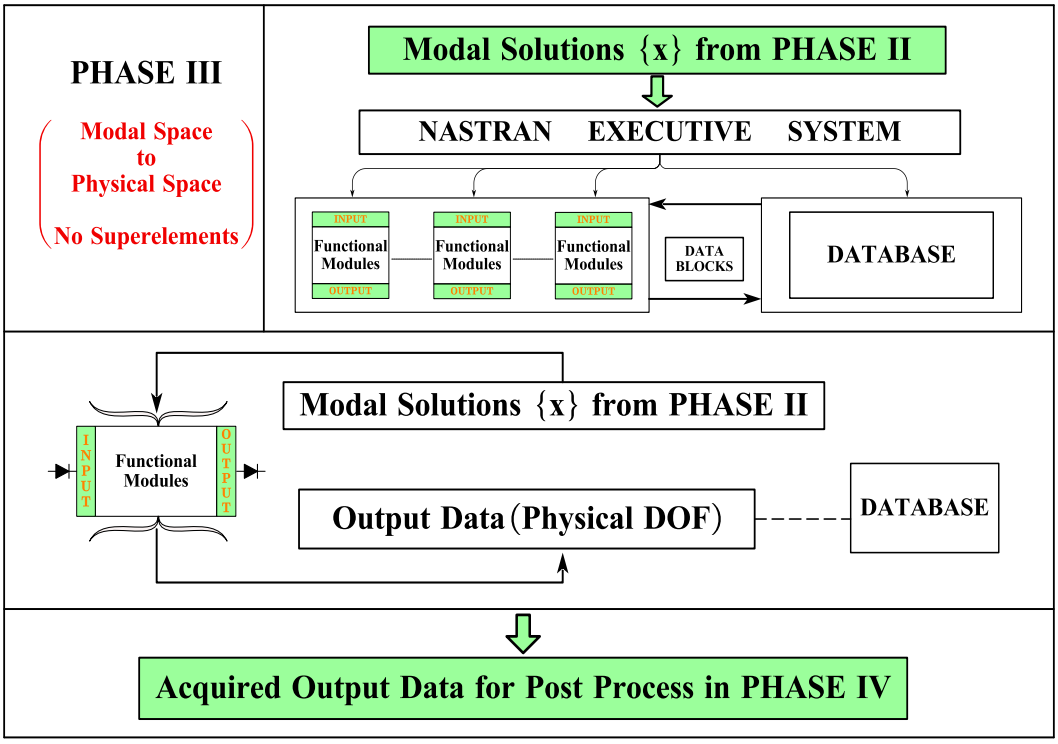


Figure 3.17: Modal data recovers into physical data in Nastran's "Phase III".

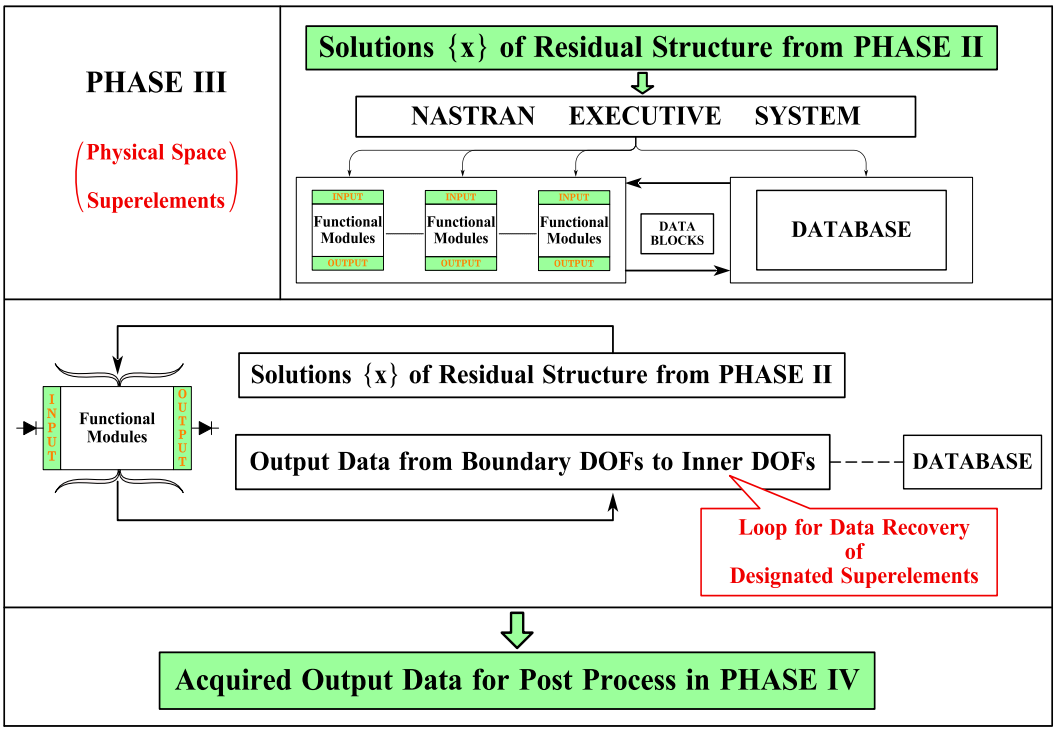


Figure 3.18: Residual structure's reduced data recovers into full data in Nastran's "Phase III".

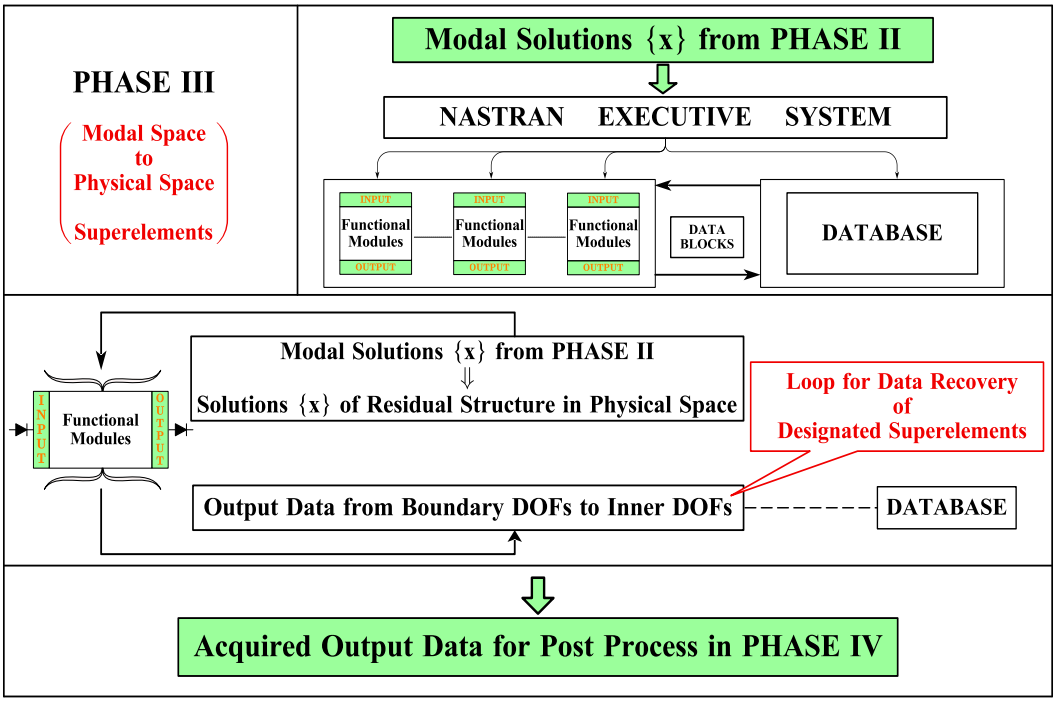


Figure 3.19: Modal data recovers into reduced data, then into full data in Nastran's "Phase III".

3.10 Simulation Designs through Nastran

3.10.1 Characteristics of Acoustic Metamaterial Samples

The panel-like metamaterial sample is a composite material consisting of periodic cubes. A cube is the unit cell of the system, and each cube consists of an aluminium core and a melamine seat. As part of the unit cell, the aluminium core has an isotropic inner structure and non-frequency-dependent elasticity. Then, the melamine seat has an orthotropic inner structure and frequency-dependent viscoelasticity.

As described before, this paper aims to study the metamaterial mechanical properties and find an efficient way to accomplish simulation tasks; therefore, a simulation design needs to consider intrinsic material properties and software capabilities. There are three focal points for a study of the metamaterial sample:

- massive DOFs (degrees of freedom),
- periodic structure,
- frequency-dependent viscoelasticity.

The periodic structure has a cost-saving potential with the clone tactic in processing (copying numerical information from one unit cell to others). In contrast, the massive DOFs and frequency-dependent viscoelasticity are two cost-increasing potentials in calculations.

3.10.2 Methods to Treat Massive DOFs

A metamaterial may have massive DOFs due to the discretisation of considerable substructures. The massive DOFs are a heavy burden in matrix-equation calculations in “Phase II”. Four methods in Nastran can reduce the number of DOFs before calculating matrix equations.

The first method is the dynamic reduction in physical space (see **Figure 3.6**). This method reduces DOFs from the whole structure to the reduced substructures. An extra cost of this method is superelements processing at “Phase I”, and an expected saving is the reduced DOFs of substructures.

The second method is the modal-space-based condensation without superelement techniques (see **Figure 3.5**). This method reduces the original physical DOFs to modal DOFs. An extra cost of this method is the eigenvalue extraction procedure at “Phase II”, and an expected saving is the reduced modal DOFs.

The third method is the modal-space-based condensation combining superelement techniques (see **Figure 3.7**). This method goes through two-stage reductions. Thus, there are two extra costs but two additional efficiency improvements. The extra costs are the superelements processing in “Phase I” and the eigenvalue extraction in “Phase II”. As for the efficiency improvement, the first one is the reduced model in extracting eigenvalues. And the second one is the small modal DOFs in solving matrix equations.

The fourth method is called the “Domain Solver” with the “ACMS Method” (see subsections **3.5.1** and **3.5.2**). This method relies on an advanced software architecture in Nastran to drive the parallel calculation based on the superelement independency. It is

dedicated to the eigenvalue extraction analysis. Thus, it is only proper for modal-space-based analyses. This method is analogous to the third method running with advanced software architecture. And the additional frequency improvement is the parallel calculation.

3.10.3 Methods to Treat the Periodic Structure

The periodic structure consists of congruent unit cells. Therefore, the "Image Superelement Method" could be a proper choice (see subsections **3.5.1** and **3.5.2**). However, this method costs more in the data recovery process. Thus, its performance needs testing to verify. Another choice is the "List Superelement Method".

3.10.4 Methods to Treat the Frequency Dependency

The melamine foam has an orthotropic inner structure. Within the inner structure are nine orthotropic elastic moduli. The nine orthotropic elastic moduli are frequency-dependent parameters with the same frequency function. In a frequency domain, the values of the nine elastic moduli are complex numbers. In these complex numbers, the real parts represent elasticity, while the imaginary parts represent structural damping. Viscoelasticity means that elastic properties have viscous effects. And the viscous effects are frequency-dependent parameters.

Dynamic analysis may solve a property-unchangeable or property-changeable system. A property-unchangeable system does not update its matrices regarding the changed frequencies. Thus, each new frequency will not activate the model rebuilding mechanism. In comparison, a property-changeable system updates its matrices regarding the changed frequencies. In this case, each new frequency will activate the model rebuilding mechanism.

Nastran supplies two patterns for defining frequency-dependent properties. One is the "Classic Frequency-Dependent Material" pattern, and the other is the "Viscoelastic Material" pattern. The "Classic Frequency-Dependent Material" pattern is more general and flexible. Users can independently set frequency-dependent rules for each elastic moduli, structural damping factor, or density through this pattern. In comparison, the "Viscoelastic Material" pattern is not as adjustable as the previous one. Users can use this pattern to set frequency-dependent scaling factors to stiffness and structural damping matrices instead of elastic moduli or structural damping factors. For more information about the mathematical models and fundamental theories of the "Classic Frequency-Dependent Material" pattern and the "Viscoelastic Material" pattern, please see subsections **7.2.2**, **7.2.3**, **7.2.4**, and **7.2.5**. The following contents are the logical layout of the two frequency-dependent material patterns in a dynamic analysis. In concert with **Figure 3.20**, **Figure 7.1** in subsection **7.2.2** gives a more specific illustration of the "Classic Frequency-Dependent Material" pattern, and **Figure 7.2** in subsection **7.2.4** gives a more specific illustration of the "Viscoelastic Material" pattern.

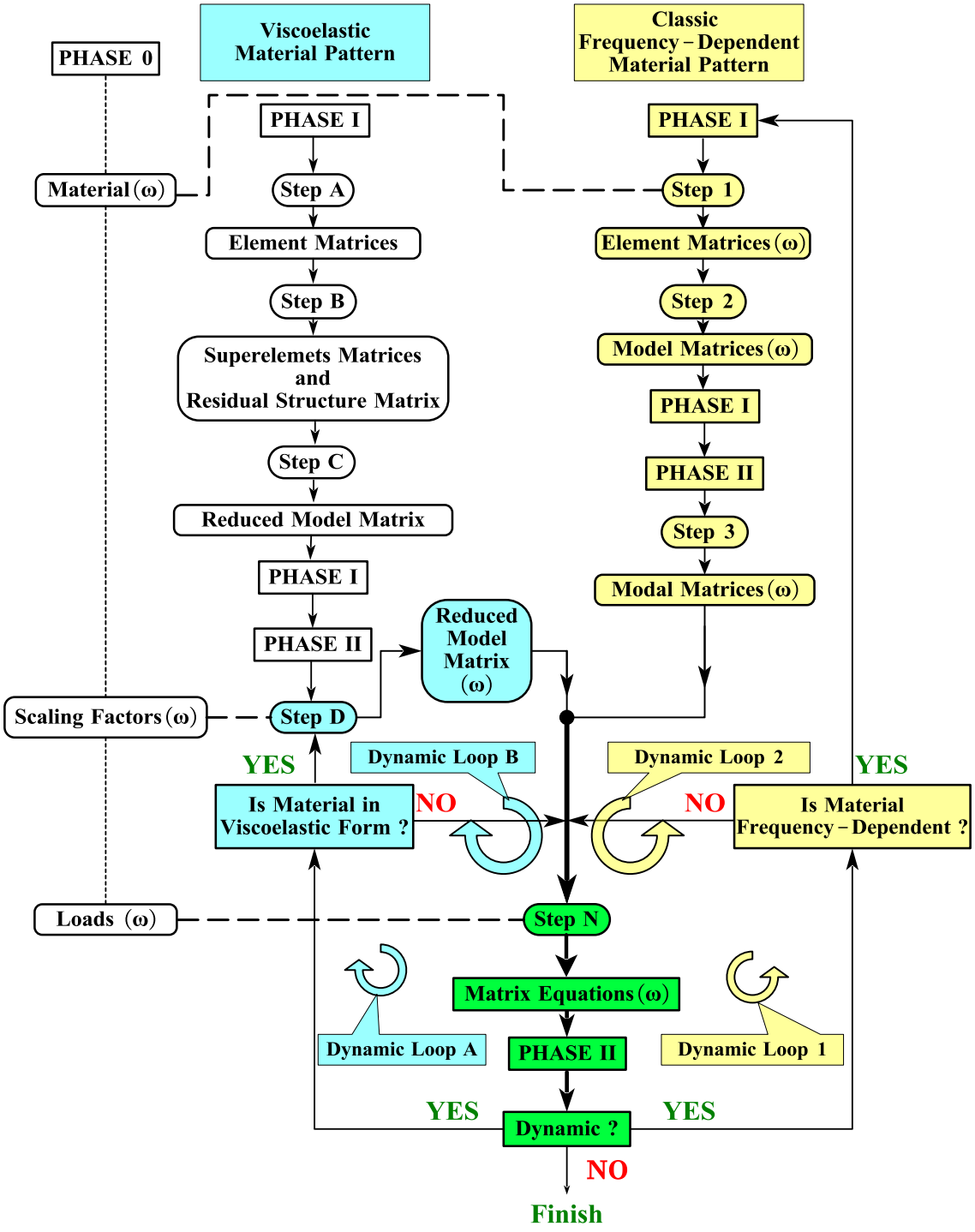


Figure 3.20: Model rebuilding mechanisms due to frequency dependency.

Figure 3.20 illustrates a frequency response analysis of a system by schematic boxes, directional arrows, and dashed lines. The schematic boxes are classified into four categories and shaded with four colours. In the first place, the left-side region contains operation

procedures of the "Viscoelastic Material" pattern. The contents in white-shading schematic boxes do not participate in any matrix-rebuilding loop, while those in blue-shading schematic boxes participate in the matrix-rebuilding loop. In the second place, the right-side region contains operation procedures of the "Classic Frequency-Dependent Material" pattern. The contents of yellow-shading schematic boxes participate model rebuilding loop. In the third place, the middle-bottom region displays common procedures of the "Viscoelastic Material" pattern and the "Classic Frequency-Dependent Material" pattern. The procedures are in the green-shading schematic boxes. Between these schematic boxes are directional arrows or dashed lines. The directional arrows show the loops of technical routes and the sequence of operating procedures. The dashed lines connect the driving parameters and the driven operations.

Figure 3.20 shows two processing routes corresponding to two frequency-dependent material patterns in a frequency response analysis. The two technical routes undergo the simulation "Phase 0", "Phase I", and "Phase II". When users define an input file, "Phase 0" translates this input file into inner instructions. The inner instructions define the material patterns and control their processing. Under the guidance of the inner instructions, "Phase I" executes matrix generations and reductions. Then, "Phase II" solves matrix equations. The operations in "Phase I" and "Phase II" may be repeated under a matrix or model rebuilding loop. Within the logic of the "Viscoelastic Material" pattern, the matrix rebuilding loop is activated after "Phase I" and before "Phase II". In comparison, within the logic of the "Classic Frequency-Dependent Material" pattern, the model rebuilding loop is activated that includes "Phase I" and "Phase II".

3.10.5 The Classic-Frequency-Dependent Material Pattern

The "Classic Frequency-Dependent Material" pattern undergoes the processing route represented by the yellow-shading schematic boxes. In dynamic analyses, two dynamic loops may be activated. The triggering condition of "Dynamic Loop 1" is that the model adopts "Non-Frequency-Dependent Material". The "Dynamic Loop 1" contains "Step N" in "Phase II". In contrast, the triggering condition of "Dynamic Loop 2" is that the model adopts frequency-dependent materials. The "Dynamic Loop 2" contains "Step 1" and "Step 2" in "Phase I" and "Step 3" and "Step N" in "Phase II".

On the right side of **Figure 3.20**, the yellow-shading schematic box "Is Material Frequency-Dependent?" judges whether to activate the "Frequency-Dependent Material" pattern. If this pattern is activated, the material frequency dependency "Material(ω)" from "Phase 0" will instruct "Step 1" to regenerate element matrices. Then, "Step 2" assembles the element matrices into a model matrix. These two procedures belong to "Phase I". After that, "Step 3" in "Phase II" may execute an eigenvalues extraction procedure. Then, a modal-space projection may repeat each time with each new frequency. Finally, the green-shading schematic box "Step N" in "Phase II" rebuilds and solves the updated matrix equations.

The "Classic Frequency-Dependent Material" pattern initiates a material-updating procedure in "Phase I" to regenerate element matrices and assemble them into a model matrix. The "Classic Frequency-Dependent Material" pattern can be adopted in a modal-space-based dynamic analysis since the updated matrices can get the updated eigenmodes. However, Nastran does not support the "Classic Frequency-Dependent Material" pattern to adopt the dynamic reduction technique. The reason is that repeating the dynamic reduction makes this

technique lose efficiency. If users set the "Classic Frequency-Dependent Material" pattern to superelement, the computation will be stopped by Nastran and prompt users to remove the setting.

The "Classic Frequency-Dependent Material" pattern in **Figure 3.20** gives an example. It is a modal-space-based frequency response analysis, and the processing procedures are displayed in a column consisting of yellow-shading and green-shading schematic boxes. The processing procedures include two simulation phases, four operating steps, and two dynamic loops.

Two simulation phases,

- "Phase I" (matrices regeneration and assembly),
- "Phase II" (modal space projection, rebuilding, and solving matrix equations).

Four operating steps,

- "Step 1" (element matrices generation) in "Phase I",
- "Step 2" (assembling element matrices into a model matrix) in "Phase I",
- "Step 3" (eigenvalues analysis and modal space projection) in "Phase II",
- "Step N" (rebuilding and solving matrix equations) in "Phase II".

Two dynamic loops,

- "Dynamic Loop 1" (material is not frequency-dependent, repeating "Step N" to rebuild and solve matrix equations),
- "Dynamic Loop 2" (material is frequency-dependent, repeating "Phase I" to regenerate matrices, repeating "Phase II" to rebuild and solve matrix equations).
- The "Classic Frequency-Dependent Material" pattern can be adopted in physical-space-based or modal-space-based frequency response analyses.

3.10.6 The Viscoelastic Material Pattern

The "Viscoelastic Material Pattern" undergoes the processing route represented by the blue-shading schematic boxes. In dynamic analyses, two dynamic loops may be activated. The triggering condition of "Dynamic Loop A" is that the model adopts "Non-Frequency-Dependent Material". The "Dynamic Loop A" contains "Step N" at "Phase II". In contrast, the triggering condition of "Dynamic Loop B" is that the model adopts frequency-dependent materials. The "Dynamic Loop B" contains "Step D" and "Step N" in "Phase II".

On the left side of **Figure 3.20**, a blue-shading schematic box "Is Material in Viscoelastic Form?" judges whether to activate the "Viscoelastic Material" pattern. If this pattern is activated, the frequency-dependent scaling factors "Scaling Factors (ω)" from "Phase 0" will instruct the blue-shading schematic box "Step D" to update stiffness and structural damping matrices. Then, the green-shading schematic box "Step N" rebuilds the matrix equations.

The "Viscoelastic Material" pattern initiates a material-updating procedure just before solving matrix equations in "Phase II". This processing route does not regenerate matrices but updates them with frequency-dependent scaling factors. A dynamic loop based on this

technique route does not include “Phase I”, which means operations in “Phase I” have only one-time execution. Therefore, “Phase I” with the “Viscoelastic Material” pattern can adopt the dynamic reduction (CMS of superelement techniques) to reduce the size of matrices but not repeat the dynamic reduction more than once. Nastran does not allow repeating the dynamic reduction because this operation makes superelement techniques lose efficiency. The “Viscoelastic Material” pattern cannot execute a modal-space-based dynamic analysis with frequency-dependent material [33]. The reason is that this material pattern does not repeat the eigenvalue extraction procedures. Then, the eigenmodes cannot be updated.

The “Viscoelastic Material” pattern defines frequency-dependent material properties to avoid the model rebuilding process [33]. Thus, a typical usage scenario of this material pattern is that models consist of frequency-dependent materials, and the analysis adopts the superelement techniques.

The “Viscoelastic Material Pattern” in **Figure 3.20** gives an example. It is a model with superelements using the “Viscoelastic Material” pattern. The processing procedures are displayed in a column consisting of white-shading, blue-shading, and green-shading schematic boxes. Within this pattern, the processing procedures include two simulation phases, five operating steps, and two dynamic loops.

Two simulation phases,

- “Phase I” (matrices generation, reduction, and assembly),
- “Phase II” (updating matrices by frequency-dependent scaling factors, rebuilding, and solving matrix equations).

Five operating steps,

- “Step A” (element matrices generation) in “Phase I”,
- “Step B” (assembling element matrices into superelement matrices and residual structure matrices) in “Phase I”,
- “Step C” (dynamic reduction to generate reduced a model matrix) in “Phase I”,
- “Step D” (updating stiffness and structural damping matrices by frequency-dependent scaling factors) in “Phase II”,
- “Step N” (rebuilding and solving matrix equations) in “Phase II”.

Two dynamic loops,

- “Dynamic Loop A” (material is not frequency-dependent, repeating “Step N” to rebuild and solve matrix equations) in “Phase II”,
- “Dynamic Loop B” (material is frequency-dependent, repeating “Step D” to update stiffness and structural damping matrices, repeating “Step N” to rebuild and solve matrix equations) in “Phase II”.

3.10.7 Section Conclusions

In the first place, the “Classic Frequency-Dependent Material” pattern and the “Viscoelastic Material” pattern in Nastran can model the frequency dependency of metamaterial samples. In addition, the “Classic Frequency-Dependent Material” pattern supports physical-space-based

and modal-space-based technical routes. In contrast, the "Viscoelastic Material" pattern only supports physical-space-based technical routes (see subsections **7.2.3** and **7.2.5**) [33]. Then, their dynamic performances need to be verified through corresponding simulations. More specifically, in the low-frequency range ($0 - 300$ Hz), the frequency dependency mainly relates to the melamine foam's structural damping (see subsection **2.2.3**). Thus, Nastran's frequency-dependent material patterns are used to model the melamine foam's structural damping (see subsections **7.2.2** and **7.2.4**).

In the second place, the efficient methods to treat the massive DOFs include the superelement techniques, the modal condensations, and the combination of superelement techniques and the modal condensations. The superelement technique is one of the signature functionalities of Nastran. Thus, in this paper, three superelement-correlated techniques will be adopted in simulation practices. They are the "List Superelement Method", the "Image Superelement Method", and the "Domain Solver" with "ACMS Method". According to theoretical assumptions, these three methods may have specific advantages in dynamic analyses, but their actual effects need corresponding simulations to verify.

However, using these three superelement techniques has constraints. More specifically, the "List Superelement Method" and the "Image Superelement Method" can be used in physical-space-based analyses, eigenvalue extraction analyses, and modal-space-based analyses. However, these two methods do not support the "Classic Frequency-Dependent Material" pattern; they only support the "Viscoelastic Material" or "Non-Frequency-Dependent Material" pattern. In addition, the "Viscoelastic Material" pattern can only be used in physical-space-based technical routes. Thus, the "List Superelement Method" and the "Image Superelement Method" can only be used with the "Viscoelastic Material" pattern in physical-space-based dynamic analyses.

In comparison, the "Domain Solver" with the "ACMS Method" can only be used with the "Classic Frequency-Dependent Material" pattern in modal-space-based dynamic analyses. The reason is that the "Viscoelastic Material" pattern can only be used in physical-space-based technical routes.

Finally, these efficient methods and material patterns will be used in simulation practices to test their performances. Simulation practices in subsections **14.1.6** and **14.1.7** show the practical accuracy of the numerical methods in eigenvalue extraction analyses. And the testing models are two perforated PVC plates. We know that the frequency dependency and damping are inactive in eigenvalue extraction analyses. Thus, the simulation practices in subsections **14.1.6** and **14.1.7** verify the practical accuracy of the three superelement-correlated techniques.

Then, simulation practices in subsection **15.1.3** show the efficiency and relative accuracy in eigenvalue extraction analyses of the three superelement-correlated techniques. And the testing models are size-controlled metamaterial panels. After that, simulation practices in subsections **15.1.4** and **15.1.5** are steady-state dynamic analyses of metamaterial samples; these practices are used to verify the efficiency and relative accuracy of the three superelement-correlated techniques and the two frequency-dependent material patterns. The frequency-dependent material patterns, the superelement-correlated techniques, and the simulation spaces (modal or physical space) are three mutual restraint factors in steady-state dynamic analyses. The mutual restraint situations are displayed in **Table 15.1**. And the proper damping models under mutual restraint are displayed in **Table 15.2**.

3.11 Vibro-Acoustic Analysis in Nastran

The vibro-acoustic analysis aims to solve problems with structural-acoustic coupling models. Nastran is not competent in this field, especially in modelling acoustic sources to assess soundproof materials. Even though Nastran can model the point source (see subsection **10.2.4**), the point source must be a grid point belonging to finite elements. Thus, it is inefficient to model a point source far from the testing domain. Because Nastran users must enlarge the acoustic model to cover the far-away point source, this operation increases calculation costs. And the point source cannot generate acoustic loads that can cause steady-state dynamic responses. The diffuse incident field can generate steady-state dynamic responses. However, Nastran cannot model the diffuse incident filed. By the way, the arguments in this paragraph are based on [34].

Chapter 4

Simulations in Abaqus

4.1 Driving Abaqus Through an Input File

Contents of an Abaqus input file can be subdivided into two subsections, “History Data” and “Model Data”. “History Data” contains analysis steps (or solution sequences) that control the simulation history. “Model Data” contains model settings, geometry, and mechanical information. **Figure 4.1** shows one of the reasonable arrangements of an Abaqus input file. Specifically, a model in Abaqus can be categorized into “Part”, “Part Instance”, and “Assembly”. The object of analysis is the “Assembly”, a collection of “Part Instances”, such as “Part Instance (1)” and “Part Instance (2)” till to “Part Instance (N)”, as shown in **Figure 4.1**. A part instance inherits characteristics of a part, and many part instances positioned at different places may be referring to one part. Therefore, the part is like a blueprint of the finite element model. By the way, the introduction in this section is based on [35].

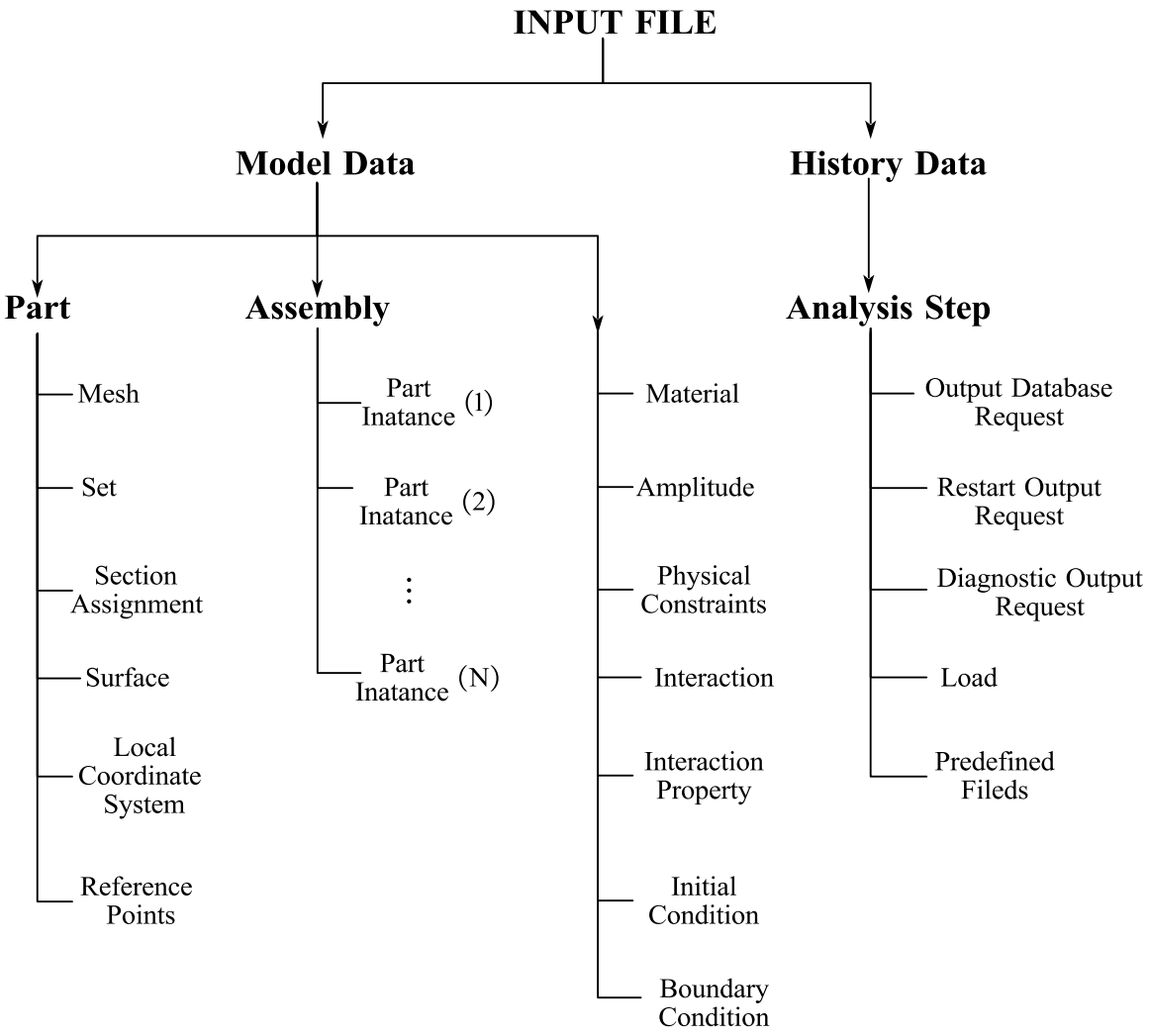


Figure 4.1: The logical layout and content of an Abaqus input file.

4.2 Abaqus Simulations

Advanced software architecture is crucially important for powerful and efficient simulation. The software architecture is the base of the executive system. Thus, based on different software architectures, software executive systems have different capabilities to drive the functional modules and manage the intermediate data. To some extent, the different software architectures of Nastran, Abaqus and Actran make them to be experts in different fields.

Firstly, Nastran has an advanced architecture, especially in treating modal-space-based dynamic analysis. Users do not need to make special settings. Nastran's executive system can automatically deal with all possible matrices, including symmetric or unsymmetric matrices. In addition, due to the advanced software architecture, Nastran can efficiently manage massive intermediate data. For example, when the model consists of frequency-dependent materials, the huge amount of intermediate data generated by the model-rebuilding loop can be easily managed by Nastran (see subsections **3.10.4**, **3.10.5**, and **3.10.6**).

Secondly, Abaqus developed two software architectures that can treat eigenvalue extraction analyses. One is the traditional architecture, and the other is the SIM architecture. The traditional architecture covers all simulation scenarios but may not always perform well, while the SIM architecture is dedicated to modal-space-based dynamic analysis. Thus, these two architectures have their superiorities and limitations. In modal-space-based dynamic analyses, the SIM architecture is more efficient than the traditional architecture. And in modal space, the SIM architecture can treat coupled or uncoupled systems, while the traditional architecture can treat only the uncoupled systems. Therefore, users must judge objective systems before choosing the SIM or traditional architecture. The arguments in this paragraph are based on [36].

Thirdly, the software architecture of Actran is weaker than that of Nastran and Abaqus. For example, Nastran and Abaqus can accomplish a modal-space-based dynamic analysis through one singular calculation cycle, but Actran needs two independent calculation cycles. In the first calculation cycle, Actran executes the eigenvalue extraction. And in the second calculation cycle, Actran executes the modal-space-based dynamic analysis by using the eigenmodes extracted from the first calculation cycle. This fact indicates that Actran cannot efficiently manage massive intermediate data. Thus, the software architecture of Actran is weaker than that of Nastran or Abaqus.

4.2.1 The Traditional Architecture

The traditional architecture is developed from the old version of the software. It cannot drive advanced tools in modal-space-based technical routes. After eigenvalue extraction analyses, with the traditional architecture, the succeeding modal-space-based steady-state dynamic analysis (see subsection **4.2.3**) can deal with only modal or global damping but material or element damping. For material or element damping, with the traditional architecture, users must turn to the physical-space-based technical routes (see subsection **4.2.5**). By the way, the arguments in this paragraph are based on [36].

4.2.2 The SIM Architecture

SIM is a high-performance architecture for dynamic analyses with modal superposition procedures, such as modal-space-based dynamic analyses. Users must declare to activate it in the eigenvalue extraction stage. Then, the following calculations will be SIM-architecture-based instead of traditional-architecture-based procedures. In addition, the SIM architecture can drive the efficient eigensolver AMS (automatic multi-level substructure). This eigensolver is analogous to the "Domain Solver" with the "ACMS Method" in Nastran (see subsection **3.10.2**). With minimal output requests, the SIM architecture is much more efficient than the traditional architecture for large-scale linear dynamic analysis. By the way, the arguments in this paragraph are based on [36].

4.2.3 Modal-Space-Based Technical Routes

Modal-space-based dynamic analyses can also be called modal superposition or modes-based dynamic analyses. These dynamic analyses need an eigenvalue extraction analysis as the prior step to supply projection bases (eigenmodes). Then, under traditional architecture, the modal-space-based dynamic analysis requires uncoupled modal systems. Thus, the solver can solve linear-independent equations. And this solver supports the setting of modal or global damping. While under the SIM architecture, the modal-space-based dynamic analysis can process uncoupled or coupled modal systems. And the solver driven by the SIM architecture supports the setting of modal, global, material, and element damping. Under the SIM architecture, the modal and global damping are directly added, while the material and element damping is projected onto the modal space after the eigenvalue extraction process. If the projected modal system is uncoupled, the solver only needs to solve linear-independent equations. While if the projected modal system is still coupled, it will turn to the solver for solving coupled matrix equations. The SIM architecture gives the capability of solving coupled systems in modal space. By the way, the arguments in this paragraph are based on [36].

4.2.4 Physical-Space-Based Technical Routes

The physical-space-based technical route indicates that the objective modal contains physical DOFs, and the model is generally coupled. In this technical route, the condensation of the physical DOFs can only be executed by substructure techniques (superelement techniques in Abaqus). The physical-space-based technical route is an analysis route in traditional architecture, and it can drive solvers to solve coupled matrix equations in physical space. The physical-space-based dynamic analysis is a typical direct dynamic analysis procedure. By the way, the arguments in this paragraph are based on [36].

4.2.5 Real Eigenvalue Extraction Analyses

The real eigenvalue extraction analysis requires real symmetric stiffness and mass matrices. Abaqus uses the Lanczos method to extract eigenvalues. The solving environment "Abaqus/Standard" (see subsection **2.7.2**) contains the solver for driving the real eigenvalue extraction analysis. The Lanczos method is also used in Nastran (see subsection **3.7.3**). While

in conditions with the SIM architecture, Abaqus can use the AMS method (see subsection **4.2.2**). By the way, the arguments in this paragraph are based on [36].

4.2.6 Solvers for Coupled Matrix Equations

Solvers for solving coupled matrix equations in Abaqus are analogous to those in Nastran. They are the direct sparse solver and the iterative solver [17]. The solving environment "Abaqus/Standard" contains these two solvers. The direct sparse solver is the default one, and users can change the default setup by making a statement through the corresponding Abaqus keyword to activate the iterative solver. Abaqus supports limited adjustability to the usage of the sparse and iterative solver. At the same time, Nastran supplies more adjustability of the solver usage. And all simulation practices in this paper adopt the direct solver. For more explanations, please review subsection **3.8.1**.

4.3 Simulation Designs through Abaqus

4.3.1 Methods for Processing Metamaterial Samples

The metamaterial samples have three characteristics: massive DOFs, periodic substructures, and viscoelasticity. Section **3.10** introduces these metamaterial characteristics and lists some efficient methods Nastran can adopt to solve the problems caused by these characteristics. Then, we know that the efficient methods for processing the metamaterial samples generally include superelement techniques, modal-space-based condensation, and specific frequency-dependent material patterns. Through Nastran, users can make trade-off choices based on these efficient methods to build the processing routes for analysing the metamaterial. However, in Abaqus, there are some obstacles in processing the metamaterial sample. Such as the limitations in applying substructure techniques (see subsection **4.3.2**). And the limitations of material patterns in defining frequency dependency (see subsection **4.3.4**).

4.3.2 Superelement (Substructure) Techniques in Abaqus

The superelement method in Abaqus is called the substructure method, and the superelement is called the substructure. A substructure process consists of three steps: generation of substructures, analysis with substructures, and data recovery of reduced models. In applications, users must manually define and generate each substructure in the first stage and then use the substructures as elements in a model in the second stage. Thus, users must repeat the first stage if many independent substructures are needed. Before using substructures, their data must be stored in substructure libraries. In Abaqus, the total substructures cannot exceed 9999 in a substructure library, and one calculation cycle can access a maximum of 500 substructure libraries. This method in Abaqus is analogous to the “External Superelement Method” in Nastran (see subsection **3.5.1**). Thus, if too many substructures need to be set, the substructure method will incur manipulation difficulties. By the way, the introductions in this paragraph are based on [37].

4.3.3 Modal-Space-Based Condensation in Abaqus

The modal-space-based condensation reduces massive physical DOFs to a smaller number of modal DOFs, and this operation needs a prior eigenvalue extraction procedure. The reduced model is a cost-saving term, but the eigenvalue extraction is an extra cost. Under the traditional architecture, the efficiency of eigenvalue extraction procedures is not very high. Fortunately, Abaqus supports the high-efficient SIM architecture, and the SIM-architecture-based AMS method can considerably improve the efficiency of the eigenvalue extraction procedure (see subsections **4.2.2** and **4.2.3**). Therefore, modal-space-based dynamic analyses under the SIM architecture are proper for treating the system with massive DOFs.

4.3.4 Frequency-Dependent Material Patterns in Abaqus

The frequency-domain linear viscoelasticity is a pattern for defining frequency-dependent materials. More specifically, in Abaqus, there are three ways to build the frequency-dependent

functions: the “power law frequency dependence”, the “tabular frequency dependence”, and the “Prony series parameters”. Abaqus uses frequency-dependent shear and bulk moduli to stand for the frequency-dependent elasticity of isotropic materials. There are no supports to set up the frequency dependency of anisotropic or orthotropic materials. And the arguments in this paragraph are based on [38].

As already introduced, a metamaterial unit cell consists of an aluminium core and a melamine base. And melamine is an orthotropic frequency-dependent material (see subsection **2.2.3**). Therefore, Abaqus’s frequency-domain linear viscoelasticity pattern is improper for defining the metamaterial sample. Furthermore, a physical-space-based technical route supports setting the frequency dependency to materials, but a modal-space-based technical route cannot [36]. A modal-space-based technical route can set mode-dependent parameters, such as modal damping, and various values of modal damping at different modes must be supplied by users [39].

4.3.5 Section Conclusions

The improper frequency-dependent material pattern is the first obstacle in processing metamaterial samples through Abaqus. Thus, the simulation practices in subsection **15.2.3** ignore the melamine foam’s frequency dependency.

Then, the substructure technique in Abaqus is improper for treating the metamaterial samples. The reason is that these metamaterial samples have too many substructures, and manual operations for treating them are inefficient.

The SIM-architecture-based AMS method can improve the efficiency of the modal analyses, and its specific effects need tests to verify. Thus, control-group tests will adopt the traditional-architecture-based “Ordinary Method” and the SIM-architecture-based “AMS Method”, respectively. Subsection **15.2.1** shows the control-group tests of eigenvalue extraction analyses. Then, subsections **15.2.2** and **15.2.3** show the control-group tests of the physical-space-based and modal-space-based steady-state dynamic analyses.

4.4 Vibro-Acoustic Analysis in Abaqus

Abaqus uses a source-standoff-point system to model different acoustic source types, including the planar source, the spherical source, and the diffuse incident field. In one calculation cycle, only one set of the source-standoff-point system should be defined. The source-standoff-point system brings many conveniences in defining acoustic sources. Neither source nor standoff point should connect any finite or infinite elements. Therefore, no matter how far the source point is away from the testing field, there is no need to enlarge the acoustic domain to reach the source location. For more information about the theory of the source-standoff-point system, please see section **11.5**. And the introductions in this paragraph are based on [40].

The diffuse incident field is an acoustic source that Abaqus uses to load the fluid-structure system in corresponding tests in section **17.2** since this acoustic source can generate steady-state dynamic responses. In Abaqus, the diffuse incident field can be used with the physical-space-based technical routes but not modal-space-based technical routes. Thus, in section **17.2**, Abaqus adopts the physical-space-based steady-state vibro-acoustic analyses to test metamaterial samples. And subsection **17.1.2** gives more specific explanations. At last, the arguments in this paragraph are based on [41].

Chapter 5

Simulations in Actran

5.1 Driving Actran Through an Input File

It is necessary to note that this section's introductions are based on [42]. Then, we can learn that in Actran's reference guide, subsections of an input file are called "Data Blocks". However, the name "Data Blocks" may confuse readers with the data block introduced before in Nastran. The "Data Blocks" in Actran is a part of the input file, while the data block in Nastran is a part of Nastran's database. They are two different things. Thus, this paper uses "Data Groups" instead of "Data Blocks" to call the subsections of the Actran input file.

Subsections of the Actran input file are specific data groups. They are categorised into eight types: "Analyses and Main Data Groups", "Topology Data Groups", "Component Data Groups", "Material Data Groups", "Connector Data Groups", "Boundary Condition Data Groups", "Output Data Groups", "Analysis Parameter Data Groups". **Figure 5.1** is a schematic diagram that lists the data groups as subsections of an input file and briefly shows some basic interrelations. Complete interrelations between these data groups are much more complex than in **Figure 5.1**. Members in these data groups have independent interconnecting routes. For example, members in the "Analysis Parameter Data Groups" may directly influence the whole simulation environment or corresponding members in other data groups.

The "Analysis and Main Data Groups" states analysis types, such as direct frequency response analysis, eigenvalues extraction analysis, modal frequency response analysis, etc. This data group is the master data group. Other data groups are slave data groups. The "Topology Data Groups" describes the geometry information of models, such as mesh, domains, partitions, etc. Some data groups inherit models of this data group by reference, such as the "Component Data Groups", the "Boundary Condition Data Groups", and the "Connector Data Groups". As their names imply, the "Material Data Groups" and the "Connector Data Groups" describe corresponding material properties and connecting objects, respectively. The "Output Data Groups" is a data group that allows users to customise their data-extracting requirements for postprocessing. The "Boundary Condition Data Groups" has diversified

loading forms, boundary-type definitions (a boundary with specific physical properties), boundary states (a boundary with some constant physical values), etc. The "Component Data Groups" consists of various Actran-cognized components. They are task-oriented subdivisions of the whole model and are proper for different simulation scenarios. During a simulation practice, these components are processing objects that are compound materials, geometry domains, physical properties, and status-adjustment options. **Figure 5.1** shows two arrow-ended solid lines from the "Component Data Groups" to the "Topology Data Groups" and "Material Data Groups", and this indicates that the former data-block group inherits the two latter data-block groups by reference.

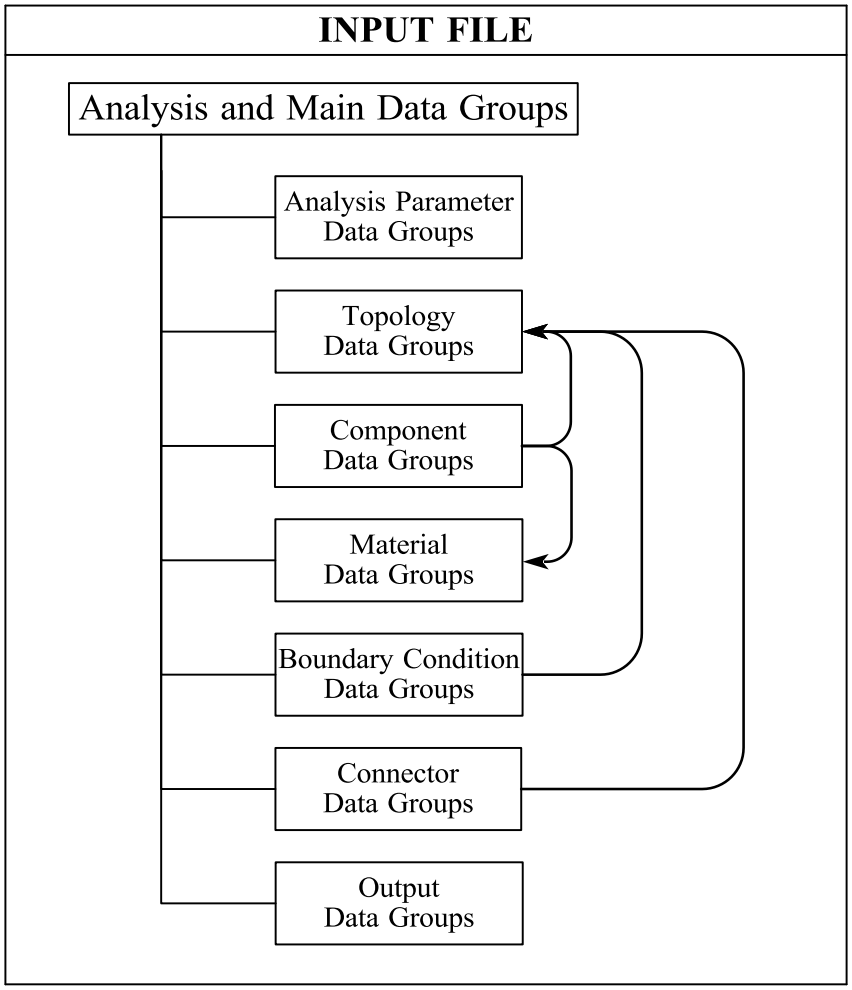


Figure 5.1: The logical structure and content of an Actran input file.

5.2 Actran Simulations

5.2.1 Modal-Space-Based Technical Route

A modal-space-based technical route needs a prior eigenvalue extraction step (see subsection **5.2.3**). As described before, Nastran and Abaqus can integrate the real eigenvalue extraction analysis step with the succeeding matrix-equations-solving step. However, the case is different in Actran. A modal-space-based dynamic analysis needs two calculation cycles in Actran. The first cycle is the eigenmode extraction step, and the extracted eigenmodes must be stored outside a database file. The second cycle calculates matrix equations with the modal-space-based condensed matrices. Following the logic, we can see that Actran's modal-space-based dynamic analyses cannot support the material's frequency dependency. The reason is that a frequency-dependent model needs an automatic model-rebuilding mechanism. A necessary condition that the automatic model-rebuilding mechanism can work is the internally integrated calculations. For more information about the model-rebuilding mechanism, please see **Figure 3.20** in subsection **3.10.4**. The theory in Nastran gives good explanations. By the way, the arguments in this paragraph are based on [43].

A modal-space-based dynamic analysis in Actran supports the setting of modal damping and global damping but material damping. Modal damping can be set independently at each mode, and it includes two types, structural and viscous modal damping. Global damping generates damping matrices proportional to stiffness or mass matrices. By the way, the descriptions in this paragraph are based on [44] and [45].

Under the modal-space-based technical route, Actran supplies four different solvers for treating the structural-acoustic-coupled model: "REFERENCE", "ITERATIVE", "WEAK", and "STRONG" (see subsection **5.2.4**). In this technical route, users cannot set frequency-dependent parameters. By the way, the arguments in this paragraph are based on [46].

5.2.2 Physical-Space-Based Technical Route

The physical-space-based dynamic analysis in Actran is analogous to those in Nastran (see subsection **3.7.2**) and Abaqus (see subsection **4.2.4**). A physical-space-based dynamic analysis generally treats the coupled system, and all available solvers and solver sets are listed in subsection **5.2.5**. Users can set frequency-dependent materials in physical-space-based dynamic analyses. By the way, the arguments in this paragraph are based on [43], [46], and [47].

5.2.3 Real Eigenvalue Extraction Analyses

Actran supplies several solvers to execute real eigenvalue extraction. These solvers include "SPARSE", "MUMPS", and "PARDISO" (see subsection **5.2.5**). In Actran, if a model contains structural and acoustic components, the real eigenvalue extraction of these two components is independent. In other words, one operation cycle of eigenmode extraction can only be applied to a structural or acoustic component. By the way, the arguments in this paragraph are based on [43].

5.2.4 Solvers for Modal-Space-Based Dynamic Analysis

For modal frequency response analysis, there are four modal solvers:

- STRONG modal solver,
- WEAK modal solver,
- REFERENCE modal solver,
- ITERATIVE modal solver.

For more information about choosing proper solvers, please see the corresponding section in the reference guide [46].

5.2.5 Solvers for Coupled Matrix Equations

The solving system in Actran is not as powerful as the ones in Nastran or Abaqus. This solving system contains many solvers and solver sets but has no organic integration. Users must choose different solvers or solver sets to match different solving practices.

There are six solvers:

- SPARSE (direct solver using a sparse storage scheme),
- MUMPS (direct multifrontal massively parallel solver),
- PARDISO (parallel direct sparse solver),
- CG_ILU (BiCGStab iterative solver with incomplete factorisation),
- KRYLOV (fast FRF synthesis solver),
- STAGGERED_SOLVER (fast staggered approach).

There are three KRYLOV-based solver couples:

- KRYLOV with SPARSE,
- KRYLOV with MUMPS,
- KRYLOV with PARDISO.

There are four STAGGERED_SOLVER-based solver sets:

- STAGGERED_SOLVER with SPARSE,
- STAGGERED_SOLVER with MUMPS,
- STAGGERED_SOLVER with PARDISO,
- STAGGERED_SOLVER with CG_ILU.

Please see the corresponding section in the reference guide [46] for more information about choosing proper solvers or solver couples.

5.3 Simulation Designs through Actran

5.3.1 Methods of Actran in Processing the Metamaterial Sample

A brief introduction about the metamaterial sample and efficient methods for processing this material is stretched in section **3.10**. As already been described, the superelement techniques, modal-space-based condensation, and frequency-dependent material are the main aspects that need to be considered for an efficient and successful analysis of the metamaterial.

Actran supports the setting of frequency-dependent isotropic and orthotropic material in physical-space-based dynamic analyses [47]. Then, even Actran can process the reduced model generated by Nastran through superelement techniques, but Actran itself cannot generate a reduced model through dynamic reduction (see subsection **5.3.2**). Thus, Actran's superelement technique is improper for processing the metamaterial samples.

5.3.2 Superelements in Actran

Actran does not have full-fledged superelement techniques. It can only execute a static condensation based on the impedance matrix. The static condensation may be applied to not altered acoustic parts for sound propagation problems. However, Actran cannot execute a dynamic reduction based on CMS (component modes synthesis) method. According to the CMS theory, the model reduced by static condensation is improper for dynamic analyses. Component modes as compensations are needed for the lost inertia effects due to static condensation (see subsection **3.6.1**). And the arguments in this paragraph are based on [48].

Although Actran can import superelements generated by dynamic reduction from Nastran, this operation is sometimes dispensable, especially for model reduction. On the one hand, calculation quality is hard to control when using reduced models between two different software platforms. Moreover, the error between the two platforms is not traceable since they have no inner communications. In addition, the imported superelements from Nastran to Actran are not frequency-dependent, so this technique in Actran is improper for analysing the metamaterial sample. On the other hand, modal-space-based condensation is more efficient. The number of eigenmodes is always much less than the number of DOFs of the reduced model from dynamic reductions. And the arguments in this paragraph are based on [49].

5.3.3 Modal-Space-Based Condensation in Actran

A modal-space-based condensation can reduce the size of a model. This operation is a prior step in the modal-space-based dynamic analysis in Actran. The modal-space-based dynamic analysis needs two-stage calculation cycles, and this technique route cannot process frequency-dependent materials. Still, it can accept mode-dependent modal damping factors (see subsection **5.2.1**).

The two-stage calculation increases the manipulation cost. However, because of the two-stage calculation, the eigenmode extraction stage is independent of the modal-space-based calculation stage. Thus, the eigenmodes can be extracted by other software, such as Nastran.

In addition, the mode-dependent modal damping factors can be calculated outside with other programs, such as Nastran's DMAP (see section **7.4**).

5.3.4 Frequency-Dependent Material Patterns in Actran

Material numerical models in Actran are analogous to the ones in Nastran; they are material patterns, each of which is an enveloped property group. Each material pattern independently contains predefined specific material properties. Such as "Isotropic Solid", "Orthotropic Solid", and "Fluid" are three of these material patterns in Actran. Some material properties in these material patterns are complex parameters and can be defined as frequency-dependent or non-frequency-dependent parameters, such as elastic moduli, structural damping factors, etc. At last, the introductions in this paragraph are based on [47].

In Actran, the physical-space-based technical route supports defining the material's frequency dependency (see subsection **5.2.2**), but the modal-space-based technical route does not (see subsection **5.2.1**).

5.3.5 Section Conclusions

In the first place, Actran does not have full-fledged superelement techniques, and reduced models by dynamic reduction of Nastran are dispensable for increasing the efficiency (see subsection **5.3.2**). Thus, simulations in sections **14.3**, **15.3**, and **17.2** do not adopt any of Actran's superelement-correlated methods.

In the second place, the modal-space-based technical routes in Actran are not as efficient as Nastran or Abaqus. Thus, except for the eigenvalue extraction analyses in section **14.3** and subsection **15.3.1**, the other simulations will not adopt Actran's modal techniques. In section **17.2** and section **17.3**, the eigenmodes are extracted through Nastran. Then, the extracted eigenmodes will be sent into Actran to execute modal-space-based steady-state vibro-acoustic analyses.

In the third place, the material's frequency dependency can be exactly modelled by physical-space-based technical routes in Actran, and the correlated simulations are introduced in sections **15.3.2** and **15.3.3**. Besides the physical-space-based technical routes, there is a compounding technical route through Nastran cooperating with Actran to model the material's frequency dependency in modal space approximately. Nastran's DMAP extracts eigenmodes and mode-dependent modal damping as the first step. Then, these extracted eigenmodes and mode-dependent modal damping can be sent into Actran to accomplish the modal-space-based steady-state vibro-acoustic analyses (see section **17.2** and section **17.3**).

5.4 Vibro-Acoustic Analysis in Actran

Actran supplies many acoustic source models, such as “SPHERICAL” (point source), “PLANE” (plane wave source), “CYLINDRICAL” (line source), and “DIPOLE” (two-points source). For more information about the fundamental theories of different acoustic source types, please see section **10.2**. However, these simple sources are not proper for steady-state vibro-acoustic analysis. Except for these simple sources, the “DIFFUSE_FIELD” (diffuse incident filed) source is proper for steady-state vibro-acoustic analyses. The reason is that this source can generate steady-state frequency response functions, and this is a good way to see the dynamic performances of testing samples. These sources are located automatically within a finite element or an infinite element domain by Actran, and users only need to assign their locations by the coordinate values. When the source is close to the testing domain, it is proper to locate it in the finite acoustic domain; if the source is far away from the testing domain, it is proper to locate it in the infinite acoustic domain. By the way, the arguments in this paragraph are based on [41], [50], and [51].

Chapter 6

Fundamentals of Damping

For a system, two kinds of damping influence dynamic performance. One is velocity-dependent, called viscous damping. And the other is displacement-dependent, called structural damping. In dynamic processes, damping is always one of the crucial factors.

6.1 Basic Parameters

Some basic parameters are used to describe a single-DOF system's dynamic performance. Let K be a general stiffness, M be a general mass, C be a general viscous damping value, and η be a general structural damping factor. Then, the following parameters and expressions are based on [52].

The natural frequency can be expressed as:

$$\omega_n = \sqrt{\frac{K}{M}} \tag{5.1}$$

The critical damping can be expressed as:

$$C_{cr} = 2\sqrt{KM} \tag{5.2}$$

The critical damping ratio can be expressed as:

$$\zeta = \frac{C}{C_{cr}} \tag{5.3}$$

Let $\widetilde{P}$ be the complex amplitude of the load. A general harmonic force can be expressed as:

$$p(t) = \widetilde{P}e^{i\omega t} \tag{5.4}$$

Let $\widetilde{U}$ be the complex amplitude of the displacement. A general harmonic response can be expressed as:

$$u(t)=\widetilde{U}e^{i\omega t} \tag{5.5}$$

6.2 Viscous Damping

It is necessary to note that the following parameters and expressions are based on [53]. Then, we can learn that the equation of motion of a single-DOF system with viscous damping in a steady-state dynamic analysis can be expressed as:

$$(-\omega^2 M+i\omega C+K)\widetilde{U}=\widetilde{P} \tag{5.6}$$

Introducing the frequency-dependent dynamic compliance $H(\omega)$ for indicating dynamic performance,

$$K\widetilde{U}=K\frac{1}{-\omega^2 M+i\omega C+K}\widetilde{P}(\bar{x})=H(\omega)\widetilde{P} \tag{5.7}$$

The real part of dynamic compliance can be expressed as:

$$\mathrm{Re}(H(\omega))=\frac{K(K-\omega^2 M)}{(K-\omega^2 M)^2+(\omega C)^2}=\frac{1-\left(\frac{\omega}{\omega_n}\right)^2}{\left(1-\left(\frac{\omega}{\omega_n}\right)^2\right)^2+\left(2\zeta\frac{\omega}{\omega_n}\right)^2} \tag{5.8}$$

The imaginary part of dynamic compliance can be expressed as:

$$\mathrm{Im}(H(\omega))=\frac{K(-\omega C)}{(K-\omega^2 M)^2+(\omega C)^2}=\frac{-2\zeta\frac{\omega}{\omega_n}}{\left(2-\left(\frac{\omega}{\omega_n}\right)^2\right)^2+\left(2\zeta\frac{\omega}{\omega_n}\right)^2} \tag{5.9}$$

The amplitude of the dynamic compliance can be expressed as:

$$|H(\omega)|=\frac{K}{\sqrt{(K-\omega^2 M)^2+(\omega C)^2}}=\frac{1}{\sqrt{\left(1-\left(\frac{\omega}{\omega_n}\right)^2\right)^2+\left(2\zeta\frac{\omega}{\omega_n}\right)^2}} \tag{5.10}$$

The phase lag of the system can be expressed as:

$$\theta_{phase} = \arctan\left(\frac{\mathrm{Im}(H(\omega))}{\mathrm{Re}(H(\omega))}\right) = \arctan\left(\frac{-\omega C}{K-\omega^2 M}\right) = \arctan\left(\frac{-2\zeta\frac{\omega}{\omega_n}}{1-\left(\frac{\omega}{\omega_n}\right)^2}\right) \tag{5.11}$$

According to the amplitude expression of the dynamic compliance, when the critical damping ratio satisfies the inequation $\zeta < 1/2$, the angular frequency at which the peak amplitude emerges can be expressed as:

$$\omega_p = \omega_n\sqrt{1-2\zeta^2} \tag{5.12}$$

Thus, we can see that the resonance frequency ω_p is lower than the natural frequency ω_n under the effect of viscous damping.

If the system is slightly damped and the values of ζ^2 is negligible compared with unity, the maximum dynamic compliance amplitude can be written as:

$$|H(\omega)|_{max} = \frac{1}{2\zeta} \tag{5.13}$$

And this value can be called the quality factor,

$$Q = \frac{1}{2\zeta} \tag{5.14}$$

6.3 Structural Damping

It is necessary to note that the following parameters and expressions are based on [53]. Then, we can learn that the equation of motion of a single-DOF system with structural damping in a steady-state dynamic analysis can be expressed as:

$$(-\omega^2 M + (1+i\eta)K)\tilde{U} = \tilde{P} \tag{5.15}$$

Introducing the frequency-dependent dynamic compliance $H(\omega)$ to express the dynamic performance,

$$K\tilde{U} = K\frac{1}{-\omega^2 M + (1+i\eta)K}\tilde{P}(\bar{x}) = H(\omega)\tilde{P} \tag{5.16}$$

The real part of dynamic compliance can be expressed as:

$$\mathrm{Re}(H(\omega)) = \frac{K(K-\omega^2 M)}{(K-\omega^2 M)^2 + (\eta K)^2} = \frac{1-\left(\frac{\omega}{\omega_n}\right)^2}{\left(1-\left(\frac{\omega}{\omega_n}\right)^2\right)^2 + \eta^2} \tag{5.17}$$

The imaginary part of dynamic compliance can be expressed as:

$$\mathrm{Im}(H(\omega)) = \frac{K(-\eta K)}{(K-\omega^2 M)^2 + (\eta K)^2} = \frac{-\eta}{\left(1-\left(\frac{\omega}{\omega_n}\right)^2\right)^2 + \eta^2} \tag{5.18}$$

The amplitude of the dynamic compliance can be expressed as:

$$|H(\omega)| = \frac{K}{\sqrt{(K-\omega^2 M)^2 + (\eta K)^2}} = \frac{1}{\sqrt{\left(1-\left(\frac{\omega}{\omega_n}\right)^2\right)^2 + \eta^2}} \tag{5.19}$$

The phase lag of the system can be expressed as:

$$\theta_{phase} = \arctan\left(\frac{\mathrm{Im}(H(\omega))}{\mathrm{Re}(H(\omega))}\right) = \arctan\left(\frac{-K\eta}{K-\omega^2 M}\right) = \arctan\left(\frac{-\eta}{1-\left(\frac{\omega}{\omega_n}\right)^2}\right) \tag{5.20}$$

According to the amplitude expression of the dynamic compliance, the angular frequency at which the peak amplitude emerges can be written as:

$$\omega_p = \omega_n \tag{5.21}$$

Thus, we can see that the resonance frequency ω_p is equivalent to the natural frequency ω_n and is not affected by the structural damping effects.

The quality factor can be written as follows:

$$Q = \frac{1}{\eta} \tag{5.22}$$

When the square of the structural damping factor satisfies the inequation $\eta^2 \ll 1$, the equivalent damping can be defined as:

$$C_{eq} = \frac{\eta K}{\omega_n} \tag{5.23}$$

And the equivalent critical damping ratio can be defined as:

$$\zeta_{eq} = \frac{\eta}{2} \tag{5.24}$$

6.4 Viscous and Structural Damping of a Discretized System

In a steady-state dynamic analysis, the equation of motion of a discretised system with viscous and structural damping terms under a harmonic excitation can be expressed as:

$$\left[-\omega^2[M]_G + i\omega[C]_{vd} + i\eta[K]_{sd} + [K]_G\right]\left\{\tilde{U}\right\} = \left\{\tilde{P}\right\} \tag{5.25}$$

In the upper equation, $[M]_G$ is the whole-model mass matrix; $[C]_{vd}$ is the viscous damping matrix; $[K]_G$ is the whole-model stiffness matrix; $[K]_{sd}$ is the stiffness matrix related to structural damping; $\{\tilde{U}\}$ is the vector of complex displacement amplitude; $\{\tilde{P}\}$ is the vector of complex load amplitude.

Chapter 7

Damping in Nastran

7.1 Brief Introduction of Damping in Nastran

On the one hand, it is common for a model that the whole structure consists of different substructures with different materials; thus, this model has a natural subdivision. On the other hand, a model may be pre-processed by technical subdivisions to fulfil the requirements of specific numerical operations. In Nastran, a model's natural subdivision corresponds to the composite material types. At the same time, a model's technical subdivision may indicate the superelement subdivision or modal-space-based condensation. The first technical subdivision subdivides the model into superelements and residual structures. The second technical subdivision subdivides the physical model into many independent modes.

Thus, as one of the mechanical properties, damping needs properly be applied to specific objects according to the subdivisions:

- damping of materials (this damping can be used in physical space or modal space, and it is a material's property),
- damping of superelements or residual structures (this damping can be used in physical space, and its values can be specifically set to superelements and residual structures),
- damping of whole models (this damping can be used in physical space or modal space, and its values can be specifically set to different models),
- damping of given modes (this damping can be used in modal space, and its value can be specifically set to different modes).

As described in sections **6.2** and **6.3**, damping, in theory, can be subdivided into viscous and structural damping. Furthermore, in FEM software, the setting of damping must fulfil the application scenarios; thus, Nastran supplies several damping forms to users [25]:

- Structural damping (can be applied to parts with specific materials, superelements, residual structures, and whole models in physical or modal space),
- Rayleigh viscous damping (can be applied to superelements and residual structures in physical or modal space),
- hybrid damping (can be applied to superelements and residual structures in physical space),
- modal structural damping (can be applied to the given modes in modal space),
- modal critical damping ratio (can be applied to the given modes in modal space),
- modal quality factor (can be applied to the given modes in modal pace).

Simulation objects in this paper are metamaterial samples consisting of aluminium alloy and melamine foam, and the structural damping factors of these two constituent materials are known parameters. Thus, the following contents only elaborate on the numerical models of the material's structural damping.

7.2 Appling Structural Damping to The Assigned Materials in Physical Space

Nastran supplies four numerical models for solid materials and one numerical model for fluid materials. And these material's numerical models can be set with structural damping factors. For setting material properties, corresponding material-defining entries are needed. They are MAT1, MAT2, MAT8, MAT9, and MAT10. These five material-defining entries represent fundamental material patterns: isotropic solid material, shell element anisotropic material, shell element orthotropic material, solid element anisotropic material, and fluid material, respectively. Based on the fundamental material patterns, two extended frequency-dependent material patterns can be used to set the numerical models of frequency-dependent materials. One is the "Classic Frequency-Dependent Material" pattern (see subsection **3.10.5**), and another is the "Viscoelastic Material" pattern (see subsection **3.10.6**). Here is an example of a simplified input file template (see **Table 7.1**). Supposing the simulation is a physical-space-based steady-state dynamic analysis SOL 108, and the objective model consists of isotropic solid material MAT1. MAT1 represents a non-frequency-dependent fundamental material pattern. The descriptions in this paragraph are based on [54].

Table 7.1: Nastran's structural damping ("Non-Frequency-Dependent Material" pattern).

SECTIONS	CONTENT
Executive Control Section	SOL 108 CEND
Bulk Data Section	BEGIN BULK MAT1, mat_id, , G_{mat}, , , , , , η_{mat} ENDDATA

Supposing the "Classic Frequency-Dependent Material" pattern is activated, besides the fundamental material-defining entries, the additional frequency-table-indicating entries MAT1F, MAT2F, MAT8F, MAT9F, and MAT10F are needed. These additional frequency-table-indicating entries are links from the corresponding material-defining entries to specific table entries, such as the table entry TABLED1. For a more specific illustration, here is an example of a simplified input file template. **Table 7.2** only displays the required settings for activating classic-frequency-dependent material patterns. Thus, other settings and non-correlated subsections are not displayed. The template uses MAT1 and MAT1F, and the simulation is a physical-space-based steady-state dynamic analysis SOL 108. The objective parameters are the shear modulus G_{mat} and structural damping factor η_{mat}, and the two distinct TABLED1 entries list their frequency-dependent values, respectively. In the setting,

the mat_id in MAT1 and MAT1F must be congruent. Then, the table_id1 and table_id2 in the MAT1F and the two distinct TABLED1 entries must be congruent. In this way, MAT1 references MAT1F, then, MAT1F references TALBELD1. By the way, the descriptions in this paragraph are based on [54] and [55].

Table 7.2: Nastran's structural damping ("Classic-Frequency-Dependent Material" pattern).

SECTIONS	CONTENT
Executive Control Section	SOL 108
	CEND
Bulk Data Section	BEGIN BULK
	MAT1, mat_id, , G_{mat}, , , , , η_{mat}
	MAT1F, mat_id, , table_id1, , , , , table_id2
	TABELD1, table_id1, , , f_1, $G_{mat}(f_1)$, ... f_n, $G_{mat}(f_n)$, , ENDT
	TABELD1, table_id2, , , f_1, $\eta_{mat}(f_1)$, ... f_n, $\eta_{mat}(f_n)$, , ENDT
	ENDDATA

Supposing the "Viscoelastic Material" pattern is activated, besides the material-defining entries, the case-control command SDAMPING in the "Case-Control Section" of the input file is needed. This case-control command must reference an assigned table entry, such as TABLED1. For a more specific illustration, here is an example of a simplified input file template. **Table 7.3** only displays the required settings. Thus, other settings and non-correlated subsections are not displayed here. The template uses MAT1 as an instance, and the simulation is a physical-space-based steady-state dynamic analysis SOL 108. The reference shear modulus G_{REF} and structural damping factor η_{REF} are the objective parameters, and the frequency-dependent scaling factors $TR(f)$ and $TI(f)$ (see subsection **7.2.4**) need two distinct TABLED1 entries, respectively. In setting, the table_id in the SDAMPING entry and the TABLED1 entry of $TR(f)$ must be congruent. Then, the TABLED1 entry of $TI(f)$ must be set with table_id + 1. For example, assuming table_id=10, then table_id + 1=11. By the way, the descriptions in this paragraph are based on [54], [55], and [56].

Table 7.3: Nastran's structural damping ("Viscoelastic Material" pattern).

SECTIONS	CONTENT
Executive Control Section	SOL 108 CEND
Case-Control Section	SDAMPING = table_id
Bulk Data Section	BEGIN BULK MAT1, mat_id, , G_{REF}, , , , , η_{REF} TABELD1, table_id, , , $f_1, TR(f_1), ... f_n$, $TR(f_n)$, , ENDT TABELD1, table_id + 1, , , f_1, $TI(f_1), ... f_n, TI(f_n)$, , ENDT ENDDATA

7.2.1 Material Structural Damping of Non-Frequency-Dependent Material Pattern

In default, the structural damping applied to assigned finite elements is a constant parameter if defining it through a "Non-Frequency-Dependent Material" pattern. Consequently, the structural damping as a constant parameter of the damped part is a summation of the structural damping of each constitutive material:

$$i\eta [K]_{sd} = i \sum_{mat} \eta_{mat} [K]_{mat} \tag{6.1}$$

In the upper equation, η is the structural damping factor of the damped part; $[K]_{sd}$ is a stiffness matrix related to structural damping; η_{mat} is the structural damping factor of material mat; $[K]_{mat}$ is the stiffness matrix of finite elements consisting of the material mat.

