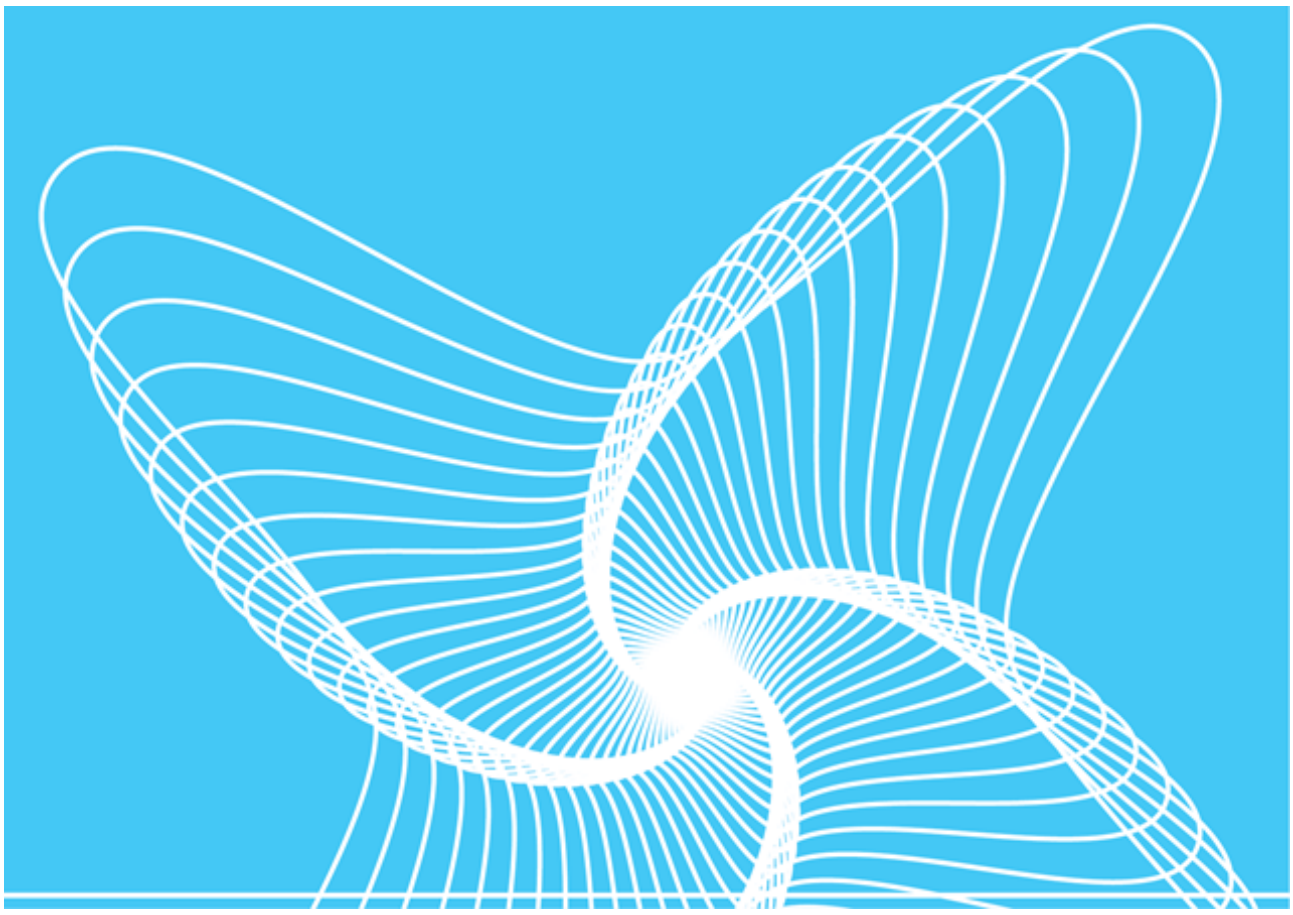
DNV·GL

SESAM USER MANUAL

SESTRA

Finite element analysis solver

Valid from program version 10.12.0



SAFER, SMARTER, GREENER

Sesam User Manual

Sestra

Date: 19/10/2020

Valid from program version 10.12.0

Prepared by DNV GL - Digital Solutions

E-mail support: software.support@dnvgl.com

E-mail sales: digital@dnvgl.com

©DNV GL AS. All rights reserved

This publication or parts thereof may not be reproduced or transmitted in any form or by any means, including copying or recording, without the prior written consent of DNV GL AS.

Table of contents

1 Introduction . . . 1
1.1 The Sestra 10.12.0 release . . . 1
1.2 Sestra: The Sesam finite element solver . . . 1
1.3 Sestra in the Sesam system . . . 2
1.4 How to read the manual . . . 2
1.5 System requirements . . . 3
1.6 Installation . . . 3
1.7 Public API . . . 3

2 Features of Sestra 10.12.0 . . . 4
2.1 Supported analyses . . . 4
2.1.1 Linear static analysis . . . 4
2.1.2 Dynamic time-domain forced response analysis — direct calculation . . . 4
2.1.3 Dynamic time-domain forced response analysis — modal superposition . . . 4
2.1.4 Dynamic frequency-domain forced response analysis — direct calculation . . . 5
2.1.5 Dynamic frequency-domain forced response analysis — modal superposition . . . 5
2.1.6 Free vibration analysis . . . 6
2.1.7 Linearized buckling analyis . . . 6
2.1.8 Tension-compression analysis . . . 7
2.1.9 Linear static analysis with nonlinear pile-soil interaction . . . 7
2.1.10 Postprocessing . . . 7
2.1.11 Stress stiffening . . . 7
2.2 Superelement analysis . . . 8
2.3 Static condensation in dynamic analysis . . . 9
2.4 Component mode synthesis reduction . . . 9
2.5 Features for dynamic spectral fatigue analysis . . . 9
2.5.1 Specification of periodic loading conditions . . . 10
2.5.2 Steady-state detection . . . 10
2.5.3 Calculation of base shear and overturning moment . . . 10
2.6 Generation and use of equivalent static loads . . . 10
2.7 Matrix output . . . 11
2.8 Model properties . . . 11
2.8.1 Stiffness properties . . . 11
2.8.2 Mass properties . . . 12
2.8.3 Hydrodynamic added mass for big volume structures . . . 12
2.8.4 Damping properties . . . 13
2.8.5 Load types . . . 14
2.8.6 Boundary conditions . . . 14
2.8.7 Initial conditions . . . 15
2.8.8 Multipoint constraints (linear dependencies) . . . 15
2.9 Elements library and element properties . . . 15
2.9.1 Element library . . . 15
2.9.2 Materials . . . 16
2.10 Input and output . . . 16
2.10.1 Input . . . 16
2.10.2 Output . . . 18
2.11 Linear solver options . . . 19
2.11.1 In-core and out-of-core matrix factorization . . . 19
2.11.2 Memory usage for loads . . . 19
2.11.3 Factorization algorithm . . . 19

3 User's guide . . . 20
3.1 Execution of Sestra 10.12.0 . . . 20
3.2 Linear static analysis . . . 20
3.3 Dynamic time-domain forced response analysis — direct calculation . . . 21
3.3.1 Overview . . . 21
3.3.2 Algorithm parameters . . . 21
3.3.3 Load specification . . . 21

3.3.4 Calculation of reaction forces and response-amplitude operators . . . 22
3.3.5 Input-file examples . . . 23
3.4 Dynamic time-domain forced response analysis — modal superposition . . . 26
3.4.1 Overview . . . 26
3.4.2 Input parameters . . . 26
3.5 Dynamic frequency-domain forced response analysis — direct calculation . . . 26
3.5.1 Overview . . . 26
3.5.2 Relevant commands . . . 27
3.5.3 Calculation of response-amplitude operators . . . 27
3.5.4 Input-file examples . . . 27
3.6 Dynamic frequency-domain forced response analysis — modal superposition . . . 29
3.6.1 Overview . . . 29
3.6.2 Input parameters . . . 29
3.7 Free vibration analysis . . . 29
3.8 Tension-compression analysis . . . 30
3.9 Postprocessing . . . 32
3.10 Stress stiffening analysis . . . 34
3.11 Superelement analysis . . . 35
3.12 Common info and warning messages . . . 36

4 Sestra input commands . . . 37
4.1 General description of input command format . . . 37
4.2 List of input commands . . . 38
BSEL . . . 38
CMAS . . . 38
DLOA . . . 40
DREA . . . 40
DRED . . . 41
DYMA . . . 41
ERST . . . 41
EIGA . . . 42
ELCH . . . 42
ELOP . . . 42
ELIM . . . 43
ESLE . . . 44
ESLI . . . 44
ESLU . . . 45
FRLO . . . 46
FRQL . . . 46
FRSP . . . 47
GENA . . . 48
GSTF . . . 48
HIST . . . 48
HYDM . . . 50
IDTY . . . 50
INAM . . . 51
INDT . . . 51
INTP . . . 51
ITOP . . . 52
LCAS . . . 52
LCOM . . . 53
LNAM . . . 53
LOHC . . . 54
LOHI . . . 55
MATR . . . 56
MDAF . . . 56
MDAP . . . 57
MDAS . . . 57
MNAM . . . 57

PARA . . . 58
PERI . . . 58
RCMS . . . 59
RNAM . . . 59
RSEL . . . 60
SCAL . . . 61
SEAS . . . 62
SOLM . . . 62
SOLT . . . 63
SOLV . . . 63
TILO . . . 64
TOUT . . . 64
WIND . . . 65

A Differences between Sestra 10.12.0 and Sestra 8.8-2 . . . 66
A.1 Differences due to time-dependent boundary conditions . . . 66
A.2 Difference in printed RBLODCMB cards on the Results Interface File . . . 66
A.3 Differences in handling of non-structural elements . . . 67
A.4 Other differences . . . 67

B Theory . . . 69
B.1 Direct time integration in Sestra 10.12.0 . . . 69
B.1.1 Evaluation of loads . . . 70
B.1.2 Evaluation of the results . . . 70
B.1.3 Adaptive time stepping . . . 71
B.1.4 Steady-state detection . . . 72
B.1.5 Prescribed displacements, velocities, and accelerations . . . 72
B.2 Forced frequency response in Sestra 10.12.0 . . . 73
B.2.1 Equations of motion in the frequency plane . . . 73
B.2.2 Modal superposition . . . 74
B.3 Reaction forces . . . 74
B.3.1 Reaction forces from support constraints . . . 74
B.3.2 Reaction forces from spring support constraints . . . 75
B.3.3 Internal constraints . . . 75
B.3.4 Sum of reaction forces . . . 75
B.4 Non-structural elements . . . 75
B.5 Static condensation in dynamic analysis . . . 76
B.6 Component mode synthesis . . . 77
B.7 Modal load factors (modal participation factors) . . . 77
B.8 Tension-compression analysis (convex quadratic optimization) . . . 78

C The `THTS` and `THQS` finite elements . . . 79

1 Introduction

1.1 The Sestra 10.12.0 release

Sestra 10 features new core implementations of several standard Sesam finite element analysis types. The new implementations offer significantly improved computational performance compared to Sestra 8.8-2. The supported analysis types are

- linear static analysis
- time-domain dynamic analysis
  - direct time integration
  - modal superposition
- frequency-domain dynamic analysis
  - direct calculation
  - modal superposition
- free vibration analysis
- linearized buckling analysis
- tension-compression analysis
- linear static analysis with nonlinear pile-soil interaction (using Splice)

Analysis including multiple superelements is supported.

Sestra 10.12.0 poses some limitations on element types, materials, and certain other features. For full support of all standard Sesam finite element analyses and features, Sestra 8.8-2 is included with the Sestra 10.12.0 installation. Important limitations is listed in the release notes. Note that if analysis is attempted for a model which is not supported by Sestra 10.12.0, the Sestra 10.12.0 log file contains an error message and the analysis must be run by manually selecting Sestra 8.8-2.

This manual describes in detail only the analyses and features directly supported by Sestra 10.12.0. See the Sestra 8.8-2 user manual [1] for a full description of analyses and features available in Sestra.

1.2 Sestra: The Sesam finite element solver

Sestra is the program for linear static and dynamic structural analysis within the Sesam system. It is based on the displacement formulation of the finite element method.

Sestra interfaces with other program modules of Sesam for

- finite element model generation (performed by the preprocessors);
- load calculation (performed by the hydrodynamic analysis programs);
- result evaluation and presentation (performed by the postprocessors).

The Sesam Interface File system provides for data transfer between the various program modules of Sesam.

The analysis capabilities of Sestra comprise

- static analysis,
- free-vibration (eigenvalue) analysis,
- frequency-domain dynamic forced-response analysis (direct calculation and modal superposition),
- time-domain dynamic forced response analysis (direct calculation and modal superposition),
- analysis of axisymmetric structures,
- linearized buckling analysis,
- static and dynamic analyses accounting for stress stiffening effects,
- inertia relief,
- tension-compression analysis,

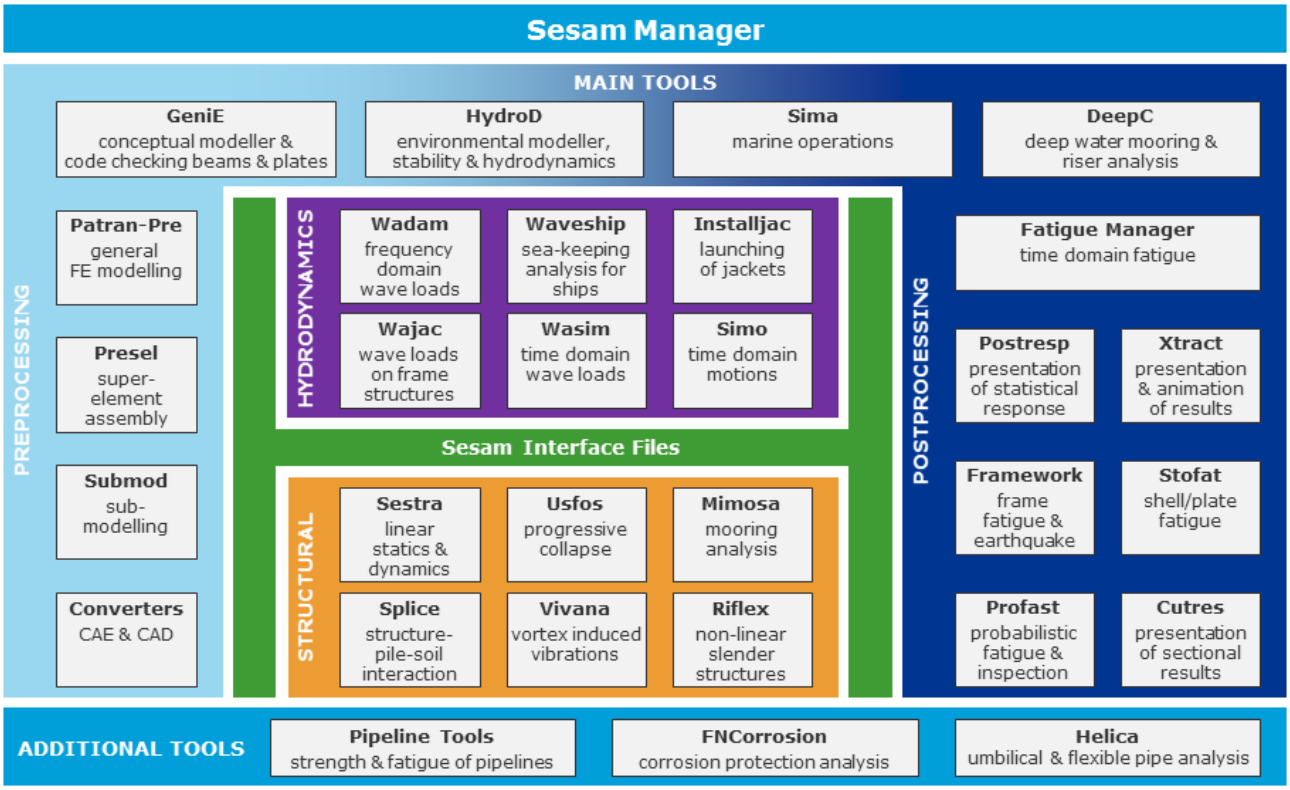


Figure 1.1: An overview of the Sesam system.

- gap-contact analysis.

Elements are available for

- trusses,
- beams,
- shells,
- membranes,
- axisymmetric solids,
- three-dimensional solids,
- transition elements between shells and solids.

All analyses reduce to a single or a series of symmetric sparse linear systems of equations. To solve these systems, Sestra uses a parallel direct sparse solver based on the LDL^{T} factorization.

1.3 Sestra in the Sesam system

Sesam is comprised of preprocessors, environmental analysis programs, structural programs and postprocessors. An overview of the Sesam system is shown in Figure 1.1.

1.4 How to read the manual

Chapter 1 gives a short introduction to Sestra and the Sesam system.

Chapter 2 summarizes the most important analysis capabilities and features of Sestra 10.12.0. Note that analyses and features available in Sestra 8.8-2 that are not available in Sestra 10.12.0 are not discussed.

Chapter 3 gives an in-depth user guide to the analyses available in Sestra 10.12.0. Examples are included for typical user scenarios.

Chapter 4 provides a comprehensive list of all analysis control commands available for Sestra 10.12.0.

Appendix A lists important differences between Sestra 10.12.0 and Sestra 8.8-2.

Appendix B provides theoretical background for features specific to Sestra 10.12.0.

1.5 System requirements

Supported operating systems:

- Windows 7 64-bit
- Windows 8.1
- Windows 10

1.6 Installation

Install Sestra 10.12.0 by running setup.exe. After installation, Sestra 10.12.0 will be recognized by the Sesam Application Version Manager (AVM) program, where it can be selected as the default Sestra version.

Note that Sestra 8.8-2 is included with the Sestra 10.12.0 installation. Sestra 8.8-2 can be found within the Sestra 10.12.0 installation folder, and is not installed as a separate application. It will thus not appear in the list of installed Windows applications.

The Sestra 10.12.0 installation includes Sestra 8.8-2 also in the case that Sestra 8.8-2 is already installed on the system. In this case, Sestra 8.8-2 can be uninstalled.

1.7 Public API

Sestra can be used programmatically through the Sestra public API. The API is written in managed c++ and can be invoked within the Microsoft .NET framework. The relevant header files can be found in the API subfolder of the Sestra installation. The header files contain the API documentation. The relevant binaries are located in the bin subfolder of the Sestra installation.

Analysis including multiple superelements is only supported through the Sestra executable application, and is not supported through the public API.

2 Features of Sestra 10.12.0

In this chapter, the analysis capabilities and features of Sestra 10.12.0 are discussed. Sestra 8.8-2 supports additional analyses and features; for a discussion of these features the reader is referred to [1].

2.1 Supported analyses

2.1.1 Linear static analysis

Linear static analysis using a displacement-based nodal finite element method results in a linear system of equilibrium equations

$$Kx = R, \tag{2.1}$$

where K is the global stiffness matrix obtained by summation of the element stiffness matrices, x is the matrix of unknown nodal displacements, and R is the load matrix obtained by summation of the node and element load matrices. The matrices x and R contain one column per load case.

Note: Sestra 10.12.0 features a single solver for the linear system (2.1). A direct and parallel sparse solver is used to form an LL^{T} or LDL^{T} factorization of the stiffness matrix.

2.1.2 Dynamic time-domain forced response analysis — direct calculation

The linear equation of dynamic equilibrium may be written as

$$M\ddot{x}(t) + C\dot{x}(t) + Kx(t) = R(t), \tag{2.2}$$

where at time t, $x(t)$ is the displacement vector, $\dot{x}(t)$ is the velocity vector, $\ddot{x}(t)$ is the acceleration vector, M is the mass matrix, C is the damping matrix, K is the stiffness matrix, and $R(t)$ is the load vector. In order to solve (2.2), appropriate initial conditions must also be provided.

Note that the damping term of (2.2) expresses viscous damping, i.e., damping forces proportional to and in phase with the structural velocities.

In a time-domain forced response analysis using direct time integration the solution $x(t)$ of (2.2) is approximated directly using a time-stepping algorithm. The following algorithms are available in Sestra 10.12.0:

- the collocation scheme of Hilber and Hughes (of order 3) [11];
- the Wilson Θ-method [12];
- the Newmark method (with the Trapezoidal rule as a special case) [9];
- the Generalized-α method [3].

The first three methods provide for backward compatibility with Sestra 8.8-2, while the Generalized-α method is a new feature in Sestra 10.12.0. Note that all methods are Newmark-like approximations [9]. The theory for the different algorithms is discussed in Appendix B.1.

The Generalized-α method is enabled through the GENA command. When this method is used, it is possible to enable adaptive time stepping so that the time-step size is chosen automatically by Sestra based on a user-specified tolerance for the local truncation error. See Chapter 3 for examples.

2.1.3 Dynamic time-domain forced response analysis — modal superposition

In a modal superposition analysis, the solution to (2.2) is approximated by a linear combination of the S eigenmodes associated with the S smallest eigenvalues. In other words,

$$x(t) = \sum_{i=1}^{S} v_i y_i(t), \tag{2.3}$$

where the y_i are time-dependent scalar-valued functions and the v_i are the eigenvectors of the solution pairs (λ_i, v_i) of the generalized eigenproblem

$$Kv_i = \lambda_i M v_i, \qquad v_i^{\mathrm{T}} M v_i = 1, \tag{2.4}$$

sorted according to increasing eigenvalues λ_i.

With $V = (v_1, \ldots, v_S) \in \mathbb{R}^{N\times S}$ and $y(t) = (y_1(t), \ldots, y_S(t))^{\mathrm{T}} \in \mathbb{R}^S$ we can write $x(t) = Vy(t)$; with $\Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_S)$ and $r(t) = V^{\mathrm{T}}R(r)$ it is then readily seen that

$$I\ddot{y}(t) + V^{\mathrm{T}}CV\dot{y}(t) + \Lambda y(t) = r(t). \tag{2.5}$$

The damping matrix C accounts for proportional damping (Reyleigh damping) and modal damping (see Section 2.8.4). It can in this case be written as

$$C = \underbrace{\alpha_1 M + \alpha_2 K}_{\text{Reyleigh}} + C_{\text{mod}}, \tag{2.6}$$

where C_{mod} accounts for modal damping. Note that C_{mod} is never formed,[1] but is defined by the relation $V^{\mathrm{T}}C_{\text{mod}}V = 2\sqrt{\Lambda}\mathrm{diag}(f_1, \ldots, f_S)$ such that the factors f_i specify the fraction of critical damping for each mode directly.

This results in a fully decoupled system of equations

$$\ddot{y}_i(t) + (\alpha_1 + \alpha_2\lambda_i + 2\sqrt{\lambda_i}f_i)\dot{y}_i(t) + \lambda y_i(t) = r_i(t), \qquad 1 \le i \le S. \tag{2.7}$$

Each of the S equations of (2.7) is integrated in time using a second-order implicit numerical integration schema.

The accuracy of the mode superposition method can be improved by adding a static contribution to (2.3). To present the update we introduce the static solution

$$Kx_s(t) = R(t) \tag{2.8}$$

and the static modal solution

$$y_{s,i} = v_i^{\mathrm{T}}R(t)/\lambda_i \tag{2.9}$$

Using this the modal approximation is

$$x(t) = x_s(t) + \sum_{i=1}^{S} v_i\big(y_i(t) - y_{s,i}\big). \tag{2.10}$$

2.1.4 Dynamic frequency-domain forced response analysis — direct calculation

In a dynamic frequency-domain forced-response analysis the load (and hence response) is assumed to be harmonic, and a separate system of equations is obtained for each considered load frequency. When applicable this approach is efficient as no integration in time is necessary. Only the steady-state response may be found.

In addition to proportional damping (viscous damping), structural damping may also be considered in the frequency domain. Structural damping is proportional to the displacements and in phase with the velocities.

The complex system of equations solved for each load frequency ω is

$$(-\omega^2 M + \omega\mathrm{i}C + K)x_a = R_a, \tag{2.11}$$

where M, C, and K are the mass-, damping,- and stiffness matrices, respectively, x_a is the complex solution (amplitude and phase) vector, and R_a is the complex load vector.

For a theoretical assessment of the approach, see B.2.

2.1.5 Dynamic frequency-domain forced response analysis — modal superposition

In a modal frequency-domain forced-response analysis the solution is, as in Section 2.1.3, assumed to be a linear combination of M mode shapes. In this case the resulting complex system of equations can be written

$$V^{\mathrm{T}}(-\omega^2 M + \omega\mathrm{i}C + K)Vy = V^{\mathrm{T}}R_a. \tag{2.12}$$

[1] If formed, the matrix C_{mod} could have been realized as $C_{\mathrm{mod}} = 2MV\sqrt{\Lambda}FV^{\mathrm{T}}M$.

Again, due to the orthonormality $V^{\mathrm{T}}MV = I$ of the mode shapes and the admissible forms of damping, the equations of (B.42) are independent such that (B.42) is equivalent to the M decoupled complex equations

$$(-\omega^2 + \omega \mathrm{i} c_i + \lambda_i) y_i = r_i, \qquad 1 \leq i \leq S. \tag{2.13}$$

Typically, S is chosen much smaller than N (the total number of degrees of freedom in the system), and thus the modal analysis can significantly reduce computational cost compared to a direct approach.

For additional details, see B.2.

2.1.6 Free vibration analysis

In a free vibration analysis, Sestra computes eigenvalues λ_i and eigenvectors v_i satisfying

$$Kv_i = \lambda_i M v_i, \quad 1 \leq i \leq S \tag{2.14}$$

Here K is the stiffness matrix, M is the mass matrix, and S is the (user-) specified number of eigenvalues and vectors to be calculated. The eigenvalues are non negative, if both the stiffness and mass matrices are nonsingular they are positive.