7.2.2 Material Structural Damping of Classic Frequency-Dependent Material Pattern

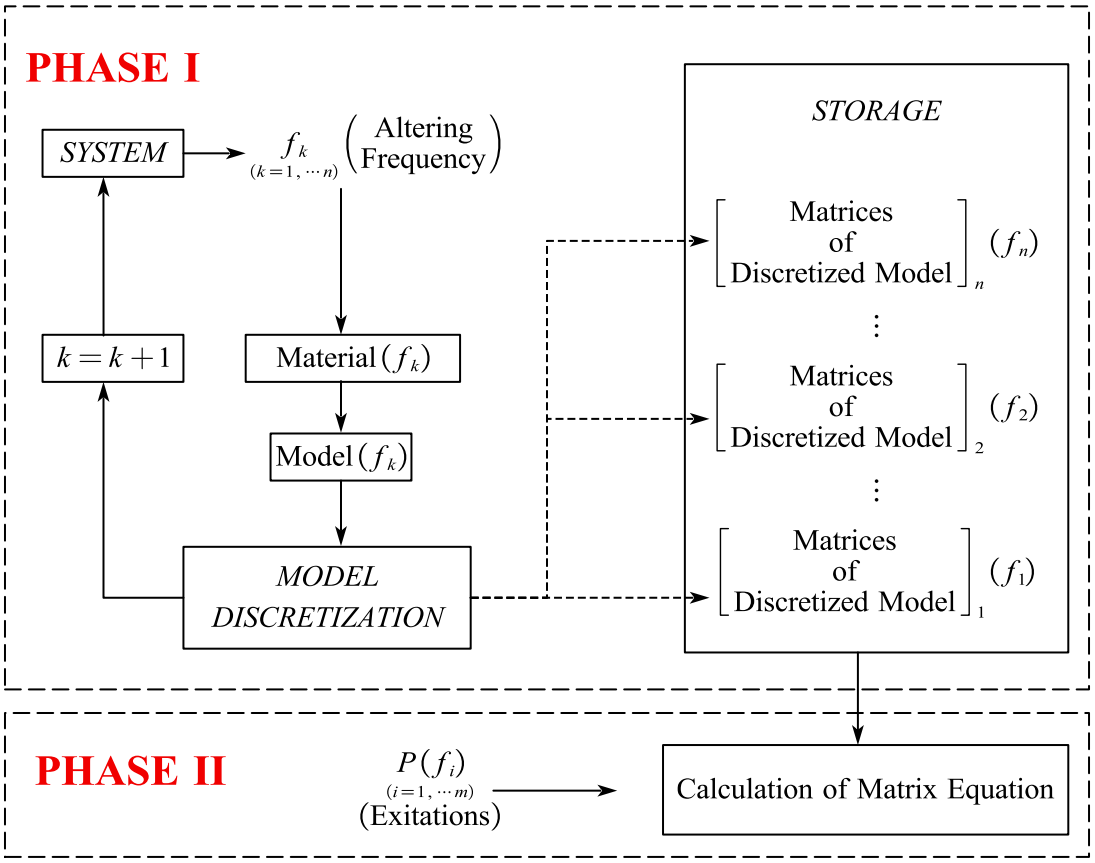


Figure 7.1: Nastran's model rebuilding mechanisms ("Classic Frequency-Dependent Material").

The structural damping as a frequency-dependent parameter of the damped part set by the "Classic Frequency-Dependent Material" pattern with the property-altering frequency f (cycles per unit time) can be expressed as:

$$i\eta(f)[K]_{sd} = i\sum_{mat}\eta_{mat}(f)[K]_{mat} \tag{6.2}$$

In the upper expression, $\eta(f)$ is the frequency-dependent structural damping factor of the damped part; the structural damping factor $\eta_{mat}(f)$ of each material is a frequency-dependent parameter, and its value can be set at each predefined property-altering frequency f_k.

Let us see the illustration in **Figure 7.1**. Assuming the number of predefined property-altering frequencies is n, the model's intrinsic properties will change n times by the material definition through the "Classic Frequency-Dependent Material" pattern. Thus, each time the frequency changes, the new property values activate the model-rebuilding mechanism in the stage "Phase I" (see section **3.4** and subsection **3.10.5**). In a model-rebuilding process, new

matrices of the discretized model will be generated. Then, property-altering frequencies drive a loop of model-rebuilding operations and generate many sets of model matrices.

Consequently, since the number of property-altering frequencies is n, there will be n sets of model matrices that become intermediate data stored in the database and waiting for dispatch by the stage "Phase II" (see section **3.7** and subsection **3.10.5**). In stage "Phase II", the excitation frequencies do not need to coincide with the property-altering frequencies. The system automatically executes a material-property interpolation to match the gap between excitation and property-altering frequencies [55]. In **Figure 7.1**, $P(f_i)$ is a general excitation load at the excitation frequency f_i.

7.2.3 Pros and Cons of The Classic Frequency-Dependent Material Pattern

In Nastran, the direct solution sequence SOL 108 and modal solution sequence SOL 111 support the "Classic Frequency-Dependent Material" pattern. When each excitation frequency is incoming, the frequency dependency activates the model rebuilding mechanism to update material properties (see subsection **3.10.5**). Then, the direct solution sequence SOL 108 (see subsection **3.4.1**) needs two steps to accomplish the solving process. The first step is rebuilding and discretising the model, and the second is solving the updated model in physical space. In comparison, the modal solution sequence SOL 111 (see subsection **3.4.2**) needs three steps to accomplish the solving process. The first step is rebuilding and discretising the model; then, an eigenvalue extraction procedure of the updated model is the second step; finally, solving the projected model in modal space is the third step.

A heavy storage burden is introduced in "Phase I" using the "Classic Frequency-Dependent Material" pattern. The reason is that the model-rebuilding operations by each property-altering frequency will generate many model matrices. Thus, this case impacts the solution efficiency due to the heavy burden of intermediate data storage and transfer.

There is another case that deserves attention. A model with superelements cannot use the "Classic Frequency-Dependent Material" pattern. The reason is that Nastran does not allow multi-times dynamic reductions (a dynamic reduction is a static condensation compounding with component mode synthesis) due to multi-times model rebuilding operations in one solution cycle. Otherwise, the multi-times dynamic reductions will decrease the solution efficiency, making no sense to adopt superelement techniques.

7.2.4 Material Structural Damping of Viscoelastic Material Pattern

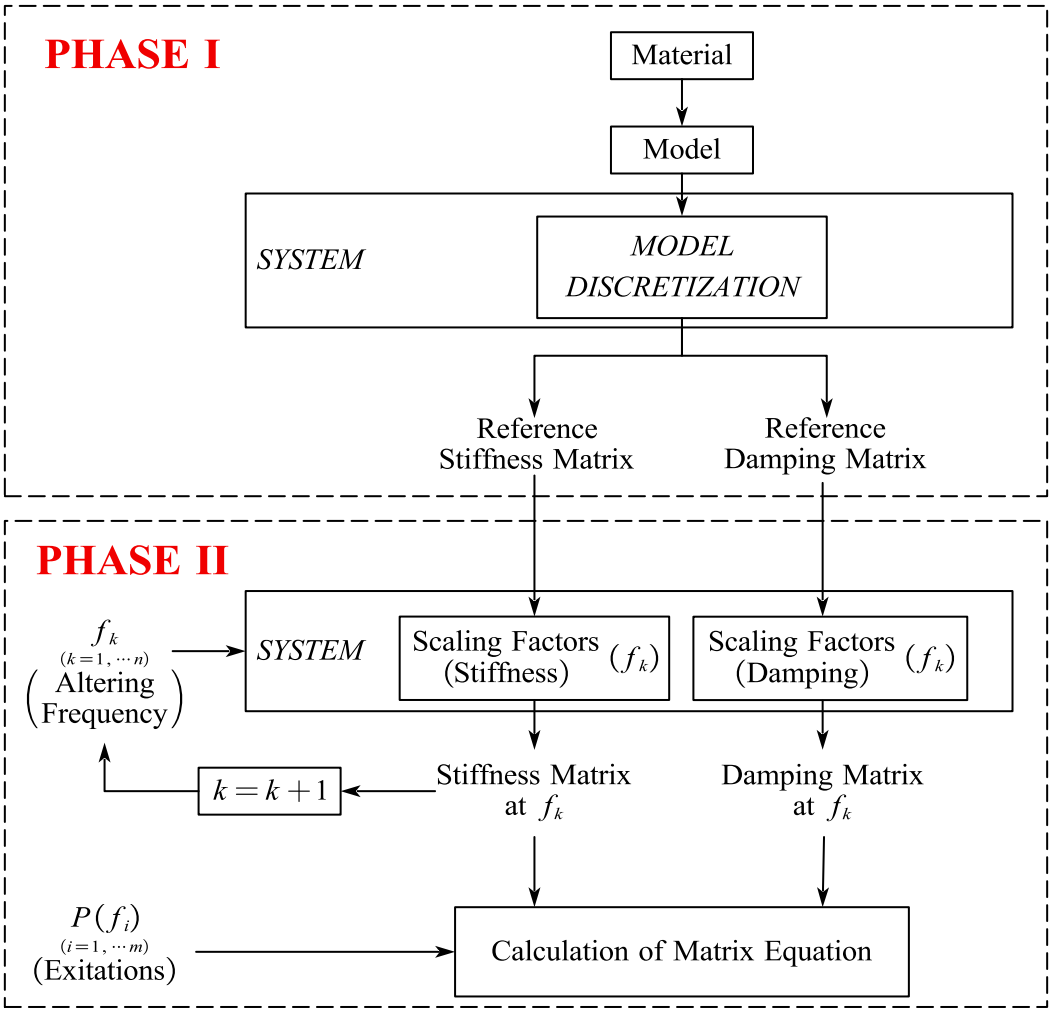


Figure 7.2: Nastran's model rebuilding mechanisms ("Viscoelastic Material").

It is necessary to note that all the arguments and deductions in this subsection are based on [33]. Then we can learn that a "Viscoelastic Material" has a complex frequency-dependent elasticity, of which the imaginary elasticity is the real elasticity times a structural damping factor. Thus, the imaginary elasticity essentially is the structural damping. In **Figure 7.2**, the system generates a stiffness matrix and a damping matrix through a model discretisation at a given state in the solution stage "Phase I" (see section **3.4** and subsection **3.10.6**). Then, the stiffness and damping matrices are chosen as references for updating at each property-altering frequency f_k in the solution stage "Phase II" (see section **3.7** and subsection **3.10.6**). Thus, the system uses predefined frequency-dependent factors to scale the reference stiffness and damping matrices at each property-altering frequency. The predefined frequency-dependent scaling factors of stiffness and damping matrices with the corresponding property-altering frequencies are listed in two distinct tables. These two scaling factors $TR(f)$ and $TI(f)$ are frequency-

dependent stiffness and damping factors, respectively. In practice, the property-altering frequencies do not need to coincide with the excitation frequencies. In the solution stage, "Phase II", the system will automatically execute interpolations based on the predefined frequency-dependent scaling factors tables before calculating the matrix equation for proper scaling factors to match the excitation frequencies. In **Figure 7.2**, $P(f_i)$ is a general excitation at the excitation frequency f_i.

Nastran describes a general 3D stage of stress in terms of a complex frequency-dependent shear modulus and real constant Poisson's ratio. A single complex frequency-dependent shear modulus can be expressed as:

$$G(f) = G'(f) + iG''(f) \tag{6.3}$$

In the upper equation, $G'(f)$ is the shear storage modulus, and $G''(f)$ is the shear loss modulus. The frequency-dependent structural damping factor (shear loss tangent) can be expressed as:

$$\eta(f) = \frac{G'(f)}{G''(f)} \tag{6.4}$$

Supposing a model constructed by "Viscoelastic Material" (emphasized by the subscript V) and with given analysing DOFs (emphasized by the subscript dd, please see subsection **3.5.3**). Without considering the global damping factor η_G, the dynamic stiffness matrix of the whole structure $[K_{dd}]_V$ consists of a general elastic stiffness matrix $[K_{dd}^1]_V$ and a general anelastic stiffness matrix $[K_{dd}^4]_V$:

$$[K_{dd}]_V = [K_{dd}^1]_V + i[K_{dd}^4]_V \tag{6.5}$$

In addition, the general anelastic stiffness matrix $[K_{dd}^4]_V$ can be expressed by the general elastic stiffness matrix $[K_{dd}^1]_V$ and a reference structural damping factor η_{REF}:

$$[K_{dd}^4]_V = \eta_{REF}[K_{dd}^1]_V \tag{6.6}$$

As the setting operation in Nastran, by introducing the frequency-dependent scaling factors to the elastic part and the anelastic part of the stiffness matrix $[K_{dd}^4]_V$ through the scaling factors $TR(f)$ and $TI(f)$, respectively, the dynamic stiffness matrix of the whole structure in a frequency-dependent form can be expressed as:

$$\begin{aligned}[K_{dd}(f)]_V &= [K_{dd}^1]_V + (TR(f) + iTI(f))[K_{dd}^4]_V \\ &= (1 + \eta_{REF}TR(f) + i\eta_{REF}TI(f))[K_{dd}^1]_V\end{aligned} \tag{6.7}$$

Furthermore, by introducing a reference shear modulus G_{REF}, a frequency-dependent shear storage modulus $G'(f)$, and a frequency-dependent shear loss modulus $G''(f)$, the frequency-dependent dynamic stiffness matrix of the whole structure can be expressed as:

$$[K_{dd}(f)]_V = \frac{G'(f)+iG''(f)}{G_{REF}}[K_{dd}^1]_V \tag{6.8}$$

Therefore, the frequency-dependent scaling factors of the elastic part can be expressed as:

$$TR(f) = \frac{1}{\eta_{REF}}\left(\frac{G'(f)}{G_{REF}} - 1\right) \tag{6.9}$$

And the frequency-dependent scaling factors of the anelastic part can be expressed as follows:

$$TI(f) = \frac{1}{\eta_{REF}}\frac{G''(f)}{G_{REF}} \tag{6.10}$$

7.2.5 Pros and Cons of The Viscoelastic Material Pattern

In Nastran, the “Viscoelastic Material” pattern does not change the material settings. Instead, in stage “Phase II”, this pattern applies the predefined specific scaling factors $TR(f)$ and $TI(f)$ to the corresponding elastic and anelastic matrices to update the model properties at each property-altering frequency. Thus, this pattern does not cause a heavy load of the intermediate data storage and transfer in “Phase I”.

The direct solution sequence SOL 108 supports the “Viscoelastic Material” pattern, but the modal solution sequence SOL 111 does not [33]. The reason is that the modal solution sequence SOL 111 needs new eigenmodes from the updated model. And the model update needs the model-rebuilding operation in “Phase I”. However, the “Viscoelastic Material” pattern cannot activate the model-rebuilding mechanism.

From another point, since the “Viscoelastic Material” pattern does not activate the model-rebuilding mechanism, it can be used with superelement techniques. For a steady-state dynamic analysis with superelement techniques, setting frequency-dependent properties through the “Viscoelastic Material” pattern does not incur multi-times dynamic reductions.

7.3 From Material Structural Damping to Modal Structural Damping

7.3.1 Non-Frequency-Dependent Material Patterns

As we know, material-level structural damping is assigned to materials in physical space. Supposing the model consists of "Non-Frequency-Dependent Material" will not trigger the model rebuilding mechanism. Thus, the material structural damping becomes the structural damping matrix after a model discretisation. In the next step, the projection of the structural material damping needs projection bases. And the projection bases are the eigenmodes from an eigenvalue extraction procedure. To accomplish the upper operations, people must execute a modal-space-based dynamic analysis, such as SOL 111. The input file template is analogous to the example shown in **Table 7.1**, but one thing that needs to do is to replace the statement SOL 108 with the statement SOL 111 in the "Executive Control Section". As the mathematical model, the structural damping matrix projects onto the modal space at the ith mode can be expressed as:

$$\begin{gathered} i[K_{dd}^{4}] = i\eta[K]_{sd} = i\sum_{mat}\eta_{mat}[K]_{mat} \\ \Downarrow \\ [K_{dd}^{4}]_{mode} = [\Phi]^{T}[K_{dd}^{4}][\Phi] \end{gathered} \tag{6.11}$$

In the upper equation, $[K_{dd}^{4}]$ is the structural damping matrix (emphasized by the superscript 4) of the model with given analysing DOFs (emphasized by the subscript dd. Please see subsection **3.5.3**). $[K_{dd}^{4}]_{mode}$ is the projected stiffness matrix of the damped part of the model. $[\Phi]$ is a matrix consisting of all extracted eigenvectors of the model.

The eigenmatrix $[\Phi]$ is extracted from the whole undamped system. It can uncouple the stiffness matrix of the whole model. In the following equation, $k_1, k_2, \cdots k_n$ are modal stiffness of corresponding modes, $[K]_G$ is the stiffness matrix of the whole model,

$$\begin{bmatrix} k_1 & & & \\ & k_2 & & \\ & & \ddots & \\ & & & k_n \end{bmatrix} = [\Phi]^{T}[K]_G[\Phi] \tag{6.12}$$

However, since the material damping factor η_{mat} of different materials is always different, there is more than one scaling factor between the elastic and anelastic parts of the whole model. Thus, the structural damping matrix $[K_{dd}^{4}]$ is always non-proportional to the stiffness matrix of the whole model $[K]_G$. One thing that needs to be noted is that the projected stiffness matrix of the damped part $[K_{dd}^{4}]_{mode}$ may still be coupled.

7.3.2 Classic Frequency-Dependent Material Patterns

When the model consists of frequency-dependent materials, the material properties need to be defined with specific material patterns. Based on the previous elaborations (see subsection **7.2.2** and subsection **7.2.3**), we know that the "Classic Frequency-Dependent Material" pattern can be used in the modal-space-based steady-state dynamic analysis SOL 111. In that case, the modal structural damping can be automatically calculated from the material structural damping at each property-altering frequency. If people set multiple property-altering frequencies, the model rebuilding mechanism will be activated multiple times. Thus, the eigenvalue extraction will be repeated multiple times. Therefore, the modal-space-based condensation of the new material structural damping will be executed multiple times. Then, the modal structural damping can be updated multiple times. To accomplish the upper operations, people must execute a modal-space-based dynamic analysis, such as SOL 111. The input file template is analogous to the example shown in **Table 7.2**, but one thing that needs to do is to replace the statement SOL 108 with the statement SOL 111 in the "Executive Control Section". **Figure 7.3** illustrates the modal structural damping $[K_{dd}^4(f_k)]_{mode}$ updating process. The property-altering frequency is f_k, and the number of this frequency is n. One property-altering frequency updates the material structural damping factor $\eta_{mat}(f_k)$, the material stiffness matrix $[K(f_k)]_{mat}$, the global stiffness damping $[K(f_k)]_G$, and the eigenmatrix $[\Phi(f_k)]$ one time.

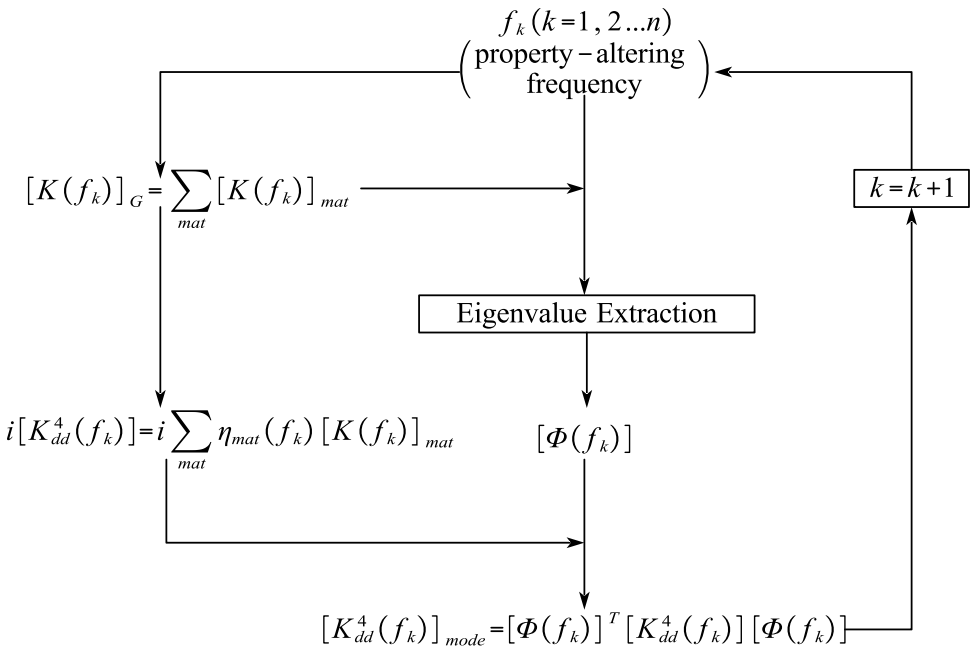


Figure 7.3: Calculation process of modal structural damping through Nastran's "SOL 111".

7.4 Extracting Modal Structural Damping Factors through DMAP

The structural damping factor is the ratio between anelastic and elastic parts (see section **6.3**). In theory, the modal structural damping factor can be calculated by using the following expression (the equal sign exists with the specific entailing condition):

$$\begin{bmatrix} \eta_1 & & & \\ & \eta_2 & & \\ & & \ddots & \\ & & & \eta_n \end{bmatrix} = or \neq \frac{[\Phi]^T [K_{dd}^4] [\Phi]}{[\Phi]^T [K]_G [\Phi]} \tag{6.13}$$

The necessary condition that the upper equal sign exists is that the structural damping matrix $[K_{dd}^4]$ can be uncoupled by the eigenmatrix $[\Phi]$. However, the modal structural damping matrix $[K_{dd}^4]_{mode}$ may still be coupled after the condensation (see subsection **7.3.1**). The modal structural damping factor can be approximately calculated with reasonable assumptions in real applications.

7.4.1 Non-Frequency-Dependent Material Patterns

Suppose the value of material structural damping is not too large. In that case, the coupling effects of the items in a modal damping matrix (the off-diagonal items in the modal damping matrix) are weak enough to be ignored. Then, a commonly used method called "Composite Modal Damping" can be used. Abaqus introduced this method (see section **8.4**), but Abaqus uses material critical damping ratios instead of material structural damping factors to calculate the modal damping through the "Composite Modal Damping" method. Similarly, using material structural damping factors to calculate modal damping factors through the "Composite Modal Damping" method is reasonable (for more information about the critical damping ratio and structural damping factor, please see section **6.3**). Then, calculations of modal damping factors through the "Composite Modal Damping" method can be expressed as:

$$\begin{gathered} k_i = \{\phi_i\}^T [K]_G \{\phi_i\} \\ \Downarrow \\ \eta_i = \frac{1}{k_i} \{\phi_i\}^T \left(\sum_{mat} \eta_{mat} [K]_{mat} \right) \{\phi_i\} \end{gathered} \tag{6.14}$$

In the upper equation, k_i is the modal stiffness of the ith mode; $\{\phi_i\}$ is the eigenvector of the ith mode; $[K]_G$ is the stiffness matrix of the whole modal; η_i is the modal structural damping factor of the ith mode.

Table 7.4: Structural modal damping (Nastran's DMAP, "Composite Modal Damping").

SECTIONS	CONTENT
Executive Control Section	SOL 103 COMPILE SEMODES ALTER 494 MATPCH LAMMAT//7 $ TYPE PARM, , I, N, C1 $ DO WHILE (C1 < *No.eigenmodes*) C1 = C1 + 1 $ MATMOD PHG/PHGC,/1/C1 $ SMPYAD PHGC, K4GG, PHGC/ KDAMP/ 3/ / /1/1/ / / /2 $ SMPYAD PHGC, KGG, PHGC/ KSTIF/ 3/ / /1/1/ / / /2 $ SOLVE KSTIF, KDAMP, , , / MDAMP/ $ MATPCH MDAMP/ /7 $ ENDDO ENDALTER CEND
Case-Control Section	METHOD = eig_id
Bulk Data Section	BEGIN BULK MAT1, mat_id, , G_m , , , , , η_m ENDDATA

Since the "Composite Modal Damping" method is not an endogenous simulation method in Nastran, there are no ready-made commands or statements to call related calculations. Fortunately, calculating the modal structural damping factor through the "Composite Modal Damping" method is a simple manipulation of intermediate matrices and vectors. And Nastran's DMAP (direct matrix abstracting program) supports inserting manipulation algorithms into any endogenous solution sequences to process and extract the intermediate data directly. As introduced before, using Nastran's DMAP to execute matrix calculations can directly utilize Nastran's executive system, which can automatically manage the intermediate data. In comparison, mathematical software like Matlab cannot automatically manage intermediate data. Thus, DMAP is indispensable for massive intermediate data calculations (see sections **2.4** and **2.5**).

This paper uses the endogenous solution sequence SOL 103 instead of SOL 111 as the DMAP inserting object. As we know, the endogenous solution sequence SOL 103 is used to execute the eigenvalue extraction analysis. In contrast, the endogenous solution sequence SOL 111 is used to execute the modal-space-based steady-state dynamic analysis. Since the eigenvalue extraction is an intermediate procedure in SOL 111, the DMAP logical layout of SOL 111 is much more sophisticated than the one in SOL 103. Then, the intermediate data of SOL 111 dives at a deep level that is hard to extract or manipulate. Thus, the endogenous solution sequence SOL 103 is the choice for controlling the difficulties of inserting manipulation algorithms. Finally, the modal structural damping factor can be calculated and extracted by inserting several specific DMAP commands into the "Executive Control Section" in an input file (see section **3.1**) of the eigenvalue extraction analysis SOL 103.

Table 7.4 displays a simplified input file template as an example for a more specific illustration. It is worth noting that the abbreviations "LAMMAT", "PHG", "K4GG", and "KGG" are systems endogenous matrix data blocks, and their meanings are the eigenvalue matrix, the normal modes eigenvector matrix, the structural damping matrix, and the stiffness matrix, respectively. In comparison, "PHGC", "KDAMP", "KSTIF", and "MDAMP" are user-assigned matrix data blocks of each mode. And they represent the eigenvector, the modal structural damping, the modal stiffness, and the modal structural damping factor, respectively. Then, "C1" is a user-defined real integer number for accounting for the cycles of the "DO WHILE" loop, and its value must be smaller than the acquired number of eigenmodes "*No. eigenmodes*". As for the meaning of other commands, please see the corresponding reference manuals. By the way, the descriptions in this paragraph are based on [16], [57], [58], [59], and [60].

7.4.2 Frequency-Dependent Material Patterns

In a numerical simulation, reasonable approximations can be made to overcome calculation difficulties once the accuracy is acceptable. Reasonable approximations must consider practical situations. In this paper, the processing object is the metamaterial panel. The metamaterial consists of non-frequency-dependent aluminium alloy and frequency-dependent melamine foam. There are at least two practical situations that need to be considered. One is that the damping factor is very small, and the simulation is executed in a low-frequency range. The other is that both the aluminium alloy and the melamine foam have elastic and anelastic properties.

Let us start talking about aluminium alloy (see subsection **2.2.2**). It is a non-frequency-dependent isotropic material, and its structural damping factor $\eta_{AL} = 0.0023$ is quite small. The value of its anelastic property is approximately three orders of magnitude lower than the values of its elastic property.

Then, let us concern the melamine foam (see subsection **2.2.3**). It is a frequency-dependent orthotropic material. Within the testing frequency of $0 - 220$ Hz, on the one hand, its damping factor changes from $\eta_{MELA} = 0.0028$ to $\eta_{MELA} = 0.0145$. The change of the structural damping factor means an approximately five-times enlargement of the anelastic property. On the other hand, its elastic property has almost no changes within the testing frequency. Furthermore, the value of its anelastic property is approximately three to two orders of magnitude lower than the values of its elastic property within the testing frequency.

Based on a comprehensive assessment, the anelastic properties of the aluminium alloy and the melamine foam are small enough that the "Composite Modal Damping" method is available. In addition, since the elastic property of the melamine foam has almost no changes within the testing frequency, this property can be seen as a constant parameter. People need not recalculate the eigenmatrix $[\Phi]$ since the elastic property of the aluminium alloy and the melamine foam can be seen as constant.

Figure 7.4 illustrates the calculation of the structural modal damping factor by the "Composite Modal Damping" method with reasonable assumptions introduced in the upper contents. In **Figure 7.4**, we can see that the stiffness matrices of aluminium alloy $[K]_{AL}$ and melamine foam $[K]_{MELA}$ are assumed to be constant. They combine to generate the constant global stiffness matrix $[K]_G$ of the metamaterial. Then, the eigenvalue extraction procedure generates a constant eigenmatrix $[\Phi]$, which contains eigenvectors with the number of "n". The eigenvector $\{\phi_i\}$ is one of them corresponding to the ith eigenmode. Then, we can get the ith modal stiffness k_i. After that, we can get the ith modal structural damping factor η_i^0 from the structural damping factors of aluminium alloy η_{AL} and melamine foam η_{MELA}^0. Repeating the procedure, in the same way, can get many structural modal damping factors η_1^0, η_1^0, ... η_n^0. One thing that needs to be noted is that, in the following process, the structural damping factor of melamine foam η_{MELA}^0 is assumed to be constant at this moment. This parameter needs further treatments to match the frequency dependency.

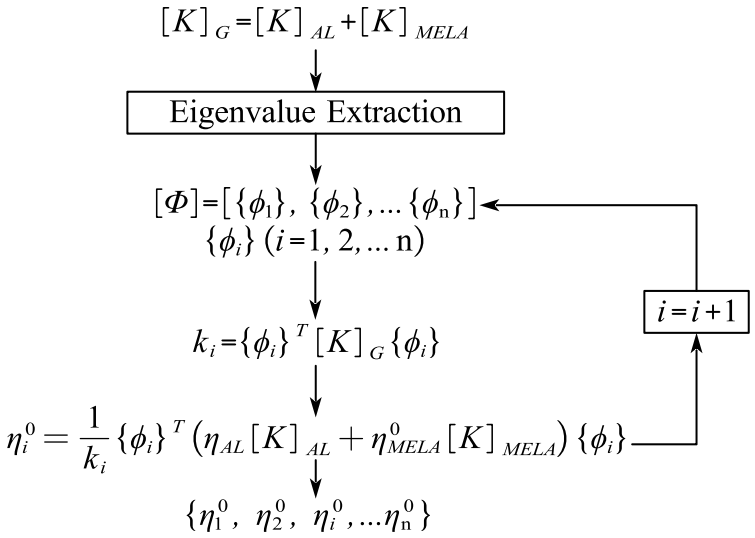


Figure 7.4: Calculations of non-frequency-dependent modal structural damping factors.

The operations illustrated in **Figure 7.4** can be accomplished by inserting several specific DMAP commands into the "Executive Control Section" in an input file (see section **3.1**) of the eigenvalue extraction analysis SOL 103. The last paragraph in subsection **7.4.1** has elaborated on the fundamental logic of calculations of a non-frequency-dependent modal structural damping. The non-frequency-dependent model is the precondition that SOL 103 can manipulate. Thus, the frequency dependency must be treated in succeeding steps after the

calculations in **Figure 7.4**. Then, additional reasonable assumptions are needed for a reliable treatment.

As we know, structural damping effects are proportional to the model's deformation. Since the melamine foam is much softer than the aluminium alloy, in dynamic analysis, the deformation of the melamine foam is larger than the deformation of the aluminium alloy. Thus, the melamine foam's structural damping effects dominate the metamaterial's damping effects. Therefore, using the melamine foam's structural damping factor to replace the aluminium alloy's structural damping factor is reasonable. Based on subsection **2.2.3**, we can get the frequency-dependent function of the melamine foam's structural damping factor:

$$\eta_{MELA}\left(\frac{f}{2\pi}\right)=\eta_{MELA}(\omega)=\frac{\left|\mathrm{Im}\left(1+\frac{b(i\omega/\beta)^{\alpha}}{1+(i\omega/\beta)^{\alpha}}\right)\right|}{\left|\mathrm{Re}\left(1+\frac{b(i\omega/\beta)^{\alpha}}{1+(i\omega/\beta)^{\alpha}}\right)\right|} \quad (f=2\pi\omega) \tag{6.15}$$

Let us set the reference melamine foam's structural damping factor η^0_{MELA} as the value at the frequency $f=1$ Hz or $\omega=2\pi$ rad/s. Then, the ratio of the frequency-dependent function $\eta_{MELA}(\omega)$ divides the reference factor η^0_{MELA} can be expressed as:

$$Ra(\omega)=\frac{\eta_{MELA}(\omega)}{\eta^0_{MELA}} \tag{6.16}$$

Assuming the frequency-dependent ratio $Ra(\omega)$ of the structural damping of the melamine foam is also available to the modal structural damping factor of the metamaterial. Thus, the frequency-dependent modal structural damping factor can be expressed as:

$$\eta_i(\omega_i)=\eta_i^0\cdot Ra(\omega_i) \tag{6.17}$$

In the upper equation, ω_i is the ith natural frequency.

7.5 Damping Setting Strategies in Specific Simulations

Section **15.1** elaborates on the efficiency benchmark of Nastran. As the subpart, subsection **15.1.4** describes the cases in which four candidate simulation strategies apply to the six simulation scenarios independently in the steady-state dynamic analyses. These six simulation scenarios are from two technical routes with three material states. We will see that the model is loaded with a concentrated force (see **Figure 15.2**). And the testing objects are four size-controlled panels (see **Figure 15.1**). We know that the panels consist of the metamaterial. And the metamaterial is composed of aluminium alloy and melamine foam. Thus, the mechanical properties of the aluminium alloy and melamine foam will influence the dynamic performances of the testing panels. Especially melamine foam is a frequency-dependent material.

Specifically, the four candidate simulation strategies are the “Ordinary Method”, the “ACMS Method”, the “List Superelement Method”, and the “Image Superelement Method”. Then, the two technical routes are the physical-space-based and the modal-space-based steady-state technical routes. Finally, the three material states are the “Classic Frequency-Dependent Material” pattern, the “Viscoelastic Material” pattern, and the “Non-Frequency-Dependent Material” pattern. These four simulation strategies, two technical routes, and three material states are concerning factors in choosing proper numerical damping models.

On the one hand, the aluminium alloy has non-frequency-dependent structural damping. Thus, the damping model introduced in subsection **7.2.1** is proper if adopting physical-space-based technical routes. Otherwise, if adopting the modal-space-based technical routes, the damping model introduced in subsection **7.3.1** is proper. On the other hand, melamine foam has frequency-dependent structural damping. Thus, if adopting physical-space-based technical routes, the numerical damping models introduced in subsection **7.2.2** and subsection **7.2.4** are proper. Otherwise, if adopting the modal-space-based technical routes, the damping model introduced in subsection **7.3.2** is proper.

Subsection **17.1.4** and section **17.2** elaborate on the modal-space-based steady-state vibro-acoustic analysis by Nastran coupling with Actran. Nastran calculates eigenmodes and the equivalent modal structural damping factors in that case. Then, multiply corresponding frequency-dependent factors with the extracted damping factors. Thus, Actran can do the modal-space-based vibroacoustic analysis by considering the frequency dependency. At the same time, the calculation method of the modal damping factor is introduced in subsection **7.4.2**, and the numerical damping modal as a direct input parameter is introduced in section **9.3**.

Chapter 8

Damping in Abaqus

8.1 Brief Introduction of General Damping in Abaqus

The general damping forms available in Abaqus are classified into three types which may be applied to three objects. Damping needs properly be applied to specific objects according to the subdivisions (see section **7.1**, natural subdivisions, technical subdivisions) of the original model [61]:

- damping of finite elements (this damping can be used in physical space or modal space, and it can be set with specific values based on finite element sets),
- damping of whole models (this damping can be used in physical space or modal space, and it can be set with specific values based on different models),
- damping of given modes (this damping can be used in modal space, and it can be set with specific values based on different modes).

The three damping types are [39]:

- structural damping (can be applied to finite element sets and whole models in physical space),
- Rayleigh viscous damping (can be applied to finite element sets and whole models in physical space),
- critical damping ratio (can be applied to specific modes or used as composite modal damping in modal space).

Simulation objects in this paper are metamaterial samples consisting of aluminium alloy and melamine foam, and the structural damping factors of these two constituent materials are known parameters. Thus, the following contents only elaborate on the numerical models of the material's structural damping.

It is worth noting that damping definitions in Nastran and Abaqus are different. In Nastran, the material-level damping needs to be defined through specific material patterns. All material properties, such as mass density, elastic modulus, and poison ratio, must be set in these specific material patterns. In contrast, Abaqus does not have predefined material patterns, and material properties are free states. Abaqus users can define materials by their ingredients. Then, users can set the material properties to assigned finite element sets or whole models. Thus, setting structural damping to assigned finite element sets in Abaqus is analogous to setting structural damping to given materials in Nastran. The structural damping assigned to specific finite element sets in Abaqus can also be called material structural damping. And the material structural damping factor is η_{mat}.

8.2 Applying Structural Damping to The Assigned Materials in Physical Space

In Abaqus, only several material properties, such as the frequency-domain viscoelasticity, can be set as frequency-dependent parameters. The frequency-domain viscoelasticity in Abaqus is a material property that can only be set with the isotropic material [38]. Its elastic part is the elasticity, and its anelastic part is analogous to structural damping. It is worth noting that the viscoelasticity and structural damping in Abaqus are two different material properties, and the structural damping in Abaqus is a non-frequency-dependent parameter. In contrast, the "Viscoelastic Material" in Nastran is a material pattern (see subsection **7.2.4**). It is based on general material patterns and utilizes two scaling factors to build the frequency-dependent function of the elastic and anelastic matrices of the model, respectively. The "Viscoelastic Material" pattern in Nastran can be set with isotropic, orthotropic, or anisotropic materials.

The numerical expression of the "Non-Frequency-Dependent Material" structural damping in Abaqus is the same as in Nastran. For more information, please see subsection **7.2.1**. For a more specific illustration of how to set the "Non-Frequency-Dependent Material" structural damping, here is a simplified input file template as an example. **Table 8.1** only displays the required settings for activating non-frequency-dependent structural damping in physical-space-based steady-state dynamic analysis. Other settings and non-correlated contents of an input file are ignored. For more information about an Abaqus input file with full contents, please see **Figure 4.1** in section **4.1**. For more information about the "KEYWORDS", please see the corresponding part in Abaqus's reference guide [61].

Table 8.1: Abaqus's non-frequency-dependent structural damping (physical space, traditional).

SECTIONS	KEYWORDS
Modal Data	∗ MATERIAL, NAME = mat_name ∗ DAMPING, STRUCTURAL = η_{mat} ∗ SOLID SECTION, ELSET = elset_name, MATERIAL = mat_name
History Data	∗ STEP ∗ STEADY STATE DYNAMICS, DIRECT ∗ END STEP

8.3 From Material Structural Damping to Modal Structural Damping

In Abaqus, the subspace-based and modal-space-based analyses are two technical routes that need the eigenvalue extraction as the prior step to generate the eigenmatrix $[\Phi]$. The eigenmatrix is the base of modal projection. Thus, both the subspace-based and modal-space-based analyses can calculate the modal damping. In addition, the modal-space-based dynamic analysis is more efficient than the subspace-based dynamic analysis. Thus, this paper adopts the modal-space-based dynamic analysis under the SIM architecture or traditional architecture to calculate the modal structural damping in the steady-state-dynamic analysis. By the way, the arguments in this paragraph are based on [36].

As we know, the structural damping in Abaqus is not a frequency-dependent parameter. Thus, the following expression of the numerical model of the modal structural damping is in a non-frequency-dependent form. Then, the structural damping matrix of a discretized model projects onto the modal space at the ith mode can be expressed as:

$$\begin{aligned} i\eta[K]_{sd} &= i\sum_{mat}\eta_{mat}[K]_{mat} \\ &\Downarrow \\ \eta[K]_{mode} &= [\Phi]^{T}\eta[K]_{sd}[\Phi] \end{aligned} \tag{7.1}$$

The eigenmatrix $[\Phi]$ can uncouple the stiffness matrix of the whole model. Supposing that $k_1, k_2, \cdots k_n$ are modal stiffness of corresponding modes and $[K]_G$ is the stiffness matrix of the whole model. The uncoupled modal stiffness matrix can be expressed as:

$$\begin{bmatrix} k_1 & & & \\ & k_1 & & \\ & & \ddots & \\ & & & k_n \end{bmatrix} = [K]_{mode} = [\Phi]^{T}[K]_{G}[\Phi] \tag{7.2}$$

However, it is worth noting that the projected stiffness matrix of the damped part $[K]_{mode}$ may still be coupled. Since the structural damping factor η_{mat} of different materials is always different, the stiffness matrix of the damped part $[K]_{sd}$ is always non-proportional to the stiffness matrix of the whole model $[K]_G$. Thus, the projected stiffness matrix $[K]_{mode}$ may still be coupled. In this situation, the SIM architecture does permit modal-space-based analyses to contain non-diagonal damping effects, but the traditional architecture does not [36]. The influences of the SIM and traditional architectures on modal damping effects will be verified in simulation practices (see subsections **15.2.2** and **15.2.3**).

Here is an example of two simplified input file templates to illustrate how to set "Non-Frequency-Dependent Material" structural damping in modal-space-based steady-state dynamic analyses under two software architectures. And the two input file templates only display the required settings. These settings are used for activating damping projections and applying the projected damping to a succeeding modal-space-based steady-state dynamic

analysis. Thus, other settings and non-correlated contents of an input file are ignored. **Table 8.2** indicates the analysis under SIM architecture, while **Table 8.3** indicates the analysis under traditional architecture. For more information about an Abaqus input file with full contents, please see **Figure 4.1** in section **4.1**. For more information about the "KEYWORDS", please see the corresponding part of Abaqus's reference guide [61].

Table 8.2: Abaqus's non-frequency-dependent structural damping (modal space, SIM).

SECTIONS	KEYWORDS
Modal Data	∗ MATERIAL, NAME = mat_name ∗ DAMPING, STRUCTURAL = η_{mat} ∗ SOLID SECTION, ELSET = elset_name, MATERIAL = mat_name
History Data	∗ STEP ∗ FREQUENCY, SIM ∗ STEADY STATE DYNAMICS ∗ END STEP

Table 8.3: Abaqus's non-frequency-dependent structural damping (modal space, traditional).

SECTIONS	KEYWORDS
Modal Data	∗ MATERIAL, NAME = mat_name ∗ DAMPING, STRUCTURAL = η_{mat} ∗ SOLID SECTION, ELSET = elset_name, MATERIAL = mat_name
History Data	∗ STEP ∗ FREQUENCY ∗ STEADY STATE DYNAMICS ∗ END STEP

8.4 Composite Modal Damping

Composite modal damping can be used in modal-space-based steady-state dynamic analyses. It firstly requires users to set critical damping ratios to each specific material, then transforms the critical damping ratios from each material to each mode. The transformation can be driven by traditional or SIM architecture. Based on the definition, the critical damping ratio of a material can be expressed as (see section **6.1**):

$$\zeta_{mat} = \frac{C_{mat}}{C_{cr,\,mat}} = \frac{C_{mat}}{\left(2\sqrt{KM}\right)_{mat}} \tag{7.3}$$

In the upper equation, C_{mat} is the viscous damping of a material; $C_{cr,\,mat}$ is the critical damping of a material; K is the general stiffness; M is the general mass.

8.4.1 Setting Composite Modal Damping through the Traditional Architecture

It is necessary to note that all arguments and deductions in this subsection are based on [36]. Then, we can learn that the critical damping ratio can be set as a material property to specific materials in traditional architecture. Through the traditional architecture, the method that transforms the critical damping ratios from each material to each mode is the modal projection of the mass-matrix weighted average. Thus, the equivalent modal critical damping ratio at the ith mode can be expressed as:

$$\zeta_i^M = \frac{1}{m_i}\{\phi_i\}^T\left(\sum_{mat}\zeta_{mat}[M]_{mat}\right)\{\phi_i\} \tag{7.4}$$

In the upper equation, ζ_i^M is the equivalent critical damping ratio of a mode generated by the mass-matrix weighted average; $\{\phi_i\}$ is a mode eigenvector; ζ_{mat} is the critical damping ratio of a material; $[M]_{mat}$ is the mass matrix of the elements with a single material. And $m_i = \{\phi_i\}^T[M]_G\{\phi_i\}$ is the modal mass; $[M]_G$ is the mass matrix of the whole model.

Before the modal projection, the prior step is the eigenvalue extraction. Then, each extracted eigenmode is used as the projection base. As we can see, the projection base is the independent eigenmode $\{\phi_i\}$ instead of eigenmatrix $\{\Phi\}$. This operation implies the calculation ignores potential coupling effects due to eigenmodes,

$$[\Phi]^T\left(\sum_{mat}\zeta_{mat}[M]_{mat}\right)[\Phi] \tag{7.5}$$

In other words, the product in equation **(7.5)** may still be coupled. The reason is that each material's critical damping ratio ζ_{mat} may differ. Thus, the global damping matrix is not proportional to the global stiffness matrix. Therefore, the modal projection of mass-matrix

weighted average under traditional architecture is an approximate method. The weak coupling effects between eigenmodes make sure its accuracy.

Here is a simplified input file template in **Table 8.4** for a more specific illustration of how to set the composite modal damping through traditional architecture. The input file template only displays the required settings. These settings activate the composite modal damping calculation and apply the calculated critical modal damping ratios to a succeeding modal-space-based steady-state dynamic analysis. Then, other settings and non-correlated contents of an input file are ignored. For more information about the "KEYWORDS", please see the corresponding part of Abaqus's reference guide [61].

Table 8.4: Abaqus's non-frequency-dependent composite modal damping (modal space, traditional).

SECTIONS	KEYWORDS
Modal Data	∗ MATERIAL, NAME = mat_name
	∗ DAMPING, COMPOSITE = ζ_{mat}
	∗ SOLID SECTION, ELSET = elset_name, MATERIAL = mat_name
History Data	∗ STEP
	∗ FREQUENCY
	∗ STEADY STATE DYNAMICS
	∗ MODAL DAMPING, VISCOUS = COMPOSITE
	∗ END STEP

8.4.2 Setting Composite Modal Damping through the SIM Architecture

It is necessary to note that all arguments and deductions in this subsection are based on [36]. Then, we can learn that with the SIM architecture, Users can directly assign critical damping ratios to the specified finite element sets or external-input matrices. Through the SIM architecture, the critical damping ratio from each material or input matrix to each mode can be calculated by a modal projection of mass-matrix or stiffness-matrix weighted averages method. Thus, the modal critical damping ratio at the ith mode using the mass-matrix weighted average method can be expressed as:

$$\zeta_i^M = \frac{1}{m_i}\{\phi_i\}^T\left(\sum_e \zeta_e^M [M]_e + \zeta_{matrix}^M [M]_{matrix}\right)\{\phi_i\} \tag{7.6}$$

In the upper equation, ζ_e^M is the critical damping ratio of the specified finite elements; $[M]_e$ is the matrix of the specified finite elements; ζ_{matrix}^M is the critical damping ratio of the input mass matrix; $[M]_{matrix}$ is the input mass matrix.

The modal critical damping ratio at the ith mode using the stiffness-matrix weighted average method can be expressed as:

$$\zeta_i^K = \frac{1}{k_i}\{\phi_i\}^T\left(\sum_e \zeta_e^K [K]_e + \zeta_{matrix}^K [K]_{matrix}\right)\{\phi_i\} \tag{7.7}$$

In the upper equation, ζ_i^K is the modal critical damping ratio at the ith mode generated by the given method; ζ_e^K is the critical damping ratio of the specified finite elements; $[K]_e$ is the stiffness matrix of the specified finite elements; ζ_{matrix}^K is the critical damping ratio of the input stiffness matrix; $[K]_{matrix}$ is the input stiffness matrix.

As in the prior step, the eigenvalue extraction must be driven by the Lanczos eigensolver under the SIM architecture. As we can see, the projection base is the independent eigenmode $\{\phi_i\}$ instead of eigenmatrix $[\Phi]$. This operation implies the calculation ignores the potential coupling effects due to eigenmodes. In other words, the mass-matrix-based product

$$[\Phi]^T\left(\sum_e \zeta_e^M [M]_e + \zeta_{matrix}^M [M]_{matrix}\right)[\Phi] \tag{7.8}$$

and the stiffness-matrix-based product

$$[\Phi]^T\left(\sum_e \zeta_e^K [K]_e + \zeta_{matrix}^K [K]_{matrix}\right)[\Phi] \tag{7.9}$$

may still be coupled. The reason is that the critical damping ratios ζ_e^K and ζ_e^M of each finite-element set may be different. Thus, the global damping matrix is not proportional to the global mass or stiffness matrix. In addition, the input mass matrix $[M]_{matrix}$ and input stiffness matrix $[K]_{matrix}$ are not always be uncoupled by the eigenmatrix $[\Phi]$, since input matrices and the inner matrices belong to different systems. Therefore, the modal projection of the mass-matrix weighted average and the stiffness-matrix weighted average under the SIM architecture are two approximations. However, if the coupling effects between eigenmodes are weak, this method only incurs small errors.

Here is a simplified input file template in **Table 8.5** for a more specific illustration of how to set the composite modal damping through the SIM architecture. And the input file template only displays the required settings for activating the composite modal damping calculation and applying the calculated critical modal damping ratios to a succeeding modal-space-based steady-state dynamic analysis. Thus, other settings and non-correlated contents of an input file are ignored. For more information about the "KEYWORDS", please see the corresponding part of Abaqus's reference guide [61].

Table 8.5: Abaqus's non-frequency-dependent composite modal damping (modal space, SIM).

SECTIONS	KEYWORDS
Modal Data	* ELSET = elset_name * MATRIX INPUT, MATRIX = MASS *or* STIFFNESS
History Data	* STEP * FREQUENCY, EIGENSOLVER = LANCZOS, SIM * COMPOSITE MODAL DAMPING, MASS MATRIX INPUT = ζ^M_{matrix}, STIFFNESS MATRIX INPUT = ζ^K_{matrix} elset_name, ζ^M_{e}, ζ^K_{e} * STEADY STATE DYNAMICS * MODAL DAMPING, VISCOUS = COMPOSITE * END STEP

8.5 Modal Structural Damping from Direct Input

In Abaqus, users can directly assign modal structural damping to given modes using one of the three modal-damping parameters (see **Table 8.6**). Supposing the modal structural damping of ith mode is η_i. Here is a simplified input file template to illustrate how to set the direct input modal structural damping [61].

Table 8.6: Abaqus's direct-input modal damping.

SECTIONS	KEYWORDS
Modal Data	…
History Data	* STEP * FREQUENCY * STEADY STATE DYNAMICS * MODAL DAMPING, STRUCTURAL mode_number-i, mode_number-j, η_i ⋮ * END STEP

8.6 Damping Setting Strategies in Specific Simulations

Section **15.2** elaborates on the efficiency and relative accuracy benchmarks of Abaqus. Subsections **15.2.2** and **15.2.3** describe the performance of two candidate simulation strategies applied to two simulation scenarios. We will see that the model is loaded with a concentrated force (see **Figure 15.2**). And the testing objects are four size-controlled testing panels (see **Figure 15.1**). We know that Abaqus cannot set the frequency dependency to orthotropic materials (see subsection **4.3.4**). Thus, simulations in section **15.2** run with non-frequency-dependent models.

The “Ordinary Method” and the “AMS Method” are candidate simulation strategies. Then, the physical-space-based and the modal-space-based steady-state dynamic analyses are the two technical routes. The simulation strategies and the technical routes must cooperate with proper numerical damping models.

We know that the damping of the metamaterial is structural. Abaqus cannot support the frequency dependency on the orthotropic material (the melamine foam). Thus, the two materials of metamaterial samples are assumed to be constant. Thus, the damping model introduced in section **8.2** is proper in physical-space-based steady-state dynamic analyses. In the same way, the damping model introduced in section **8.3** is proper in modal-space-based steady-state dynamic analyses.

Subsection **17.1.4** elaborates on the physical-space-based steady-state vibro-acoustic analysis through Abaqus. The numerical damping model introduced in section **8.2** is proper in this case.

Chapter 9

Damping in Actran

On the one hand, the material's numerical models in Actran are analogous to the ones in Nastran but different from the ones in Abaqus. Material numerical models in Actran and Nastran are material patterns. Each one is an enveloped property group. A property group means that a material pattern independently compounds with many specific material properties. In contrast, Abaqus's material properties are free-state parameters because Abaqus does not have predefined material patterns. On the other hand, material properties in Actran are in complex form, while material properties in Nastran and Abaqus must be in real form.

To some extent, the imaginary parts of the complex material properties in Actran behave like damping. Such as the complex elastic moduli, its real part is the elasticity, its imaginary part is the anelasticity, and the anelasticity is analogous to structural damping. Besides the complex material properties, Actran supports directly setting the damping factors in material patterns. Such as the "Isotropic Solid" and "Orthotropic Solid" material patterns can be set with the structural damping factor. In contrast, the "Fluid" material pattern can be set with the critical viscous damping factor. In other words, Actran has two ways to set damping effects at the material level. And setting damping effects through the complex material properties is a way to control damping parameters specifically. For example, in the "Orthotropic Solid" material pattern, the six independent elastic moduli can be set independently with real, imaginary, or complex values. And the material properties and the material-level damping factors in Actran can be constant or frequency-dependent in physical space. This case is analogous to the ones in Nastran but different from the ones in Abaqus. By the way, the arguments in this paragraph are based on [47].

Besides the material level, Actran can set the global structural damping factor to the whole structural model and the global viscous damping factor to the whole acoustic model. The global damping factors in Actran can be constant or frequency-dependent in physical space. This case is different from the ones in Nastran and Abaqus. In Nastran and Abaqus, the global damping cannot be a frequency-dependent parameter. At the same time, besides the physical-space-based analyses, Actran supports modal structural damping and modal viscous damping from the external input in modal space. The modal damping factor in Actran can be a constant or

mode-dependent parameter. This case is analogous to the ones in Nastran and Abaqus. The arguments in this paragraph are based on [44] and [45].

For simplicity, the following content only elaborates on the material's complex elasticities, the structural damping at the material level, and the modal structural damping from direct input. These three damping forms will be applied to testing metamaterial samples in subsections **15.3.2**, **15.3.3**, and section **17.2**.

9.1 Examples of Setting Material's Complex Elasticity in Physical Spaces

As we know, the material properties in Actran can be frequency-dependent or non-frequency-dependent in physical space. Here, two simplified input file templates illustrate how to set the material's complex elasticity. One is correlated with non-frequency-dependent elasticity (see **Table 9.1**), and another is correlated with frequency-dependent elasticity (see **Table 9.2**). For example, the material pattern adopted in the two templates is the "Isotropic Solid", and the simulation is the physical-space-based steady-state dynamic analysis. These two input file templates only display the required settings for frequency-dependent and non-frequency-dependent complex elasticities since other settings and non-correlated subsections of an input file are not displayed. For more information about the full input file template, please see section **5.1**.

9.1.1 Non-Frequency-Dependent Complex Elasticity

Table 9.1: Actran's non-frequency-dependent complex elasticity.

SECTIONS	STATEMENTS
Main Data Groups	BEGIN ACTRAN BEGIN FREQUENCY_DOMAIN ⋮ END FREQUENCY_DOMAIN
Material Data Groups	BEGIN MATERIAL material_id ISOTROPIC_SOLID SOLID_DENSITY ρ_{mat} YOUNG_MODULUS $\left\{\mathrm{Re}\left(\tilde{E}_{mat}\right),\ \mathrm{Im}\left(\tilde{E}_{mat}\right)\right\}$ POISSON_RATIO ν_{mat} END MATERIAL material_id
Main Data Groups	END ACTRAN

The numerical expression of the "Non-Frequency-Dependent Material" structural damping in Actran is the same as in Nastran. For more information, please see subsection **7.2.1**. In the "Material Data Groups" of **Table 9.1**, ρ_{mat} indicates the material mass density, and it is a real constant parameter; $\tilde{E}_{mat}$ indicates the material elastic modulus, and it is a complex constant parameter in this case; v_{mat} indicates the material poison's ratio, and it is a real constant parameter. The material_id indicates the material identity number and is a real integer number. Please see Actran's reference guide [47] and [62] for the meaning of specific statements.

9.1.2 Frequency-Dependent Complex Elasticity

The numerical expression of the frequency-dependent material structural damping in Actran is the same as in Nastran. For more information, please review subsection **7.2.2**.

Table 9.2: Actran's frequency-dependent complex elasticity.

SECTIONS	STATEMENTS
Main Data Groups	BEGIN ACTRAN BEGIN FREQUENCY_DOMAIN ⋮ END FREQUENCY_DOMAIN
Material Data Groups	BEGIN MATERIAL material_id ISOTROPIC_SOLID SOLID_DENSITY ρ_{mat} YOUNG_MODULUS TABLE table_id POISSON_RATIO v_{mat} END MATERIAL material_id
Analysis Parameter Data Groups	BEGIN TABLE table_id table_type (1) table_size (n) f_1 $\left\{\mathrm{Re}\left(\tilde{E}_{mat}(f_1)\right),\ \mathrm{Im}\left(\tilde{E}_{mat}(f_1)\right)\right\}$ f_2 $\left\{\mathrm{Re}\left(\tilde{E}_{mat}(f_2)\right),\ \mathrm{Im}\left(\tilde{E}_{mat}(f_2)\right)\right\}$ ⋮ ⋮ f_n $\left\{\mathrm{Re}\left(\tilde{E}_{mat}(f_n)\right),\ \mathrm{Im}\left(\tilde{E}_{mat}(f_n)\right)\right\}$ END TABLE table_id
Main Data Groups	END ACTRAN

From **Table 9.1** to **Table 9.2**, the complex elastic modulus becomes a frequency-dependent parameter, and the way is to add a table to describe the corresponding frequency function. See

"Analysis Parameter Data Groups" in **Table 9.2**; there is a table with a given identity number table_id that lists the frequency function of the complex elastic modulus $\tilde{E}_{mat}$, and the number of the frequency is n. This table will be referenced by the "YOUNG_MODULUS" in the "Isotropic Material" data group through the table_id. The table_type equals to 1 means this table is a frequency table that recoding real and imaginary data at each frequency, and the table_size equals to the number of frequencies. Please see Actran's reference guide [47], [62], and [63] for the meaning of specific statements.

9.2 Applying Structural Damping to The Assigned Materials in Physical Space

As we know, the material structural damping in Actran can be frequency-dependent or non-frequency-dependent in physical space. Here, two simplified input file templates illustrate the settings of material structural damping. One is correlated with the non-frequency-dependent structural damping factor (see **Table 9.3**), and another is correlated with the frequency-dependent structural damping factor (see **Table 9.4**). For example, the material pattern adopted in the two templates is the "Isotropic Solid", and the simulation is the physical-space-based steady-state dynamic analysis. These two input file templates only display the required settings for frequency-dependent and non-frequency-dependent structural damping factors. Thus, other settings and non-correlated contents of an input file are ignored. For more information about a full input file template, please see section **5.1**.

9.2.1 Non-Frequency-Dependent Structural Damping

In "Material Data Groups" of **Table 9.3**, E_{mat} indicates the material elastic modulus. The material_id indicates the material identity number. And they are set to be real constant parameters in this case. Please see Actran's reference guide [47] and [62] for the meaning of specific statements.

Table 9.3: Actran's non-frequency-dependent structural damping.

SECTIONS	STATEMENTS
Main Data Groups	BEGIN ACTRAN
	BEGIN FREQUENCY_DOMAIN ⋮ END FREQUENCY_DOMAIN
Material Data Groups	BEGIN MATERIAL material_id ISOTROPIC_SOLID SOLID_DENSITY ρ_{mat} YOUNG_MODULUS $\{E_{mat},\ 0\}$ POISSON_RATIO v_{mat} DAMPING η_{mat} END MATERIAL material_id
Main Data Groups	END ACTRAN

9.2.2 Frequency-Dependent Structural Damping

From **Table 9.3** to **Table 9.4**, the complex elastic modulus becomes a frequency-dependent parameter, and the way is to add a table to describe the corresponding frequency function. See "Analysis Parameter Data Groups" in **Table 9.4**; there is a table with a given table identity number table_id that lists the frequency function of the structural damping factor η_{mat}, and the number of frequencies is n. This table will be referenced by the "DAMPING" in the "Isotropic Material" data group through the table_id. The table_type equals to 1 means this table is a frequency table that recoding real and imaginary data at each frequency, and the table_size equals to the number of frequencies. Please see Actran's reference guide [47], [62], and [63] for the meaning of specific statements.

Table 9.4: Actran's frequency-dependent structural damping.

SECTIONS	STATEMENTS
Main Data Groups	BEGIN ACTRAN
	BEGIN FREQUENCY_DOMAIN ⋮ END FREQUENCY_DOMAIN
Material Data Groups	BEGIN MATERIAL material_id ISOTROPIC_SOLID SOLID_DENSITY ρ_{mat} YOUNG_MODULUS $\{E_{mat},\ 0\}$ POISSON_RATIO ν_{mat} DAMPING TABLE table_id END MATERIAL material_id
Analysis Parameter Data Groups	BEGIN TABLE table_id table_type (1) table_size (n) f_1 $\{\eta_{mat}(f_1),\ 0\}$ f_2 $\{\eta_{mat}(f_2),\ 0\}$ ⋮ ⋮ f_n $\{\eta_{mat}(f_n),\ 0\}$ END TABLE table_id
Main Data Groups	END ACTRAN

9.3 Modal Structural Damping from Direct Input

In Actran, users can directly assign modal structural damping to given modes. Supposing the modal structural damping of ith mode is η_i, a simplified input file template only displays the required settings to direct input the modal structural damping factor. As an example, the specific component adopted in the template is the "MODAL_ELASTIC", and the simulation is the modal-space-based steady-state dynamic analysis. Please see Actran's reference guide [44], [45], and [62] for the meaning of specific statements.

Table 9.5: Actran's direct-input modal structural damping.

SECTIONS	STATEMENTS
Main Data Groups	BEGIN ACTRAN_MODAL BEGIN FREQUENCY_DOMAIN ⋮ END FREQUENCY_DOMAIN
Component Data Groups	BEGIN COMPONENT MODAL_ELASTIC DOMAIN domain_name_list MODES_FORMAT NASTRAN_OP2 MODES_FILE modes_fname NUMBER_OF_MODES modes_number(n) BEGIN MODAL_DAMPING modes_number(n) (id_number) (type) (value) mode_id_1 2 η_1 mode_id_i 2 η_i ⋮ 2 ⋮ mode_id_n 2 η_n END MODAL_DAMPING END COMPONENT
Main Data Groups	END ACTRAN_MODAL

In the "Component Data Group" of **Table 9.5**, domain_name_list indicates the topology domains of the specific component; modes_fname indicates the file name which contains the input eigenmodes; modes_number(n) means the number of input modes is n; the id_number indicate the identification number of each mode, and the lowest number is 1, while the highest

number is n; type indicate modal damping type, number 2 means the damping is modal structural damping; value means a modal structural damping value of each mode.

9.4 Damping Setting Strategies in Specific Simulations

Section **15.3** elaborates on the efficiency benchmark of Actran. Subsections **15.3.2** and **15.3.3** describe the process of steady-state dynamic analyses in Actran. In these analyses, the model is loaded with a concentrated force. And the testing objects are four testing panels of different sizes. We know that the testing panels are constituted of the metamaterial. And the metamaterial is composed of aluminium alloy and melamine foam. Thus, the mechanical properties of the aluminium alloy and melamine foam influence the dynamic performances of the testing panels.

The candidate simulation method is the “Ordinary Method”, and the technical route is the physical-space-based steady-steady dynamic analysis. We know that the damping type of the metamaterial is structural damping from two composite materials. On the one hand, the aluminium alloy has non-frequency-dependent structural damping. Thus, the numerical model of damping introduced in subsection **9.2.1** is proper if the technical route is the physical-space-based analysis. On the other hand, melamine foam has frequency-dependent anelasticity. Thus, if the technical route is the physical-space-based analysis, the numerical damping model introduced in subsection **9.2.2** is proper.

Section **17.2** elaborates on physical-space-based steady-state vibro-acoustic analysis by Actran. The numerical damping model introduced in subsection **9.2.2** is proper in this case. Section **17.2** and section **17.3** elaborate on the modal-space-based steady-state vibro-acoustic analysis by Nastran with Actran.

Nastran calculates eigenmodes of acoustic components, structural components, and base-state modal structural damping factors at each mode. In addition, the extracted modal structural damping factors must multiply a frequency function to calculate their frequency-dependent values. After that, these extracted acoustic and structural eigenmodes and the modes-dependent modal damping are modal parameters for modal-space-based analyses in Actran. Actran can directly read the files that contain the modal parameters from Nastran. More specifically, the calculation method of the equivalent modal damping factor is introduced in subsection **7.4.2**, and the numerical model of damping as a direct input parameter is introduced in subsection **9.3**.

Chapter 10

Fundamentals of Acoustic Theories

The study of steady-state vibro-acoustic analyses mainly concerns three objects: acoustic sources, acoustic wave propagations, and boundary conditions. Simulations of vibro-acoustic systems are sophisticated due to the derivatives of these three objects. Thus, vibro-acoustic simulations always have heavy calculations. In commercial FEM software, well-developed mathematical models and competent software architectures can efficiently drive numerical solutions. However, it is not easy to understand the intrinsic meaning of the mathematical models in the FEM software. These mathematical models have sophisticated varieties due to complex application scenarios. Thus, it is necessary to make clear some fundamental theories of acoustics before discussing the mathematical models in FEM software.

10.1 Governing Equations of Fluid Media

10.1.1 The Relation Between Acoustic Pressure and Fluid Velocity

The mathematical model of a linear acoustic system in steady-state dynamic analyses comes from small oscillation theory. Thus, governing equations can be established from the harmonic motion of a system. Let us start with the harmonic motion [64]. The acoustic pressure can be expressed as:

$$p_f(\bar{\boldsymbol{x}},t)=\widetilde{P}_f(\bar{\boldsymbol{x}})e^{i\omega t},\ \widehat{P}_f(\bar{\boldsymbol{x}})=\left|\widetilde{P}_f(\bar{\boldsymbol{x}})\right| \tag{9.1}$$

And the fluid velocity can be expressed as:

$$v_f(\bar{\boldsymbol{x}},t)=\widetilde{V}_f(\bar{\boldsymbol{x}})e^{i\omega t},\ \widehat{V}_f(\bar{\boldsymbol{x}})=\left|\widetilde{V}_f(\bar{\boldsymbol{x}})\right| \tag{9.2}$$

In the upper equations, $p_f(\bar{\boldsymbol{x}},t)$ is the harmonic fluid pressure and $v_f(\bar{\boldsymbol{x}},t)$ is the harmonic fluid velocity; they are functions of position vector and time. $\bar{\boldsymbol{x}}=[x,y,z]^T$ is the position vector in the Cartesian coordinate system. $\widetilde{P}_f(\bar{\boldsymbol{x}})$ is the complex amplitude of fluid pressure; $\widehat{P}_f(\bar{\boldsymbol{x}})$ is the modulus of this complex amplitude. $\widetilde{V}_f(\bar{\boldsymbol{x}})$ is the complex amplitude of fluid velocity; $\widehat{V}_f(\bar{\boldsymbol{x}})$ is the modulus of this complex amplitude.

Between acoustic pressure and fluid velocity, the bridge is acoustic impedance, and its inverse is acoustic admittance [65]. The acoustic impedance $z(\bar{\boldsymbol{x}})$ is a function of the position vector. From fluid velocity to acoustic pressure, the relation bridged by acoustic impedance can be expressed as:

$$p_f(\bar{\boldsymbol{x}},t)=z(\bar{\boldsymbol{x}})v_f(\bar{\boldsymbol{x}},t) \tag{9.3}$$

10.1.2 Linearized Euler's Equation

For studying the dynamic performance of acoustic media based on acoustic pressure and fluid velocity [64], the governing equation is the linearised Euler's equation:

$$\nabla p_f(\bar{\boldsymbol{x}},t)+\rho_f\frac{\partial v_f(\bar{\boldsymbol{x}},t)}{\partial t}=0 \tag{9.4}$$

A typical usage of this equation is to formulate boundary conditions of a fluid-structure coupling model. In the frequency domain, this equation can be expressed as:

$$\nabla\widetilde{P}_f(\bar{\boldsymbol{x}})+i\omega\rho_f\widetilde{V}_f(\bar{\boldsymbol{x}})=0 \tag{9.5}$$

In the upper equations, ρ_f is the fluid mass density; ω is the angular speed.

10.1.3 Wave Equation and Helmholtz Equation

For studying the fluid media properties of a linear system in dynamics [64], the governing equation is the linearised wave equation:

$$\nabla^2 p_f(\bar{\boldsymbol{x}},t) - \frac{1}{c^2}\frac{\partial^2 p_f(\bar{\boldsymbol{x}},t)}{\partial t^2} = 0 \tag{9.6}$$

From the time domain to the frequency domain, the wave equation becomes the Helmholtz equation:

$$\nabla^2 \widetilde{P}_f(\bar{\boldsymbol{x}}) + k^2 \widetilde{P}_f(\bar{\boldsymbol{x}}) = 0 \tag{9.7}$$

Sound speed and wave numbers must be introduced in the upper two equations. The sound speed [66] can be expressed as:

$$c = \sqrt{\frac{K_f}{\rho_f}} \tag{9.8}$$

And the wave numbers [67] can be expressed as:

$$k = \frac{2\pi f}{c} = \frac{\omega}{c} = \frac{2\pi}{\lambda} \tag{9.9}$$

In the expressions of sound speed and wave numbers, ρ_f is the fluid mass density, K_f is the fluid bulk modulus, and λ is the wavelength.

10.2 Acoustic Sources

In general, acoustic sources are vibrations in the fluid domain that generate acoustic waves, which propagate outward from the sources. The waves' propagation relies on acoustic media. Thus, the media's mechanical properties dominate the characteristics of the waves' propagation. However, the wave propagation mechanism changes at the interface of different media, such as fluid and structure. Based on the linearised Euler's equation (see equation **(9.4)**), we can see how fluid's accelerations transform into pressures. The fluid's accelerations at the interface are equivalent to the ones of the structure.

The following contents introduce some commonly used sources to enhance fundamental concepts. These sources include the spherical source, the planar source, the cylindrical source, and the simple source. Their mathematical models are based on specific chapters in technical books [68] and [69]. Then, the acoustic source applied in this paper is the diffuse incident field. And the mathematical model of the diffuse incident field is over the fundamental theory scope,

10.2.1 Harmonic Spherical Sources

The spherical source can also be called the point source. It is a source that radiates acoustic waves from a point to around domains in a spherical form, and the wave strength is inversely proportional to the radius of the wave-spreading sphere [68]. The spherical source in the form of a harmonic acoustic pressure can be expressed as:

$$p_f(r,t) = \tilde{P}_f(r)e^{i(\omega t - kr)} = \frac{\tilde{A}}{r}e^{i(\omega t - kr)},\ \hat{A} = \left|\tilde{A}\right| \tag{9.10}$$

In the upper equation, r is the radius of the spherical wave; $\tilde{P}_f(r) = \tilde{A}/r$ is the complex pressure amplitude on the sphere surface; kr is the motion delay of points away from the; k is the wave number; $\tilde{A}$ is the complex source pressure amplitude; $\hat{A}$ is the modulus of the complex source amplitude.

The intensity of the spherical source can be expressed as:

$$I(r) = \frac{\tilde{A}\tilde{A}^H}{r^2}\frac{1}{2\rho_f c} = \frac{\hat{A}^2}{r^2}\frac{1}{2\rho_f c} \tag{9.11}$$

In the upper equation, $\tilde{A}^H$ is the complex conjugate of $\tilde{A}$.

10.2.2 Harmonic Plane Wave Sources

The plane wave source will generate acoustic waves with the same amplitude and phase on any plane perpendicular to the wave travelling direction [68]. Thus, on the perpendicular plane, the source in the form of a harmonic acoustic pressure can be expressed as:

$$p_f(r,t) = \tilde{P}_f e^{i(\omega t - kr)} = \tilde{A}e^{i(\omega t - kr)},\ \hat{A} = \left|\tilde{A}\right| \tag{9.12}$$

In the upper equation, r is the distance between the evaluating point and the source plane. k is the wave number. $\tilde{A}$ is the complex source pressure amplitude; $\hat{A}$ is the modulus of the complex source amplitude.

The intensity of the plane wave source can be expressed as:

$$I = \frac{\tilde{A}\tilde{A}^H}{2\rho_f c} = \frac{\hat{A}^2}{2\rho_f c} \tag{9.13}$$

10.2.3 Harmonic Cylindrical Sources

The cylindrical source can also be called the line source [68]. The expression of this kind of source is more sophisticated:

$$p_f(r,t) = \tilde{A}H_0^{(2)}(kr)e^{i(\omega t)},\ \hat{A} = \left|\tilde{A}\right| \tag{9.14}$$

In the upper equation, r is the distance between the evaluating point and the central axis of the cylinder. The Hankel function of an outgoing cylindrical wave with azimuthal symmetry can be expressed as:

$$H_0^{(2)}(kr) = J_0(kr) - iY_0(kr) \tag{9.15}$$

The Hankel function consists of Bessel functions J_0 and Y_0, and their asymptotic forms can be expressed as:

$$\begin{aligned} J_0(kr) &\to (2/\pi kr)^{1/2}\cos(kr - \pi/4) \\ Y_0(kr) &\to (2/\pi kr)^{1/2}\sin(kr - \pi/4) \end{aligned} \tag{9.16}$$

With the assumption of large kr, the cylindrical source in an asymptotic form can be expressed as:

$$p_f(r,t) = \tilde{A}(2/\pi kr)^{1/2}e^{i(\omega t - kr + \pi/4)} \tag{9.17}$$

When the situation $kr \ll 1$ is satisfied, the intensity of the cylindrical source can be expressed as:

$$I(r) = \frac{\tilde{A}\tilde{A}^H}{\pi kr}\frac{1}{\rho_f c} = \frac{\hat{A}^2}{\pi kr}\frac{1}{\rho_f c} \tag{9.18}$$

10.2.4 Harmonic Simple-Sources

The simple source comes from the theory of pulsating sphere, and the pulsating sphere is a source with spherical waves [69]. The simple source in the form of a harmonic acoustic pressure can be expressed as:

$$p_f(r,t) = i\frac{1}{2}\rho_f c\left(\tilde{Q}/\lambda r\right)e^{i(\omega t - kr)},\ \hat{Q} = \left|\tilde{Q}\right| \tag{9.19}$$

In the upper equation, r is the radius of the spherical wave. λ is the acoustic wavelength. $\tilde{Q}$ is the complex source strength, and it can be expressed as:

$$\tilde{Q} = 4\pi r^2 \tilde{V}_0 \tag{9.20}$$

Then, $\hat{Q}$ is the modulus of the complex source strength. And $\tilde{V}_0$ is the complex amplitude of the vibrating velocity of a pulsating sphere.

The source intensity of the simple source can be expressed as:

$$I(r) = \frac{1}{8}\rho_f c\frac{\tilde{Q}\tilde{Q}^H}{(\lambda r)^2} = \frac{1}{8}\rho_f c\frac{\hat{Q}^2}{(\lambda r)^2} \tag{9.21}$$

In the upper equation, $\tilde{Q}^H$ is the complex conjugate of $\tilde{Q}$.

10.2.5 Diffuse Incident Field

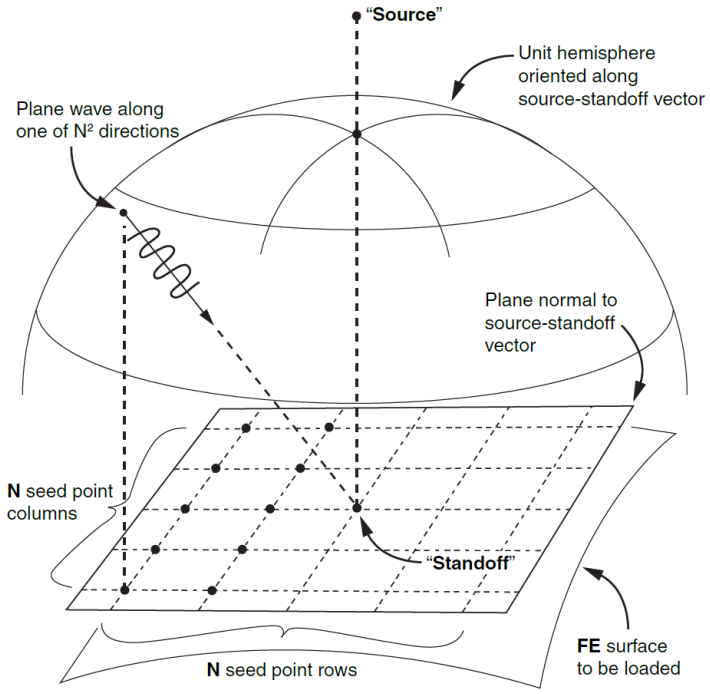


Figure 10.1: Diffuse incident field [40].

The Institute of Noise Control Engineering (INCE-USA) proposes the definition for a diffuse field: a sound field in which the time average of the mean-square sound pressure is everywhere the same and the flow of acoustic energy in all directions is equally probable. Such a diffuse

acoustic field is usually produced experimentally by activating acoustic sources in a reverberant chamber, the multiple reflections along the boundary walls leading to a diffuse field. Some simulation software supports the application of the diffuse incident field, such as Actran and Abaqus; however, software like Nastran does not. The fundamental theory of the diffuse indent field of different software is analogous. For more information, please see the corresponding section in the Actran theory guide. Making a clear understanding of the diffuse incident field theory is not easy. Its mathematical models are much more complex than the upper four acoustic sources. Different simulation software may have different methodologies, and the varieties are dependent on the software architecture and application scenarios. Thus, this paper does not elaborate on the theory of the diffuse incident field. The focal point is its application. By the way, the arguments in this paragraph are based on [51].

For a glance at a diffuse field model, the corresponding section in the Abaqus theory guide [40] gives an illustration. Then, we can understand the mathematical models of the diffuse incident field in Actran and Abaqus are analogous but not congruent. Firstly, the meaning of the mathematical models and the model application scenarios in Actran and Abaqus are analogous. The meaning is that the diffuse incident field simulates the characteristics of reverberant spaces or other situations in which waves from many directions strike a surface. Then, the application scenario is the intentionally constructed reverberant chambers in acoustic test facilities for sound transmission loss measurements. And the diffuse incident field is a proper loading form used in steady-state dynamic analysis to observe the frequency response function. It models the sound field incident on a surface exposed to a reverberant chamber (see **Figure 10.1**): the field is assumed to be equivalent to multiple plane waves arriving from directions distributed on a hemisphere. In comparison, the loading forms like the “spherical source”, the “plane wave source”, the “cylindrical source”, and the “simple source” are specifically defined as distinct sound sources. Thus, using these acoustic sources in FEM software is analogous to transient analysis.

Secondly, the definition of controlling parameters and the simulating-space access (physical-space-based, modal-space-based analysis) of the diffuse incident field in Actran and Abaqus is different. More controlling parameters are adjustable for defining a diffuse incident filed in Actran than in Abaqus. Abaqus must use the “source-standoff point” system, which differs from the system in Actran. On the other hand, Actran supports applying the diffuse incident field in the physical-space-based or modal-space-based steady-state dynamic analysis. In contrast, Abaqus supports applying the diffuse incident field in the physical-space-based but not the modal-space-based steady-state dynamic analysis.

10.3 Acoustic Analysis in Abaqus

10.3.1 Acoustic Materials

In Abaqus, the "*ACOUSTIC MEDIUM" statement keyword [70] defines acoustic material properties. Users can set the real bulk modulus, the complex bulk modulus, the complex density, the cavitation pressure limit, the options of the porous model, and the volumetric drag. In addition, the complex bulk modulus and the complex density can be frequency-dependent parameters. The cavitation pressure limit and the real bulk modulus can be physical-filed-dependent (temperature, air humidity, water salinity) parameters. Then, the volumetric drag can be a frequency-dependent and physical-filed-dependent parameter.

10.3.2 Boundary Conditions

The default acoustic boundary is rigid. For breaking the default, a simple manipulation is to set a static pressure to the acoustic boundary, such as setting a zero pressure to let the boundary be a free surface. While considering the dynamic performance, Abaqus allows users to set acoustic boundaries with a reactive boundary condition, a radiating (or non-reflecting) boundary condition, a fluid-structure interface boundary condition, and mix a fluid-structure interface with a reactive or a radiating boundary condition. In these boundary conditions, the radiating boundary is a specific case of the reactive boundary, and both boundary properties are controlled by the acoustic impedance (or the admittance). Abaqus supplies many adjustment choices for modifying the properties of the radiating and reactive boundary conditions. By the way, the arguments in this paragraph are based on [41].

10.3.3 Loading Method in Acoustic Perturbation Analysis

In Abaqus, acoustic perturbations are small motivations in acoustic fields. The analysis of acoustic perturbations mainly concerns the study of acoustic wave propagations. In general, direct acoustic loading mainly indicates the acoustic waves come from the exterior acoustic field and propagate into the interior acoustic field. In this situation, two loading types can externally generate acoustic waves. One is the concentrated pressure conjugate on specific acoustic boundaries; the other is external acoustic sources such as a spherical source, a planar source, and a diffuse incident field. And Abaqus can analyse acoustic models with high-speed underlying flow conditions. Since this paper does not concern the underlying flow of the acoustic field, this topic will not stretch in the following elaborating sections. By the way, the arguments in this paragraph are based on [41].

10.4 Acoustic Analysis in Actran

10.4.1 Acoustic Materials

In Actran, three kinds of material models [71] are classified as acoustic materials: the “FLUID” material model, the “WATER” material model, and the “PERFECT_GAS” material model. When writing an input file, these material models are referenced by the “ACOUSTIC” component [72]. Between these three material models, the “FLUID” is commonly used. It is the standard material model for defining viscous and non-viscous acoustic mediums.

10.4.2 Boundary Conditions

In Actran, boundary conditions of an acoustic domain are well-developed mathematical models [73] for typical application scenarios. They have relatively laconic formats and do not let users change their intrinsic characteristics. In other words, the application of these well-developed boundary conditions has restrictions. For example, some acoustic boundary conditions can be applied in the physical-space-based dynamic analysis but cannot be applied in modal-space-based dynamic analysis. This case will not happen in Abaqus. Since the well-developed mathematical models of acoustic boundary conditions are enveloped in Actran, this paper will not elaborate more on the corresponding background theories. For more information about these well-developed acoustic boundary models in Actran, please see the corresponding introductions in the Actran user guide.

10.4.3 Loading Method in Acoustic Perturbation Analysis

In Actran, acoustic loadings may be through specific boundary conditions or exterior acoustic sources [73]. In a physical-space-based dynamic analysis, the loading boundary conditions could be the “ACCELERATION” and the “VELOCITY” statements. At the same time, the exterior acoustic sources could be the “ROTATING_SOURCE” statement, the “SOURCE” statement, and the “CLOUD_SOURCE” statement. In a modal-space-based dynamic analysis, the loading boundary conditions could be the “FLUID_ACCELERATION” statement, the “FLUID_DISPLACEMENT” statement, and the “FLUID_VELOCITY” statement. At the same time, the exterior acoustic source could be the “SOURCE”. In addition, Actran can simulate acoustic wave propagation with an underlining flow. The mean flow velocity can be set in the “ACOUSTIC” statement [72]. Since this paper does not concern the underlying flow of the acoustic field, this topic will not stretch in the following elaborating sections.

10.5 Acoustic Analysis in Nastran

10.5.1 Acoustic Materials

In Nastran, the mathematical model of acoustic and structural materials is analogous. The acoustic material defining entry is “MAT10”. Then, the acoustic material will be referenced by the element property definition entry “PSOLID” by adding an extra declaration. No extra declaration is needed if the “PSOLID” references solid material. By the way, the arguments in this paragraph are based on [74].

10.5.2 Boundary Conditions

In Nastran, the default state of the acoustic boundaries is rigid. At this state, the pressure gradient at the orthogonal direction is zero. If any structural element attaches to the acoustic boundary, the acoustic boundary default state is broken. The structure acceleration will be transformed into the acoustic pressure gradient. As the media between fluid and structure, the fluid-structure interface is the only specific boundary condition that can be directly applied to the acoustic boundary. And this boundary condition is set by the interface-defining entry “ACMODL”. And Nastran supports setting acoustic boundaries by adding specific elements, such as absorber and barrier elements. The boundary consisting of absorber elements is set by combining the element-defining entry “CHACAB” and the property-defining entry “PACABS”. Similarly, the boundary consisting of barrier elements is set by combining the element-defining entry “CHACABR” and the property-defining entry “PACABR”. By the way, the arguments in this paragraph are based on [34].

10.5.3 Loading Method for Acoustic Perturbation Analysis

There are two ways to load acoustic elements in Nastran. One is the enforced acoustic pressure at grid points, which can be a constant, frequency-dependent, or time-dependent parameter. The other is the simple acoustic source characterised by a volumetric flow rate and located at an acoustic grid point. Since the acoustic equations in Nastran are based on small motion theory with negligible convective momentum terms and locally linear pressure-density relationship, Nastran cannot simulate acoustic wave propagation with an underlining flow. By the way, the arguments in this paragraph are based on [34].

10.6 Chapter Conclusions

The acoustic mathematical models in FEM software are developed from fundamental theories and stretched to specific applications. Transforming a mathematical definition to a finite element model needs a discretisation process. After that, the assigned problems can be solved by numerical simulations. For specific applications, the FEM software must be able to efficiently build the acoustic models, such as the mechanical properties of the acoustic medium, the diversified acoustic boundary conditions, the proper acoustic loading formulations, etc.

Commercial FEM software, such as Abaqus, Actran, and Nastran, have analogous discretisation methods (Galerkin method). But they have different mathematical models to define the acoustic medium, the acoustic boundary conditions, the acoustic loadings, and the sound wave formulations. Thus, they have different capabilities to cover acoustic problems. Specifically, Actran is an expert in acoustic analysis, and Abaqus is compatible, but Nastran is not so competent in this field.

The topics of acoustic mathematical models include four aspects:

- the governing equations of the acoustic and structural field,
- the acoustic boundary condition definitions,
- the discretisation of the acoustic-structural model,
- the loading method to acoustic domains.

Different FEM software has different methodologies to define the acoustic mathematical models, but the background fundamental theory is analogous. Abaqus gives the most elaborate methodologies introduction between the three software. Thus, a major part of the following content comes from the acoustic theory methodologies in Abaqus (see chapter **11**). Then, the introductions of acoustic theory methodologies in Actran (see chapter **12**) and Nastran (see chapter **13**) are supplementary explanations.

Chapter 11

Acoustics in Abaqus

It is necessary to note that from sections **11.1** to **11.3**, the theory reference is [75]. Then, from sections **11.4** to **11.5**, the theory reference is [76]. Based on references [75] and [76], this paper quotes the formula and symbols; then reorganises the logic and expressions.

Since this paper concerns the steady-state dynamic analysis of the acoustic-structural model, the main forms of mathematical models in the following elaborations are harmonic functions. Such as the harmonic fluid pressure,

$$p_f(\bar{\boldsymbol{x}},t)=\widetilde{P}_f(\bar{\boldsymbol{x}})e^{i\omega t} \tag{10.1}$$

The harmonic fluid displacement,

$$u_f(\bar{\boldsymbol{x}},t)=\widetilde{U}_f(\bar{\boldsymbol{x}})e^{i\omega t} \tag{10.2}$$

The harmonic structure displacement,

$$u_s(\bar{\boldsymbol{x}},t)=\widetilde{U}_s(\bar{\boldsymbol{x}})e^{i\omega t} \tag{10.3}$$

In the upper equations, $\widetilde{P}_f(\bar{\boldsymbol{x}})$, $\widetilde{U}_f(\bar{\boldsymbol{x}})$, and $\widetilde{U}_s(\bar{\boldsymbol{x}})$ are the corresponding complex amplitudes; $\bar{\boldsymbol{x}}=[x,y,z]^T$ is the position vector.

11.1 The Acoustic-Field Governing Equation

Abaqus uses the modified Euler's equation and constitutive equation to deduce the acoustic-field governing equation. Then, Abaqus introduces the structure-field governing equation to match the fluid-structure coupling model. Thus, the solutions of the vibro-acoustic analysis can be calculated by combining the acoustic-field and structural-field governing equations.

11.1.1 Euler's Equation from Time Domain to Frequency Domain

Abaqus extends Euler's equation by adding the volumetric drag. The modified Euler's equation in the time domain can be expressed as:

$$\frac{\partial p_f(\bar{\boldsymbol{x}},t)}{\partial \bar{\boldsymbol{x}}} + \gamma(\bar{\boldsymbol{x}},\theta_i)\frac{\partial u_f(\bar{\boldsymbol{x}},t)}{\partial t} + \rho_f(\bar{\boldsymbol{x}},\theta_i)\frac{\partial^2 u_f(\bar{\boldsymbol{x}},t)}{\partial t^2} = 0 \tag{10.4}$$

In the upper equation, $\gamma(\bar{\boldsymbol{x}},\theta_i)$ is the volumetric drag, and $\rho_f(\bar{\boldsymbol{x}},\theta_i)$ is the fluid mass density. In these two parameters, θ_i is an independent physical field variable such as temperature, air humidity, water salinity, etc. Without considering the physical-field dependency and the spatial distribution, the volumetric drag γ and the fluid mass density ρ_f are set as constant parameters in the following contents. Then, in the frequency domain, the modified Euler's equation deduced from the harmonic function can be expressed as:

$$\frac{\partial \widetilde{P}_f(\bar{\boldsymbol{x}})}{\partial \bar{\boldsymbol{x}}} - \omega^2 \widetilde{\rho}_f \widetilde{U}_f(\bar{\boldsymbol{x}}) = 0,\ \widetilde{\rho}_f = \frac{\gamma}{i\omega} + \rho_f \tag{10.5}$$

In the upper equation, $\widetilde{\rho}_f$ is called the complex fluid mass density.

11.1.2 The Constitutive Fluid Equation from Time Domain to Frequency Domain

In Abaqus, the constitutive behaviour of fluid is assumed to be inviscid, linear, and compressible. The constitutive fluid equation can be expressed as:

$$p_f(\bar{\boldsymbol{x}},t) = -K_f(\bar{\boldsymbol{x}},\theta_i)\frac{\partial u_f(\bar{\boldsymbol{x}},t)}{\partial \bar{\boldsymbol{x}}} \tag{10.6}$$

In the upper equation, $K_f(\bar{\boldsymbol{x}},\theta_i)$ is the bulk modulus of fluid, which is assumed to be a constant parameter in the following elaboration. This paper does not concern the varieties of the bulk modulus. Then, the constitutive fluid equation deduced from harmonic function can be expressed as:

$$-\frac{1}{K_f}\widetilde{P}_f(\bar{\boldsymbol{x}}) = \frac{\partial \widetilde{U}_f(\bar{\boldsymbol{x}})}{\partial \bar{\boldsymbol{x}}} \tag{10.7}$$

11.1.3 The Simplified Expression of The Acoustic-Field Governing Equation in Abaqus

Bringing the constitutive equation into Euler's equation can introduce the updated constitutive equation. The updated constitutive equation can be expressed as:

$$-\omega^2 \frac{1}{K_f} \tilde{P}_f(\bar{\boldsymbol{x}}) - \frac{\partial}{\partial \bar{\boldsymbol{x}}} \cdot \frac{1}{\tilde{\rho}_f} \frac{\partial \tilde{P}_f(\bar{\boldsymbol{x}})}{\partial \bar{\boldsymbol{x}}} = 0 \tag{10.8}$$

Based on the updated constitutive equation, the acoustic-field governing equation can be obtained by executing a volumetric integration with green's theorem. Then, through a partitioning operation, the governing equation is automatically subdivided into the inner part and the boundary part,

$$\int_{V_f} \left[-\delta p \frac{\omega^2}{K_f} \tilde{P}_f + \frac{1}{\tilde{\rho}_f} \frac{\partial \delta p}{\partial \bar{\boldsymbol{x}}} \cdot \frac{\partial \tilde{P}_f(\bar{\boldsymbol{x}})}{\partial \bar{\boldsymbol{x}}} \right] + \int_S \delta p \left[\frac{1}{\tilde{\rho}_f} \frac{\partial \tilde{P}_f(\bar{\boldsymbol{x}})}{\partial \bar{\boldsymbol{x}}} \cdot \bar{\boldsymbol{n}}^- \right] dS = 0 \tag{10.9}$$

In the upper equation, the functions in the volumetric integration represent the properties of the acoustic domain inner part; the function in the surface integration represents the properties of the acoustic domain boundary part, and $\bar{\boldsymbol{n}}^-$ is an inward-direction vector normal to the boundary. Then, it is necessary to introduce boundary traction to elaborate on the acoustic boundary conditions. The boundary traction can be expressed as the fluid mechanical property:

$$\tilde{T}(\bar{\boldsymbol{x}}) = -\frac{1}{\tilde{\rho}_f} \frac{\partial \tilde{P}_f(\bar{\boldsymbol{x}})}{\partial \bar{\boldsymbol{x}}} \cdot \bar{\boldsymbol{n}}^- = -\omega^2 \tilde{U}_f(\bar{\boldsymbol{x}}) \cdot \bar{\boldsymbol{n}}^- \tag{10.10}$$

The boundary traction expressed in fluid mechanical property is a bridge, which can transform different boundary conditions into an equivalent boundary effect. In real practice, boundary tractions may present in different forms because the exact meaning of boundary tractions are dependent on the boundary conditions. These boundary conditions include the "Enforced Boundary Condition" (see subsection **11.2.1**), the "Reactive Boundary Condition" (see subsection **11.2.2**), the "Non-Reflecting Boundary Condition" (see subsection **11.2.3**), the "Acoustic-Structural Interface" (see subsection **11.2.4**), and the "The Mixed Boundary Condition" (see subsection **11.2.5**).

11.2 Specific Boundary Conditions

11.2.1 The Enforced Boundary Condition

The enforced boundary condition S_{ft} can be used to model the oscillation of a rigid body exciting a fluid domain. In a steady-state dynamic process, the boundary traction is an inward acceleration normal to the fluid boundary. The boundary traction of an enforced boundary condition can be expressed as:

$$\widetilde{T}_{\text{ft}}(\bar{\boldsymbol{x}}) = T_0 = a_{in} \tag{10.11}$$

In the upper equation, a_{in} is the inward acceleration normal to the fluid boundary.

11.2.2 The Reactive Boundary Condition

The reactive boundary condition S_{fr} can absorb and reflect acoustic waves. The property of this boundary condition is equivalent to acoustic admittance. Like spring and dashpot connected in series between the acoustic medium and the rigid wall. Thus, the properties of reactive boundary conditions can be expressed as:

$$-\dot{u}_f(\bar{\boldsymbol{x}},t)\cdot\bar{\boldsymbol{n}}^- = \frac{1}{k_1}\dot{p}_f(\bar{\boldsymbol{x}},t) + \frac{1}{c_1}p_f(\bar{\boldsymbol{x}},t) \tag{10.12}$$

In the upper equation, $\dot{u}_f(\bar{\boldsymbol{x}},t)\cdot\bar{\boldsymbol{n}}^-$ is the inward fluid velocity normal to the fluid boundary, $\dot{p}_f(\bar{\boldsymbol{x}},t)$ is the first-order time derivative of the acoustic pressure, $1/k_1$ is a parameter between acoustic pressure and acceleration, and $1/c_1$ is a parameter between acoustic pressure and velocity. Now, making a further time derivative to the upper equation, the traction of a reactive boundary condition in a steady-state dynamic analysis can be expressed as:

$$\widetilde{T}_{\text{fr}}(\bar{\boldsymbol{x}}) = -\left(-\frac{\omega^2}{k_1} + \frac{i\omega}{c_1}\right)\widetilde{P}_f(\bar{\boldsymbol{x}}) \tag{10.13}$$

11.2.3 The Non-Reflecting (Radiating) Boundary Condition

Many studies are concerned that acoustic waves propagate to the faraway domain from a vibrating structure. But the problem is that it is impossible to extend the acoustic medium infinitely away from the vibrating structure. Based on the Abaqus reference guide, there are two ways to solve this problem; one method uses the infinite acoustic element, and the other uses a non-reflecting acoustic boundary to divide the inner and infinite domains. When using the non-reflecting boundary method, the inner domain is filled with finite elements. And it is better to set the thickness of the finite-element layer to more than one-third of the acoustic wavelength for sufficient accuracy. The non-reflecting boundary condition can also be called the radiating boundary condition S_{fi}. It permits incoming acoustic waves to pass through the

boundary without reflection. The traction of the non-reflecting boundary condition has the same form as the traction of the reactive boundary condition,

$$\widetilde{T}_{\text{fi}}(\bar{\boldsymbol{x}}) = -\left(\frac{i\omega}{c_1} + \frac{1}{a_1}\right)\widetilde{P}_f(\bar{\boldsymbol{x}}) \tag{10.14}$$

In the upper equation, $1/a_1$ is a parameter between acoustic pressure and displacement.

11.2.4 The Acoustic-Structural Interface

On the acoustic-structural interface S_{fs}, the fluid and structure displacements normal to the interface are equivalent,

$$u_f(\bar{\boldsymbol{x}},t)\cdot\bar{\boldsymbol{n}}^- = u_s(\bar{\boldsymbol{x}},t)\cdot\bar{\boldsymbol{n}}^- \tag{10.15}$$

Thus, based on the traction-acceleration relationship in a steady-state dynamic analysis, the traction of the fluid-structure interface regarding the structure displacement can be expressed as:

$$\widetilde{T}_{\text{fs}}(\bar{\boldsymbol{x}}) = -\omega^2\widetilde{U}_s(\bar{\boldsymbol{x}})\cdot\bar{\boldsymbol{n}}^- \tag{10.16}$$

11.2.5 The Mixed Boundary Condition

The mixed boundary condition S_{frs} permits relative motions between the structural and acoustic boundaries, and the structural boundary connects the acoustic boundary with a spring-dashpot system. The characteristic of this kind of boundary can be expressed as follows:

$$(\dot{u}_s(\bar{\boldsymbol{x}},t) - \dot{u}_f(\bar{\boldsymbol{x}},t))\cdot\bar{\boldsymbol{n}}^- = \frac{1}{k_1}\dot{p}_f(\bar{\boldsymbol{x}},t) + \frac{1}{c_1}p_f(\bar{\boldsymbol{x}},t) \tag{10.17}$$

Thus, based on the traction-acceleration relationship and the reactive boundary condition, the traction of the mixed boundary condition in steady-state dynamic analysis can be expressed as:

$$\widetilde{T}_{\text{frs}}(\bar{\boldsymbol{x}}) = -\omega^2\widetilde{U}_s(\bar{\boldsymbol{x}})\cdot\bar{\boldsymbol{n}}^- - \frac{i\omega}{c_1}\widetilde{P}_f(\bar{\boldsymbol{x}}) + \frac{\omega^2}{k_1}\widetilde{P}_f(\bar{\boldsymbol{x}}) \tag{10.18}$$

11.3 The Discretisation of Acoustic and Structural Models

A vibro-acoustic analysis concerns the fluid-structure interactions on boundaries and in inner domains. The acoustic-field and structural-field governing equations with specific boundary conditions are keys to analysing the fluid-structure system. They will be introduced as mathematical models in subsection **11.3.1** and subsection **11.3.5**. It is known that from the mathematical model to the numerical solutions, a discretisation process is needed. After the discretisation, the acoustic-field and structural-field governing equations become matrix equations that software solvers can directly process. Then, the discretised acoustic-structural system consists of a huge amount of DOFs (degrees of freedom), including the fluid inner domain DOFs, the fluid boundary DOFs, the solid inner domain DOFs, and the solid boundary DOFs. In the Cartesian coordinate system, it is important to note that one fluid point corresponds to one DOF, which is the acoustic pressure. In comparison, one solid point corresponds to six DOFs. Three DOFs are displacements; another three DOFs are rotations.

The interpolation functions of the acoustic element are H^P and H^Q. The superscript P indicates the interior DOFs, while Q indicates the boundary DOFs. Similarly, the interpolation functions of the structural element are $\boldsymbol{N}^N$ and $\boldsymbol{N}^M$. As for the strain tensor, the interpolation function is $\boldsymbol{\beta}^N$. Then, for the stress tensor of residual stress caused by boundary deformation, the interpolation function is $\partial\boldsymbol{\beta}^N/\partial\{\tilde{U}_s\}^M$. The superscript N indicates the boundary DOFs, while M indicates the inner DOFs.

11.3.1 The Acoustic-Field Governing Equation with Specific Boundary Conditions

The inverse of the complex fluid mass density can be split into a real and an imaginary part,

$$\frac{1}{\tilde{\rho}_f} = \frac{\rho_f}{{\rho_f}^2 + \gamma^2/\omega^2} + i\frac{\gamma/\omega}{{\rho_f}^2 + \gamma^2/\omega^2} \tag{10.19}$$

For introducing the specific boundary conditions into the original acoustic-field governing equation, the updated governing equation in a steady-state dynamic analysis can be expressed as:

$$\begin{aligned}
&\int_{V_f} -\omega^2 \delta p \left(\frac{1}{K_f} \widetilde{P}_f(\bar{\boldsymbol{x}}) \right) dV \\
&+ \int_{V_f} \frac{\rho_f}{{\rho_f}^2 + \gamma^2/\omega^2} \frac{\partial \delta p}{\partial \bar{\boldsymbol{x}}} \cdot \frac{\partial \widetilde{P}_f(\bar{\boldsymbol{x}})}{\partial \bar{\boldsymbol{x}}} dV \\
&+ \int_{V_f} (i\omega) \frac{\gamma/\omega^2}{{\rho_f}^2 + \gamma^2/\omega^2} \frac{\partial \delta p}{\partial \bar{\boldsymbol{x}}} \cdot \frac{\partial \widetilde{P}_f(\bar{\boldsymbol{x}})}{\partial \bar{\boldsymbol{x}}} dV \\
&- \int_{S_{\mathrm{ft}}} \delta p T_0 \, dS \\
&+ \int_{S_{\mathrm{fs}} \cup S_{\mathrm{frs}}} \omega^2 \delta p \widetilde{U}_s(\bar{\boldsymbol{x}}) \cdot \bar{\boldsymbol{n}}^- dS \\
&+ \int_{S_{\mathrm{fr}} \cup S_{\mathrm{fi}} \cup S_{\mathrm{frs}}} \delta p \left(\frac{i\omega}{c_1} \widetilde{P}_f(\bar{\boldsymbol{x}}) - \frac{\omega^2}{k_1} \widetilde{P}_f(\bar{\boldsymbol{x}}) \right) dS = 0
\end{aligned} \tag{10.20}$$

11.3.2 The Interpolation Functions of The Acoustic-Field Governing Equation

In the acoustic-field governing equation, the δp indicates the virtual acoustic pressure of any fluid point at the interior section, and its interpolation function can be expressed as:

$$\delta p = H^P \{\delta p\}^P \tag{10.21}$$

The $\widetilde{P}_f(\bar{\boldsymbol{x}})$ indicates the complex acoustic pressure of any fluid point at the boundary section, and its interpolation function can be expressed as:

$$\widetilde{P}_f(\bar{\boldsymbol{x}}) = H^Q \left\{ \widetilde{P}_f \right\}^Q \tag{10.22}$$

The $\widetilde{U}_s(\bar{\boldsymbol{x}})$ indicates the complex solid displacement of any solid point at the inner section, and its interpolation function can be expressed as:

$$\widetilde{U}_s(\bar{\boldsymbol{x}}) = \boldsymbol{N}^M \left\{ \widetilde{U}_s \right\}^M \tag{10.23}$$

T_0 indicates the predefined acceleration of any fluid point at the boundary section, and it does not need interpolation.

11.3.3 The Discretisation of an Acoustic Domain

Galerkin's principle is the operator that transforms the acoustic-domain interpolation functions into finite-element matrices. The matrix corresponding to the acoustic inertia of the inner acoustic domain:

$$[M_{\mathrm{f}}]^{PQ} = \int_{V_f} \frac{1}{K_f} H^P H^Q dV \tag{10.24}$$

The matrix corresponding to the acoustic inertia of the reactive and non-reflecting boundary:

$$[M_{\mathrm{fr}\cup\mathrm{fi}}]^{PQ} = \int_{S_{\mathrm{fr}}\cup S_{\mathrm{fi}}} \frac{1}{k_1} H^P H^Q dS \tag{10.25}$$

The matrix corresponding to acoustic damping due to the volumetric drag of the inner acoustic domain:

$$[C_{\mathrm{f}}]^{PQ} = \int_{V_f} \frac{\gamma/\omega^2}{{\rho_f}^2 + \gamma^2/\omega^2} \frac{\partial H^P}{\partial \bar{\boldsymbol{x}}} \cdot \frac{\partial H^Q}{\partial \bar{\boldsymbol{x}}} dV \tag{10.26}$$

The matrix corresponding to acoustic damping of the reactive and non-reflecting boundary:

$$[C_{\mathrm{fr}\cup\mathrm{fi}}]^{PQ} = \int_{S_{\mathrm{fr}}\cup S_{\mathrm{fi}}} \frac{1}{c_1} H^P H^Q dS \tag{10.27}$$

The matrix corresponding to global structural damping of fluid material:

$$\left[C_{\mathrm{f}(\eta_f)}\right]^{PQ} = \eta_f [K_{\mathrm{f}}]^{PQ} \tag{10.28}$$

The matrix corresponding to the acoustic elasticity of the inner acoustic domain:

$$[K_{\mathrm{f}}]^{PQ} = \int_{V_f} \frac{\rho_f}{{\rho_f}^2 + \gamma^2/\omega^2} \frac{\partial H^P}{\partial \bar{\boldsymbol{x}}} \cdot \frac{\partial H^Q}{\partial \bar{\boldsymbol{x}}} dV \tag{10.29}$$

The interface matrix that connects the structural boundary to the inner acoustic domain:

$$[S_{\mathrm{fs}}]^{PM} = \int_{S_{\mathrm{fs}}} H^P \bar{\boldsymbol{n}}^- \cdot \bar{\boldsymbol{N}}^M dS \tag{10.30}$$

The loading vector of the concentrated pressure-conjugate load on the boundary of the acoustic domain:

$$\{\tilde{p}_f\}^P = \int_{S_{\mathrm{ft}}} H^P T_0 dS \tag{10.31}$$

11.3.4 The Acoustic-Field Matrix Equation

The acoustic-field matrix equation can be written as:

$$\begin{aligned} &-\omega^2\left[[M_{\mathrm{f}}]^{PQ}+[M_{\mathrm{fr}\cup\mathrm{fi}}]^{PQ}\right]\left\{\widetilde{P}_f\right\}^Q \\ &+\left[i\omega\left[[C_{\mathrm{f}}]^{PQ}+[C_{\mathrm{fr}\cup\mathrm{fi}}]^{PQ}\right]+i\left[C_{\mathrm{f}(\eta_f)}\right]^{PQ}\right]\left\{\widetilde{P}_f\right\}^Q \\ &+[K_{\mathrm{f}}]^{PQ}\left\{\widetilde{P}_f\right\}^Q \\ &=-\omega^2[S_{\mathrm{fs}}]^{PM}\left\{\widetilde{U}_s\right\}^M+\left\{\widetilde{p}_f\right\}^P \end{aligned} \tag{10.32}$$

11.3.5 The Structural-Field Governing Equation with Specific Boundary Conditions

The structural-field governing equation with boundary traction in a steady-state dynamic analysis can be expressed as:

$$\begin{aligned} &\int_{V_S}-\omega^2\rho_s\delta\boldsymbol{u}_S\cdot\widetilde{U}_s(\bar{\boldsymbol{x}})\,dV \\ &+\int_{V_S}i\omega\alpha_c\rho_s\delta\boldsymbol{u}_S\cdot\widetilde{U}_s(\bar{\boldsymbol{x}})\,dV \\ &+\int_{V_S}\delta\boldsymbol{\varepsilon}:(\boldsymbol{\sigma}_0+\boldsymbol{D}^{el}:(\Delta\boldsymbol{\varepsilon}+i\eta\Delta\boldsymbol{\varepsilon}+i\omega\beta_c\Delta\dot{\boldsymbol{\varepsilon}}))\,dV \\ &+\int_{S_{\mathrm{fs}}}\delta\boldsymbol{u}_S\cdot\bar{\boldsymbol{n}}^-\widetilde{P}_f(\bar{\boldsymbol{x}})\,dS \\ &-\int_{S_{\mathrm{t}}}\delta\boldsymbol{u}_S\cdot t(\bar{\boldsymbol{x}})\,dS=0 \end{aligned} \tag{10.33}$$

In the upper equation, ρ_s is the structure mass density; $\delta\boldsymbol{u}_S$ is the virtual displacement vector of the structure; α_c is the mass-proportional damping factor; β_c is the stiffness proportional damping factor; $t(\bar{\boldsymbol{x}})$ is the surface traction applied to the structure; $\delta\boldsymbol{\varepsilon}$ is the virtual strain tensor that is compatible with $\delta\boldsymbol{u}_S$; $\Delta\boldsymbol{\varepsilon}$ is the strain tensor variation; $\Delta\dot{\boldsymbol{\varepsilon}}$ is the first-order time derivative of the strain tensor variation; $\boldsymbol{\sigma}_0$ is the stress tensor in the base state; $\boldsymbol{D}^{el}$ is the material elasticity matrix; η is the structural damping factor.

11.3.6 The Interpolation Functions of The Structural-Field Governing Equation

In the structural-field governing equation, the $\delta \boldsymbol{u}_S$ indicates the virtual displacement of any solid point at the interior section, and its interpolation function can be expressed as:

$$\delta \boldsymbol{u}_S = \boldsymbol{N}^N \{\delta u_S\}^N \tag{10.34}$$

The $\delta \boldsymbol{\varepsilon}$ indicates the virtual strain tensor of any solid point at the interior section, and its interpolation function can be expressed as:

$$\delta \boldsymbol{\varepsilon} = \boldsymbol{\beta}^N \{\delta u_S\}^N \tag{10.35}$$

The $\Delta \boldsymbol{\varepsilon}$ indicates the strain-tensor variation of any solid point at the boundary section, and its interpolation function can be expressed as:

$$\Delta \boldsymbol{\varepsilon} = \boldsymbol{\beta}^M \left\{\Delta \widetilde{U}_s\right\}^M \tag{10.36}$$

The $\widetilde{P}_f(\bar{\boldsymbol{x}})$ indicates the complex acoustic pressure of any fluid point at the boundary section, and its interpolation function can be expressed as:

$$\widetilde{P}_f(\bar{\boldsymbol{x}}) = H^Q \left\{\widetilde{P}_f\right\}^Q \tag{10.37}$$

The $t(\bar{\boldsymbol{x}})$ indicates the predefined boundary traction of any solid point at the boundary section, and it does not need interpolation.

11.3.7 The Discretisation of a Structural Domain

Galerkin's principle is the operator that transforms the structural-domain interpolation functions into finite-element matrices. The matrix corresponding to the structural inertia of the inner acoustic domain:

$$[M_s]^{NM} = \int_{V_S} \rho_s \boldsymbol{N}^N \cdot \boldsymbol{N}^M dV \tag{10.38}$$

The matrix corresponding to mass-proportional Rayleigh viscous damping:

$$[C_{s(\mathrm{m})}]^{NM} = \int_{V_S} \alpha_c \rho_s \boldsymbol{N}^N \cdot \boldsymbol{N}^M dV \tag{10.39}$$

The matrix corresponding to elasticity-proportional Rayleigh viscous damping:

$$[C_{s(\mathrm{k})}]^{NM} = \int_{V_S} \beta_c \boldsymbol{\beta}^N : \boldsymbol{D}^{el} : \boldsymbol{\beta}^M dV \tag{10.40}$$

The matrix corresponding to global-level and material-level structural damping:

$$\left[C_{s(\eta_S)}\right]^{NM} = \int_{V_S} \eta_s \boldsymbol{\beta}^N : \boldsymbol{D}^{el} : \boldsymbol{\beta}^M \, dV + \eta_g \left[K_s\right]^{NM} \tag{10.41}$$

The matrix corresponding to the structural elasticity of the inner structural domain:

$$\left[K_s\right]^{NM} = \int_{V_S} \left[\frac{\partial \boldsymbol{\beta}^N}{\partial \left\{\widetilde{U}_s\right\}^M} : \boldsymbol{\sigma}_0 + \boldsymbol{\beta}^N : \boldsymbol{D}^{el} : \boldsymbol{\beta}^M \right] dV \tag{10.42}$$

And there is a term that represents the stress stiffening by residual stress:

$$\int_{V_S} \frac{\partial \boldsymbol{\beta}^N}{\partial \left\{\widetilde{U}_s\right\}^M} : \boldsymbol{\sigma}_0 \, dV \tag{10.43}$$

A term represents the loading stress from the boundary to the inner acoustic domain of a structure,

$$\int_{V_S} \boldsymbol{\beta}^N : \boldsymbol{D}^{el} : \boldsymbol{\beta}^M \, dV \tag{10.44}$$

The interface matrix that connects the acoustic boundary to the inner structural domain:

$$\left[S_{\mathrm{fs}}\right]^{NQ} = \int_{S_{\mathrm{fs}}} \boldsymbol{N}^N \cdot \bar{\boldsymbol{n}}^- H^Q \, dS \tag{10.45}$$

The loading vector of direct traction on the structural boundary:

$$\left\{\widetilde{p}_s\right\}^N = \int_{S_{\mathrm{t}}} \boldsymbol{N}^N \cdot t(\bar{\boldsymbol{x}}) \, dS \tag{10.46}$$

11.3.8 The Structural-Field Matrix Equation

The structural-field matrix equation can be written as:

$$\begin{aligned} &-\omega^2 \left[M_{\mathrm{s}}\right]^{NM} \left\{\widetilde{U}_s\right\}^M \\ &+ \left[i\omega \left[\left[C_{\mathrm{s(m)}}\right]^{NM} + \left[C_{\mathrm{s(k)}}\right]^{NM} \right] + i \left[C_{\mathrm{s}(\eta_S)}\right]^{NM} \right] \left\{\widetilde{U}_s\right\}^M \\ &+ \left[K_{\mathrm{s}}\right]^{NM} \left\{\widetilde{U}_s\right\}^M \\ &= -\left[S_{\mathrm{fs}}\right]^{NQ} \left\{\widetilde{P}_f\right\}^Q + \left\{\widetilde{p}_s\right\}^N \end{aligned} \tag{10.47}$$

11.4 The Scattered Wave Formulation of a Fluid-Structure System

In Abaqus, acoustic loads can be applied in transient or steady-state dynamic analysis. There are two acoustic loading types:

- concentrated pressure-conjugate load on the acoustic boundary,
- acoustic sources from the external acoustic domain.

The effect of acoustic loading is the acoustic wave propagating into the inner acoustic domain and causing responses. Then, the total acoustic pressure at the loaded acoustic boundaries can be written as a superposition of an incident part and a scattered part:

$$\left\{\tilde{P}_f\right\}^Q = \left\{\tilde{P}_f\right\}_I^Q + \left\{\tilde{P}_f\right\}_S^Q \tag{10.48}$$

Similarly, the total traction force at the loaded structure boundary can be written as a superposition of an incident part and a scattered part:

$$-\omega^2\left\{\tilde{U}_s\right\}^M = \left\{\tilde{T}_s\right\}^M = \left\{\tilde{T}_s\right\}_I^M + \left\{\tilde{T}_s\right\}_S^M \tag{10.49}$$

11.4.1 Incoming Acoustic Waves from External Acoustic Sources

The external acoustic sources generate the acoustic wave that propagates around and enters the inner acoustic domain. And the waves that enter the inner acoustic domain are called the incident wave. When the inner acoustic domain consists of fluid and structure components, the incoming incident wave will pass through the fluid component and encounter the structure component. Then, part of the incident wave can pass through or be absorbed by the structure component, and the structure reflects the residual. The summation of the reflected waves through multi-time reflections is called the scattered wave. On the side with the structure component reflection, the acoustic pressure field is affected by the incident wave and the scattered wave.

For solving the acoustic-field problems of a fluid-structure system due to the incident wave and the scattered wave, Abaqus supplies two solving methods. One is the total wave formulation, and the other is the scattered wave formulation.

The total wave formulation treats the total-pressure response caused by the load on fluid boundaries, and the load, in this case, is analogous to the structural load on structure boundaries. This formulation can treat transient dynamic analyses but cannot treat steady-state dynamic analyses. Only the outside acoustic surfaces should be loaded when the total wave formulation is adopted, and the loading point must be located exterior to the fluid model.

In comparison, the scattered wave formulation assumes systems to be linear. Thus, in this case, the total wave can be seen as the superposition of the known incident wave and the unknown scattered wave. The fluid and structure boundaries on the fluid-structure interface

should be loaded when the scattered wave formulation is adopted. This formulation can treat transient dynamic analyses and steady-state dynamic analyses. Since this paper concerns the steady-state dynamic analysis, the following contents only elaborate on the scattered wave formulation and omit the total wave formulation.

11.4.2 The Scattered Wave Formulation of an Acoustic Domain

Consequently, the discretised acoustic governing equation with the known incident wave and structural traction can be expressed as:

$$\begin{aligned} &-\omega^2\left[[M_{\mathrm{f}}]^{PQ}+[M_{\mathrm{fr}\cup\mathrm{fi}}]^{PQ}\right]\left\{\widetilde{P}_f\right\}_I^Q \\ &+\left[i\omega\left[[C_{\mathrm{f}}]^{PQ}+[C_{\mathrm{fr}\cup\mathrm{fi}}]^{PQ}\right]+i[C_{\mathrm{f}(\eta)}]^{PQ}\right]\left\{\widetilde{P}_f\right\}_I^Q \\ &+[K_{\mathrm{f}}]^{PQ}\left\{\widetilde{P}_f\right\}_I^Q \\ &=[S_{\mathrm{fs}}]^{PM}\left\{\widetilde{T}_s\right\}_I^M \end{aligned} \tag{10.50}$$

Similarly, the discretised acoustic-field governing equation with the unknown scattered wave and structural traction can be expressed as:

$$\begin{aligned} &-\omega^2\left[[M_{\mathrm{f}}]^{PQ}+[M_{\mathrm{fr}\cup\mathrm{fi}}]^{PQ}\right]\left\{\widetilde{P}_f\right\}_S^Q \\ &+\left[i\omega\left[[C_{\mathrm{f}}]^{PQ}+[C_{\mathrm{fr}\cup\mathrm{fi}}]^{PQ}\right]+i[C_{\mathrm{f}(\eta)}]^{PQ}\right]\left\{\widetilde{P}_f\right\}_S^Q \\ &+[K_{\mathrm{f}}]^{PQ}\left\{\widetilde{P}_f\right\}_S^Q \\ &=[S_{\mathrm{fs}}]^{PM}\left\{\widetilde{T}_s\right\}_S^M+\left\{\widetilde{p}_f\right\}^P \end{aligned} \tag{10.51}$$

11.4.3 The Scattered Wave Formulation of a Structural Domain

Then, the discretised structural governing equation with the incident and scattered acoustic pressures on the interface can be expressed as:

$$\begin{aligned} &-\omega^2[M_{\mathrm{s}}]^{NM}\left\{\widetilde{U}_s\right\}^M \\ &+\left[i\omega\left[[C_{\mathrm{s(m)}}]^{NM}+[C_{\mathrm{s(k)}}]^{NM}\right]+i[C_{\mathrm{s(s)}}]^{NM}\right]\left\{\widetilde{U}_s\right\}^M \\ &+[K_{\mathrm{s}}]^{NM}\left\{\widetilde{U}_s\right\}^M \\ &=-[S_{\mathrm{fs}}]^{NQ}\left(\left\{\widetilde{P}_f\right\}_I^Q+\left\{\widetilde{P}_f\right\}_S^Q\right)+\{\widetilde{p}_s\}^N \end{aligned} \tag{10.52}$$

11.5 The Incident Wave Load

In Abaqus, users must set a source point with a standoff point to build the incident-wave loading system. For defining incident waves, several important parameters need to be considered. These important parameters are roughly divided into four aspects:

- The first is the direction of incident wave propagation. Abaqus defines it as the position vector from a source point to a standoff point.
- The second is the speed of acoustic wave propagation. Generally, it is defined as the sound speed in the air.
- The third is the spatial-decay function of the wave amplitude in sound wave propagation. This function depends on the source type. For example, planar waves maintain constant amplitude. In contrast, the amplitude of spherical waves is inversely proportional to the distance away from the source.
- The fourth is the time-history function in a transient analysis or the phase-shift function in a steady-state analysis.

Abaqus does not directly use the source point to define the incident wave but uses the standoff point. Thus, the settings of the upper four important parameters are based on the standoff point. In steady-state dynamic analyses, users define a real (in-phase) or an imaginary (out-of-phase) magnitude of the incident wave at the standoff point. At this point, waves have a zero-phase shift. There is a time delay from the standoff point to the loaded points of the wave propagation. Thus, in steady-state dynamic analysis, the incident wave at loaded points has a phase shift regarding the standoff point. The following contents elaborate on the phase-shift and spacial-decay functions of planar and spherical sources in a steady-state dynamic analysis.

11.5.1 The Incident Wave Function in Time Domains

The incident wave function at an acoustic-field point can be expressed as:

$$p_I(\bar{\boldsymbol{x}}_j, t) = p_t(\tau_j)\, p_{\bar{x}}(\bar{\boldsymbol{x}}_j) \tag{10.53}$$

In the upper equation, $p_I(\bar{\boldsymbol{x}}_j, t)$ is the expression of an incident wave function; $\bar{\boldsymbol{x}}_j = [x_j, y_j, z_j]^T$ is the position vector of a loaded point; t is the real-time history of a point; $p_t(\tau_j)$ is the time-history function of the incident wave; τ_j is the retarded time of the given point regarding the standoff point; $p_{\bar{x}}(\bar{\boldsymbol{x}}_j)$ is the spatial-decay function of the incident wave. Then, the retarded time can be expressed as:

$$\tau_j = t - (R_j - R_0)/c \tag{10.54}$$

In the upper equation, c is the sound speed in fluid; R_j is the distance between the source point and the loaded point; R_0 is the distance between the source point and the standoff point. Then, R_j of a spherical wave can be expressed as:

$$R_j = \| \bar{\boldsymbol{x}}_s - \bar{\boldsymbol{x}}_j \| \tag{10.55}$$

On the other hand, R_j of a planar wave can be expressed as:

$$R_j = \frac{|(\bar{\boldsymbol{x}}_j - \bar{\boldsymbol{x}}_s) \cdot (\bar{\boldsymbol{x}}_0 - \bar{\boldsymbol{x}}_s)|}{\| \bar{\boldsymbol{x}}_s - \bar{\boldsymbol{x}}_0 \|} \tag{10.56}$$

At the same time, R_0 can be expressed as:

$$R_0 = \| \bar{\boldsymbol{x}}_s - \bar{\boldsymbol{x}}_0 \| \tag{10.57}$$

In the upper equations, $\bar{\boldsymbol{x}}_s = [x_s, y_s, z_s]^T$ is the position vector of the source point; $\bar{\boldsymbol{x}}_0 = [x_0, y_0, z_0]^T$ is the position vector of the standoff point. Then, assuming the time-history function can be expressed as a harmonic function in steady-state dynamic analyses:

$$p_t(\tau_j) = \tilde{p}_t e^{i\omega\left(t - \frac{R_j - R_0}{c}\right)} = \tilde{p}_t e^{-i\omega\left(\frac{R_j - R_0}{c}\right)} e^{i\omega t} = \tilde{p}_t e^{-i\theta} e^{i\omega t} \tag{10.58}$$

In the upper equation, $\tilde{p}_t$ is the complex magnitude of the incident pressure at the standoff point; $\theta = \omega(R_j - R_0)/c$ is the quantity of phase shift due to the retarded time that incident waves propagate from the standoff point to the loaded point. Then, a general form of the spatial-decay function can be expressed as:

$$p_{\bar{x}}(\bar{\boldsymbol{x}}_j) = \left(\frac{R_0}{R_j}\right)^{\alpha(R_j, R_0)} \tag{10.59}$$

In the upper equation, the exponent can be written as:

$$\alpha(R_j, R_0) = \frac{(A+1) R_j}{C R_0 + (B+1) R_j} \tag{10.60}$$

In the expression of the exponent, A, B, and C are three dimensionless constants, and they are used to modify the wave properties. When A, B, and C are zero, the spatial-decay function describes the behaviour of the spherical wave propagation. In comparison, when A approaches minus one, the spatial-decay function describes the behaviour of the planar wave propagation. Thus, the spatial-decay function of the spherical wave can be expressed as:

$$p_{\bar{x}}(\bar{\boldsymbol{x}}_j) = \frac{R_0}{R_j} \tag{10.61}$$

And the spatial-decay function of the planar wave can be expressed as:

$$p_{\bar{x}}(\bar{\boldsymbol{x}}_j) = 1 \tag{10.62}$$

11.5.2 The Incident Wave Function in Frequency Domains

According to the upper deduction, the incident wave function in the frequency domain can be expressed as:

$$p_I(\bar{\boldsymbol{x}}_j,t)=\tilde{p}_t e^{-i\omega\left(\frac{R_j-R_0}{c}\right)}e^{i\omega t}p_{\bar{x}}(\bar{\boldsymbol{x}}_j) \tag{10.63}$$

Thus, the complex amplitude of a spherical incident wave can be written as:

$$\tilde{P}_{f,I}(\bar{\boldsymbol{x}}_j)=\tilde{p}_t e^{-i\omega\left(\frac{R_j-R_0}{c}\right)}\frac{R_0}{R_j} \tag{10.64}$$

And the complex amplitude of a planar incident wave can be written as:

$$\tilde{P}_{f,I}(\bar{\boldsymbol{x}}_j)=\tilde{p}_t e^{-i\omega\left(\frac{R_j-R_0}{c}\right)} \tag{10.65}$$

Chapter 12

Acoustics in Actran

It is necessary to note that this chapter's theory reference is [77]. Based on reference [77], this paper quotes the formula and symbols; then reorganises the logic and expressions. Then, we can learn that when treating the interaction between the acoustic source and boundary conditions, Actran has two mathematical routes, each with two varieties. The first one decomposes the total wave into the incident and the scattered waves, relying on the homogeneous Helmholtz Equation. The second one treats the total wave with the non-homogenous Helmholtz Equation. As for the two varieties, homogenous and non-homogenous Helmholtz Equations can be in a non-convective form for the acoustic analysis in a non-moving flow, or they can be in a convective form for the acoustic analysis in a moving flow. In steady-state dynamic analysis, the harmonic acoustic pressure can be expressed as:

$$p_f(\bar{\boldsymbol{x}},t) = \widetilde{P}_f(\bar{\boldsymbol{x}})e^{i\omega t} \tag{11.1}$$

12.1 The Homogeneous Helmholtz Equation

12.1.1 The Non-Convective Homogeneous Helmholtz Equation

Without considering the fluid flow at a given fluid point in a harmonic equation, the homogenous Helmholtz Equation in the non-convective form can be expressed as:

$$\nabla^2 \widetilde{P}_f(\bar{\boldsymbol{x}}) + k^2 \widetilde{P}_f(\bar{\boldsymbol{x}}) = 0 \tag{11.2}$$

One thing that needs to be noted is that the symbol Δ expressed in the Actran users guide is the same as the conventional form ∇^2. They represent the Laplacian operator. Then, the complex amplitude of the total fluid pressure can be decomposed into an incident term and a scattered term,

$$\widetilde{P}_f(\bar{\boldsymbol{x}}) = \widetilde{P}_{f,I}(\bar{\boldsymbol{x}}) + \widetilde{P}_{f,S}(\bar{\boldsymbol{x}}) \tag{11.3}$$

In the upper equation, $\widetilde{P}_{f,I}(\bar{\boldsymbol{x}})$ is the complex pressure amplitude of the incident wave; $\widetilde{P}_{f,S}(\bar{\boldsymbol{x}})$ is the complex pressure amplitude of the scattered wave; $\bar{\boldsymbol{x}} = [x, y, z]^T$ is the position vector of any given point. In convention, the incident wave is known. From a plane wave source, the incident wave in propagation has a constant pressure amplitude. While from other sources, the pressure must be the solution of the homogeneous Helmholtz Equation compensated with a Sommerfeld radiation condition,

$$\nabla^2 \widetilde{P}_{f,I}(\bar{\boldsymbol{x}}) + k^2 \widetilde{P}_{f,I}(\bar{\boldsymbol{x}}) = 0 \tag{11.4}$$

In contrast, the scattered wave is unknown. It is the solution of the homogeneous Helmholtz Equation considering the specific boundary conditions,

$$\nabla^2 \widetilde{P}_{f,S}(\bar{\boldsymbol{x}}) + k^2 \widetilde{P}_{f,S}(\bar{\boldsymbol{x}}) = 0 \tag{11.5}$$

12.1.2 The Convective Homogeneous Helmholtz Equation

In considering the fluid flow in steady-state dynamic analysis, the homogenous Helmholtz Equation in the convective form can be expressed as:

$$\nabla^2 \widetilde{P}_f(\bar{\boldsymbol{x}}) - \left(ik + \vec{M} \cdot \nabla\right)^2 \widetilde{P}_f(\bar{\boldsymbol{x}}) = 0 \tag{11.6}$$

In the upper equation, $\vec{M}$ is the uniform mean flow velocity vector divided by the sound speed. As for the solution of the convective form of the homogenous Helmholtz Equation, the processing logic is analogous to the logic in the non-convective homogenous Helmholtz Equation.

12.1.3 Expressions of Boundary Conditions in Actran

The Actran theory guide gives a brief description of the acoustic boundary conditions. And these boundary conditions are in the theoretical form. The theoretical boundary conditions include the Dirichlet boundary condition (boundary with given normal pressures)

$$\widetilde{P}_{f,S}(\bar{\boldsymbol{x}}) = \widetilde{P}_f - \widetilde{P}_{f,I}(\bar{\boldsymbol{x}}) \tag{11.7}$$

the Neumann boundary condition (boundary with given normal velocity)

$$\frac{\partial \widetilde{P}_{f,S}(\bar{\boldsymbol{x}})}{\partial \bar{\boldsymbol{n}}} = \frac{\partial \widetilde{P}_f(\bar{\boldsymbol{x}})}{\partial \bar{\boldsymbol{n}}} - \frac{\partial \widetilde{P}_{f,I}(\bar{\boldsymbol{x}})}{\partial \bar{\boldsymbol{n}}} = -ick\rho_f \widetilde{V}_f - \frac{\partial \widetilde{P}_{f,I}(\bar{\boldsymbol{x}})}{\partial \bar{\boldsymbol{n}}} \qquad \left(\frac{\partial}{\partial \bar{\boldsymbol{n}}} = \frac{\partial}{\partial \bar{\boldsymbol{x}}} \cdot \bar{\boldsymbol{n}}\right) \tag{11.8}$$

and the Robin boundary condition (boundary with given admittance)

$$\frac{\partial \widetilde{P}_{f,S}(\bar{\boldsymbol{x}})}{\partial \bar{\boldsymbol{n}}} = -ick\rho_f A_n \widetilde{P}_{f,S}(\bar{\boldsymbol{x}}) - ick\rho_f A_n \widetilde{P}_{f,I}(\bar{\boldsymbol{x}}) - \frac{\partial \widetilde{P}_{f,I}(\bar{\boldsymbol{x}})}{\partial \bar{\boldsymbol{n}}} \tag{11.9}$$

In the upper equations, $\widetilde{P}_f$ is the given complex amplitude of fluid pressure; $\partial/\partial\bar{\boldsymbol{n}} = \partial/\partial\bar{\boldsymbol{x}} \cdot \bar{\boldsymbol{n}}$ is an operator for normal projection of the gradient; $\bar{\boldsymbol{n}}$ is the normal-direction vector outward to the acoustic boundary; c is the sound speed in fluid; k is the wave numbers; ρ_f is the fluid real mass density; $\widetilde{V}_f$ is the complex amplitude of fluid vibrating velocity; A_n is the boundary admittance.

12.2 The Non-Homogeneous Helmholtz Equation

12.2.1 The Non-Convective Non-Homogeneous Helmholtz Equation

The acoustic sources can be accounted for by handling a mass source density. Thus, a source field can be described by its mass flow density q_m as a contribution to the monopole sources and $\vec{\nabla} \cdot \vec{f}/\rho_f$ as a contribution to dipole sources, with $\vec{f}$ being a unitary momentum. Then, without considering the fluid flow at a given fluid point in a harmonic equation, the non-homogenous Helmholtz Equation in the non-convective form can be expressed as:

$$\nabla^2 \tilde{P}_f(\bar{x}) + k^2 \tilde{P}_f(\bar{x}) = -ickq_m + \vec{\nabla} \cdot \vec{f} \tag{11.10}$$

Actran uses the velocity potential instead of the pressure as unknown. The complex amplitude of fluid pressure $\tilde{P}_f(\bar{x})$ can be deduced from the complex amplitude of the fluid potential $\tilde{\psi}_f(\bar{x})$:

$$\tilde{P}_f(\bar{x}) = -ick\tilde{\psi}_f(\bar{x}) \tag{11.11}$$

Thus, the non-homogenous Helmholtz Equation in the non-convective form can be rewritten as:

$$\nabla^2 \tilde{\psi}_f(\bar{x}) + k^2 \tilde{\psi}_f(\bar{x}) = q_m - \frac{\vec{\nabla} \cdot \vec{f}}{i\omega} \qquad (ck = \omega) \tag{11.12}$$

12.2.2 The Convective Non-Homogeneous Helmholtz Equation

In considering the fluid flow in steady-state dynamic analysis, the non-homogenous Helmholtz Equation in the convective form can be expressed as:

$$\begin{aligned} &\nabla^2 \tilde{\psi}_f(\bar{x}) - \left(ik + \vec{M} \cdot \nabla\right)^2 \tilde{\psi}_f(\bar{x}) \\ &= \frac{\rho_T}{i\omega\rho_f}\left(\nabla \cdot (\vec{v}q_m) + i\omega q_m - \vec{\nabla} \cdot \vec{f} + \vec{\nabla} \cdot \left(\vec{M}\vec{M} \cdot \vec{f}\right) + ik\vec{M} \cdot \vec{f}\right) \end{aligned} \tag{11.13}$$

The complex amplitude of the fluid pressure can be expressed as:

$$\tilde{P}_f(\bar{x}) = -\frac{\rho_T}{\rho_f} c\left(ik\tilde{\psi}_f(\bar{x}) + \vec{M} \cdot \vec{\nabla}\tilde{\psi}_f(\bar{x})\right) \tag{11.14}$$

The acoustic velocity can be derived from the velocity potential ψ_f:

$$\vec{v} = \frac{1}{\rho_T}\vec{\nabla}\psi_f \tag{11.15}$$

In addition, the non-homogenous Helmholtz Equation needs to be noted. The right-side terms correspond to acoustic sources, and the left-side terms correspond to the research domain. Thus, the expressions of the left-side terms containing a position vector $\bar{\boldsymbol{x}} = [x,y,z]^T$, and this position vector indicates a given point in the research domain.

12.2.3 Expressions of Boundary Conditions of Non-homogenous Helmholtz Equation

The non-homogenous Helmholtz Equation treats total waves. Thus, the Dirichlet boundary condition of the total acoustic pressure can be expressed as:

$$\widetilde{P}_f(\bar{\boldsymbol{x}}) = \widetilde{P}_f \tag{11.16}$$

The Neumann boundary condition of the total acoustic pressure can be expressed as:

$$\frac{\partial \widetilde{P}_f(\bar{\boldsymbol{x}})}{\partial \bar{\boldsymbol{n}}} = -ick\rho_f \widetilde{V} \tag{11.17}$$

The Robin boundary condition of the total acoustic pressure can be expressed as:

$$\frac{\partial \widetilde{P}_f(\bar{\boldsymbol{x}})}{\partial \bar{\boldsymbol{n}}} = -ick\rho_f A_n \widetilde{P}_f(\bar{\boldsymbol{x}}) \tag{11.18}$$

12.3 Homogeneous Helmholtz Equation VS Non-Homogeneous Helmholtz Equation

Whether adopting a homogenous or non-homogenous Helmholtz Equation is one thing that users cannot make their choices. Actran automatically matches the proper equation to specific application scenarios. Two factors influence the equation application. One is the acoustic source type; the other is the location of the acoustic source. If Actran adopts the non-homogenous Helmholtz Equation, the first triggering condition is that sources must be in the finite-element domain. Then the second triggering condition is that the sources must be spherical in 3D or axisymmetric cases. Be cylindrical in 2D cases. Be dipole in 2D or 3D cases. If Actran adopts the homogenous Helmholtz Equation, the first triggering condition is that sources must be in the infinite domain. Then the second triggering condition is that the sources must be spherical in 3D or axisymmetric cases. Be cylindrical in 2D or 3D cases. Be planar in 2D, 3D or axisymmetric cases.

Chapter 13

Acoustics in Nastran

It is necessary to note that this chapter's theory reference is [34]. Based on reference [34], this paper quotes the formula and symbols; then reorganises the logic and expressions. Then, we can learn that acoustic mathematical models in Nastran are based on small motion theory. No convective momentum terms are considered. The acoustic model obeys the linear pressure-density relationship like a structure model. Thus, the acoustic equation in Nastran is analogous to the total-wave equation in Abaqus or the non-homogenous Helmholtz equation in Actran.

13.1 Acoustic-Field Governing Equation

The acoustic-field governing equation in Nastran is the same as in Abaqus, except that Nastran does not account for the volumetric drag. The acoustic-field governing equation in Nastran in steady-state dynamic analysis can be expressed as:

$$\begin{gathered} -\omega^2 \frac{1}{K_f} \widetilde{P}_f(\bar{\boldsymbol{x}}) - \frac{1}{\rho_f} \nabla^2 \widetilde{P}_f(\bar{\boldsymbol{x}}) = 0 \\ \left(\nabla^2 \widetilde{P}_f(\bar{\boldsymbol{x}}) = \frac{\partial}{\partial \bar{\boldsymbol{x}}} \cdot \frac{\partial \widetilde{P}_f(\bar{\boldsymbol{x}})}{\partial \bar{\boldsymbol{x}}} \right) \end{gathered} \tag{12.1}$$

In the upper equation, K_f is the fluid bulk modulus; ρ_f is the fluid real mass density; $\widetilde{P}_f(\bar{\boldsymbol{x}})$ is the complex amplitude of the acoustic pressure; ∇^2 is the Laplacian operator.

Chapter 14

Practical Accuracy Benchmarks of Simulation Software

It is necessary to note that this chapter's experimental data come from [78]. Before starting a series of simulations for verifying software performances, practical accuracy benchmarks are needed to examine the reliability of different software's numerical methods. The samples used in the practical accuracy benchmark are two PVC panels with 300 and 600 holes [78], respectively. Their natural frequencies have been tested in experiments, and the data from the laboratory is a practical reference. Consequently, the verification practices are a series of eigenvalue extraction analyses driven by numerical methods from different software.

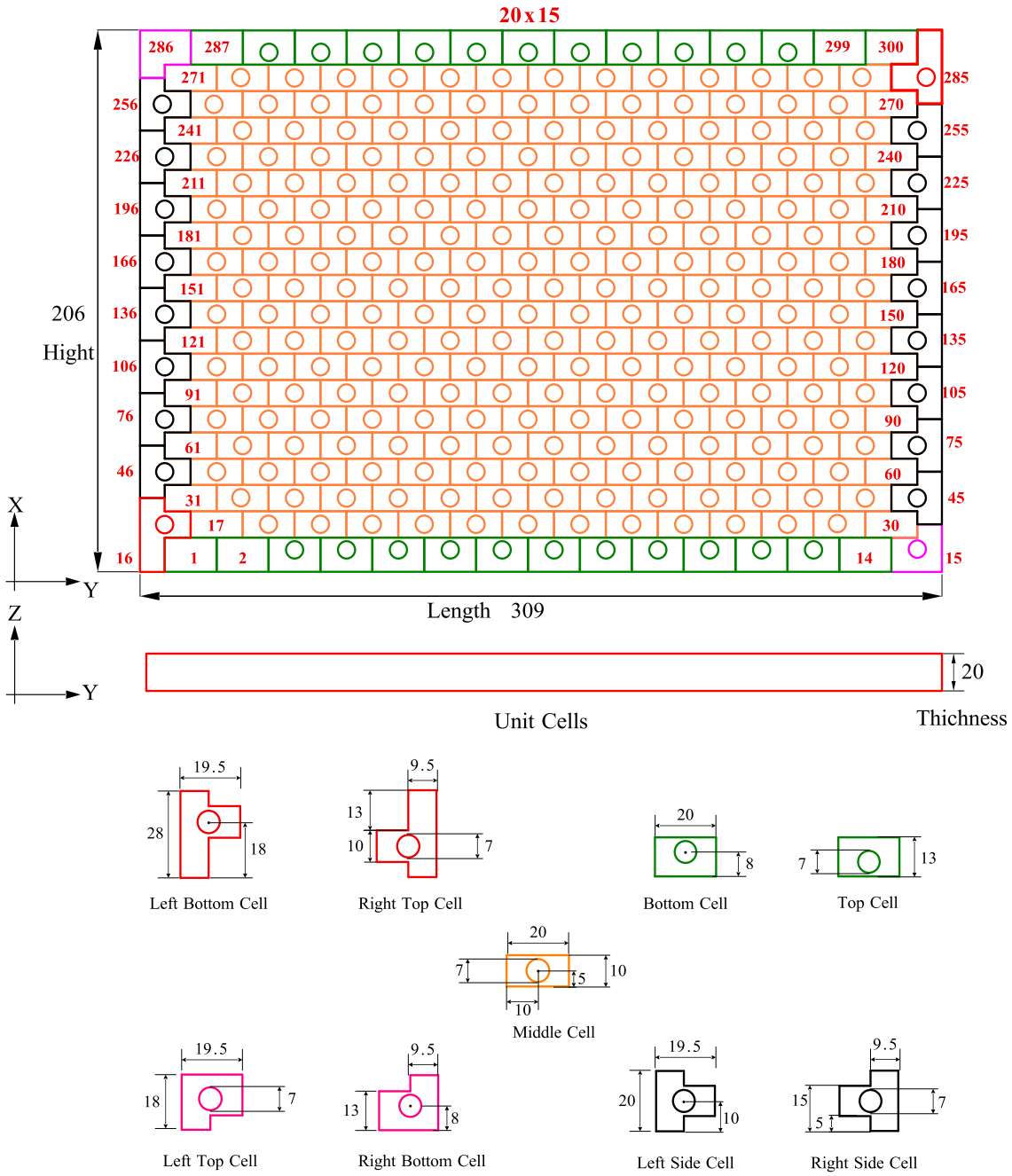


Figure 14.1: The 300-hole PVC panel with specific subdivisions.