The square roots of the eigenvalues are the natural angular frequencies of the construction, the natural frequencies are $\sqrt{\lambda_i}/2\pi$ while the eigenvectors are the normal modes or eigenmodes.

Typically, the S smallest eigenfrequencies are of interest. In some cases the frequencies close to a given frequency is of interest.

A shift and invert procedure is applied. Her a positive number σ is selected and the eigenvalues and eigenvectors satisfying

$$(K - \sigma M)^{-1} M v_i = \mu_i v_i \tag{2.15}$$

are computed. The eigenvalues of interest, i.e. λ_k can easily be computed from μ_k. Note that as long as $K - \sigma M$ is nonsingular, one or both of the stiffness and mass matrices may be singular. Furthermore, if σ is equal to an eigenvalue, the system is singular. Thus the shift should not be too close to an eigenvalue.

If the stiffness matrix is nonsingular the shift σ can be taken to be zero. If the stiffness matrix is singular, a positive shift must be selected. Any positive number smaller that the lowest eigenfrequency will do. If the eigenfrequencies close to a given frequency is of interest, choose σ equal to it.

Sestra can compute all the eigenvalues in a specified interval.

The algorithm employed in Sestra 10.12.0 is an implicitly restarted Lanczos eigensolver.

2.1.7 Linearized buckling analyis

Linearized buckling analysis consists of two steps. First a load is selected and a linear analysis is performed. Based on this result a geometric stiffness matrix K_g is computed and a few of the eigenpairs of the the eigenvalue problem

$$Kv = \lambda K_g v \tag{2.16}$$

is computed. The smallest eigenvalue is the stability factor, i.e. the stability factor times the buckling load is the critical load for which buckling occurs. The eigenvectors represent buckling modes and it may be of interest to compute several of the buckling modes in order to gain insight into the behaviour of the construction.

The eigenvalue problem is reformulated using the shift-invert formulation

$$(K - \sigma K_g)^{-1} K v = \mu v \tag{2.17}$$

where the shift σ is a positive number. Note that a nonzero shift is required in this formulation. The eigenvalue λ is related to μ and σ as follows

$$\lambda = \frac{\mu\sigma}{\mu - 1} \tag{2.18}$$

The eigenpairs are computed using the same implicitly restarted Lanczos algorithm used for the free vibration analysis.

2.1.8 Tension-compression analysis

A tension-compression analysis is a particular type of non-linear static analysis, where truss elements may be defined to be *tension only* or *compression only*. There are therefore material non-linearities because the stiffness properties of the trusses (in this case “on” or “off”) depends on whether a truss is in tension or in compression.

In Sestra 10.12.0, a tension-compresion problem is reformulated internally as a contact problem: the original equation (2.1) is complemented by inequality constraints of the form $Ax \geq b$. The static problem (2.1) can be reformulated as a minimization problem: find $x = \arg\min \frac{1}{2}x^T Kx - x^T R$. For a tension-compression problem, the minimizer is subjected to the additional inequalities:

$$x = \arg\min \frac{1}{2}x^T Kx - x^T R \tag{2.19}$$

$$\text{s.t.} \quad Ax \geq b. \tag{2.20}$$

The algorithm employed in Sestra 10.12.0 for solving this problem is a general interior-point method for convex quadratic optimization (see e.g. [10]).

2.1.9 Linear static analysis with nonlinear pile-soil interaction

Sestra may be used to statically condense the stiffness matrix and load vectors to a set of defined supernodes (Schur-complement calculation). This functionality is currently used only from the external program Splice (version 7.5 and newer) in the context of nonlinear pile-soil interaction. In this case, the data transfer between Splice and Sestra is programmatic through a private API (no file transfer).

2.1.10 Postprocessing

In postprocessing, element forces and stresses may be calculated based on a Results Interface File containing nodal displacements. A typical workflow is as follows.

1. Perform a Sestra analysis, but do not calculate element forces and stresses.
2. Perform postprocessing with the Results Interface File generated in step 1 as input.

Postprocessing may be applied to analysis results from any analysis type performed in step 1.

For an example, see Section 3.9.

2.1.11 Stress stiffening

Initial stress or structural loading will influence the structural stiffness in a way which is not accounted for by the ordinary linear stiffness matrix and which in some cases may be important. For example:

- The centrifugal forces on a rapidly rotating body will significantly influence the eigenfrequencies of the system.
- The effect of vertical loading from deck structures and other superstructures on offshore frame structures (jackets, jackups and guyed towers) may be of significance to the dynamic behaviour of the system.

In Sestra this stress stiffening effect may be accounted for in analysis of beam and shell structures by the following two step procedure:

1. A static analysis with the ordinary linear stiffness matrix and the load causing the initial stress level is performed. The stresses and corresponding geometric stiffness (or initial stress) matrices of the basic elements are calculated.
2. The geometric stiffness matrices are added to the ordinary stiffness matrices of the basic elements and an updated stiffness matrix for the model is achieved. Based on the updated stiffness matrix a new analysis is performed. This may be a linear static, tension-compression, free vibration or forced dynamic analysis. Stress stiffening may also be used with non-linear pile-soil interaction (using Splice).

When a stress stiffening analysis is requested Sestra performs this two-step procedure automatically.

It should be noted that the initial stresses are not added to the stresses in the final analysis.

When stress stiffening is used with non-linear pile-soil interaction (using Splice) the following must be noted.

- For standard non-linear pile-soil interaction analysis all Schur complement nodes (in practice pile nodes and pile-head nodes) are fixated in the generation of the geometric stiffness matrix.
- When stress stiffening is used in the dynamic analysis for generation of equivalent static loads (performed using the ESLE command) the geometric stiffness calculated in this step is automatically transferred to the subsequent static analysis (performed using the ESLI command). Thus, the boundary conditions applied in the dynamic step automatically applies to the geometric stiffness used in the static step (such as linearized spring pile foundations).

2.2 Superelement analysis

Analysis including multiple and multilevel superelements is supported using an automated workflow comprising the following steps:

1. The hierarchical superelement model is read and merged to a single, global model; a new *temporary* Input Interface File (.FEM) is generated.
2. The new global model is subject to a standard Sestra analysis; a *temporary* global Results Interface File is produced.
3. The results on the temporary global Results Interface File are distributed to the original superelements; the final Results Interface File(s) (binary .SIN or formatted .SIF) are produced. Note that if binary format is requested, the utility program Prepost is automatically run to convert temporary formatted files to binary format.

With this automated workflow the superelement technique is supported as a *modelling technique* offering the user an efficient and logical way of modelling different parts of a larger structure. Repeated superelements may also be used.

Note: The superelement technique is not supported as a *solution algorithm* (often referred to in literature as sub-structuring) in Sestra 10.12.0. Consequently, a superelement analysis in Sestra 10.12.0 does not imply a reduction of internal degrees of freedom to the supernodes. As a result, there is no reason from a computational (i.e. time-saving) point of view to model geometrically identical parts as a single superelement for repeated assembly in the global model.

Further, it is important to note that for a dynamic analysis, all degrees of freedom in the model are subjected to dynamic effects. Sestra 10.12.0 supports static condensation, also called master-slave reduction, and component mode synthesis in dynamic analyses. To active this use DRED command, see Sections 2.3 or 2.4. To select component mode synthesis the RCMS command must also be given.

The superelement capabilities of Sestra 10.12.0 poses some limitations on the finite element model:

- All degrees of freedom of a supernode must be defined as super. If this is not the case, Sestra 8.8-2 must be used instead.
- A local coordinate system cannot be specified on a supernode. If this is required, Sestra 8.8-2 may be used instead.
- A node with fixations (or prescribed displacements) on only one or two translational degrees of freedom, and/or only one or two rotational degrees of freedom must either
  - have a local coordinate system, or
  - have the fixations specified on the top-level superelement, or
  - reside within a superelement which has a trivial (translation-only) transformation with respect to the top-level superelement.
- A node which is the dependent or independent node of a linear dependency (rigid link) must either
  - have a local coordinate system, or
  - have the linear dependency specified on the top-level superelement, or
  - reside within a superelement which has a trivial (translation-only) transformation with respect to the top-level superelement.

- Hinges (on two-noded beam elements) must be defined with respect to either
  - a specified local coordinate system, or
  - the element-local coordinate system

  unless the hinge resides within a superelement which has a trivial (translation-only) transformation with respect to the top-level superelement.

2.3 Static condensation in dynamic analysis

Static condensation is used to reduce the size of the dynamical system. The reduced system approximate the given system, i.e. the solution vectors of the reduced system approximate the solution of the full system. The choice of the degrees of freedom in the reduced system is crucial for the accuracy of the method.

Static condensation is often used in free vibration analysis to suppress local, low frequency effects, e.g. local, low frequency normal modes in a free vibration analysis. These modes can often be suppressed if the rotational degrees of freedom are constrained to be in static equilibrium.

To do static condensation the model must have super degrees of freedom, i.e. some BNBCD cards with boundary code 4 must be in the Input Interface file <prefix>T<n>.FEM. These will be used as dynamic degrees of freedom, while the remaining will satisfy static equilibrium. In addition the DRED command must be in the sestra input file, otherwise the super degrees of freedom is ignored and a direct analysis is performed. For further details related to static condensation see Appendix B.5.

Static condensation can be used together with time domain, frequency domain and free vibration analysis.

Static condensation can also be used in combination with non-structural elements, see Section 2.8.1. In this case some super degrees of freedom must be specified on nodes connected to both non-structural and regular elements in order to add non-structural load, mass and damping to the reduced system. Also, the non-structural part of the model must be properly fixed to avoid singular sub matrices. If the super degrees of freedom connected to non-structural nodes is not sufficient to avoid rigid body motion, additional boundary conditions must be specified.

2.4 Component mode synthesis reduction

Component mode synthesis reduction is, as static condensation, used to reduce the number of degrees of freedom of the dynamical system. In this case a number eigenmodes are used to approximate the non-super degrees of freedom. The solution of the reduced system is an approximation of the solution of the full system.

To use component mode synthesis the model must have super degrees of freedom, see section 2.3, however the location of these degrees or freedom are not as crucial as they are for static condensation in dynamic analysis. Also, the number of super degrees of freedom can usually be reduced considerably compared to the static condensation reduction.

Component mode synthesis requires the computation of the eigenmodes and eigenfrequencies related to the non-super degrees of freedom. On the other hand, computing the reduced matrices is more efficient and the storage of the matrices requires less memory compared to the static condensation reduction. Note that the input to the eigenvalue solver is taken from the RCMS command. For further information see B.6.

Component mode synthesis can be used together with free vibration analsysis, and both time domain and frequency domain analysis. It can also be combined with modal analysis. Furthermore, it can also be used in combination with nonstructural elements, see Sections 2.3 and 2.8.1.

2.5 Features for dynamic spectral fatigue analysis

A dynamic spectral fatigue analysis typically involves

1. time-domain forced response (direct time integration) analysis in Sestra to determine the steady-state (periodic) response of a structure subjected to periodic loading conditions;
2. fatigue calculation in Framework based on the steady-state response.

Sestra 10.12.0 has several new features developed specifically to facilitate spectral fatigue analyses. These new features are discussed below.

2.5.1 Specification of periodic loading conditions

In the case of a periodic loading condition, Sestra 10.12.0 allows the user to specify only a single cycle of the load. Sestra then repeats this loading condition any given number of times. This automated procedure has several advantages, including

- significantly reduced computational time for load generation;
- significantly reduced size of load files;
- simplified load specification on the Sestra input file.

See the PERI command for details on how to specify a periodic loading condition.

2.5.2 Steady-state detection

In the context of spectral fatigue analysis, the steady-state (periodic) response of the structure is of interest and hence the analysis must be performed for a potentially large number of load cycles. However, the number of required load cycles may be difficult to know *a priori*.

Sestra 10.12.0 can automatically detect when a structure subjected to a periodic loading condition has reached its steady-state (periodic) response, and thus terminate the analysis. See the PERI command for details on how to enable steady-state detection.

2.5.3 Calculation of base shear and overturning moment

Two post-processing capabilities intended for spectral fatigue analysis are available as integrated features in Sestra 10.12.0.

- **Calculation of reaction-force sums (base shear and overturning moment).** Calculation at each timestep of sums of reaction forces over a specified set of nodes. The sums are calculated separately for each of the translational and rotational degrees of freedom.

  In the case that all nodes with support (i.e. with prescribed degrees of freedom) are selected, these reaction force sums correspond to the base shear (for translation) and overturning moment (for rotation) of the structure.

- **Calculation of steady-state response-amplitude operators (RAOs).** Automatic calculation of the steady-state response (the amplitudes of the sums of reaction-forces) as functions of wave period. This is relevant
  - in a time-domain analysis when the structure is subjected to periodic loading;
  - in a frequency-domain analysis.

A more detailed description of these features can be found in Section 3.3.4 (for time-domain analysis) and in Section 3.5.3 (for frequency-domain analysis). Theoretical background is included in Appendix B.3.

2.6 Generation and use of equivalent static loads

Equivalent static loads can be used to include the effects of inertia and damping in a static analysis. Recalling the equations of dynamic equilibrium

$$M\ddot{x}(t) + C\dot{x}(t) + Kx(t) = R(t), \tag{2.21}$$

the equivalent static loads at time $\tilde{t}$ are defined as

$$R_{\mathrm{esl}}(\tilde{t}) = R(\tilde{t}) - C\dot{x}(\tilde{t}) - M\ddot{x}(\tilde{t}). \tag{2.22}$$

If we assume that $\dot{x}(\tilde{t})$ and $\ddot{x}(\tilde{t})$ are known, then we may consider for $t = \tilde{t}$ an equivalent static problem to (2.21) given as

$$K\tilde{x} = R_{\mathrm{esl}}(\tilde{t}). \tag{2.23}$$

It is readily seen that the solution $\tilde{x}$ to the static problem (2.23) is equivalent to the solution $x(\tilde{t})$ of the dynamic problem (2.21). Clearly if $\dot{x}(\tilde{t})$ and $\ddot{x}(\tilde{t})$ are known then so is $x(\tilde{t})$ and the calculation (2.23) is excessive. In practice, equivalent static loads are used when (2.23) is replaced by a slightly perturbed problem.

As an example, equivalent static loads can be used to include dynamic effects in an analysis of a jacket including non-linear pile-soil interaction (modelled as non-linear springs). In this case (2.23) is replaced by a non-linear static problem; the solution $\tilde{x}$ thus includes both the effects of the non-linear springs coupling the soil with the piles as well as dynamic effects in the main structure through the inertia and damping terms of the equivalent static loads (2.22).

Such analysis is different from a fully coupled non-linear dynamic pile-soil analysis in that the non-linear behavior is not accounted for in calculation of the inertia and damping terms.

The generation (export) of the equivalent static loads is controlled with the ESLE command; the use (import) of the equivalent static loads is controlled with the ESLI command. It is critical that no modification is made to the finite element mesh (element and node numbering and connectivity) in the Input Interface File between these two analyses.

When stress stiffening is used in the generation of equivalent static loads the geometric stiffness calculated in this step is automatically transferred to the subsequent static analysis.

2.7 Matrix output

Reduced matrices may be exported to a Matrix Interface File (M-file). The Matrix Interface File contains in this case data for reduced stiffness, mass and damping matrices and reduced load vectors. The matrices and vectors are reduced to any super degrees of freedom present in the model.

Matrix data related to generalized degrees of freedom from Component Mode Synthesis reduction is also stored if requested (see Section 2.4).

Matrix output is controlled with the MATR command, and may be combined with Component Mode Synthesis using the RCMS command. A Results Interface File will be written by default (but can be switched off using the RNAM command). It will contain internal mode shapes as calculated by the Component Mode Synthesis method.

Matrix output is handled internally as a separate analysis type, and may not be combined with other analyses.

2.8 Model properties

2.8.1 Stiffness properties

The stiffness matrix of the model is obtained by adding the contributions from the element stiffness matrices. The matrix is singular and a static solution is impossible unless the model is constrained against rigid-body motion. In a dynamic analysis, however, rigid-body motion is in certain cases allowed.

For certain shell and membrane elements[2] singularity problems arise when all elements connected to a node are co-planar (lie in a single plane). This is because these shell elements have no stiffness associated with the rotation about the axis perpendicular to the element plane. To avoid these singularities small artificial stiffness contributions are inserted by Sestra in the element stiffness matrices. See also [1, Section B6.6] and the description of the CMAS command.

Non-structural elements

Any element may be specified as non-structural. Non-structural elements transfer load and mass to the structure, but do not contribute to the structural stiffness. Non-structural elements may for instance be used to model the conductors of a jacket platform.

Sestra 10.12.0 allows modelling of multiple interconnected non-structural elements — *a non-structural substructure* — without introducing singularities. Loads and mass (and damping when applicable) are condensed to the nodes connecting the non-structural substructure to regular elements.

Note that the one-noded mass element (MGMASS) is always interpreted as being non-structural. Such elements may thus be part of a non-structural substructure without being explicitly specified as a non-structural element.

For the two-noded non-structural beam element (BEAS), it is possible to neglect mass connected to the longitudinal degree of freedom (element-local x-direction).

The method used to handle non-structural elements is described in B.4.

[2] The shell elements FTRS, FQUS, SCTS, SCQS, THTS, THQS; the membrane elements CSTA, ILST, IQQE, LQUA.

For dynamic analysis non-structural elements can be used in combination with static condensation. In this case loads and mass (and damping when applicable) are condensed to any super degrees of freedom on the nodes connecting the non-structural substructure to regular elements. See also the information box below.

Note:

- Sestra 10.12.0 allows an element of any type to be defined as non-structural.
- Sestra 10.12.0 allows modelling of multiple interconnected non-structural elements without using the superelement modelling technique. A node on a non-structural element not connected to another (structural or non-structural) element need not be fixed to ground.

  In contrast, in Sestra 8.8-2, all degrees of freedom for all non-structural elements must either be connected to other (structural) parts of the structure, or be fixed to ground.

  Alternatively, the superelement modelling technique may be used to define whole superelements as non-structural. This feature is supported in Sestra 8.8-2 as well as in Sestra 10.12.0.
- Sestra 10.12.0 does not calculate displacements, stresses, and forces in non-structural elements. Displacements are set to zero for nodes internal to a non-structural part of the structure. Stresses and forces are not calculated for non-structural elements.
- If a non-structural substructure (multiple interconnected non-structural elements) is defined in a dynamic analysis using static condensation (using the DRED command) there must be super degrees of freedom on the nodes connecting this substructure to regular elements.

  Interior nodes in a non-structural substructure cannot be defined as super as this will result in a singular stiffness matrix.

  It is not required to specify super degrees of freedom on the nodes connecting singleton non-structural elements to regular elements.
- In a tension-compression analysis, any node on a non-structural element cannot be connected only to tension-only or compression-only elements.

2.8.2 Mass properties

The mass matrix is generated by Sestra based on the volume and density of the elements. Mass may also be explicitly defined in terms of nodal masses.

Note that Sestra 10.12.0 supports consistent (full) element mass matrices only.

For certain shell and membrane elements singularity problems arise when all elements connected to a node are co-planar. This problem is similar to the one for the stiffness matrix. A small mass is therefore inserted by Sestra for the massless degrees of freedom to prevent a singularity. The magnitude of this mass can be specified on the CMAS command.

2.8.3 Hydrodynamic added mass for big volume structures

The eigenfrequencies for a structure in water are different from the eigenfrequences for the same structure when vibrating in air. This is due to the contribution of the water causing an added mass effect. This effect is important when analysing floating structures (ships, offshore platforms, etc.) excited to vibrate in their flexural modes (by disturbances from engine, propeller, wave impact, etc.).

Such hydrodynamic added mass may be computed by a higher-order boundary integral equation method (BIEM) in Sestra. The method requires pressure (added mass) nodes on the submerged surface of the structure. The fluid-structure coupling is accounted for by matching the normal surface velocity defined in the elastic equilibrium relationship to that defined by the governing equations of the fluid field. In the coupling of the finite element (structure) and boundary integral (fluid) formulations shape functions approximating the fluid field and the normal surface velocity are used in a manner consistent with the finite element discretisation.

The method rely on the following assumptions.

- The effect of surface waves can be neglected.

- The structure surface is smooth (sharp edges may give local disturbances).
- Forward speed is zero.
- The fluid is incompressible (compressibility effects are neglected).

For details of the boundary integral method used in Sestra, the reader is referred to [2].

The implementation in Sestra 10.12.0 requires that the union of the wet surface and the (extension of the) water level form a closed volume. Submerged elements must be properly tagged with definition of inside and outside. This may be handled by the preprocessor, e.g. GeniE.

Note:

- Sestra 10.12.0 does not require a separate superelement for specification of hydrodynamic added mass. Note that this is different from previous versions of Sestra.
- Separate membrane elements (3-noded and 4-noded) are no longer used.
- Hydrodynamic added mass can be applied only to 3-noded and 4-noded (i.e. first-order) shell elements.
- To invoke hydrodynamic added mass in Sestra 10.12.0, submerged elements must be tagged using hydrodynamic dummy hydro pressure (BEUSLO) cards. The cards provide Sestra with definition of the element inside and outisde.
- Hydrodynamic added mass is invoked with the HYDM command. This command is no longer preceeded by an IDTY (superelement type) command.
- For reanalysis of existing models which already have a hydrodynamic added mass superelement:
  - If the model has dummy hydro pressure defined for all (and only all) submerged/wet elements, the model can be analysed in Sestra 10.12.0 after simply removing the hydrodynamic added mass superelement.
  - If the model does not have dummy hydro pressure defined for submerged/wet elements, this must be defined (using GeniE) and the hydrodynamic added mass superelement must be removed before analysis.

2.8.4 Damping properties

Proportional damping

Proportional damping (Rayleigh damping) is relevant for dynamic analyses in time domain and in frequency domain.

Proportional damping is a viscous damping where a damping matrix C is specified as a linear combination of the stiffness- and mass matrices in the dynamic equation (2.2). That is to say,

$$M\ddot{x}(t) + C\dot{x}(t) + Kx(t) = R(t), \tag{2.24}$$

where $C = \alpha_1 M + \alpha_2 K$. The coefficients α_1 and α_2 are specified on the input command MDAP.

Note that if the modal damping ratios λ_j, λ_k associated with two eigenfrequencies ω_j, ω_k are known, then the proportional damping factors may be calculated as

$$\alpha_1 = 2\omega_j\omega_k \frac{\lambda_j\omega_k - \lambda_k\omega_j}{\omega_k^2 - \omega_j^2}, \qquad \alpha_2 = 2\frac{\lambda_k\omega_k - \lambda_j\omega_j}{\omega_k^2 - \omega_j^2}. \tag{2.25}$$

A derivation of this formula can be found in [1, Appendix B].

Modal damping

Modal damping is relevant in a time-domain or frequency-domain modal superposition analysis. In the time-domain case, the equations to integrate in time reduce to single-degree-of-freedom expressions of the form

$$\ddot{y}_i(t) + c_i\dot{y}_i(t) + \lambda_i y_i(t) = r_i(t), \tag{2.26}$$

for each eigenmode. Here, λ_i is the eigenvalue associatied with eigenmode i, and c_i is the specified damping coefficient.

Critical damping is achieved when $c_i = 2\sqrt{\lambda_i}$. In Sestra, (2.26) is instead formulated as

$$\ddot{y}_i(t) + 2f_i\sqrt{\lambda_i}\dot{y}_i(t) + \lambda_i y_i(t) = r_i(t), \tag{2.27}$$

such that the user may specify f_i directly as the desired fraction of critical damping for each mode.

In the frequency-domain case, the amount of modal damping is specified as in (2.27); these equations are then subsequently transformed to the frequency domain.

Modal damping is specified on the MDAF command.

Structural damping

Structural damping is relevant for dynamic analyses in the frequency domain.

Structural damping is proportional to the displacements and in phase with the velocities. In the case of a frequency-domain analysis, the equations of motions are extended to the complex plane as

$$(K + \mathrm{i}C_s)x + C\dot{x} + M\ddot{x} = R, \tag{2.28}$$

where C_s is the structural damping matrix and C is the viscous (typically proportional) damping matrix.

The program input is the structural damping factor η defined by $C_s = \eta K$.

This factor is specified on the MDAS command. In the case of a frequency-domain modal superposition analysis, η can be different for different (ranges of) mode shapes.

Dashpots and axial dampers

Damping may also be introduced explicitly using damping elements. The available damping elements are listed in Section 2.9.1. Note that damping elements must be enabled via the DYMA command on the Sestra input file in order to be included in the analysis.

2.8.5 Load types

Loads may be given as

- nodal loads (including moments),
- line loads,
- surface pressure,
- surface load in component form,
- temperature loads / initial strains, supported for all elements except first- and second-order solids,
- gravity,
- general inertia load,
- rigid body acceleration.

See [1, Appendix C] for a detailed description of the applicability of the different loads to the different element types.

2.8.6 Boundary conditions

Sestra 10.12.0 supports the following boundary conditions.

- prescribed displacement,
- prescribed acceleration (time-domain forced response only),
- elastic support (spring-to-ground element).

From Sestra 10.12 prescribed displacements and prescribed accelerations may be combined within the same analysis (load history) on different nodes. For details about the implementation of time-dependent boundary conditions, see Appendix B.1.5.

Group	Name	Description	Number of nodes	d.o.f. per node
Spring	GSPR	Spring-to-ground	1	6
	GLSH	General spring	2	6
Mass	GMAS	General mass element	1	6
Damping	GLDA	General damping element	2	3 or 6
	AXDA	Axial damper	2	3
	GDAM	Damper to ground	1	3 or 6
Beam or truss	TESS	Truss	2	3
	BEAS	Straight beam	2	6
	BTSS	Curved beam	3	6
Membranes	CSTA	Plane constant strain triangle	3	3
	LQUA	Plane linear quadrilateral membrane element	4	3
	ILST	Plane linear strain triangle	6	3
	IQQE	Plane quadratic quadrilateral membrane element	8	3
Shell	FTRS	Triangular flat thin shell	3	6
	FQUS	Quadrilateral flat thin shell	4	6
	FTAS	Triangular flat thin shell with drilling degrees of freedom	3	6
	FQAS	Quadrilateral flat thin shell with drilling degrees of freedom	4	6
	SCTS	Subparametric curved triangular thick or thin shell	6	6
	SCQS	Subparametric curved quadrilateral thick or thin shell	8	6
	THTS	Triangular flat thick shell (see Appendix C)	3	6
	THQS	Quadrilateral flat thick shell (see Appendix C)	4	6
Solid	TETR	Tetrahedron (first-order)	4	3
	TPRI	Triangular prism (first-order)	6	3
	LHEX	Incompatible linear hexahedron (first-order)	8	3
	ITET	Isoparametric tetrahedron (second-order)	10	3
	IPRI	Isoparametric prism (second-order)	15	3
	IHEX	Isoparametric hexahedron (second-order)	20	3
Matrix	MATR	Matrix element	Any	3 or 6

Table 2.1: List of available elements

2.8.7 Initial conditions

For time-domain forced-response analyses Sestra 10.12.0 supports the following initial conditions.

- initial displacement,
- initial velocity.

The initial acceleration is calculated from the equations of motion (2.2) and the initial displacement and velocity.

If initial conditions are not explicitly specified, Sestra assumes a zero displacement and velocity.

Also, note that the specified initial conditions apply to the value of t (time) for which the initial load is specified.

2.8.8 Multipoint constraints (linear dependencies)

Arbitrary multipoint constraints may be specified. Any degree of freedom of any node may be dependent with user-defined weight factors on any degrees of freedom of any nodes.

2.9 Elements library and element properties

2.9.1 Element library

Table 2.1 lists the elements supported by Sestra 10.12.0. A detailed description of all elements except the THTS and THQS elements can be found in [1, Appendix C]. A detailed description of the THTS and THQS elements can be found in Appendix C.

The beam elements may have the following properties.

- eccentricities (2-noded beam and 3-noded beam)

- flexible hinges (2-noded beam only)

A word of caution related to the use of truss and membrane elements is appropriate. The truss element is one dimensional, i.e. only axial displacement and stresses are modelled and it has only one independent degree of freedom in each node. Consequently, modelling using several truss elements along one straight line or in one plane will result in a singular stiffness matrix.

The membrane elements are two dimensional, i.e. only displacements along the local x and y axes are modelled. As a consequence, the rows and columns in the element matrix corresponding to the z degrees of freedom are zero. In order to obtain a nonsingular stiffness matrix small numbers are added to the element matrix, for further detailse see [1]. Note that if several membrane elements form a planar patch, loads in the transverse direction can not be used since the corresponding displacements will become very large due to the small numbers added to the element matrices.

The matrix elements (MATR) are used to include user-defined stiffness and mass matrices and load vectors, for example as generated by external software. In addition to any number of regular nodes, such elements may contain any number of generalized nodes (nodes which are not present in the finite element model) with any number of generalized degrees of freedom.

2.9.2 Materials

Recall that the stress-strain relation my be written as

$$\sigma = D\epsilon, \tag{2.29}$$

where σ denotes the stresses, ϵ denotes the strains, and D denotes the elasticity matrix expressed in terms of the material properties

For all element types Sestra 10.12.0 supports isotropic materials, meaning that the stress-strain relation is direction invariant.

For shell and membrane elements anisotropic materials are also supported. In this case the material has different properties in different directions. However, an element-local orientation of the axes of anisotropy for a given material is not supported, and thus a different material must instead be used for each such orientation.

Note that all materials which may be generated using the current version of GeniE are supported in Sestra 10.12.0.

2.10 Input and output

2.10.1 Input

The following *input* files are relevant for Sestra 10.12.0.

Input Interface File. The model input is contained in the Input Interface File <prefix>T<n>.FEM or <prefix>T<n>.SIN produced by the preprocessor programs (e.g. GeniE). The Input Interface File may also contain loads. Note that <prefix> is specified on the INAM command on the analysis control data file.

Sestra will first look for a formatted file with the .FEM extension. If this file is not found, Sestra will attempt to load a direct access (binary) file with the .SIN extension. The latter is supported using Genie 7.9 or later.

Loads Interface File. Loads may be specified separately on a Loads Interface File <prefix>L<n>.FEM produced by the hydrodynamic analysis programs. This file may be in binary or text format. The loads on this file will be added to any loads present on the Input Interface File. Note that <prefix> is specified on the LNAM command on the analysis control data file.

Analysis control data file. Analysis type and all relevant analysis parameters are specified on the Sestra input file (sestra.inp). The file must be located in the Sestra working directory.

The complete command reference for Sestra 10.12.0 is found in Chapter 4.

Each command involves one or more 80-character records, and consists of the command type and the command parameters. The first four characters of the first record of the command identify the command type. On continuation records the identifier field is blank. As an example, an input command may be defined as follows.

1——5	6——-10	11——-15	16——-20	21——-25	26——-30	31——-35	36——-40
TYPE	FIELD1		FIELD3				

41——-45	46——-50	51————60	61————70	71————80
				FIELD12

The numbers above each field indicate the columns in which the values of that field must be entered. See Section 4.1 for a general description of the input commands.

Analysis control data file (extension). In certain situations additional control data produced by the hydrodynamic analysis programs is required. This file is named <prefix>S<n>.FEM, and is interpreted by Sestra as a literal extension of sestra.inp. Note that <prefix> is specified on the INAM command on the analysis control data file.

Note that <n> denotes the superelement number and is specified on the ITOP command on the analysis control data file.

2.10.2 Output

Note: From Sestra 10.7 the file names of the Message Log File and the Results Listing File have changed:

- The Message Log File (operational log) is now named Sestra.MLG (this file was named Sestra.Lis in Sestra 10.6 and earlier).
- The Results Listing File (summary results file) is now named Sestra.Lis (this file was named <prefix>_summary_results.txt in Sestra 10.6 and earlier).

The reason for this change is to align with existing conventions in Sesam.

The following *output* files are relevant for Sestra 10.12.0. (Note that <prefix> is specified on the RNAM command, <n> is specified on the ITOP command, and <l> is the load history number specified on the LOHI command.)

Results Interface File. Primary and derived results. The Results Interface File (<prefix>R<n>.SIF or <prefix>R<n>.SIN) is used for transfer of results to a program for result presentation or postprocessing (e.g. Xtract or Framework), or for long-term storage of results. The format (SIF or SIN, i.e. text or binary) of the result file is specified on the RNAM command.[3] The derived results can be computed and stored for the elements in selected sets only. For further information consult the ERST command.

Matrix Interface File. The Matrix Interface File is described in the SESAM Interface File Description, and is used for export of reduced matrices and load vectors using the MATR command. See Section 2.7.

Results Listing File (Sestra.LIS). A file with basic analysis results. The file includes

- reaction force sums, load sums, and load/force balances for each result case;
- mapping from load case numbers to result case numbers (note that even for relatively simple cases the result case number is not equal to the load case number, for example whenever the load cases are not specified as a continuous sequence starting from load case 1);
- mass sum and mass centre;
- eigenvalues.

Dynamic reaction forces. When calculation of dynamic nodal reaction forces is selected (on the DREA command) several CSV-formatted files with reaction forces are produced. The files have the name <prefix>_reactions_lohi<l>.csv, where <prefix> is the file-name prefix defined on the RNAM command and <l> is the id of the relevant load history (see the LOHI command).

Response-amplitude operators. When calculation of response-amplitude operators (RAO) is selected (using the DREA command) a CSV-formatted file with RAOs is produced. The file is named <prefix>_reactions_RAO.csv.

Message Log File (Sestra.MLG). Upon execution of Sestra, the file Sestra.MLG will appear in the working directory. This file is an operational log that contains any warnings and errors from Sestra. It also contains all information necessary to reproduce the analysis. In particular, it contains references to the input command files (sestra.inp and <prefix>S<n>.FEM), Input Interface File (<prefix>T<n>.FEM), and Loads Interface File (<prefix>L<n>.FEM).

[3] The unformatted sequential format (SIU) is not available for the Results Interface File.

2.11 Linear solver options

2.11.1 In-core and out-of-core matrix factorization

When matrix factorization is performed in in-core mode, the linear solver requires all associated data to fit in main (physical) memory. When matrix factorization is performed in out-of-core mode, the linear solver uses temporary data files in the Sestra working directory to limit the memory footprint.

The default behavior of Sestra is in-core mode; if Sestra detects that there is not sufficient physical memory for in-core mode, out-of-core mode is automatically enabled.

Even in out-of-core mode there is a minimum physical memory requirement for factorization. If Sestra detects that this requirement is not satisfied, Sestra (in default mode) attempts to use virtual (swap) memory in *in-core* mode instead.

The in-core/out-of-core solver options can be controlled with the SOLV command.

Note that in addition to the memory needed for matrix factorization Sestra requires memory for internal storage of matrices, model data and loads. The memory requirement for such data depends on model size, the number of load cases and the analysis type to be performed.

2.11.2 Memory usage for loads

In a linear static analysis all load vectors are with default options assembled and stored in memory for maximum performance. It is possible to limit the number of load vectors to keep in memory at the same time using the SOLV command. This typically results in only a modest increase in analysis time.

Similarly, in a time-domain direct analysis all load vectors for all load histories are by default kept in memory. This can be also be changed by the user with the SOLV command, and can result in reduction of memory usage for time-domain direct analyses of relatively small models (in terms of number of elements) with a relatively large number of load vectors.

2.11.3 Factorization algorithm

The standard LL^{T} and LDL^{T} factorizations are available for symmetric positive definite and symmetric indefinite matrices, respectively. For linear static analyses LL^{T} factorization is used as default. This option can be controlled from the SOLM command.

3 User's guide

3.1 Execution of Sestra 10.12.0

When running the Sestra program the analysis control data file (sestra.inp), see Section 2.10.1, must be present in the working directory.

It is important to note the following:

- To perform a linear static or direct time-domain or frequency-domain dynamic analysis Sestra 10.12.0 must be run with the command-line parameter /dsf, that is,

  ```
  sestra.exe /dsf
  ```

  If this argument is not present, the analysis will be routed to Sestra 8.8-2. In such case the analysis will still succeed if only input commands supported by Sestra 8.8-2 have been used, but the performance will typically drop significantly.

- Any other analysis must be run without the /dsf command-line argument. The analysis will then be automatically routed to Sestra 8.8-2.

Note that if the analysis is run through Sesam Manager 6.5 or later, the command-line argument is appended automatically for dynamic analyses. Note that Sesam Manager 6.5 does not append the command-line argument automatically for static analyses.

3.2 Linear static analysis

For a linear static analysis, the static equilibrium equation (2.1) is solved. The load matrix R (and thus solution matrix x) may contain real or complex values.

A typical sestra input file for a linear static analysis is shown below.

```
COMM <-1-><-2-><-3-><-4-><-5-><-6-><-7-><-8-><-9-><---10---><---11---><---12--->
COMM <-1-><----2---><----3---><----4---><----5---><----6---><----7---><----8--->
COMM      ANTP MSUM
CMAS      1.   1.
COMM ITYP1
ITOP    1.
COMM PREFIX
INAM ModelInput
COMM PREFIX  FORMAT
RNAM ModelResults   SIN
```

The commands on this input file are interpreted as follows.

CMAS: The analysis type is static (ANTYP=1) and the mass sum will be computed (MSUM=1).

ITOP: The superelement number is 1, and hence the Input Interface File must be on the form <prefix>T1.FEM. Any Loads Interface File must be on the form <prefix>L1.FEM.

INAM: The full filename for the Input Interface File is ModelInputT1.FEM. If a Loads Interface File named "ModelInputL1.FEM" is present, any loads on this file will also be included in the analysis.

RNAM: The full filename for the Results Interface File is ModelResultsR1.SIN. The Results Interface File format is binary.

In addition, the LOHI command may be used to assign (e.g. frequency or time-history) metadata to the static result cases.

3.3 Dynamic time-domain forced response analysis — direct calculation

3.3.1 Overview

In a time-domain forced response analysis, Sestra solves the initial-value problem

$$\begin{aligned} M\ddot{x}(t) + C\dot{x}(t) + Kx(t) &= R(t), \quad t_{\text{init}} \leq t \leq t_{\text{final}}, \\ x(t_{\text{init}}) &= d_{\text{init}}, \\ \dot{x}(t_{\text{init}}) &= v_{\text{init}}, \end{aligned} \tag{3.1}$$

where at time t, $x(t)$ is the displacement vector, $\dot{x}(t)$ is the velocity vector, $\ddot{x}(t)$ is the acceleration vector, M is the mass matrix, C is the damping matrix, K is the stiffness matrix, $R(t)$ is the load vector, d_{init} is the specified initial displacement, and v_{init} is the specified initial velocity.

The execution of a direct-time integration analysis is controlled through several commands on the analysis control data files sestra.inp and <prefix>S<n>.FEM. These commands can be divided into algorithm parameters and load specification. Each set of commands is discussed below.

3.3.2 Algorithm parameters

For specification of algorithmic parameters, the CMAS and FRSP commands below are required for a successful analysis; the remaining commands are optional.

CMAS: Specification of analysis type. The field ANTYP = 2 (for dynamic analysis) must be specified.

FRSP: Specification of the parameters for execution of the time-stepping algorithm. The field CALTYP = 4 (direct time integration) must be specified together with the desired value for the time-step size h.

MDAP: Specification of the damping matrix C.

GENA: Use the Generalized-α method [3] to advance the solution in time. Adaptive time-step selection can also be selected through this command.

PERI: Specification of periodic load. Indicates that the load history is specified only for a single period, and that Sestra repeats the specified load any given number of times. This command is also used for automatic steady-state detection (see below).

DREA: Computation of sums of reaction forces and response-amplitude operators for steady-state reaction-force amplitude.

TOUT, TILO, INDT: Cards used for selection of time instants for output.

3.3.3 Load specification

For load specification, the LOHI, TILO, and LCOM commands are required. The HIST command is only used together with OPT=1 on the LOHI command.

LOHI: The top-level command for definition of a load history and associated initial conditions. An analysis can consist of one or several load histories and thus the input file may contain one or several LOHI commands. The LOHI command contains references to all other commands specific to the load history.

TILO: Definition of time instants for time-dependent load specification (if the TOUT command is not present, the time instants on TILO will also be used for output).

LCOM: Selection of load cases for time-dependent load specification.

HIST: Specification of a time history (to be associated with a single load case). Only used if OPT=1 on the LOHI command.

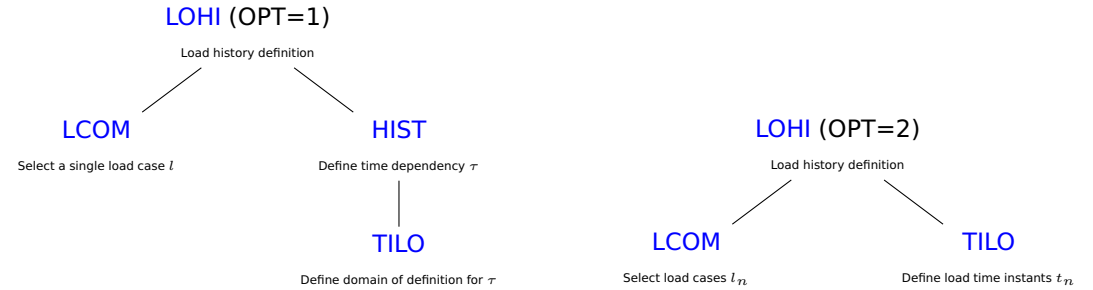
PERI: Specification of periodic loads and automatic steady-state detection.

SCAL: Specification of scaling factors or upramping associated with a load history.

A load history R is defined from a top-level LOHI command and can be specified in two distinct ways.

- $R(t)$ is given for any value of t as $R(t) = l\tau(t)$, where l corresponds to a single load case and $\tau(t)$ specifies the scaling of this load case with time.

  In this case, OPT=1 must be specified on the LOHI command. The load case number is specified on the associated LCOM command, and the scaling with time is specified on the associated HIST and TILO commands.



(a) The load history is expressed as $R(t) = l\tau(t)$. The spatial, constant-in-time load distribution l is defined via a single load case with load-case number specified on an LCOM command; the temporal dependence τ is defined via a HIST command.

(b) The load history $R(t)$ is defined through pairs of time instants t_n and spatial load distributions $l_n = R(t_n)$. The time instants are specified on a TILO command; each spatial load distribution l_n corresponds to a load case with load-case number specified on an LCOM command.

Figure 3.1: Specification of a load history

The dependency graph between the relevant commands is illustrated in Figure 3.1a.

The time dependency function τ can be specified at discrete time instants, corresponding to OPT=1 on the HIST command, or as an explicit function expression, corresponding to OPT=2 on the HIST command. In the former case, linear interpolation is used to evaluate the load function for values of t between the specified instants. An example Sestra input file (for the case OPT=2) is included in Section 3.3.5 (Example 1).

- $R(t)$ is given as snapshots l_n, $1 \leq n \leq N$, corresponding to associated time instants t_n such that $l_n = R(t_n)$. The l_n correspond to N load cases with load-case numbers specified on the LCOM command; the t_n are specified on the TILO command.

  In this case, OPT=2 must be specified on the LOHI command.

  Note that the HIST command is not used. The dependency graph between the relevant commands is illustrated in Figure 3.1b. An example Sestra input file is included in Section 3.3.5 (Example 2).

  Linear interpolation is used to evaluate the load history for values of time between the specified time instants. Linear extrapolation is used to evaluate the load history for values of time larger than t_N.

3.3.4 Calculation of reaction forces and response-amplitude operators

Reaction forces (base shear and overturning moment)

With the DREA command, it is possible to calculate at each timestep the sums of reaction forces over a specified set of nodes. The sums are calculated separately for each of the translational and rotational degrees of freedom, and stored on a separate CSV file for each load history (see Section 2.10).

In the case that all nodes with support (i.e., with prescribed degrees of freedom) are selected, these reaction force sums correspond to the base shear (for translation) and overturning moment (for rotation) of the structure.

The DREA command has an option for automatic selection of all nodes with support, or manual selection of a set of nodes. Note that if the automatic selection is used, the node will be included as long as at least one degree of freedom is prescribed.

The theoretical background for the calculation of reaction forces is discussed in Appendix B.3.

Steady-state response-amplitude operators

For analysis of a structure subjected to periodic wave loading conditions it is often useful to study the response-amplitude operators for the steady-state response of the structure as a function of the wave frequency or period.

The response amplitude is defined as the range of the reaction-force sums when the structure has reached its periodic response (steady state) divided by the wave height. The range is the largest value minus the smallest value over one response cycle.

If the DREA command is present, and if for a load history

- a periodic load is specified with the PERI command, and
- a seastate including wave height, wave period, and wave direction is specified with the SEAS command,

then this load history will be automatically included in the calculation of the response-amplitude operator for the relevant wave direction.

Note that the amplitude of the reaction-force sum is always calculated from the final cycle of the direct time integration analysis regardless of whether the response has reached the steady state. However, automatic steady-state detection can be enabled on the PERI command.

3.3.5 Input-file examples

Note that some of the commands below, typically LOHI, LCOM, TOUT, TILO, HIST, and PERI, are in practical analyses normally found in the <prefix>S<n>.FEM file rather than in the sestra.inp file.

Example 1: loads specified via a time-dependent function

The following sestra.inp input file may be used to run an analysis for which the time-dependent load is specified as an explicit function. A simple analysis using this input file can be found in the installation folder under <installation folder>/Doc/Examples/Example1.

```
COMM SESTRA 10 installation examples. Example 1.
COMM Load history given as (load case 1)*sin(3.14*t)
COMM <-1-><-2-><-3-><-4-><-5-><-6-><-7-><-8-><-9-><---10---><---11---><---12--->
COMM <-1-><----2---><----3---><----4---><----5---><----6---><----7---><----8--->
COMM      ANTP MSUM                                    CSING
CMAS      2.   1.                                      1.e-8
COMM CALTY                                                          H
FRSP 4                                                              0.01
COMM ID        OPT  LCOM HIST TOUT
LOHI 1.        1.   1.   1.   1.
COMM ID             OPT       LC1
LCOM 1.             1.        1.
COMM ID   TILO      OPT       FUNC      NSTEP     A         B         C
HIST 1.   1.        2.        1         4         1.0       3.14      0.0
COMM ID             OPT       T1        T2        T3        T4
TILO 1.             1.        0.        0.5       1.0       1.5
COMM ID             OPT       T1        DELT      TN
TOUT 1.             2.        0.0       0.05      1.5
COMM ITYP1
ITOP 1.
INAM Example1
RNAM Example1 FORMATTED
```

The commands on this input file are interpreted as follows (only the commands with direct relevance for the direct time integration are discussed).

CMAS: The analysis type is dynamic (ANTYP=2) and the mass sum is calculated (MSUM=1).

FRSP: The calculation type is direct time integration (CALTYP=4) and the time-step size parameter is 0.01 (H).

LOHI: OPT=1 so that the load is given as $R(t) = l\tau(t)$ where l refers to a single load case (to be specified on LCOM), and τ refers to an explicitly specified time dependence (to be specified on HIST).

LCOM: l is specified as load case 1 (LC1=1).

HIST: The explicit time dependence τ is $\tau(t) = \sin(3.14t)$. NSTEP=4, which means that this function is followed until the 4'th time instant specified on the referenced TILO command.

TILO: Four time instants are specified for the load. Note that only the value of T4 is relevant in this example because Sestra need only know the initial and the final time for the load.

TOUT: The results are written for $t = 0, 0.05, 0.1, 0.15, \ldots, 1.45, 1.5$.

Note that the MDAP command is not present and hence there is no damping included in the model. Also note that the GENA command is not present and hence the Wilson Θ-method is used.

Example 2: loads specified as load cases at time instants

The following sestra.inp input file may be used to run an analysis for which the time-dependent load is specified via several load cases. It also demonstrates the use of the Generalized-α algorithm using adaptive time stepping. A simple analysis using this input file can be found in the installation folder under <installation folder>/Doc/Examples/Example2.

```
COMM SESTRA 10 installation examples. Example 2.
COMM Load history specified as snapshot load cases.
COMM <-1-><-2-><-3-><-4-><-5-><-6-><-7-><-8-><-9-><---10---><---11---><---12--->
COMM <-1-><----2---><----3---><----4---><----5---><----6---><----7---><----8--->
COMM      ANTP MSUM                                      CSING
CMAS      2.   1.                                        1.e-8
COMM CALTY                                                                H
FRSP 4                                                                    5.e-4
COMM RHO  ADAPTOL   PRINT
GENA 0.8  0.001     1.
COMM      ALPHA_1             ALPHA_2
MDAP      1.e-3               1.e-3
COMM ID        OPT  LCOM TILO TOUT
LOHI 1.        2.   1.   1.   1.
COMM ID             OPT       LC1       LSTEP     LCN
LCOM 1.             2.        1.        1.        9.
COMM ID             OPT       T1        DELT      TN
TILO 1.             2.        0.        0.1       0.8
COMM ID             OPT       T1        DELT      TN
TOUT 1.             2.        0.        0.01      0.8
COMM ITYP1
ITOP 1.
INAM Example2
RNAM Example2 FORMATTED
```

The commands on this input file are interpreted as follows.

CMAS: The analysis type is dynamic (ANTYP=2) and the mass sum is calculated (MSUM=1).

FRSP: The calculation type is direct time integration (CALTYP=4) and the *initial* time-step size parameter is $5 \cdot 10^{-4}$ (H).

GENA: The Generalized-α method is used with dissipation parameter 0.8 (RHO). The time-step sizes are controlled adaptively by Sestra (ADAPTOL specified) with relative tolerance 0.001. Note that H specified on FRSP is in this case only a suggested initial time-step size. A log containing the time-step sizes chosen by the time-stepping algorithm is printed to a file (PRINT=1).

MDAP: Proportional damping is used so that $C = 10^{-3}M + 10^{-3}K$.

LOHI: OPT=2 so that the load $R(t)$ is given as snapshots (to be specified as load cases on LCOM with ID=1) at specific points in time (to be specified on TILO with ID=1).

LCOM: The load cases 1 through 9 are used to specify the time-dependent load.

TILO: The load cases specified on LCOM are to be used at $t = 0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8$. Linear interpolation is used to evaluate the load for other values of t.

TOUT: The results are written to file for $t = 0, 0.01, 0.02, \ldots, 0.8$.

Example 3: specification of periodic load and steady-state detection

The following sestra.inp input file demonstrates the use of the PERI command for periodic load specification and steady-state detection, and the DREA command for calculation of reaction forces. A simple analysis using this input file can be found in the installation folder under <installation folder>/Doc/Examples/Example3.

```
COMM SESTRA 10 installation examples. Example 3.
COMM Periodic load specification and steady-state detection.
COMM <-1-><-2-><-3-><-4-><-5-><-6-><-7-><-8-><-9-><---10---><---11---><---12--->
COMM <-1-><----2---><----3---><----4---><----5---><----6---><----7---><----8--->
COMM      ANTP MSUM                                      CSING
CMAS      2.   1.                                        1.e-8
COMM CALTY                                                          H
FRSP 4                                                              1.e-4
GENA
COMM      ALPHA_1             ALPHA_2
MDAP      0                   1.e-5
COMM ID        OPT  LCOM TILO TOUT                           PERI
LOHI 1.        2.   1.   1.   1.                            1.
COMM ID             OPT       LC1       LC2       LC3       LC4       LC5
LCOM 1.             1.        1.        2.        3.        4.        5.
COMM      LC6       LC7       LC8       LC9
          6.        7.        8.        1.
COMM ID             OPT       T1        T2        T3        T4        T5
TILO 1.             1.        0.        0.0875    0.175     0.2625    0.35
COMM      T6        T7        T8        T9
          0.4375    0.525     0.6125    0.7
COMM ID             OPT       T1        DELT      TN
TOUT 1.             2.        0.        0.01      0.7
COMM ITYP1
ITOP 1.
INAM Example3
RNAM Example3 FORMATTED
COMM ID   CYCLEMAX  PERIOD    STEADYTOL CYCLESTORE
PERI 1.   100       0.7       0.01      3
COMM OPT
DREA 0.
```

The commands on this input file are interpreted as follows.