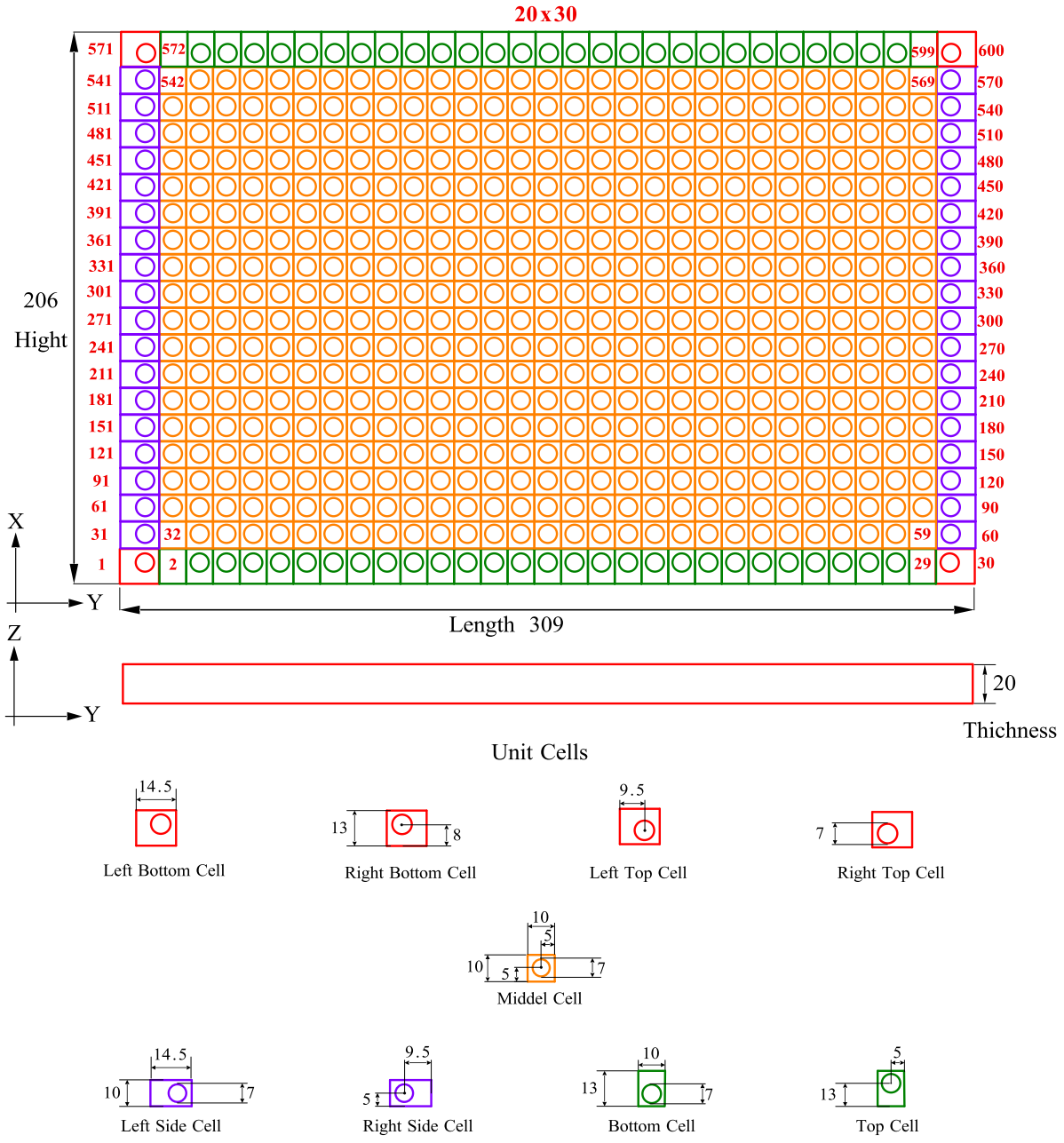


Figure 14.2: The 600-hole PVC panel with specific subdivisions.

14.1 Practical Accuracy Benchmarks of Nastran

14.1.1 Testing Objects of Nastran

Nastran is a software that can adopt many different numerical methods to execute eigenvalue extraction analyses, and this paper adopts four representatives of them. They are the "Ordinary Method", the "ACMS Method", the "List Superelement Method", and the "Image Superelement Method".

The "Ordinary Method" is commonly used and needs no specific elaboration. Because the "Ordinary Method" does not do any additional modifications to the original model. At the same time, the "ACMS Method" is a derivative of the superelement techniques. The test with this method is executed internally and automatically. The ACMS means automated component mode synthesis.

In comparison, the "List Superelement Method" and the "Image Superelement Method" must set each substructure as the superelement. Thus, the superelement setting is a massive work when the model contains plenty of periodic structures. As a preparation step, the two PVC plates need partition operations to fulfil the requirements of the "List Superelement Method" and "Image Superelement Method". These partition operations do not influence the settings of other numerical methods. After the partition operation, the PVC panel with 300 holes becomes a panel consisting of 300 cells (see **Figure 14.1**), and the PVC panel with 600 holes becomes a panel of 600 cells (see **Figure 14.2**).

14.1.2 Partitions of 300-hole Panel for The List Superelement Method

Figure 14.1 shows the partitioned 300-hole panel. There are 20 columns along the X-axis and 15 rows along the Y-axis, and the whole panel presents a staggered arrangement. Along the X-axis, the length of the panel is 309 mm; along the Y-axis, the length of the panel is 206 mm; along the Z-axis, the thickness of the panel is 20 mm. All 300 cells come from nine different cell patterns. They are the "Left Bottom Cell", the "Right Top Cell", the "Bottom Cell", the "Top Cell", the "Right Bottom Cell", the "Left Top Cell", the "Left Side Cell", and the "Right Side Cell". Their sides length can be read in **Figure 14.1**, and they are in the millimetre unit.

Based on this kind of partition, the 300 cells can be defined as superelements by manually setting the inner DOFs of each cell while adopting the "List Superelement Method". In **Figure 14.1**, the red ID number of cells on the XY-plane indicates the superelement sequence number. The sequence number rules the superelement processing order. In each row, cells are listed from left to right in ascending order, and then each row of cells from bottom to top piles up in ascending order.

14.1.3 Partitions of 300-hole Panel for The Image Superelement Method

In comparison, the "Image Superelement Method" defines each superelement by manually setting the inner DOFs of a "Primary Superelement" and the boundary DOFs of a "Secondary Superelement". The boundary DOFs of a "Secondary Superelement" have positioning effects;

thus, the “Secondary Superelements” can copy the matrices of the “Primary Superelements” and place them in corresponding positions.

Based on practical manipulations, the cell-ID1, cell-ID2, cell-ID15, cell-IDs6, cell-ID17, cell-ID30, cell-ID31, cell-ID45, cell-ID46, cell-ID285, cell-ID286, cell-ID287, and cell-ID300 must be defined as the “Primary Superelements”. The reasons to let them be the “Primary Superelements” are different, and regarding the reasons, they can be subdivided into three categories.

In **Figure 14.1**, for the first category, we can see that the red cell-ID16 is called as “Left Bottom Cell”, the red cell-ID285 is called as “Right Top Cell”, the pink cell-ID15 is called as “Right Bottom Cell”, and the pink cell-ID286 be called as “Left Top Cell”. These four cells do not belong to any cell groups due to their unique geometry shapes, and their matrices cannot be copied by other cells. Thus, these four cells can be called nonreplicable “Primary Superelements”.

Then, for the second category, the cell-ID1 and the cell-ID300 have analogous reasons to be defined as the “Primary Superelement”. The green cell-ID1 belongs to the “Bottom Cell” group, and the cells in this group have the same geometry shape. However, the cell-ID1 connects three cells, while other cells in the same group connect four. Similarly, the green cell-ID300 belongs to the “Top Cell” group, and the cells in this group have the same geometry shape. However, the cell-ID300 connects three cells, while other cells in the same group connect four. We can see that the cell-ID1 and the cell-ID300 have distinct boundary connections regarding the other cells of their cell groups. Since the boundary connections have the positioning effect, the cell-ID1 and the cell-ID300 cannot be the positioning reference to the other cells in their cell groups. Therefore, these two cells can be called positioning-disabled “Primary Superelements”.

Finally, for the third category, the green cell-ID2 belongs to the “Bottom Cell” group, the yellow cell-ID17 belongs to the “Middle Cell” group, the yellow cell-ID30 belongs to the right-side part of the “Middle Cell” group, the yellow cell-ID31 belongs to the left-side part of the “Middle Cell” group, the black cell-ID45 belongs to the “Right Side Cell” group, the black cell-ID46 belongs to the “Left Side Cell” group, and the green cell-ID287 belongs to the “Top Cell” group. These seven cells can be called the replicable “Primary Superelements”. In the “Bottom Cell” group, from the cell-ID3 to the cell-ID14 with an ID increment of 1 are the “Secondary Superelements”, and they copy the matrices of the cell-ID2. Then, in the right-side part of the “Middle Cell” group, from the cell-ID60 to the cell-ID270 with an ID increment of 30 are the “Secondary Superelements”, and they copy the matrices of the cell-ID30. Similarly, in the left-side part of the “Middle Cell” group, from the cell-ID61 to the cell-ID271 with an ID increment of 30 are the “Secondary Superelements”, and they copy the matrices of the cell-ID31. Then, in the residual part of the “Middle Cell” group, from the cell-ID18 to the cell-ID284 with an ID increment of 1 in each row and an ID jump of 2 when the row number changes. They copy the matrices of the cell-ID17. As for the “Right Side Cell” group, from the cell-ID75 to the cell-ID255 with an ID increment of 30 are the “Secondary Superelements”, and they copy the matrices of the cell-ID45. In the same way, in the “Left Side Cell” group, from the cell-ID76 to the cell-ID256 with an ID increment of 30 are the “Secondary Superelements”, and they copy the matrices of the cell-ID46.

14.1.4 Partition of 600-hole Panel for The List Superelement Method

Figure 14.2 shows the partitioned 600-hole panel. There are 20 cells along the X-axis and 30 cells along the Y-axis, and the whole panel presents an in-line arrangement. Along the Y-axis, the length of the panel is 309 mm; along the X-axis, the length of the panel is 206 mm; along the Z-axis, the thickness of the panel is 20 mm. All 600 cells come from nine different cell forms. They are the "Left Bottom Cell", the "Right Top Cell", the "Bottom Cell", the "Top Cell", the "Right Bottom Cell", the "Left Top Cell", the "Left Side Cell", the "Right Side Cell" and their sizes can be read in Figure 14.2.

The superelement definition of the 600-cell panel is analogous to the 300-cell panel. Each superelement is defined by setting the inner DOFs and ordering the superelements with a given sequence. The sequence rules the superelements from left to right in each row with ascending IDs, then each row from bottom to top in ascending order.

14.1.5 Partition of 600-hole Panel for The Image Superelement Method

The in-line arrangement of the 600-cells panel makes the definition of the "Primary Superelement" relatively simple compared to the 300-cell panel. Based on practical manipulations, the cell-ID1, cell-ID2, cell-ID30, cell-ID31, cell-ID32, cell-ID60, cell-ID571, cell-ID572, and cell-ID600 must be defined as the "Primary Superelements". The reasons to let them be the "Primary Superelements" are different, and regarding the reasons, they can be subdivided into two categories.

In Figure 14.2, as the first category, we can see that the red cell-ID1 is called "Left Bottom Cell", the red cell-ID30 is called "Right Bottom Cell", the red cell-ID571 is called "Left Top Cell", and the red cell-ID600 be called as "Right Top Cell". These four cells do not belong to any cell groups due to their unique geometry shapes, and their matrices cannot be copied by other cells. Thus, these four cells can be called nonreplicable "Primary Superelements".

Then, as the second category, the green cell-ID2 belongs to the "Bottom Cell" group, the black cell-ID31 belongs to the "Right Side Cell" group, the yellow cell-ID32 belongs to the "Middle Cell" group, the black cell-ID60 belongs to the "Left Side Cell" group, and the green cell-ID572 belongs to the "Top Cell" group. These five cells can be called the replicable "Primary Superelements". In the "Bottom Cell" group, from the cell-ID3 to the cell-ID29 with an ID increment of 1 are the "Secondary Superelements", and they copy the matrices of the cell-ID2. As for the "Right Side Cell" group, from the cell-ID31 to the cell-ID541 with an ID increment of 30 are the "Secondary Superelements", and they copy the matrices of the cell-ID31. In the same way, in the "Left Side Cell" group, from the cell-ID90 to the cell-ID570 with an ID increment of 30 are the "Secondary Superelements", and they copy the matrices of the cell-ID46. Then, in the "Middle Cell" group, from the cell-ID33 to the cell-ID569 with an ID increment of 1 in each row and an ID jump of 2 when the row number changes. They copy the matrices of the cell-ID32.

14.1.6 Testing Results of the 300-hole Panel

Table 14.1: Eigenfrequencies of the 300-hole panel (Nastran vs Experiment).

Eigenmode Number	f_{EXP} (Hz)	f_{OD} (Hz)	f_{ACMS} (Hz)	f_{SESET} (Hz)	f_{CSUPER} (Hz)
1	272	277.4	277.4	277.4	277.4
2	289	293.6	293.6	293.6	293.6
3	621	628.4	628.7	628.4	628.4
4	681	689.2	689.9	689.3	689.2
5	748	765	765.9	765.1	765
6	901	923.6	924.6	923.8	923.6
7	1116	1134.4	1136.1	1134.9	1134.4
8	1271	1296.2	1300.9	1296.8	1296.2
9	1548	1582.3	1591.5	1583.2	1582.3

Table 14.2: Eigenfrequencies' practical accuracy of the 300-hole panel (Nastran vs Experiment).

Eigenmode Number	ERR_{OD} (%)	ERR_{ACMS} (%)	ERR_{SESET} (%)	ERR_{CSUPER} (%)
1	-1.99	-1.99	-1.99	-1.99
2	-1.59	-1.59	-1.59	-1.59
3	-1.19	-1.24	-1.19	-1.19
4	-1.2	-1.31	-1.22	-1.2
5	-2.27	-2.39	-2.29	-2.27
6	-2.51	-2.62	-2.53	-2.51
7	-1.65	-1.8	-1.69	-1.65
8	-1.98	-2.35	-2.03	-1.98
9	-2.22	-2.81	-2.27	-2.22

14.1.7 Testing Results of the 600-hole Panel

Table 14.3: Eigenfrequencies of the 600-hole panel (Nastran vs Experiment).

Eigenmode Number	f_{EXP} (Hz)	f_{OD} (Hz)	f_{ACMS} (Hz)	f_{SESET} (Hz)	f_{CSUPER} (Hz)
1	180	190.2	190.2	190.2	190.2
2	273	287.8	287.8	287.7	287.7
3	459	480.5	480.6	480.5	480.5
4	626	640.7	645.9	645.7	645.7
5	690	721.9	722.1	721.9	721.9
6	772	807.4	808	807.4	807.4
7	925	960.5	960.8	960.6	960.5
8	1005	1030.8	1032.3	1031	1030.8
9	1442	1443.1	1451.7	1443.2	1443.1

Table 14.4: Eigenfrequencies' practical accuracy of the 600-hole panel (Nastran vs Experiment).

Eigenmode Number	ERR_{OD} (%)	ERR_{ACMS} (%)	ERR_{SESET} (%)	ERR_{CSUPER} (%)
1	-5.67	-5.67	-5.67	-5.67
2	-5.42	-5.42	-5.38	-5.38
3	-4.68	-4.71	-4.68	-4.68
4	-2.35	-3.18	-3.15	-3.15
5	-4.62	-4.65	-4.62	-4.62
6	-5.13	-5.22	-5.13	-5.13
7	-3.84	-3.87	-3.85	-3.84
8	-2.57	-2.71	-2.59	-2.57
9	-0.076	-0.67	-0.083	-0.076

14.2 Practical Accuracy Benchmark of Abaqus

14.2.1 Testing Objects of Abaqus

In comparison, Abaqus has two representative numerical methods that may be compatible with Nastran. These numerical methods are the “SIM-Architecture-Based Ordinary Method” and the “SIM-Architecture-Based AMS Method”. Notably, the “SIM-Architecture-Based AMS Method” in Abaqus is analogous to the “ACMS Method with Lanczos Eigensolver” in Nastran. They are derivatives of superelement techniques.

As introduced before, the “SIM-Architecture-Based Ordinary Method” and the “SIM-Architecture-Based AMS Method” are two promising methods in Abaqus doing eigenvalue extraction analyses. The first one is compatible with the “Ordinary Method”, and the second is compatible with the “ACMS Method” in Nastran. These two methods do not apply any modifications to the original model. Thus, no further elaborations are present here. The accuracy benchmark results are listed in the following tables.

14.2.2 Testing Results of the 300-hole Panel

Table 14.5: Eigenfrequencies of the 300-hole panel (Abaqus vs Experiment).

Eigenmode Number	f_{EXP} (Hz)	f_{OD} (Hz)	f_{AMS} (Hz)
1	272	288.8	288.8
2	289	304.6	304.6
3	621	642.5	643.2
4	681	669.2	669.7
5	748	777.9	779.1
6	901	873.8	875.6
7	1116	1171.4	1174.5
8	1271	1357.7	1360.5
9	1548	1491.6	1499.3

Table 14.6: Eigenfrequencies' practical accuracy of the 300-hole panel (Abaqus vs Experiment)

Eigenmode Number	ERR_{OD} (%)	ERR_{AMS} (%)
1	-6.18	-6.18
2	-5.4	-5.4
3	-3.46	-3.57
4	1.73	1.67
5	-4	-4.16
6	3.02	2.82
7	-4.96	-5.24
8	-6.28	-7.04
9	3.64	3.15

14.2.3 Testing Results of the 600-hole Panel

Table 14.7: Eigenfrequencies of the 600-hole panel (Abaqus vs Experiment).

Eigenmode Number	f_{EXP} (Hz)	f_{OD} (Hz)	f_{AMS} (Hz)
1	180	207.2	207.2
2	273	282.1	281.1
3	459	501.4	501.6
4	626	621.9	622.3
5	690	742.3	743
6	772	759.2	760.2
7	925	970.8	972.6
8	1005	1051.4	1053.8
9	1442	1445.8	1452.5

Table 14.8: Eigenfrequencies' practical accuracy of the 600-hole panel (Abaqus vs Experiment).

Eigenmode Number	ERR_{OD} (%)	ERR_{AMS} (%)
1	-15.11	-15.11
2	-3.33	-2.97
3	-9.24	-9.28
4	0.66	0.59
5	-7.58	-7.68
6	1.66	1.53
7	-4.58	-5.15
8	-4.62	-4.86
9	-0.26	-0.73

14.3 Practical Accuracy Benchmark of Actran

14.3.1 Testing Objects of Actran

As the third software candidate, Actran does not have a very efficient numerical method as powerful as Nastran or Abaqus to execute eigenvalues extraction analyses. The numerical method in Actran is analogous to the "Ordinary Method" in Nastran or Abaqus. Thus, no further elaboration is needed here.

14.3.2 Testing Results of the 300-hole Panel

Table 14.9: Eigenfrequencies of the 300-hole panel (Actran vs Experiment).

Eigenmode Number	f_{EXP} (Hz)	f_{OD} (Hz)
1	272	280.9
2	289	303.2
3	621	640.2
4	681	717.9
5	748	787.3
6	901	959.3
7	1116	1163.8
8	1271	1329.1
9	1548	1647

Table 14.10: Eigenfrequencies' practical accuracy of the 300-hole panel (Actran vs Experiment).

Eigenmode Number	ERR_{OD} (%)
1	-3.27
2	-4.91
3	-3.09
4	-5.42
5	-5.25
6	-6.47
7	-4.28
8	-4.57
9	-6.4

14.3.3 Testing Results of the 600-hole Panel

Table 14.11: Eigenfrequencies of the 600-hole panel (Actran vs Experiment).

Eigenmode Number	f_{EXP} (Hz)	f_{OD} (Hz)
1	180	228.5
2	273	297.8
3	459	546.7
4	626	673.9
5	690	762.9
6	772	858
7	925	1047.2
8	1005	1152.6
9	1442	1552.1

Table 14.12: Eigenfrequencies' practical accuracy of the 600-hole panel (Actran vs Experiment).

Eigenmode Number	ERR_{OD} (%)
1	-26.94
2	-9.08
3	-19.12
4	-7.65
5	-10.57
6	-11.14
7	-13.21
8	-14.69
9	-7.63

14.4 Chapter Conclusions

This chapter adopts the relative error as the accuracy indicator. It is the ratio that the eigenfrequency error divides the experimental eigenfrequency. And the eigenfrequency error is the discrepancy between the experimental and numerical eigenfrequency. Based on the definition, the relative error can be expressed as:

$$\mathrm{ERR}(\%) = \frac{f_{\mathrm{EXP}} - f_{\mathrm{NUM}}}{f_{\mathrm{EXP}}} \cdot 100\% \tag{13.1}$$

In the upper equation, f_{EXP} indicates the experimental eigenfrequency; f_{NUM} indicates the numerical eigenfrequency; $\mathrm{ERR}(\%)$ indicates the relative error. The relative error is negative if the numerical frequency exceeds the experimental frequency. On the contrary, if the numerical frequency is smaller than the experimental frequency, the relative error is positive. In **Table 14.1**, **Table 14.3**, **Table 14.5**, **Table 14.7, Table 14.9**, and **Table 14.11**, f_{OD} indicates the numerical eigenfrequencies that come from the "Ordinary Method". Then, in **Table 14.1** and **Table 14.3,** f_{ACMS}, f_{SESET}, and f_{CSUPER} indicate the numerical eigenfrequencies come from the "ACMS Method", the "List Superelement Method", and the "Image Superelement Method" in Nastran, respectively. In the same way, in **Table 14.5** and **Table 14.7**, f_{AMS} indicates the numerical eigenfrequencies come from the "SIM-Architecture-Based AMS Method" in Abaqus. As for the relative error, in **Table 14.2**, **Table 14.4**, **Table 14.6**, **Table 14.8**, **Table 14.10**, and **Table 14.12**, $\mathrm{ERR}_{\mathrm{OD}}(\%)$ indicate relative errors of the "Ordinary Method". Then, in **Table 14.2** and **Table 14.4**, $\mathrm{ERR}_{\mathrm{ACMS}}(\%)$, $\mathrm{ERR}_{\mathrm{SESET}}(\%)$, and $\mathrm{ERR}_{\mathrm{CSUPER}}(\%)$ indicate the relative errors come from the "ACMS Method", the "List Superelement Method", and the "Image Superelement Method" in Nastran, respectively. Similarly, in **Table 14.5** and **Table 14.7**, $\mathrm{ERR}_{\mathrm{AMS}}(\%)$ indicates the relative errors of the "SIM-Architecture-Based AMS Method" in Abaqus.

Based on a glance view of the eigenvalue extraction analyses, on the one hand, Nastran has the highest practical accuracy; then, the practical accuracy of Abaqus is relatively moderate; finally, Actran has the relatively lowest practical accuracy. On the other hand, we can see that the practical accuracies of different numerical methods in the corresponding software are very close to each other.

Based on the numerical eigenfrequencies of the two distinct testing samples, the 300-hole panel gets more accurate results than the 600-hole panel. The reason is that the 300-hole panel has a lower degree of perforation than the 600-hole panel. In other words, the structure of the 300-hole panel is relatively more homogenous than the 600-hole panel. In common sense, the structure's homogeneity always influences the accuracy of FEM simulation.

In a more specific view of Nastran simulations, the relative errors of the results of the 300-hole panel are less than 2.7% (see **Table 14.2**). And the relative error of the first numerical eigenfrequency is less than 2%. In the same way, the relative errors of the results of the 600-hole panel are less than 5.7% (see **Table 14.4**). And the relative error of the first numerical eigenfrequency is the highest, around 5.7%. Then, the relative error of the ninth numerical eigenfrequencies of the 600-hole panel is the lowest and is less than 0.7%. Additionally, we can see that all of Nastran's simulation results are bigger than the experimental results. Therefore, Nastran is stable and accurate in eigenvalue extraction analyses.

In a more specific view of Abaqus simulations, the relative errors of the results of the 300-hole panel are less than 7.1% (see **Table 14.6**). And the relative error of the first numerical eigenfrequency is close to the highest, around 6.2%. In the same way, the relative errors of the results of the 600-hole panel are less than 15.2% (see **Table 14.8**). And the relative error of the first numerical eigenfrequency is the highest, around 15.2%. Then, the relative error of the ninth numerical eigenfrequency of the 600-hole panel is the lowest, less than 0.8%. Additionally, we can see that some of Abaqus's simulation results are bigger, and some are smaller regarding the experimental results. Therefore, Abaqus is not as stable and accurate as Nastran in eigenvalue extraction analyses.

In a more specific view of Actran simulations, the relative errors of the results of the 300-hole panel are less than 6.5% (see **Table 14.10**). And the relative error of the first numerical eigenfrequency is around 3.3%. In the same way, the highest relative error of the results of the 600-hole is around 27%, and this relative error happens in the first numerical eigenfrequency. At the same time, the second highest relative error is around 20%, and this relative error happens at the third numerical eigenfrequency (see **Table 14.12**). Then, the relative error of the ninth numerical eigenfrequency of the 600-hole panel is the lowest, around 7.7%. Additionally, we can see that Actran's simulation results are bigger than the experimental results. Then, Actran's accuracy in eigenvalue extraction analyses of the 600-hole panel is much less than that of Nastran.

Chapter 15

Relative Accuracy and Efficiency Benchmarks of Simulation Software

This chapter's relative accuracy and efficiency benchmarks are subdivided into two topics that correlate with the metamaterial's mechanical properties. One topic is the steady-state dynamic analysis, while another is the eigenvalue extraction analysis. The candidate software platforms are Nastran, Abaqus, and Actran. They are platforms that can supply different numerical methods and simulation strategies.

The eigenvalue extraction analysis is a way to test the eigensolver capability of the software. Similarly, the steady-state dynamic analysis is a way to test the matrix-equations-solving capability of the software. In addition, if the frequency dependency of the model is active in steady-state dynamic analyses, the model rebuilding or matrix updating mechanism is activated. Thus, processing the frequency-dependent model in steady-state dynamic analyses can test the software's intermediate data management capability.

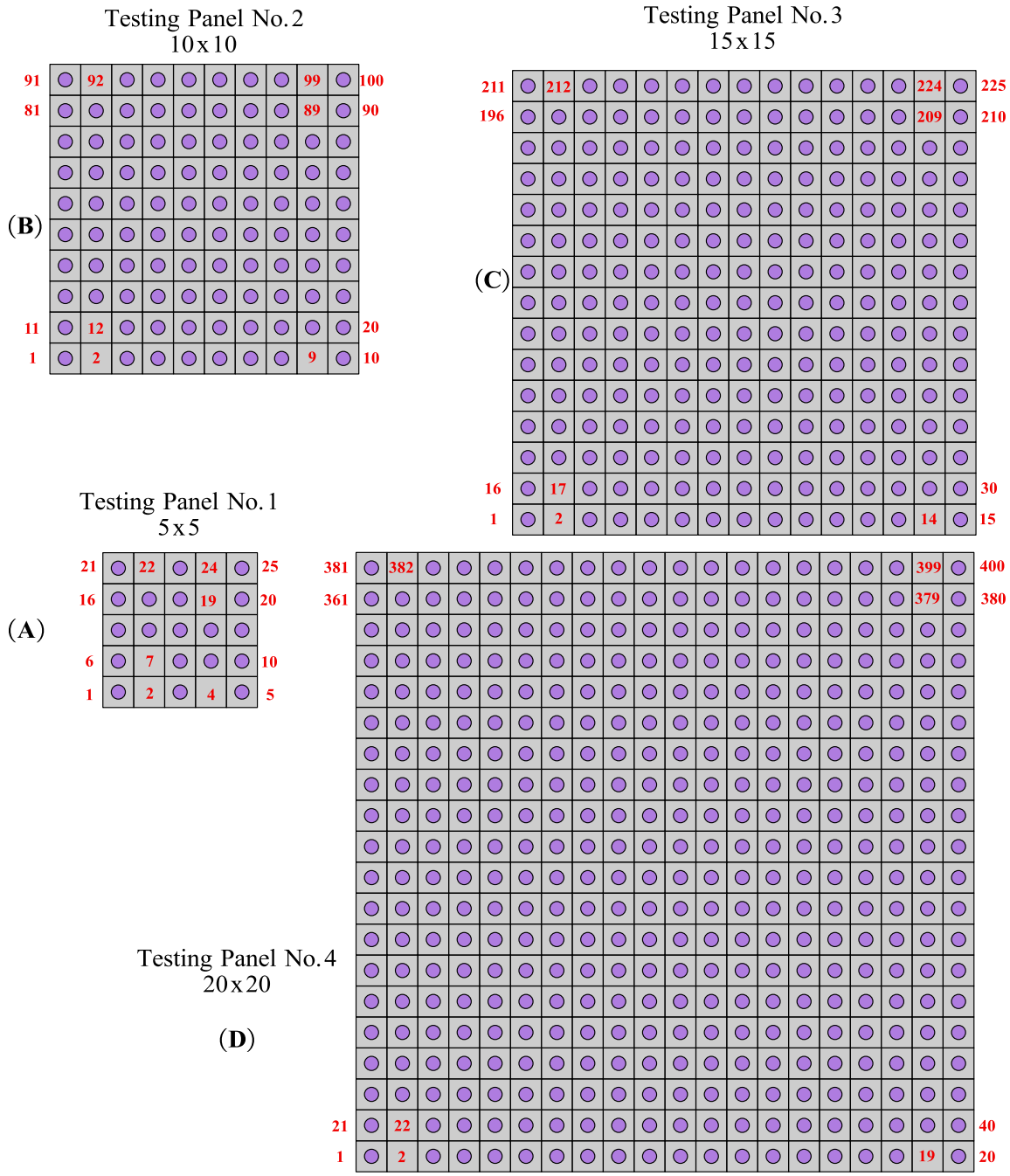


Figure 15.1: Four size-controlled acoustic metamaterial panels.

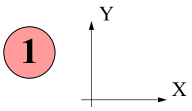
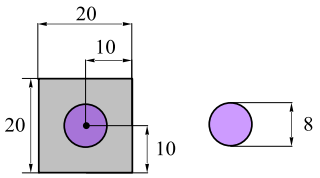
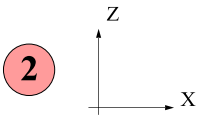
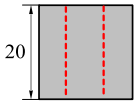
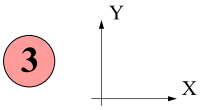
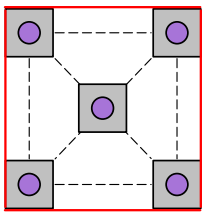
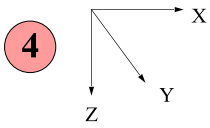
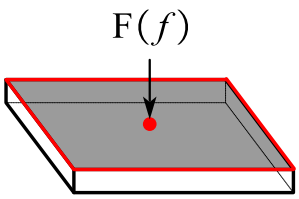
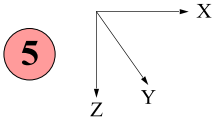
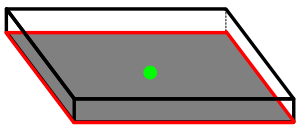
Coordinate	Layout Drawing	Parameters		
1 		Seat (Cube)	Length = 20 mm (X coordinate)	Width = 20 mm (Y coordinate)
		Core (Bar)	Diameter = 8 mm (Radial coordinate)	Centre: (10, 10) (X, Y coordinates)
2 		Seat (Cube)	Thickness = 20 mm (Z coordinate)	
3 	Number of Cells: NxN 	Num.cells (NxN)	Sizes in X, Y, Z (mm)	Num.Grids (Mesh)
		5x5	100x100x20	3605
		10x10	200x200x20	14205
		15x15	300x300x20	38105
		20x20	400x400x20	56405
4 	Harmonic Concentrated Force 	Num.cells (NxN)	Locations (X, Y, Z)	
		5x5	(50, 50, 0)	
		10x10	(100, 100, 0)	
		15x15	(150, 150, 0)	
		20x20	(200, 200, 0)	
5 	Response Point (Centre Point) 	Num.cells (NxN)	Locations (X, Y, Z)	
		5x5	(50, 50, 20)	
		10x10	(100, 100, 20)	
		15x15	(150, 150, 20)	
		20x20	(200, 200, 20)	

Figure 15.2: Mechanical information of four size-controlled acoustic metamaterial panels.

In steady-state dynamic analyses, the frequency dependency of materials can be active or inactive. In addition, two simulation spaces can be the choice: the physical space or the modal space. Furthermore, for each software, the numerical methods available in eigenvalue extraction analyses are also available in steady-state dynamic analyses. In simulations, the combinations of simulation spaces and numerical methods can be called simulation strategies.

At this moment, the material's frequency dependency, the simulation space, and the numerical method are three control factors in joint works of specific simulation scenarios. However, these three control factors are not always compatible with each other. Proper arrangements of them need clearly understand their matching conditions. Then, subsections **15.1.4**, **15.2.2**, and **15.3.2** discuss corresponding matching conditions of the material's frequency dependency, the simulation space, and the numerical method in Nastran, Abaqus, and Actran, respectively.

In eigenvalues extraction analyses, the frequency dependency of materials is inactive. Nastran has four available numerical methods: the "Ordinary Method", the "ACMS Method", the "List Superelement Method", and the "Image Superelement Method". In comparison, Abaqus and Actran are not as capable as Nastran. Abaqus has two available numerical methods: the "Ordinary Method" and the "AMS Method". While Actran has only one numerical method: the "Ordinary Method".

The testing objects are size-controlled metamaterial panels for controlling the simulation burden. The fundamental configuration of metamaterial has been introduced in chapter **2**. For more information, please see subsection **2.2.1**. And this paper adopts four size-controlled quadrate panels. These four panels are displayed in **Figure 15.1**. They are "Panel No.1" with 25 cells, "Panel No.2" with 100 cells, "Panel No.3" with 225 cells, and the "Panel No.4" with 400 cells. We can see that some cells of the four panels are marked with red numbers. The reason is that they correlate with the settings of the "List Superelement Method" and the "Image Superelement Method" in Nastran. For specific descriptions of these two superelement-correlated methods in Nastran, please see subsections **15.1.1** and **15.1.2**.

These four testing panels consist of many cells; thus, these panels' sizes increase with the cell numbers increase. In subregions number 1 and number 2 of **Figure 15.2**, we can see that a unit cell's size (length in X, width in Y and thickness in Z coordinates) is $20\times20\times20\,\mathrm{mm}$. Subsequently, in subregion number 3, we can see that on the XY-plane, the size (length in X and width in Y coordinates) of "Panel No.1" is $100\times100\,\mathrm{mm}$, and "Panel No.2" is $200\times200\,\mathrm{mm}$. Then, the size of "Panel No.3" is $300\times300\,\mathrm{mm}$, and "Panel No.4" is $400\times400\,\mathrm{mm}$. Since these panels have only one-layer cells, their thickness on the Z-axis is $20\,\mathrm{mm}$. With the panel size increasing, the number of mesh grids increases; thus, the DOFs increase. And the number of mesh grids of the four testing panels is listed in the last column of subregion number 3. In addition, we can see that the four lateral boundaries of the testing panels are constrained (highlighted in red lines). And this boundary condition will be applied in the following eigenvalue extraction analyses and steady-stat dynamic analyses.

The frequency response functions of a response point are performance indicators in steady-state dynamic analyses. For simplicity, the external load to the panel is a concentrated force on the centre point of one side, and the response point is the centre point on the opposite side. In subregions number 4 and number 5 of **Figure 15.2**, we can see the layout drawing of the loading point and response point. The coordinates of the loading and response points are the same. The positive direction of the Z-axis coincides with the load's normal direction. Then, in subsections **15.1.5**, **15.1.6**, **15.1.7**, **15.2.3**, **15.3.3**, and section **15.4**, the concentrated load amplitude is set to be 1 newton (F=1N) with a harmonic driving function.

15.1 Relative Accuracy and Efficiency Benchmarks of Nastran

We know that Nastran has four available numerical methods for the benchmark tests. The "Ordinary Method" and the "ACMS Method" need no extra user manipulations to the original model. These two methods are fully automatic. In contrast, the "List Superelement Method" and the "Image Superelement Method" need extra operations by users to the original model. Users must manually define each superelement. The superelement definition of the "List Superelement Method" is elaborated in subsection **15.1.1**. Similarly, the superelement definition of the "Image Superelement Method" is elaborated in subsection **15.1.2**.

15.1.1 Panels Partition for the List Superelement Method

In **Figure 15.1**, we can see that testing "Panel No.1", testing "Panel No.2", testing "Panel No.3", and testing "Panel No.4" have 25, 100, 225, and 400 cells, respectively. Cells marked with red ID numbers in ascending order with each increment of 1, from left to right, then from bottom to top. In this way, the definition of superelements through the "List Superelement Method" is manually setting the inner DOFs of each cell based on the predefined order.

15.1.2 Panels Partition for the Image Superelement Method

Definitions of superelements in the "Image Superelement Method" are more complex than in the "List Superelement Method". The "Image Superelement Method" must define the "Primary Superelements" and the "Secondary Superelements" at the same time. Defining the "Primary Superelements" needs to set the inner DOFs of the corresponding cells, while defining the "Secondary Superelements" needs to set the boundary DOFs of the corresponding cells. More specifically, the boundary DOFs of the "Secondary Superelements" have positioning effects; thus, the "Secondary Superelements" can copy the matrices of the "Primary Superelements" and place them in corresponding positions. The following four paragraphs are the "Image Superelements" setting practices of the four testing panels.

For testing "Panel No.1", cell-ID1, cell-ID2, cell-ID5, cell-ID6, cell-ID7, cell-ID10, cell-ID21, cell-ID22, and cell-ID25 must be defined as the "Primary Superelements". In these "Primary Superelements", cell-ID1, cell-ID5, cell-ID21, and cell-ID25 are called the non-replicable "Primary Superelements". The reason is that these four cells have two connecting boundaries, and their vicinity cells do not have the same boundary connections. They cannot be copied by their vicinity cells. At the same time, cell-ID2, cell-ID6, cell-ID7, cell-ID10, and cell-ID22 are called the replicable "Primary Superelements". For example, cell-ID3 and cell-ID4 have analogous boundary conditions as cell-ID2; thus, they can copy the matrices of cell-ID2, and these two cells are defined as the "Secondary Superelements". Similarly, in the top row, cell-ID21 is the replicable "Primary Superelement", and from cell-ID22 to cell-ID24, with each ID increments of 1, are the "Secondary Superelements". In the left-side column, cell-ID6 is the replicable "Primary Superelement", and from cell-ID11 to cell-ID16, with each ID increment of 5, are the "Secondary Superelements". In the right-side column, the cell-ID10 is the replicable "Primary Superelement", and from cell-ID15 to cell-ID20, with each ID increment of 5, are the "Secondary Superelements". In the middle section, cell-ID7 is the replicable "Primary Superelement", and from cell-ID8 to cell-ID19, with each ID increment of

1 in each row and an ID increment of 2 when row number plus 1, a total of three-rows cells, are the “Secondary Superelements”.

For testing “Panel No.2”, the cell-ID1, cell-ID10, cell-ID91, and cell-ID100 are the non-replicable “Primary Superelements”. In the bottom row, the cell-ID2 is the replicable “Primary Superelement”, and from cell-ID3 to cell-ID9, with each ID increment of 1, are the “Secondary Superelements”. In the top row, cell-ID92 is the replicable “Primary Superelement”, and from cell-ID93 to cell-ID99, with each ID increment of 1, are the “Secondary Superelements”. In the left-side column, cell-ID11 is the replicable “Primary Superelement”, and from cell-ID21 to cell-ID81, with each ID increment of 10, are the “Secondary Superelements”. In the right-side column, the cell-ID20 is the replicable “Primary Superelement”, and from cell-ID30 to cell-ID90, with each ID increment of 10, are the “Secondary Superelements”. In the middle section, cell-ID12 is the replicable “Primary Superelement”, and from cell-ID13 to cell-ID89, with each ID increment of 1 in each row and an ID increment of 2 when row number plus 1, a total of eight-rows cells, are the “Secondary Superelements”.

For testing “Panel No.3”, the cell-ID1, cell-ID15, cell-ID211, and cell-ID225 are the non-replicable “Primary Superelements”. In the bottom row, the cell-ID2 is the replicable “Primary Superelement”, and from cell-ID3 to cell-ID14, with each ID increment of 1, are the “Secondary Superelements”. In the top row, cell-ID212 is the replicable “Primary Superelement”, and from cell-ID213 to cell-ID224, with each ID increment of 1, are the “Secondary Superelements”. In the left-side column, cell-ID16 is the replicable “Primary Superelement”, and from cell-ID31 to cell-ID196, with each ID increment of 15, are the “Secondary Superelements”. In the right-side column, the cell-ID30 is the replicable “Primary Superelement”, and from cell-ID45 to cell-ID210, with each ID increment of 15, are the “Secondary Superelements”. In the middle section, cell-ID17 is the replicable “Primary Superelement”, and from cell-ID18 to cell-ID209, with each ID increment of 1 in each row and an ID increment of 2 when row number plus 1, a total of thirteen-rows cells, are the “Secondary Superelements”.

For testing “Panel No.4”, the cell-ID1, cell-ID20, cell-ID381, and cell-ID400 are the non-replicable “Primary Superelements”. In the bottom row, cell-ID2 is the replicable “Primary Superelement”. Thus, the “Secondary Superelements” are cell-ID3 to cell-ID19 with an ID increment of 1. In the top row, cell-ID382 is a replicable “Primary Superelement”, and from cell-ID383 to cell-ID399, with each ID increment of 1, are the “Secondary Superelements”. In the left-side column, cell-ID21 is the replicable “Primary Superelement”, and from cell-ID41 to cell-ID361, with each ID increment of 20, are the “Secondary Superelements”. In the right-side column, cell-ID40 is the replicable “Primary Superelement”, and from cell-ID60 to cell-ID380, with each ID increment of 20, are the “Secondary Superelements”. In the middle section, cell-ID22 is the replicable “Primary Superelement”, and from cell-ID23 to cell-ID379, with each ID increment of 1 in each row and an ID increment of 2 when row number plus 1, a total of eighteen-rows cells, are the “Secondary Superelements”.

15.1.3 Eigenvalue Extraction Analyses in Nastran

This subsection displays the eigenfrequencies distribution of different testing panels at the frequency range of $0-250$ Hz. To some extent, eigenfrequencies distribution in this frequency range can show the low-frequency performance of the testing panel. The reason is that the testing panel is more sensible at the given frequency range if more eigenfrequencies are found.

These eigenfrequencies come from corresponding eigenvalue extraction analyses with different numerical methods. And these numerical methods may have different performances in accuracy and efficiency. The following contents illustrate the eigenfrequencies distribution in five subsections of the frequency range of the four testing panels (see **Figure 15.1**). Then, the time consumption of the eigenvalue extraction analyses through the "Ordinary Method", the "ACMS Method", the "List Superelement Method", and the "Image Superelement Method" will be displayed in subsection **15.1.8**.

From **Figure 15.3** to **Figure 15.6**, the corresponding numerical methods are the "Ordinary Method", the "ACMS Method", the "List Superelement Method", and the "Image Superelement Method", respectively. We can see that the eigenfrequencies of the four testing panels extracted by these numerical methods present analogous distribution tendencies except for the "List Superelement Method". The discrepancies of the distribution tendency in **Figure 15.5** indicate that the "List Superelement Method" is not stable in eigenvalue extraction analyses. In contrast, the other three numerical methods are stable and competent in eigenvalue extraction analyses.

Based on the eigenfrequencies distribution tendency, we can find two rules. In the first case, with the panel's size increases, more eigenfrequencies can be found. In the second case, more eigenfrequencies are found in the high-frequency range. Thus, controlling the panel's size can control the eigenfrequencies distribution. Subsequently, in steady-state dynamic analyses, we know that the eigenfrequencies distribution can directly influence the emergence of resonance-antiresonance points (see subsections **15.1.5**, **15.1.6**, and **15.1.7**).

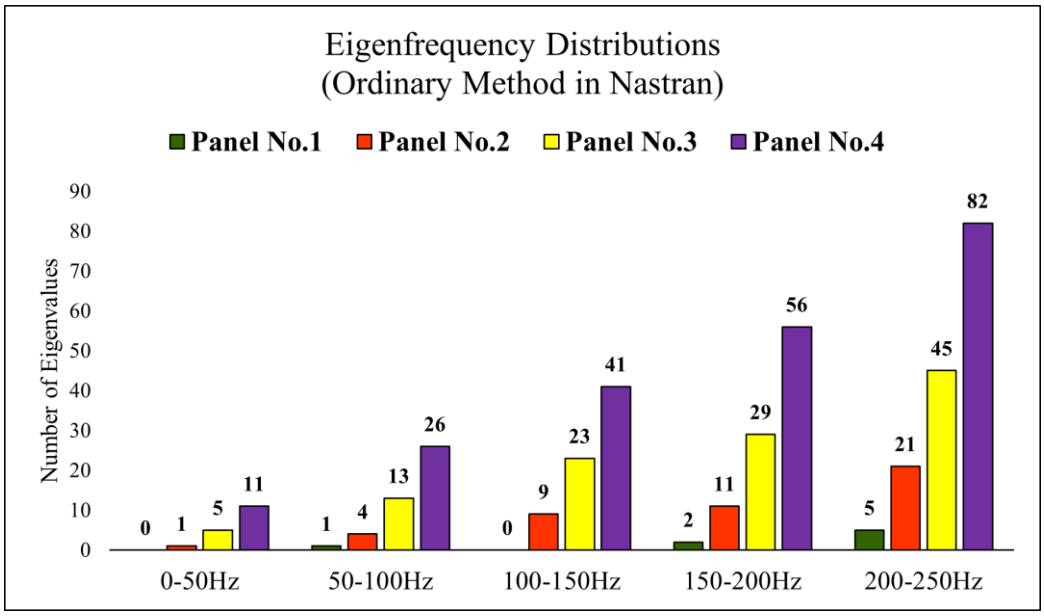


Figure 15.3: Eigenfrequencies distribution of size-controlled panels (Nastran's "Ordinary Method").

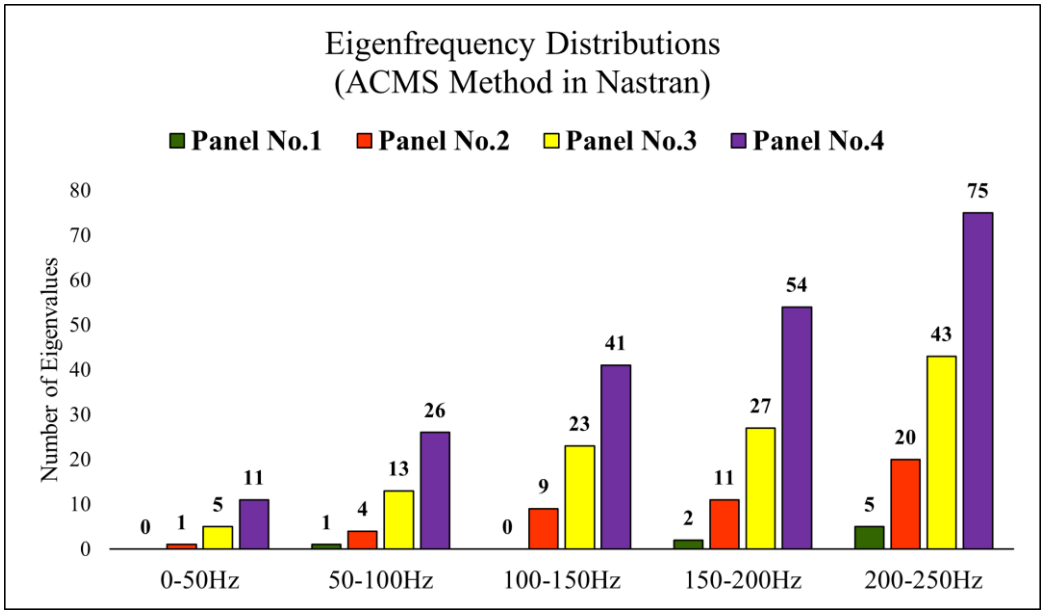


Figure 15.4: Eigenfrequencies distribution of size-controlled panels ("ACMS Method").

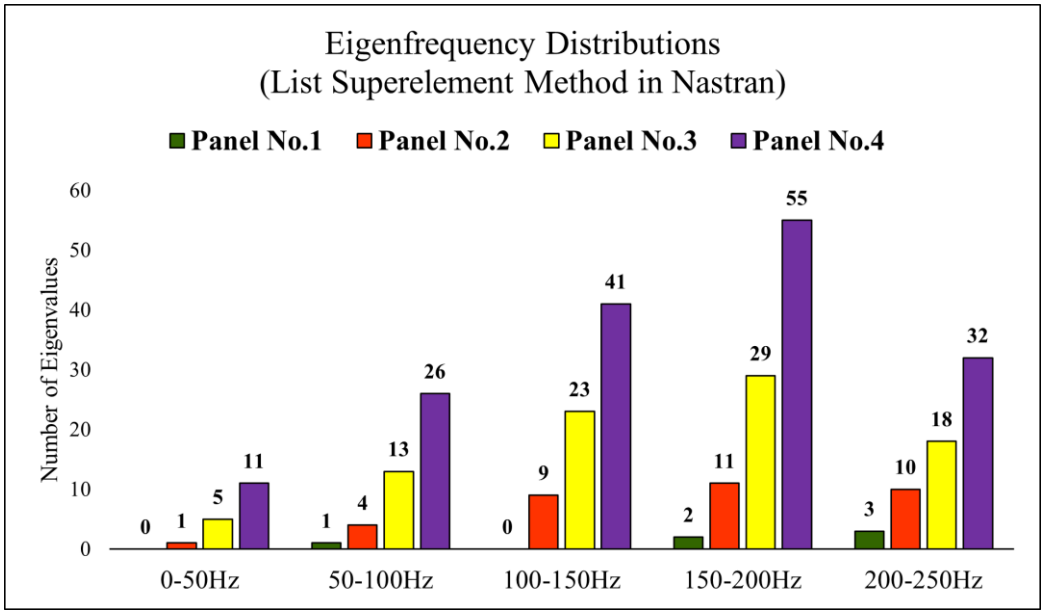


Figure 15.5: Eigenfrequencies distribution of size-controlled panels ("List Superelement Method").

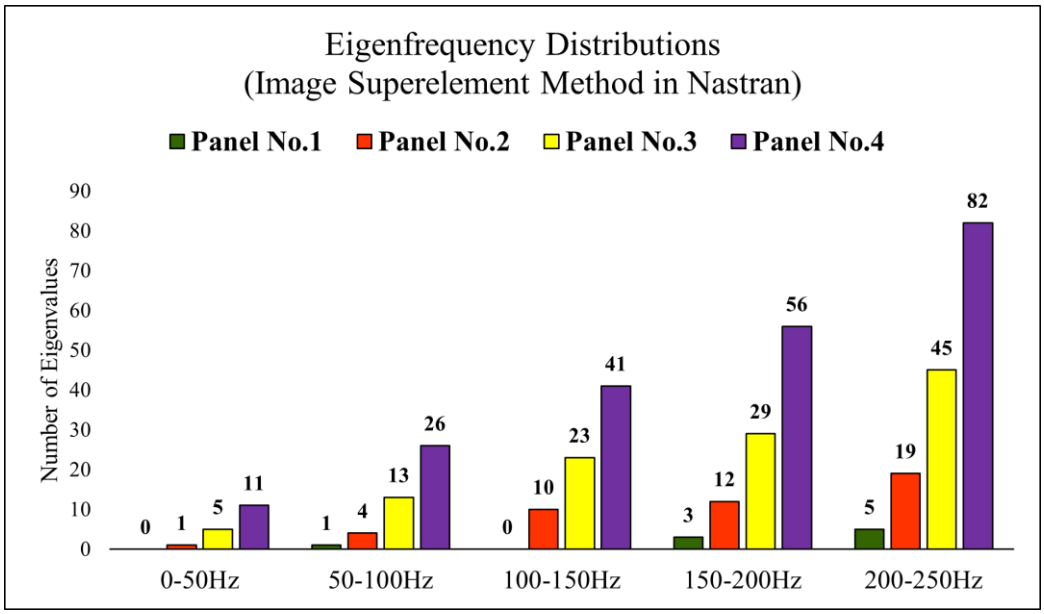


Figure 15.6: Eigenfrequencies distribution of size-controlled panels ("Image Superelement Method").

15.1.4 Steady-State Dynamic Analyses in Nastran

As already known, the testing sample consists of composite materials (melamine foam and aluminium alloy), and the melamine foam has frequency dependency (see subsection **2.2.3**). For defining the frequency-dependent material properties, Nastran supplies two material patterns; one is the "Classic Frequency-Dependent Material" pattern (see subsection **7.2.2**), and the other is the "Viscoelastic Material" pattern (see subsection **7.2.4**). Once different frequency-dependent material patterns are adopted, the processing logic in Nastran is different (see subsections **3.10.4**, **3.10.5**, and **3.10.6**). Due to different processing logic, the "Classic Frequency-Dependent Material" pattern and the "Viscoelastic Material" pattern have their pros and cons (see subsections **7.2.3** and **7.2.5**).

The following tests adopt three material patterns: the "Classic Frequency-Dependent Material" pattern, the "Viscoelastic Material" pattern, and the "Non-Frequency-Dependent Material" pattern. And each pattern can be applied independently to two simulation spaces (the physical and modal spaces). Then, because of the material patterns and the simulation spaces, different constraints are added to the "Ordinary Method", the "ACMS Method", the "List Superelement Method", and the "Image Superelement Method" in steady-state dynamic analysis. **Table 15.1** lists the matching conditions of the four numerical methods, the two simulation spaces, and the three material patterns. As we can see, if a numerical method matches a simulation space and a material pattern, the corresponding intersection box records "YES". If not, the corresponding intersection box records "NO".

As an important parameter in steady-state dynamic analyses, damping needs more concerns. Successful and efficient dynamic analysis needs support from proper numerical

damping models. The proper damping models in different simulation scenarios are different. Chapter **7** has elaborated on the damping models in Nastran that can be used in the steady-state dynamic analyses in subsections **15.1.5** to **15.1.7**. And **Table 15.2** lists the links of corresponding damping model elaborations. Please click the corresponding links in the intersection blanks for more information about the damping models of different simulation scenarios. For proper damping models applied to the aluminium alloy and the melamine foam in different situations, please review subsection **7.5**.

In **Table 15.1** and **Table 15.2**, the abbreviation "M. cfFreq" represents the "Classic Frequency-Dependent Material", the abbreviation "M. veFreq" represents the "Viscoelastic Material", and the "M. NFreq" represents the "Non-Frequency-Dependent Material". Then, the abbreviation "PS" means physical space, and "MS" means modal space.

Table 15.1: Matching conditions in Nastran (simulation space, material patterns, numerical methods).

Numerical Methods	M. cfFreq		M. veFreq		M. NFreq	
	PS	MS	PS	MS	PS	MS
Ordinary Method	YES	YES	YES	NO	YES	YES
ACMS Method	NO	YES	NO	NO	NO	YES
List Superelement Method	NO	NO	YES	NO	YES	YES
Image Superelement Method	NO	NO	YES	NO	YES	YES

Table 15.2: Proper damping in Nastran (simulation space, material patterns, numerical methods).

Numerical Methods	M. cfFreq		M. veFreq		M. NFreq	
	PS	MS	PS	MS	PS	MS
Ordinary Method	**7.2.2**	**7.3.2**	**7.2.4**		**7.2.1**	**7.3.1**
ACMS Method		**7.3.2**				**7.3.1**
List Superelement Method			**7.2.4**		**7.2.1**	**7.3.1**
Image Superelement Method			**7.2.4**		**7.2.1**	**7.3.1**

15.1.5 Relative Accuracy Assessments of the "Ordinary Method" in Steady State Dynamic Analyses

As introduced before, the "Ordinary Method" is the numerical method without adopting superelement or superelement-correlated techniques. This method is much more general than the other three methods (the "ACMS Method", the "List Superelement Method", and the "Image Superelement Method"). It can be used in physical-space-based technical routes or

modal-space-based technical routes. Furthermore, it can treat the "Non-Frequency-Dependent Material" pattern, the "Classic Frequency-Dependent Material" pattern, and the "Viscoelastic Material" pattern. For more information on matching conditions of the "Ordinary Method" in corresponding simulation scenarios, please see **Table 15.1**. In general, the "Ordinary Method" has relatively higher accuracies. In theory, the most accurate simulation strategy is the physical-space-based analyses with the "Ordinary Method". Thus, the testing results from the physical-space-based analyses through the "Ordinary Method" can be seen as a relative baseline for judging the accuracy of the other simulation strategies.

The following contents display the results calculated through the "Ordinary Method", which are frequency response functions of the given response points on four testing metamaterial panels. **Figure 15.7** to **Figure 15.10** are frequency-amplitude figures that show the point results of "Panel No.1 (with 25 cells and indicated in an abbreviation P25)", "Panel No.2 (with 100 cells and indicated in an abbreviation P100)", "Panel No.3 (with 225 cells and indicated in an abbreviation P225)", "Panel No.4 (with 400 cells and indicated in an abbreviation P400)", respectively. We can see that each figure contains two subplots, and the title word "FRF" in each subplot means frequency response function.

Subplot (a) corresponds to frequency-dependent models, and the title word in bracket "M. Freq" means the model consists of frequency-dependent materials. In this subplot, we can see that three simulation strategies are chosen for treating frequency-dependent models. At the same time, subplot (b) corresponds to non-frequency-dependent models, and the title word in bracket "M. NFreq" means the model consists of "Non-Frequency-Dependent Material".

Let us read the legend box at the right-top corner. The abbreviation "P. Response" means the response point. Then, the simulation strategy labels in brackets indicate the panel's type, the material's pattern, the simulation space, and the calculating method. Take "Panel No.1" as an example (see **Figure 15.7**). Because this panel has 25 cells, it will be indicated by the abbreviation "P25". Following the panel-type indicator, the abbreviation "cf" indicates the "Classic Frequency-Dependent Material" pattern, while "ve" indicates the "Viscoelastic Material" pattern. If the panel-type indicator does not follow a specific abbreviation, the testing model adopts the "Non-Frequency-Dependent Material" pattern. After that, the abbreviation "PSFRF" means the physical-space-based frequency response functions, while "MSFRF" means the modal-space-based frequency response functions. Then, at the tail, the abbreviation "OD" means ordinary methods. Therefore, as an example, the task label "P25cf_PSFRF_OD" means the frequency response function comes from a physical-space-based dynamic analysis using the "Ordinary Method", and the objective model is the "Panel No.1" with 25 cells which adopts the "Classic Frequency-Dependent Material" pattern. Then, the other task labels in the legend box can be interpreted like this. And the rules are congruent from **Figure 15.7** to **Figure 15.10**. Then, we can see the harmonic response functions of the structural response point on corresponding panels. The Y-axis is in the log scale, while the X-axis is in the linear scale.

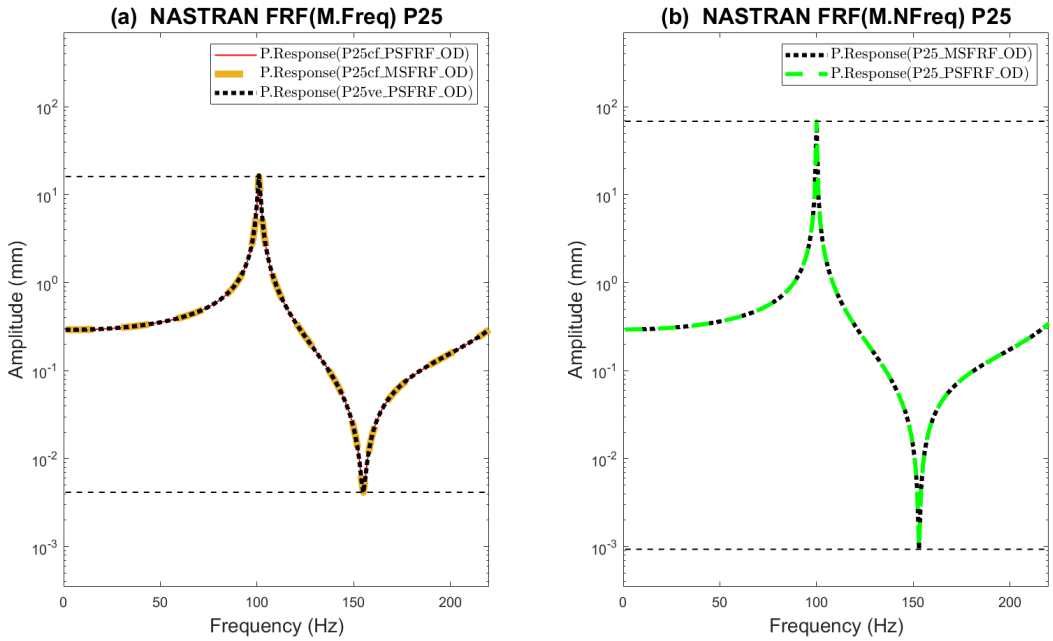


Figure 15.7: FRF of structural response point ("Panel No.1", Nastran, "Ordinary Method").

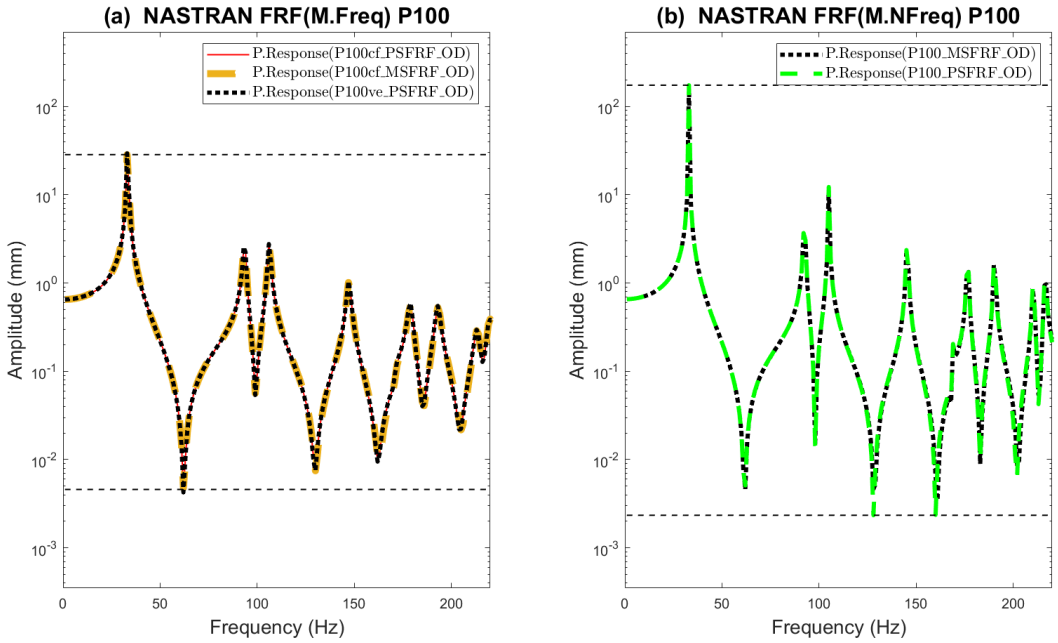


Figure 15.8: FRF of structural response point ("Panel No.2", Nastran, "Ordinary Method").

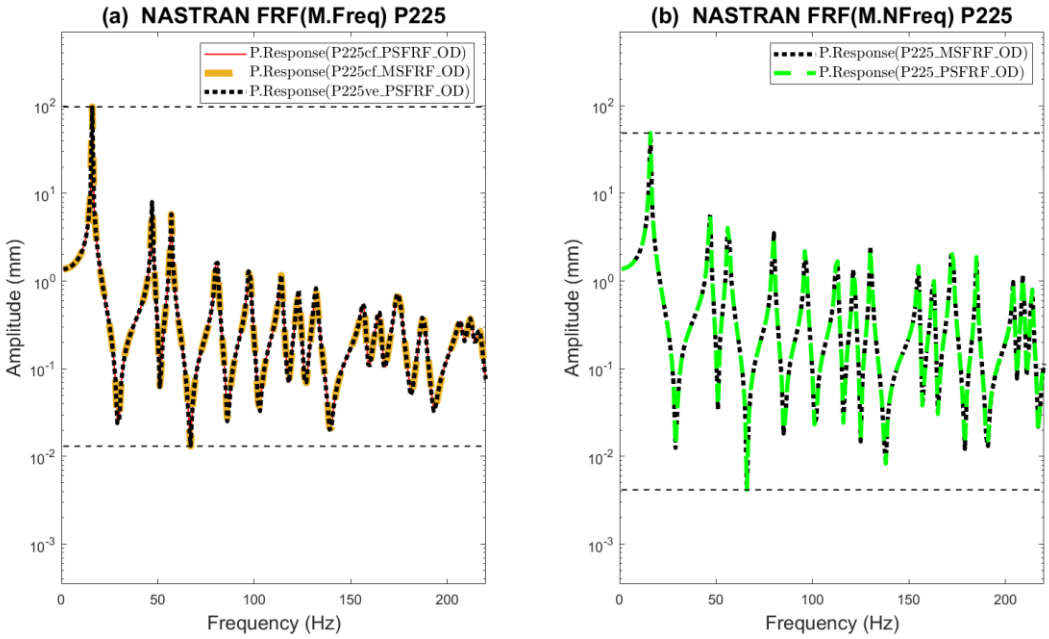


Figure 15.9: FRF of structural response point ("Panel No.3", Nastran, "Ordinary Method").

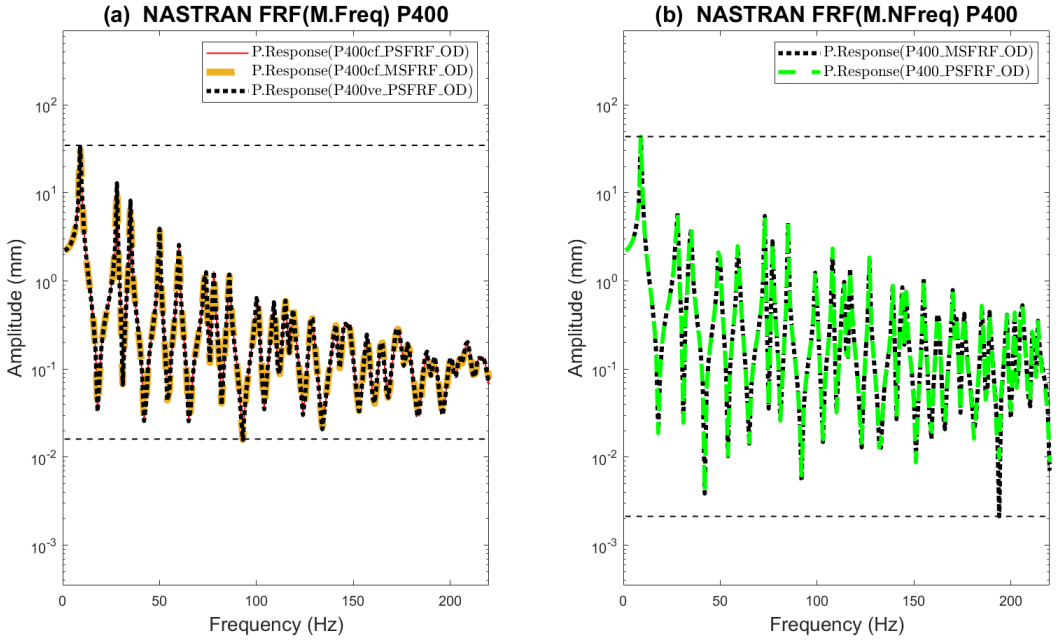


Figure 15.10: FRF of structural response point ("Panel No.4", Nastran, "Ordinary Method").

Let us see the information from the four figures. With the panel's size increase, more resonant and anti-resonant points emerge. This phenomenon is in concert with the eigenfrequencies distribution in subsection **15.1.3**. Then, let us concern the frequency-dependent models in subplots (a) and the non-frequency-dependent models in subplots (b). The peak values of the response amplitudes of the non-frequency-dependent models are higher than the response amplitudes of the frequency-dependent models. The reason is that the structural damping value of the melamine foam in the metamaterial increases with the frequency value increases (see **Figure 2.11**). Thus, the frequency-dependent models have stronger damping effects than the non-frequency-dependent models.

Finally, let us talk about the accuracy of the "Ordinary Method". The corresponding testing results from different simulation strategies are perfectly matched, whether in physical or modal space, no matter whether the model adopts the "Classic Frequency-Dependent Material" pattern or "Viscoelastic Material" pattern. In other words, changing simulation spaces or frequency-dependent material patterns does not influence the "Ordinary Method" accuracy. Therefore, when treating frequency-dependent models, the results of the physical-space-based analysis with the "Classic Frequency-Dependent Material" or "Viscoelastic Material" patterns using the "Ordinary Method" can be the relative baseline. And this baseline will be used in subsection **15.1.6**. In the same way, when treating non-frequency dependent models, the results calculated by the physical-space-based analysis with the "Non-Frequency-Dependent Material" pattern in using the "Ordinary Method" can be the relative baseline. And this baseline will be used in subsection **15.1.7**.

15.1.6 Relative Accuracy Assessments of The Two Frequency-Dependent Material Patterns in Steady State Dynamic Analyses

The following contents cover simulation scenarios based on two frequency-dependent material patterns. As introduced before, the frequency-dependent models can be built by the "Classic Frequency-Dependent Material" pattern and the "Viscoelastic Material" pattern. And proper simulation strategies must obey the matching conditions of the simulation space, the material patterns, and the numerical method (see **Table 15.1**).

We can see that three simulation strategies support the "Classic Frequency-Dependent Material" patterns. The first is the modal-space-based analysis using "ACMS Method". Then the second is the modal-space-based analysis using the "Ordinary Method". And the third is the physical-space-based analysis using the "Ordinary Method", and results from this technical route are the relative baseline. At the same time, three simulation strategies support the "Viscoelastic Material" patterns. The first is the physical-space-based analysis using the "Image Superelement Method". The second is the physical-space-based analysis using the "List Superelement Method". Then, the third is the physical-space-based analysis using the "Ordinary Method", and results from this technical route are the relative baseline.

The following contents display the results calculated through corresponding simulation strategies, and these results are frequency response functions of the given response points on four testing metamaterial panels. From **Figure 15.11** to **Figure 15.14** are frequency-amplitude figures that show the point results of "Panel No.1 (with 25 cells and indicated in an abbreviation P25)", "Panel No.2 (with 100 cells and indicated in an abbreviation P100)", "Panel No.3 (with 225 cells and indicated in an abbreviation P225)", "Panel No.4 (with 400 cells and indicated in

an abbreviation P400)", respectively. We can see that each figure contains two subplots, and the title word "FRF" in each subplot means the "frequency response function".

The title word in bracket "M. cfFreq" of the subplot (a) means the model adopts the "Classic Frequency-Dependent Material" pattern. At the same time, the title word in bracket "M. veFreq" of subplot (b) means the model adopts the "Viscoelastic Material" pattern. Let us read the legend box at the right-top corner of subplot (a) and subplot (b). The abbreviation "P. Response" means response point. Then, the task labels in brackets indicate panel types, material patterns, simulation spaces, and calculation methods.

Let us take "Panel No.1" as an example (see **Figure 15.11**) to illustrate the meaning of task labels. Then the meaning of the task labels of "Panel No.2", "Panel No.3", and the "Panel No.4" are the same. For more information about the definition rules of the task labels, please review subsection **15.1.5**.

In the legend box of subplot (a), the label "P25cf_MSFRF_ACMS" means the frequency response function comes from a modal-space-based dynamic analysis using the "ACMS Method". And the objective model is "Panel No.1" with 25 cells. This panel adopts the "Classic Frequency-Dependent Material" pattern. Then, the label "P25cf_MSFRF_OD" means the frequency response function comes from a modal-space-based dynamic analysis using the "Ordinary Method". And the objective model is "Panel No.1" with 25 cells. This panel adopts the "Classic Frequency-Dependent Material" pattern. After that, the label "P25cf_PSFRF_OD" means the frequency response function comes from a physical-space-based dynamic analysis using the "Ordinary Method". And the objective model is "Panel No.1" with 25 cells. This panel adopts the "Classic Frequency-Dependent Material" pattern.

At the same time, In the legend box of subplot (b), the label "P25ve_PSFRF_CSUPER" means the frequency response function comes from a physical-space-based dynamic analysis using the "Image Superelement Method". And the objective model is "Panel No.1" with 25 cells. This panel adopts the "Viscoelastic Material" pattern. Then, the label "P25ve_PSFRF_OD" means the frequency response function comes from a physical-space-based dynamic analysis using the "Ordinary Method". And the objective model is "Panel No.1" with 25 cells. This panel adopts the "Viscoelastic Material" pattern. At last, the label "P25ve_PSFRF_SESET" means the frequency response function comes from a physical-space-based dynamic analysis using the "List Superelement Method". And the objective model is "Panel No.1" with 25 cells. This panel adopts the "Viscoelastic Material" pattern.