CMAS: The analysis type is dynamic (ANTYP=2) and the mass sum is calculated (MSUM=1).

FRSP: The calculation type is direct time integration (CALTYP=4) and the *initial* time-step size parameter is $1 \cdot 10^{-4}$ (H).

GENA: The Generalized-α method is used with default parameters. A constant time-step value is used (as specified on FRSP).

MDAP: Proportional damping is used so that $C = 10^{-5}K$.

LOHI: OPT=2 so that the load $R(t)$ is given as snapshots (to be specified as load cases on LCOM width ID=1) at specific points in time (to be specified on TILO with ID=1). The load is periodic (periodicity and steady-state detection parameters to be specified on PERI with ID=1).

LCOM: The load cases 1 through 8 are used to specify the time-dependent load. Note that load case 1 is repeated for the 9th snapshot. Since the PERI command is specified, the load specification is for a single load cycle.

TILO: The load cases specified on LCOM are to be used at equidistant times

$$t = 0, 0.0875, 0.175, 0.2625, 0.35, 0.4375, 0.525, 0.6125, 0.7.$$

Since the PERI command is specified, the load specification is for a single load cycle.

TOUT: The results are written to file for time instants

$$t = 0, 0.01, 0.02, \ldots, 0.7$$

within each load cycle. Since the PERI command is specified, the result will be written to file for a specified number of load periods.

PERI: The load period is 0.7 (PERIOD). The load specified through LCOM and TILO is repeated up to 100 times (CYCLEMAX), but is terminated if steady state is reached. Steady state is detected if the system response is periodic with a tolerance of 0.01 (STEADYTOL). The results are written to file for the last 3 load cycles (CYCLESTORE).

DREA: The sum of reaction forces about the superelement origin (0,0,0) are computed and printed to file for all load cycles. All nodes with prescribed displacement are included in the calculation.

3.4 Dynamic time-domain forced response analysis — modal superposition

3.4.1 Overview

In a modal superposition (time domain) analysis, the initial-value problem (3.1) is approximated by requiring that the solution $x(t)$ is (for all values of t) a linear combination of the S eigenvectors (mode shapes) corresponding to the S smallest eigenvalues. This, together with the application of proportional damping (Reyleigh damping) and modal damping, results in S decoupled single-degree-of-freedom equations which can be solved very efficiently. As S is typically chosen much smaller than N (the number of degrees of freedom in the original model), this results in significant computational savings compared to a direct integration approach.

This analysis approach consists of the following steps

1. Solution of the generalized eigenproblem

$$KV = M\Lambda V, \qquad V^{\mathrm{T}}MV = I, \tag{3.2}$$

   for S eigenpairs (v_i, λ_i). Typically, the eigenpairs corresponding to the S smallest eigenvalues are used. The step is equivalent to a standard free vibration analysis.

2. Projection of the equation (3.1) onto the subspace spanned by the selected mode shapes. In particular, the solution (displacement) $x(t)$ is required to be a linear comination of the mode shapes, such that $x(t) = Vy(t)$. This step results in S decoupled time-dependent equations for each of the components y_i of y.

3. Integration of the uncoupled equations in time.

4. Expansion of the displacement solution as $x(t) = Vy(t)$ for selected values of t, using the calculated numerical approximations to $y(t)$.

3.4.2 Input parameters

The input for a modal superposition (time-domain) analysis is the same as for direct time integration, with the following exceptions:

EIGA: Required input for solution of the eigenproblem (step 1 above). All eigenmodes calculated in this step are used in the subsequent modal analysis.

FRSP: Specification of the parameters for execution of the time-stepping algorithm. The field CALTYP = 3 (modal superposition) must be specified together with the desired value for the time-step size. Note that the time-step size is specified implicitly by the number of time steps n within each period T_i associated with each uncoupled equation. Thus the time-step size for each equation is $h_i = T_i/n = 2\pi/(n\sqrt{(\lambda_i)})$ where λ_i is the eigenvalue associated with mode shape number i.

MDAF: Specification of modal damping. In addition, proportional damping is supported using the MDAP command.

PERI: Periodic load specification and steady-state detection is not supported.

HIST: Specification of a load history using this command is not supported.

3.5 Dynamic frequency-domain forced response analysis — direct calculation

3.5.1 Overview

In a frequency-domain forced-response analysis, Sestra solves the problem

$$M\ddot{x}(t) + C\dot{x}(t) + Kx(t) = R(t) \tag{3.3}$$

in the frequency domain, subjected to harmonic loading conditions $R(t)$. A separate complex-valued system of equations is obtained for each load frequency. See Appendix B.2 for a detailed description of this analysis.

Note that in addition to the viscous damping term present in (3.3), structural damping (that is, damping proportional to the displacements and in phase with the velocities) may also be considered (see Section 2.8.4).

3.5.2 Relevant commands

CMAS: Specification of analysis type. The field ANTYP = 2 (for dynamic analysis) must be specified.

FRSP: Specification of dynamic forced-response algorithm. The field CALTYP = 2 (frequency-domain direct calculation) must be specified.

LOHI: Specification of a load history with reference to LCOM (load cases), FRLO (load frequencies), and SEAS (optional seastate specification).

LCOM: Specification of load cases to be included within one load history.

FRLO: Specification of frequencies of load cases specified on the corresponding LCOM command.

SEAS: Specification of seastate parameters.

FRQL: Alternative and simplified specification of combination of load cases and load frequencies. If present, any LOHI (and thus LCOM, FRLO, and SEAS) commands may not be present.

DREA: Calculation of response-amplitude operators (base shear/overturning moment). If present together with SEAS commands containing wave height and wave direction parameters the corresponding load history will be included in the calculation.

3.5.3 Calculation of response-amplitude operators

The response amplitude is defined as the range of the reaction-force sums divided by the wave height. Thus for a frequency-domain dynamic analysis with no specified wave height, the response amplitudes may be computed directly as the modulus of the (complex) sums of reaction forces (if no wave height is specified a wave height equal to two and thus unity wave amplitude is assumed).

If

- the DREA command is present, and
- from a LOHI command a SEAS command is referenced, and
- the seastate specified by this SEAS command has a specified wave direction,

then the response associated with this LOHI command will be included in the calculation of the response-amplitude operator for the relevant wave direction.

3.5.4 Input-file examples

Note that some of the commands below, typically LOHI, LCOM, TOUT, TILO, HIST, and PERI, are in practical analyses normally found in the <prefix>S<n>.FEM file rather than in the sestra.inp file.

Example 4: loads specified as load histories

The following sestra.inp input file may be used to run an analysis for which the loads are specified as load histories using the LOHI, LCOM, and FRLO commands. A simple analysis using this input file can be found in the installation folder under <installation folder>/Doc/Examples/Example4.

```
COMM SESTRA 10 installation examples. Example 4.
COMM Dynamic forced response analysis using two load histories
COMM <-1-><-2-><-3-><-4-><-5-><-6-><-7-><-8-><-9-><---10---><---11---><---12--->
COMM <-1-><----2---><----3---><----4---><----5---><----6---><----7---><----8--->
COMM      ANTP MSUM                                    CSING
CMAS      2.   1.                                      1.e-8
COMM CALTY
FRSP 2.
COMM      ALPHA_1             ALPHA_2
MDAP      0.01                0.01
COMM      ETA
MDAS      0.02
COMM ID        OPT  LCOM FRLO               SEAS
```

```
LOHI 1.        6.   1.   1.                  1.
LOHI 2.        6.   2.   2.                  2.
COMM ID             OPT       LC1       LC2       LC3
LCOM 1.             1.        1.        3.        5.
LCOM 2.             1.        2.        4.        6.
COMM ID             OPT       F1        F2        F3
FRLO 1.             1.        0.3       0.6       0.9
FRLO 2.             1.        1.1       1.2       1.3
COMM ID                                           DIR
SEAS 1.                                           120.
SEAS 2.                                           45.
COMM ITYP1
ITOP 1.
INAM Example4
RNAM Example4 FORMATTED
DREA 0.
```

The commands on this input file are interpreted as follows.

CMAS: The analysis type is dynamic (ANTYP=2) and the mass sum is calculated (MSUM=1).

FRSP: The calculation type is direct forced-frequency response (CALTYP=2).

LOHI: Two load histories are given (ID=1 and ID=2). OPT=6 so that the load is interpreted as a frequency-domain load. References are made to LCOM, FRLO, and SEAS commands.

LCOM: For the command with ID=1, load cases 1, 3, and 5 are selected. For the command with ID=2, load cases 2, 4, and 6 are selected.

FRLO: For the command with ID=1, frequencies 0.3, 0.6, and 0.9 are selected. For the command with ID=2, frequencies 1.1, 1.2, and 1.3 are selected.

SEAS: For the command with ID=1, the wave direction is 120 degrees; for the command with ID=2, the wave direction is 45 degrees.

MDAP: Proportional damping is used.

MDAS: Structural damping is used.

DREA: Calculation of response-amplitude operators (RAO) for base shear and overturning moment is requested.

Example 5: simplified load specification

The following sestra.inp input file may be used to run an analysis for which the loads and frequencies are specified using the FRQL command. A simple analysis using this input file can be found in the installation folder under <installation folder>/Doc/Examples/Example5.

```
COMM SESTRA 10 installation examples. Example 5.
COMM Dynamic forced response analysis with simplified load specification
COMM <-1-><-2-><-3-><-4-><-5-><-6-><-7-><-8-><-9-><---10---><---11---><---12--->
COMM <-1-><----2---><----3---><----4---><----5---><----6---><----7---><----8--->
COMM      ANTP MSUM                                      CSING
CMAS      2.   1.                                        1.e-8
COMM CALTY
FRSP 2.
COMM      ALPHA_1             ALPHA_2
MDAP      0.01                0.02
COMM      F1        FSTEP     FN        LC1       LSTEP     LCN
FRQL      0.1       0.1       1.0       1.        1.        4.
COMM ITYP1
ITOP 1.
INAM Example5
RNAM Example5 FORMATTED
```

The commands on this input file are interpreted as follows.

CMAS: The analysis type is dynamic (ANTYP=2) and the mass sum is calculated (MSUM=1).

FRSP: The calculation type is direct forced-frequency response (CALTYP=2).

FRQL: Ten load frequencies $(0.1, 0.2, \ldots, 1.0)$ and four load cases $(1, 2, 3, 4)$ are given so that in total forty analyses will be performed.

MDAP: Proportional damping is used so that $C = 0.01M + 0.02K$..

3.6 Dynamic frequency-domain forced response analysis — modal superposition

3.6.1 Overview

In a modal superposition (frequency domain) analysis, the steady-state solution to (3.3) is approximated by requiring that the solution $x(t)$ is (for all values of t) a linear combination of the S eigenvectors (mode shapes) corresponding to the S smallest eigenvalues. This, together with the application of proportional damping (Reyleigh damping) and modal damping (as fraction of critical damping or structural damping), results in S decoupled complex-valued equations which are solved very efficiently. As S is typically chosen much smaller than N (the number of degrees of freedom in the original model), this results in significant computational savings compared to a direct integration approach.

This analysis approach consists of the following steps

1. Solution of the generalized eigenproblem

$$KV = M\Lambda V, \qquad V^{\mathrm{T}}MV = I, \tag{3.4}$$

   for S eigenpairs (v_i, λ_i). Typically, the eigenpairs corresponding to the S smallest eigenvalues are used. The step is equivalent to a standard free vibration analysis.

2. Projection of the equation (3.3) (in the frequency domain) onto the subspace spanned by the selected mode shapes. This step results in S decoupled complex-valued equations for each load frequency.

3. For each load frequency, solution of S uncoupled complex-valued equations (in practice solved as $2S$ real-valued equations)

4. Expansion of the displacement solution to physical (frequency-domain) space

3.6.2 Input parameters

The input for a modal superposition (frequency-domain) analysis is the same as for direct analysis, with the following exceptions:

EIGA: Required input for solution of the eigenproblem (step 1 above). All eigenmodes calculated in this step are used in the subsequent modal analysis.

FRSP: Specification of frequency-domain forced-response analysis. The field CALTYP = 1 (modal superposition) must be specified.

MDAF: Specification of modal damping. In addition, proportional damping is supported using the MDAP command and structural damping is supported using the MDAS command.

3.7 Free vibration analysis

In a free vibration analysis, Sestra solves the generalized eigenvalue problem

$$Kv_i = \lambda_i M v_i, \tag{3.5}$$

where K is the stiffness matrix, M is the mass matrix, (v_i, λ_i) are pairs of eigenvectors and eigenvalues. Sestra computes a few eigenpairs for eigenvalues close to a specified shift value (zero unless specified), or all eigenvalues in a specified interval.

Note that the natural angular frequencies of the finite element model is $\omega_i = \sqrt{\lambda_i}$. In addition to eigenvalues, Sestra also reports

- the *natural frequencies* $f_i = \sqrt{\lambda_i}/(2\pi)$.
- the *natural periods* $T_i = 1/f_i$

The eigenvalues of the system (3.5) are always real and nonnegative. In case of an unconstrained model up to six eigenvalues may be zero, with the remaining eigenvalues being positive. Due to rounding errors some of the computed approximations to the zero eigenvalues may be negative. In this case a warning is issued and the user should check that the reported frequencies are sufficiently close to zero since modeling errors may also result in, possibly large, negative eigenvalues.

An example of a sestra input file for a free vibration analysis is shown below.

```
COMM <-1-><-2-><-3-><-4-><-5-><-6-><-7-><-8-><-9-><---10---><---11---><---12--->
COMM <-1-><----2---><----3---><----4---><----5---><----6---><----7---><----8--->
COMM      ANTP MSUM
CMAS      2.   1.
COMM ENR       ITMA NLVEC EPS                    ELOW      EHIGH     SHIFT
EIGA 15                                                              0.0
COMM ITYP1
ITOP   1.
COMM PREFIX
INAM
COMM PREFIX  FORMAT
RNAM         SIN
```

The commands on this input file are interpreted as follows.

CMAS: The analysis type is dynamic (ANTYP=2) and the mass sum will be computed (MSUM=1).

ITOP: The superelement number is 1.

EIGA: The control card for the implicitly restarted Lanczos algorithm specifies that the 15 eigenpairs (ENR) with eigenvalues closest to 0.0 (SHIFT) are to be calculated. Note that default values will be used for the algorithm parameters ITMAX, NLVEC and EPS.

3.8 Tension-compression analysis

Truss elements may be defined to be *tension only* or *compression only*. Hence the analysis is non-linear as the stiffness properties of the trusses (in this case “on” or “off”) depends on whether a truss is in tension or in compression.

In a tension-compression analysis, the original model with tension-compression elements is converted internally in Sestra to an equivalent model with standard elements and contact conditions. The solution of this contact problem is the minimizer x of a quadratic optimization problem:

$$x = \arg\min \frac{1}{2} x^T K x - x^T R \tag{3.6}$$

$$\text{s.t.} \quad Ax \geq b; \tag{3.7}$$

here x is the displacement vector, K is the stiffness matrix, R is the load vector, and $Ax \geq b$ represents the tension-compression conditions as equivalent inequalities.

The algorithm employed in Sestra 10.12.0 for solving this problem is an iterative primal-dual interior-point method for convex quadratic optimization [10]. The method is controlled by a tolerance parameter and a parameter for the maximum number of iterations. The tolerance parameter specifies an upper bound for the maximum of three residual quantities — a “primal residual norm,” a “dual residual norm,” and a “duality gap” — calculated at each iteration of the optimization procedure. The precise definition of the residual quantities is provided in Section B.8.

The residual quantities indicate the error in the solution. For each iteration, the algorithm drives the residuals towards zero (note it is possible for the residuals to increase between iterations, typically seen in the first few iterations). The residuals are relative quantities, and a maximum residual smaller than, say, 1.e-8 could be achieved in most cases.

The residual/convergence history is printed to the sestra operational log file (Sestra.MLG); a good quality check is to verify that for the final few iterations the residual drops quickly and monotonically towards zero.

Note that the algorithm will not be able to reach an exact zero residual. Thus, if the tolerance parameter is set too small, the convergence history will reach a plateau.

The algorithm will stop regardless of the value of the residual value when the maximum number of iterations has been performed.

It is possible to select specific load cases for tension-compression analysis. Remaining load cases will not be analyzed by Sestra.

Tension-compression analyses are computationally more demanding than standard linear static analyses. In the interest of saving analysis time, the following workflow is recommended for large models with many load cases:

1. Select only a few load cases for analysis, and specify a small tolerance parameter (e.g. 1.e-10).
2. Run the analysis and inspect the convergence histories on Sestra.MLG. If a plateau is observed, consider whether an increase of the tolerance parameter to a value above the plateau level is acceptable.
3. Select all relevant load cases for analysis, adjust the tolerance parameter (if desired), and run the analysis again.

An example of a sestra input file for a tension-compression analysis is shown below.

```
COMM <-1-><-2-><-3-><-4-><-5-><-6-><-7-><-8-><-9-><---10---><---11---><---12--->
COMM <-1-><----2---><----3---><----4---><----5---><----6---><----7---><----8--->
COMM      ANTP MSUM
CMAS      3.   1.
COMM ITER   TOL
INTP 100  1.e-10
COMM                     OPT       LC1       LC2       LC3       ...       LCN
LCAS                      1.        3.        5.
COMM ITYP1
ITOP   1.
COMM PREFIX
INAM TCModel
COMM PREFIX  FORMAT
RNAM TCModel   SIN
```

The commands on this input file are interpreted as follows.

CMAS: The analysis type is tension-compression (ANTYP=3) and the mass sum will be computed (MSUM=1).

ITOP: The superelement number is 1.

INAM: The full filename for the top-level Input Interface File is TCModelT1.FEM. Any Loads Interface Files must follow named the naming convention <ModelInput>L<n>.FEM, where n is the number of the associated low-level superelement.

INTP: The interior-point algorithm is specified with maximum number of iterations 100 (ITER) and with tolerance parameter 1.e-8 (TOL).

LCAS: Load cases 3 and 5 are specified for tension-compression analysis.

RNAM: The full filename for the Results Interface File is TCModelR1.SIN (binary format).

3.9 Postprocessing

In postprocessing, element forces and stresses may be calculated based on a Results Interface File containing nodal displacements. A typical workflow is as follows.

Step 1: Perform a Sestra analysis, but do not calculate element forces and stresses (controlled with the RSEL command). The Results Interface File must be stored in SIN binary format.

Step 2: Perform postprocessing with the Results Interface File generated in step 1 as input. Note that the original Input Interface File from step 1 is not used.

Postprocessing may be applied to analysis results from any analysis type performed in step 1. It is enabled with the CMAS command.

Step 1

This step is a regular Sestra analysis, where calculation and print of element forces and stresses is disabled. This is controlled with ISEL3 and ISEL4 fields on the the RSEL command as in the following example.

```
COMM <-1-><-2-><-3-><-4-><-5-><-6-><-7-><-8-><-9-><---10---><---11---><---12--->
COMM ISEL1          ISEL3ISEL4     ISEL7
RSEL   0.            -1.  -1.        1.
```

Note that in this example, ISEL1=0 and ISEL7=1. These parameters control the transfer of loads from the Input and Loads Interface Files to the Results Interface File.

It is critical to pay particular attention to the loads transferred to the Results Interface File in step 1. This is because, in step 2, forces and moments for any two-noded beam element is calculated from the nodal

displacements as well as from the loads applied to the element (including inertia forces). In step 2, these loads are read from any Loads Interface File as well as from the Results Interface File generated in step 1.

In order to obtain correct calculation of forces and moments in step 2 it is therefore critical that loads are correctly represented either on the Results Interface File from step 1 *or* in a separate Loads Interface File.

Note: In order to apply the correct load case (or load combination) to each result case in step 2, Sestra 10.12.0 requires RBLODCMB cards to be present on the Results Interface File generated in step 1. Such cards are always present on Results Interface Files produced by Sestra 10.12.0 or later. Earlier versions of Sestra does not always store this card.

The following situations must be considered. The instructions assume that step 1 is performed using Sestra 10.

- If in step 1 loads (applicable to two-noded beam elements) are present on the Input Interface File, the RSEL command must be used in order to correctly transfer the loads to the Results Interface File. Consider the following example.

```
COMM <-1-><-2-><-3-><-4-><-5-><-6-><-7-><-8-><-9-><---10---><---11---><---12--->
COMM ISEL1          ISEL3ISEL4     ISEL7
RSEL  0.             -1.  -1.        1.
```

  Here,

  - ISEL7=1 instructs Sestra to transfer all loads from the Input and Loads Interface Files to the Results Interface File.
  - ISEL1=0 instructs Sestra *not* to convert inertia loads for two-noded beam elements to equivalent distributed line loads and subsequently transfer these converted loads to the Results Interface File)

    Note: ISEL1=0 is critical for a model with two-noded beam elements. With ISEL1=1 both inertia loads converted to distributed line loads and the original inertia loads are transferred to the Results Interface File. This causes the calculation of stresses and forces in the Sestra postprocessing to be incorrect. ISEL1=1 is used when distributed line loads are required for postprocessing in applications such as Framework and Xtract.

    ISEL1=0 is correct only if step 1 is performed using Sestra 10. If Sestra 8.8-2 or earlier is used for step 1, ISEL1=11 must be used. Consult the Sestra 8.8-2 user's manual for details.

  - ISEL3=-1 and ISEL4=-1 instruct Sestra not to calculate and store element forces and stresses.

  Recall that in this situation loads are copied from the Input and Loads Interface Files to the Results Interface File. Thus, any Loads Interface File (containing loads applicable to two-noded beam elements) from step 1 must not be applied in step 2 as it would cause duplication of loads and thus incorrect calculation of forces and moments for two-noded beam elements.

  A Loads Interface File is applied to the analysis if the file name corresponds to the prefix specified on the INAM command or, if present, the LNAM command.

- If in step 1 loads (applicable to two-noded beam elements) are not present on the Input Interface File (but only on the Loads Interface File), the user has two options:

  - Do not transfer any loads to the Results Interface File (ISEL1=0 and ISEL7=0 on the RSEL command), and keep the Loads Interface File in the postprocessing.

- Transfer all loads from the Loads Interface File to the Results Interface File (ISEL7=1 on the RSEL command), and do not use the Loads Interface File in the postprocessing.

  It is critical that no inertia loads for two-noded beam elements on the Loads Interface File are converted to distributed line loads and transferred to the Results Interface File as it would cause incorrect calculation of beam element forces in postprocessing. Hence ISEL1=0 on the RSEL command. ISEL1=0 is here correct only if step 1 is performed using Sestra 10. If Sestra 8.8-2 or earlier is used for step 1, ISEL1=11 must be used. Consult the Sestra 8.8-2 user's manual for details.

Step 2

For postprocessing the following Sestra input file serves as an example regardless of the analysis type performed in step 1.

```
COMM <-1-><-2-><-3-><-4-><-5-><-6-><-7-><-8-><-9-><---10---><---11---><---12--->
COMM <-1-><----2---><----3---><----4---><----5---><----6---><----7---><----8--->
COMM      ANTP
CMAS        11.
COMM ITYP1
ITOP   1.
COMM PREFIX
INAM Displacements
COMM PREFIX       FORMAT
RNAM AllResults   SIN
COMM ISEL1     ISEL3ISEL4          ISEL7
RSEL    1.         1.   1.             1.
```

The key commands on this input file are

CMAS: The analysis type is postprocessing (ANTYP=11).

INAM: Since this is postprocessing, INAM in this case refers to the file DisplacementsR1.SIN. The superelement number 1 is specified on the ITOP command.

RNAM: The Results Interface File containing displacements, stresses and forces is AllResultsR1.SIN. The file is binary.

RSEL: Stresses and forces are calculated (ISEL3=1 and ISEL4=1). To (in this example) accommodate postprocessing of forces and moments for two-noded beam elements in Framework and Xtract, inertia loads for two-noded beam elements are converted to line loads (ISEL1=1) and loads are transferred to the Results Interface File (ISEL7=1).

Note that a Results Interface File generated in postprocessing does not necessarily have identical file size to the equivalent Results Interface File generated in a standard analysis. This is due to the way data is stored internally in the SIN database file.

3.10 Stress stiffening analysis

Stress stiffening may be applied to a linear static analysis, contact/tension-compression analysis, free vibration analysis or forced dynamic analysis. When stress stiffening is used, Sestra first performs a static analysis. The resulting stresses are used to compute element geometric stiffness matrices, which are added to the regular element stiffness matrices in the subsequent analysis.

Stress stiffening is enabled using the STIF option on the CMAS command. Options for the generation of the geometric stiffness matrix may be controlled from the separate GSTF command.

Results (including element forces and stresses) from the initial static analysis are not added to the results from the second analysis.

A static analysis with stress stiffening gives differences in the sums of moments from loads and sums of reaction moments. This is because the geometric stiffness matrix is added to the regular stiffness matrix, and the geometric stiffness matrix contains the forces from the "stress stiffening" load. The reaction moments

are calculated with the modified stiffness matrix and will then in addition to the moments from the load have (stiffening) moments from the stress stiffening load.

Consider for example a cantilever beam with length L that has an axial tension load as stress stiffening load and a perpendicular point load at the end of the beam. This point load P makes the cantilever bend with the displacement D. The tension load S will then give a (stiffening) moment DS. This moment will be subtracted from the reaction moment PL in the fixed end.

Limitations:

- Geometric stiffness matrices are generated only for two-noded beam elements (BEAS); for three-noded (FTRS, THTS) and four-noded (FQUS, THQS) first-order shell elements; and for six-noded (SCTS) and eight-noded (SCQS) second-order curved shell elements. For any other element types present in the model the geometric stiffness matrices will be zero.
- An approximation is made for calculation of forces and moments for two-noded beam elements (BEAS): *i)* "static" forces and moments are calculated from the stiffness matrix and the displacements at the ends of the beam; *ii)* the values at the midpoint of the beam are calculated as the mean of the end-point values. The effect of a distributed load acting on the beam is consequently neglected for the midpoint.
- Experience shows that very high tension loads for two-noded beams (tension above the absolute value of the buckling load) give a smaller stiffening effect than predicted by theory: the eigenfrequencies calculated by Sestra are too small.

3.11 Superelement analysis

In a superelement analysis, the superelement hierarchy assembled by Presel is first merged to an equivalent global model. Subsequently, Sestra performs the desired analysis using this global model. Finally, results are distributed back to the original superelements.

Note: The Sestra superelement merger module assumes that the superelement hierarchy generated by Presel is self consistent. It is therefore critical that Presel is run as a final step following any modification to the Sesam Input Interface file for each superelement.

An example of a Sestra input file for a linear static superelement analysis is shown below.

```
COMM <-1-><-2-><-3-><-4-><-5-><-6-><-7-><-8-><-9-><---10---><---11---><---12--->
COMM <-1-><----2---><----3---><----4---><----5---><----6---><----7---><----8--->
COMM      ANTP MSUM
CMAS      1.   1.
COMM ITYP1
ITOP   10.
COMM PREFIX
INAM ModelInput
COMM PREFIX  FORMAT
RNAM ModelResults   SIN
IDTY  2.
SOLT       1.
```

The commands on this input file are interpreted as follows.

CMAS: The analysis type is static (ANTYP=1) and the mass sum will be computed (MSUM=1).

ITOP: The top-level superelement number is 10, and hence the top-level Input Interface File (generated by e.g. Presel) must be on the form <prefix>T10.FEM.

INAM: The full filename for the top-level Input Interface File is ModelInputT10.FEM. Any Loads Interface

Files must follow named the naming convention <ModelInput>L<n>.FEM, where n is the number of the associated low-level superelement.

RNAM: The full filename for the Results Interface File is ModelResultsR10.SIN. The Results Interface File format is binary and contains results for the individual superelements.

IDTY and SOLT: The SOLT command with option NONS=1 is specified for superelement number 2, indicating that this superelement is to be considered as non-structural.

3.12 Common info and warning messages

During execution of Sestra 10.12.0, information, warning, and error messages are reported to the operational log file Sestra.MLG.

Some warning messages may indicate that there is a serious issue with the analysis, for example due to a modelling error. Other warning messages are less severe.

A list of common warning messages is provided below.

- **Stiffness singularity coefficient (CSING) or mass singularity coefficient (SINGM) is zero.** The CSING and SINGM fields on the CMAS command controls the singularity coefficient for the stiffness matrix and mass matrix of shell and membrane elements without the drilling degree of freedom. This is the case for the shell elements FTRS, FQUS, SCTS, and SCQS, and for the membrane elements CSTA, LQUA, ILST, and IQQE.

  If the CSING or SINGM coefficient is set to zero, the stiffness or mass matrix, respectively, may for certain models become singular. A singular stiffness matrix may result in termination of Sestra or invalid results; a singular mass matrix may result in termination of Sestra or invalid results for direct time integration analysis.

  In fact, if the sestra.inp file is generated automatically by some preprocessors, the value of CSING may be set to zero. The reason for this is that older versions of Sestra interprets a zero value as an indication that a default value determined by Sestra should be used. Sestra 10.12.0, on the other hand, interprets the zero value as specified. A default value is used if the field is left blank.

  It is therefore important to set the CSING and SINGM parameters to a desired non-zero value, or leave the fields blank to use the default values.

4 Sestra input commands

Note: The input commands for Sestra 10.12.0 is a union of a subset of the commands available for Sestra 8.8-2, and new commands for controlling new features.

In this chapter, all input commands available for Sestra 10.12.0 are listed.

For commands relevant for analyses only available in Sestra 8.8-2, see [1].

Note that if an input file which contains new commands available only in Sestra 10.12.0 is used with Sestra 8.8-2, the analysis will fail with an error message.

4.1 General description of input command format

The Analysis Control Data consists of commands given on records consisting of eighty characters.

The *first four* characters are used for the command identifier (type). The identifier field is blank when the record is a continuation of a command, i.e. more than one record is needed for the data of the command. The fifth character is blank.

For example, a command with the identifier TYPE may be defined as follows.

1——-5	6——-10	11——-15	16——-20	21——-25	26——-30	31——-35	36——-40
TYPE	FIELD1		FIELD3				

41——-45	46——-50	51————60	61————70	71————80
				FIELD12

The numbers above each field indicate the columns in which the data for that field must be entered. The data does not have to fill the entire space available for each field, and does not have to be left- or right adjusted.[4]

For the TYPE command above, an associated record on the input file may read

```
COMM 6    11   16   21   26   31   36   41   46   51        61        71
COMM < 1 >< 2 >< 3 >< 4 >< 5 >< 6 >< 7 >< 8 >< 9 ><   10   ><   11   ><   12   >
TYPE  2.72        9                                                     1.04
```

with values entered for the first, third, and twelfth fields. Note that COMM is a special command type for comments (used here to indicate column and field numbers), which is ignored by Sestra.

In Sestra 10.12.0, the order in which the input commands appear on the input file is not relevant.

Note that a *blank field* is interpreted by Sestra 10.12.0 as *not entered*. In this case, a default value is used in certain cases. Default values are documented in the list of input commands in the next section.

[4] In Sestra 8.8-2 values were required to have punctuation marks unless they were right adjusted within their fields. This is not necessary in Sestra 10.12.0.

4.2 List of input commands

In this section all available commands of Sestra 10.12.0 are listed alphabetically.

BSEL

Is present if the load combinations shall be composed from the BSELL data types at the Input Interface File.

In addition, it allows selection of load combinations for analysis.

Only applicable to linear static analysis, quasistatic analysis and pile soil analysis.

1——5	6——10	11————20	21————30	31————40
BSEL			OPT	
			= 0	
			= 1	LC1
			= 2	LC1

41————50	51————60	61————70	71————80
LC2	...		LCN
LSTEP	LCN		

OPT: Load combination selection option.

=0 All load combinations as specified at the BSELL data type is applied. It is noted that a blank BSEL behaves as if OPT = 0.

=1 Load combinations specified as LC1, LC2, LC3, . . . LCN. In this case continuation records may be used to specify as many load combinations as desired. Load combinations continue in column 11 on continuation records.

=2 Load combinations defined by LC1:LSTEP:LCN. For example, LC1=4, LSTEP=2, LCN=10 gives load combinations 4, 6, 8, 10.

LC1: First selected load combination.

LSTEP: Step in load combination number.

LCN: Last selected load combination.

CMAS

Master control data for the analysis.

1——5	6——10	11——15	16——20	21——25	26——30	31——35	36——40
CMAS		ANTYP	MSUM	MOLO	STIF		LBUCK

41——45	46——50	51————60	61————70	71————80
		CSING	SINGM	

ANTYP: Analysis type.

=1 Static analysis.

=2 Dynamic analysis.

=3 Tension-compression analysis.

=11 Postprocessing analysis (calculation of element stresses and forces from a Results Interface File).

MSUM: Computation of global effective mass matrix and centroid.

=0 No computation

=1 Effective mass matrix and centroid are computed and printed (default)

MOLO: Generation of modal load factors (modal participation factors) in a free vibration analysis. Modal load factors are used for earthquake postprocessing by the response spectrum method and for gust wind fatigue analysis in Framework.

=0 Modal load factors are not calculated.

=1 Modal load factors are calculated and stored on the Results Interface File. Also, stresses and forces associated with the mode shapes are stored on the Results Interface File. The mode shapes are normalized with respect to the mass matrix.

STIF: Stress stiffening parameter.

=0 No stress stiffening.

=2 Stress stiffening analysis. The first step is static analysis.

=3 Stress stiffening analysis for tension-compression analysis. The first step is a tension-compression analysis.

If enabled (STIF=2 or STIF=3), stress stiffening is always taken into account for all superelements.

The last load case (or, if load combinations are specifed, the load combination with the largest number) is used for the generation of element geometric stiffness matrices.

An optional GSTF command may be used in addition to select a different load case for the stress stiffening load.

A stress stiffening analysis consists of a two-step procedure automatically performed by Sestra. First, a static analysis is performed to find the element stresses. Based on these stresses geometric stiffness (or initial stress) matrices are calculated for the basic elements. Second, the main analysis (static or dynamic as specified in input) is performed. In this analysis the geometric stiffness matrices are added to the regular stiffness matrices. For tension-compression analysis the first analysis may alternatively be taken to be a tension-compression analysis. To achive this set STIF=3.

Be aware that when using stress stiffening in a tension-compression analysis the state (tension or compression) of tension-only or compression-only elements may not be consistent between the initial stiffening step and the subsequent tension-compression analysis step.

The following elements support calculation of geometric stiffness matrices: two-noded beam element (BEAS); three-noded (FTRS and THTS) and four-noded (FQUS and THQS) shell elements; six-noded (SCTS) and eight-noded (SCQS) curved shell elements. If any other element is used in a stress-stiffening analysis the geometric stiffness matrix for this element is zero and the regular (unmodified) element stiffness matrix is used for this element in the analysis.

If geometric stiffness is imported from file in an analysis using the ESLI command, the STIF parameter is ignored.

LBUCK: Linearized buckling parameter.

=0 No linearized buckling.

=1 Linearized buckling analysis, the last load case (or, if load combinations are specifed, the load combination with the largest number) is used for the generation of element geometric stiffness matrix.

The GSTF command may be used in addition to specify the load used to compute the geometric stiffness matrix.

A linearized buckling analysis consists of two steps. First a static analysis is performed, then, using the results from the former analysis, the geometric stiffness matrix is computed and an eigenvalue analysis compute the stability factor. An EIGA command must be present and a nonzero shift must be given. The shift should be positive and smaller than the smallest eigenvalue.

CSING: Coefficient used for the drilling[5] degree of freedom for the element stiffness matrix of FTRS, FQUS, SCTS, SCQS, THTS, and THQS shell elements. The default value is 10^{-8}.

SINGM: Coefficient used for the drilling degree of freedom for the element mass matrix of FTRS, FQUS, SCTS, SCQS, THTS and THQS shell elements. The default value is 10^{-12}.

[5] Rotation about the axis perpendicular to the element plane.

DLOA

Specify load sum computation in a time-domain dynamic forced-response analysis.

When this card is present, the sum of loads is computed for each time step and stored to CSV format. A separate file is written for each load history, marked with the load history id (IDLOHI on LOHI card).

The dynamic load-sum calculation has two inaccuracies.

- Load absorbed by fixations (or prescribed) degrees of freedom on non-structural elements is included in the load sum. For static load sums this contribution is removed. See A.3 for details.
- Load on hinged degrees of freedom (relevant for the two-noded beam element) is not included.

If the card is present in combination with the ESLE command additional CSV files are stored containing the equivalent static load inertia ($-M\ddot{x}$) and viscous ($-C\dot{x}$) contributions for the final load cycle.

The CSV output is a representation of the RDYNLAOD SIF card specification.

1——5	6——-10	11————20	21————30	31————40
DLOA				

41————50	51————60	61————70	71————80

DREA

Specify reaction force sum computation in a dynamic forced-response analysis (time domain or frequency domain).

When this card is present, the sum of reaction forces in the given set of nodes is computed for each time step and stored to CSV format. A separate file is written for each load history, marked with the load history id (IDLOHI on LOHI card). See Appendix B.3 for a definition of the reaction forces computed.

1——5	6——-10	11————20	21————30	31————40
DREA	OPT			
	= 0	O_x	O_y	O_z
	= 1	O_x	O_y	O_z

41————50	51————60	61————70	71————80
id1	id2	...	idN

OPT: Option for specifying nodes to include in the reaction sum.

=0 Compute the sum of reaction forces in nodes with support, including spring-to-ground elements (default). A node is interpreted as a support node if: 1) at least one degree of freedom is fixed (at zero or non-zero displacement) or 2) the node is connected to a spring-to-ground element.

=1 Compute the sum of reaction forces in a user-defined set of nodes. Node ids (external) are specified by id1, id2, . . . idN. Continuation records may be used to specify as many node ids as desired. Node ids continue in column 11 on continuation records.

O_x, O_y, O_z: Reaction force sum centre (x-, y- and z-component). Moments are computed around this point (see Appendix B.3). If omitted, the global origin (0, 0, 0) is used.

id1, id2, . . . , idN: Applies when OPT=1. List of nodes to include in the reaction force sum, given by external node ids.

DRED

Specify that static condensation is used to reduce the matrices and load vectors for dynamic analysis.

When this card is present the non-super degrees of freedom are in static equilibrium and are eliminated from the dynamical system. The dynamic analysis is for the super degrees of freedom.

1——-5	6——-10	11——-15	16——-20	21——-25	26——-30	31——-35	36——-40
DRED							

41——-45	46——-50	51————60	61————70	71————80

The card has no parameters. If the card is not present the super degrees of freedom are ignored.

DYMA

Control for generation of element matrices

1——-5	6——-10	11——-15	16——-20	21——-25	26——-30	31——-35	36——-40
DYMA		IDAM					

41——-45	46——-50	51————60	61————70	71————80

IDAM: Parameter controlling the generation of all element damping matrices associated with axial damper elements, damper-to-ground elements, general two-noded damper element, and matrix elements.

=0 No generation of element damping matrices (default)

=1 Generation of all element damping matrices.

ERST

Selection of element results sets.

1——-5	6——-10	11————20	21————30	31————40
ERST				ES1

41————50	51————60	61————70	71————80
ES2	...		ESN

ES1: First selected element set number.

ES2: Second selected element set number.

ESN: Last selected element set number.

The ERST command can be used to select any subset of the element sets defined on the Input Interface file. If the ERST command is present element results for the selected sets only will be stored to the Results Interface File.

Use of the ERST command is not currently compatible with postprocessing in Xtract. There is a limitation in the current version of Xtract (5.2-10) which causes only limited postprocessing capabilities if the Results Interface File does not contain results for all elements. This limitation will be fixed in a future version of Xtract.

Use of the ERST command is compatible with postprocessing in Stofat and Framework.

EIGA

Parameters for the implicitly restarted Lanczos method. Required for free vibration and mode superposition analyses.

1——-5	6——-10	11——-15	16——-20	21——-25	26——-30	31——-35	36——-40
EIGA	ENR		ITMAX	NLVEC	EPS		

41——-45	46——-50	51————60	61————70	71————80
		ELOW	EHIGH	SHIFT

ENR: The number of eigenpairs to be calculated.

ITMAX: The maximum number of Lanczos iterations.

NLVEC: The number of vectors in a Lanczos decomposition. If n is the number of degrees of freedom in the eigensystem, then the requirement is $n \geq$ NLVEC $>$ ENR.

The default value in Sestra 10.12.0 is NLVEC $= 2 \cdot$ ENR. Note that the program terminates with an error if NLVEC $> n$, and thus NLVEC must be specified if $2 \cdot$ ENR $> n$.

EPS: Convergence criterion (largest allowed relative error in the eigenvalues). The default value is $10^{-}7$.

ELOW, EHIGH: Lower and upper limits of eigenvalue interval (specified in Hertz by the frequency). When an interval is specified, ENR is interpreted as the number of eigenvalues to be found in each iteration og the algorithm. A large value of ENR is recommended (100-300, or larger, depending on the model size and the available memory of the computer). The algorithm will check that all eigenvalues in the specified interval are found.

SHIFT: Frequency shift value. Sestra will calculate the ENR eigenpairs (v_i, λ_i) with associated frequency $\sqrt{\lambda_i}/(2\pi)$ closest to SHIFT. The default shift value is 0.

Note that EIGA must be present in order to perform free vibration analysis with ANTYP=2.

ELCH

Parameters for element check.

1——-5	6——-10	11——-15	16——-20	21——-25	26——-30	31——-35	36——-40
ELCH	CNTERR						

CNTERR: Stop or continue analysis on element shape error.

=0 Stop analysis on shape error (default)

=1 Continue analysis on shape error

ELOP

Element options for shell elements.

1——-5	6——-10	11——-15	16——-20	21——-25	26——-30	31——-35	36——-40
ELOP				WCOR	THCK		

WCOR: Warp correction selection for first-order 4-noded shell elements.

=0 No warp correction.

=1 Warp correction (default).

THCK: Override all first-order thin shell elements FTRS and FQUS by first-order thick shell elements THTS and THQS.

=0 No element override.

=1 First-order thick shell elements override first-order thin shell elements. All FTRS elements are replaced by THTS; all FQUS elements are replaced by THQS.

ELIM

Selection of elements to ignore in analysis by external element number. Mass, stiffness, and loads for listed elements will be ignored.

1——-5	6——-10	11————20	21————30	31————40
ELIM			OPT	
			= 1	EL1
			= 2	EL1

41————50	51————60	61————70	71————80
EL2	...		ELN
ELSTEP	ELN		

OPT: Element selection option

=1 External element numbers specified as EL1, EL2, EL3, . . . ELN. In this case continuation records may be used to specify as many elements as desired. Element numbers continue in column 11 on continuation records.

=2 External element numbers defined by EL1:ELSTEP:ELN. For example, EL1=4, ELSTEP=2, ELN=10 gives element numbers 4, 6, 8, 10.

EL1: First selected external element number.

ELSTEP: Step in external element number.

ELN: Last selected external element number.

ESLE

Control of export of equivalent static loads to separate file. If present in a time-domain dynamic analysis Sestra will export equivalent static loads to "SestraESL.srs" in the working directory. This file is to be imported in a subsequent static analysis using the ESLI command.

If this command is used together with stress stiffening, data for geometric stiffness is exported to "SestraESL_PD.srs," and is imported in the subsequent static analysis.

Equivalent static loads at time t are defined as

$$R_{\text{esl}}(t) = R(t) - M\ddot{x}(t) - C\dot{x}(t). \tag{4.1}$$

1——5	6——-10	11——-15	16——-20	21——-25	26——-30	31——-35	36——-40
ESLE	OPT	ESLF	ESLLC	RES	FORM	FIX	

41——-45	46——-50	51————60	61————70	71————80

OPT: Option for specification of equivalent static load times.

=3 The points in time associated with the input loads on TILO are used (default)

=11 Options are specified individually for each load history using ESLU commands. In this case the ESLF and ESLLC fields in the ESLE command are ignored.

ESLF: A factor to be multiplied with the equivalent static loads so that the stored loads are $\text{ESLF} \cdot R_{\text{esl}}$. Default value ESLF = 1.0. Ignored if OPT=11.

ESLLC: A load combination to add to all equivalent static loads so that the stored loads are $\text{ESLF} \cdot R_{\text{esl}} + \text{ESLLC}$ (optional). Ignored if OPT=11.

RES: Storage of Results Interface File.

=0 Results Interface File is not stored (default).

=1 Results Interface File is stored.

FORM: Format of the file "SestraESL.srs." Note that the same option must be used when importing loads from this file with the ESLI command.

=0 Binary format (default)

=1 Text format

FIX: Identification number of an element set for which all nodes are fixated in the ESL export step. This field is typically used to suppress pile elements in dynamic step of a pile-soil analysis.

ESLI

Control of import of equivalent static loads from separate file. If present, Sestra will import equivalent static loads from a file "SestraESL.srs" generated from a previous time-domain dynamic analysis using the ESLE command.

If stress stiffening was specified in the dynamic analysis data for geometric stiffness is imported from "SestraESL_PD.srs" and automatically applied in the analysis. The STIF parameter on the CMAS command is ignored if ESLI is specified.

1——5	6——-10	11——-15	16——-20	21——-25	26——-30	31——-35	36——-40
ESLI					FORM		

41——-45	46——-50	51————60	61————70	71————80

FORM: Format of the file "SestraESL.srs." Must be set to the value of FORM on the ESLE command used to generate this file.

=0 Binary format (default)

=1 Text format

ESLU

Control of export of equivalent static loads to separate file. If present in a time-domain dynamic analysis Sestra will export equivalent static loads to “SestraESL.srs” in the working directory. This file is to be imported in a subsequent static analysis using the ESLI command.

Equivalent static loads at time t are defined as

$$R_{\text{esl}}(t) = R(t) - M\ddot{x}(t) - C\dot{x}(t). \tag{4.2}$$

1——-5	6——-10	11——-15	16——-20	21——-25	26——-30	31——-35	36——-40
ESLU	IDESLU		ESLLC			OPT	

41——-45	46——-50	51————60	61————70	71————80
			ESLF1	ESLF2

OPT: Option for specification of equivalent static load times.

=6 Sestra calculates and uses the points in time corresponding to minimum base shear.

=7 Sestra calculates and uses the points in time corresponding to maximum base shear.

=8 Sestra calculates and uses the points in time corresponding to minimum overturning moment.

=9 Sestra calculates and uses the points in time corresponding to maximum overturning moment.

ESLLC: Reference to a (static) load combination R_{ESLLC} to add to the dynamic load histories used in the analysis so that

$$R(t)_{\text{total}} = R_{\text{ESLLC}} + R(t)$$

(optional).

ESLF1, ESLF2: Factors to be multiplied with the equivalent static loads so that the stored loads are

$$R(t)_{\text{total}} = R_{\text{ESLLC}} + \text{ESLF1} \cdot (R(t) - \text{ESLF2} \cdot (M\ddot{x}(t) + C\dot{x}(t))).$$

Default values are ESLF1 = ESLF2 = 1.0.

FRLO

Selection of frequencies for forced-response analysis in the frequency domain.

1——5	6——-10	11————20	21————30	31————40
FRLO	IDFRLO		OPT	
			= 1	F1
			= 2	F1

41————50	51————60	61————70	71————80
F2	...		FN
FSTEP	FN		

IDFRLO: Integer ID of this frequency set.

OPT: Frequency selection option

=1 Frequencies specified as F1, F2, F3, . . . FN. In this case continuation records may be used to specify as many frequencies as desired. Frequencies continue in column 11 on continuation records.

=2 Frequencies defined by F1:FSTEP:FN. For example, F1=0.1, FSTEP=0.05, FN=0.2 gives frequencies 0.1,0.15,0.2.

F1: First selected frequency.

FSTEP: Step in frequency.

FN: Last selected frequency.

Note that if F<n> is smaller than $1.e-16$ it will be considered by Sestra as zero frequency. The analysis of the associated load case is considered by Sestra as linear static.

FRQL

Selection of load cases and frequencies for forced-response analysis in the frequency domain.

1——5	6——-10	11————20	21————30	31————40
FRQL		F1	FSTEP	FN

41————50	51————60	61————70	71————80
LC1	LSTEP	LCN	

F1: First selected frequency.

FSTEP: Step in frequency.

FN: Last selected frequency.

LC1: First selected load case.

LSTEP: Step in load case.

LCN: Last selected load case.

This command specifies combinations of frequencies and load cases as a tensor product such that if n_f is the number of specified frequencies and n_l is the number of specified load cases then all $n_f n_l$ combinations will be generated.

FRSP

Control command for the forced-response analysis algorithm.

1——-5	6——-10	11——-15	16——-20	21——-25	26——-30	31——-35	36——-40
FRSP	CALTYP						
	= 1		SCORR				
	= 2						
	= 3		SCORR	n	BCDYN		
	= 4				BCDYN		

41——-45	46——-50	51————60	61————70	71————80
RESVA				
RESVA		β	Θ	h

CALTYP: Type of forced response analysis.

=1 Frequency-domain forced response analysis (modal superposition)

=2 Frequency-domain forced response analysis (direct method)

=3 Time-domain forced response analysis (modal superposition)

=4 Time-domain forced response analysis (direct method)

Any other value: The analysis is routed to Sestra 8.8-2

SCORR: Static back substitution for mode superposition analysis.

=0 Do not perform static back substitution.

=1 Perform static back substitution.

RESVA: Storing of results for time-domain analysis for OPT=3 or OPT=4.

=0 Velocities and accelerations are not stored (default)

=1 Velocities and accelerations are stored on result file

β: Algorithm parameter for direct time integration (OPT=4). Default: $\beta = 1/6$

Θ: Algorithm parameter for direct time integration (OPT=4). Default: $\Theta = 1.420815$

n: Time-step size parameter for time integration in time-domain modal superposition analysis (OPT=3). n timesteps is used for each period T_i, so that the time-step size $h_i = T_i/n$ is different for each of the uncoupled equations. The period is $T_i = 2\pi/\sqrt{\lambda_i}$ where λ_i is the eigenvalue associated with eigenvector number i. If not specified, $n = 100$ is used.

BCDYN: Toggle inertia and viscous load effects resulting from time-varying boundary conditions. See Section B.1.5. Also note that BCDYN has an effect on velocity and acceleration of prescribed nodes.

=0 Inertia and viscous effects from boundary conditions are included (all terms of (B.32) are included) (default).

=1 Inertia effects from boundary conditions are included (the term $-C_{ib}\dot{x}_b$ is not included; the term $-M_{ib}\ddot{x}_b$ is included). Prescribed nodes will have zero velocity.

=2 Viscous effects from boundary conditions are included (the term $-C_{ib}\dot{x}_b$ is included; the term $-M_{ib}\ddot{x}_b$ is not included). Prescribed nodes will have zero acceleration.