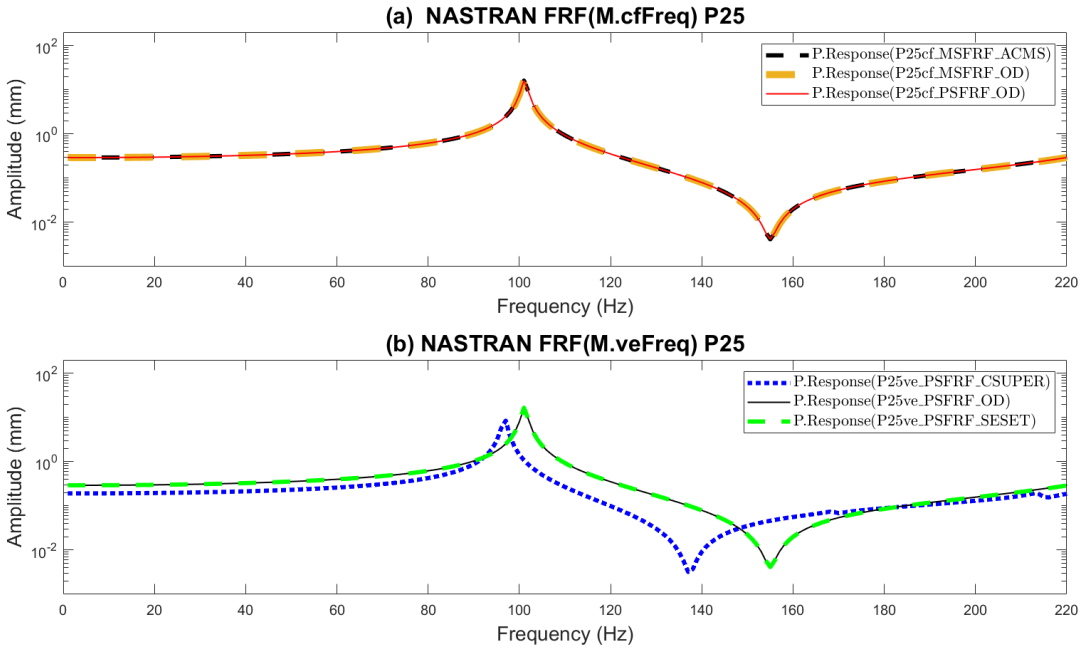


Figure 15.11: FRF of structural response point ("Panel No.1", Nastran, two frequency-dependent materials).

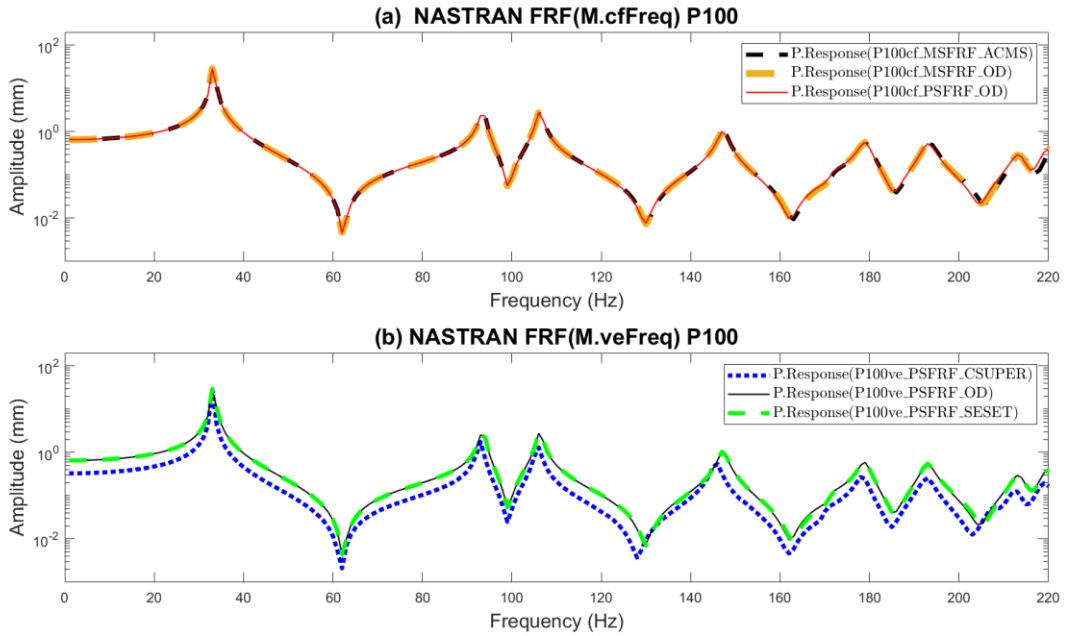


Figure 15.12: FRF of structural response point ("Panel No.2", Nastran, two frequency-dependent materials).

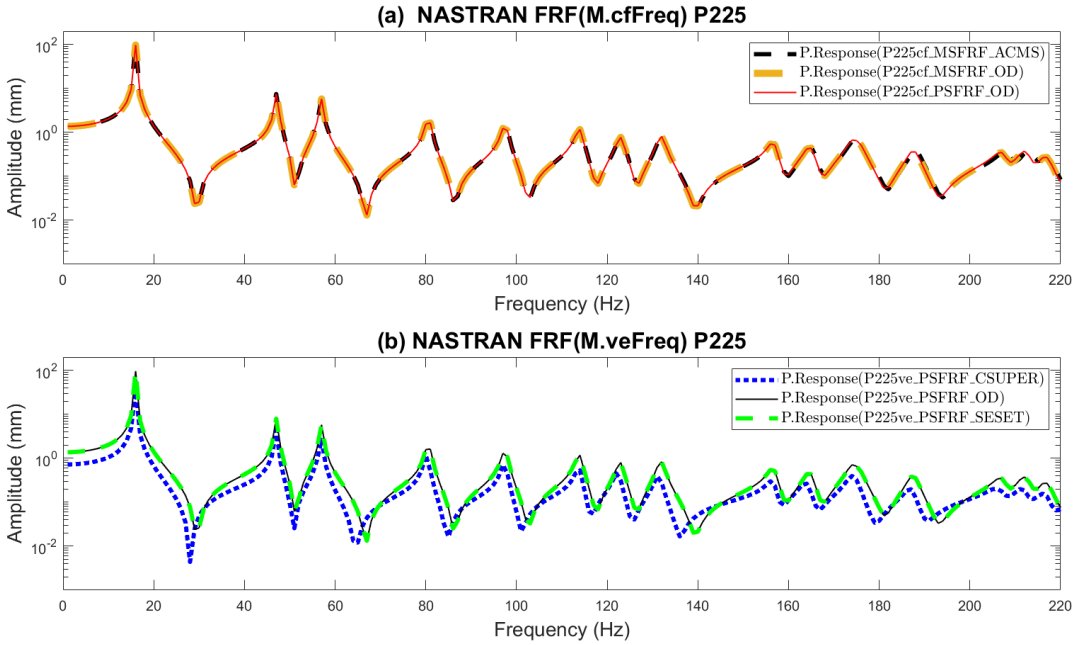


Figure 15.13: FRF of structural response point ("Panel No.3", Nastran, two frequency-dependent materials).

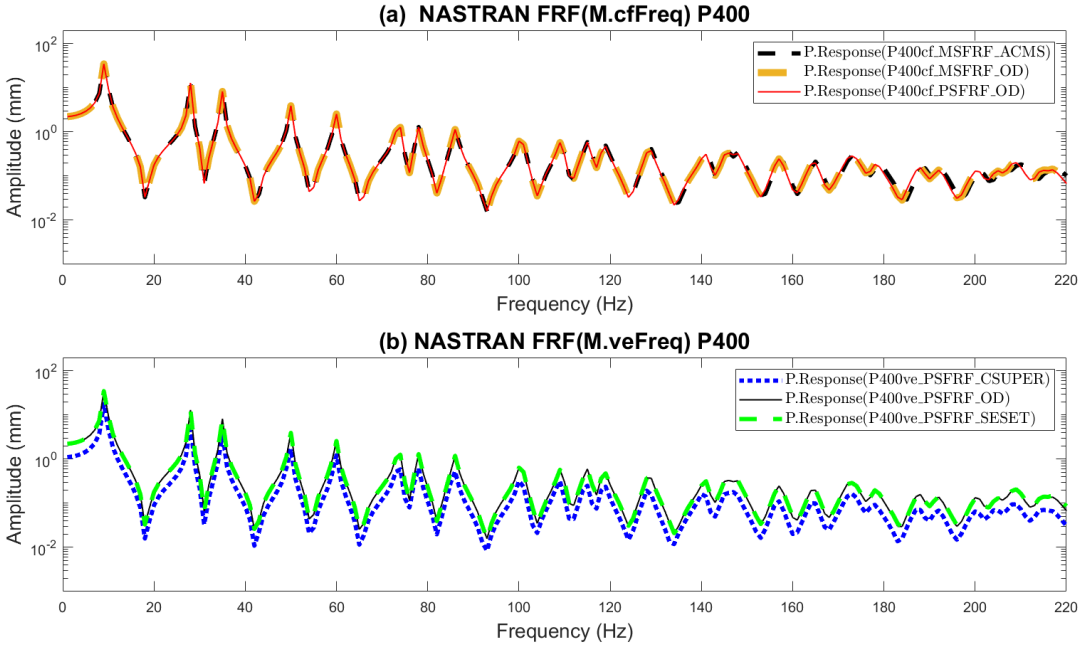


Figure 15.14: FRF of structural response point ("Panel No.4", Nastran, two frequency-dependent materials).

Based on the results in subplots (a) from **Figure 15.11** to **Figure 15.14**, we can see that the three available simulation strategies can accurately treat the models with the "Classic Frequency-Dependent Material" pattern. The results calculated from the modal-space-based analyses with the "ACMS Method" and with the "Ordinary Method" are perfectly matched with the relative-baseline results calculated from the physical-space-based analyses with the "Ordinary Method".

At the same time, based on the results in subplots (b) from **Figure 15.11** to **Figure 15.14**, we can see that the results from physical-space-based analyses using the "List Superelement Method" and the "Ordinary Method" can perfectly match together. However, the results from physical-space-based analyses using the "Image Superelement Method" cannot perfectly match them. In other words, the physical-space-based analyses with the "List Superelement Method" and with the "Ordinary Method" can accurately treat models with "Viscoelastic Material" patterns, but with the "Image Superelement Method" will lose some accuracy.

15.1.7 Relative Accuracy Assessments Modal-Space-Based and Physical-Space-Based Steady-State Dynamic Analyses

Subsection **15.1.6** has covered simulation scenarios based on two frequency-dependent material patterns. Then, the following contents in this section will cover the simulation scenarios based on the physical space and the modal space with "Non-Frequency-Dependent Material" patterns. Proper simulation strategies must obey the matching conditions of the simulation space, the material patterns, and the numerical method (see **Table 15.1**). Thus, we can see that four numerical methods are available in modal-space-based technical routes with the "Non-Frequency-Dependent Material" pattern. These four methods are the "ACMS Method", the "Image Superelement Method", the "Ordinary Method", and the "List Superelement Method". At the same time, three methods are available in physical-space-based technical routes with the "Non-Frequency-Dependent Material" pattern. These three methods are the "Image Superelement Method", the "Ordinary Method", and the "List Superelement Method".

The following contents display the results calculated through corresponding available simulation strategies, and these results are frequency response functions of given response points on four testing metamaterial panels. From **Figure 15.15** to **Figure 15.18** are frequency-amplitude figures that show the point results of "Panel No.1 (with 25 cells and indicated in an abbreviation P25)", "Panel No.2 (with 100 cells and indicated in an abbreviation P100)", "Panel No.3 (with 225 cells and indicated in an abbreviation P225)", "Panel No.4 (with 400 cells and indicated in an abbreviation P400)", respectively. We can see that each figure contains two subplots, and the title word "FRF" in each subplot means frequency response function.

The title word in brackets, "M. NFreq", of subplots (a) and (b) means the model adopts the "Non-Frequency-Dependent Material" patterns. Let us read the legend box at the right-top corner of the subplots (a) and (b). The abbreviation "P. Response" means response point. Then, the task labels in brackets indicate panel types, material patterns, simulation spaces, and calculation methods.

Let us take "Panel No.1" as an example (see **Figure 15.15**) to illustrate the meaning of task labels. Then the meaning of the task labels of "Panel No.2", "Panel No.3", and the "Panel

No.4" are the same. For more information about the definition rules of the task labels, please review subsection **15.1.5**.

In the legend box of subplot (a), the label "P25MSFRF_ACMS" means the frequency response function comes from a modal-space-based dynamic analysis using the "ACMS Method". And the objective model is "Panel No.1" with 25 cells. This panel adopts the "Non-Frequency-Dependent Material" pattern. Then, the label "P25MSFRF_CSUPER" means the frequency response function comes from a modal-space-based dynamic analysis using the "Image Superelement Method". And the objective model is "Panel No.1" with 25 cells. This panel adopts the "Non-Frequency-Dependent Material" pattern. Then, the label "P25MSFRF_OD" means the frequency response function comes from a modal-space-based dynamic analysis using the "Ordinary Method". And the objective model is "Panel No.1" with 25 cells. This panel adopts the "Non-Frequency-Dependent Material" pattern. After that, the label "P25MSFRF_SESET" means the frequency response function comes from a modal-space-based dynamic analysis using the "List Superelement Method". And the objective model is "Panel No.1" with 25 cells. This panel adopts the "Non-Frequency-Dependent Material" pattern.

At the same time, in the legend box of subplot (b), the label "P25PSFRF_CSUPER" means the frequency response function comes from a physical-space-based dynamic analysis with the "Image Superelement Method". And the objective model is "Panel No.1" with 25 cells. This panel adopts the "Non-Frequency-Dependent Material" pattern. Then, the label "P25PSFRF_OD" means the frequency response function comes from a physical-space-based dynamic analysis using the "Ordinary Method". And the objective model is "Panel No.1" with 25 cells. This panel adopts the "Non-Frequency-Dependent Material" pattern. At last, the label "P25PSFRF_SESET" means the frequency response function comes from a physical-space-based dynamic analysis using the "List Superelement Method". And the objective model is "Panel No.1" with 25 cells. This panel adopts the "Non-Frequency-Dependent Material" pattern.

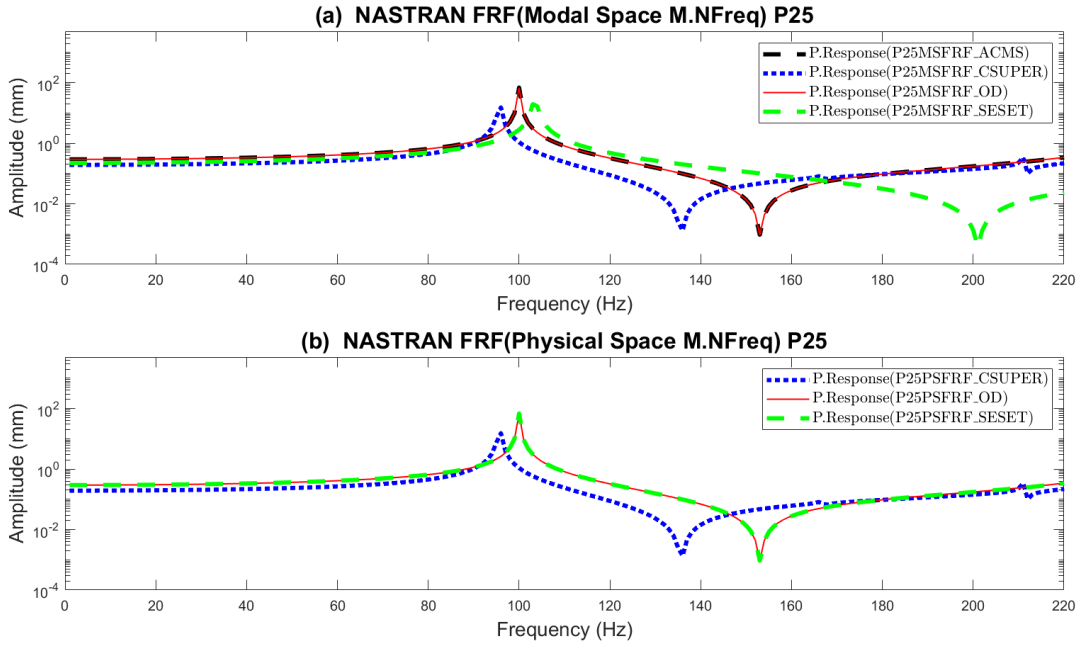


Figure 15.15: FRF of structural response point ("Panel No.1", Nastran, two simulation spaces).

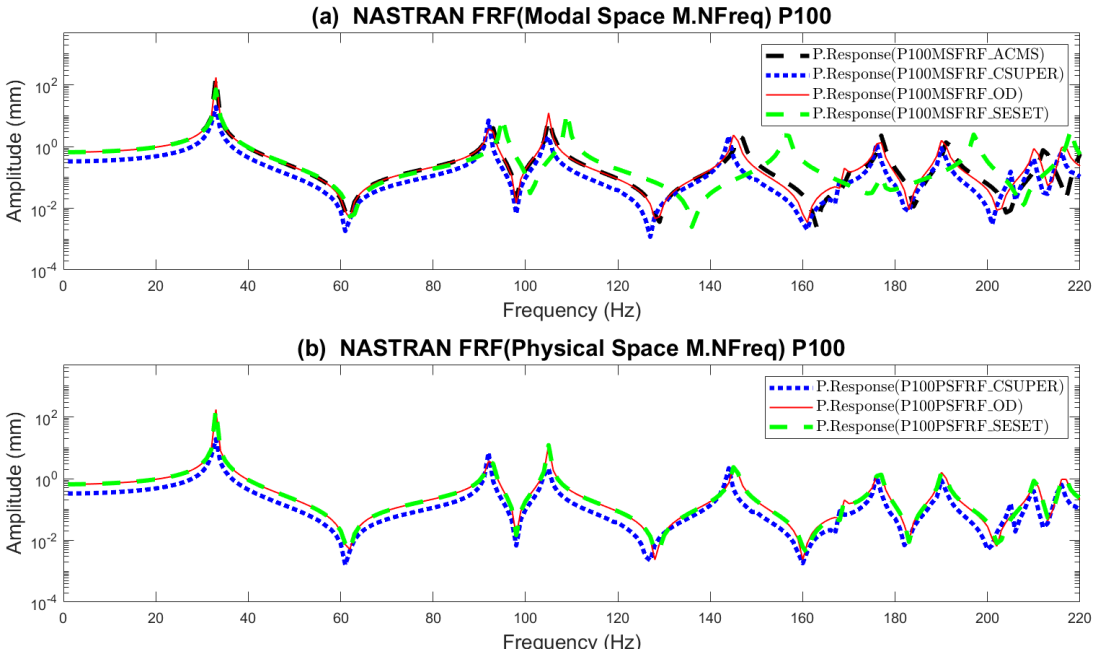


Figure 15.16: FRF of structural response point ("Panel No.2", Nastran, two simulation spaces).

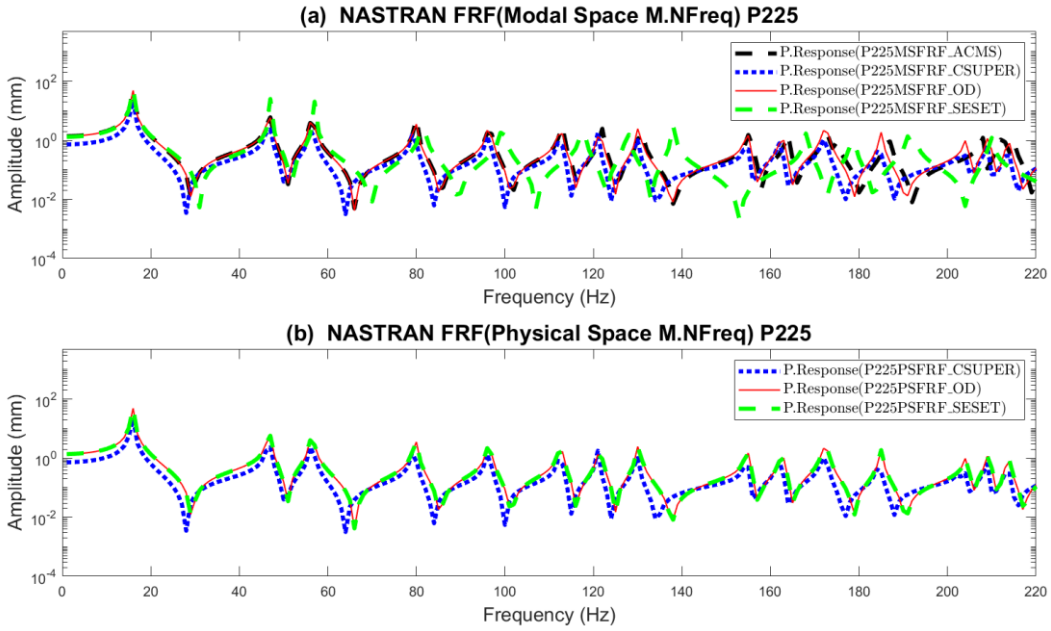


Figure 15.17: FRF of structural response point ("Panel No.3", Nastran, two simulation spaces).

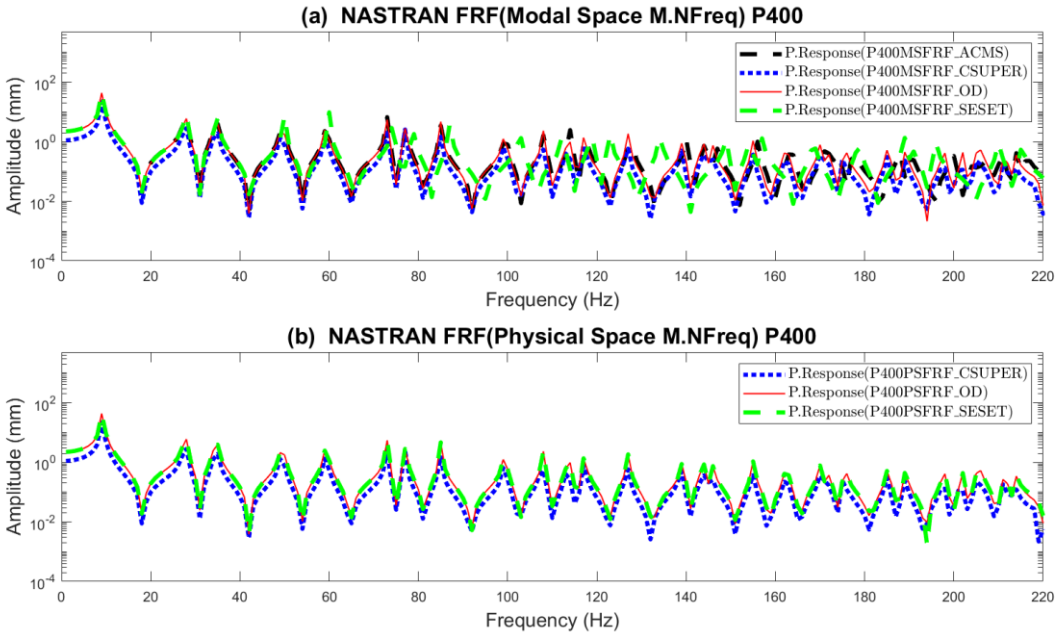


Figure 15.18: FRF of structural response point ("Panel No.4", Nastran, two simulation spaces).

Based on the results of non-frequency-dependent models from modal-space-based analyses in subplots (a) from **Figure 15.15** to **Figure 15.18**, we can see that the results calculated through the "ACMS Method" and the "Ordinary Method" are perfectly matched with each other. In contrast, the results calculated through the "Image Superelement Method" and the "List Superelement Method" are partially matched with the results by the "ACMS Method" and the "Ordinary Method". The results from the modal-space-based analyses with the "Ordinary Method" can be seen as the relative baselines (see subsection **15.1.5**). Thus, the "ACMS Method" and the "Ordinary Method" in modal-space-based analyses are relatively accurate in treating non-frequency-dependent models. In comparison, the "List Superelement Method" and the "Image Superelement Method" are relatively inaccurate.

At the same time, based on the results of non-frequency-dependent models from physical-space-based analyses in subplots (b) from **Figure 15.15** to **Figure 15.18**, we can see that the results calculated through the "List Superelement Method" and the "Ordinary Method" perfectly match each other. In contrast, the results calculated through the "Image Superelement Method" and the "Ordinary Method" partially match each other. As we know, the results from the physical-space-based analyses with the "Ordinary Method" can be seen as the relative baselines (see subsection **15.1.5**). Thus, the "List Superelement Method" and the "Ordinary Method" in physical-space-based analyses are relatively accurate in treating non-frequency-dependent models. In comparison, the "Image Superelement Method" are relatively inaccurate.

15.1.8 Section Conclusions

Let us have a brief review of the contents in section **15.1**. Subsections **15.1.1** and **15.1.2** are two compensative descriptions of the settings of the "List Superelement Method" and the "Image Superelement Method".

Subsection **15.1.3** illustrates the testing panels' dynamic performances regarding their eigenfrequencies distribution. And simulations in this section will evaluate Nastran's available numerical methods in eigenvalue extraction analyses. Based on the results in subsection **15.1.3**, the "List Superelement Method" performance is unstable in eigenvalue extraction analyses. And this phenomenon is in concert with the results listed in **Table 15.3**.

Besides stability, efficiency is another important factor that must be considered when choosing the best numerical method for eigenvalue extraction analyses. Based on the results in **Table 15.4**, the most efficient method in eigenvalue extraction analyses is the "ACMS Method". With the model's DOFs increasing, the "ACMS Method" superiority is more obvious regarding the other three numerical methods. In contrast, the "List Superelement Method" is the most inefficient numerical method, and the "Image Superelement Method" is the second most inefficient numerical method.

Table 15.3: The total number of extracted eigenmodes through Nastran's four numerical methods (four size-controlled panels, 0–250Hz).

	Panel No.1	Panel No.2	Panel No.3	Panel No.4
Ordinary Method	8	46	115	216
ACMS Method	8	45	111	207
List Superelement Method	6	35	88	165
Image Superelement Method	9	46	115	216
Number of Mesh Grids	3605	14205	31805	56405

Table 15.4: The time consumption (second) of eigenvalue extractions through Nastran's four numerical methods (four size-controlled panels, 0–250Hz).

	Panel No.1	Panel No.2	Panel No.3	Panel No.4
Ordinary Method	1.7s	4.3s	15.9s	50.4s
ACMS Method	2.5s	3.7s	8.4s	16.5s
List Superelement Method	3.8s	28.2s	119.2s	268.8s
Image Superelement Method	2.5s	10.1s	37.6s	117.9s
Number of Mesh Grids	3605	14205	31805	56405

Subsection **15.1.4** is a technical description of how to set available simulation strategies. The theoretical background of this subsection is chapters **3** and **7**. They supply theoretical analyses and arguments. Then, subsection **15.1.5** is dedicated to evaluating the baseline simulation strategies. And the results from these baseline simulation strategies can be the relative references of the results from other simulation strategies. Then, regarding the baseline results verified in subsection **15.1.5**, subsection **15.1.6** discusses the relative accuracy of the four numerical methods in treating models with two frequency-dependent material patterns. And the control factors in subsection **15.1.6** is the frequency-dependent material patterns. At the same time, subsection **15.1.7** discusses the relative accuracy of the four numerical methods applied to different simulation spaces in treating non-frequency-dependent models. And the control parameters in subsection **15.1.7** is the simulation spaces.

In the following contents, based on the testing results from subsections **15.1.5**, **15.1.6**, and **15.1.7**, **Table 15.5** lists the grades of the relative accuracy of the four numerical methods in corresponding simulation scenarios. The accuracy grades from the lowest to the highest are denoted with numbers 1 to 5. And if the accuracy grade is higher than 4, this method is recommended.

Table 15.5: The accuracy grades of Nastran's simulation strategies in steady-state dynamic analyses.

Numerical Methods	M. cfFreq		M. veFreq		M. NFreq	
	PS	MS	PS	MS	PS	MS
Ordinary Method	5	5	5		5	5
ACMS Method		5				5
List Superelement Method			4		4	2
Image Superelement Method			3		3	3

The tables from **Table 15.6** to **Table 15.9**, corresponding to "Panel No.1" to "Panel No.4", list the time consumption of the four numerical methods in corresponding simulation scenarios based on steady-state dynamic analyses. And the time consumption is an indicator of the simulation efficiency.

Table 15.6: The time (second) consumption of Nastran's simulation strategies in steady-state dynamic analyses ("Panel No.1", the number of mesh grids is 3605, the number of excitation frequencies is 220).

Numerical Methods	M. cfFreq		M. veFreq		M. NFreq	
	PS	MS	PS	MS	PS	MS
Ordinary Method	52.7s	6.2s	55.2s		47.4s	2.0s
ACMS Method		5.9s				2.7s
List Superelement Method			4.5s		4.3s	2.8s
Image Superelement Method			40.8s		34.9s	2.7s

Table 15.7: The time (second) consumption of Nastran's simulation strategies in steady-state dynamic analyses ("Panel No.2", the number of mesh grids is 14205, the number of excitation frequencies is 220).

Numerical Methods	M. cfFreq		M. veFreq		M. NFreq	
	PS	MS	PS	MS	PS	MS
Ordinary Method	253.2s	27.6s	249.5s		221.7s	6.1s
ACMS Method		29.5s				4.4s
List Superelement Method			44.7s		26.6s	13.5s
Image Superelement Method			265.3s		226.6s	11.7s

Table 15.8: The time (second) consumption of Nastran's simulation strategies in steady-state dynamic analyses ("Panel No.3", the number of mesh grids is 31805, the number of excitation frequencies is 220).

Numerical Methods	M. cfFreq		M. veFreq		M. NFreq	
	PS	MS	PS	MS	PS	MS
Ordinary Method	666.7s	126.9s	587.1s		523.3s	20.1s
ACMS Method		136.1s				11.3s
List Superelement Method			141.8s		131.9s	76.9s
Image Superelement Method			686.7s		578.2s	45.6s

Table 15.9: The time (second) consumption of Nastran's simulation strategies in steady-state dynamic analyses ("Panel No.4", the number of mesh grids is 56405, the number of excitation frequencies is 220).

Numerical Methods	M. cfFreq		M. veFreq		M. NFreq	
	PS	MS	PS	MS	PS	MS
Ordinary Method	1658.1s	397.9s	1255.7s		1111.6s	65.5s
ACMS Method		430.9s				22.2s
List Superelement Method			320.0s		293.0s	166.1s
Image Superelement Method			1373.6s		1186.8s	129.5s

Looking at the four tables, we can see that the efficiency of modal-space-based analyses is much higher than the physical-space-based analyses. We need to take more attention to the data in **Table 15.9**. The reason is that "Panel No.4" has the highest number of mesh grids that induce the heaviest computation load. Thus, the advantages of some efficient numerical methods in modal space become more significant, such as the "ACMS Method" in treating non-frequency-dependent models. At the same time, the degradation of some efficient numerical methods in physical space is enlarged, such as the "List Superelement Method" in treating viscoelastic or non-frequency-dependent models.

Let us concern the physical-space-based analyses in the first place. If models adopt the "Classic Frequency-Dependent Material" pattern in physical space, the "Ordinary Method" is the only choice. Please see the first column in the data section of **Table 15.9**. At the same time, if models adopt the "Viscoelastic Material" pattern, the "Ordinary Method", the "List Superelement Method", and the "Image Superelement Method" are available. Please see the third column in the data section of **Table 15.9**, where the "List Superelement Method" is most efficient. Then, if the model adopts the "Non-Frequency-Dependent Material" pattern, the

"Ordinary Method", the "List Superelement Method", and the "Image Superelement Method" are available. Please see the fifth column in the data section of **Table 15.9**, where the "List Superelement Method" is most efficient.

Therefore, the "List Superelement Method" is most efficient in physical-space-based dynamic analyses. In contrast, the "Image Superelement Method" does not expose supposed superiorities. This conclusion partially supports the theoretical arguments in subsections **3.5.2** and **3.10.3**. The conclusion convinces that the reduced model by the "List Superelement Method" can increase efficiency, and the increased efficiency can cover the cost of superelement generations. Although, the reduced model and the matrix-copying operation by the "Image Superelement Method" can theoretically increase efficiency. However, the process management and the data recovery of the "Image Superelement Method" cost more.

Let us concern the modal-space-based analyses in the second place. In modal space, if models adopt the "Classic Frequency-Dependent Material" pattern, the "Ordinary Method" and the "ACMS Method" are available. Please see the second column in the data section of **Table 15.9**. We can perceive that the "Ordinary Method" is more efficient than the "ACMS Method". In contrast, if models adopt the "Non-Frequency-Dependent Material" pattern, the "Ordinary Method", the "ACMS Method", the "List Superelement Method", and the "Image Superelement Method" are available. Please see the sixth column in the data section of **Table 15.9**. We can perceive that the "ACMS Method" is the most efficient, and the "List Superelement Method" is the most inefficient.

Therefore, the conclusion is that the "Ordinary Method" is the best choice in modal-space-based dynamic analyses treating the frequency-dependent model. At the same time, the "ACMS Method" is the best choice in modal-space-based analyses treating the non-frequency-dependent model. In contrast, the "List Superelement Method" is inefficient in modal-space-based analyses. This conclusion partially supports the theoretical arguments in subsections **3.5.2** and **3.10.3**. The "Classic Frequency-Dependent Material" pattern will activate the model rebuilding mechanism. The superelement correlated techniques always follow the model-building process in the simulation "Phase I" (see subsection **3.4.3**, **Figure 3.20** in subsection **3.10.4**, and subsection **3.10.5**). Thus, the model rebuilding operations will repeat the superelement correlated processing. It is why the "ACMS Method" is not the most efficient method in modal-space-based dynamic analyses to treat frequency-dependent models.

Let us concern the three material patterns in the third place. The "Ordinary Method" is an example. Please see the first row in the data section of **Table 15.9**. The "Viscoelastic Material" pattern is more efficient for frequency-dependent models than the "Classic Frequency-Dependent Material" pattern in physical-space-based steady-state dynamic analyses. This conclusion supports the theoretical arguments in subsections **3.10.4** to **3.10.6** and **7.2.2** to **7.2.5**. The processing time for the "Non-Frequency-Dependent Material" pattern is less than that for frequency-dependent materials since they do not activate the model rebuilding mechanism or matrix updating mechanism.

Finally, considering the efficiency and accuracy, the best simulation strategy for testing the dynamic performance of the metamaterial samples is the modal-space-based steady-state dynamic analyses with the "Ordinary Method". And models must adopt the "Classic Frequency-Dependent Material" pattern to build their frequency dependency. At the same time, the best numerical method for eigenvalue extraction analyses is the "ACMS Method". And Nastran's "ACMS Method" will be adopted in chapter **16** for eigenvalue extraction analyses of

panels with different aluminium inclusions and in chapter **17** with Actran for modal-space-based vibro-acoustic analyses.

15.2 Relative Accuracy and Efficiency Benchmarks of Abaqus

The efficiency benchmarks of Abaqus will assess the performance of two candidate numerical methods in eigenvalue extraction and steady-state dynamic analyses. They are the "Ordinary Method" and the "AMS Method". In eigenvalue extraction analyses, the frequency dependency of the model is inactive, and the damping effect is ignored. However, in steady-state dynamic analysis, the frequency dependency of the orthotropic melamine foam becomes an obstacle. As already known, Abaqus cannot set the frequency dependency to orthotropic materials (see subsection **4.3.4**). Thus, the steady-state dynamic analysis in Abaqus omits the frequency dependency. Then, to some extent, the accuracy of the results will be impacted.

15.2.1 Eigenvalue Extraction Analyses in Abaqus

This section displays the eigenfrequencies distribution of different testing panels at the frequency range of $0-250$ Hz. And these eigenfrequencies come from corresponding eigenvalue extraction analyses with different numerical methods. As one of the concerning points, these numerical methods may present different performances in accuracy and efficiency. The following contents illustrate the eigenfrequency distributions in five subsections of the frequency range of the four testing panels. Then, the time consumption of the eigenvalue extraction analyses through the "Ordinary Method" and "AMS Method" will be displayed in subsection **15.2.4**.

Figure 15.19 and **Figure 15.20** correspond to the "Ordinary Method" and the "AMS Method", respectively. We can see that the eigenfrequencies distribution of the four testing panels presents an analogous tendency. And the two numerical methods from Abaqus are stable in eigenvalue extraction analyses. Then, in steady-state dynamic analyses, the eigenfrequencies distribution can directly influence the emergence of resonant and anti-resonant points (see **Figure 15.21**).

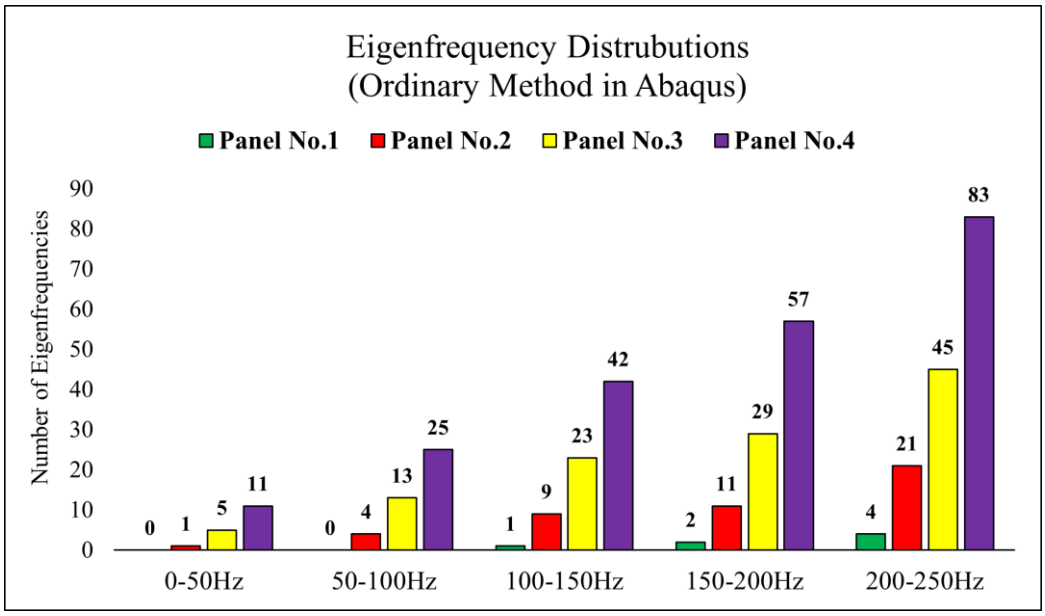


Figure 15.19: Eigenfrequencies distribution of size-controlled panels (Abaqus's "Ordinary Method").

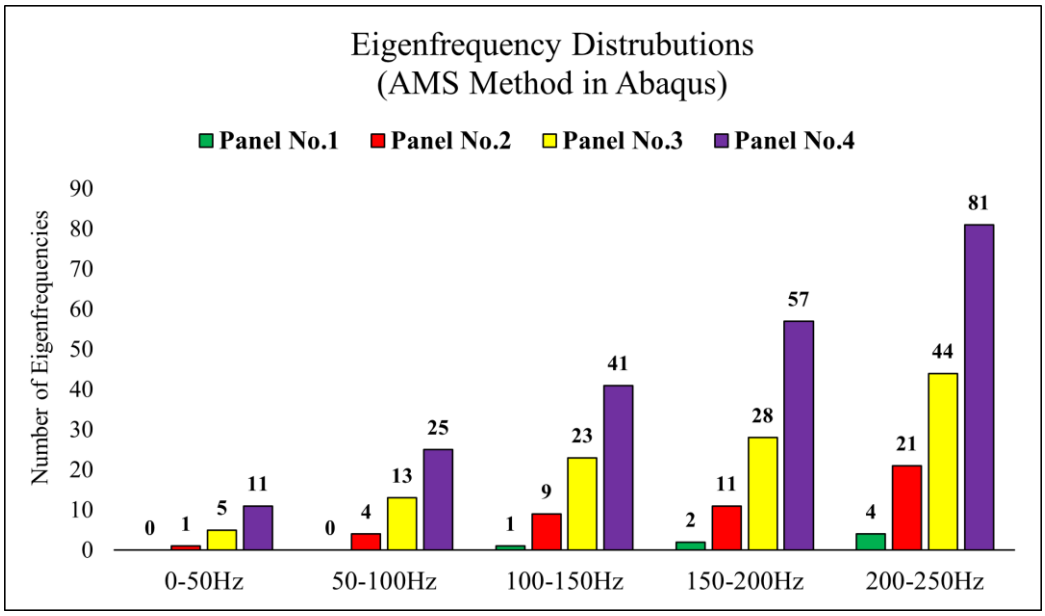


Figure 15.20: Eigenfrequencies distribution of size-controlled panels ("AMS Method").

15.2.2 Steady-State Dynamic Analyses in Abaqus

When steady-state dynamic analysis happens in physical space, it can be called a physical-space-based steady-state dynamic analysis. Similarly, when steady-state dynamic analysis

happens in modal space, it can be called a modal-space-based steady-state dynamic analysis. Furthermore, simulation spaces influence the choice of software architecture in Abaqus.

As introduced in subsections **4.2.1** and **4.2.2**, Abaqus has a traditional and high-performance SIM architecture. Traditional architecture does not have any application constraints. In contrast, the SIM architecture is applicable only in modal space. Thus, the physical-space-based steady-state dynamic analysis must be driven by the traditional architecture with the "Ordinary Method". Besides, the "Ordinary Method" can run in traditional architecture or SIM architecture. Since the "AMS Method" is a SIM-architecture-based method, it can be used in modal space. This paper adopts SIM architecture in modal-space-based analyses to drive the "Ordinary Method" and the "AMS Method".

In the steady-state dynamic analyses of the four testing panels (see **Figure 15.1**), Abaqus ignores the frequency dependency of the melamine foam. Thus, the models have no frequency dependency. The varieties are simulation spaces (physical and modal spaces). Because of the simulation spaces, constraints are applied to the "Ordinary Method" and the "AMS Method" in steady-state dynamic analyses. **Table 15.10** records the matching conditions of the two numerical methods to the two simulation spaces. As we can see, if the numerical method matches the specific simulation space, the corresponding intersection box records "YES". While if the numerical method does not match the specific simulation space, the corresponding intersection box records "NO".

Successful and efficient dynamic analysis needs proper numerical damping models. The proper numerical damping models in different simulation scenarios are different. Chapter **8** has elaborated on the damping models in Abaqus that can be used in the steady-state dynamic analyses in subsection **15.2.3**. And **Table 15.11** lists the links of the corresponding elaborations of the damping models behind the specific simulation scenarios. Please click the corresponding links in the intersection blanks for more information about the damping models of different simulation scenarios. For proper damping models applied to the aluminium alloy and the melamine foam in different situations, please review subsection **8.6**.

Table 15.10: Matching conditions in Abaqus (simulation space, numerical methods).

Numerical Methods	PS	MS
Ordinary Method	YES	YES
AMS Method	NO	YES

Table 15.11: Proper damping in Abaqus (simulation space, numerical methods).

Numerical Methods	PS	MS
Ordinary Method	**8.2**	**8.3**
AMS Method		**8.3**

15.2.3 Relative Accuracy Evaluations of Available Methods in Steady-State Dynamic Analyses

As introduced before, the "Ordinary Method" is the method without adopting superelement techniques or superelement-correlated techniques. This method is much more general than the "AMS Method". It can be used in physical-space-based technical routes or modal-space-based technical routes. In general, the "Ordinary Method" has relatively higher accuracies. In theory, the most accurate simulation strategy is the physical-space-based analyses with the "Ordinary Method". Thus, the testing results from the physical-space-based analyses through the "Ordinary Method" can be seen as a relative baseline for judging the accuracy of the other simulation strategies.

Figure 15.21 displays the frequency response functions of the given response points on four testing metamaterial panels. Subplot (a) is the frequency-amplitude plot that shows the point response of "Panel No.1 (with 25 cells and indicated in an abbreviation P25)". Then, subplot (b) is the frequency-amplitude plot that shows the point response of "Panel No.2 (with 100 cells and indicated in an abbreviation P100)". After that, subplot (c) is the frequency-amplitude plot that shows the point response of "Panel No.3 (with 225 cells and indicated in an abbreviation P225)". Lastly, subplot (d) is the frequency-amplitude plot that shows the point response of "Panel No.4 (with 400 cells and indicated in an abbreviation P400)". We can see that in each subplot, the title word "FRF" means the "frequency response function", and the title word in bracket "M. NFreq" means the model consists of "Non-Frequency-Dependent Material".

Let us read the legend box at the right-top corner of each subplot. The abbreviation "P. Response" means the response point. Then, the task labels in brackets indicate the panel's type, the material's pattern, the simulation space, and the calculation method. Let us take "Panel No.1", corresponding to subplot (a), as an example. In the legend box, the label "P25MSFRF_AMS" means the frequency response function comes from a modal-space-based dynamic analysis using the "AMS Method", and the objective model is the "Panel No.1" with 25 cells which adopts the "Non-Frequency-Dependent Material". After that, the label "P25MSFRF_OD" means the frequency response function comes from a modal-space-based dynamic analysis using the "Ordinary Method", and the objective model is the "Panel No.1" with 25 cells which adopts the "Non-Frequency-Dependent Material". At last, the label "P25PSFRF_OD" means the frequency response function comes from a physical-space-based dynamic analysis using the "Ordinary Method", and the objective model is the "Panel No.1" with 25 cells which adopts the "Non-Frequency-Dependent Material".

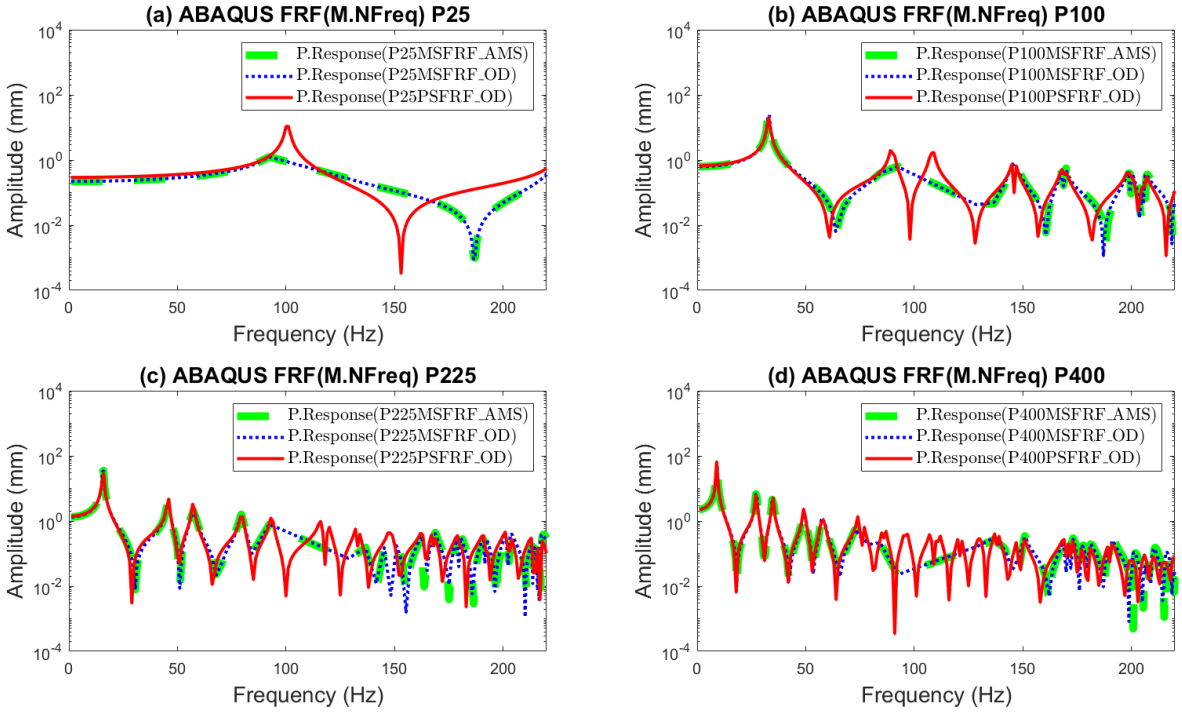


Figure 15.21: FRF of structural response point (four panels, Abaqus, different simulation strategies).

Let us talk about the relative accuracy of the three simulation strategies. We know that the testing results from the physical-space-based analyses with the "Ordinary Method" are chosen to be the relative baseline. Thus, this simulation strategy has the highest relative accuracy. Based on observation, the testing results from the modal-space-based analyses through the "Ordinary Method" and the "AMS Method" can be matched. However, they cannot completely match the baseline results. In the frequency range $0-90\,\text{Hz}$, the testing results from modal-space-based analyses can almost coincide with the baseline results, which means they have good relative accuracy. However, in the frequency range around $90-140\,\text{Hz}$, the testing results from modal-space-based analyses lost their resonant points, and big errors happen in this frequency range. Then, in the frequency range around $140-220\,\text{Hz}$, the testing results from modal-space-based analyses partially match the baseline results.

The unmatching condition in the frequency range around $90-140\,\text{Hz}$ and the partially unmatching condition in the frequency range around $140-220\,\text{Hz}$ are two inaccuracy cases. And the inaccuracy caused by the user's setting errors can be excluded; otherwise, the testing results from modal-space-based analyses will present a completely unmatching condition instead of a partially unmatching one. Then, the only reasonable explanation for the partially unmatched conditions is that Abaqus's modal-space-based steady-state dynamic analyses are unstable. Compared with analogous settings, the modal-space-based analyses in Nastran have higher relative accuracy (see sections **15.1.4**, **15.1.6**, and **15.1.7**).

15.2.4 Section Conclusions

Let us have a brief review of the contents in section **15.2**. As subparts, subsection **15.2.1** is a preliminary illustration of the testing panels' dynamic performances regarding their eigenfrequencies distribution. In this section, simulations will evaluate the capabilities of the two numerical methods from Abaqus in eigenvalue extraction analyses. We can perceive the cases from the results listed in **Table 15.12**, the "Ordinary Method" and the "AMS Method" of Abaqus are stable in eigenvalue extraction analyses. Besides stability, efficiency is another important factor that must be considered when choosing the best numerical method for eigenvalue extraction analyses. Based on the results in **Table 15.13**, the most efficient method in eigenvalue extraction analyses is the "AMS Method". With the model's DOFs increasing, the "AMS Method" superiority is more obvious. This conclusion supports the theoretical arguments in sections **4.2.2** and **4.2.5**.

Table 15.12: The total number of extracted eigenmodes through Abaqus's two numerical methods (four size-controlled panels, 0–250Hz).

	Panel No.1	Panel No.2	Panel No.3	Panel No.4
Ordinary Method	7	46	115	218
AMS Method	7	46	113	215
Number of Mesh Grids	3605	14205	31805	56405

Table 15.13: The time consumption (second) of eigenvalue extractions through Abaqus's two numerical methods (four size-controlled panels, 0–250Hz).

	Panel No.1	Panel No.2	Panel No.3	Panel No.4
Ordinary Method	1.9s	8.8s	26.0s	74.3s
AMS Method	1.6s	5.5s	11.5s	21.0s
Number of Mesh Grids	3605	14205	31805	56405

Subsection **15.2.2** is a technical description of how to set available simulation strategies. The theoretical background of this subsection is chapters **4** and **8**. These two chapters supply theoretical analyses and arguments. Then, subsection **15.2.3** illustrates the relative accuracy of the two numerical methods in treating models with the "Non-Frequency-Dependent Material". And the control factor in subsection **15.2.3** is the simulation spaces. In the following contents, **Table 15.14** lists the relative accuracy grades of the two numerical methods in corresponding simulation scenarios. The relative accuracy grades from the lowest to the highest are denoted with numbers 1 to 5. And if the accuracy grade is higher than 4, this method is recommended.

Table 15.14: The accuracy grades of Abaqus's simulation strategies in steady-state dynamic analyses.

Numerical Methods	PS (M. NFreq)	MS (M. NFreq)
Ordinary Method	5	2
AMS Method		2

Besides relative accuracy, efficiency is another important factor that must be considered in choosing proper simulation strategies. From **Table 15.15** to **Table 15.18**, corresponding to "Panel No.1" to "Panel No.4", list the time consumption of the two numerical methods in corresponding simulation scenarios. And the time consumption is an indicator of the simulation efficiency. Let us do a brief introduction about the tables from **Table 15.15** to **Table 15.18** before analyzing the data. In the column title, the abbreviation "PS" means physical-space-based technical routes, and "MS" means modal-space-based technical routes. Then, in the bracket of the column title, the abbreviation "M. NFreq" means "Non-Frequency-Dependent Material".

Based on an overview of the time consumption listed in the four tables, we can perceive that the efficiency of modal-space-based analyses is much higher than the physical-space-based analyses. In detail, let us concern the physical-space-based analyses in the first place. The only numerical method in physical space is the "Ordinary Method". Then, let us concern the modal-space-based analyses. The "Ordinary Method" and the "AMS Method" are available in modal space, but the "AMS Method" is more efficient than the "Ordinary Method" since models adopt the "Non-Frequency-Dependent Material" pattern. Finally, the conclusion is that the physical-space-based analysis with the "Ordinary Method" is the most accurate but inefficient simulation strategy, and the modal-space-based analysis with the "AMS Method" is the most efficient but not accurate enough.

Table 15.15: The time (second) consumption of Abaqus's simulation strategies in steady-state dynamic analyses ("Panel No.1", the number of mesh grids is 3605, the number of excitation frequencies is 220).

Numerical Methods	PS (M. NFreq)	MS (M. NFreq)
Ordinary Method	131.0s	2.3s
AMS Method		1.7s

Table 15.16: The time (second) consumption of Abaqus's simulation strategies in steady-state dynamic analyses ("Panel No.2", the number of mesh grids is 14205, and the number of excitation frequencies is 220).

Numerical Methods	PS (M. NFreq)	MS (M. NFreq)
Ordinary Method	557.8s	9.0s
AMS Method		5.2s

Table 15.17: The time (second) consumption of Abaqus's simulation strategies in steady-state dynamic analyses ("Panel No.3", the number of mesh grids is 31805, the number of excitation frequencies is 220).

Numerical Methods	PS (M. NFreq)	MS (M. NFreq)
Ordinary Method	1287.8s	26.2s
AMS Method		11.9s

Table 15.18: The time (second) consumption of Abaqus's simulation strategies in steady-state dynamic analyses ("Panel No.4", the number of mesh grids is 56405, the number of excitation frequencies is 220).

Numerical Methods	PS (M. NFreq)	MS (M. NFreq)
Ordinary Method	2548.8s	74.5s
AMS Method		21.8s

Finally, the conclusion is that the "AMS Method" is the most efficient for modal-space-based steady-state dynamic analyses in treading non-frequency-dependent models. This conclusion supports the theoretical arguments in sections **4.2.2**, **4.3.3**, and **4.3.5**. However, modal-space-based dynamic analyses in Abaqus with the "Ordinary Method" and the "AMS Method" are not accurate enough. Elaborations in subsection **15.2.3** and accuracy grades in **Table 15.14** convince the inaccuracy. Thus, Abaqus's physical-space-based analyses with the "Ordinary Method" will be used in section **17.2** to conduct vibro-acoustic analyses on non-frequency-dependent models.

15.3 Efficiency Benchmarks of Actran

Since Actran does not support superelement techniques in dynamic analysis (see subsection **5.3.2**), the efficiency benchmark of Actran will not assess numerical methods based on superelement techniques. Thus, the "Ordinary Method" is the only choice for Actran in eigenvalue extraction and steady-state dynamic analyses. In eigenvalue extraction analyses, the material's frequency dependency is inactive, and the damping effect is ignored; thus, there are no constraints in using the "Ordinary Method". However, in steady-state dynamic analyses, the materials' frequency dependency can be set in physical-space-based analyses (see subsection **5.3.4**). Still, the materials' frequency dependency cannot be set in modal-space-based analyses (see subsection **5.3.3**). The reason is that the modal-space-based analyses in Actran need two independent calculation cycles (see subsection **5.2.1**): an eigenvalue extraction analysis and a subsequent steady-state dynamic analysis. Thus, due to the lost intermediate data management, the frequency dependency cannot be automatically updated in the two independent calculation cycles.

15.3.1 Eigenvalue Extraction Analyses in Actran

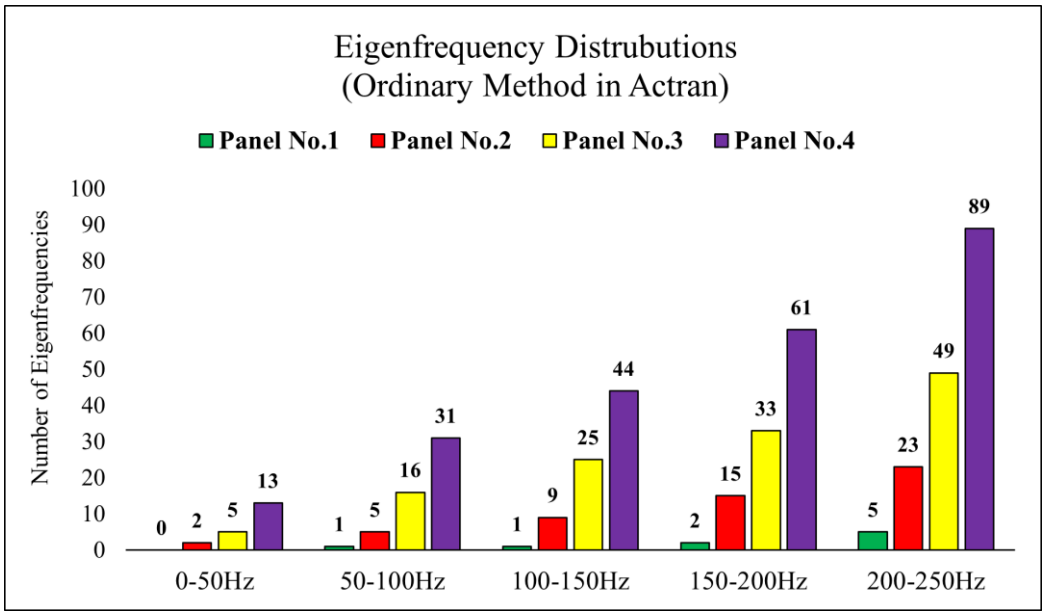


Figure 15.22: Eigenfrequencies distribution of size-controlled panels (Actran's "Ordinary Method").

This subsection displays the eigenfrequencies distribution of different testing panels at the frequency range of $0-250\,\mathrm{Hz}$. These eigenfrequencies come from corresponding eigenvalue extraction analyses with the "Ordinary Method". **Figure 15.22** illustrates that the eigenfrequencies are distributed in five frequency range subsections, and the distributions of the four testing panels present an analogous tendency. Additionally, the "Ordinary Method" from Actran is stable in eigenvalue extraction analyses. And the time consumption of the

eigenvalue extraction analyses through the "Ordinary Method" will be displayed in subsection **15.3.4**. Finally, we can perceive that these distributed eigenfrequencies are directly in concert with the resonant and anti-resonant points in frequency response functions (see **Figure 15.23** in subsection **15.3.3**).

15.3.2 Steady-State Dynamic Analyses in Actran

Compared with Nastran and Abaqus, Actran is not a highly integrated simulation platform. Users must choose proper solvers for different simulation tasks in Actran. For efficiency, the testing objects adopt the solver MUMPS for the eigenvalue extraction analysis (see subsection **5.2.3**) and the physical-space-based steady-state dynamic analysis (see subsection **5.2.5**). The modal-space-based analyses in Actran need two calculation cycles since Nastran or Abaqus only need one (see subsection **5.2.1**). Thus, the modal-space-based analyses in Actran are incompatible with those in Nastran or Abaqus. In addition, the modal-space-based dynamic analyses do not support frequency dependency. Due to the upper reasons, the steady-state dynamic analyses will not be executed in the modal space. Thus, the physical-space-based analysis with the "Ordinary Method" is the unique simulation strategy for steady-state dynamic analyses in Actran (see **Table 15.19**).

Successful and efficient dynamic analysis needs support from proper numerical damping models. Chapter **9** has elaborated on the damping models in Actran that can be used in the steady-state dynamic analyses in subsection **15.3.3**. And **Table 15.20** lists the link of the corresponding elaborations of the damping model behind the specific simulation scenario. For proper damping models applied to the aluminium alloy and the melamine foam in different situations, please review subsection **9.4**.

Table 15.19: The matching condition in Actran (simulation space, numerical method).

	PS
Ordinary Method	YES

Table 15.20 Proper damping in Actran (simulation space, numerical method).

	PS
Ordinary Method	**9.2.2**

15.3.3 Results of Steady-State Dynamic Analyses in Actran

This paper only uses the physical-space-based steady-state dynamic analyses with the "Ordinary Method" in Actran as the simulation strategy to treat frequency-dependent models. **Figure 15.23** displays the frequency response functions of the given response points on four testing metamaterial panels (see **Figure 15.1**). Subplot (a) is the frequency-amplitude plot that shows the point response of "Panel No.1 (with 25 cells and indicated in an abbreviation P25)".

Subplot (b) is the frequency-amplitude plot that shows the point response of "Panel No.2 (with 100 cells and indicated in an abbreviation P100)". Then, subplot (c) is the frequency-amplitude plot that shows the point response of "Panel No.3 (with 225 cells and indicated in an abbreviation P225)". Lastly, subplot (d) is the frequency-amplitude plot that shows the point response of "Panel No.4 (with 400 cells and indicated in an abbreviation P400)". We can see that in each subplot, the title word "FRF" means the "frequency response function", and the title word in bracket "M. Freq" means the model consists of frequency-dependent materials.

Let us read the legend box at the right-top corner of each subplot. The abbreviation "P. Response" means the response point. Then, the task labels in brackets indicate the panel's type, the material's pattern, the simulation space, and the calculation method. Let us take "Panel No.1", corresponding to subplot (a), as an example. In the legend box, the label "P25f_PSFRF_OD" means the frequency response function comes from a physical-space-based dynamic analysis using the "Ordinary Method", and the objective model is the "Panel No.1" with 25 cells which adopts the frequency-dependent material pattern.

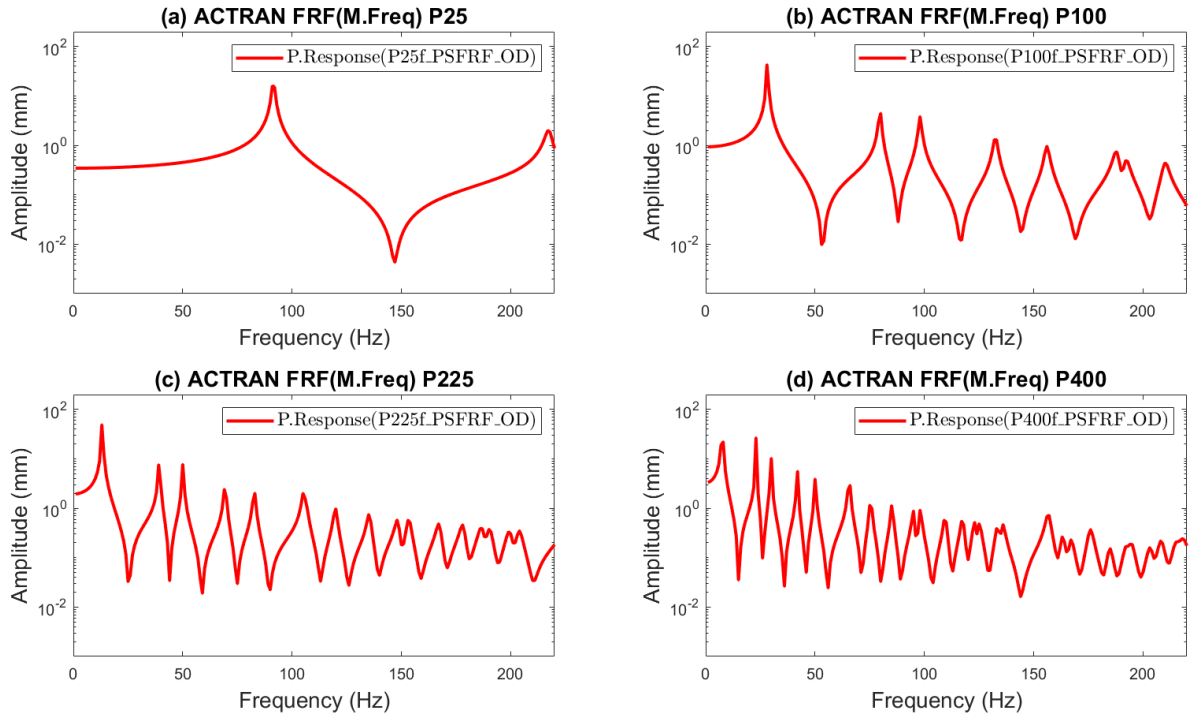


Figure 15.23: FRF of structural response point (four panels, Actran, on simulation strategy).

Let us see the information from the four figures. With the panel's size increase, the more resonant and anti-resonant points present. This phenomenon is in concert with the eigenvalue extraction analyses in subsection **15.3.1**.

15.3.4 Section Conclusions

Let us have a brief review of the contents in section **15.3**. Subsection **15.3.1** is a preliminary illustration of the testing panels' dynamic performances regarding their eigenfrequency distributions. From the situations presented in subsection **15.3.2** and based on the results in **Table 15.21**, we know that the "Ordinary Method" from Actran is stable in eigenvalue extraction analyses. At the same time, **Table 15.22** lists the time consumption of the eigenvalue extraction analyses of the four testing panels.

Table 15.21: The total number of extracted eigenmodes through Actran's numerical method (four size-controlled panels, 0–250Hz).

	Panel No.1	Panel No.2	Panel No.3	Panel No.4
Ordinary Method	9	54	128	238
Number of Mesh Grids	3605	14205	31805	56405

Table 15.22: The time (second) consumption of eigenvalue extractions through Actran's "Ordinary Method" (four size-controlled panels, 0–250Hz).

	Panel No.1	Panel No.2	Panel No.3	Panel No.4
Ordinary Method	5.0s	14.0s	63.0s	214.0s
Number of Mesh Grids	3605	14205	31805	56405

Then, subsection **15.3.3** illustrates the performance of the physical-space-based dynamic analyses with the "Ordinary Method" in treating models with the frequency-dependent material pattern. **Table 15.23** lists the time consumption of the physical-space-based steady-state dynamic analyses with the "Ordinary Method" in testing the four metamaterial panels.

Table 15.23: The time (second) consumption of Actran's simulation strategy in steady-state dynamic analyses (four size-controlled panels, the number of excitation frequencies is 220).

	Panel No.1	Panel No.2	Panel No.3	Panel No.4
Ordinary Method	101.0s	415.0s	939.0s	1752.0s
Number of Mesh Grids	3605	14205	31805	56405

15.4 Chapter Conclusions

Let us have a brief review of chapter **15**. Section **15.1** elaborates on the relative accuracy and efficiency of the best simulation strategy from Nastran for testing the dynamic performances of metamaterial samples setting with the frequency-dependent material pattern. Finally, based on the conclusions in subsection **15.1.8**, the best method for eigenvalue extraction analyses in Nastran is the "ACMS Method". At the same time, the best simulation strategy for testing the metamaterial's dynamic performance is that the model adopts the "Classic Frequency-Dependent Material" pattern and is treated by the modal-space-based steady-state dynamic analyses with the "Ordinary Method" from Nastran.

Section **15.2** elaborates on the relative accuracy and efficiency of the best simulation strategy from Abaqus for testing the dynamic performances of the metamaterial samples that are set with the "Non-Frequency-Dependent Material" pattern. Finally, based on the conclusions in subsection **15.2.4**, the best method for eigenvalue extraction analyses in Abaqus is the "AMS Method". At the same time, the only proper simulation strategy for testing the metamaterial's dynamic performances is that the model adopts the "Non-Frequency-Dependent Material" pattern and is treated by the physical-space-based steady-state dynamic analyses with the "Ordinary Method" from Abaqus.

Subsection **15.3** elaborates on the dynamic performances of the metamaterial samples. And the simulation strategy in this section is that the model adopts the frequency-dependent material pattern and is treated by the physical-space-based steady-state dynamic analyses with the "Ordinary Method" from Actran. Finally, based on the conclusions in subsection **15.3.4**, the "Ordinary Method" from Actran is competent in the eigenvalue extraction and physical-space-based steady-state dynamic analyses.

Comparisons of simulation strategies within the software have already been made. Then, comparisons of simulation strategies between software are needed to make conclusions in this chapter. In the following contents, the first part is the efficiency comparison between the representative numerical methods from Nastran, Abaqus, and Actran in eigenvalue extraction analyses. After that, the second part is the accuracy comparison between the baseline simulation strategies from Nastran, Abaqus, and Actran in physical-space-based steady-state dynamic analyses. Finally, the third part is the efficiency comparison between the representative simulation strategies from Nastran, Abaqus, and Actran in steady-state dynamic analyses.

Starting from the first conclusion, let us discuss the efficiency comparison between the representative numerical methods from Nastran, Abaqus, and Actran in eigenvalue extraction analyses. As we know, the most efficient numerical method in Nastran for doing eigenvalue extraction analyses is the "ACMS Method". At the same time, the most efficient numerical method in Abaqus for doing eigenvalue extraction analyses is the "AMS Method". These two methods are similar. They are superelement-technique-based numerical methods in Nastran and Abaqus, respectively (see subsections **3.10.2** and **4.2.2**).

In contrast, the only proper method in Actran for eigenvalue extraction analyses is the "Ordinary Method". Its efficiency is not very high. Finally, based on the time consumption in **Table 15.24**, Nastran's "ACMS Method" is the most efficient for eigenvalue extraction analyses.

Table 15.24: The time (second) consumption of eigenvalue extraction analyses through different software's numerical methods (four size-controlled panels, 0–250Hz).

	Panel No.1	Panel No.2	Panel No.3	Panel No.4
Ordinary Method (Actran)	5.0s	14.0s	63.0s	214.0s
AMS Method (Abaqus)	1.6s	5.5s	11.5s	21.0s
ACMS Method (Nastran)	2.5s	3.7s	8.4s	16.5s
Number of Mesh Grids	3605	14205	31805	56405

Then, starting from the second conclusion, let us discuss the accuracy comparison between the baseline simulation strategies from Nastran, Abaqus, and Actran in physical-space-based steady-state dynamic analyses. As we know, supposing treating frequency-dependent models in Nastran, the baseline simulation strategy uses the "Classic Frequency-Dependent Material" pattern to build the frequency dependency, then uses the physical-space-based dynamic analyses with the "Ordinary Method" to test it. In the same way, supposing treating non-frequency-dependent models in Nastran, the baseline simulation strategy is building the model with the "Non-Frequency-Dependent Material" pattern, then using the physical-space-based dynamic analyses with the "Ordinary Method" to test it. The physical-space-based dynamic analyses with the "Ordinary Method" are not efficient but are relatively the most accurate among all available software simulation strategies. This rule is also valid in Abaqus and Actran.

Figure 15.24 displays the frequency response functions of the given response points on four testing metamaterial panels with non-frequency-dependent models. And the results are calculated through the physical-space-based steady-state dynamic analyses with the "Ordinary Method" from Nastran and Abaqus, respectively. At the same time, **Figure 15.25** displays the frequency response functions of the given structural response points on four testing metamaterial panels with frequency-dependent models. And the results are calculated through the physical-space-based steady-state dynamic analyses with the "Ordinary Method" from Nastran and Actran, respectively. In the two figures, subplot (a) is the frequency-amplitude plot that shows the point response of "Panel No.1 (with 25 cells and indicated in an abbreviation P25)". Then, subplot (b) is the frequency-amplitude plot that shows the point response of "Panel No.2 (with 100 cells and indicated in an abbreviation P100)". After that, subplot (c) is the frequency-amplitude plot that shows the point response of "Panel No.3 (with 225 cells and indicated in an abbreviation P225)". Lastly, subplot (d) is the frequency-amplitude plot that shows the point response of "Panel No.4 (with 400 cells and indicated in an abbreviation P400)". The "FRF" title in each subplot means the "frequency response function". In **Figure 15.24**, the title word in bracket "M. NFreq" means the model consists of non-frequency-dependent material. In **Figure 15.25**, the title word in bracket "M. Freq" means the model consists of frequency-dependent materials.

Let us read the legend box at the right-top corner of each subplot. The abbreviation "P. Response" means the response point. Then, the task labels in brackets indicate the panel's type,

the material's pattern, the simulation space, and the calculating method. Let us take "Panel No.1", corresponding to subplot (a), as an example. In the legend box of **Figure 15.24**, the label "Nastran_P25PSFRF_OD" means the frequency response function comes from a physical-space-based dynamic analysis using the "Ordinary Method" from Nastran, and the objective model is "Panel No.1" with 25 cells. This panel adopts the "Non-Frequency-Dependent Material" pattern. After that, the label "Abaqus_P25PSFRF_OD" means the frequency response function comes from a physical-space-based dynamic analysis using the "Ordinary Method" from Abaqus. And the objective model is "Panel No.1" with 25 cells. This panel adopts the "Non-Frequency-Dependent Material" pattern.

In the legend box of **Figure 15.25**, the label "Nastran_P25cf_PSFRF_OD" means the frequency response function comes from a physical-space-based dynamic analysis using the "Ordinary Method" from Nastran. And the objective model is "Panel No.1" with 25 cells. And this panel adopts the "Classic Frequency-Dependent Material" pattern. After that, the label "Actran_P25f_PSFRF_OD" means the frequency response function comes from a physical-space-based dynamic analysis using the "Ordinary Method" from Actran. And the objective model is "Panel No.1" with 25 cells. And this panel adopts the "Classic Frequency-Dependent Material" pattern.