=3 Inertia and viscous effects from boundary conditions are not included (the terms $-M_{ib}\ddot{x}_b$ and $-C_{ib}\dot{x}_b$ in equation (B.32) are not included). Prescribed nodes will have zero velocity and acceleration.

Note that the term $K_{ib}x_b$ is always included.

h: Time-step size parameter for direct time integration (OPT=4). If the Generalized-α method is used together with adaptive time-stepping (see the specification of the GENA card), h is used to specify the size of the initial time step.

If the FRSP command is present with CALTYP=1 or CALTYP=3 (modal superposition), an EIGA command is necessary in order to solve the generalized eigenvalue problem associated with the modal superposition analysis. All computed mode shapes will be included in the modal superposition analysis.

If the FRSP command is present with CALTYP=4 (direct time integration) and the GENA command is not present, the Collocation scheme of Hilber and Hughes [11] is employed with the Wilson Θ-method as a special case for $\beta = 1/6$. For $\Theta = 1$ the Newmark method family is obtained [9]. The parameters β and Θ are further discussed in Appendix B.

GENA

Specification of the Generalized-α time-integration method [3] used in time-domain forced response analysis / direct time integration. Note that the FRSP command must also be present.

If the GENA card is not present, the Collocation scheme of Hilber and Hughes [11] is used with the Wilson Θ-method [12] as a special case if $\beta = 1/6$ on the FRSP command ($\beta = 1/6$ is the default value). If the GENA card is present, the Generalized-α method is used; in this latter case the parameters β and Θ from the FRSP are not relevant.

1——-5	6——-10	11————20	21————30	31————40
GENA	ρ	ADAPTOL	PRINT	

ρ: Algorithm parameter to control the numerical dissipation, $0 \leq \rho \leq 1$. The method exhibits maximum numerical dissipation for $\rho = 0$ and no numerical dissipation for $\rho = 1$. Default value: $\rho = 0.8$.

ADAPTOL: If not specified, a constant time step is used with value equal to h from the FRSP command. If specified, Sestra employs an adaptive time-stepping algorithm so that a variable time step is used. In this case, an initial time-step suggestion is required from the user, and is given as h on the FRSP command. All time steps are chosen so that an estimate for the local truncation error is smaller than ADAPTOL.

PRINT: Print of time-step and error-estimate history. Only relevant in combination with adaptive time stepping.

=0 No print (default).

=1 Data for load history l is written to Sestra_adaptive_time_step_lohi_<l>.dat.

The precise meaning of the algorithm parameters ρ and ADAPTOL, and a description of the adaptive time-stepping algorithm can be found in Appendix B.

GSTF

Selection of load case used for generation of the geometric stiffness matrix for use in a stress stiffening analysis or linearized buckling analysis.

1——-5	6——-10	11——-15	16——-20	21——-25	26——-30	31——-35	36——-40
GSTF	LOAD						

41——-45	46——-50	51————60	61————70	71————80

LOAD: Load case or load combination for generation of geometric stiffness matrix.

=0 The last load case (or, if load combinations are specifed, the load combination with the largest number) is used for the generation of element geometric stiffness matrices. (Default)

=N Load case N (or, if load combinations are specifed, load combination N) is used for the generation of element geometric stiffness matrices.

HIST

Specification of a time history (to be associated with a single load case)

1——-5	6——-10	11————20	21————30	31————40
HIST	IDHIST	IDTILO	OPT	
			= 1	H1
			= 2	FUNC

41————50	51————60	61————70	71————80
H2	...		HN
NSTEP	A	B	C

(new line, OPT=2)

1————10	11————20	21————30	31————40
	D	E	F

IDHIST: Integer ID of this time history.

IDTILO: The ID of the TILO card associated with this time history.

OPT: Time history option.

=1 dependence on time specified as scalar factors associated with each time instant specified on the referenced TILO command.

=2 dependence on time specified as an explicit function

FUNC: Specification of time variation $\tau(t)$.

=1 $\tau(t) = A\sin(Bt + C) + D$

=2 $\tau(t) = A\exp(Bt + C) + D$

=3 $\tau(t) = A\exp(Bt + C)\sin(Dt + E) + F$

NSTEP: Number of time steps of the associated TILO command for which the function chosen by FUNC is followed. Outside this region $\tau(t) = 0$.

A,B,C,D,E,F: Coefficients of $\tau(t)$. Set equal to zero if not specified.

H<n>: Load variation factor for the n'th point in time as specified on TILO. The number of specified factors must equal the number of points in time specified on TILO.

HYDM

Specification of hydrodynamic added mass (applicable only for first order shell elements).

1——-5	6——-10	11————20	21————30	31————40
HYDM	IHYD	WLEV	WRHO	

41————50	51————60	61————70	71————80
			EPSW

(new line, (optional)

1————10	11————20	21————30	31————40
	LCKEY		

IHYD: Main control parameter for calculation of hydrodynamic added mass.

=0 No computation of hydrodynamic added mass (default)

=1 Computation of hydrodynamic added mass by a boundary integral equation method.

WLEV: Z-coordinate of the water level (global coordinate system).

WRHO: Water density (mass/volume).

EPSW: Nodes above WLEV-EPSW·WDEP are excluded from the calculation. WDEP is the depth of the structure in water and is automatically determined by Sestra. The reason for this is that nodes close to the water level can cause numerical problems during calculation. Their contribution to the mass is small. (Default value 0.001.)

LCKEY: Load case as specified on dummy hydrodynamic load cards. (Default value 1).

IDTY

Introduction of a set of command pertaining to a particular superelement.

1——-5	6——-10	11——-15	16——-20	21——-25	26——-30	31——-35	36——-40
IDTY	SELTY						

SELTY: Superelement number.

INAM

Prefix for input interface file (<prefix>T<n>.FEM).

1——-5	6————————————————————————————————80
INAM	PREFIX

PREFIX: Interface file prefix. The file name will be on the form <PREFIX>T<n>.FEM.

If INAM is omitted or the prefix field is empty, PREFIX is defaulted to an empty string.

Note that when the analysis type is postprocessing (ANTYP=11 on the CMAS command) INAM refers to the Results Interface File <PREFIX>R<n>.SIN.

INDT

Selection of subset of specified times for output (time-domain forced response). Specify the indices (starting from 1) of the time instants specified on TOUT (or, if not present, on TILO) for which results are computed.

Note that the INDT command cannot be used in combination with the PERI command.

1——-5	6——-10	11——-15	16——-20	21——-25	26——-30	31——-35	36——-40
INDT	IDINDT		OPT				
			=1	I1	I2	...	IN
			=2	I1	ISTEP	IN	

IDINDT: Integer ID of this index definition.

OPT: Index definition option

=1 Indices specified as I1, I2, I3, . . . IN. In this case continuation records may be used to specify as many load cases as desired. Load cases continue in column 6 on continuation records.

=2 Indices defined by I1:ISTEP:IN. For example, I1=4, ISTEP=2, IN=10 gives indices 4, 6, 8, 10.

I1: First selected index.

ISTEP: Step in index.

IN: Last selected index.

INTP

Specify the interior-point solver for contact or tension-compression analysis.

1——-5	6——-10	11————20	21————30	31————40
INTP	ITER	TOL		UPDT

ITER: The maximum number of iterations in the algorithm (default is ITER=100)

TOL: Convergence criterion (default is TOL=10^{-10})

UPDT: Type of matrix factor update algorithm used during interior-point iteration. In rare cases the iterative algorithm may produce a system which is numerically sensitive. In such cases the interior-point algorithm may not converge to within the desired tolerance and Sestra will issue a warning or an error. One of the non-default matrix factor update algorithms may then be used. Note that the full-rank update algorithm (UPDT=2) is significantly slower than the low-rank update algorithms.

=0 Fast low-rank update algorithm (default).

=1 Fast low-rank update algorithm (alternative).

=2 Full-rank update algorithm.

INTP may be specified without arguments, in which case the default values will be employed.

ITOP

ID of superelement on the Input Interface File.

1——5	6——-10	11——-15	16——-20	21——-25	26——-30	31——-35	36——-40
ITOP	ITYP1						

ITYP1: Superelement number.

LCAS

Selection of load cases for analysis. Only applicable to tension-compression and contact-gap analyses.

1——5	6——-10	11————20	21————30	31————40
LCAS			OPT	
			= 1	LC1
			= 2	LC1

41————50	51————60	61————70	71————80
LC2	...		LCN
LSTEP	LCN		

OPT: Load case selection option

=1 Load cases specified as LC1, LC2, LC3, . . . LCN. In this case continuation records may be used to specify as many load cases as desired. Load cases continue in column 11 on continuation records.

=2 Load cases defined by LC1:LSTEP:LCN. For example, LC1=4, LSTEP=2, LCN=10 gives load cases 4, 6, 8, 10.

LC1: First selected load case.

LSTEP: Step in load case number.

LCN: Last selected load case.

The LCAS command can be used to select any subset of load cases present on the Input Interface File or Loads Interface File for analysis. The command is only applicable to tension-compression and contact-gap analyses.

LCOM

Selection of load cases.

1——5	6——10	11————20	21————30	31————40
LCOM	IDLCOM	IDSCAL	OPT	
			= 1	LC1
			= 2	LC1

41————50	51————60	61————70	71————80
LC2	...		LCN
LSTEP	LCN		

IDLCOM: Integer ID of this load case selection.

IDSCAL: Optional reference to scaling factors or upramping associated with the load case selection.

OPT: Load case selection option

=1 Load cases specified as LC1, LC2, LC3, . . . LCN. In this case continuation records may be used to specify as many load cases as desired. Load cases continue in column 11 on continuation records.

=2 Load cases defined by LC1:LSTEP:LCN. For example, LC1=4, LSTEP=2, LCN=10 gives load cases 4, 6, 8, 10.

LC1: First selected load case.

LSTEP: Step in load case number.

LCN: Last selected load case.

LNAM

Prefix for loads interface file.

1——5	6——*	*————80
LNAM	PREFIX	FORMAT

PREFIX: Optional prefix. The file name will be on the form <PREFIX>L<n>.FEM.

FORMAT: Optional file format.

any of {SIF, FORMATTED, F}: Formatted sequential format

any of {SIU, UNFORMATTED, U}: Unformatted (binary) sequential format

Note that if PREFIX field is left empty, the FORMAT specification must not start before the 8th column.

LOHC

Selection and combination of load histories for time-domain dynamic analysis.

The LOHC command is used to combine (add) multiple load histories defined with the LOHI command together. When at least one LOHC command is present, LOHC replaces the LOHI command as the top-level command for load-history specification. In this case there will be a one-one correspondence between the LOHC commands and the calculated time series results.

1——5	6——10	11————20	21————30	31————40
LOHC	IDLOHC	INCONO	IDTOUT	IDINDT
			OPT	
			= 1	LOHI1
			= 2	LOHI1

41————50	51————60	61————70	71————80
IDSEAS	IDWIND		IDESLU
LOHI2	...		LOHIN
LOHISTEP	LOHIN		

IDLOHC: Integer ID of this load history combination.

INCONO: The ID of the initial condition associated with this load history combination. This number refers to the ID of BNINCO cards on the model file. If not specified, zero initial conditions are assumed.

IDTOUT: The ID of the TOUT card associated with this load history combination.

IDINDT: The ID of the INDT card associated with this load history combination. If present, a subset of specified times on TOUT (or TILO) will be used for output.

IDSEAS: The ID of the SEAS card associated with this load history combination. If present, sea-state information will be included in the result file.

IDWIND: The ID of the WIND card associated with this load history combination. If present, wind condition information will be included in the result file.

OPT: Load history selection option

=1 Load histories specified as LOHI1, LOHI2, LOHI3, . . . LOHIN. In this case continuation records may be used to specify as many load histories as desired. Load histories continue in column 11 on continuation records.

=2 Load histories defined by LOHI1:LOHISTEP:LOHIN. For example, LOHI1=4, LOHISTEP=2, LOHIN=10 gives load histories 4, 6, 8, 10.

LOHI1: First selected load history.

LOHISTEP: Step in load history number.

LOHIN: Last selected load history.

Note in particular the following.

- References to INCONO (initial condition number), TOUT (output time specification), INDT (output time index selection), SEAS (sea-state specification) and WIND (wind-condition specification) from LOHC override any references to these commands from any LOHI commands.
- A TOUT command with reference from LOHC is mandatory when LOHC is used. The TOUT command is used to specify points in time for which to store results.
- When LOHC is used in combination with periodic load histories (specified by referensing a PERI command from each LOHI command), all included load histories must have identical period. Furthermore the CYCLEMAX, STEADYTOL and CYCLESTOR parameters on the PERI command are aggregated over all included load histories such that the maximum CYCLEMAX, the minimum STEADYTOL and the maximum CYCLESTOR values are used.

LOHI

Main command for forced response loading and output.

1——-5	6——-10	11——-15	16——-20	21——-25	26——-30	31——-35	36——-40
LOHI	IDLOHI	INCONO	OPT				
			=1	IDLCOM	IDHIST	IDTOUT	IDINDT
			=2,12,13	IDLCOM	IDTILO	IDTOUT	IDINDT
			=6,7	IDLCOM	IDFRLO		

41——-45	46——-50	51————60	61————70	71————80
	IDSEAS	IDWIND	IDPERI	IDESLU
	IDSEAS	IDWIND	IDPERI	IDESLU
	IDSEAS			

IDLOHI: Integer ID of this load history.

INCONO: The ID of the initial condition associated with this load history. This number refers to the ID of BNINCO cards on the model file. If not specified, zero initial conditions are assumed.

OPT: Load history option

=1 Load history $f(t)$ is specified as $f(t) = l\tau(t)$ where l corresponds to a load case specified on LCOM and τ depends only on time and is specified on HIST.

=2 Load history $f(t)$ is specified with N load cases at N discrete points in time through LCOM and TILO.

=6 Definition of harmonic loading. Each load case specified in the referenced LCOM command vibrates with a frequency defined in the referenced FRLO command.

=7 As OPT=6. OPT=7 is used to indicate that the load is generated by certain preprosessors (Wajac/Wadam).

=12 As OPT = 2 but static analysis in Sestra. Deterministic and time-domain simulation analyses in Wajac will produce S-files containing LOHI with OPT = 12. IDTOUT and IDINDT cannot be defined for OPT = 12.

=13 Definition of input for gust-wind-induced fatigue analysis. Wajac will for gust-wind fatigue analysis make an S-file with input for fatigue calculation. The LCOM command will refer to three loadcases, one for the wind direction and the others for wind components perpendicular to the wind direction. SEAS and WIND cards are mandatory for gust wind analyses. The fatigue calculation is performed in Framework. Static analysis in Sestra.

IDLCOM: The ID of the LCOM card associated with this load history.

IDHIST: The ID of the HIST card associated with this load history (if OPT=1).

IDTILO: The ID of the TILO card associated with this load history (if OPT= 2, 12 or 13).

IDFRLO: The ID of the FRLO card associated with this load history (if OPT=6 or 7).

IDTOUT: The ID of the TOUT card associated with this load history.

IDINDT: The ID of the INDT card associated with this load history. If present, a subset of specified times on TOUT (or TILO) will be used for output.

IDSEAS: The ID of the SEAS card associated with this load history. If present, sea-state information will be included in the result file.

IDWIND: The ID of the WIND card associated with this load history. If present, wind condition information will be included in the result file.

IDPERI: The ID of the PERI card associated with this load history. If present, the specified load is considered as the load for one period; the load cycle is then repeated as specified on the PERI card.

MATR

Matrix output.

The command is used for matrix and load vector exchange between analysis programs through the Matrix Interface File (M-file). The command may only be given for the whole structure, and not for individual superelements. The global matrices and vectors are reduced to the super degrees of freedom.

Super degrees of freedom must be set in the preprocessor and enabled in Sestra using the DRED command.

The MATR command may be combined with the RCMS (the Component Mode Synthesis method) command in order to include contributions from internal mode shapes (generalized degrees of freedom) to the exported matrices and vectors.

1——5	6——-10	11——-15	16——-20	21——-25	26——-30	31——-35	36——-40
MATR		STIFMT	DAMPMT	MASSMT	LOADVC		

41——-45	46——-50	51————60	61————70	71————80

STIFMAT: Parameter for storage of stiffness matrix.

=0 No storing of stiffness matrix.

=1 Storing of stiffness matrix.

DAMPMT: Parameter for storage of damping matrix.

=0 No storing of damping matrix.

=1 Storing of damping matrix.

MASSMAT: Parameter for storage of mass matrix.

=0 No storing of mass matrix.

=1 Storing of mass matrix.

LOADVC: Parameter for storage of load vectors.

=0 No storing of load vectors.

=1 Storing of load vectors.

The name and format of the Matrix Interface File (M-file) may be specified in an MNAM command.

If a local coordinate system has been given for supernodes then the associated matrix or vector data will be exported in this transformed coordinate system.

MDAF

General modal damping. Relevant for modal modal superposition dynamic analysis (time-domain or frequency-domain).

1——5	6——-10	11————20	21————30	31————40
MDAF	FRNO	CD	FRNO	CD

41————50	51————60	61————70	71————80
FRNO	CD	FRNO	CD

FRNO: Eigenvector (mode shape) number up to which (inclusive) modal damping has the following CD value. If the number is negative the following CD value is used for all eigenvectors.

CD: Fraction of critical damping (see Section 2.8.4). CD $= f_i$ in the uncoupled equation $\ddot{y}_i + 2f_i\sqrt{\lambda_i}\dot{y} + \lambda_i y_i = r_i$

MDAP

Proportional damping. Relevant for dynamic forced-response analysis (time domain or frequency domain).

1——5	6——10	11————20	21————30	31————40
MDAP		α_1		α_2

The MDAP card specifies the coefficients α_1 and α_2 such that the damping matrix in a forced response analysis is given as $C = \alpha_1 M + \alpha_2 K$ for mass matrix M and stiffness matrix K. The MDAP card must be present in a time-domain forced response analysis. If the values of α_1 and α_2 are not set, default values of zero are used.

MDAS

Structural damping. Relevant for dynamic forced-response analysis in the frequency domain. In a modal superposition analysis the stuctural damping can be specified as modal damping.

1——5	6——10	11————20	21————30	31————40
MDAS	FRNO	η	FRNO	η

41————50	51————60	61————70	71————80
FRNO	η	FRNO	η

In a direct analysis:

FRNO: Not relevant.

η: Coefficient of structural damping (see Section 2.8.4). Note that in this case only the first occurence of η on the MDAS command is used.

In a modal superposition analysis:

FRNO: Eigenvector (mode shape) number up to which (inclusive) structural damping has the following η value.

η: Coefficient of structural damping (see Section 2.8.4).

MNAM

Prefix and format for Matrix Interface File, used together with the MATR command.

1——5	6——-*	*————80
MNAM	PREFIX	FORMAT

PREFIX: Optional prefix. The file name of the Matrix Interface File will be on the form <PREFIX>R<n>.SIN or <PREFIX>R<n>.SIF.

FORMAT: Output file format.

any of {SIN, NORSAM, N}: Binary .SIN file format (default)

any of {SIF, FORMATTED, F}: Formatted .SIF file format.

Note that the unformatted sequential format .SIU is not available.

If the MNAM command is included but with an empty PREFIX field, no prefix will be used for result files. In this case, the FORMAT specification must not start before the 8th column.

PARA

Multicore settings. If this card is not present Sestra may use all available cores.

1——-5	6——-10	11——-15	16——-20	21——-25	26——-30	31——-35	36——-40
PARA	OPT						

41——-45	46——-50	51————60	61————70	71————80

OPT: Multicore option.

=0 Sestra may use all available cores (default).

=1 Solver is restricted to a single core.

PERI

Periodic load specification. Optional for time-domain forced response analysis.

1——-5	6——-10	11————20	21————30	31————40
PERI	IDPERI	CYCLEMAX	PERIOD	STEADYTOL

41————50	51————60	61————70	71————80
CYCLESTOR			

IDPERI: Integer ID of this periodic load specification (referenced from LOHI).

CYCLEMAX: The maximum number of periods for which the specified periodic load for this load history is to be repeated.

PERIOD: Length of period. If not specified, the period is inferred from the values given on the TILO record. If the load at the final time of the period is not specified explicitly, the period must be provided here.

STEADYTOL: Tolerance parameter for steady-state detection. A blank field implies that steady-state detection is not enabled.

CYCLESTOR: The number of periods for which to write results, counting backwards from the final time as determined by either steady-state detection or CYCLEMAX. If not specified, the value CYCLESTOR$= 1$ will be used. Note that if CYCLESTOR is larger than the number of calculated result cycles, results are written for all cycles.

RCMS

Control data for component mode synthesis reduction.

1——5	6——-10	11——-15	16——-20	21——-25	26——-30	31——-35	36——-40
RCMS		FNR		ITMAX	NLVEC	EPS	

41——-45	46——-50	51————60	61————70	71————80
		FS1	FS2	SHIFT

FNR: Number of mode shapes, i.e. eigenvectors, used in the component mode synthesis reduction.

ITMAX The maximum number of Lanczos iterations.

NLVEC The number of vectors in a Lanczos decomposition. If n is the number of degrees of freedom in the eigensystem, then the requirement is $n \geq$ NLVEC $>$ FNR.

The default value in Sestra 10.12.0 is NLVEC $= 2 \cdot$ FNR. Note that the program terminates with an error if NLVEC $> n$, and thus NLVEC must be specified if $2 \cdot$ FNR $> n$.

EPS: Convergence criterion (largest allowed relative error in the eigenvalues). The default value is 10^-7.

FS1, FS2: Lower and upper frequency limits of the mode shapes used in the reduction. If the numbers are negative their absolute value refers to the mode shape numbers.

SHIFT: Frequency shift value. Sestra will calculate the FNR eigenpairs (v_i, λ_i) with associated frequency $\sqrt{\lambda_i}/(2\pi)$ closest to SHIFT. The default shift value is 0.

The RCMS command is used with the DRED command.

The input to the eigenvalue solver used for component mode synthesis is taken from the RCMS command. Specifying FNR sets up the eigenvalue solver to compute the FNR mode shapes corresponding to the lowest eigenfrequencies. If, in addition, FS1 and FS2 are negative numbers then the eigenmodes from -FS1 to -FS2 are used in the component mode synthesis analysis.

Specifying non-negative FS1 and FS2 will use the eigenvalue solver to compute all eigenvalues in the interval given by FS1 and FS2. The number FNR is used to indicate the number of eigenfrequencies and eigenmodes found in each step.

Note that FNR is used as ENR, and that positive FS1 and FS2 are used as ELOW and EHIGH in the EIGA command. The remaining parameters are algorithm parametes for the eigenvalue solver, identical to the parameters on the EIGA command.

RNAM

Prefix and format for Results Interface File.

1——5	6——-*	*————80
RNAM	PREFIX	FORMAT

PREFIX: Optional prefix. The file name of the Results Interface File will be on the form <PREFIX>R<n>.SIN or <PREFIX>R<n>.SIF. For writing of result files other than the Results Interface File (for example a .CSV-file with mass sum), the value of PREFIX *excluding any relative path specified* is used as an additional file-name prefix.

FORMAT: Output file format.

any of {SIN, NORSAM, N}: Binary .SIN file format (default)

any of {SIF, FORMATTED, F}: Formatted .SIF file format.

NONE: No Results Interface File is written.

Note that the unformatted sequential format .SIU is not available.

If the RNAM command is included but with an empty PREFIX field, no prefix will be used for result files. In this case, the FORMAT specification must not start before the 8th column.

RSEL

Selection of data types written to the result file.

1——-5	6——-10	11——-15	16——-20	21——-25	26——-30	31——-35	36——-40
RSEL	ISEL1		ISEL3	ISEL4			ISEL7

ISEL1: Specifies whether inertia loads are written to the Results Interface File as distributed loads (line loads) for two-noded beam elements.

=0 or -1: No loads are written (default).

=1: Inertia loads specified as nodal accelerations (BNACCLO), or inertia loads (BGRAV or BRIGAC) are stored on the Results Interface File as equivalent distributed loads (BELOAD1) for all two-noded beam elements.

ISEL3: Specifies whether element forces and moments are stored on the Results Interface File.

=-1: Element forces are not stored.

=0 or 1: Element forces are stored (default).

ISEL4: Specifies whether element stresses are stored on the Results Interface File.

=-1: Element stresses are not stored.

=0 or 1: Element stresses are stored (default).

ISEL7: Specifies whether loads are transferred to the Results Interface File.

=0 or -1: No loads are transferred (default)

=1: All loads are transferred from the Input Interface File and Loads Interface File to the Results Interface File.

It is important to note the following:

- For two-noded beam elements with distributed line load, transfer of loads to the Results Interface File is necessary for correct calculation of force and moment diagrams in postproecessors (ISEL7=1).
- If inertia loads are present, these loads must for the same reason be stored on the Results Interface File as equivalent distributed line loads (ISEL1=1).
- ISEL7=1 applies to all load cards specified on the Input Interface File and Loads Interface File. Thus if ISEL1=1 and ISEL7=1, the Results Interface File will contain a load specification inconsistent with the input because the inertia load specification is also transferred.
- There is a performance cost for writing loads to result files when running models with large external load files (L-files) since this requires copying the loads from the Loads Interface File to the Results Interface File file.
- The definition of the ISEL1 and ISEL7 options on RSEL differ somewhat between Sestra 8.8-2 and Sestra 10.12.0. See Appendix A.

SCAL

Scaling of load cases defined in LCOM commands. Only applicable in dynamic (time-domain or frequency-domain) analyses.

1——-5	6——-10	11————20	21————30	31————40
SCAL	IDSCAL		OPT	
			= 1	F1
			= 2	F
			= 3,4	
			= 23,24	F

41————50	51————60	61————70	71————80
F2	...		FN
t_{ramp}			
t_{ramp}			

IDSCAL: Integer ID of this scaling factors definition

OPT: Scaling factors definition option.