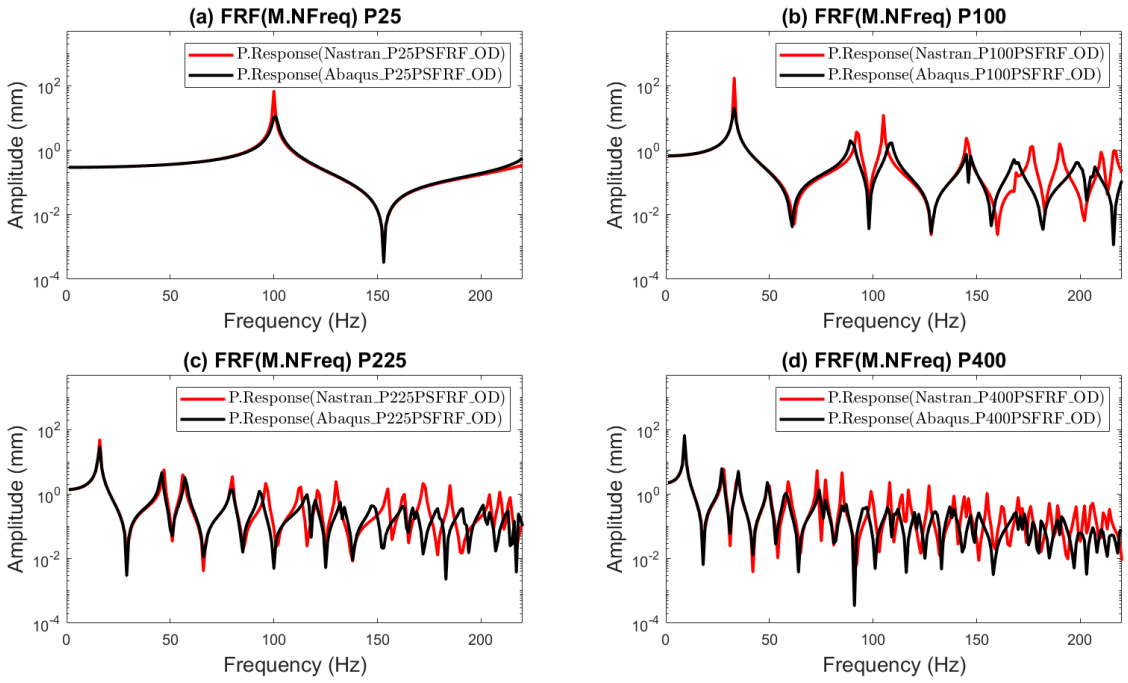


Figure 15.24: FRF of structural response point (non-frequency-dependent panels, Nastran vs Abaqus, baseline).

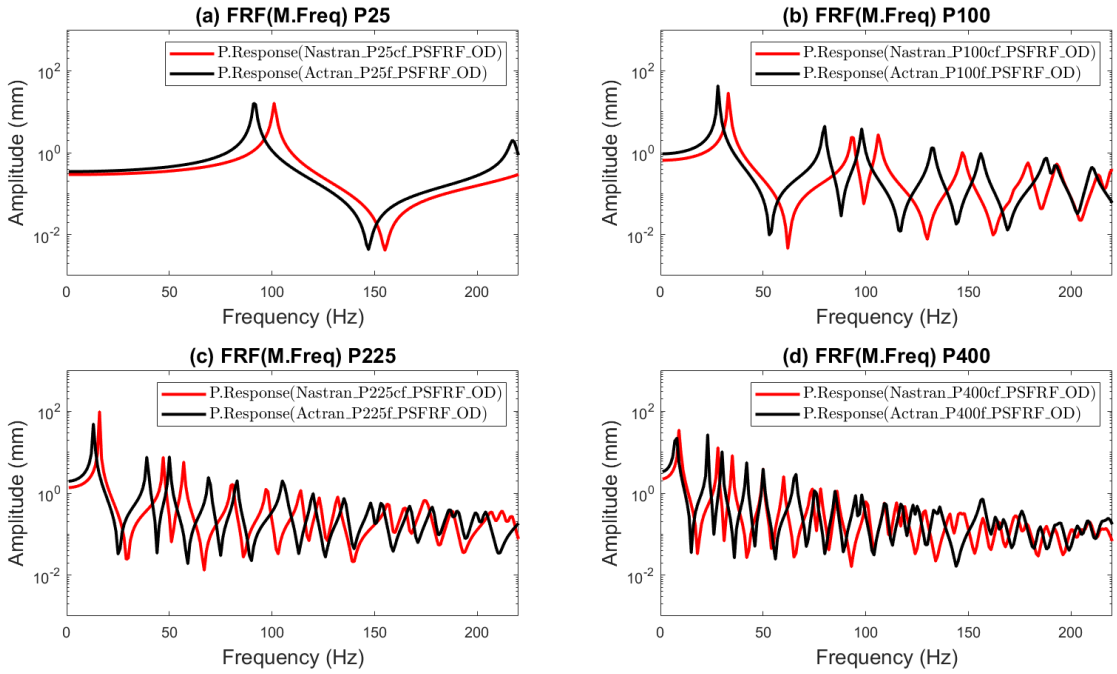


Figure 15.25: FRF of structural response point (frequency-dependent panels, Nastran vs Actran, baseline).

Let us see the information from **Figure 15.24**. Subplot (a) shows that the results of "Panel No.1" from Nastran's analysis and Abaqus's analysis can match together below the frequency of 200 Hz. Then, subplot (b) shows that the results of "Panel No.2" from Nastran's analysis and Abaqus's analysis can match together below the frequency of 150 Hz. After that, subplot (c) shows that the results of "Panel No.3" from Nastran's analysis and Abaqus's analysis can match the frequency below 100 Hz. At last, subplot (d) shows that the results of "Panel No.4" from Nastran's analysis and Abaqus's analysis can match together below the frequency of 70 Hz. Thus, if using Nastran's results as a reference, the accuracy of Abaqus's results is inaccurate.

Let us see the information from **Figure 15.25**. From subplot (a) to subplot (b) corresponding to "Panel No.1" to "Panel No.4", we can see that the results from Nastran's analysis and Actran's analysis cannot match together. Even though the shapes of the frequency response functions from Nastran's and Actran's analyses are analogous, there are obvious horizontal and vertical offsets between the two frequency response functions regarding the X-axis and Y-axis, respectively. Thus, if using Nastran's results as the reference, the convergence of Actran's numerical solution has an offset regarding the convergence of Nastran's numerical solution.

We can see the discrepancies between the results of non-frequency-dependent models from Nastran's baseline simulation strategy and Abaqus's baseline simulation strategy. At the same time, we can see the offsets between the results of frequency-dependent models from Nastran's baseline simulation strategy and Actran's baseline simulation strategy. It is reasonable to believe that discrepancies and offsets are software's intrinsic problems. User setting errors do not cause this situation. Otherwise, the shapes and amplitudes of the frequency

response function may change a lot due to wrong material properties, wrong boundary conditions, or wrong loading conditions by user's setting errors, etc.

At last, starting from the third conclusion, let us discuss the efficiency comparison between the representative simulation strategies from Nastran, Abaqus, and Actran in steady-state dynamic analyses. On the one hand, for models consisting of "Non-Frequency-Dependent Material" patterns, the most efficient simulation strategy in Nastran is the modal-space-based steady-state dynamic analyses with the "ACMS Method". At the same time, the most efficient simulation strategy in Abaqus is the modal-space-based steady-state dynamic analyses with the "AMS Method". In **Table 15.25**, the modal-space-based steady-state dynamic analyses with the "ACMS Method" is denoted with the abbreviation "MSFRF_ACMS", and the modal-space-based steady-state dynamic analyses with the "AMS Method" is denoted with the abbreviation "MSFRF_AMS". And the abbreviation "M. NFreq" means the "Non-Frequency-Dependent Material" pattern. Then, the time consumption in **Table 15.25** shows that the efficiencies of the two simulation strategies from the two software are at the same level for treating non-frequency-dependent models in steady-state dynamic analyses. However, based on the verifications in subsection **15.2.3** and conclusions in subsection **15.2.4**, the modal-space-based steady-state dynamic analyses with the "AMS Method" from Abaqus are not an accurate simulation strategy.

On the other hand, for models consisting of frequency-dependent materials, one of the most efficient simulation strategies in Nastran is the modal-space-based steady-state dynamic analyses with the "Ordinary Method". And to process the model that adopts the "Classic Frequency-Dependent Material" pattern. And this simulation strategy is the most accurate one. At the same time, the proper simulation strategy in Actran is the physical-space-based steady-state dynamic analyses with the "Ordinary Method". In **Table 15.26**, the modal-space-based steady-state dynamic analyses with the "Ordinary Method" is denoted with the abbreviation "MSFRF_OD", and the physical-space-based steady-state dynamic analyses with the "Ordinary Method" is denoted with the abbreviation "PSFRF_OD". And the abbreviation "M. Freq" means the frequency-dependent material pattern. Then, the time consumption in **Table 15.26** shows that the efficiency of modal-space-based steady-state dynamic analyses with the "Ordinary Method" from Nastran is the highest for treating frequency-dependent models in dynamic analyses. Even in physical space, the efficiency of the "Ordinary Method" from Nastran is higher than the "Ordinary Method" from Actran for treating frequency-dependent models in dynamic analyses. In addition, based on the results shown in **Figure 15.25**, we know the offsets between the results from Nastran's physical-space-based "Ordinary Method" and Actran's physical-space-based "Ordinary Method". Thus, if Nastran's results are the reference, we can say that Actran's results are not so accurate.

Table 15.25: The time (second) consumption of steady-state dynamic analyses (four non-frequency-dependent size-controlled panels, Nastran vs Abaqus).

	Panel No.1	Panel No.2	Panel No.3	Panel No.4
MSFRF_AMS (Abaqus, M. NFreq)	1.6s	5.5s	11.5s	21.0s
MSFRF_ACMS (Nastran, M. NFreq)	2.7s	4.4s	11.3s	22.2s
Number of Mesh Grids	3605	14205	31805	56405

Table 15.26: The time (second) consumption of steady-state dynamic analyses (four frequency-dependent size-controlled panels, Nastran vs Actran).

	Panel No.1	Panel No.2	Panel No.3	Panel No.4
PSFRF_OD (Actran, M. Freq)	101.0s	415.0s	939.0s	1752.0s
PSFRF_OD (Nastran, M. Freq)	52.7s	253.2s	666.7s	1658.1s
MSFRF_OD (Nastran, M. Freq)	6.2s	27.6s	126.9s	397.9s
Number of Mesh Grids	3605	14205	31805	56405

Chapter 16

Eigenvalue Extraction Analyses of Aluminium-Inclusion-Controlled Panels

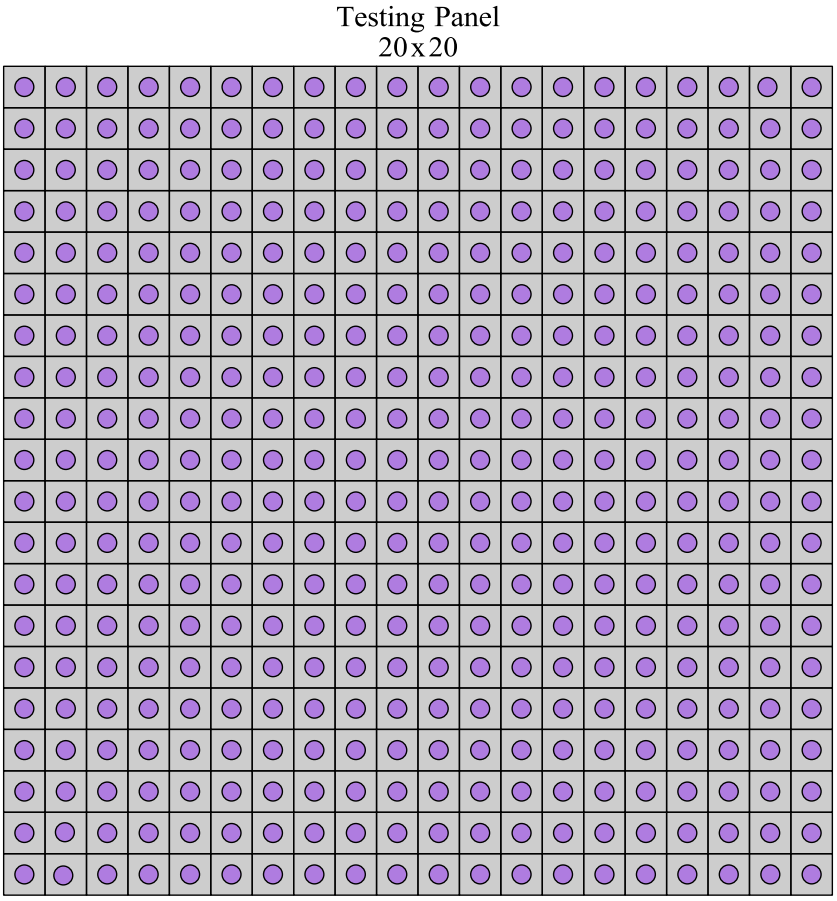


Figure 16.1: The layout drawing of the testing panels.

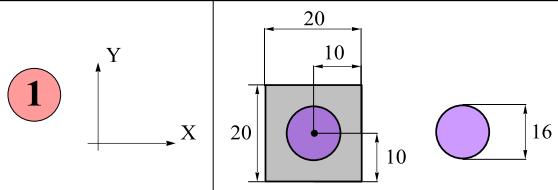
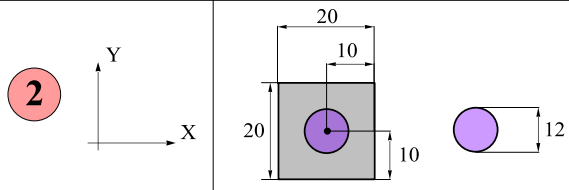
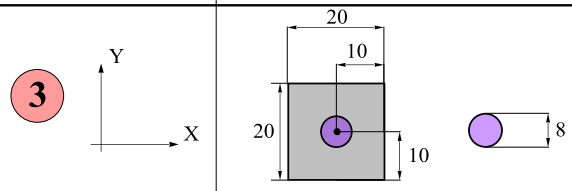
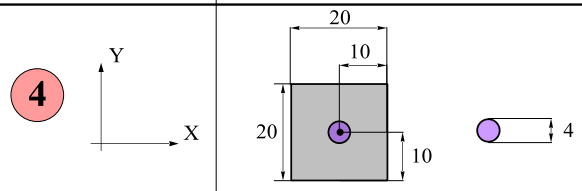
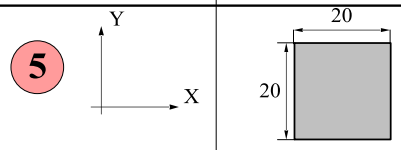
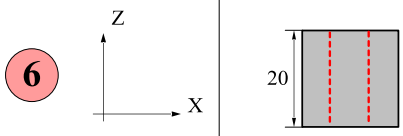
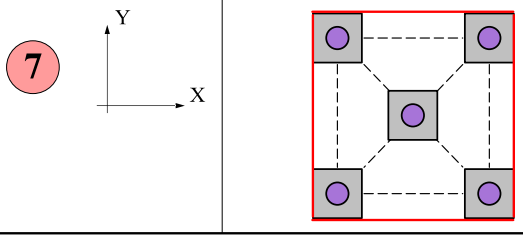
Coordinate	Layout Drawing	Parameters		
1		Seat (Cube)	Length = 20 mm (X coordinate)	Width = 20 mm (Y coordinate)
		Core (Bar)	Diameter = 16 mm (Radial coordinate)	Centre: (10, 10) (X, Y coordinates)
2		Seat (Cube)	Length = 20 mm (X coordinate)	Width = 20 mm (Y coordinate)
		Core (Bar)	Diameter = 12 mm (Radial coordinate)	Centre: (10, 10) (X, Y coordinates)
3		Seat (Cube)	Length = 20 mm (X coordinate)	Width = 20 mm (Y coordinate)
		Core (Bar)	Diameter = 8 m (Radial coordinate)	Centre: (10, 10) (X, Y coordinates)
4		Seat (Cube)	Length = 20 mm (X coordinate)	Width = 20 mm (Y coordinate)
		Core (Bar)	Diameter = 4 mm (Radial coordinate)	Centre: (10, 10) (X, Y coordinates)
5		Seat (Cube)	Length = 20 mm (X coordinate)	Width = 20 mm (Y coordinate)
6		Seat (Cube)	Thickness = 20 mm (Z coordinate)	
7		No.cells (NxN)	Sizes X, Y, Z (mm)	
		20 x 20	400 x 400 x 20	

Figure 16.2: Testing panels with different aluminium-alloy inclusions.

In chapter **15**, the control parameter of the testing panels is the panel size, and the panel cells are the same (see **Figure 15.1** and **Figure 15.2**). As elaborated in subsection **15.1.3**, the distribution of the eigenvalues in the low-frequency range influences the low-frequency performance of the testing samples. As we know, the thinner (higher ratio of the flat surface by volume) the testing panels are, the more eigenvalues will be found at the given frequency range of $0-250\,\mathrm{Hz}$. Besides the panel's flat-surface sizes, it is reasonable to assume that the degree of aluminium-alloy inclusions will influence eigenvalues distribution. Then, the degree of aluminium-alloy inclusions can be the control parameter while fixing the panel size.

Since the panel size is fixed, the number of unit cells is fixed. Therefore, the number of aluminium bars is fixed (see **Figure 16.1**). Thus, the operation to control the degree of aluminium-alloy inclusions is to control the aluminium-bar diameters. The testing panels adopt unit cells with five aluminium alloy inclusions (see **Figure 16.2**). The last one contains melamine foam only. Then, the panels with aluminium alloy, from the first to the fourth, consist of the aluminium bar with diameters of $4\,\mathrm{mm}$, $8\,\mathrm{mm}$, $12\,\mathrm{mm}$, and $16\,\mathrm{mm}$, respectively. All in all, the size of these five testing panels is $400\,\mathrm{x}\,400\,\mathrm{x}\,20\,\mathrm{mm}$ regarding X, Y, and Z-coordinates, and their lateral boundaries are fixed.

16.1 The Eigenfrequencies Distribution of Aluminium-Inclusion-Controlled Panels

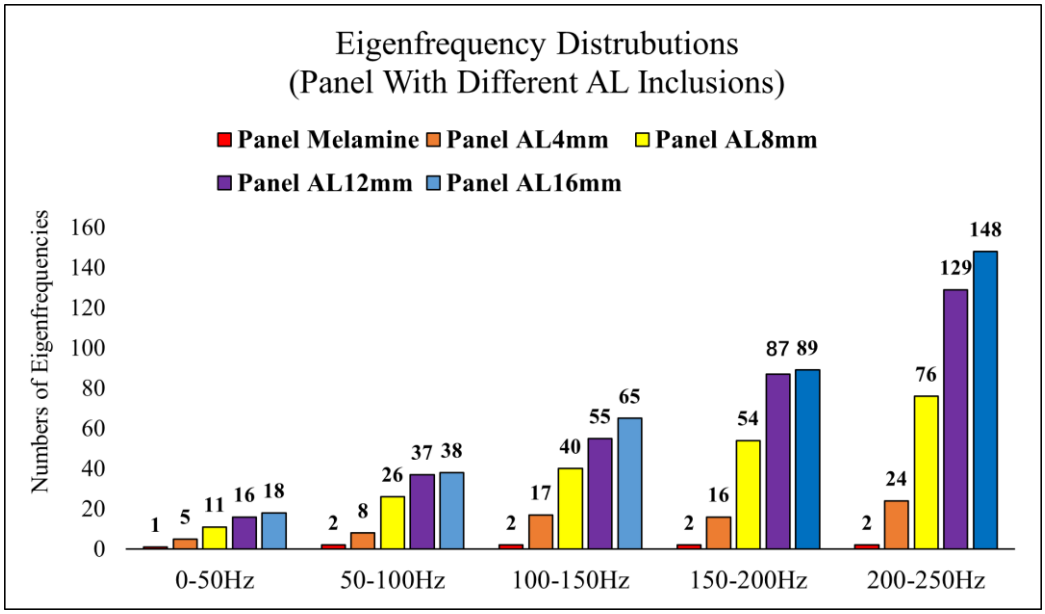


Figure 16.3: Eigenfrequencies distribution of aluminium-inclusion-controlled panels.

This section displays the eigenfrequencies distribution of testing panels in the same size but with different degrees of aluminium alloy inclusions at the frequency range of $0-250\,\mathrm{Hz}$. These eigenfrequencies come from eigenvalue extraction analyses by the "ACMS Method"

from Nastran. This numerical method is convinced with good accuracy and the highest efficiency (see section **15.4**). In **Figure 16.3**, the abbreviation "Panel Melamine" means the panel consists of only melamine foam. Then, the abbreviations "Panel AL4mm", "Panel AL8mm", "Panel AL12mm", and "Panel AL16mm" mean the panels contain the aluminium bar with diameters of 4mm, 8mm, 12mm, and 16mm, respectively. Finally, we can find two rules. In the first case, with the degree of aluminium alloy inclusion increases, more eigenfrequencies can be found. In the second case, more eigenfrequencies are distributed in the higher frequency range.

16.2 Chapter Conclusions

We know that the objectives of eigenvalue extraction analyses in chapter **15** are four size-controlled metamaterial panels; they have different ratios of the flat size by volume. And the distribution of the eigenvalues is influenced by the flat-volume ratios. More specifically, the larger the flat-volume ratios, the more eigenvalues of the corresponding panels can be found in the given frequency range of $0-250\,\mathrm{Hz}$. In addition, based on the efficiency benchmark tests in chapter **15**, Nastran's "ACMS Method" is the most efficient mothed for eigenvalue extraction analyses. Thus, this method is the choice for eigenvalue extraction analyses in this chapter.

To some extent, the eigenvalues distribution of the metamaterial panel in the given frequency range can show the dynamic performance of this panel. The testing panel is more sensible if more eigenvalues are found in the given frequency range. Besides panels' surface-volume ratios, other factors, such as panels' aluminium bar inclusions, can influence the distribution of the eigenvalues.

In **Table 16.1** and **Table 16.2**, the abbreviations "P. Melamine", "P. AL4", "P. AL8", "P. AL12", and "P. AL15" mean the panel includes the aluminium bar with diameters of 4mm, 8mm, 12mm, and 16mm, respectively. Based on the results in **Table 16.1**, we can see that as the degree of aluminium alloy inclusion increases, more eigenfrequencies can be found. Then, based on the records of time consumption in **Table 16.2**, we know that the "ACMS Method" efficiency from Nastran is very high in eigenvalue extraction analyses.

Table 16.1: The total number of extracted eigenmodes (aluminium-inclusion-controlled panels, Nastran's "ACMS Method", 0–250Hz).

The Number of Total Extracted Eigenmodes					
	P. Melamine	P. AL4	P. AL8	P. AL12	P. AL16
ACMS Method	9	70	207	324	358
Number of Mesh Grids	56405	72405	56405	40405	40405

Table 16.2: The time (second) consumption of the eigenvalue extraction analyses (aluminium-inclusion-controlled panels, Nastran's "ACMS Method", 0–250Hz).

Time Consumptions of Eigenvalue Extraction Analyses					
	P. Melamine	P. AL4	P. AL8	P. AL12	P. AL16
ACMS Method	12.3s	16.7s	15.9s	13.1s	14.3s
Number of Mesh Grids	56405	72405	56405	40405	40405

Chapter 17

Vibro-Acoustic Analyses of Metamaterial Samples

17.1 Simulation Strategy Design for Vibro-Acoustic Analyses

The mandatory requirements of applying vibro-acoustic analysis to the metamaterial samples include at least two aspects. One is that the software can supply proper acoustic sources. And another is that the software can properly build the numerical model of the metamaterial.

On the one hand, the aim of testing metamaterial samples is to observe their acoustic performances. The frequency response function in steady-state dynamic analyses is a good indicator. Thus, the diffuse incident field is the only choice as an acoustic source. In FEM software, the diffuse incident field can generate steady-state dynamic loadings, but other sources, such as the spherical and planar sources, cannot (see subsection **10.2.5**). Abaqus and Actran can model the diffuse incident field (see sections **4.4** and **5.4**), but Nastran cannot (see section **3.11**). Thus, the main platform for vibro-acoustic analysis must be an alternative between Abaqus and Actran.

On the other hand, the best simulation strategy must be found before starting simulation practices. In other words, before running assessments of metamaterial samples, it is necessary to do simulation strategy designs. This idea has already been mentioned in sections **2.2** and **2.3**. Overall, finding the most efficient simulation strategy to do the vibro-acoustic analyses of metamaterial samples is one of the main topics in this paper.

The simulation strategy designs consist of two stages. The first stage is the theoretical argument, and the second is the numerical verification. In the first stage, chapters **3**, **4**, and **5** elaborate on the theoretical arguments for the simulation strategy design based on Nastran, Abaqus, and Actran, respectively. Then, chapters **7**, **8**, and **9** compensate for the elaborations on the numerical damping models of Nastran, Abaqus, and Actran, respectively. In addition,

chapters **11**, **12**, and **13** compensate for elaboration on fundamental mathematical models of fluid-structure systems in Abaqus, Actran and Nastran, respectively.

In the second stage, chapter **14** discusses the practical accuracy assessments of different numerical methods in eigenvalue extraction analyses. At the same time, chapter **15** discusses the efficiency and relative accuracy of different numerical methods in eigenvalue extraction analyses and different simulation strategies in steady-state dynamic analyses. Then, as an extension, chapter **16** uses the most efficient numerical method to do the eigenvalue extraction analyses of the metamaterial panels with different aluminium bar inclusions. Finally, sections **17.1** and **17.2** discuss and certify the best simulation strategies for vibro-acoustic analyses. Then, the best simulation strategy for metamaterial designs will be used in the tests in section **17.3**.

17.1.1 Simulation Strategies from Nastran

Let us talk about the theoretical arguments and the numerical verifications of simulation strategies in Nastran. In the part of theoretical arguments, based on elaborations in section **3.10**, we know that models' frequency dependency and massive DOFs are two challenges. And the repeated periodic unit cells may bring shortcuts to increase the calculation efficiency.

In the part of theoretical arguments, based on the primary estimations in subsection **3.10.2**, four potential simulation strategies in Nastran may be capable of efficiently treating the massive DOFs. On the one hand, based on the estimation in subsection **3.10.3**, the "Image Superelement Method" can reduce the DOFs and utilize the periodic-structure advantages. However, this method may cost more in data recovery. Then, the "List Superelement Method" can reduce the DOFs and have less cost in data recovery. On the other hand, based on the elaborations in subsections **3.10.4**, **3.10.5**, **3.10.6**, **7.2.2**, **7.2.3**, **7.2.4**, and **7.2.5**, we know that the "Classic Frequency-Dependent Material" pattern and the "Viscoelastic Material" patterns can be used to build the frequency-dependent models in Nastran.

In the part of the numerical verifications, based on the simulation practices in section **14.1**, the "Ordinary Method", the "ACMS Method", the "List Superelement Method", and the "Image Superelement Method" from Nastran have the same practical accuracy level in eigenvalue extraction analyses of two PVC panels. Then, based on the simulation practices in section **15.1**, we know that the "ACMS Method" is the most efficient for eigenvalue extraction analyses. Then, the modal-space-based analyses with the "Ordinary Method" is the most efficient simulation strategy for steady-state dynamic analyses of frequency-dependent models. For more information, please review subsection **15.1.8**. In addition, Nastran cannot build diffuse incident fields, so it cannot execute steady-state vibro-acoustic analyses (see section **3.11**). Thus, Nastran's steady-state dynamic analyses are meaningless for this chapter. Instead, Nastran's eigenvalue extraction analyses with the "ACMS Method" are the focal point for this chapter. The reason is that Nastran's eigenvalue extraction analyses can cooperate with Actran's modal-space-based steady-state dynamic analyses. Nastran has the highest practical accuracy in eigenvalue extraction analyses regarding Abaqus and Actran (see section **14.4**), and Nastran's "ACMS Method" is most efficient in eigenvalue extraction analyses regarding Abaqus's "AMS Method" and Actran's "Ordinary Method" (see section **15.4**).

17.1.2 Simulation Strategies from Abaqus

Let us talk about the theoretical arguments and the numerical verifications of simulation strategies in Abaqus. In the part of theoretical arguments, the elaborations in subsection **4.3.2** are about superelement techniques in Abaqus. Then we know that superelement techniques in Abaqus are improper for treating metamaterial samples. Thus, the "Ordinary Method" and the "AMS Method" can be the candidates for the simulations of the metamaterial samples. After that, based on the elaborations in subsections **4.2.2**, **4.2.3** and **4.3.3**, modal-space-based technical routes in Abaqus may be capable of efficiently treating the massive DOFs, and the SIM-architecture-based "AMS Method" may have the highest efficiency. However, the modal-space-based technical route in Abaqus does not support the numerical model of the diffuse incident field (see section **4.4**). Thus, the only choice for Abaqus to do vibro-acoustic analyses is the physical-space-based technical route, and SIM architecture cannot drive this technical route. Then, subsection **4.3.4** shows that Abaqus does not support frequency-dependent anisotropic or orthotropic materials. Thus, the frequency dependency of the melamine foam cannot be modelled by Abaqus. However, the anelastic part of the melamine foam is more sensible to the frequency, and the elastic part of the melamine foam has almost no changes in the low-frequency range (see **Figure 2.11**); in addition, the value of the anelasticity is much smaller than the value of the elasticity. Thus, ignoring the frequency dependency of the melamine foam in Abaqus's simulations may be acceptable.

In the part of the numerical verifications, based on the simulation practices in section **14.2**, the "Ordinary Method" and the SIM-architecture-based "AMS Method" from Abaqus have the same practical accuracy level in eigenvalue extraction analyses. Then, based on the simulation practices in section **15.2**, we know that the SIM-architecture-based "AMS Method" is the most efficient in both eigenvalue extraction and steady-state dynamic analyses. However, modal-space-based technical routes cannot accurately run steady-state dynamic analyses (see subsection **15.2.3**). In addition, modal-space-based steady-state dynamic analyses cannot model the diffuse incident field (see section **4.4**).

17.1.3 Simulation Strategies from Actran

Let us talk about the theoretical arguments and the numerical verifications of simulation strategies in Actran. In the part of theoretical arguments, based on the elaborations in subsection **5.3.2**, Actran does not have full-fledged superelement techniques. Thus, the "Ordinary Method" is the only choice that Actran can use for treating the metamaterial samples. At the same time, based on the elaborations in section **5.4**, we know that diffuse incident fields in Actran can be applied in modal and physical spaces.

Based on the elaborations in subsections **5.2.1** and **5.3.3**, modal-space-based dynamic analyses in Actran need two calculation cycles. Thus, modal-space-based technical routes in Actran cannot directly execute frequency-dependent analyses. Fortunately, Actran can accept eigenmodes and modal parameters from other software, such as Nastran. Thus, the effects of frequency dependency in physical space can be approximately compensated with the modal frequency dependency in modal space, such as using the modal-frequency-dependent modal damping to replace the frequency-dependent physical damping.

Based on the elaborations in subsection **5.3.4**, physical-space-based technical routes in Actran can execute frequency-dependent analyses. And frequency dependency can be set to orthotropic materials. However, the calculation efficiency of physical-space-based dynamic analyses to frequency-dependent models is very low.

In the part of the numerical verifications, based on the simulation practices in section **14.3**, Actran exhibits the lowest practical accuracy in eigenvalue extraction analyses regarding Nastran and Abaqus. In contrast, Nastran has the highest practical accuracy. Then, based on the simulation practices in section **15.3**, we know that Actran is inefficient in eigenvalue extraction analyses. And its "Ordinary Method" in physical-space-based steady-state dynamic analyses for the frequency-dependent models is the most inefficient simulation strategy (see section **15.4**).

17.1.4 Available Simulation Strategies for Vibro-Acoustic Analyses

As we know, the numerical model of the diffuse incident field has application constraints. Nastran does not support it, and Abaqus does not support it in modal-space-based but in physical-space-based technical routes. Fortunately, Actran can model the diffuse incident filed in physical-space-based and modal-space-based technical routes. Furthermore, Actran can accept the eigenmodes supplied by Nastran. Thus, in modal-space-based technical routes, Actran can get accurate eigenmodes from very efficient eigenvalue extraction analyses by Nastran. Then, three simulation strategies for vibro-acoustic analyses of the metamaterial samples are available.

The first simulation strategy is the physical-space-based steady-state vibro-acoustic analysis with the "Ordinary Method" in Abaqus. The model's frequency dependency is ignored due to Abaqus limitations.

Then, the second simulation strategy is the physical-space-based steady-state vibro-acoustic analysis with the "Ordinary Method" in Actran, and the frequency dependency of the model is active in this case.

Finally, the third simulation strategy is the modal-space-based steady-state vibro-acoustic analysis through the cooperation of Nastran with Actran. The model's frequency dependency can be compensated by proper modal structural damping input at each independent mode. For calculating the proper modal damping values, Nastran's DMAP is a powerful tool (see section **7.4**).

17.2 Tests for Choosing the Best Simulation Strategy

17.2.1 Settings of the Fluid-Structure System

The testing sample for simulation strategy assessments is "Panel No.4" (see **Figure 15.1**), with 400 cells. Its size on the XY plane is 400×400 mm since the size of one cell on the XY plane is 20×20 mm, and the diameter of the aluminium bar is 8mm (see **Figure 15.2**). **Figure 17.1** displays the XY-plane view of the testing panel, and the left bottom side (indicated with a red point) is defined as the starting border on the XY plane. Thus, in **Figure 17.1**, the XY-coordinate values of the source point, the standoff point, and the response point are equal to (200, 200, Z).

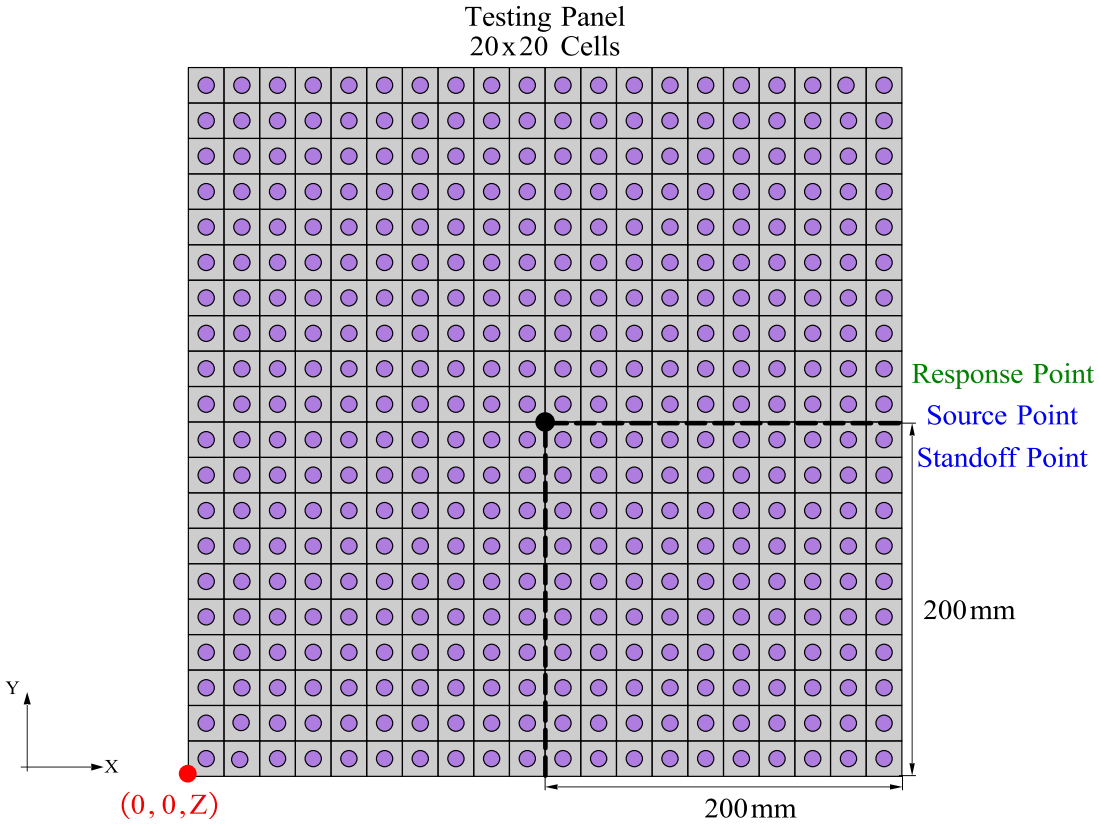


Figure 17.1: The XY-plane view of the testing "Panel No.4".

In **Figure 17.2**, subplot (a1) displays the layout of the structural-acoustic system in a 3D view. "A1" indicates the outside air domain, "A2" indicates the inside air domain, and "S" indicates the testing panel. Then, subplot (a2) is the loading condition by the diffuse incident filed on the YZ-plane view. We can see that the source point stays 300 mm from "Interface 1",

and the maximum incident angle is 120°. And the number of samples of the diffuse incident filed is 10.

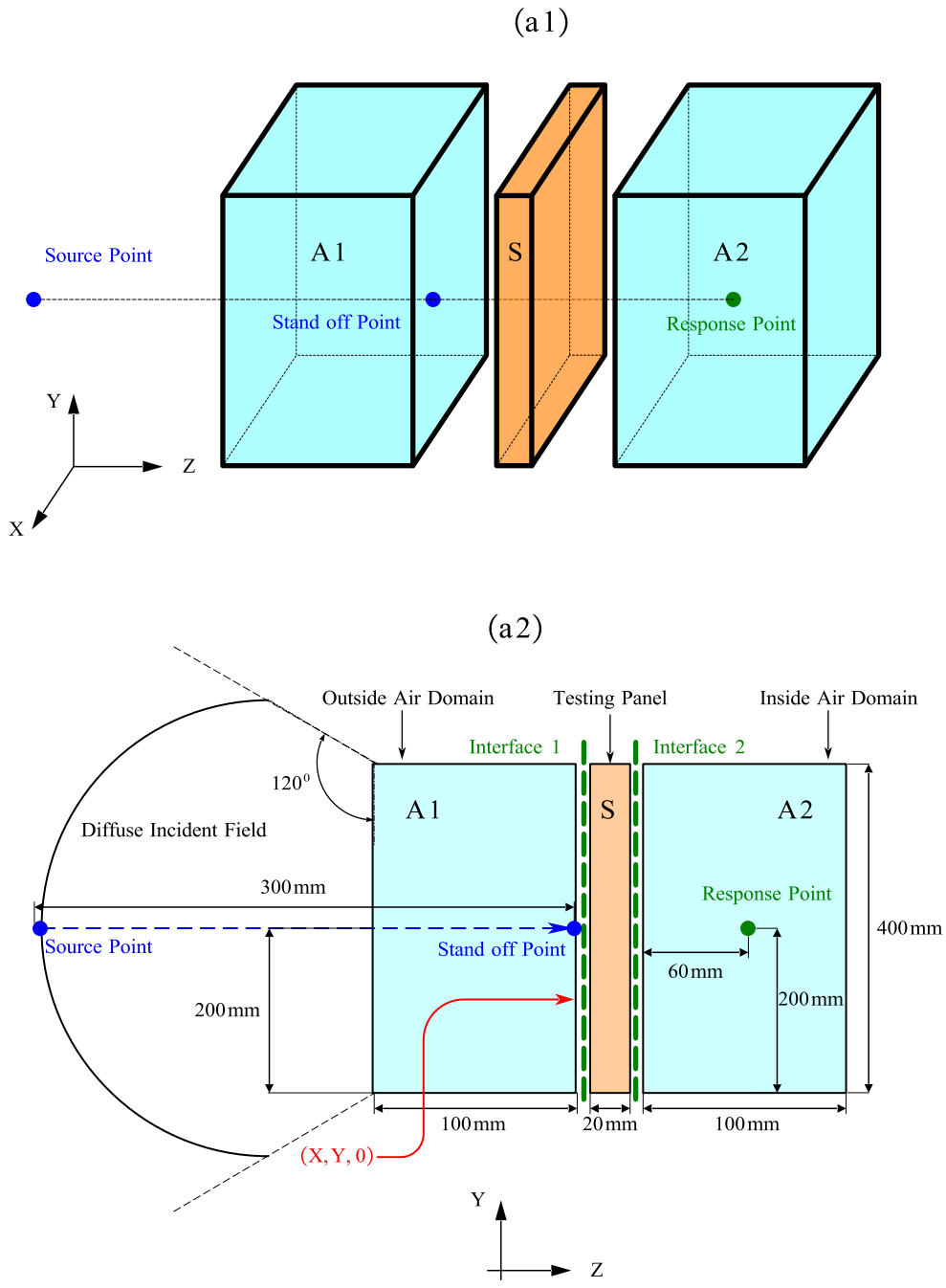


Figure 17.2: Loading condition of the fluid-structure system (basic structure).

Figure 17.2 displays the layout of the fluid-structure system. From left to right, the first part is the outside air domain; its size in XYZ coordinates is $400\times400\times100$ mm. Then the

second part is the interface between the outside air domain and the testing panel. Next, the third part is the testing panel; its size in XYZ coordinates is $400\times400\times20\,\mathrm{mm}$. Aside from the testing panel, the fourth part is the interface between the testing panel and the inside air domain. Lastly, the fifth part is the inside air domain; its size in XYZ coordinates is $400\times400\times100\,\mathrm{mm}$. "Interface 1" is the interface between the outside air domain and the testing panel, and the Z coordinate value of this interface is set to 0. Thus, the left side of "Interface 1" is the negative Z-coordinate-value domain, while the right is the positive Z-coordinate-value domain. Then, considering the YZ plane with the YX plane, the XYZ-coordinate values of the source point are (-300, 200, 200), of the standoff point are (0, 200, 200), and of the response point are (80, 200, 200).

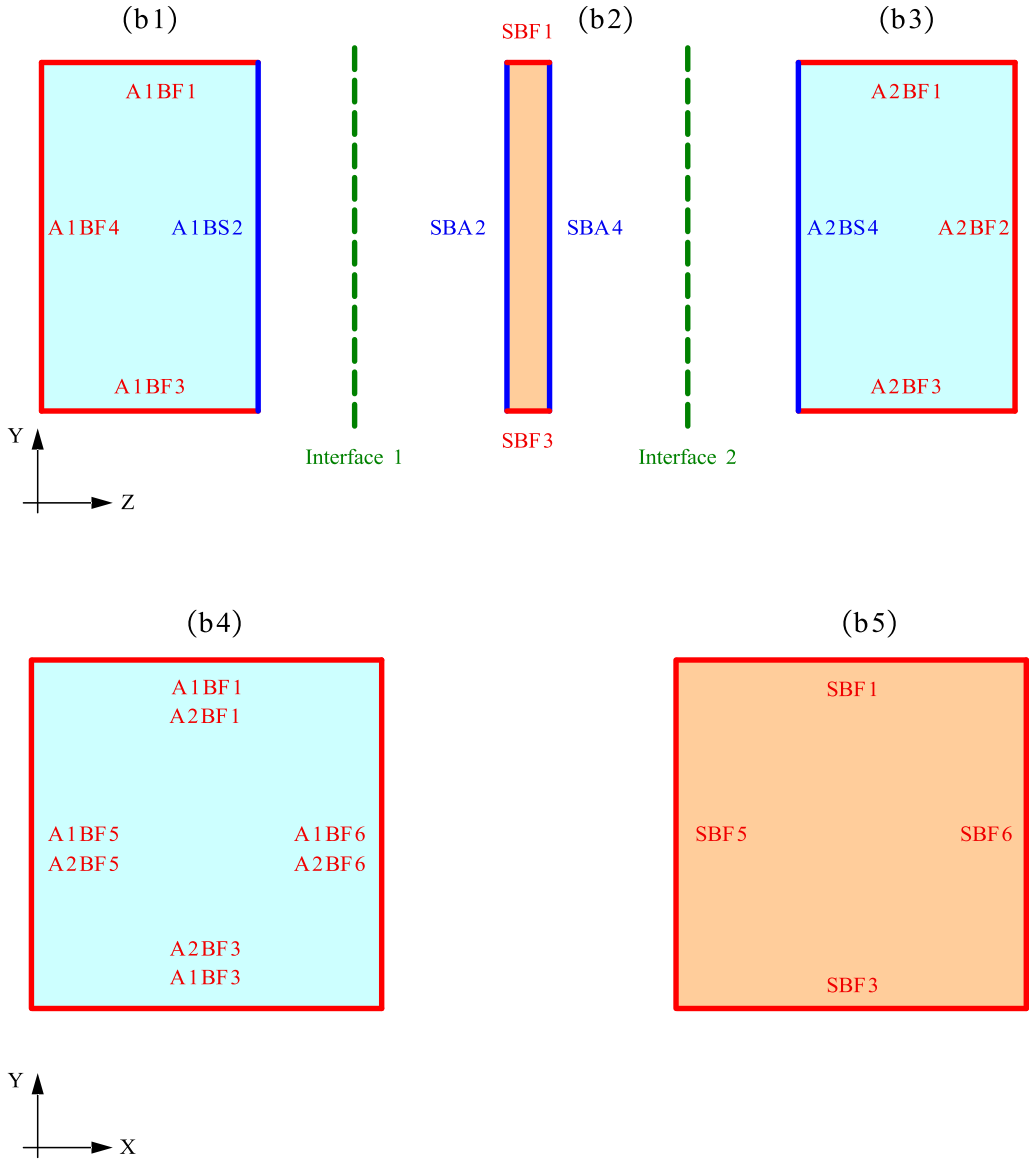


Figure 17.3: The boundary conditions of the fluid-structure system.

Figure 17.3 displays the boundary conditions of the structural-acoustic system. On the YZ plane, from left to right, there are three subplots.

The first is subplot (b1), representing the outside air domain with four boundary conditions. These four boundary conditions displayed on the YZ plane are in clockwise order. Specifically, from the first to the fourth, the boundary "A1BF1", highlighted in red, is the first boundary of domain "A1" in a fixed condition. Similarly, the boundary "A1BS2", highlighted in blue, is the second boundary of domain "A1", coupled with a structural boundary. Then, the boundary "A1BF3", highlighted in red, is the third boundary of domain "A1" in a fixed condition. Finally, the boundary "A1BF4", highlighted in red, is the fourth boundary of domain "A1" in a fixed condition.

The second is subplot (b2), representing the testing panel with four boundary conditions. These four boundary conditions displayed on the YZ plane are in anticlockwise order. Specifically, from the first to the fourth, the boundary "SBF1", highlighted in red, is the first boundary of domain "S" in a fixed condition. Similarly, the boundary "SBA2", highlighted in blue, is the second boundary of domain "S", coupled with an acoustic boundary. Then, the boundary "SBF3", highlighted in red, is the third boundary of domain "S" in a fixed condition. Finally, the boundary "SBA4", highlighted in blue, is the fourth boundary of domain "S", coupled with an acoustic boundary.

The third is subplot (b3), representing the inside air domain with four boundary conditions. These four boundary conditions displayed on the YZ plane are in clockwise order. Specifically, from the first to the fourth, the boundary "A2BF1", highlighted in red, is the first boundary of domain "A2" in a fixed condition. Similarly, the boundary "A2BS2", highlighted in red, is the second boundary of domain "A2" in a fixed condition. Then, the boundary "A2BF3", highlighted in red, is the third boundary of domain "A2" in a fixed condition. Finally, the boundary "A2BF4", highlighted in blue, is the fourth boundary of domain "A2", coupled with a structural boundary.

Between subplot (b1) and subplot (b2), "Interface 1", highlighted in green, is the surface holding the connection between the boundary "A1BS2" and the boundary "SBA2". Similarly, between subplot (b2) and subplot (b3), "Interface 2", highlighted in green, is the surface holding the connection between the boundary "SBA4" and the boundary "A1BS4".

In **Figure 17.3**, the section under the YZ-plane view is the XY-plane view. On the XY plane, from left to right, two subplots represent three domains and corresponding boundary conditions.

The first subplot (b4) represents the outside or inside air domain. These two domains have analogous boundary conditions regarding the XY-plane view. The boundaries "A1BF1", "A1BF3", "A1BF5", and "A1BF6", highlighted in red, are boundaries of domain "A1" in fixed conditions. In the same way, the boundaries "A2BF1", "A2BF3", "A2BF5", and "A2BF6", highlighted in red, are boundaries of domain "A2" in fixed conditions.

The second subplot (b5) represents the testing panel on the XY plane. The boundaries "SBF1", "SBF3", "SBF5", and "SBF6", highlighted in red, are boundaries of domain "S" in fixed conditions.

17.2.2 The Mathematical Model of Vibro-Acoustic Systems

As one of the main topics, this paper needs to accomplish a preliminary design of metamaterials. Thus, the arrangement of the testing environment must be simple enough. We can see that the crucial simulation items are the outside and inside air domains, the acoustic structure interfaces, the response point, and the acoustic source from the diffuse incident field. These simulation items are sufficient to test the metamaterial's dynamic performances.

Let us talk about the acoustic source in the first place. The diffuse incident field is analogous to a reverberant chamber (see subsection **10.2.5**). And the inside acoustic loads are homogenous. In simulation practice, the FEM software does not build a diffuse incident field but applies a homogenous load to the assigned surfaces (see **Figure 10.1**). In FEM software, such as Abaqus or Actran, the acoustic load generated by the diffuse incident field must be directly loaded to the fluid-structure interface (see **Figure 17.2**). For the vibro-acoustic analyses in this paper, the acoustic load from the diffuse incident field is directly applied to the interfaces between the outside air domain and the testing panel. Then, the testing panel is driven by the acoustic load. At the same time, the motivated panel drives the interface between the panel and the inside air domain to generate the acoustic pressure in the inside air domain. It is the case that the dynamic motivation of the intermediate panel dominates the acoustic pressure of the inside air domain. If there are discrepancies between the outside and inside acoustic pressures, it is the intermediate panel's soundproof effect.

In the second place, let us talk about the acoustic domain. Please see **Figure 17.3** in subsection **17.2.1** and its correlated descriptions in the contents below the figure. We can see that the outer boundaries of the two air domains are set to be fixed, and the flexible boundaries are the fluid-structure interfaces. Thus, the inside and the outside air domains are analogous to two reverberant chambers connected with an intermediate panel. Furthermore, the inside and outside air domains are considered conservative systems. Thus, no energy dissipation in the two air domains. As a result, the only energy dissipation term in this fluid-structure system is the structural damping of the intermediate panel.

In the third place, considering the specific settings of the fluid-structure system, the mathematical models of the air and structure domains need to be modified based on the fundamental forms. Please review equation **(10.51)** in subsection **11.4.2** and equation **(10.52)** in subsection **11.4.3**; the modification of the scattered wave equation of the air domain can be expressed as:

$$\begin{aligned} &-\omega^2\left[[M_{\mathrm{f}}]^{PQ}+[M_{\mathrm{fr}\cup\mathrm{fi}}]^{PQ}\right]\left\{\tilde{P}_f\right\}_S^Q \\ &+\left[i\omega\left[[C_{\mathrm{f}}]^{PQ}+[C_{\mathrm{fr}\cup\mathrm{fi}}]^{PQ}\right]+i\left[C_{\mathrm{f}(\gamma_f)}\right]^{PQ}\right]\left\{\tilde{P}_f\right\}_S^Q \\ &+[K_{\mathrm{f}}]^{PQ}\left\{\tilde{P}_f\right\}_S^Q \\ &=[S_{\mathrm{fs}}]^{PM}\left\{\tilde{T}_s\right\}_S^M+\left\{\tilde{P}_f\right\}^P \end{aligned} \tag{16.1}$$

Once the reactive and non-reflecting boundary conditions are neglected, the corresponding acoustic inertia $[M_{\mathrm{fr}\cup\mathrm{fi}}]^{PQ}$ and the acoustic damping $[C_{\mathrm{fr}\cup\mathrm{fi}}]^{PQ}$ are ignored. At the same

time, neglecting the acoustic damping terms $[C_{\mathrm{f}}]^{PQ}$ and $[C_{\mathrm{f}(\eta_f)}]^{PQ}$. Thus, the modified scattered wave formulation in the air domain can be expressed as:

$$-\omega^2[M_{\mathrm{f}}]^{PQ}\left\{\tilde{P}_f\right\}_S^Q+[K_{\mathrm{f}}]^{PQ}\left\{\tilde{P}_f\right\}_S^Q=[S_{\mathrm{fs}}]^{PM}\left\{\tilde{T}_s\right\}_S^M \tag{16.2}$$

Then, the modification of the structure domain equation can be expressed as:

$$\begin{aligned}&-\omega^2[M_{\mathrm{s}}]^{NM}\left\{\tilde{U}_s\right\}^M\\&+\left[i\omega\left[[C_{\mathrm{s(m)}}]^{NM}+[C_{\mathrm{s(k)}}]^{NM}\right]+i[C_{\mathrm{s}(\eta_s)}]^{NM}\right]\left\{\tilde{U}_s\right\}^M\\&+[K_{\mathrm{s}}]^{NM}\left\{\tilde{U}_s\right\}^M\\&=-[S_{\mathrm{fs}}]^{NQ}\left(\left\{\tilde{P}_f\right\}_I^Q+\left\{\tilde{P}_f\right\}_S^Q\right)+\left\{\tilde{p}_s\right\}^N\end{aligned} \tag{16.3}$$

Without considering the viscous damping effects, the viscous damping correlated terms $[C_{\mathrm{s(m)}}]^{NM}$ and $[C_{\mathrm{s(k)}}]^{NM}$ are neglected. Then, structural loading on the structure surface by the direct external traction force $\{\tilde{p}_s\}^N$ is neglected. Thus, the modified structural field governing equation can be expressed as:

$$\begin{aligned}&-\omega^2[M_{\mathrm{s}}]^{NM}\left\{\tilde{U}_s\right\}^M+i[C_{\mathrm{s}(\eta_s)}]^{NM}\left\{\tilde{U}_s\right\}^M+[K_{\mathrm{s}}]^{NM}\left\{\tilde{U}_s\right\}^M\\&=-[S_{\mathrm{fs}}]^{NQ}\left(\left\{\tilde{P}_f\right\}_I^Q+\left\{\tilde{P}_f\right\}_S^Q\right)\end{aligned} \tag{16.4}$$

Based on the scattered wave theory, we know that the incident wave is the known parameter, and the scattered wave is the unknown parameter. In equation **(16.2)**, the unknown parameters are the scattered acoustic pressure vector $\{\tilde{P}_f\}_S^Q$ and the scattered traction force vector $\{\tilde{T}_s\}_S^M$. At the same time, in equation **(16.4)**, the unknown parameter is the structure's displacement vector $\{\tilde{U}_s\}^M$, and the known parameter is the incident acoustic pressure vector $\{\tilde{P}_f\}_I^Q$. Then, with a new condition in equation **(10.49)**

$$-\omega^2\{\tilde{U}_s\}^M=\{\tilde{T}_s\}_I^M+\{\tilde{T}_s\}_S^M \tag{16.5}$$

The incident traction force vector $\{\tilde{T}_s\}_I^M$ is the known parameter. Finally, all unknown parameters can be solved by combining equations **(16.2)**, **(16.4)**, and **(16.5)**. And the frequency response function of the response point is one of the terms in the scattered acoustic pressure vector $\{\tilde{P}_f\}_S^Q$.

17.2.3 Section Conclusions

Based on the elaborations in subsection **17.1.4**, we know three simulation strategies may be choices. The first is the physical-space-based steady-state vibro-acoustic analysis with the "Ordinary Method" in Abaqus. The model's frequency dependency is ignored due to Abaqus limitations. The second one is the physical-space-based steady-state vibro-acoustic analysis with the "Ordinary Method" in Actran, and the frequency dependency of the model is active in this case.

At the same time, the third one needs the cooperation between Nastran and Actran. The cooperation is that Nastran supplies modal parameters, and Actran executes modal-space-based vibro-acoustic analyses. Nastran uses the "ACMS Method" to do eigenvalue extraction analyses. The eigenmodes of the inside air domain, the outside air domain, the intermediate panel, and the base-state modal damping are products of Nastran's analyses. In addition, the extracted base-sate modal damping needs externally add a frequency function (see subsection **7.4.2**). After that, the extracted eigenmodes and the externally treated modal damping are sent into Actran. Then, Actran executes the modal-space-based steady-state vibro-acoustic analysis.

For these three simulation strategies, the objective system is introduced in subsection **17.2.1,** and the corresponding mathematical model is introduced in subsection **17.2.2**. **Figure 17.4** displays the fluid-pressure response functions of the response point through the three simulation strategies. More specifically, subplot (a) displays the results through the first simulation strategy, and the abbreviation "M. NFreq" in the title indicates the non-frequency-dependent metamaterial. At the same time, subplot (b) displays the results through the second simulation strategy, and the abbreviation "M. Freq" in the title indicates the frequency-dependent metamaterial. Then, subplot (c) displays the results through the third simulation strategy, and the abbreviation "M. MFreq" in the title indicates the modal-frequency-dependent material. In these three subplots, the abbreviation "FP" means the fluid pressure, and the unit is Pa (pascal).

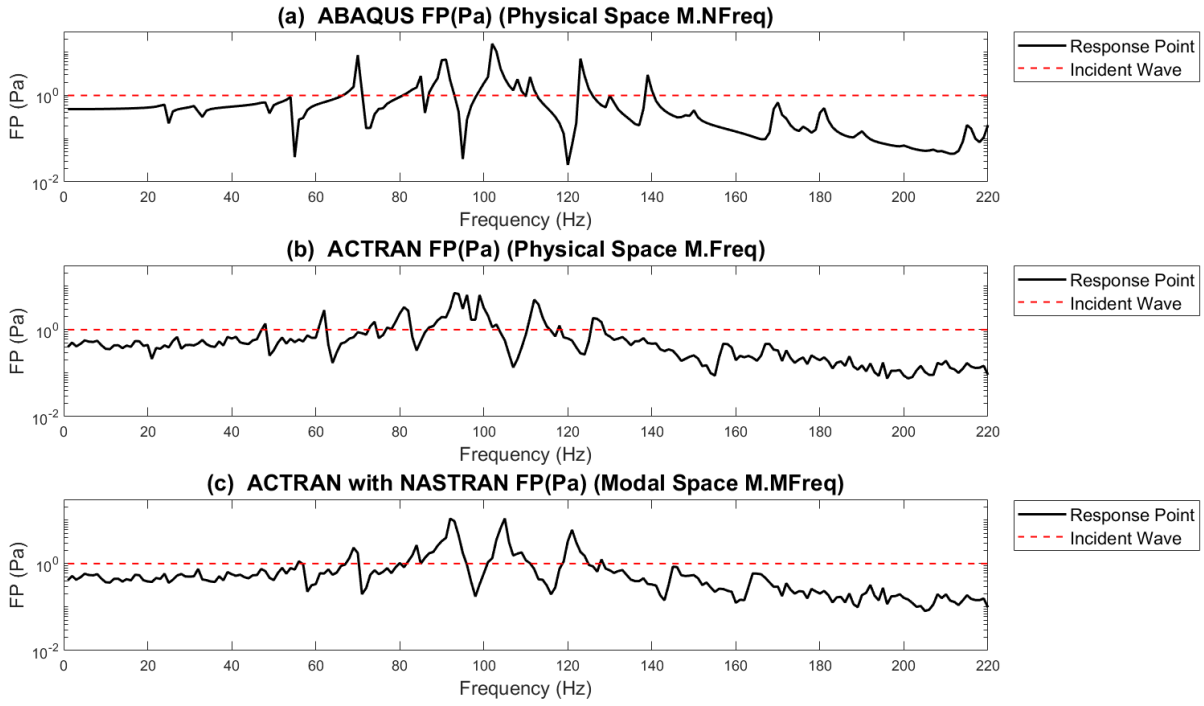


Figure 17.4: FRF of fluid response point (Abaqus vs Actran vs Actran & Nastran).

In **Figure 17.4**, the red dashed line indicates the incident wave's pressure amplitude of 1Pa. Based on the results, we can see that the shapes of the frequency response functions in the three subplots are analogous. Although, there are obvious discrepancies between these three frequency response functions. The first simulation strategy adopts a non-frequency-dependent metamaterial panel in Abaqus. The damping effect, in this case, is underestimated. Thus, higher amplitudes of resonant and anti-resonant points emerge in subplot (a). Then, the second simulation strategy adopts a frequency-dependent metamaterial panel in Actran. There is no estimation in this case. Thus, subplot (b) results can be seen as a reference. At last, the third simulation strategy adopts the modal-frequency-dependent "Composite Modal Damping" to replace the frequency-dependent structural damping of the metamaterial. Then, the results in subplot (c) show that the damping values at the frequency range of $40-80\,\mathrm{Hz}$ are overestimated, and at the frequency range of $80-130\,\mathrm{Hz}$ are underestimated.

Besides relative accuracy, efficiency is another important factor in choosing the best simulation strategy. **Table 17.1** displays the time consumption of the three simulation strategies for the vibro-acoustic analysis of the system shown in **Figure 17.2**. For physical-space-based and modal-space-based vibro-acoustic analyses, the fluid-structure system has 70602 fluid mesh grids and 56405 solid mesh grids. And the frequency range of the eigenvalue extraction analyses for the panel is 0–250Hz and for air is 0–1500Hz. Then, for modal-space-based analyses, the air parts have 26 eigenmodes, and the solid part has 208 eigenmodes. After that, the frequency range of vibro-acoustic analyses is 0–220Hz, and the number of excitation frequencies is 220.

We can see that the third simulation strategy is much faster than the other two. And it is reasonable to believe that the bigger the finite element model is, the faster the third simulation strategy will be. The reason is that the eigenvalue extraction process occupies a small part of the time consumption, and the number of extracted eigenmodes is much smaller than the number of physical DOFs. The time consumption of modal-space-based analyses is dependent on the number of eigenmodes. In contrast, the time consumption of physical-space-based analyses depends on the number of physical DOFs. Finally, considering the accuracy and efficiency, the third simulation strategy is the best choice, and it will be applied in the assessment tests of the preliminary design of metamaterial samples.

Table 17.1: The time consumption (second) of vibro-acoustic analyses (Abaqus vs Actran vs Actran & Nastran).

	Eigenvalue Extraction	Dynamic Analysis	Full Process
Abaqus (Physical Space M. NFreq)	0	7364.5s	7364.5s
Actran (Physical Space M. Freq)	0	6109.0s	6109.0s
Actran with Nastran (Modal Space M. MFreq)	97.4s	921.0s	1018.4s

17.3 Specific Simulation Practices for Testing Samples

According to the assessments of simulation strategies in subsection **17.2.3**, the best simulation strategy is the cooperation of Nastran with Actran in modal-space-based steady-state vibro-acoustic analyses.

In this section, the size of testing panels is fixed at $400\times400\times20\,\text{mm}$ regarding XYZ-coordinates, and these panels consist of 400 cells. Then, except for the panel size, the panel's lateral-boundary-constrained conditions, the panel's aluminium inclusions, and the number of panel layers will be three control parameters in the preliminary design of metamaterial samples. Firstly, numerical verifications in subsection **17.3.1** correspond to the panel's lateral-boundary-constrained conditions. Secondly, numerical verifications in subsection **17.3.3** correspond to the panel's aluminium inclusions. At last, numerical verifications in subsection **17.3.4** correspond to the number of panel layers.

For comparison, a compensation group contains the panels with pure melamine foam with different lateral-boundary-constrained conditions (see subsection **17.3.2**). In the following tests, the loading condition is the same as in section **17.2**, and the response point does not change (see **Figure 17.2**). Finally, the frequency response function of the fluid response point will be displayed in three numerical forms. The first one is the fluid pressure (FP), which is accounted for with the unit of Pascal (Pa). The second one is the sound pressure level (SPL), which is accounted for with the Decibel unit (dB). At last, the third one is the sound transmission loss (STL), which is accounted for with the Decibel unit (dB).

Supposing the amplitude of the scattered acoustic pressure at the response point is $\hat{P}_{fs}$, the amplitude of the incident pressure is $\hat{P}_{fi}$, and the amplitude of the reference pressure is $\hat{P}_{REF}=2e^{-5}Pa$. Then, the sound pressure level can be expressed as:

$$\text{SPL}\,(\text{dB})=20\cdot\log_{10}\left(\frac{\hat{P}_{fs}}{\hat{P}_{REF}}\right) \tag{16.6}$$

At the same time, the sound transmission loss can be expressed as:

$$\text{STL}\,(\text{dB})=20\cdot\log_{10}\left(\frac{\hat{P}_{fi}}{\hat{P}_{fs}}\right) \tag{16.7}$$

17.3.1 Constrained-Lateral-Boundary-Controlled Metamaterial Panels

In this control group, the metamaterial panels contain aluminium bars with a diameter of 8mm, and we know the full size of the panel is $400\times400\times20\,\text{mm}$. Then, supposing using an ideally rigid casing or casing matrix to contain the metamaterial and constrain the lateral boundaries of metamaterial panels. And once using a casing matrix, the casing matrix will physically subdivide the whole panel and constrain the lateral boundaries of each subpart. Thus, the variable in this control group is constrained lateral boundaries. **Figure 17.5** to **Figure 17.8** are 3D views that illustrate four lateral-boundary-constrained conditions. And **Figure 17.9** is a 2D view that further illustrates the comparison of the lateral-boundary-constrained conditions. Specifically, **Figure 17.5** represents the first constraint condition, and the whole panel's lateral boundaries are constrained. **Figure 17.5** corresponds to subplots (a1) and (a2) of **Figure 17.9**.

Similarly, **Figure 17.6** represents the second constraint condition, and the lateral boundaries of four subparts are constrained. **Figure 17.6** corresponds to subplots (b1) and (b2) of **Figure 17.9**. Then, **Figure 17.7** represents the third constraint condition, and the lateral boundaries of sixteen subparts are constrained. **Figure 17.7** corresponds to subplots (c1) and (c2) of **Figure 17.9**. After that, **Figure 17.8** represents the fourth constraint condition, and the lateral boundaries of four hundred subparts are constrained. **Figure 17.8** corresponds to subplots (d1), (d2) and (d3) of **Figure 17.9**.

It is necessary to note that the ideal casing or casing matrix will not be modelled in simulations. Since the casing or casing matrix is supposed to be ideally rigid and they can completely fix the panel's lateral boundaries through some boundary techniques. The equivalent operation is to set the DOFs of the lateral boundaries to zero.

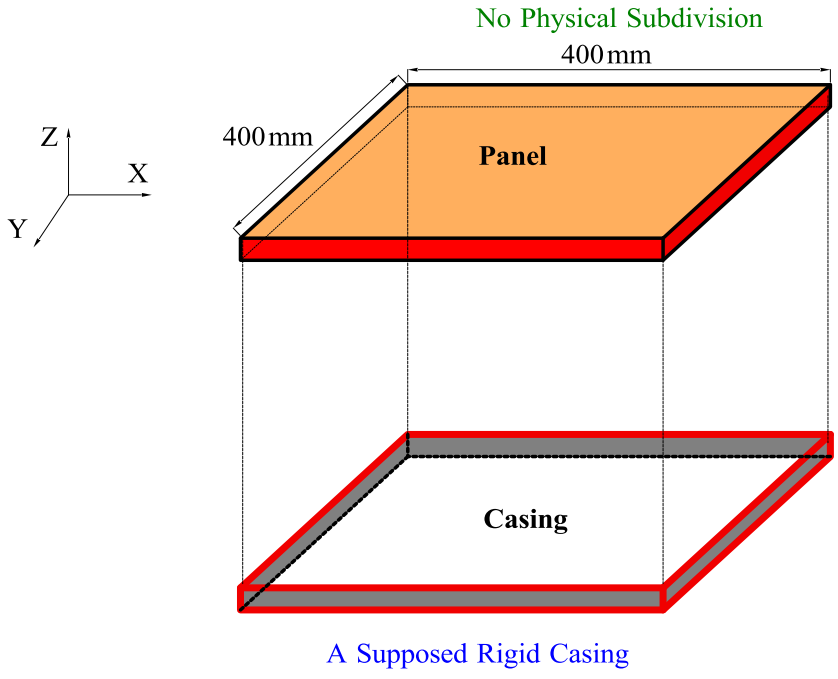


Figure 17.5 Lateral boundaries' constrained condition (the whole panel).

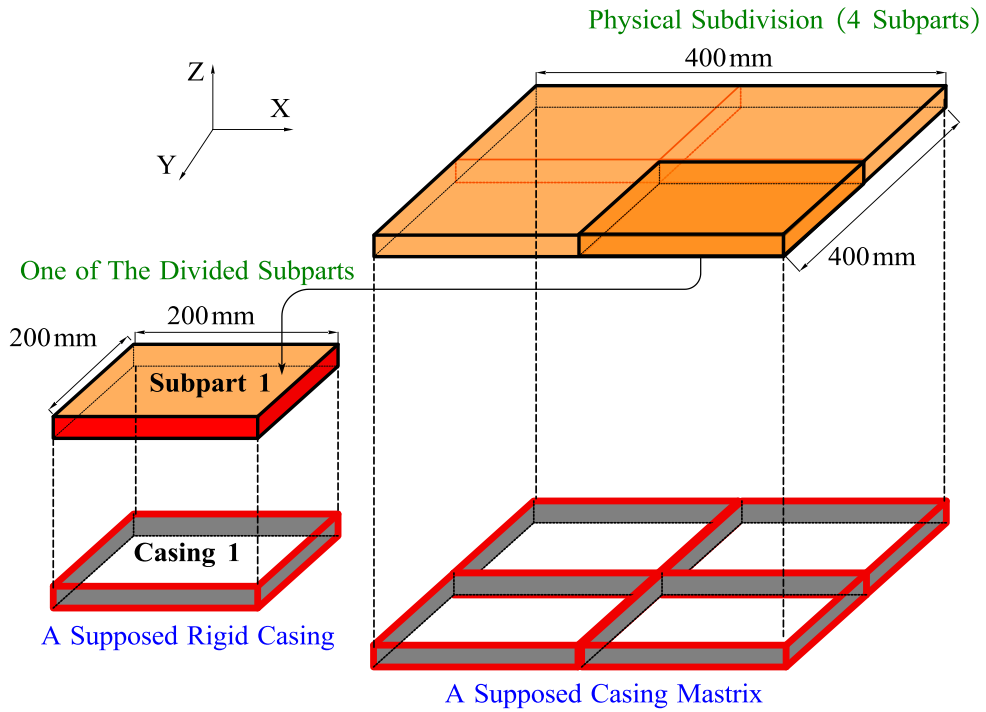


Figure 17.6 Lateral boundaries' constrained condition (a panel is physically subdivided into 4 subparts).

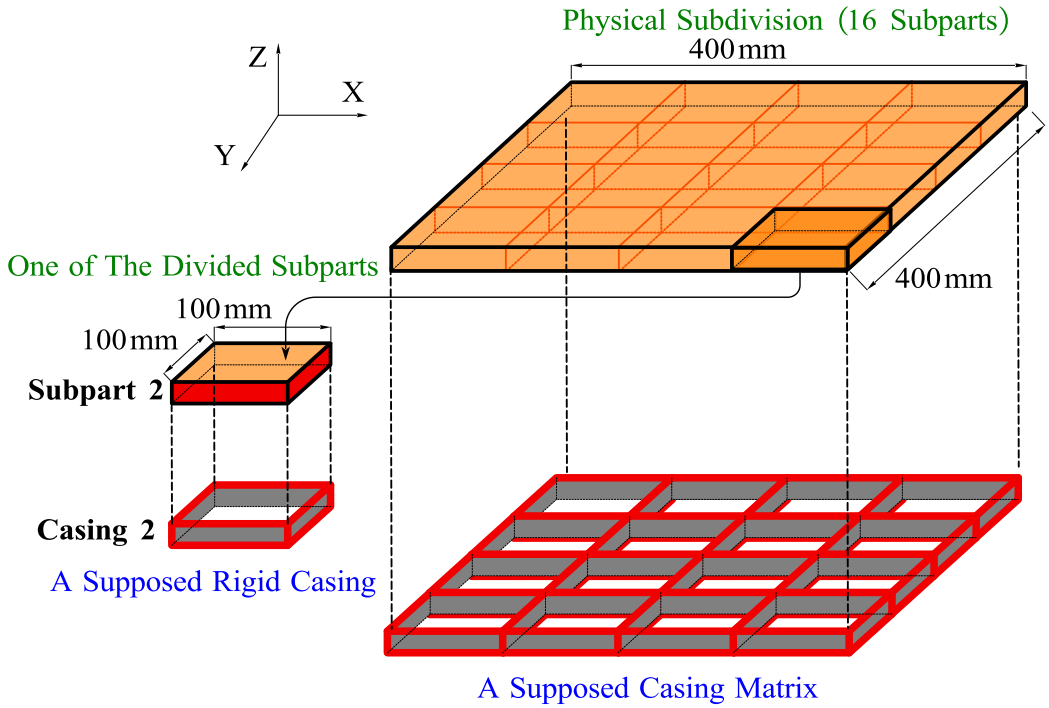


Figure 17.7 Lateral boundaries' constrained condition (a panel is physically subdivided into 16 subparts).

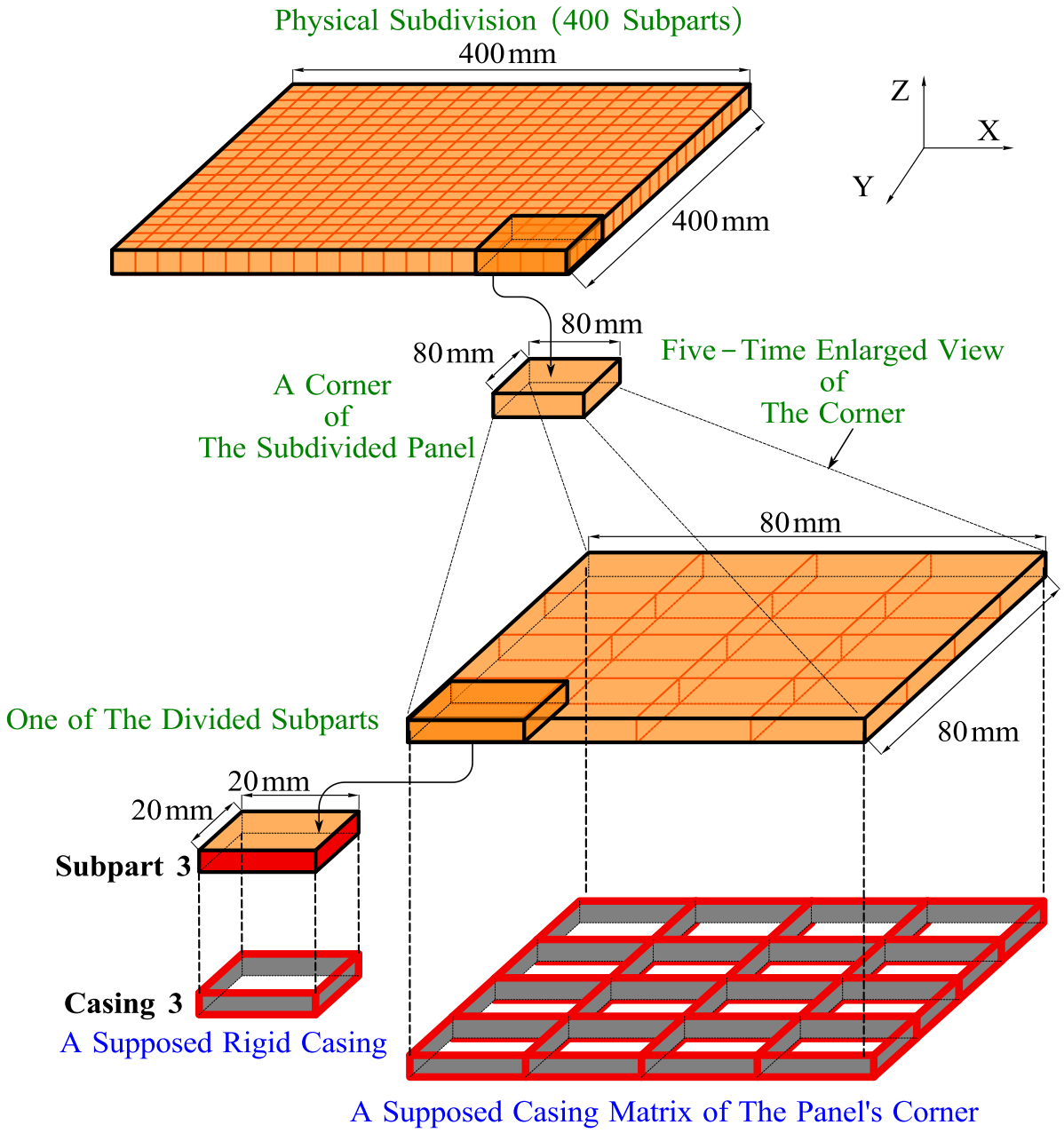


Figure 17.8 Lateral boundaries' constrained condition (a panel is physically subdivided into 400 subparts).

In **Figure 17.8**, it is necessary to note that all the subparts of the panel are supposed to be constrained, not only the corner part. The highlighted corner part is an example to show one of the subsections of the panel, which means the lateral-boundary-constrained forms of other subsections are analogous. In this case, the lateral boundaries of every unit cell of the metamaterial panel are constrained. At the same time, the highlighted corner part in **Figure 17.8** corresponds to the subplot (d3) of **Figure 17.9**. They are five-time enlarged views.

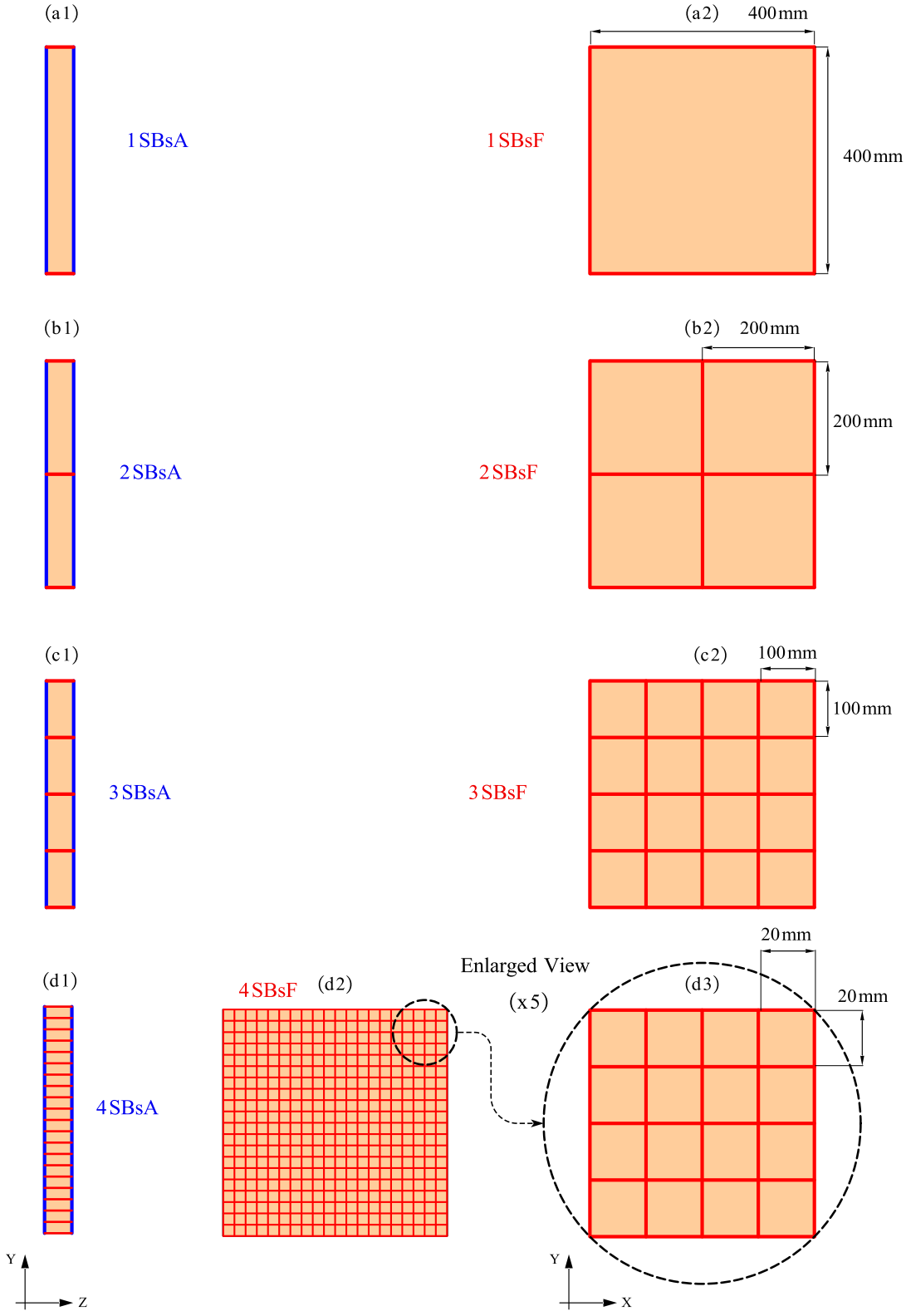


Figure 17.9: Illustrations of the four different lateral-boundary-constrained conditions.

In **Figure 17.9**, subplots (a1) and (a2) represent the first lateral-boundary-constrained condition. They are YZ and XY-plan views of the testing panels. Between these two plots, "1SBsA" indicates the structural-acoustic boundaries, highlighted in blue. And "1SBsF" indicates the constrained boundaries, highlighted in red.

Similarly, subplot (b1) on the YZ-plane and subplot (b2) on the XY-plane represent the second lateral-boundary-constrained condition. Between these two subplots, "2SBsA" indicates the structural-acoustic boundaries, highlighted in blue. And "2SBsF" indicates the constrained boundaries, highlighted in red.

Then, subplot (c1) on the YZ-plane and subplot (c2) on the XY-plane represent the third lateral-boundary-constrained condition. "3SBsA" indicates the structural-acoustic boundaries between these two subplots, highlighted in blue. And "3SBsF" indicates the constrained boundaries, highlighted in red.

After that, subplot (d1) on the YZ-plane and subplot (d2) on the XY-plane represent the fourth lateral-boundary-constrained condition. Furthermore, subplot (d3) on the XY-plane is a five-time enlarged view of one corner of the subplot (d2). "4SBsA" indicates the structural-acoustic boundaries between these two subplots, highlighted in blue. And "4SBsF" indicates the constrained boundaries, highlighted in red.

The following four figures from **Figure 17.10** to **Figure 17.13** correspond to the first (see **Figure 17.5**), the second (see **Figure 17.6**), the third (see **Figure 17.7**), and the fourth (see **Figure 17.8**) lateral-boundary-constrained conditions introduced in the upper paragraph. Each figure contains three subplots. We can see that subplot (a) is the fluid pressure (FP) with the unit of Pa. Then, subplot (b) is the sound pressure level (SPL) with the dB unit. Lastly, subplot (c) is the sound transmission loss (STL) with the dB unit. In these subplots, the abbreviations have specific meanings. "P400" means a 400-cell panel with the first lateral-boundary-constrained condition; "P400.CONS2" means a 400-cell panel with the second lateral-boundary-constrained condition. Then, "P400.CONS3" means a 400-cell panel with the third lateral-boundary-constrained condition; "P400.CONS4" means a 400-cell panel with the fourth lateral-boundary-constrained condition. Finally, "AL8mm" means the aluminium bar's diameter is 8mm.

Based on the results of the first constraint condition in **Figure 17.10**, we can see that the resonant and anti-resonant points are mainly distributed in the range of $65-125\,\mathrm{Hz}$. The sound transmission loss under the frequency value of 65 Hz is no more than 10 dB. And the sound transmission loss over the frequency value of 125 Hz is no more than 20 dB.

Based on the results of the second constraint condition in **Figure 17.11**, we can see that the resonant and anti-resonant points are mainly distributed in the range of $70-125\,\mathrm{Hz}$. The sound transmission loss under the frequency value of 70 Hz is no more than 10 dB. And the sound transmission loss over the frequency value of 125 Hz fluctuates around 10 dB.

Based on the results of the third constraint condition in **Figure 17.12**, we can see one resonant point at the frequency value of around 125 Hz. The sound transmission loss under the frequency value of 100 Hz changes from around 10 dB to 18 dB. And the sound transmission loss over the frequency value of 150 Hz changes from around 20 dB to 40 dB.

Based on the results of the fourth constraint condition in **Figure 17.13**, we can see no resonant point in the given frequency range. The sound transmission loss in the given frequency range is between 40 dB and 60 dB and close to 50 dB.

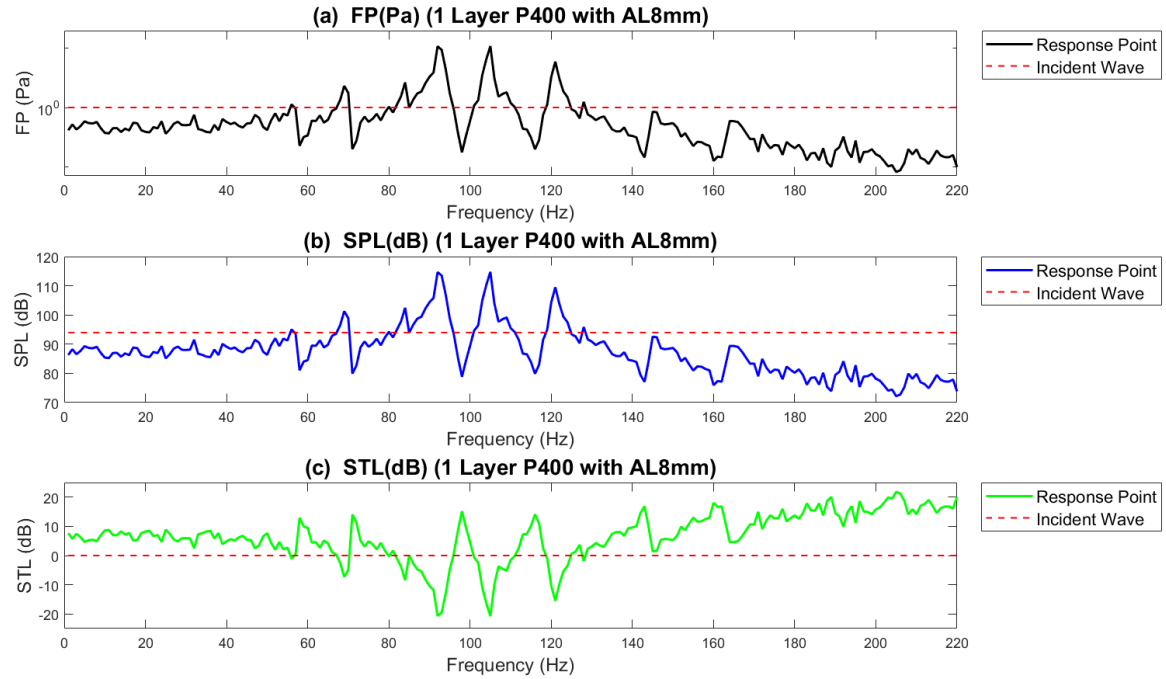


Figure 17.10: FRF of the fluid response point (l Layer P400 AL8).

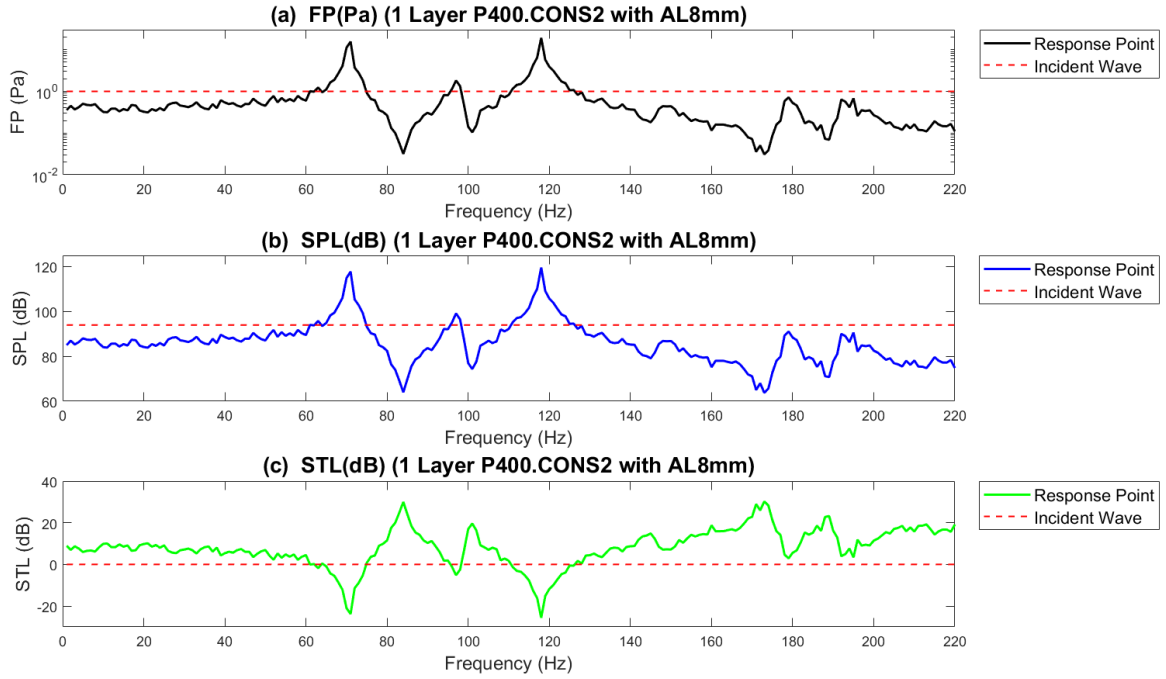


Figure 17.11: FRF of the fluid response point (l Layer P400.CONS2 AL8).

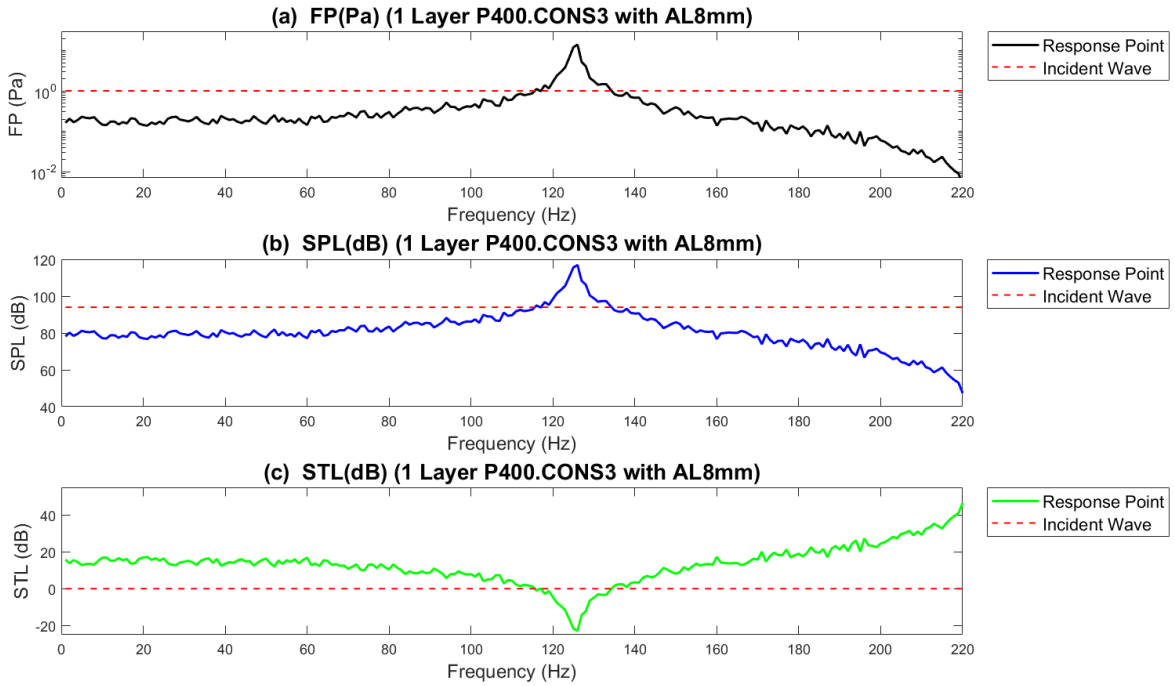


Figure 17.12: FRF of the fluid response point (l Layer P400.CONS3 AL8).

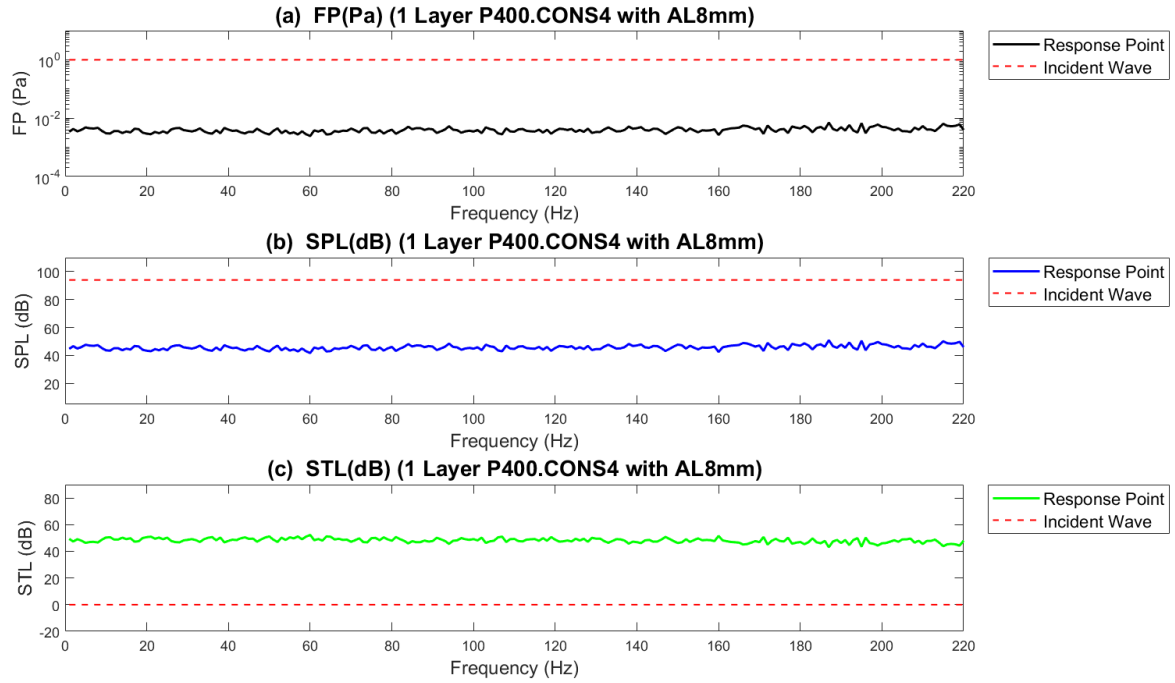


Figure 17.13: FRF of the fluid response point (l Layer P400.CONS4 AL8).

17.3.2 Constrained-Lateral-Boundary-Controlled Melamine Panels

In this control group, the melamine panels' full size is $400\times400\times20\,\text{mm}$. Then, supposing using an ideally rigid casing or casing matrix to constrain the lateral boundaries of metamaterial panels. And once using a casing matrix, the casing matrix will physically subdivide the whole panel and constrain the lateral boundaries of each subpart. Thus, the variable in this control group is constrained lateral boundaries. And the lateral-boundary-constrained conditions are the same as the ones in subsection **17.3.1** (see **Figure 17.5** to **Figure 17.9**).

The following three figures from **Figure 17.14** to **Figure 17.17** correspond to the first (see **Figure 17.5**), the second (see **Figure 17.6**), the third (see **Figure 17.7**) and the fourth (see **Figure 17.8**) lateral-boundary-constrained conditions. Each figure contains three subplots. We can see that subplot (a) is the fluid pressure (FP) with the unit of Pa. Then, subplot (b) is the sound pressure level (SPL) with the dB unit. Lastly, subplot (c) is the sound transmission loss (STL) with the dB unit. In these subplots, the abbreviations have specific meanings: "Melamine" (a melamine panel with the first lateral-boundary-constrained condition), "Melamine CONS2" (a melamine panel with the second lateral-boundary-constrained condition), "Melamine CONS3" (a melamine panel with the third lateral-boundary-constrained condition), "Melamine CONS4" (a melamine panel with the fourth lateral-boundary-constrained condition).

Based on the results of the first constraint condition in **Figure 17.14**, we can see that there are no resonant or anti-resonant points in the frequency range of $0-220\,\text{Hz}$. The sound transmission loss is around the range of $5-10\,\text{dB}$.

Based on the results of the second constraint condition in **Figure 17.15**, we can see that there are no resonant or anti-resonant points in the frequency range of $0-220\,\text{Hz}$. The sound transmission loss is around the range of $6-11\,\text{dB}$.

Based on the results of the third constraint condition in **Figure 17.16**, we can see that there are no resonant or anti-resonant points in the frequency range of $0-220\,\text{Hz}$. The sound transmission loss is around the range of $12-17\,\text{dB}$.

Based on the results of the fourth constraint condition in **Figure 17.17**, we can see no resonant point in the given frequency range. The sound transmission loss in the given frequency range is between 60 dB and 80 dB, close to 70 dB.

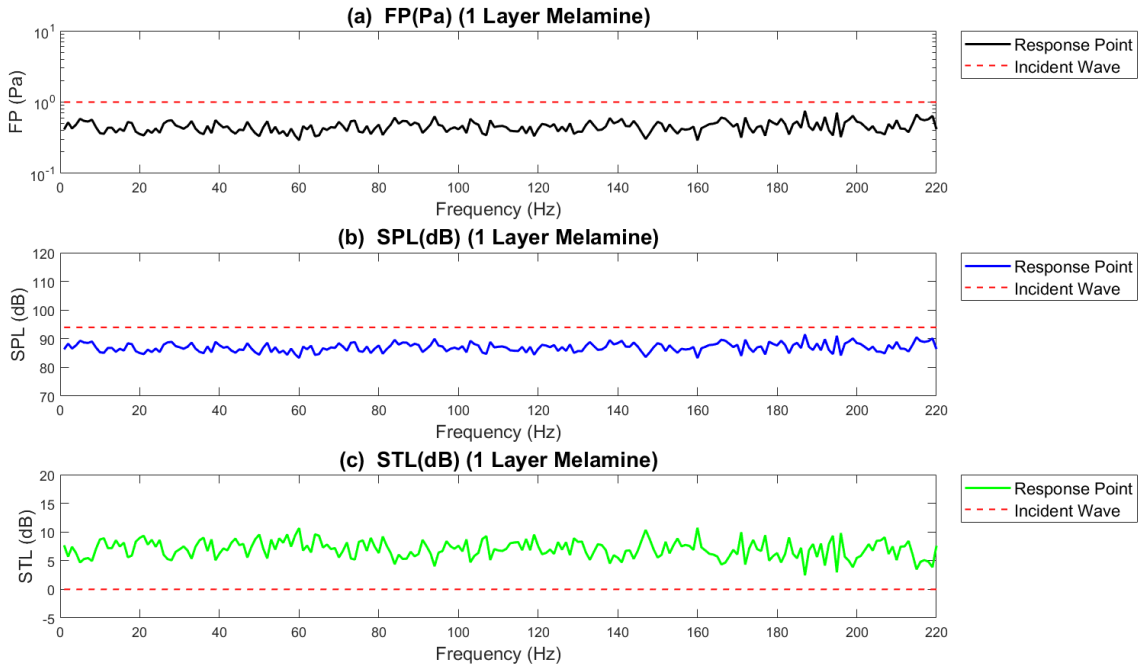


Figure 17.14: FRF of the fluid response point (l Layer Melamine).

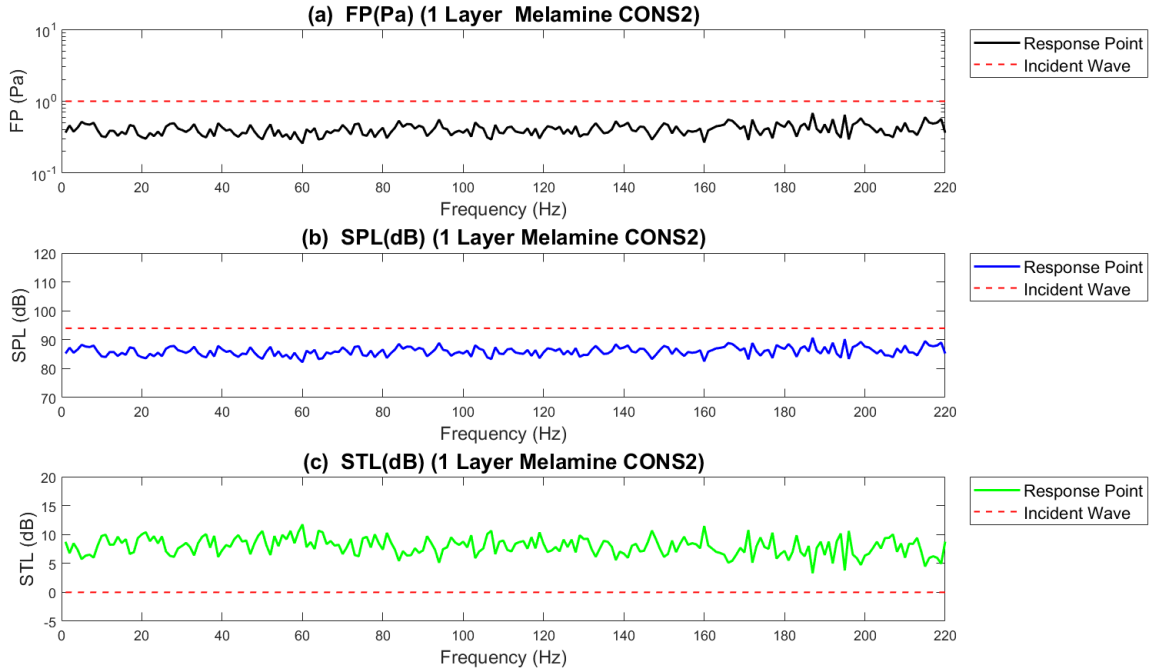


Figure 17.15: FRF of the fluid response point (l Layer Melamine CONS2).

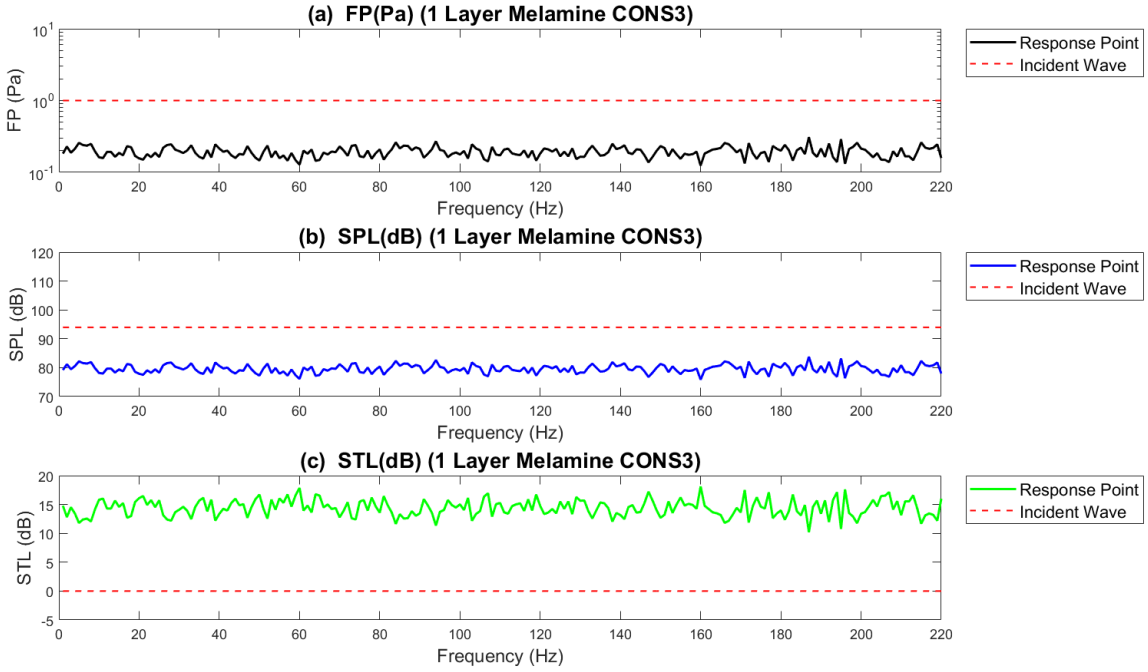


Figure 17.16: FRF of the fluid response point (l Layer Melamine CONS3).

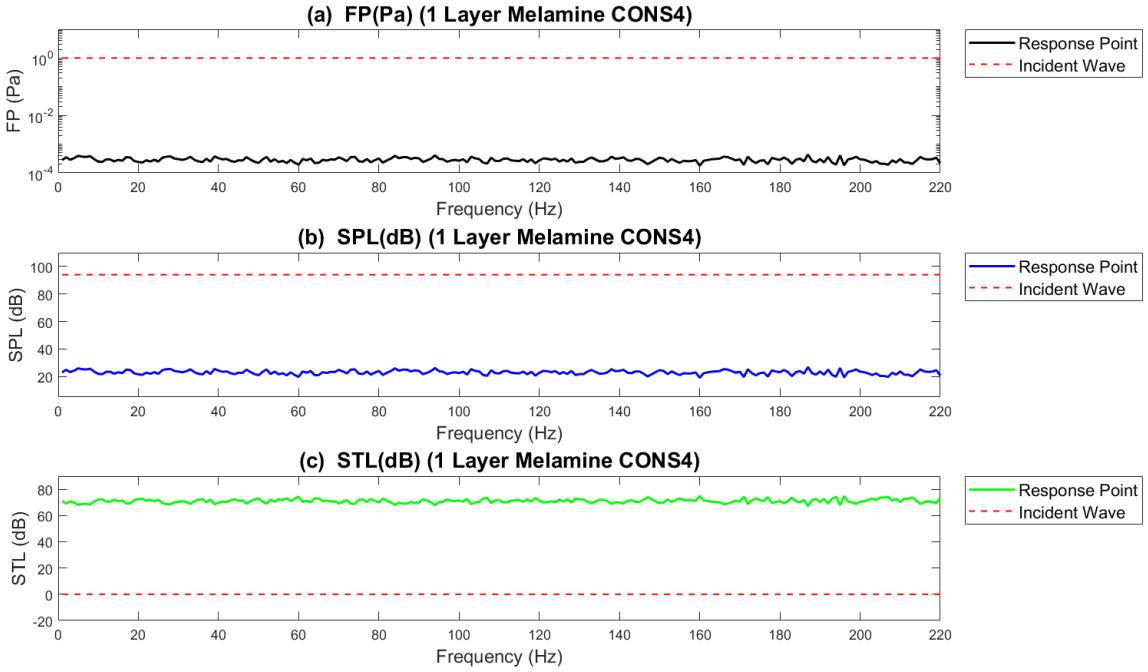


Figure 17.17: FRF of the fluid response point (l Layer Melamine CONS4).

17.3.3 Aluminium-Inclusion-Controlled Metamaterial Panels

In this control group, the variable is the aluminium bar inclusions. The testing panels are subdivided into six cases from the lowest to the highest inclusion. In the first case, the testing panel consists solely of melamine foam. From the second case to the fifth, aluminium bar inclusion increases with the bar's diameter increases. The aluminium bars' diameters are 4mm, 8mm, 12mm, and 16mm from the smallest to the biggest. The testing panels in the first five cases are congruent with the ones in chapter **16**. For more information about the testing panels, please review **Figure 16.1** and **Figure 16.2**. Then, as compensation, the sixth case is the panel consisting solely of aluminium alloy. The lateral-boundary-constrained condition of these six panels is the first condition introduced in **Figure 17.5** and subplots (a1) and (a2) of **Figure 17.9**.

We notice that the metamaterial panel contains aluminium bars with a diameter of 8mm is the same as the one with the frequency response function of **Figure 17.10**. And the pure melamine panel is the same as the one with the frequency response function of **Figure 17.14**. Thus, the following contents show the metamaterial panels with the other three aluminium bar inclusions and a pure aluminium case.

The following four figures from **Figure 17.18** to **Figure 17.21** correspond to the metamaterial panels with the aluminium bar's diameters of 4mm, 12mm, 16mm, and a pure aluminium panel, respectively. Each figure contains three subplots. We can see that subplot (a) is the fluid pressure (FP) with the unit of Pa. Then, subplot (b) is the sound pressure level (SPL) with the dB unit. Lastly, subplot (c) is the sound transmission loss (STL) with the dB unit. In these subplots, "P400" means the testing panel has 400 cells; "AL4mm" means the aluminium bar's diameter is 4mm. Then, "AL12mm" means the aluminium bar's diameter is 12mm; "AL16mm" means the aluminium bar's diameter is 16mm. Finally, "Aluminium" means the panel consists of pure aluminium.

Based on the results in **Figure 17.18**, we can see that the resonant and anti-resonant points are mainly distributed in the range of $160-220\,\mathrm{Hz}$. The sound transmission loss under the frequency value of 100 Hz is mainly within 10 dB.

Based on the results in **Figure 17.19**, we can see that the resonant and anti-resonant points are mainly distributed in the range of $40-100\,\mathrm{Hz}$. The sound transmission loss under the frequency value of 40 Hz is mainly within 10 dB. And the sound transmission loss over the frequency value of 100 Hz changes from around 10 dB to 30 dB.

Based on the results in **Figure 17.20**, we can see that the resonant and anti-resonant points are mainly distributed in the range of $35-70\,\mathrm{Hz}$. The sound transmission loss under the frequency value of 35 Hz is mainly within 10 dB. And the sound transmission loss over the frequency value of 100 Hz changes from around 20 dB to 35 dB.

Based on the results in **Figure 17.21**, we can see no resonant point in the given frequency range. The sound transmission loss in the given frequency range is between 60 dB and 80 dB, close to 73 dB.

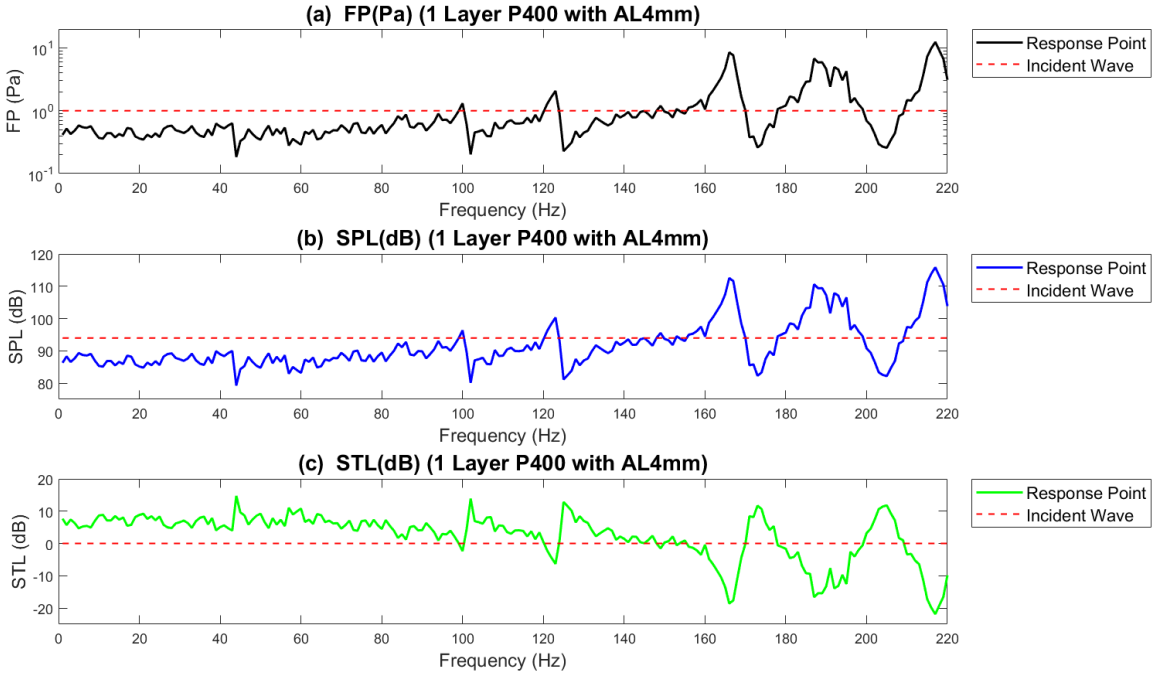


Figure 17.18: FRF of the fluid response point (l Layer P400 AL4mm).

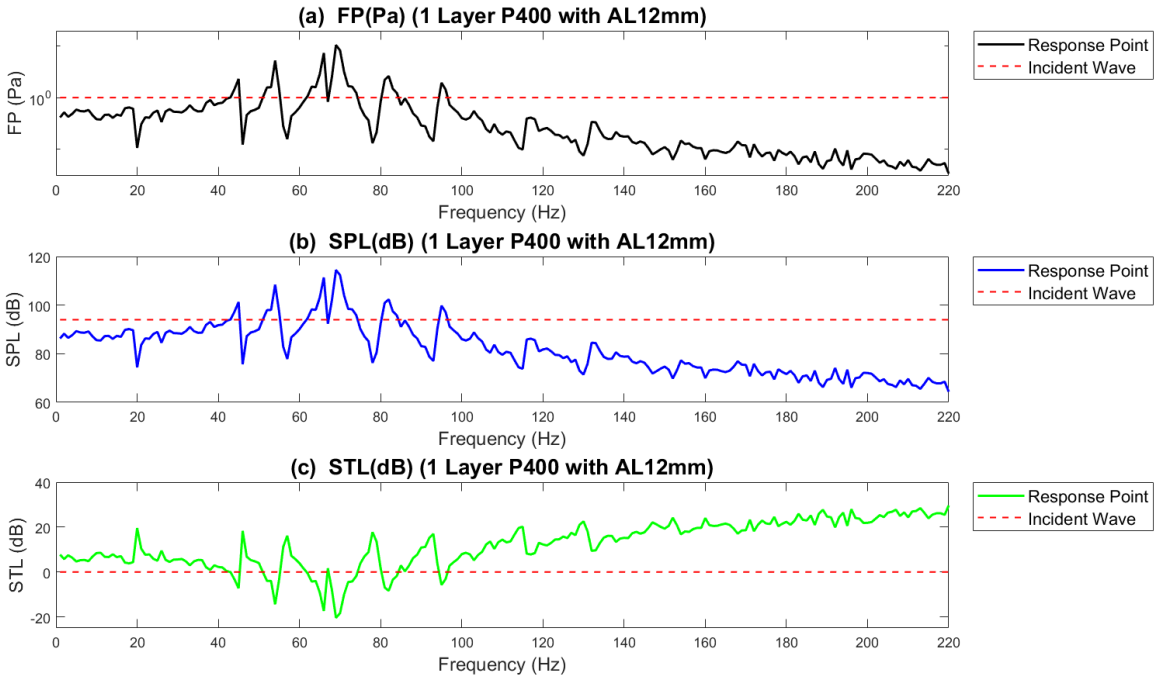


Figure 17.19: FRF of the fluid response point (l Layer P400 AL12mm).

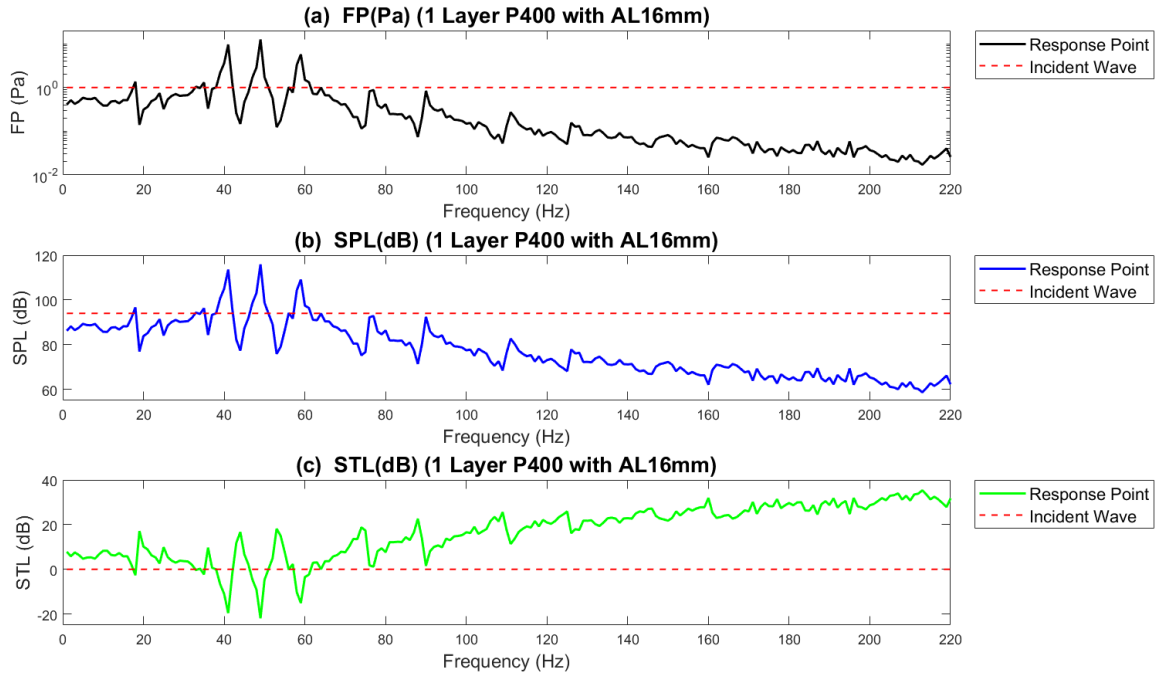


Figure 17.20: FRF of the fluid response point (l Layer P400 AL16mm).

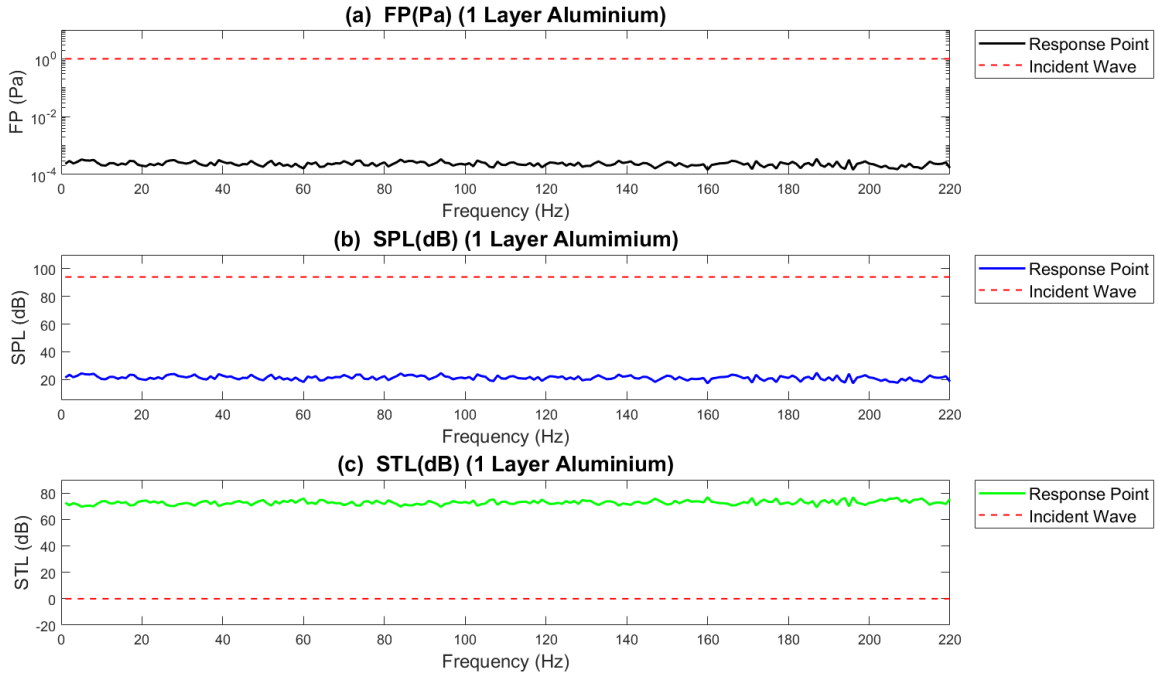


Figure 17.21: FRF of the fluid response point (l Layer Aluminium).

17.3.4 Number-of-Layers-Controlled Metamaterial Panels

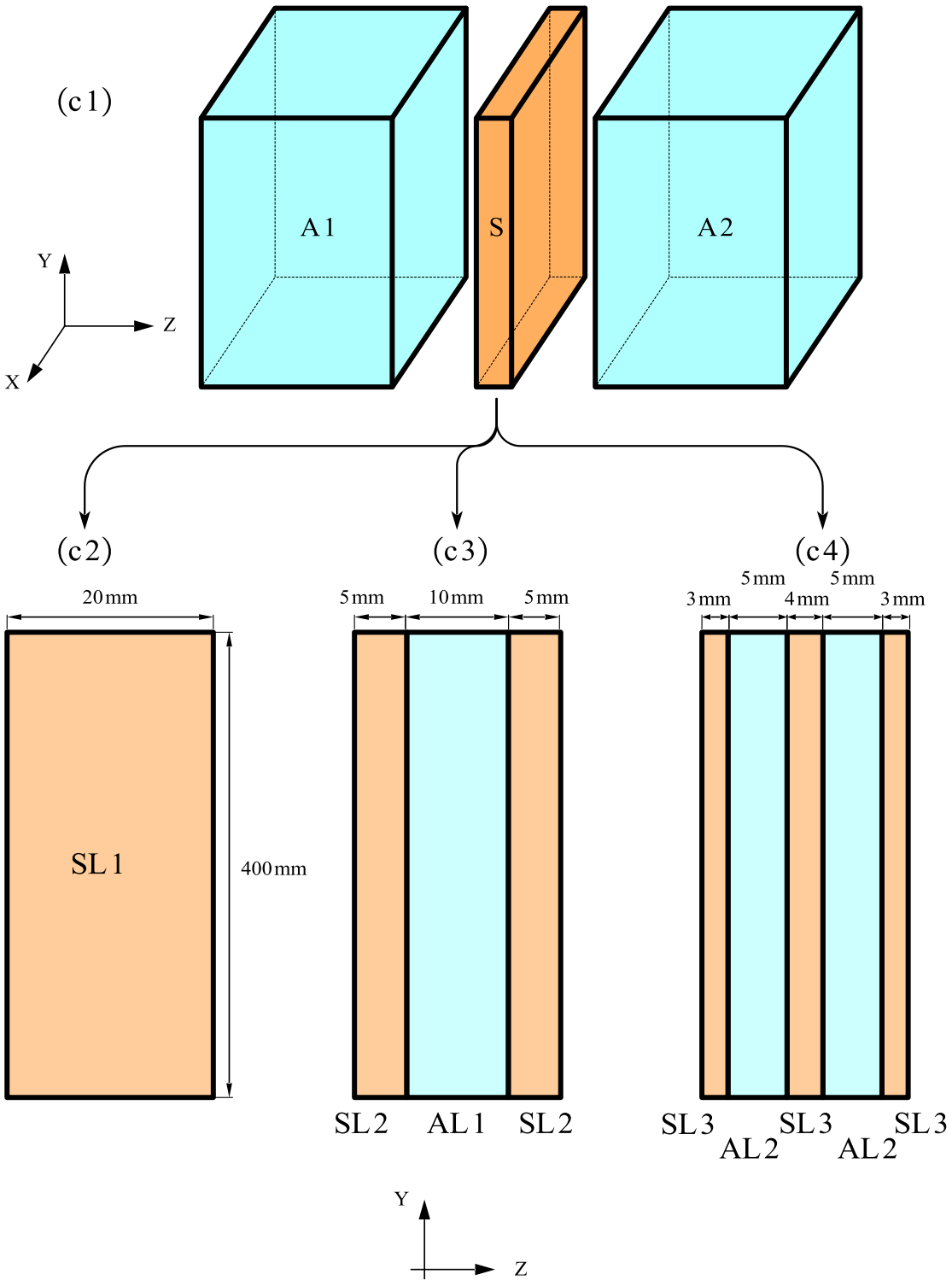


Figure 17.22: Single and multi-layer structures (meta panels).

In this control group, the variable is the number of panel layers. The panel size, the total sample thickness (including the panel layers and intermediate air layers), and the aluminium bar diameter are fixed. The unchanged panel size is $400\times400\times20\,\mathrm{mm}$ regarding to XYZ coordinates. And the unchanged total sample thickness is 20mm. Then, the unchanged aluminium bar diameter is 8mm. As for the number of panel layers, there are three cases.

In **Figure 17.22**, subplot (c1) displays the layout of the structural-acoustic system in a 3D view. "A1" is the outside air domain, "A2" is the inside air domain, and "S" is the testing panel. Below subplot (c1), subplots (c2), (c3), and (c4) are enlarged views of three varieties of the testing samples. From left to right, subplot (c2) indicates the first variety, that the testing panel has only one layer consisting of the metamaterial. "SL1" means one-layer solid material. Then, subplot (c3) indicates the second variety, that the testing panel consists of two layers of metamaterial and one layer of acoustic medium. "SL2" means two-layer solid material, while "AL1" means one-layer air. Finally, subplot (c4) indicates the third variety, that the testing panel consist of three layers of metamaterial and two layers of acoustic medium. "SL3" means three-layer solid material, while "AL2" means two-layer air.

In subplot (c2), we can see that the thickness of the whole solid material is 20mm. Then, in subplot (c3), the thicknesses of the left and right solid layers are the same, equal to 5mm. And the thickness of the middle acoustic medium layer is 10mm. Finally, in subplot (c4), the thicknesses of the left and right solid layers are the same, equal to 3mm; and the thickness of the middle solid layer equals 4mm. After that, the two middle acoustic layers have the same thickness, which equals 5mm.

The lateral-boundary-constrained condition of the panel layers is the first condition introduced in **Figure 17.5** and subplots (a1) and (a2) of **Figure 17.9**. We notice that the one-layer panel contains aluminium bars with a diameter of 8mm is the same as the one with the frequency response function of **Figure 17.10**. Thus, the following contents only show the results of the two-layer and three-layer samples.

Figure 17.23 and **Figure 17.24** correspond to the two-layer and three-layer samples, respectively. Each figure contains three subplots. We can see that subplot (a) is the fluid pressure (FP) with the unit of Pa. Then, subplot (b) is the sound pressure level (SPL) with the dB unit. Lastly, subplot (c) is the sound transmission loss (STL) with the dB unit. In these subplots, "P400" means the testing panel has 400 cells; "AL8mm" means the aluminium bar's diameter is 8mm.

Based on the results of the first constraint condition in **Figure 17.23**, we can see that the resonant and anti-resonant points are mainly distributed in the range of $120-160\,\mathrm{Hz}$. The sound transmission loss under the frequency value of 100Hz is mainly within 10dB.

Based on the results of the first constraint condition in **Figure 17.24**, we can see that the resonant and anti-resonant points are mainly distributed in the range of $120-160\,\mathrm{Hz}$. The sound transmission loss under the frequency value of 100Hz is mainly within 10dB.

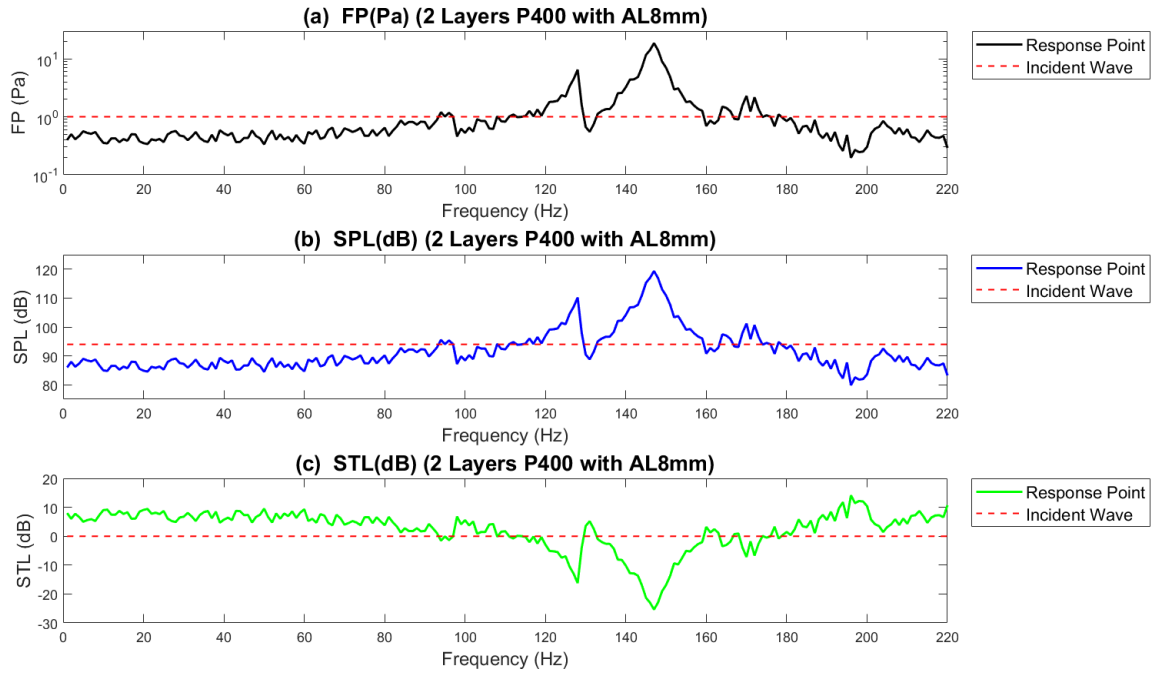


Figure 17.23: FRF of the fluid response point (2 Layers P400 AL8mm).

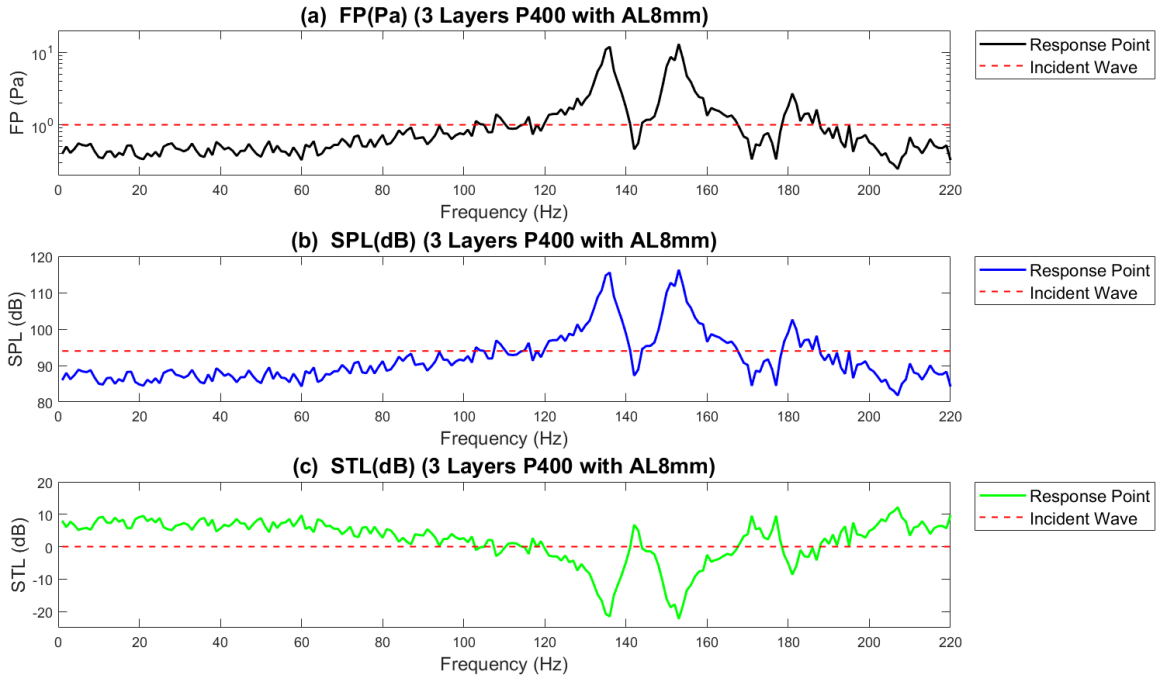


Figure 17.24: FRF of the fluid response point (3 Layers P400 AL8mm).

17.4 Chapter Conclusions

Let us have a brief review of chapter **17**. Section **17.1** theoretically discusses the available simulation strategies for vibro-acoustic analyses of metamaterial samples. Section **17.2** utilizes the preliminary tests to ensure the best simulation strategy. Consequently, the best simulation strategy for the preliminary metamaterial designs is used in the tests. Subsequently, section **17.3** elaborates on modal-space-based steady-state vibro-acoustic analyses of fourteen fluid-structure systems. These fluid-structure systems are classified into three control groups and one compensation group based on three control parameters. The specific samples and the corresponding simulations are introduced in subsections **17.3.1**, **17.3.2**, **17.3.3**, and **17.3.4**. Finally, based on the results and phenomena illustrated in section **17.3**, this section will further extend analyses and assessments of metamaterial designs.

17.4.1 Simulation Efficiencies

Table 17.2, **Table 17.3**, and **Table 17.4** record the number of mesh grids, the number of total extracted eigenmodes of the fluid-structure systems, and the time consumption corresponding to vibro-acoustic analyses, respectively. The abbreviations in the first column of these tables represent corresponding fluid-structure systems with specific settings. And cross-references in brackets are the figure names of corresponding frequency response functions.

Based on the data in **Table 17.2** and **Table 17.3**, we perceive that the number of the fluid mesh grids and the number of the fluid eigenmodes of the fourteen fluid-structure systems do not have big discrepancies. In comparison, the number of the structure mesh grids depends on the degrees of panels' aluminium bar inclusions and the number of panel layers. Then, the number of the structure eigenmodes depends on the degrees of aluminium bar inclusions, the constraint conditions, and the number of panel layers.

Based on the data in **Table 17.4**, we perceive that the number of extracted eigenmodes is the main factor that positively influences the time consumption of the corresponding eigenvalue extraction analyses. That is because the "ACMS Method" from Nastran can efficiently treat the massive physical DOFs before eigenvalue extraction analyses; thus, the time consumption of eigenvalue extraction analyses mainly depends on the number of extracted eigenmodes instead of the number of mesh grids. In comparison, the time consumption of the modal-space-based steady-state dynamic analyses depends on the number of eigenmodes and the number of mesh grids but not simply in positive relations.

Table 17.2: The number of mesh grids (fourteen fluid-structure systems).

Fluid-Structure Systems	Fluid	Structure
1 Layer P400 AL8mm (**Figure 17.10**)	70602	56405
1 Layer P400.CONS2 AL8mm (**Figure 17.11**)	70602	56405
1 Layer P400.CONS3 AL8mm (**Figure 17.12**)	70602	56405
1 Layer P400.CONS4 AL8mm (**Figure 17.13**)	70602	56405
1 Layer Melamine (**Figure 17.14**)	70602	56405
1 Layer Melamine CONS2 (**Figure 17.15**)	70602	56405
1 Layer Melamine CONS3 (**Figure 17.16**)	70602	56405
1 Layer Melamine CONS4 (**Figure 17.17**)	70602	56405
1 Layer P400 AL4mm (**Figure 17.18**)	70602	72405
1 Layer P400 AL12mm (**Figure 17.19**)	70602	40405
1 Layer P400 AL16mm (**Figure 17.20**)	70602	40405
1 Layer Aluminium (**Figure 17.21**)	70602	56405
2 Layers P400 AL8mm (**Figure 17.23**)	75645	45124
3 Layers P400 AL8mm (**Figure 17.24**)	77326	67686

Table 17.3: The number of the total extracted eigenmodes (fourteen fluid-structure systems).

Fluid-Structure Systems	Fluid	Structure
1 Layer P400 AL8mm (**Figure 17.10**)	26	208
1 Layer P400.CONS2 AL8mm (**Figure 17.11**)	26	178
1 Layer P400.CONS3 AL8mm (**Figure 17.12**)	26	126
1 Layer P400.CONS4 AL8mm (**Figure 17.13**)	26	404
1 Layer Melamine (**Figure 17.14**)	26	75
1 Layer Melamine CONS2 (**Figure 17.15**)	26	52
1 Layer Melamine CONS3 (**Figure 17.16**)	26	16
1 Layer Melamine CONS4 (**Figure 17.17**)	26	400
1 Layer P400 AL4mm (**Figure 17.18**)	26	70
1 Layer P400 AL12mm (**Figure 17.19**)	26	324
1 Layer P400 AL16mm (**Figure 17.20**)	26	358
1 Layer Aluminium (**Figure 17.21**)	26	8
2 Layers P400 AL8mm (**Figure 17.23**)	39	833
3 Layers P400 AL8mm (**Figure 17.24**)	52	1417

Table 17.4: The time consumption (second) of modal-space-based vibro-acoustic analyses (fourteen fluid-structure systems, the number of excitation frequencies is 220).

Fluid-Structure Systems	Eigenvalue Extraction	Dynamic Analysis	Full Process
1 Layer P400 AL8mm (**Figure 17.10**)	97.4s	921.0s	1018.4s
1 Layer P400.CONS2 AL8mm (**Figure 17.11**)	80.3	892.0s	972.3s
1 Layer P400.CONS3 AL8mm (**Figure 17.12**)	74.8s	834.0s	908.8s
1 Layer P400.CONS4 AL8mm (**Figure 17.13**)	142.1s	1048.0s	1190.1s
1 Layer Melamine (**Figure 17.14**)	50.0s	923.0	973.0s
1 Layer Melamine CONS2 (**Figure 17.15**)	47.6s	916.0s	963.6s
1 Layer Melamine CONS3 (**Figure 17.16**)	28.7s	890.0s	918.7s
1 Layer Melamine CONS4 (**Figure 17.17**)	207.8s	958.0s	1165.8s
1 Layer P400 AL4mm (**Figure 17.18**)	59.4s	1127.0s	1186.4s
1 Layer P400 AL12mm (**Figure 17.19**)	105.5s	663.0s	768.5s
1 Layer P400 AL16mm (**Figure 17.20**)	133.4s	706.0s	819.4s
1 Layer Aluminium (**Figure 17.21**)	31.0s	945.0s	976.0s
2 Layers P400 AL8mm (**Figure 17.23**)	289.4s	887.0s	1176.4s
3 Layers P400 AL8mm (**Figure 17.24**)	777.1s	1146.0s	1923.1s

17.4.2 Assessments of Modal-Space-Based Steady-State Vibro-Acoustic Analyses (First Step, Eigenvalue Extraction Analyses)

The eigenvalue extraction analysis is a prior step of the modal-space-based steady-state vibro-acoustic analysis. In addition, the number of eigenfrequencies and their distribution is a forecasting indicator of the system's dynamic performance. Thus, this subsection will discuss the eigenfrequencies and their distribution of the fourteen fluid-structure systems.

In these fourteen fluid-structure systems, the outside air (A1) and inside air (A2) domains are two indispensable subparts, and the other subparts depend on the components of the intermediate panel (S) (see **Figure 17.22**). In the first case, the intermediate panel is a one-layer (SL1) metamaterial. And the first case includes seven fluid-structure systems: the "1 Layer P400 AL4mm", the "1 Layer P400 AL8mm", the "1 Layer P400 AL12mm", the "1 Layer P400 AL16mm", the "1 Layer P400.CONS2 AL8mm", the "1 Layer P400.CONS3 AL8mm", and the "1 Layer P400.CONS4 AL8mm". In the second case, the intermediate panel is a one-layer (SL1) melamine foam. And the second case includes four fluid-structure systems: the "1 Layer Melamine", the "1 Layer Melamine CONS2", the "1 Layer Melamine CONS3", and the "1 Layer Melamine CONS4". In the third case, the intermediate panel is a one-layer (SL1) aluminium alloy. And the third case includes one fluid-structure system: the "1 Layer

Aluminium". In the fourth case, the media consists of a two-layer (SL2) metamaterial and a one-layer (AL1) air. And the fourth case includes one fluid-structure system: the "2 Layers P400 AL8mm". In the fifth case, the media consists of a three-layer (SL3) metamaterial and a two-layer (AL2) air. And the fifth case includes one fluid-structure system: the "3 Layers P400 AL8mm".

All in all, there are nine subparts in the fourteen fluid-structure systems: the outside air (A1) domain, the inside air (A2) domain, the one-layer (SL1) metamaterial, the one-layer (SL1) melamine foam, the one-layer (SL1) aluminium, the two-layer (SL2) metamaterial, the one-layer (AL1) air, the three-layer (SL3) metamaterial, and the two-layer (AL2) air. Theoretically, the eigenfrequencies of the subparts are distributed in different frequency ranges. Thus, these subparts must be set with different extraction frequency ranges in eigenvalue extraction analyses. For the outside and inside air domains, the one-layer air, and the two-layer air, the extraction frequency range is 0–1500Hz. For the one-layer, two-layer, and three-layer metamaterial, except for the "1 Layer P400.CONS4 AL8mm", the extraction frequency range is 0–250Hz. For the one-layer melamine foam, except for the "1 Layer Melamine CONS4", the extraction frequency range is 0–750Hz. Then, for the one-layer aluminium alloy, the extraction frequency range is 0–5000Hz. Consequently, the eigenvalue extraction analyses find the corresponding eigenfrequencies with diversified distributions of the subparts. After that, from **Table 17.5** to **Table 17.9**, they specifically illustrate the eigenfrequencies distribution in six assigned frequency ranges. The six assigned frequency ranges of **Table 17.5** to **Table 17.9** are 0–250Hz, 250–500Hz, 500–750Hz, 750–1500Hz, 1500–3000Hz, and 3000–5000Hz, respectively.

Table 17.5: The number of extracted eigenmodes (fourteen fluid-structure systems, the frequency range is 0–250Hz).

Fluid-Structure Systems	Fluid	Structure
1 Layer P400 AL8mm (**Figure 17.10**)	1	208
1 Layer P400.CONS2 AL8mm (**Figure 17.11**)	1	178
1 Layer P400.CONS3 AL8mm (**Figure 17.12**)	1	126
1 Layer P400.CONS4 AL8mm (**Figure 17.13**)	1	0
1 Layer Melamine (**Figure 17.14**)	1	11
1 Layer Melamine CONS2 (**Figure 17.15**)	1	4
1 Layer Melamine CONS3 (**Figure 17.16**)	1	0
1 Layer Melamine CONS4 (**Figure 17.17**)	1	0
1 Layer P400 AL4mm (**Figure 17.18**)	1	70
1 Layer P400 AL12mm (**Figure 17.19**)	1	324
1 Layer P400 AL16mm (**Figure 17.20**)	1	358
1 Layer Aluminium (**Figure 17.21**)	1	0
2 Layers P400 AL8mm (**Figure 17.23**)	2	833
3 Layers P400 AL8mm (**Figure 17.24**)	4	1417

Table 17.6: The number of extracted eigenmodes (fourteen fluid-structure systems, the frequency range is 250–500Hz).

Fluid-Structure Systems	Fluid	Structure
1 Layer P400 AL8mm (**Figure 17.10**)	4	0
1 Layer P400.CONS2 AL8mm (**Figure 17.11**)	4	0
1 Layer P400.CONS3 AL8mm (**Figure 17.12**)	4	0
1 Layer P400.CONS4 AL8mm (**Figure 17.13**)	4	404 (all 387Hz)
1 Layer Melamine (**Figure 17.14**)	4	26
1 Layer Melamine CONS2 (**Figure 17.15**)	4	16
1 Layer Melamine CONS3 (**Figure 17.16**)	4	16 (all 499Hz)
1 Layer Melamine CONS4 (**Figure 17.17**)	4	0
1 Layer P400 AL4mm (**Figure 17.18**)	4	0
1 Layer P400 AL12mm (**Figure 17.19**)	4	0
1 Layer P400 AL16mm (**Figure 17.20**)	4	0
1 Layer Aluminium (**Figure 17.21**)	4	0
2 Layers P400 AL8mm (**Figure 17.23**)	6	0
3 Layers P400 AL8mm (**Figure 17.24**)	8	0

Table 17.7: The number of extracted eigenmodes (fourteen fluid-structure systems, the frequency range is 500–750Hz).