=1 Scaling factors specified as F1, F2, F3, . . . , FN. In this case continuation records may be used to specify as many scaling factors as needed. Scaling factors continue in column 11 on continuation records. Note that the number of scaling factors must be equal to the number of load cases specified on the associated LCOM command.

=2 A single scaling factor F is used for all load cases.

=3 Linear upramping of load history specified using load cases. Effectively the function

$$f(t) = \begin{cases} 1 - \frac{t_{\text{ramp}} - (t - t_{\text{init}})}{t_{\text{ramp}}} & t_{\text{init}} < t < t_{\text{ramp}} \\ 1 & t \geq t_{\text{init}} + t_{\text{ramp}} \end{cases} \tag{4.3}$$

is multiplied with the load function. Note that $f(t)$ is continuous and piecewise linear.

=4 Smooth upramping of load history specified using load cases. Define

$$h(x) = \begin{cases} e^{-1/x} & x > 0 \\ 0 & x \leq 0 \end{cases}. \tag{4.4}$$

The function

$$g(t) = \frac{h(\hat{t})}{h(\hat{t}) + h(1 - \hat{t})}, \tag{4.5}$$

where $\hat{t} = (t - t_{\text{init}})/t_{\text{ramp}}$, is multiplied with the load function. Note that $g(t)$ is a smooth (in fact $C^{\infty}(\mathbb{R})$) function.

=23 Combination of scaling factor and linear upramping: the function F$\cdot f(t)$ is multiplied with the load function.

=24 Combination of scaling factor and smooth upramping: the function F$\cdot g(t)$ is multiplied with the load function.

t_{ramp}: Specification of time for upramping.

The parameter t_{init} in the equations (4.3) and (4.4) above is the initial time for the time-domain analysis. This parameter is automatically inferred by Sestra from the TILO command.

SEAS

Sea-state specification for a load history. This command is typically created by a load generator (e.g. Sesam Wadam or Wajac), and the information is transferred by Sestra to the result file for use by post processors.

For a time-domain dynamic analysis, the wave-height field, the wave-direction field, and the wave-period field must be present (together with a DREA command) in order to compute response-amplitude operators.

For a frequency-domain dynamic analysis, only the wave direction field must be present (together with a DREA command) in order to compute response-amplitude operators. Additionally, the wave-height field may be present. If omitted, a wave height equal to two (i.e. a unity wave amplitude) is assumed.

1——5	6——10	11————20	21————30	31————40
SEAS	IDSEAS			

41————50	51————60	61————70	71————80
	DIR	HEIGHT	PERIOD

(new line)

1————10	11————20	21————30	31————40
	DEPTH		ZMUD

IDSEAS: Integer ID of this sea-state specification.

DIR: Wave direction (angle in degrees).

HEIGHT: Wave height

PERIOD: Wave period

DEPTH: Water depth

ZMUD: z-coordinate for mud line in the top-level coordinate system.

Note that for an analysis in the frequency domain the HEIGHT field and PERIOD field are not in use. The wave height is assumed to be equal to two (unity wave amplitude); the wave period is specified via the load frequency on the FRLO or FRQL commands.

SOLM

Options for the sparse linear solver (factorization). If the card is not present, default options will be used.

1——5	6——10	11——15	16——20	21——25	26——30	31——35	36——40
SOLM	SOLVER		ORDER				

41——45	46——50	51————60	61————70	71————80

SOLVER: Factorization algorithm.

=0 Automatically determined by Sestra depending on analysis type.

=1 LL^T factorization.

=2 LDL^T factorization.

ORDER: Method for reordering of the degrees of freedom in the equation system.

=0 Automatically determined by Sestra.

=1 Minimum degree reordering.

=2 Nested dissection reordering.

=4 Parallel nested dissection reordering.

SOLT

Defines a superelement as non-structural, i.e. it contributes with no stiffness (but does transfer load and mass to the remaining structure).

1——5	6——10	11——15	16——20	21——25	26——30	31——35	36——40
SOLT		NONS					

NONS: Parameter for non-structural superelement.

=0 Normal structural superelement (default)

=1 Non-structural superelement.

Note that a SOLT command can only be given after an IDTY command.

SOLV

Options controlling memory usage for matrix factorization and internal storage of loads.

1——5	6——10	11——15	16——20	21——25	26——30	31——35	36——40
SOLV				TMEM	NUMLC	FORCE	OOC

41——45	46——50	51————60	61————70	71————80
		OOCMEM		

Memory options for the dynamic time-domain direct and modal analyses (loads).

TMEM: Parameter controlling memory footprint in internal storage and computation of time-dependent load.

=0 All load vectors are cached (default).

=1 Sestra reads load vectors from the Loads Interface File as the time-integration algorithm marches in time. Internal caching is used with default settings. This setting may save significant memory for analyses with many load cases in each load history.

> 1 As 2, but internal caching of 2^{TMEM} vectors is used.

Memory options for linear static analyses (loads).

NUMLC: Number of load cases to keep in memory at the same time. The parameter can be used to reduce memory footprint at a modest increase in analysis time. If blank or equal to zero, all load cases are loaded into memory.

Memory options for matrix factorization.

FORCE: Enforcement of memory options (OOC, OOCMEM).

=0 Solver memory settings can be changed by Sestra (default).

=1 Solver memory settings are strictly enforced as specified on OOC and OOCMEM

OOC: In-core or out-of-core matrix factorization (sparse linear solver).

=0 In-core (in physical memory) matrix factorization (default). Faster if sufficient physical memory is available.

=1 Out-of-core (utilizing the HDD) matrix factorization. Requires significantly less physical memory.

OOCMEM: Upper limit for physical memory usage when out-of-core factorization is enabled, in Megabytes (the default value is 5000). Only relevant when OOC=1. Sestra may modify this value if FORCE=0.

TILO

Definition of time for time-dependent load specification

1——5	6——-10	11————20	21————30	31————40
TILO	IDTILO		OPT	
			= 1	T1
			= 2	T1

41————50	51————60	61————70	71————80
T2	...		TN
DELT	TN		

IDTILO: Integer ID of this time definition

OPT: Time definition option.

=1 Time instants specified as T1, T2, T3, . . . , TN. In this case continuation records may be used to specify as many time instants as desired. Time instants continue in column 11 on continuation records.

=2 Time instants defined by T1:DELT:TN. For example T1=0,DELT=0.1,TN=0.3 gives time instants 0, 0.1, 0.2, 0.3.

T1: First point in time

DELT: Time increment

TN: Last point in time

TOUT

Specification of times for which results are stored on the result file. If a reference to a TOUT record is not made from the LOHI record, the times specified on the TILO record will be used instead. (In this case the TOUT record itself may be omitted.)

1——5	6——-10	11————20	21————30	31————40
TOUT	IDTOUT		OPT	
			= 1	T1
			= 2	T1

41————50	51————60	61————70	71————80
T2	...		TN
DELT	TN		

IDTOUT: Integer ID of this time definition

OPT: Time definition option.

=1 Time instants specified as T1, T2, T3, . . . , TN. In this case continuation records may be used to specify as many time instants as desired. Time instants continue in column 11 on continuation records.

=2 Time instants defined by T1:DELT:TN. For example T1=0,DELT=0.1,TN=0.3 gives time instants 0, 0.1, 0.2, 0.3.

T1: First point in time

DELT: Time increment

TN: Last point in time

WIND

Wind-condition specification. This card is typically created by a load generator (e.g. Sesam Wadam or Wajac). The information on the card is not used by Sestra, but is transferred to the result file for use in post processors. Sestra will transfer all non-empty fields to the result file.

1——5	6——-10	11————20	21————30	31————40
WIND	IDWIND	VEL	ANGLE	GUSTF

41————50	51————60	61————70	71————80
H0	HEXP	PRAT	LENFAC

(new line)

1——5	6——-10	11————20	21————30	31————40
	IFORM			

IDWIND: Integer ID of this wind condition specification

VEL: Mean wind velocity

ANGLE: Wind direction angle in degrees

GUSTF: Gust factor

H0: Mean wind velocity level relative to the still water

HEXP : Height exponent

PRAT : Mean period ratio (Ta/Ta0)

LENFAC : Wind length adjustment factor.

IFORM : Wind profile formula number.

A Differences between Sestra 10.12.0 and Sestra 8.8-2

Below, important differences between Sestra 10.12.0 and Sestra 8.8-2 are listed. In some cases, these differences will yield differences in the calculated results.

A.1 Differences due to time-dependent boundary conditions

In a time-dependent dynamic analysis nodes may be given a prescribed time-dependent motion. This motion is specified as prescribed displacements or as prescribed accelerations as a function of time.

The handling of time-dependent boundary conditions is different in Sestra 10.12.0 and Sestra 8.8-2 resulting in different results in some cases. We consider first the method of Sestra 10.12.0 and then briefly outline the method in Sestra 8.8-2.

In Sestra 10.12.0, whenever time-dependent prescribed accelerations or accelerations are specified, a function $x_b(t)$ is constructed internally. $x_b(t)$ is here vector valued containing the specified motion of all prescribed (boundary) degrees of freedom. In the time integration, the explicit time-dependent load $R_i(t)$ for the remaining structure (internal degrees of freedom) is replaced by a modified load

$$R_i'(t) = R_i(t) \underbrace{-M_{bi}^T \ddot{x}_b(t) - C_{bi}^T \dot{x}_b(t)}_{\text{BCDYN}} - K_{bi}^T x_b(t), \tag{A.1}$$

where the mass, damping and stiffness terms account for the loading effect on remaining structure by the boundary nodes. Sestra 10.12.0 uses numerical integration and differentiation internally to construct $x(t), x'(t)$ and $x''(t)$ depending on the user-specified data. This method provides consistent treatment of time-dependent boundary conditions whether provided as displacements, accelerations or a combination.

In Sestra 8.8-2, the program distinguishes between three cases: only prescribed displacements, a combination of prescribed displacements and prescribed accelerations, or only prescribed accelerations. When **only prescribed displacements** are given, Sestra 8.8-2 neglects the term labled BCDYN in (A.1). For backward compatibility this term can be neglected in Sestra 10.12.0 using a parameter on the FRSP command. When **a combination of prescribed displacements and prescribed accelerations** is used, the damping term in (A.1) is neglected.

The case of **only prescribed accelerations** is different and may yield rather different results in Sestra 10.12.0 and in Sestra 8.8-2: In short, the displacement results of Sestra 8.8-2 must in this case be interpreted as displacements relative to an accelerated system frame. This is discussed in detail in [1, Appendix B].

It should be noted that for models in which

- all prescribed accelerations are given the same time history; and
- there is no specified damping,

displacement results, velocity results and acceleration results will be different, while derived results (stresses and forces) will be identical with Sestra 8.8-2.

A.2 Difference in printed RBLODCMB cards on the Results Interface File

An RBLODCMB card defines the linear combination of load cases relevant for a particular result case. In a time-domain analysis the print of RBLODCMB cards from Sestra 10.12.0 may be different when compared directly with the print of RBLODCMB cards from Sestra 8.8-2. However, the prints of the cards are equivalent.

In a time-domain analysis a result case corresponds to a particular value of time t. Generally the load $R(t)$ associated with t can be written as $R(t) = a \cdot \text{load case A} + \text{b} \cdot \text{load case B}$. The coefficients a, b and the load case numbers A and B are printed on the RBLODCMB card. The following differences between Sestra 10.12.0 and Sestra 8.8-2 may occur.

- With particular analysis configurations, A and B may refer to the same load case. In such analyses Sestra 10.12.0 may explicitly print the combination $a \cdot \text{load case A} + \text{b} \cdot \text{load case B}$ while Sestra 8.8-2 prints the combination $\hat{a} \cdot \text{load case A}$, where $\hat{a} = a + b$.
- The order of the coefficient-load case pairs (a, A) and (b, B) may be swapped.

A.3 Differences in handling of non-structural elements

Contribution to load sum

Sestra 10.12.0: If a loaded non-structural element is connected to a node with fixation, the part of the load absorbed by this fixation is excluded from the load sum as it does not act on the regular structure.

Sestra 8.8-2: If a loaded non-structural element is connected to a node with fixation, the load contributes fully to the load sum.

Reaction forces

Sestra 10.12.0: Non-structural elements do not directly contribute to nodal reaction forces (or reaction force sum). For example, there is no reaction force in a fixed node connected only to a non-structural element.

Sestra 8.8-2: Loaded non-structural beams with fixation contribute directly to nodal reaction forces (and reaction force sum) as if these elements have zero stiffness. For example, there is a non-zero reaction force in a fixed node connected only to a loaded non-structural element. Non-structural superelements are not retracked and give no direct contribution to reaction forces.

Result evaluation (retracking)

Sestra 10.12.0 does not calculate displacements, stresses, and forces in non-structural elements. Displacement values are set to zero while stresses and forces are not computed.

For (single) non-structural beams, Sestra 8.8-2 calculates element forces as if these elements have no stiffness. As a result, non-zero element forces occur for beams with element loads. Sestra 8.8-2 does not perform retracking of non-structural superelements.

A.4 Other differences

Consistent vs. lumped mass matrices

Sestra 10.12.0 only supports consistent (full) element mass matrices, while Sestra 8.8-2 also supports lumped (diagonalized) element mass matrices. Furthermore, Sestra 8.8-2 has built-in logic (if enabled) to automatically determine whether to employ consistent or lumped element mass matrices depending on model properties.[6]

Consequently, the results of an analysis may be different due to

- a consistent vs. lumped mass matrix used in a dynamic analysis;
- modified gravity loads (these are for most elements calculated by multiplying the element mass matrix by a gravity acceleration vector).

Note, however that Sestra 8.8-2 can be forced to employ full element mass matrices [1]; in this case there will not be differences in the results due to different element mass matrices.

Contribution to gravity load from stiff ends on beam elements (OPT=1 on the BGRAV record)

Sestra 10.12.0 does not support OPT=1 (stiff ends contribute to gravity load) on the BGRAV record. If OPT=1 is present, the analysis continues as if OPT=0 with a warning printed on the listing file.

Loads written to the Results Interface File

The RSEL command on the Sestra input file can be used to control whether or not loads are transferred from the Input Interface File or Loads Interface File to the Results Interface File. In Sestra 8.8-2, the parameter ISEL1 controls the BELOAD1 (distributed line load for two-noded beam elements) while the parameter ISEL7 controls all other load cards. ISEL1 also controls conversion of inertia loads to distributed line loads.

In Sestra 10.12.0, ISEL7 controls all load cards present on the Input Interface File or Loads Interface File, while ISEL1 is used to switch on or off transfer of inertia loads as equivalent distributed line loads (for two-noded beams only).

Note that distributed line loads must be present on the Results Interface File for correct calculation of force and moment diagrams for two-noded beam elements in postprocessor applications.

Matrix output in combination with other analyses

In Sestra 10.12.0, matrix output (using the MATR command) is handled internally as a separate analysis type, and may not be combined with other analyses. This is possible in Sestra 8.8-2.

[6] In particular, unless a consistent mass matrices are explicitly requested, Sestra 8.8-2 will employ lumped element mass matrices for all load cases so long as gravity load is specified for at least one load case.

Overriding non-zero prescribed displacements when degree of freedom is fixed at zero

When the boundary condition code on BNBCD is 1 (fixed at zero displacement, temperature, etc.) and the corresponding degree of freedom on BNDISPL is non-zero, Sestra 8.8-2 uses the non-zero boundary value. Sestra 10.12.0 will in this case override the non-zero boundary value, fixing the degree of freedom at zero displacement.

Results Interface File format

Sestra 10.12.0 supports only the text format SIF and the direct-access binary format SIN. The unformatted sequential format SIU is not supported.

Superelement support

Sestra 10.12.0 supports analysis of multiple superelements through an automated workflow comprising the following steps: i) merge of the superelement hierarchy to a global model; ii) standard Sestra analysis; iii) distribution of results back to the original superelements. Note that Sestra 10.12.0 does not support the superelement technique (sub-structuring) as a solution algorithm.

B Theory

B.1 Direct time integration in Sestra 10.12.0

We are given a second-order system of equations in the unknown vector $x(t)$

$$M\ddot{x}(t) + C\dot{x}(t) + Kx(t) = R(t), \qquad t \in [t_{\text{init}}, t_{\text{final}}], \tag{B.1}$$

$$x(t_{\text{init}}) = d_0, \tag{B.2}$$

$$\dot{x}(t_{\text{init}}) = v_0, \tag{B.3}$$

where $M, C, K \in \mathbb{R}^{N\times N}$ are the mass, damping, and stiffness matrices, respectively, and $R \in \mathbb{R}^N$ is the time-dependent load vector. N is the number of (unprescribed) degrees of freedom in the system.

Equations (B.1)–(B.3) represent a finite-element (spatial) discretization of the continuous equations of linear elasticity. To discretize (B.1)–(B.3) in time, Sestra 10.12.0 employs a generalization of the *Wilson Θ-method* [12, 11]. This generalization also incorporates the Collocation scheme of Hilber and Hughes [11] (of degree 3), and the Generalized-α method [3]. The latter allows for improved control of the numerical dissipation properties; and moreover, the implementation of the Generalized-α method allows an adaptive time-step control to be used [8].

Below, we first state the Generalized-α method as introduced in [3]; we then generalize it to incorporate the Wilson Θ-method by the introduction of an additional algorithm parameter.

The Generalized-α method of [3] is given as follows. For a partition $t_{\text{init}} = t_0, t_1, \ldots, t_n, \ldots, t_K = t_{\text{final}}$ of the time line, let $d_n \approx x(t_n)$, $v_n \approx \dot{x}(t_n)$, $a_n \approx \ddot{x}(t_n)$, and $h = t_{n+1} - t_n$, $0 \le n \le K-1$. Then

$$d_{n+1} = d_n + hv_n + h^2\Big(\Big(\frac{1}{2} - \beta\Big)a_n + \beta a_{n+1}\Big) \tag{B.4}$$

$$v_{n+1} = v_n + h\big((1-\gamma)a_n + \gamma a_{n+1}\big) \tag{B.5}$$

$$Ma_{n+1-\alpha_m} + Cv_{n+1-\alpha_f} + Kd_{n+1-\alpha_f} = R(t_{n+1-\alpha_f}), \tag{B.6}$$

where $r_{n+1-\alpha} \equiv (1-\alpha)r_{n+1} + \alpha r_n$. Note that (B.4) and (B.5) represent the standard Newmark [9] update equations for the displacement and the velocity, while (B.6) is the equations of force equilibrium weighted between successive time instances. Finally, the initial acceleration is determined directly from (B.1) as

$$a_0 = M^{-1}(R(t_{\text{i}}) - Cv_0 - Kd_0). \tag{B.7}$$

Let $\hat{\alpha}_m \equiv (1-\alpha_m)$ and $\hat{\alpha}_f \equiv (1-\alpha_f)$. If we solve for a_{n+1} in the balancing equation (B.6) and substitute the expressions for d_{n+1} and v_{n+1}, we obtain

$$\begin{aligned}\big(\hat{\alpha}_m M + \hat{\alpha}_f h\gamma C + \hat{\alpha}_f h^2\beta K\big)a_{n+1} \\ = R(t_n + \hat{\alpha}_f h) - C(v_n + \hat{\alpha}_f h(1-\gamma)a_n) \\ - K(d_n + \hat{\alpha}_f h v_n + \hat{\alpha}_f h^2\Big(\frac{1}{2} - \beta\Big)a_n) - \alpha_m M a_n.\end{aligned} \tag{B.8}$$

This is the linear system solved at each time step if the Generalized-α method is used with a constant time step size h.

We now derive the generalization which also incorporates the Wilson Θ-method. To this end, we introduce a new scalar parameter Θ and an associated weighted acceleration $k_n \equiv \Theta a_{n+1} + (1-\Theta)a_n$ (note for $\Theta = 1$ we have $k_n = a_{n+1}$). If we set $\Theta = 1$, we can write (B.8) as

$$\begin{aligned}\big(\hat{\alpha}_m M + \hat{\alpha}_f h\gamma C + \Theta\hat{\alpha}_f h^2\beta K\big)k_n \\ = R(t_n + \hat{\alpha}_f h) - C(v_n + \hat{\alpha}_f h(1-\gamma)a_n) \\ - K(d_n + \hat{\alpha}_f h v_n + \Theta\hat{\alpha}_f h^2\Big(\frac{1}{2} - \beta\Big)a_n) - \alpha_m M a_n,\end{aligned} \tag{B.9}$$

where $a_{n+1} = k_n/\Theta - (1-\Theta)a_n/\Theta$.

The time-integration scheme of Sestra is given by the two displacement and velocity Newmark updates (B.4) and (B.5) together with the Θ-parameterized balancing equation (B.9).

The value $\gamma = 1/2$ is hard coded in Sestra, and is required for second-order accuracy of the algorithm. The desired algorithm can be obtained from (B.9) as follows.

Collocation scheme of Hilber and Hughes

For

$$\Theta \neq 1, \qquad \alpha_f = 1 - \Theta, \qquad \alpha_m = 0, \tag{B.10}$$

the equation (B.9) yields the collocation scheme of Hilber and Hughes [11] (of degree 3).

The method has parameters Θ and β. Provided that $\Theta \geq 1$, the method is second-order accurate and absolutely stable provided that

$$\frac{2\Theta^2 - 1}{8\Theta^3 - 4} \leq \beta \leq \frac{\Theta}{2(\Theta + 1)}. \tag{B.11}$$

Wilson Θ method

For

$$\Theta \neq 1, \qquad \alpha_f = 1 - \Theta, \qquad \alpha_m = 0, \qquad \beta = 1/6 \tag{B.12}$$

the equation (B.9) yields the Wilson Θ-method [12]. In this case, the value

$$\Theta = 0.5 + \sqrt{(7 + 2(4)^{1/3})/12} \approx 1.420815 \tag{B.13}$$

can be shown to yield optimal numerical dissipation properties [12], in which the high-frequency modes of the solution are subjected to the most numerical damping.

Newmark method

For $\Theta = 1$, $\alpha_m = 0$, $\alpha_f = 0$, (B.9) yields the Newmark method [9]. The method is second-order accurate. For absolute stability, $\beta = 1/4$ is required (see the requirement (B.11)).

Generalized-α method

For $\Theta = 1$, (B.9) yields the Generalized-α method [3]. The method has parameters β, γ, α_f and α_m, but is controlled with a single parameter $\rho \in [0, 1]$.

$$\alpha_m = (2\rho - 1)/(\rho + 1), \tag{B.14}$$
$$\alpha_f = \rho/(\rho + 1), \tag{B.15}$$
$$\beta = (1 - \alpha_m + \alpha_f)^2/4, \tag{B.16}$$
$$\gamma = 1/2 - \alpha_m + \alpha_f. \tag{B.17}$$

All values of $\rho \in [0, 1]$ yields a second-order accurate and absolutely stable method. The parameter ρ controls the amount of numerical high-frequency damping. A value $\rho = 0$ yields the maximum amount of high-frequency numerical damping; $\rho = 1$ yields no numerical damping.

It is important to remark that, in the general case $\rho \neq 1$, the computed *accelerations* of the Generalized-α method are only first-order accurate while in the special case $\rho = 1$ accelerations are second-order accurate [7, 4]. Independent of ρ, the computed velocities and displacements are second-order accurate. Thus, if accelerations is the output of interest from the analysis, either

- a smaller timestep value should be considered; or
- $\rho = 1$ shoud be considered (losing the numerical damping properties of the method); or
- the Generalized-α-method should not be used.

B.1.1 Evaluation of loads

The input to Sestra allows loads to be specified at discrete time instants. When the equations above require the evaluation of loads for values of the time parameter different from that of the specified loads, linear interpolation is used.

B.1.2 Evaluation of the results

If accelerations, velocities, or displacements are requested for $t = t_n$, Sestra will provide the respective results directly from (B.4), (B.5), and (B.9) above. When results are requested for $t_n \leq t \leq t_{n+1}$, linear interpolation is used.

B.1.3 Adaptive time stepping

If the Generalized-α method is used, an adaptive time-stepping algorithm may be enabled. In this case the time-step value h in Section B.1 must be replaced by

$$h_{n+1} \equiv t_{n+1} - t_n; \tag{B.18}$$

hence (B.9) is effectively replaced by

$$\begin{aligned}\big(\hat{\alpha}_m M + \hat{\alpha}_f h_{n+1}\gamma C + \Theta\hat{\alpha}_f h_{n+1}^2\beta K\big)k_n \\ = R(t_n + \hat{\alpha}_f h_{n+1}) - C(v_n + \hat{\alpha}_f h_{n+1}(1-\gamma)a_n) \\ - K(d_n + \hat{\alpha}_f h_{n+1} v_n + \Theta\hat{\alpha}_f h_{n+1}^2\Big(\frac{1}{2} - \beta\Big)a_n) - \alpha_m M a_n.\end{aligned} \tag{B.19}$$

An adaptive time-stepping algorithm attempts at minimizing the number of required time steps by selecting values $h_i, h_{i+1}, \ldots$ which are as large as possible, but sufficiently small that a user-specified tolerance for the numerical integration error is satisfied. However, it is important to note that when an adaptive time-stepping algorithm is employed, the left-hand-side matrix of (B.19) (the *effective mass matrix*) is changed whenever h_i is modified. As a result, whenever $h_{i+1} \neq h_i$ an expensive refactorization of the system matrix is required. Hence the algorithm should balance the number of time steps and the number of changes in time-step value.