Fluid-Structure Systems	Fluid	Structure
1 Layer P400 AL8mm (**Figure 17.10**)	2	0
1 Layer P400.CONS2 AL8mm (**Figure 17.11**)	2	0
1 Layer P400.CONS3 AL8mm (**Figure 17.12**)	2	0
1 Layer P400.CONS4 AL8mm (**Figure 17.13**)	2	0
1 Layer Melamine (**Figure 17.14**)	2	38
1 Layer Melamine CONS2 (**Figure 17.15**)	2	32
1 Layer Melamine CONS3 (**Figure 17.16**)	2	0
1 Layer Melamine CONS4 (**Figure 17.17**)	2	0
1 Layer P400 AL4mm (**Figure 17.18**)	2	0
1 Layer P400 AL12mm (**Figure 17.19**)	2	0
1 Layer P400 AL16mm (**Figure 17.20**)	2	0
1 Layer Aluminium (**Figure 17.21**)	2	0
2 Layers P400 AL8mm (**Figure 17.23**)	3	0
3 Layers P400 AL8mm (**Figure 17.24**)	4	0

Table 17.8: The number of extracted eigenmodes (fourteen fluid-structure systems, the frequency range is 750–1500Hz).

Fluid-Structure Systems	Fluid	Structure
1 Layer P400 AL8mm (**Figure 17.10**)	18	0
1 Layer P400.CONS2 AL8mm (**Figure 17.11**)	18	0
1 Layer P400.CONS3 AL8mm (**Figure 17.12**)	18	0
1 Layer P400.CONS4 AL8mm (**Figure 17.13**)	18	0
1 Layer Melamine (**Figure 17.14**)	18	0
1 Layer Melamine CONS2 (**Figure 17.15**)	18	0
1 Layer Melamine CONS3 (**Figure 17.16**)	18	0
1 Layer Melamine CONS4 (**Figure 17.17**)	18	0
1 Layer P400 AL4mm (**Figure 17.18**)	18	0
1 Layer P400 AL12mm (**Figure 17.19**)	18	0
1 Layer P400 AL16mm (**Figure 17.20**)	18	0
1 Layer Aluminium (**Figure 17.21**)	18	1
2 Layers P400 AL8mm (**Figure 17.23**)	27	0
3 Layers P400 AL8mm (**Figure 17.24**)	36	0

Table 17.9: The number of extracted eigenmodes (fourteen fluid-structure systems, the frequency range is 1500–3000Hz).

Fluid-Structure Systems	Fluid	Structure
1 Layer P400 AL8mm (**Figure 17.10**)	0	0
1 Layer P400.CONS2 AL8mm (**Figure 17.11**)	0	0
1 Layer P400.CONS3 AL8mm (**Figure 17.12**)	0	0
1 Layer P400.CONS4 AL8mm (**Figure 17.13**)	0	0
1 Layer Melamine (**Figure 17.14**)	0	0
1 Layer Melamine CONS2 (**Figure 17.15**)	0	0
1 Layer Melamine CONS3 (**Figure 17.16**)	0	0
1 Layer Melamine CONS4 (**Figure 17.17**)	0	0
1 Layer P400 AL4mm (**Figure 17.18**)	0	0
1 Layer P400 AL12mm (**Figure 17.19**)	0	0
1 Layer P400 AL16mm (**Figure 17.20**)	0	0
1 Layer Aluminium (**Figure 17.21**)	0	2
2 Layers P400 AL8mm (**Figure 17.23**)	0	0
3 Layers P400 AL8mm (**Figure 17.24**)	0	0

Table 17.10: The number of extracted eigenmodes (fourteen fluid-structure systems, the frequency range is 3000–5000Hz).

Fluid-Structure Systems	Fluid	Structure
1 Layer P400 AL8mm (**Figure 17.10**)	0	0
1 Layer P400.CONS2 AL8mm (**Figure 17.11**)	0	0
1 Layer P400.CONS3 AL8mm (**Figure 17.12**)	0	0
1 Layer P400.CONS4 AL8mm (**Figure 17.13**)	0	0
1 Layer Melamine (**Figure 17.14**)	0	0
1 Layer Melamine CONS2 (**Figure 17.15**)	0	0
1 Layer Melamine CONS3 (**Figure 17.16**)	0	0
1 Layer Melamine CONS4 (**Figure 17.17**)	0	400 (all 3263Hz)
1 Layer P400 AL4mm (**Figure 17.18**)	0	0
1 Layer P400 AL12mm (**Figure 17.19**)	0	0
1 Layer P400 AL16mm (**Figure 17.20**)	0	0
1 Layer Aluminium (**Figure 17.21**)	0	5
2 Layers P400 AL8mm (**Figure 17.23**)	0	0
3 Layers P400 AL8mm (**Figure 17.24**)	0	0

Fluid Subparts of Fluid-Structure Systems

Let us discuss the eigenfrequencies of the outside and inside air domains, the one-layer air, and the two-layer air at the beginning. The number of extracted eigenfrequencies of the outside air domain, the inside air domain, and the one-layer air is 13. Then, the number of extracted eigenfrequencies of the two-layer air is 26. The fluid subparts are the outside and inside air domains for twelve one-layer-panel systems. For the two-layer-panel system, the fluid subparts include the outside air domain, the inside air domain, and the one-layer intermediate air. Then, for the three-layer-panel system, the fluid subparts include the outside air domain, the inside air domain, and the two-layer intermediate air. Therefore, in **Table 17.3**, we can see that the number of fluid eigenvalues of the twelve one-layer-panel systems is 26, the two-layer-panel system is 39, and the three-layer-panel system is 52. From **Table 17.5** to **Table 17.8**, contents show that the fluid subparts' eigenfrequencies are concentrated in the range of 750–1500Hz. At the same time, with few eigenfrequencies in the frequency range of 0–250Hz, 250–500Hz, and 500–750Hz. Since the excitation frequency of the subsequent vibro-acoustic analyses is 0–220Hz, there will be no resonances and anti-resonances of the fluid subparts.

First Control Group of Metamaterial Panels of Fluid-Structure Systems (The Variable Is Constrained Lateral Boundaries, subsection 17.3.1)

Let us discuss the eigenfrequencies of the one-layer metamaterial panels. And the panels have different lateral-boundary-constrained conditions. In the first control group, the fluid-structure systems include the "1 Layer P400 AL8mm", the "1 Layer P400.CONS2 AL8mm", the "1

Layer P400.CONS3 AL8mm", and the "1 Layer P400.CONS4 AL8mm". For the eigenvalue extraction analyses, the extraction frequency range of the panels in the first three systems is 0–250Hz, and the fourth is 0–450Hz. Then, in **Table 17.5** and **Table 17.6**, we can see the panels' total number of extracted eigenfrequencies.

Firstly, for the panel in the "1 Layer P400 AL8mm", the total number of extracted eigenfrequencies is 208. Secondly, for the panel in the "1 Layer P400.CONS2 AL8mm", the total number of extracted eigenfrequencies is 178. Thirdly, for the panel in the "1 Layer P400.CONS3 AL8mm", the total number of extracted eigenfrequencies is 126. Fourthly, for the panel in the "1 Layer P400.CONS4 AL8mm", the total number of extracted eigenfrequencies is 404, and all eigenfrequencies are around 382Hz. Consequently, no resonances and anti-resonances of the "1 Layer P400.CONS4 AL8mm" can be found in the frequency range of 0–220Hz (see **Figure 17.13**).

We notice that the fourth case of the first control group is extraordinary. Theoretically, "1 Layer P400.CONS4 AL8mm" should have eigenmodes less than 126 instead of 404 because the lateral boundaries of four hundred subparts of this panel are constrained (see **Figure 17.8** and **Figure 17.9**). However, at the start, Nastran didn't find any eigenmodes of the "1 Layer P400.CONS4 AL8mm" in the frequency range of 0–250Hz. Then, we set the frequency range of 0–450Hz to avoid computation crashes. Finally, Nastran got 404 results with the same value of around 382Hz. Because every unit cell in the panel is constrained, Nastran deems the constrained cells as independent structures.

Compensation Group of Melamine Panels of Fluid-Structure Systems (The Variable is Constrained Lateral Boundaries, subsection 17.3.2)

Let us discuss the eigenfrequencies of the one-layer melamine foam with different lateral-boundary-constrained conditions. In the compensation group, the fluid-structure systems include the "1 Layer Melamine", the "1 Layer Melamine CONS2", the "1 Layer Melamine CONS3", and the "1 Layer Melamine CONS4". For the eigenvalue extraction analyses, the extraction frequency range of the panels in the first three systems is 0–750Hz, and the fourth is 0–3300Hz. Then, in **Table 17.5**, **Table 17.6**, **Table 17.7**, and **Table 17.10**, we can see the number of extracted eigenfrequencies of the panels distributed in the four frequency ranges.

Firstly, 75 eigenfrequencies of the panel in the system "1 Layer Melamine" are extracted. Specifically, in the frequency ranges around 0–250Hz, 250–500Hz, and 500–750Hz, the number of eigenfrequencies is 11, 26, and 38, respectively. Secondly, 50 eigenfrequencies of the panel in the system "1 Layer Melamine CONS2" are extracted. Specifically, in the frequency ranges around 0–250Hz, 250–500Hz, and 500–750Hz, the number of eigenfrequencies is 4, 16, and 32, respectively. Thirdly, for the panel in the system "1 Layer Melamine CONS3", 16 eigenfrequencies are extracted, and their values are around 499Hz. Fourthly, for the panel in the system "1 Layer Melamine CONS4", 400 eigenfrequencies are extracted, and their values are around 3263Hz. Since the eigenfrequencies of the panels are mainly distributed outside the range of 0–220Hz, no resonances and anti-resonances of the corresponding fluid-structure systems are found in the excitation frequency range of the vibro-acoustic analyses (see **Figure 17.14** to **Figure 17.17**).

We notice that the third and fourth cases of the compensation group are extraordinary. The reason is analogous to the one in the first control group. Because the lateral boundaries constraint of the panel in the "1 Layer Melamine CONS3" generates 16 independent subparts

(see Figure 17.7), and the lateral boundaries constraint of the panel in the "1 Layer Melamine CONS4" generates 400 independent subparts (see Figure 17.8). Then, Nastran finds only one eigenfrequency in the assigned extraction frequency range of these independent subparts.

Second Control Group of Metamaterial Panels of Fluid-Structure Systems (The Variable is Aluminium Inclusions, subsection 17.3.3)

Let us discuss the eigenfrequencies of the one-layer metamaterial with different aluminium bar inclusions. In the second control group, the fluid-structure systems include the "1 Layer P400 AL4mm", the "1 Layer P400 AL8mm", the "1 Layer P400 AL12mm", the "1 Layer P400 AL16mm", and "1 Layer Aluminium". For the eigenvalue extraction analyses, the extraction frequency range of the panels in the first four systems is 0–250 Hz, and the fifth is 0–5000 Hz. Then, in Table 17.5, Table 17.8, Table 17.9, and Table 17.10, we can see the number of extracted eigenfrequencies of the panels distributed in the four frequency ranges.

Firstly, for the panel in the "1 Layer P400 AL4mm", the total number of extracted eigenfrequencies is 70, and the frequency range is around 0–250Hz. Secondly, for the panel in the "1 Layer P400 AL8mm", the total number of extracted eigenfrequencies is 208, and the frequency range is around 0–250 Hz. Thirdly, for the panel in the "1 Layer P400 AL12mm", the total number of extracted eigenfrequencies is 324, and the frequency range is around 0–250 Hz. Fourthly, for the panel in the "1 Layer P400 AL16mm", the total number of extracted eigenfrequencies is 358, and all eigenfrequencies are around 0–250 Hz. Fifthly, the total number of extracted eigenfrequencies for the panel in the "1 Layer Aluminium" is 8. Specifically, in the frequency ranges around 750–1500 Hz, 1500–3000 Hz, and 3000–5000 Hz, the number of eigenfrequencies is 1, 2, and 5, respectively. Consequently, no resonances and anti-resonances of the "1 Layer Aluminium" can be found in the frequency range of 0–220 Hz (see Figure 17.21).

Third Control Group of Metamaterial Panels of Fluid-Structure Systems (The Variable Is the Number of Panel Layers, subsection 17.3.4)

Let us discuss the eigenfrequencies of the multi-layer metamaterial panels. In the third control group, the fluid-structure systems include the "1 Layer P400 AL8mm", the "2 Layers P400 AL8mm", and "3 Layers P400 AL8mm". For the eigenvalue extraction analyses, the extraction frequency range of the panels in the three systems is 0–250Hz. From Table 17.5, we can see the number of the extracted eigenfrequencies of the panels in these three systems is 208, 833, and 1417, respectively.

In Figure 17.22, we can see the three panel's structures. One layer means only one panel in the fluid-structure system; two layers mean two independent panels, and three layers mean three independent panels. Thus, 208 is the number of one panel's eigenfrequencies. Then, two independent panels added together have 833 eigenfrequencies. And three independent panels added together have 1417 eigenfrequencies. It is worth noting that many repeated eigenfrequencies exist in the two-layer and three-layer panels. Indeed, the exact number of the different eigenfrequencies of the two-layer and three-layer panels is hard to ensure because the eigenvalue extraction analyses have many numerical uncertainties. Approximately, the number of the different eigenvalues of the two-layer panel is around 415, half of 833. And the number of the different eigenvalues of the three-layer panel is around 417, one-third of 1417.

17.4.3 Assessments of Modal-Space-Based Steady-State Vibro-Acoustic Analyses (Second Step, Vibro-Acoustic Analyses)

The vibro-acoustic analysis is an important method to verify the preliminary metamaterial designs. People can directly perceive the dynamic performances of the metamaterial samples through the frequency response functions.

This paper uses three control parameters to guide the metamaterial designs: the panel's constrained lateral boundaries, the panel's aluminium inclusions, and the number of panel layers. And these three control parameters classify the metamaterial samples into three control groups. In addition, the melamine foam panels with different constraint conditions are classified into a compensation group. Tests in this compensation group can verify some residual problems.

Section **17.3** introduces the tests and illustrates the results of the fourteen fluid-structure systems. These fluid-structure systems are classified into three control groups and one compensation group with three different control parameters. And the simulations applied in tests are the modal-space-based steady-state vibro-acoustic analyses. Thus, the eigenvalue extraction analysis is a prior step before the vibro-acoustic analysis. In subsection **17.4.2**, the assessments of the prior eigenvalue extraction analyses give us a forecasting knowledge of the mechanical properties of the corresponding testing samples. Based on the eigenfrequencies distribution, we can directly judge whether the testing samples are sensible at the low-frequency range.

To some extent, the eigenfrequencies distribution corresponds with the testing panels' resonances and anti-resonances. For more information about the structural frequency response functions of the metamaterial panels, please review the sections in chapter **15**. It is worth noting that the response point in the dynamic analyses of chapter **15** is a structural point on the testing panel (see **Figure 15.2**), and the response point in vibro-acoustic analyses of this chapter is a fluid point in the inside air domain (see **Figure 17.2**). Subsection **17.4.3** is an extension of section **17.3** and corresponds to subsection **17.4.2**.

As the first control group of the metamaterial panels, the numerical verifications in subsection **17.3.1** show that the panel's lateral-boundary-constrained conditions highly influence the resonance and anti-resonance phenomena of the fluid point's frequency response functions. Then, we can say that the panel's lateral-boundary-constrained conditions can adjust the dynamic sensibility of metamaterial panels. In addition, the more lateral boundaries are constrained, the less resonance and anti-resonance emerge. And this phenomenon leads to more acoustic wave attenuations (see **Figure 17.10** to **Figure 17.13**). In the extreme case, all unit cells' lateral boundaries are constrained (see **Figure 17.8**). Then, no resonance and anti-resonance emerge (see **Figure 17.13**) in the low-frequency range. The reason is that the extreme case of lateral-boundary constraint makes the panel lose much flexibility. And no eigenfrequencies are found in the frequency range of 0-220Hz. Thus, the metamaterial panel's soundproof capability with the extreme lateral-boundary constraint case is considerably increased.

As the compensation of the first and second control groups, the numerical verifications in subsection **17.3.2** show that the melamine panel's lateral-boundary-constrained conditions highly influence the dynamic reactions of the fluid response point. Since the resonance and

anti-resonance phenomena do not exist in the low-frequency range, the melamine panel's lateral-boundary-constrained conditions do not influence the resonance and anti-resonance effects of the fluid response point. Then, the more lateral boundaries are constrained, the more acoustic wave is attenuated (see **Figure 17.14** to **Figure 17.17**). In the extreme case, lateral boundaries of the four hundred subparts of the melamine panel are constrained (see **Figure 17.8**). Then, the soundproof capability of the melamine panel reaches a very high level in the low-frequency range (see **Figure 17.17**). The reason is that the extreme case of lateral-boundary constraint makes the panel lose much flexibility. And no eigenfrequencies are found in the frequency range of 0–220Hz. Thus, the soundproof capability of the melamine panel with extreme constraints is considerably increased.

As the second control group, the numerical verifications in subsection **17.3.3** show that the panel's degree of aluminium inclusions influences the panel's acoustic performance. And the panel's lateral-boundary-constrained condition of the second control group is the first condition (see **Figure 17.5**). Then, based on the results in **Figure 17.10**, **Figure 17.18**, **Figure 17.19**, and **Figure 17.20**, the aluminium inclusions influence the resonance and anti-resonance distributions of the fluid response point. And the aluminium inclusions also influence the soundproof capability of metamaterial panels. In the extreme case, the panel consists of pure aluminium alloy. Then, no resonance and anti-resonance emerge in the low-frequency range (see **Figure 17.21**). The reason is that the aluminium alloy has a relatively low mass density and a high rigidity, so its eigenfrequencies are over the frequency range of 0–220Hz.

As the third control group, the numerical verifications in subsection **17.3.4** show that the number of panel layers can influence the resonance and anti-resonance distributions of the fluid response point. And the panel's lateral-boundary-constrained condition of the third control group is the first condition (see **Figure 17.5**). Theoretically, the number of eigenfrequencies positively correlates with the number of resonances and anti-resonances. But we can find an abnormal situation in the frequency response function of this control group. On the one hand, the one-layer panel has more resonance and anti-resonance points than the two-layer and three-layer panels (see **Figure 17.10**, **Figure 17.23**, and **Figure 17.24**).

On the other hand, the one-layer panel has fewer eigenfrequencies than the two-layer and three-layer panels (see **Table 17.3** and corresponding descriptions in subsection **17.4.1**). One reasonable explanation of the abnormal phenomena is that different panel layers in the multi-layer structure interfere with each other. Specifically, different panel layers keep a distance from each other, and the incoming acoustic waves on different panel layers will have phase lags because of the distance between them. Once the adjacent panel layers have the same eigenvalues, their resonances and anti-resonances will comprehensively interfere. Then, the multi-layer interference effects may diminish some resonances and anti-resonances.

17.4.4 Theoretical Arguments of Low-Frequency Sensibilities

The low-frequency sensibility of a mechanical structure depends on the number of its eigenfrequencies distributed in the low-frequency range. Thus, controlling low-frequency sensibility needs to control three factors. The first is the value of eigenfrequency. Then, the second is the number of eigenfrequencies in the low-frequency range. After that, the third is the eigenfrequencies distribution in the given frequency range.

Let us discuss the values of eigenfrequency in the following four cases. From **Figure 17.25** to **Figure 17.28**, the models are the melamine cube, the aluminium cube, the melamine seat with centre aluminium, and the aluminium seat with centre melamine, respectively. Subplot (a) in these four cases is the XY-plane view of the cubes. And these cubes' lateral boundaries are fixed. For reviewing the layout of cells, please see **Figure 15.2**. Although these four cubes are continuous substances, specific discretized models can emphasize their characteristics. Thus, subplot (b) is the simplified structure of the corresponding cube with reasonable exaggerations.

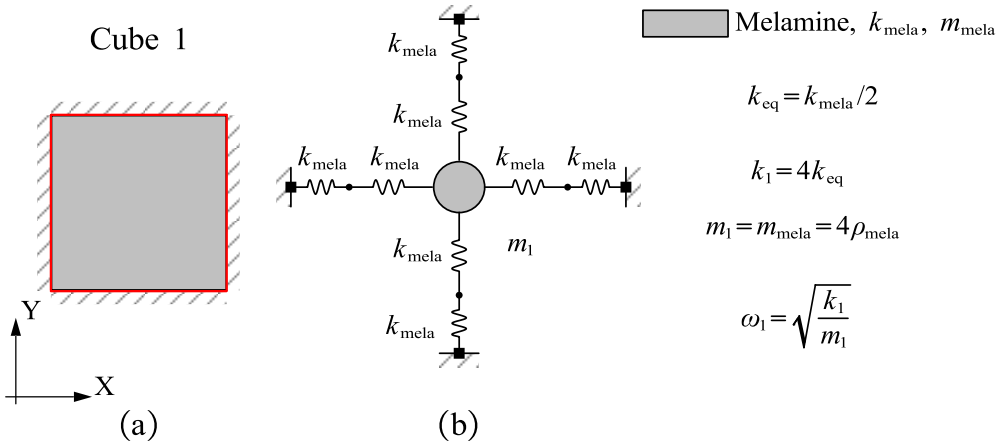


Figure 17.25: The melamine cube and its simplified model.

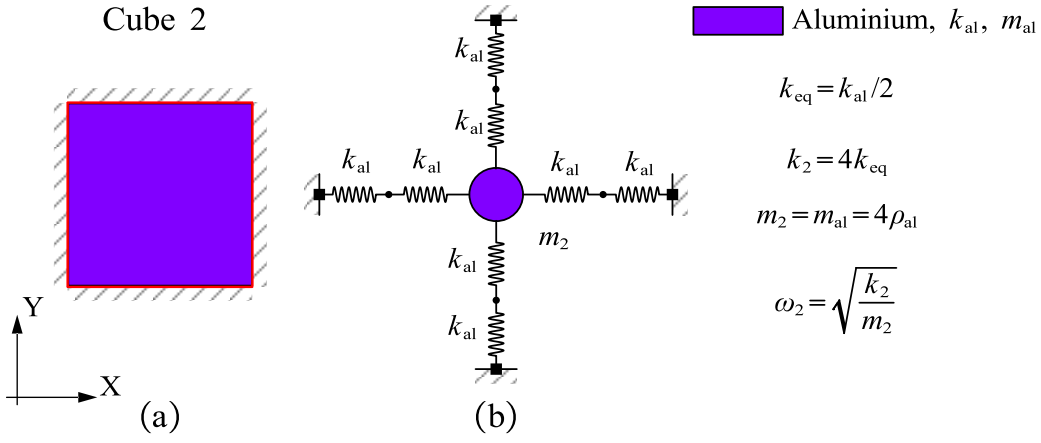


Figure 17.26: The aluminium cube and its simplified model.

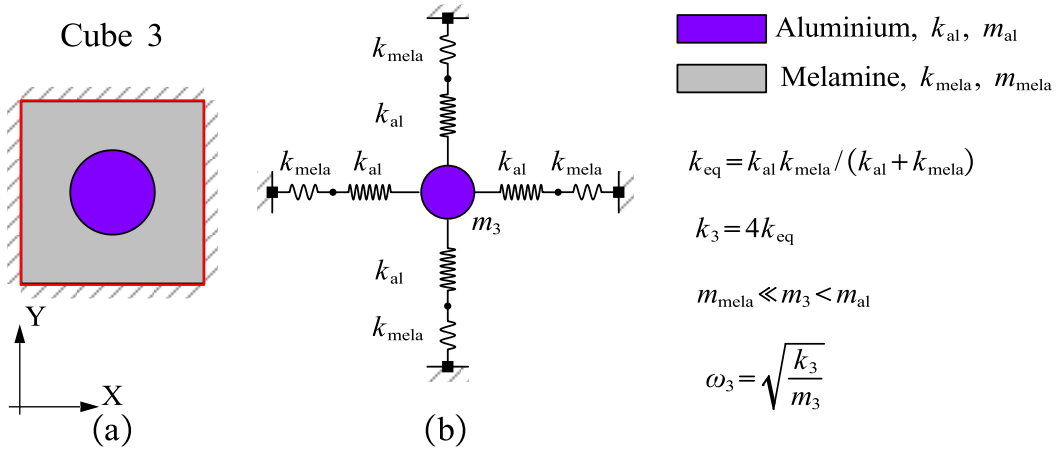


Figure 17.27: The centre-mass-intensive cube and its simplified model.

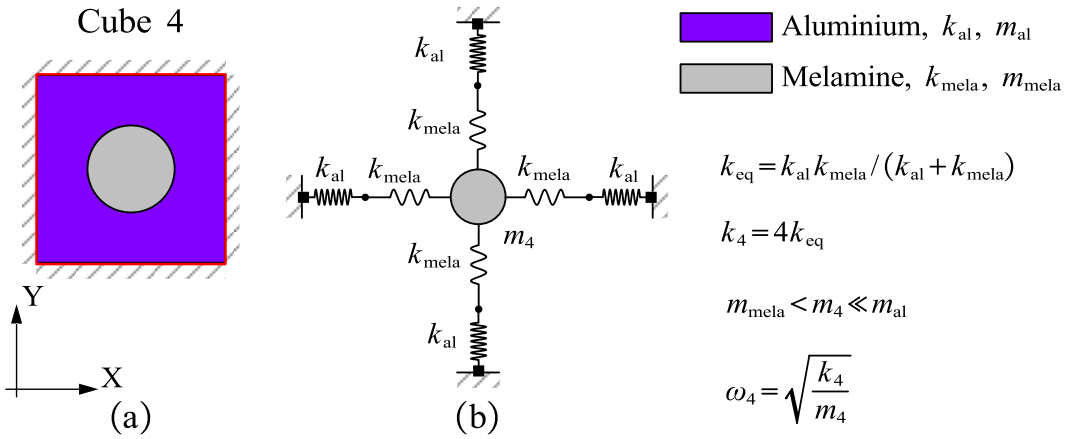


Figure 17.28: The centre-mass-diluted cube and its simplified model.

In these four figures, k_{mela} and m_{mela} is the effective stiffness and effective centre mass of melamine; k_{al} and m_{al} is the effective stiffness and effective centre mass of aluminium; k_{eq} is the equivalent stiffness of composite materials; k_1 to k_4 are the total effective stiffness of the four cubes; m_1 to m_4 are the total effective centre mass of the four cubes; ω_1 to ω_4 are the effective eigenfrequencies of the four cubes.

For discussing the eigenfrequencies of these four cubes, we need to review the property of the aluminium alloy and the melamine foam in subsections **2.2.2** and **2.2.3**. **Table 17.11** lists the material properties of the aluminium alloy and the melamine foam. Since melamine foam is an orthotropic material, its stiffness is not homogeneous. Thus, the melamine foam's stiffness is between the lowest and the highest elastic moduli. Supposing the volume of these four cubes is $4\,m^3$, and their centre mass highly depends on the centre material. Then, based on the formulas in **Figure 17.25** to **Figure 17.28**, the effective mass m_i, stiffness k_i, and eigenfrequency ω_i of the four cubes can be estimated. And the estimated effective mechanical properties of these four cubes are listed in **Table 17.12**. For quantitative analyses, these

estimated mechanical properties of the four cubes are competent in convincing the low-frequency sensibility.

The estimated eigenfrequencies in **Table 17.12** convince us that the third cube has the highest low-frequency sensibility. And the panel in periodic structure with the third cube can get the extra gradient of mass-stiffness dispersion. Thus, the periodic structure with the third cube enhances the eigenfrequency's emergence in the low-frequency range. Finally, the theoretical arguments in this subsection are in concert with the testing results in chapter **16** and subsection **17.3.3** (aluminium-inclusion-controlled panels).

Table 17.11: The mass density and elasticity (aluminium alloy, melamine foam).

	ρ (kg/m^3)	k (Pa)	k/ρ
Aluminium Alloy	2700	6.89×10^{10}	2.55×10^{7}
Melamine Foam	9.6	$(1.01\sim4.48)\times10^{5}$	$(1.05\sim4.67)\times10^{4}$

Table 17.12: The estimated effective mechanical properties (four cubes).

$i=1,\cdots,4$	m_i (kg)	k_i (Pa)	ω_i (rad/s)	ω_i (Hz)
Cube 1	9.6×4	$(1.01\sim4.48)\times2\times10^{5}$	$73\sim153$	$12\sim24$
Cube 2	2700×4	$6.89\times2\times10^{10}$	3572	569
Cube 3	$\approx2700\times4$	$\approx(1.01\sim4.48)\times4\times10^{5}$	$\approx6\sim13$	$\approx1\sim2$
Cube 4	$\approx9.6\times4$	$\approx(1.01\sim4.48)\times4\times10^{5}$	$\approx103\sim216$	$\approx16\sim34$

17.4.5 Theoretical Arguments of Sound-Proof Capabilities

The sound-proof capability of acoustic materials is superimposed by acoustic-wave reflection and absorption. The reflection means the material impedes the acoustic-wave transfer, while the absorption means the material consumes the acoustic-wave energy. We know that the damping effect is the main mechanism of energy consumption. Without damping, a system is generally conservative. However, solid material's damping is very small. It is reasonable to believe that the solid material's energy absorption does not dominate the sound-proof effect. Thus, the impeding effect of solid material is the main factor in diminishing sound.

In this paper, the fundamental layout of the fluid-structure systems consists of an outside air domain, an intermediate solid panel, and an inside air domain. Thus, the acoustic wave travels from outside to inside through two-stage transfers. The first stage is the outside air

domain motivates the solid media by the traction force on the interface between them. Then, the second stage is the solid media drives the inside air domain by the traction force on the interface between them. At this moment, a proper mathematical model is necessary to elaborate on the acoustic wave travelling process.

Please review equations **(16.2)**, **(16.4)**, and **(16.5)** in subsection **17.2.2**; these scattered wave equations are competent in describing the acoustic wave travelling process. Since the structural damping of the aluminium alloy and melamine foam is very small in the frequency range of 0–220Hz, the structural damping can be neglected in the following discussion. Thus, equation **(16.4)** can be further simplified by neglecting the structural damping term $[C_{s(\eta s)}]^{NM}$:

$$\begin{aligned} -\omega^2[M_s]^{NM}\{\tilde{U}_s\}^{M}+i[C_{s(\eta s)}]^{NM}\{\tilde{U}_s\}^{M}+[K_s]^{NM}\{\tilde{U}_s\}^{M} \\ =-[S_{fs}]^{NQ}\left(\{\tilde{P}_f\}_I^{Q}+\{\tilde{P}_f\}_S^{Q}\right) \end{aligned} \tag{16.8}$$

And the new structural field governing equation can be expressed as:

$$-\omega^2[M_s]^{NM}\{\tilde{U}_s\}^{M}+[K_s]^{NM}\{\tilde{U}_s\}^{M}=-[S_{fs}]^{NQ}\left(\{\tilde{P}_f\}_I^{Q}+\{\tilde{P}_f\}_S^{Q}\right) \tag{16.9}$$

Then, the structural displacement can be solved by combining Equations **(16.2)**, **(16.4)**, and **(16.9)**:

$$\begin{aligned} &\left(-\omega^2[M_s]^{NM}-\omega^2[S_{fs}]^{NQ}\left(-\omega^2[M_f]^{QP}+[K_f]^{QP}\right)^{-1}[S_{fs}]^{PM}+[K_s]^{NM}\right)\{\tilde{U}_s\}^{M} \\ &=-[S_{fs}]^{NQ}\{\tilde{P}_f\}_I^{Q}+[S_{fs}]^{NQ}\left(-\omega^2[M_f]^{QP}+[K_f]^{QP}\right)^{-1}[S_{fs}]^{PM}\{\tilde{T}_s\}_I^{M} \end{aligned} \tag{16.10}$$

In equation **(16.10)**, the incident pressure $\{\tilde{P}_f\}_I^{Q}$ and incident traction force $\{\tilde{T}_s\}_I^{M}$ by acoustic source are known parameters. Supposing the fluid properties, the fluid-structural interface, the acoustic source, and the angular frequency are constant, equation **(16.10)** can be further simplified as:

$$\left(-\omega^2[M_s]^{NM}+\text{C.Fluid}+[K_s]^{NM}\right)\{\tilde{U}_s\}^{M}=\text{C.Source} \tag{16.11}$$

In equation **(16.11)**, the constant fluid part is:

$$\text{C.Fluid}=-\omega^2[S_{fs}]^{NQ}\left(-\omega^2[M_f]^{QP}+[K_f]^{QP}\right)^{-1}[S_{fs}]^{PM} \tag{16.12}$$

Then, the constant source part is:

$$\text{C.Source}=-[S_{fs}]^{NQ}\{\tilde{P}_f\}_I^{Q}+[S_{fs}]^{NQ}\left(-\omega^2[M_f]^{QP}+[K_f]^{QP}\right)^{-1}[S_{fs}]^{PM}\{\tilde{T}_s\}_I^{M} \tag{16.13}$$

Finally, the mathematical model for describing the acoustic wave transfers can be expressed as follows:

$$\begin{gathered}\left(-\omega^2[M_s]^{NM}+\text{C.Fluid}+[K_s]^{NM}\right)\left\{\tilde{U}_s\right\}^M=\text{C.Source}\\ \Downarrow\\ -\omega^2\left\{\tilde{U}_s\right\}^M=\left\{\tilde{T}_s\right\}_I^M+\left\{\tilde{T}_s\right\}_S^M\\ \Downarrow\\ \left(-\omega^2[M_f]^{PQ}+[K_f]^{PQ}\right)\left\{\tilde{P}_f\right\}_S^Q=[S_{fs}]^{PM}\left\{\tilde{T}_s\right\}_S^M\end{gathered} \tag{16.14}$$

For more information about the meaning of the terms in Equations **(16.8)** to **(16.14)**, please review section **11.3**.

Let us concern equation **(16.14)** and make it a reference. In the first stage of acoustic wave transfer, the outside air domain motivates the solid media through the "C.Source". The motivation result is the structural displacement $\{\tilde{U}_s\}^M$, which depends on the matrix equation $[K_s]^{NM}-\omega^2[M_s]^{NM}$. Then, since the incident traction force $\{\tilde{T}_s\}_I^M$ and the angular speed are fixed, the scattered traction force depends on the structural displacement $\{\tilde{U}_s\}^M$. In the second stage of acoustic wave transfer, the solid media drives the inside air domain through the scattered traction force $\{\tilde{T}_s\}_S^M$. And the result is the scattered fluid pressure $\{\tilde{P}_f\}_S^Q$, which depends on the traction force since the fluid properties $[M_f]^{PQ}$ and $[K_f]^{PQ}$ are assumed to be unchanged. Thus, a smaller scattered fluid pressure $\{\tilde{P}_f\}_S^Q$ needs a smaller scattered traction force $\{\tilde{T}_s\}_S^M$. And a smaller traction force needs a smaller structural displacement $\{\tilde{U}_s\}^M$. Finally, the way is to control the matrix equation $[K_s]^{NM}-\omega^2[M_s]^{NM}$.

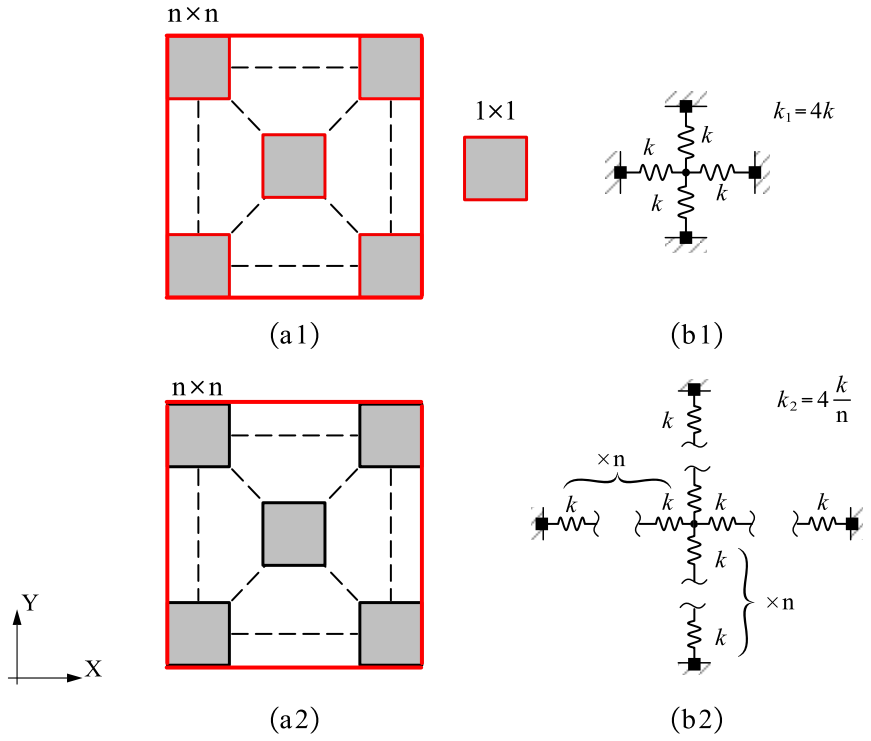


Figure 17.29: The effective stiffness (panels with two lateral-boundary-constrained conditions).

In the matrix equation $[K_s]^{NM} - \omega^2 [M_s]^{NM}$, the structural stiffness $[K_s]^{NM}$ and structural mass $[M_s]^{NM}$ terms are adjustable. However, modifying the structural stiffness term is easier than the structural mass term. A simple idea to change the structural stiffness term is to change the solid media's degree of constraint, and this manipulation changes little of the structural mass term. In comparison, changing the structural mass term needs changing the material's intrinsic property, such as using composite techniques, and this manipulation will modify the structural stiffness term simultaneously.

Figure 17.29 displays two solid panels with the same number of unit cells but with different lateral-boundary-constrained conditions. And the number of unit cells is n. The parameter k for each cell is a general stiffness, and its value equals a quarter of the unit cell's total stiffness. Then, the parameter $k_1 = 4k$ is the effective stiffness of the panel in the first constraint condition. In comparison, the parameter $k_2 = 4k/\mathrm{n}$ is the effective stiffness of the panel in the second constraint condition. Then, based on subplots (b1) and (b2), we can perceive that more panel's lateral boundaries are constrained, making the panel more rigid.

We can see that each unit cell of the solid panel in subplot (a1) is constrained (the cell's lateral boundaries are highlighted in red). Thus, the panel's effective stiffness is equivalent to a cell's effective stiffness in the first constraint condition. And subplot (b1) is a simplified sketch of a cell's stiffness model. This case is analogous to the fourth lateral-boundary-constrained condition in subsection **17.3.1**. For more information, please see **Figure 17.8** and read the corresponding descriptions.

In comparison, the solid panel in subplot (a2) is constrained with four lateral boundaries (lateral boundaries are highlighted in red). And subplot (b2) is a simplified sketch of the panel's stiffness model. This case is analogous to the first lateral-boundary-constrained condition in subsection **17.3.1**. For more information, please see **Figure 17.5** and read the corresponding descriptions.

From the simulation results listed in subsections **17.3.1** and **17.3.2**, we know that the sound transmission loss increases with the increase of the panel's constrained lateral boundaries. And based on the upper discussions, it is reasonable to believe that the lateral-boundary constraint changes the structural stiffness matrix $[K_s]^{NM}$. Finally, the value of the scattered pressure $\{\tilde{P}_f\}_S^O$ of the inside air domain becomes lower than the incident pressure $\{\tilde{P}_f\}_I^O$ from the outside air domain.

Finally, let us discuss the soundproof capability of the following three cases regarding mass density and stiffness. Please review the fluid-structure systems "1 Layer P400.CONS4 AL8mm" in **Figure 17.13** in subsection **17.3.1**, "1 Layer Melamine CONS4" in **Figure 17.17** in subsection **17.3.2**, and "1 Layer Aluminium" in **Figure 17.21** in subsection **17.3.3**.

It is necessary to note that the intermediate panels of the system "1 Layer P400.CONS4 AL8mm" and the system "1 Layer Melamine CONS4" have the same lateral-boundary-constrained condition (see **Figure 17.8**). But their soundproof capability is different. The latter is more competent in attenuating sound than the former. The reason is that the latter consists of pure melamine foam, and the former consists of metamaterial (see **Figure 17.25** and **Figure 17.27**). From **Table 17.12**, the estimated value of the unit cell convinces us that the metamaterial's stiffness is approximately 2 times larger than that of the melamine foam. Still, the metamaterial's mass density is approximately 281 times larger than the melamine foam. It is reasonable to believe that the items in the matrix equation $[K_s]^{NM} - \omega^2 [M_s]^{NM}$ of the

metamaterial have a smaller magnitude than that of the melamine foam. Based on equation **(16.14)**, the items' magnitude in the scattered fluid pressure $\{\widetilde{P}_f\}_S^Q$ of the inside air domain of the system "1 Layer P400.CONS4 AL8mm" is higher than that of the system "1 Layer Melamine CONS4".

Additionally, it is necessary to note that the panels of the system "1 Layer Melamine CONS4" and the system "1 Layer Aluminium" have analogous soundproof capabilities. The former has a highly lateral-boundary-constrained condition (see **Figure 17.5**), while the latter has a slightly lateral-boundary-constrained condition (see **Figure 17.8**). From **Table 17.11**, we can see that the ratio between the stiffness and mass density of the aluminium alloy is much higher than that of the melamine foam. However, the melamine panel becomes much more rigid because of the highly lateral-boundary-constrained condition. It is reasonable to believe that the ratio between the stiffness and mass density of the highly lateral-boundary-constrained melamine panel is close to that of the slightly lateral-boundary-constrained aluminium panel. Thus, the items in the matrix equation $[K_s]^{NM} - \omega^2[M_s]^{NM}$ of the two panels have closed magnitudes. Based on equation **(16.14)**, the items' magnitude in the scattered fluid pressure vector $\{\widetilde{P}_f\}_S^Q$ of the inside air domain of the system "1 Layer Melamine CONS4" is close to that of the system "1 Layer Aluminium".

Chapter 18

Conclusions

18.1 Paper Conclusions

As mentioned in the abstract, this paper has three main contents: the simulation strategy designs, the acoustic metamaterial designs, and the FEM-software-based theoretical models of fluid-structure systems.

18.1.1 Conclusions of Simulation Strategy Designs

We know that the metamaterial's chemical and physical properties are not the concerning point. Thus, there is no ready-made simulation software expert in metamaterial designs and analyses. Since the acoustic metamaterials in this paper are analogous to mechanical systems, FEM mechanical software is a reasonable choice. Therefore, the two main challenges the simulation strategy must overcome are massive intermediate data and complex simulation environments.

As the first challenge, the massive intermediate data mainly comes from the dynamic process. Specifically, the massive intermediate data in the first stage is the discretized model of the many periodic structures since each meta-atom must be sufficiently meshed to maintain enough information. Then, the massive intermediate data in the second stage may be dynamically generated by the eigenvalue extraction or matrix factorization process of the data from the first stage. Besides the data size from the first stage, the amount of intermediate data in the eigenvalue extraction process depends on the number of eigenmodes. In contrast, the matrix factorization process depends on the number of excitation frequencies.

Additionally, if the model is frequency-dependent, the model rebuilding mechanism will regenerate a multi-time discretized model with each property-altering frequency. Furthermore, if the solution sequence adopts superelement techniques, the dynamic reduction is another source that generates massive intermediate data. All in all, simulation software must be able to control the massive intermediate data by the solving system, and the solving system consists of the functional modules, the database, and the executive system. This way, all possible

solution sequences are predefined by the software's logic layout, and the database convinces the internal data conventions. Thus, the intermediate data's lifetime is traceable and manageable, which is the simulation software's data-flow-control capability. A simulation cannot be accomplished by self-build codes or mathematical software.

As the second challenge, the simulation complexity comes from the meta-atom's frequency dependency, the interactions between fluid and structural components, and the effective acoustic sources. And the simulation software may not be able to supply a proper simulation environment to cover these simulation complexities. Specifically, Nastran cannot model diffuse incident field, while Abaqus cannot model frequency-dependent orthotropic material. In contrast, Actran can conquer these obstacles. However, Actran does not have a competent solving system to process the simulations efficiently.

Besides the two simulation challenges, the best simulation strategy designs must consider efficiency and accuracy in different application scenarios. The application scenarios in this paper include the real eigenvalue extraction analyses, the steady-state dynamic analyses, and the steady-state vibro-acoustic analyses.

For practical accuracy of eigenvalue extraction analyses, let us review section **14.4**. Then, the conclusion is that Nastran has the highest practical accuracy in eigenvalue extraction analyses; its "Ordinary Method", "ACMS Method", "List Superelement Method", and "Image Superelement Method" have the same accuracy level.

For the relative accuracy and efficiency of eigenvalue extraction analyses and steady-state dynamic analyses, let us review chapter **15** and see section **15.4** and subsection **15.1.8**. In eigenvalue extraction analyses, the best numerical method is Nastran's "ACMS Method". Then, in steady-state dynamic analyses of a frequency-dependent system, the best simulation strategy is Nastran's modal-space-based analyses with the "Ordinary Method" to process the model that adopts the "Classic Frequency-Dependent Material" pattern. At the same time, one of the most efficient simulation strategies is Nastran's physical-space-based analyses with the "List Superelement Method" to process the model that adopts the "Viscoelastic Material" pattern. Still, this simulation strategy is not the most accurate one. After that, the best simulation strategy in steady-state dynamic analyses of a frequency-dependent system is Nastran's modal-space-based analyses with the "ACMS Method". Finally, the conclusion of simulation strategies is that Nastran has the most efficient and accurate simulation strategies to execute structural analyses of acoustic metamaterials. Specifically, the conclusion of simulation spaces is that modal-space-based analyses are more efficient than physical-space-based analyses. Then, the conclusion of frequency-dependent material patterns is that the "Viscoelastic Material" pattern is more efficient than the "Classic Frequency-Dependent Material" pattern.

Additionally, the conclusion of the superelement correlated techniques in Nastran is that these techniques are improper in simulations with model-rebuilding operations. The "List Superelement Method" and the "Image Superelement Method" do not support the model-rebuilding mechanism, and the "ACMS Method" is deficient under the model-rebuilding mechanism. In dynamic analyses with many substructures, the dynamic reduction of superelement techniques is dispensable and can be replaced with modal reduction. The extra cost of dynamic reduction is the superelement generations and residual data recovery. In contrast, the extra cost of the modal reduction is the process of eigenvalue extraction and physical data recovery. However, modal reduction can condense the original system much more than dynamic reduction. Thus, the exact superiority of the superelement techniques in the

dynamic analysis of a system with many substructures is not the dynamic reduction but the independence of substructures. Once the substructures are independent, they can be processed by parallel calculations. And this is the reason that "ACMS Method" is very efficient in eigenvalue extraction analyses.

For steady-state vibro-acoustic analyses, please review chapter **17** and see subsection **17.2.3**. Then, the conclusion is that the most efficient simulation strategies in steady-state vibro-acoustic analyses need Nastran cooperates with Actran. Specifically, in the beginning, the eigenmodes and base-state modal damping factor of fluid-structure systems are extracted through Nastran's "ACMS Method". The extracted base-state modal damping factor is called the "Composite Modal Damping", and its calculation needs Nastran's DMAP algorithms. Then, externally adding a frequency function to the base-state modal damping factor to make it becomes a frequency-dependent parameter. And the frequency function is a numerical trick based on the accuracy requirements. This numerical trick is efficient and reasonable since the damping factor is not a dominant parameter of the acoustic metamaterial. In the end, the extracted eigenmodes and estimated frequency-dependent modal damping factor of fluid-structure systems will be sent into Actran to do the modal-space-based vibro-acoustic analyses. And in this process, Actran supplies the diffuse incident field. This way, Nastran and Actran can hide their shortcomings and mutually enhance their merits in joint simulations with complex simulation environments.

18.1.2 Conclusions of Acoustic Metamaterial Designs

We know that the low-frequency (0-220Hz) sensibility and soundproof capability are two gauges to evaluate the acoustic metamaterial designs. In general, these two dynamic properties are naturally independent or mutually exclusive, which means the acoustic metamaterial may be sensible in the low-frequency range; otherwise, it may have a strong soundproof capability in the low-frequency range. However, suppose there is a method to isolate the resonance and anti-resonance phenomena and let them diminish each other. In that case, the low-frequency sensibility can coincide with the soundproof capability.

Let us discuss low-frequency sensibility. Based on the eigenvalue extraction analyses in Chapters **15** and **16**, we know that increasing the size of the acoustic metamaterial samples or increasing the aluminium alloy inclusions can increase the low-frequency sensibility. In other words, the former operation increases the number of meta-atoms, while the latter enhances the centre-mass-intensive form of meta-atoms. Thus, one of the reasonable explanations is that these two operations can increase the opportunity for eigenfrequencies' emergence and distribution in the low-frequency range. At the same time, based on the steady-state dynamic analyses in chapter **15**, the resonance and anti-resonance phenomena of the structural response point on acoustic metamaterial samples are a side view that convinces the low-frequency sensibility.

Based on the eigenvalue extraction analyses in chapter **17**, we know that the panel's lateral-boundary-constrained conditions and the number of panel layers are two new variables that can influence the properties of acoustic metamaterials. And the conclusion is that increasing the panel's constrained lateral boundaries can decrease the low-frequency sensibility while increasing the number of panel layers can increase the low-frequency sensibility. In essence, the former operation is equivalent to subdividing the whole panel into smaller ones, which is analogous to decreasing the panel size. In contrast, the latter operation is equivalent

to adding panels. At the same time, based on the steady-state vibro-acoustic analyses in chapter **17**, the resonance and anti-resonance phenomena of the fluid response point in the inside air domain can partially convince the panel's low-frequency sensibility. The conclusion is that increasing the panel's constrained lateral boundaries can attenuate the resonance and anti-resonance phenomena while increasing the panel's aluminium inclusions can shift and enhance the resonance and anti-resonance. However, the abnormal situation happens when the number of panel layers increases. One of the reasonable explanations for the abnormal situation is that the resonance and anti-resonance of different layers may interfere with or diminish each other because of the phase lag of the acoustic waves on different layers.

Let us discuss the soundproof capability. Based on the steady-state vibro-acoustic analyses in chapter **17**, the acoustic metamaterial can amplify the sound at resonance frequency range and diminish sound at anti-resonance or normal frequency range. The conclusion is that increasing the acoustic metamaterial's aluminium inclusions, or the number of layers, can moderately increase its soundproof capability in the low-frequency range. However, increasing the panel's constrained lateral boundaries can considerably increase the soundproof capability in the low-frequency range. One of the reasonable explanations for this phenomenon is that increasing the panel's constrained lateral boundaries can increase the rigidity of the panel.

Moreover, increasing the panel's constrained lateral boundaries has a better effect on the pure melamine foam than on the acoustic metamaterial. The reason is that the acoustic metamaterial is more sensible than the pure melamine foam in the low-frequency range. At the same time, subsections **17.4.4** and **17.4.5** give simplified deductions of the fundamental theory between the panel's lateral-boundary-constrained conditions, the panel's low-frequency sensibility, and the panel's rigidity. Then, the conclusion is that the soundproof capability is directly correlated with the material's mass density and bulk modulus. And the structural damping effects are too small to consume the acoustic wave energy. Based on equation **(16.14)**, the simplest way to increase the soundproof capability is to make the material more rigid.

18.1.3 Conclusions of Practical Vibro-Acoustic Theories

Based on corresponding elaborations in chapters **10**, **11**, **12**, and **13**, we know that the boundary conditions of acoustic components, the acoustic sources, and the fluid-structure interface are the key points for building the FEM model of fluid-structure systems. At the same time, the scattered wave equation of the fluid-structure system is indispensable for governing the dynamic process in steady-state vibro-acoustic analyses. In the scattered wave equation, the incident wave and scattered wave are linear and independent. Their relations are tied with fluid boundary conditions and boundary tractions between different media, such as the interactions between fluid and structure. Subsequently, the diffuse incident field is the only choice as the acoustic source in FEM software for getting the steady-state frequency response function of fluid response points.

As the FEM software, Abaqus, Actran, and Nastran can model fluid and structural components and build interfaces between them. However, Nastran cannot model the diffuse incident field and has no ready-made fluid boundary condition models. Meanwhile, Actran can model the diffuse incident field and has ready-made fluid boundary condition models. Still, Actran's fluid boundary condition models are not flexible enough because users cannot customize their endogenous settings. In comparison, Abaqus can model the diffuse incident field and has many ready-made fluid boundary condition models.

Furthermore, Abaqus's fluid boundary condition models are flexible because users can customize their settings. However, due to the low efficiency and moderate accuracy, this paper does not choose Abaqus to do the vibro-acoustic analyses for testing the acoustic metamaterial samples. And except for the fluid-structure interface, the other fluid boundary conditions are unimportant. Thus, Abaqus is dispensable in this paper.

18.2 Outlooks

Based on the study in this paper, there is no universal simulation strategy proper for all kinds of simulation tasks. The acoustic metamaterial designs and the simulation strategy designs are mutually dependent. In a future study, Nastran may be a good choice for solving the linear problems of structural systems. It can cooperate with Actran to do the vibro-acoustic analyses of some simple fluid-structure systems. However, if the fluid boundary conditions are complex, Abaqus may have more merits.

For the single-layer acoustic metamaterial samples tested in this paper, their fortes are low-frequency sensibility, and the sensibility's range and intensity are adjustable. Thus, based on the centre-mass-intensive cells, the single-layer acoustic metamaterial with the periodic structure has the potential to make low-frequency sound transducers. In contrast, for the multi-layer acoustic metamaterial tested in this paper, their capability for interfering and diminishing resonance and anti-resonance effects needs further investigation. Then, the multi-layer acoustic metamaterial may potentially make low-frequency sound filters.

Based on the corresponding simulation practices and analyses in this paper, the acoustic wave reflection needs more attention than the acoustic wave absorption in the future study of the material's soundproof capability. The reason is that the applicable soundproof materials always need to be made as thin as possible, and the structural damping of solid materials is always very small. Thus, this kind of material cannot effectively absorb acoustic wave energy. Furthermore, if the acoustic source has a permanent power output, the effects of acoustic wave absorption are moderate. The corresponding simulations convince us that reflecting the acoustic wave is equivalent to partially transferring it. And the transferring mechanism of the intermediate material is dominated by the mass density and bulk modulus. Specifically, increasing the value of bulk modulus can increase the intermediate material's soundproof capability, and an easy way is to increase the material's constrained lateral boundaries. It is convinced that a highly lateral-boundary-constrained condition can give a panel a considerable soundproof capability. The panel only needs to consist of soft, light material, such as melamine foam. Therefore, future studies can be dedicated to designing a casing or casing matrix (see **Figure 17.5** to **Figure 17.8**) with rigid material to constrain the lateral boundaries of the soundproof material samples. The wall of the casing or casing matrix can be very thin, so they are light enough. There is no need to worry about their vertical rigidity. For example, a knife is thin but vertically very rigid that can cut tough substances and get few vertical deformations. Then, the focal point is how to sufficiently connect the lateral boundaries of the soundproof material to the wall of the casing or casing matrix.

References

[1] Z. Yang, J. Mei, M. Yang, N. H. Chan and P. Sheng, "Membrane-type acoustic metamaterial with negative dynamic mass," *Physical Review Letters*, vol. 101, no. 20, 2008.

[2] J. Mei, G. Ma, M. Yang, J. Yang and P. Sheng, "Dynamic mass density and acoustic metamaterials," in *Acoustic Metamaterials and Phononic Crystals*. Berlin: Springer-Verlag, 2013, pp. 159-199.

[3] S. A. Cummer, J. Christensen and A. Alù, "Controlling sound with acoustic metamaterials," *Nature Reviews Materials,* vol. 1, no. 3, 2016.

[4] Z. Liu, X. Zhang, Y. Mao, Y. Y. Zhu, Z. Yang, C. T. Chan and P. Sheng, "Locally resonant sonic materials," *Science,* vol. 289, no. 5485, 2000.

[5] S. H. Lee, C. M. Park, Y. M. Seo, Z. G. Wang and C. K. Kim, "Acoustic metamaterial with negative density," *Physics Letters A*, vol. 373, no.48, 2009.

[6] N. Fang, D. Xi, J. Xu, M. Ambati, W. Srituravanich, S. Sun and X. Zhang, "Ultrasonic metamaterials with negative modulus," *Nature Materials*, vol. 5, no. 6, 2006.

[7] S. H. Lee, C. M. Park, Y. M. Seo, Z. G. Wang and C. K. Kim, "Acoustic metamaterial with negative modulus," *Journal of Physics: Condensed Matter*, vol. 21, no.17, 2009.

[8] S. H. Lee, C. M. Park, Y. M. Seo, Z. G. Wang and C. K. Kim, "Composite acoustic medium with simultaneously negative density and modulus," *Physical Review Letters*, vol. 104, no. 5, 2010.

[9] J. Christensen, L. Martín-Moreno and F. J. García-Vidal, "All-angle blockage of sound by an acoustic double-fishnet metamaterial", *Applied Physics Letters*, vol. 97, no. 13, 2010.

[10] H. M. Shruthi and V. Doddaswamy, "Vibration characteristics of Al6061-TiO2-Gr hybrid metal matrix composites-an experimental approach," *International Journal of Engineering Research in Mechanical and Civil Engineering*, vol. 5, no.7, 2020.

[11] J. Cuenca, V. D. Kele and P. Göransson , "A general methodology for inverse estimation of the elastic and anelastic properties of anisotropic open-cell porous materials—with application to a melamine foam", *Journal of Applied Physics*, vol. 115, no. 8, 2014.

[12] Hexagon AB, "Solution sequences," in *MSC Nastran 2020 Service Pack 1: Reference Guide*. 2020 MSC Software Cooperation, 2020, pp. 336-434.

[13] Hexagon AB, “Hardware,” in *MSC Nastran 2020 Service Pack 1: High Performance Computing User’s Guide*. 2020 MSC Software Cooperation, 2020, pp. 22-40.

[14] Hexagon AB, “Database concepts,” in *MSC Nastran 2020 Service Pack 1: Reference Guide*. 2020 MSC Software Cooperation, 2020, pp. 452-459.

[15] Hexagon AB, “Utility tools and functions,” in *MSC Nastran 2020 Service Pack 1: Numerical Methods User’s Guide*. 2020 MSC Software Cooperation, 2020, pp. 17-34.

[16] Hexagon AB, “Direct matrix abstraction,” in *MSC Nastran 2020 Service Pack 1: DMAP Program’s Guide*. 2020 MSC Software Cooperation, 2020, pp. 1-38.

[17] Dassault Systèmes Simulia Corp, “Analysis procedures: Introduction,” in *Abaqus 6.14 Analysis User’s Guide Volume II: Analysis*. Dassault Systèmes, 2014, pp. 35-74.

[18] Hexagon AB, “Nast20200 command and input data,” in *MSC Nastran 2020 service pack 1: Reference Guide*. 2020 MSC Software Cooperation, 2020, pp. 10-34.

[19] Hexagon AB, “Introduction and fundamentals,” in *MSC Nastran 2020 Service Pack 1: Superelements and Modulus User’s Guide*. 2020 MSC Software Cooperation, 2020, pp. 1-34.

[20] Hexagon AB, “Introduction to dynamic analysis using superelements,” in *MSC Nastran 2020 Service Pack 1: Superelements and Modulus User’s Guide*. 2020 MSC Software Cooperation, 2020, pp. 383-436.

[21] Hexagon AB, “How to define a superelement,” in *MSC Nastran 2020 Service Pack 1: Superelements and Modulus User’s Guide*. 2020 MSC Software Cooperation, 2020, pp. 35-132.

[22] Hexagon AB, “Practical image superelement,” in *MSC Nastran 2020 Service Pack 1: Superelements and Modulus User’s Guide*. 2020 MSC Software Cooperation, 2020, pp. 677-752.

[23] Hexagon AB, “Executive control statements: DOMAINSOLVER,” in *MSC Nastran 2020 Service Pack 1: Quick Reference Guide*. 2020 MSC Software Cooperation, 2020, pp. 126-132.

[24] Hexagon AB, “Degree-of-freedom sets,” in *MSC Nastran 2020 Service Pack 1: Quick Reference Guide*. 2020 MSC Software Cooperation, 2020, pp. 1111-1116.

[25] Hexagon AB, “Damping,” in *MSC Nastran 2020 Service Pack 1: Dynamic Analysis User’s Guide*. 2020 MSC Software Cooperation, 2020, pp. 253-284.

[26] Hexagon AB, “Frequency response analysis,” in *MSC Nastran 2020 Service Pack 1: Dynamic Analysis User’s Guide*. 2020 MSC Software Cooperation, 2020, pp. 123-198.

[27] Hexagon AB, “Real eigenvalue analysis,” in *MSC Nastran 2020 Service Pack 1: Dynamic Analysis User’s Guide*. 2020 MSC Software Cooperation, 2020, pp. 41-122.

[28] Hexagon AB, “Direct solution of linear systems,” in *MSC Nastran 2020 Service Pack 1: Numerical Methods User’s Guide*. 2020 MSC Software Cooperation, 2020, pp. 75-82.

[29] Hexagon AB, “Matrix decomposition,” in *MSC Nastran 2020 Service Pack 1: Numerical Methods User’s Guide*. 2020 MSC Software Cooperation, 2020, pp. 57-74.

[30] Hexagon AB, “Iterative solution of systems of linear systems,” in *MSC Nastran 2020 Service Pack 1: Numerical Methods User’s Guide*. 2020 MSC Software Cooperation, 2020, pp. 83-104.

[31] Hexagon AB, “Case control commands: SMETHOD,” in *MSC Nastran 2020 Service Pack 1: Quick Reference Guide*. 2020 MSC Software Cooperation, 2020, p. 578.

[32] Hexagon AB, “Bulk data entries: ITER,” in *MSC Nastran 2020 Service Pack 1: Quick Reference Guide*. 2020 MSC Software Cooperation, 2020, pp. 2166-2168.

[33] Hexagon AB, “Viscoelastic material properties,” in *MSC Nastran 2020 Service Pack 1: Dynamic Analysis User’s Guide*. 2020 MSC Software Cooperation, 2020, pp. 747-756.

[34] Hexagon AB, “Coupled fluid-structure analysis,” in *MSC Nastran 2020 Service Pack 1: Dynamic Analysis User’s Guide*. 2020 MSC Software Cooperation, 2020, pp. 391-496.

[35] Dassault Systèmes Simulia Corp, “Spacial modelling: Assembly definition,” in *Abaqus 6.14 Analysis User’s Guide Volume I: Introduction, Spatial Modeling, Execution & Output*. Dassault Systèmes, 2014, pp. 311-330.

[36] Dassault Systèmes Simulia Corp, “Analysis procedures: Dynamic stress/displacement analysis,” in *Abaqus 6.14 Analysis User’s Guide Volume II: Analysis*. Dassault Systèmes, 2014, pp. 143-306.

[37] Dassault Systèmes Simulia Corp, “Modeling abstractions: Substructuring,” in *Abaqus 6.14 Analysis User’s Guide Volume II: Analysis*. Dassault Systèmes, 2014, pp. 703-740.

[38] Dassault Systèmes Simulia Corp, “Elastic mechanical properties: Linear viscoelasticity,” in *Abaqus 6.14 Analysis User’s Guide Volume III: Materials*. Dassault Systèmes, 2014, pp. 171-198.

[39] Dassault Systèmes Simulia Corp, “Other material properties: Mechanical properties,” in *Abaqus 6.14 Analysis User’s Guide Volume III: Materials*. Dassault Systèmes, 2014, pp. 593-620.

[40] Dassault Systèmes Simulia Corp, “Prescribed conditions: Loads,” in *Abaqus 6.14 Analysis User’s Guide Volume V: Prescribed Conditions, Constraints & Interactions*. Dassault Systèmes, 2014, pp. 109-210.

[41] Dassault Systèmes Simulia Corp, “Analysis procedures: Acoustic and shock analysis,” in *Abaqus 6.14 Analysis User’s Guide Volume II: Analysis*. Dassault Systèmes, 2014, pp. 505-533.

[42] MSC Software Belgium SA, *MSC Actran 2020 User's Guide Vol.2: Extended DAT Input File Syntax*. Free Field Technologies, 2019.

[43] MSC Software Belgium SA, "Actran calculation procedures," in *MSC Actran 2020 User's Guide Vol.1: Installation, Operations, Theory, and Utilities*. Free Field Technologies, 2019, pp. 51-56.

[44] MSC Software Belgium SA, "Component data block: MODAL_ACOUSTIC, MODAL_ELASTIC," in *MSC Actran 2020 User's Guide Vol.2: Extended DAT Input File Syntax*. Free Field Technologies, 2019, pp. 181-187, 202-210.

[45] MSC Software Belgium SA, "Analysis parameter data blocks: MODAL_DAMPING," in *MSC Actran 2020 User's Guide Vol.2: Extended DAT Input File Syntax*. Free Field Technologies, 2019, pp. 684-685.

[46] MSC Software Belgium SA, "Algebraic solvers," in *MSC Actran 2020 User's Guide Vol.1: Installation, Operations, Theory, and Utilities*. Free Field Technologies, 2019, pp. 57-80.

[47] MSC Software Belgium SA, "Material data blocks: MATERIAL, FLUID, ISOTROPIC_SOLID, ORTHOTROPIC_SOLID," in *MSC Actran 2020 User's Guide Vol.2: Extended DAT Input File Syntax*. Free Field Technologies, 2019, pp. 280-283, 301-307, 313-316.

[48] MSC Software Belgium SA, "Modal reduction," in *MSC Actran 2020 User's Guide Vol.1: Installation, Operations, Theory, and Utilities*. Free Field Technologies, 2019, pp. 507-512.

[49] MSC Software Belgium SA, "Use of superelements in vibro-acoustic simulations," in *MSC Actran 2020 User's Guide Vol.1: Installation, Operations, Theory, and Utilities*. Free Field Technologies, 2019, pp. 513-531.

[50] MSC Software Belgium SA, "Acoustic source: incident field of acoustic sources," in *MSC Actran 2020 User's Guide Vol.1: Installation, Operations, Theory, and Utilities*. Free Field Technologies, 2019, pp. 364-368.

[51] MSC Software Belgium SA, "Random vibro-acoustics: diffuse incident pressure field," in *MSC Actran 2020 User's Guide Vol.1: Installation, Operations, Theory, and Utilities*. Free Field Technologies, 2019, pp. 391-428.

[52] G. Genta, "Free vibration of damped systems," in *Vibration Dynamics and Control*. New York: Springer Science+Business Media, 2009, pp. 115-133.

[53] G. Genta, "Forced response in the frequency domain: damped systems," in *Vibration Dynamics and Control*. New York: Springer Science+Business Media, 2009, pp. 153-178.

[54] Hexagon AB, “Bulk data entries: MAT1, MAT1F, MAT2, MAT2F, MAT8, MAT8F, MAT9, MAT9F, MAT10, MAT10F,” in *MSC Nastran 2020 Service Pack 1: Quick Reference Guide*. 2020 MSC Software Cooperation, 2020, pp. 2207-2237.

[55] Hexagon AB, “Bulk data entries: TABLED1,” in *MSC Nastran 2020 Service Pack 1: Quick Reference Guide*. 2020 MSC Software Cooperation, 2020, pp. 3187-3189.

[56] Hexagon AB, “Case control commands: SDAMPING,” in *MSC Nastran 2020 Service Pack 1: Quick Reference Guide*. 2020 MSC Software Cooperation, 2020, pp. 558-559.

[57] Hexagon AB, “Executive control statements: ALTER, COMPILE, ENDALTER,” in *MSC Nastran 2020 Service Pack 1: Quick Reference Guide*. 2020 MSC Software Cooperation, 2020, pp. 112-120, 134.

[58] Hexagon AB, “Nastran data definition language (NDDL),” in *MSC Nastran 2020 Service Pack 1: DMAP Program’s Guide*. 2020 MSC Software Cooperation, 2020, pp. 1461-1478.

[59] Hexagon AB, “DMAP modules and statements: MATMOD, MATPCH, SMPYAD, SOLVE,” in *MSC Nastran 2020 Service Pack 1: DMAP Program’s Guide*. 2020 MSC Software Cooperation, 2020, pp. 2004-2062, 2470-2473.

[60] Hexagon AB, “Glossary: data block glossary,” in *MSC Nastran 2020 Service Pack 1: DMAP Program’s Guide*. 2020 MSC Software Cooperation, 2020, pp. 2609-2691.

[61] Dassault Systèmes Simulia Corp, “*Composite modal damping, *Damping, *Frequency, *Global damping, *Material, *Modal damping, *Solid section, *Steady state dynamics, *Step,” in *Abaqus 6.14 Keywords Reference Guide*. Dassault Systèmes, 2014, pp. 201, 441-444, 757-760, 799-800, 967-968, 985-990, 1315-1317, 1349-1352, 1357-1361.

[62] MSC Software Belgium SA, “Analyses and main data blocks: ACTRN, ACTRN_MODAL, FREQUENCY_DOMAIN,” in *MSC Actran 2020 User’s Guide Vol.2: Extended DAT Input File Syntax*. Free Field Technologies, 2019, pp. 48, 51, 58-62.

[63] MSC Software Belgium SA, “Analysis parameter data blocks: TABLE,” in *MSC Actran 2020 User’s Guide Vol.2: Extended DAT Input File Syntax*. Free Field Technologies, 2019, pp. 744-749.

[64] J. H. Ginsberg, “Principles and equations for multidimensional phenomena,” in *Acoustics–A Textbook for Engineers and Physicists–Volume I: Fundamentals*. Cham, Switzerland: Springer International Publishing AG, 2018, pp. 295-346.

[65] J. H. Ginsberg, “Plane waves: frequency-domain solutions,” in *Acoustics–A Textbook for Engineers and Physicists–Volume I: Fundamentals*. Cham, Switzerland: Springer International Publishing AG, 2018, pp. 191-294.

[66] J. H. Ginsberg, “Plane waves: time domain solutions,” in *Acoustics–A Textbook for Engineers and Physicists–Volume I: Fundamentals*. Cham, Switzerland: Springer International Publishing AG, 2018, pp. 91-189.

[67] D. Halliday, R. Resnick and J. Walker, “Waves–I,” in *Fundamental of Physics*. Hoboken, USA: John Wiley & Sons, 2015, p. 444.

[68] L. E. Kinsler, A. R. Frey, A. B. Coppens and J. V. Sanders, “The acoustic wave equation and simple solution,” in *Fundamentals of Acoustics*. New York: John Wiley & Sons, 2000, pp. 121-130, 133-135.

[69] L. E. Kinsler, A. R. Frey, A. B. Coppens and J. V. Sanders, “Radiation and reception of acoustic waves,” in *Fundamentals of Acoustics*. New York: John Wiley & Sons, 2000, pp. 171-176.

[70] Dassault Systèmes Simulia Corp, “*ACOUSTIC MEDIUM,” in *Abaqus 6.14 Keywords Reference Guide*. Dassault Systèmes, 2014, pp. 27-30.

[71] MSC Software Belgium SA, “Material data blocks: FLUID, WATER, PERFECT_GAS,” in *MSC Actran 2020 User’s Guide Vol.2: Extended DAT Input File Syntax*. Free Field Technologies, 2019, pp. 301-304, 317-320, 352-354.

[72] MSC Software Belgium SA, “Component data blocks: ACOUSTIC,” in *MSC Actran 2020 User’s Guide Vol.2: Extended DAT Input File Syntax*. Free Field Technologies, 2019, pp. 105-111.

[73] MSC Software Belgium SA, “Boundary condition data blocks,” in *MSC Actran 2020 User’s Guide Vol.2: Extended DAT Input File Syntax*. Free Field Technologies, 2019, pp. 389-576.

[74] Hexagon AB, “Bulk data entries: MAT10, PSOLID,” in *MSC Nastran 2020 Service Pack 1: Quick Reference Guide*. 2020 MSC Software Cooperation, 2020, pp. 2235-2236, 2941-2945.

[75] Dassault Systèmes Simulia Corp, “Procedures: Coupled fluid-solid analysis,” in *Abaqus 6.11 Theory Manual*. Dassault Systèmes, 2011, pp. 261-291.

[76] Dassault Systèmes Simulia Corp, “Loading and constraints: incident wave loading,” in *Abaqus 6.11 Theory Manual*. Dassault Systèmes, 2011, pp. 1099-1117.

[77] MSC Software Belgium SA, “Acoustic sources,” in *MSC Actran 2020 User’s Guide Vol.1: Installation, Operations, Theory, and Utilities*. Free Field Technologies, 2019, pp. 359-376.

[78] P. Langlet, A. C. Hladky and J. N. Decarpigny, “Analysis of the propagation of plane acoustic waves in passive periodic materials using the finite element method,” *Journal of the Acoustical Society of America*, vol. 98, no. 5, 1995.