The adaptive time-stepping algorithm in Sestra 10.12.0 is specifically developed for the Generalized-α method, and was introduced in [8]. The method is described briefly below. The following error estimate e_{n+1} for the local displacement error $E_{n+1} = d_{n+1} - x(t_{n+1})$ (the error at time t_{n+1} given an exact solution at time t_n) is used.

$$e_{n+1} \equiv h_{n+1}^2\left(\Big(\beta - \frac{1-\alpha_m}{6(1-\alpha_f)}\Big)(a_{n+1} - a_n) + \Big(\frac{1}{6-6\alpha_f} - \frac{1}{2}\Big)w_n\right), \tag{B.20}$$

$$w_n = \frac{\alpha_m - \alpha_f}{(1-\alpha_f)^2}(a_n - a_{n-1}) - \frac{\alpha_f}{1-\alpha_f}w_{n-1}, \tag{B.21}$$

$$w_0 = 0. \tag{B.22}$$

For the derivation of this estimate, the reader is referred to [8]. In practice, a measure of the relative error is of interest. To this end, let

$$e_{n+1}^{\mathrm{rel}} \equiv \frac{\|e_{n+1}\|_2}{K_{n+1}}, \qquad K_{n+1} \equiv \frac{h_{n+1}}{m}\sum_{i=n-m+1}^{n}\frac{\|d_{i+1} - d_i\|}{h_{i+1}}, \tag{B.23}$$

where m is a parameter controlled by the algorithm. K_{n+1} is thus a measure of the average change in the solution over the last m timesteps.

Given a user-specified tolerance ϵ_{tol}, a user-specified suggested value for h_1 (the initial time-step value), and algorithmic parameters c_{lb}, p_{inc} and p_{dec}, the algorithm may now be described as follows for $n = 1, \ldots$.

First, a candidate solution (d_n, v_n, a_n) is computed with one step of the Generalized-α method with the proposed time-step value h_n. The error approximation e_n is then calculated from (B.20). If

$$e_n^{\mathrm{rel}} \leq \epsilon_{\mathrm{tol}}, \tag{B.24}$$

then the candidate solution is accepted. If in addition

$$e_n^{\mathrm{rel}} \leq c_{\mathrm{lb}}\epsilon_{\mathrm{tol}}, \tag{B.25}$$

then a time-step increase is possible for the next iteration. However, the time-step value is not increased unless (B.25) is satisfied for L successive iterations, in order to avoid too frequent system matrix factorizations. In the case of a time-step increase, the value

$$h_{n+1} = h_n\Big(\frac{\epsilon_{\mathrm{tol}}}{e_{n,\max}^{\mathrm{rel}}}\Big)^{1/p_{\mathrm{inc}}} \tag{B.26}$$

is used. Here, $e_{n,\max}^{\mathrm{rel}}$ is the maximum value of e_i^{rel} that has occurred over the last L iterations (that is, e_i^{rel} is tracked over successive iterations for which (B.25) is satisfied).

If on the other hand

$$e_n^{\text{rel}} > \epsilon_{\text{tol}}, \tag{B.27}$$

then the error is too large and the candidate solution is rejected. A time-step decrease must be performed, and a new candidate solution must be computed. If the time step was increased at the previous iteration, then the value prior to the increase is used. Otherwise, the time-step value is decreased according to

$$h_n = h_n \left(\frac{\epsilon_{\text{tol}}}{e_n^{\text{rel}}} \right)^{1/p_{\text{dec}}}. \tag{B.28}$$

In Sestra 10.12.0, the parameters $p_{\text{dec}} = p_{\text{inc}} = 2$ and $c_{\text{lb}} = 0.75$ are used and may not be modified by the user. The parameter L is calculated internally by the program based on the specified error tolerance and the heuristic discussed in [8, Eq. (42), Eq. (53)].

Note: For problems in which $d_0 = v_0 = 0$ (quiescent initial conditions) and $R(t = 0) = 0$ (zero initial load), the algorithm will employ the user-specified h_1 as the initial time-step value. For problems with non-zero initial conditions or a non-zero initial load, h_1 is merely a guess for the initial time-step value.

B.1.4 Steady-state detection

If steady-state detection is enabled, the time-integration analysis is automatically terminated when the response reached is steady (periodic) state. Sestra detects the time of steady state based on the response of the entire system.

In short, Sestra computes the relative differences in the system response over a number of successive periods. When all these differences are below a user-specified tolerance, the system is said to have reached its steady state.

In more detail, the steady-state detection algorithm is implemented as follows. Let the load be periodic with period T, and label then periods with a number p, $p = 0, 1 \ldots$. For each period p, let $x_i^p \in \mathbb{R}^N$ denote the state of the system at time $t_i^p = pT + i\Delta t$, where Δt is a temporal resolution parameter, $i = 0, 1, \ldots, T/\Delta T$, and N is the number of degrees of freedom in the system. Recall that acceleration is the primary variable of the solver and hence x_i^p is an acceleration vector.

Let $r \in \mathbb{R}^N$ be a vector of pseudo-uniformly distributed random values in $(-1, 1)$. For period p and at each time step i, Sestra computes the inner product $\alpha_i^p = r^{\text{T}} x_i^p$. For each period p, we may then define the vector

$$A_p = \begin{bmatrix} \alpha_0^p \\ \alpha_1^p \\ \vdots \\ \alpha_M^p \end{bmatrix} \tag{B.29}$$

(where $M = T/\Delta T$). Sestra now determines whether steady state has been reached by consideration of the relative differences of the A_p for successive values of p. When the quantity $E_p = \|A_p - A_{p-1}\|_2 / \|A_p\|_2 < \text{TOL}$ over a window of P successive periods, steady state is reached.

The value of TOL is user controlled. The parameters P and Δt are internal to Sestra ($P = 5$ and $\Delta t = T/99$ are used) and may not be modified by the user.

A few comments are in order.

- The inner-products $\alpha_i^p = r^{\text{T}} x_i^p$ correspond to weighted sums of system state vectors at each point in time. The weights are chosen randomly so that any symmetries in the model do not cause a spatially symmetric response which accidentally leads to false detection of steady state.
- The vector r is computed with a random number generator with a fixed seed so that the results of Sestra remain reproducible.

B.1.5 Prescribed displacements, velocities, and accelerations

Sestra 10.12.0 supports time-dependent boundary conditions specified as prescribed displacements, prescribed accelerations, or a combination.

The displacement, velocity, and acceleration vectors may be written as

$$x(t) = \begin{bmatrix} x_i(t) \\ x_b(t) \end{bmatrix}, \qquad \dot{x}(t) = \begin{bmatrix} \dot{x}_i(t) \\ \dot{x}_b(t) \end{bmatrix}, \qquad \ddot{x}(t) = \begin{bmatrix} \ddot{x}_i(t) \\ \ddot{x}_b(t) \end{bmatrix}, \tag{B.30}$$

respectively, where the subscript $_i$ indicates unknown (internal) degrees of freedom, and the subscript $_b$ indicates prescribed (boundary) degrees of freedom. With a particular ordering of the unknowns (the $_i$ followed by the $_b$), the equation (B.1) becomes (omitting the dependence on t for simplicity)

$$\begin{bmatrix} M_{ii} & M_{ib} \\ M_{bi} & M_{bb} \end{bmatrix} \begin{bmatrix} \ddot{x}_i \\ \ddot{x}_b \end{bmatrix} + \begin{bmatrix} C_{ii} & C_{ib} \\ C_{bi} & C_{bb} \end{bmatrix} \begin{bmatrix} \dot{x}_i \\ \dot{x}_b \end{bmatrix} + \begin{bmatrix} K_{ii} & K_{ib} \\ K_{bi} & K_{bb} \end{bmatrix} \begin{bmatrix} x_i \\ x_b \end{bmatrix} = \begin{bmatrix} R_i \\ R_b \end{bmatrix}. \tag{B.31}$$

Because the functions $x_b(t)$ are known, the associated equations (rows) may be eliminated from (B.31). The contributions from $x_b(t)$ to the remaining equations (columns) are moved to the right-hand side so that we obtain

$$M_{ii}\ddot{x}_i + C_{ii}\dot{x}_i + K_{ii}x_i = R_i - M_{ib}\ddot{x}_b - C_{ib}\dot{x}_b - K_{ib}x_b \tag{B.32}$$

The time-integration of (B.32) may now be performed as described above, but only for the internal degrees of freedom and with the right-hand side $R(t)$ of (B.1) replaced by the right-hand side of (B.32).

In actual practice, the user may only input either displacements $x(t)$ or accelerations $\ddot{x}(t)$ for a node. If $x(t)$ is given Sestra will internally apply a numerical differentiation scheme to calculate $\dot{x}(t)$ and $\ddot{x}(t)$; if $\ddot{x}(t)$ is given Sestra will apply a numerical integration scheme to calculate $x(t)$ and $\dot{x}(t)$. Note that the user may select to neglect the inertia term $M_{ib}\ddot{x}_b$ and the viscous term $C_{ib}\dot{x}_b$ in (B.32) with the parameter BCDYN on the FRSP command. This option is provided for backward compatibility with earlier versions.

A technical detail to note is that for BCDYN=0,1,2 displacements are represented internally in Sestra as piecewise cubic polynomials, while for BCDYN=3 displacements are represented as piecewise linear functions. Thus if comparing results for, say, BCDYN=1 to results for BCDYN=3 there will be small differences in results even if the boundary elements have no mass (e.g. springs). The magnitude of these differences will depend on the size of the timesteps used for load specification.

B.2 Forced frequency response in Sestra 10.12.0

B.2.1 Equations of motion in the frequency plane

We are as in Appendix B.1 given a second-order system of equations in the unknown vector $x(t)$

$$M\ddot{x}(t) + C\dot{x}(t) + Kx(t) = R(t), \qquad t \in [t_{\text{init}}, t_{\text{final}}], \tag{B.33}$$

where $M, C, K \in \mathbb{R}^{N\times N}$ are the mass, damping, and stiffness matrices, respectively, and $R \in \mathbb{R}^N$ is the time-dependent load vector. Recall that N is the number of (unprescribed) degrees of freedom in the system.

Now, assume that the time-dependent load vector $R \in \mathbb{R}^N$ is harmonic, i.e. load for each degree of freedom is represented by the harmonic function $R_l \sin(\omega t + \theta_l)$. Here R_l is the amplitude, θ_l is the phase angle and ω is the angular frequency. The natural frequency in Hertz [Hz] is

$$f = 2\pi/\omega, \tag{B.34}$$

It is convenient to replace this real function by the complex function $R_l \mathrm{e}^{\mathrm{i}(\omega t+\theta_l)} = R_l \mathrm{e}^{\mathrm{i}\theta_l}\mathrm{e}^{\mathrm{i}\omega t}$. These nodal quantities is used to define the complex load vector $\hat{R}$. Using this the harmonic load can be written

$$R(t) = \mathrm{e}^{\mathrm{i}\omega t}\hat{R}, \tag{B.35}$$

When the load is harmonic, the structural response is harmonic as well, i.e.

$$x(t) = \mathrm{e}^{\mathrm{i}\omega t}\hat{x}_\omega, \tag{B.36}$$

$$\dot{x}(t) = \omega \mathrm{i} \mathrm{e}^{\mathrm{i}\omega t}\hat{x}_\omega, \tag{B.37}$$

$$\ddot{x}(t) = -\omega^2 \mathrm{e}^{\mathrm{i}\omega t}\hat{x}_\omega, \tag{B.38}$$

where $\hat{x}_\omega$ is the complex unknown vector for load frequency ω. The vector is the solution of the linear system

$$(-\omega^2 M + \omega \mathrm{i} C + K)\hat{x}_\omega = \hat{R}. \tag{B.39}$$

The structural response is then described by (B.36)–(B.38).

B.2.2 Modal superposition

In a modal superposition analysis an approximation to the solution of (B.33) is calculated by requiring that

$$x(t) = \sum_{i=1}^{S} y_i(t) v_i, \tag{B.40}$$

where the y_i are unknown time-dependent coefficients and the v_i are the eigenmodes (mode shapes) associated with the generalized eigenproblem

$$Kv_i = \lambda_i M v_i, \qquad v_i^{\mathrm{T}} M v_i = 1, \qquad 1 \leq i \leq S, \tag{B.41}$$

sorted according to increasing eigenvalue λ_i. Here, S is the number of mode shapes to consider for the approximation (typically $S \ll N$).

Letting $V = [v_1, \ldots, v_M] \in \mathbb{R}^{N \times M}$, we have $x = Vy$ and thus also $\hat{x}_\omega = V\hat{y}_\omega$ for complex-valued frequency-dependent (amplitude and phase) coefficients $\hat{y}_\omega$.

The resulting complex system of equations can now be written

$$V^{\mathrm{T}}(-\omega^2 M + \omega \mathrm{i} C + K) V \hat{y}_\omega = V^{\mathrm{T}} \hat{R}_\omega \equiv \hat{r}_\omega. \tag{B.42}$$

Again, due to the orthonormality $V^{\mathrm{T}} M V = I$ of the mode shapes and the admissible forms of damping, the equations of (B.42) are independent such that (B.42) is equivalent to the S decoupled complex equations

$$(-\omega^2 + \omega \mathrm{i} c_i + \lambda_i)\hat{y}_{\omega,i} = \hat{r}_{\omega,i}, \qquad 1 \leq i \leq S. \tag{B.43}$$

Separating the real and imaginary parts of (B.43) yields the two coupled scalar equations

$$\mathrm{Re}(\hat{y}_{\omega,i})(\lambda_i - \omega^2) - \omega c_i \mathrm{Im}(\hat{y}_{\omega,i}) = \mathrm{Re}(\hat{r}_{\omega,i}), \tag{B.44}$$

$$\mathrm{Im}(\hat{y}_{\omega,i})(\lambda_i - \omega^2) + \omega c_i \mathrm{Re}(\hat{y}_{\omega,i}) = \mathrm{Im}(\hat{r}_{\omega,i}), \tag{B.45}$$

$$\tag{B.46}$$

or equivalently

$$\mathrm{Re}(\hat{y}_{\omega,i}) = \frac{(\lambda_i - \omega^2)\mathrm{Re}(\hat{r}_{\omega,i}) + \omega c_i \mathrm{Im}(\hat{r}_{\omega,i})}{(\lambda_i - \omega^2)^2 + \omega^2 c_i^2}, \tag{B.47}$$

$$\mathrm{Im}(\hat{y}_{\omega,i}) = \frac{(\lambda_i - \omega^2)\mathrm{Im}(\hat{r}_{\omega,i}) - \omega c_i \mathrm{Re}(\hat{r}_{\omega,i})}{(\lambda_i - \omega^2)^2 + \omega^2 c_i^2}. \tag{B.48}$$

Finally, the complex displacement vector can be recovered as

$$\hat{x}_\omega(t) = V\hat{y}_\omega(t), \tag{B.49}$$

and the physical response is described by (the real part of)

$$x(t) = \mathrm{e}^{\mathrm{i}\omega t}\hat{x}_\omega(t) = \mathrm{e}^{\mathrm{i}\omega t} V \hat{y}_\omega(t). \tag{B.50}$$

B.3 Reaction forces

We define reaction forces as the forces imposed on the system by boundary constraints. Sestra supports two different types of boundary constraints: support constraints and spring support constraints.

B.3.1 Reaction forces from support constraints

Support constraints are prescribed displacements (zero or non-zero) on system nodes. The forces from support constraints are given by the residual

$$r(t) = M\ddot{x}(t) + C\dot{x}(t) + Kx(t) - R(t), \tag{B.51}$$

where K and $R(t)$ are the stiffness and load matrix of the unconstrained system, and $x(t)$ is the solution of the constrained system. In Sestra 10.12.0 these are approximated by the static analysis residual:

$$r(t) = Kx(t) - R(t). \tag{B.52}$$

B.3.2 Reaction forces from spring support constraints

Spring-to-ground elements impose Robin boundary conditions on the system. That is, the boundary normal stress is set proportional to the boundary displacement. In the discrete case, this corresponds to the application of nodal forces $r_i(t)$ in the connected nodes,

$$r_i(t) = -K_s x_i(t), \tag{B.53}$$

where K_s is the spring stiffness matrix and $x_i(t)$ is the displacement in the connected node i.

B.3.3 Internal constraints

Sestra does not include forces from internal constraints (e.g. linear dependencies or linear inequalities) in the system reactions. That is, internal constraints are imposed on the system matrices in (B.52). Note that internal constraints can result in an imbalance between the sum of applied forces and the reaction force sum.

Internal constraints eliminate degrees of freedom from the system. If reaction forces are requested on eliminated (slave) degrees of freedom, the support constraint reaction (B.52) is set to zero.

B.3.4 Sum of reaction forces

The reaction-force sum is comprised of a force vector

$$F = \sum_i f_i, \tag{B.54}$$

and a moment vector

$$M = \sum_i m_i + r_i \times f_i, \tag{B.55}$$

where the sums are over a set of nodes i; f_i and m_i is the reaction force and moment in node i, respectively, and r_i is the distance vector from the coordinate system origin to node i.

B.4 Non-structural elements

In this section, the method for handling non-structural elements is described. The discussion is limited to linear static analysis, but non-structural elements are supported also for dynamic analyses.

If non-structural elements are specified, a subproblem for the non-structural part of the structure may be identified as

$$\begin{bmatrix} K_{\mathrm{n,ii}} & K_{\mathrm{n,ib}} \\ K_{\mathrm{n,bi}} & K_{\mathrm{n,bb}} \end{bmatrix} \begin{bmatrix} x_{\mathrm{n,i}} \\ x_{\mathrm{n,b}} \end{bmatrix} = \begin{bmatrix} R_{\mathrm{n,i}} \\ R_{\mathrm{n,b}} \end{bmatrix}, \tag{B.56}$$

where the $_n$ stands for *non-structural*, and the degrees of freedom are grouped according to whether they are internal ($_i$) to the non-structural part of the structure, or whether they are boundary ($_b$) degrees of freedom connected to the structural part of the structure.

The load contribution from the non-structural interior to the non-structural boundary is then calculated as

$$\hat{R}_{\mathrm{n,b}} = R_{\mathrm{n,b}} - K_{\mathrm{n,bi}} K_{\mathrm{n,ii}}^{-1} R_{\mathrm{n,i}}. \tag{B.57}$$

The solution in the structural part of the problem can now be calculated as the solution $[x_{\mathrm{s,i}}, x_{\mathrm{s,b}}]^{\mathrm{T}}$ to

$$\begin{bmatrix} K_{\mathrm{s,ii}} & K_{\mathrm{s,ib}} \\ K_{\mathrm{s,bi}} & K_{\mathrm{s,bb}} \end{bmatrix} \begin{bmatrix} x_{\mathrm{s,i}} \\ x_{\mathrm{s,b}} \end{bmatrix} = \begin{bmatrix} R_{\mathrm{s,i}} \\ R_{\mathrm{s,b}} + \hat{R}_{\mathrm{n,b}} \end{bmatrix}, \tag{B.58}$$

where the $_s$ stands for *structural*, and the degrees of freedom are grouped according to whether they are internal ($_i$) to the structural part of the structure, or whether the are boundary ($_b$) degrees of freedom connected to the non-structural part of the structure.

Finally, the solution in the non-structural part of the structure may be calculated[7] as

$$x_{\mathrm{n,b}} = x_{\mathrm{s,b}}, \tag{B.59}$$

$$K_{\mathrm{n,ii}} x_{\mathrm{n,i}} = R_{\mathrm{n,i}} - K_{\mathrm{n,ib}} x_{\mathrm{n,b}}. \tag{B.60}$$

Note that the procedure described here is identical to standard domain decomposition (static condensation) with the exception that stiffness from the non-structural domain is not taken into account in the calculation of the solution in the structural domain.

[7] This retracking for non-structural elements is not performed in dynamic analyses.

B.5 Static condensation in dynamic analysis

In static condensation, or Guyan reduction [5], a subset of the degrees of freedom is satisfying static equilibrium while the remaining is satisfying a dynamic system.

The degrees of freedom x is subdivided into two sets, x_1 and x_2. Using this the load vector is also written as R_1 and R_2, and the mass, damping and stiffness matrix are subdivided into two by two block matrices. In particular the stiffness matrix is written

$$K = \begin{bmatrix} K_{11} & K_{12} \\ K_{12}^T & K_{22} \end{bmatrix}$$

For a given x_1 let x_2 satisfy

$$K_{22}x_2 = -K_{12}^T x_1$$

I.e. x_2 is in static equilibrium and can be computed from x_1. Now

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} x_1 \\ -K_{22}^{-1}K_{12}^T x_1 \end{bmatrix} = Qx_1$$

where

$$Q = \begin{bmatrix} I \\ -K_{22}^{-1}K_{12}^T \end{bmatrix}$$

Using this in the system B.1 we obtain

$$Q^T MQ\ddot{x}_1(t) + Q^T CQ\dot{x}_1(t) + Q^T KQx_1(t) = Q^T R(t) \tag{B.61}$$

The reduced matrices can be used in direct time-domain, direct frequency-domain and free vibration analyses.

Once the accelerations, velocities and displacements, $\ddot{x_1}$, $\dot{x_1}$ and x_1 is computed, the components corresponding to x_2 satisfies:

$$K_{22}\ddot{x}_2 = -K_{12}^T\ddot{x}_1 \quad K_{22}\dot{x}_2 = -K_{12}^T\dot{x}_1$$

and

$$K_{22}x_2 = R_2 - K_{12}^T x_1$$

Note that the lower block of Q is dense. Thus, the computation of the reduced mass and damping matrix is costly, both in terms of computational time and use of memory. Due to some cancellations the computation of the stiffness matrix is not a costly but all the reduce matrices wil be dense in general. If n is the number of degrees of freedom in the model and m is the number of degrees of freedom in x_1, the number of floating point operations to compute the reduce mass and damping matrix are of the order m^2n. The amount of memory is m^2 double precision floating point numbers.

Non-structural elements, see Section B.4, can be combined with static condensation. To present this the degrees of freedom is divided in four sets. The first set, x_1, is the set of super degrees of freedoms connected to both structural and non-structural elements, x_2 is the remaining super degrees of freedom. The non-super degrees of freedom connected to structural elements are x_3, finally, the degrees of freedom connected only to the non-structural elements are x_4. Here x_3 and x_4 is eliminated, thus

$$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ -K_{33}^{-1}(K_{13}^T x_1 + K_{23}^T x_2) \\ -K_{44}^{-1}K_{14}^T x_1 \end{bmatrix} = Q \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

where

$$Q = \begin{bmatrix} I & 0 \\ 0 & I \\ -K_{33}^{-1}K_{13}^T & -K_{33}^{-1}K_{23}^T \\ -K_{44}^{-1}K_{14}^T & 0 \end{bmatrix}$$

The matrix Q is used in B.63, where the contribution from the non-structural elements is not added to the stiffness matrix.

Note that there must be some degrees of freedom in x_1 in order to transfer mass, damping and loads to the dynamical system. I.e. there must be super degrees of freedom connected to both non-structural and

regular elements. Furthermore, the sub matrices K_{33} and K_{44} must be nonsingular. To ensure that the latter is nonsingular it may be necessary to add fixations to the non-structural part of the model.

Once x_1 and x_2 is known, x_3 satisfies

$$K_{33}x_3 = R_3 - K_{13}^T x_1 - K_{23}^T x_2 \tag{B.62}$$

The velocities and the accelerations are computed similarly, except that R_3 vanish. The degrees of freedom in x_4 are non-structural, and are set to zero.

B.6 Component mode synthesis

Component mode synthesis is used to reduce the number of degrees of freedom in a dynamical system. It is an alternative to static condensation. To present the method we use the same notation and partitioning of the degrees of freedom as in Section B.5. Note, however, that in a typical application the super nodes are chosen such that fewer degrees of freedom are super, i.e. the dimension of x_1 will be smaller for component mode synthesis compared to static condensation. To compensate for this the eigenfrequencies and the eigenmodes of the slave degrees of freedom are used as presented below.

First, the Schur decomposition

$$K_{22}V = M_{22}VD$$

where $V^T M_{22} V = I$, is computed. The columns of the matrix V are the eigenmodes and the elements of the diagonal matrix D are the eigenvalues. Usually the number of eigemodes computed are much less than the dimensions of the matrices K_{22} and M_{22}. Then a number of generalized degrees of freedom, denoted α where α is a vector, are introduced. Now x_2 can be computed from x_1 and α by

$$x_2 = -K_{22}^{-1} K_{12}^T x_1 + V\alpha$$

Furthermore, the Q matrix used in static condensation is augmented using the eigenvectors

$$Q = \begin{bmatrix} I & 0 \\ -K_{22}^{-1} K_{12}^T & V \end{bmatrix}$$

Using this in the system B.1 we obtain

$$Q^T M Q \ddot{w}(t) + Q^T C Q \dot{w}_1(t) + Q^T K Q w_1(t) = Q^T R(t) \tag{B.63}$$

where

$$w = \begin{bmatrix} x_1 \\ \alpha \end{bmatrix}$$

As above, the reduced matrices can be used in direct time-domain, direct frequency-domain and free vibration analyses. Time domain analysis can also be combined with modal superposition.

Compared to static condensation a suitable number of eigenpairs must be computed. However, the dimension of the vector can be chosen considerable smaller than in static condensations and as a result the total computational time and memory consumption can be reduced considerably.

B.7 Modal load factors (modal participation factors)

Modal load factors – α_j – can be defined from a modal load as

$$R_j^* = -\Phi_j^T M_{ii} I' \ddot{r}(t) = \alpha_j \ddot{r}(t), \tag{B.64}$$

where the dimension of α_j is 1×3 or 1×6 when rotational accelerations are taken into account. The three (or six) values in α_j are the modal load factors.

In B.64, Φ_j is the jth eigenvector, M_{ii} is the mass and I' is an influence matrix. Finally, $\ddot{r}(t)$ is the acceleration vector.

Please, refer to the users manual of Sestra 8.8-2 for the derivation and the assumptions behind the expression B.64.

B.8 Tension-compression analysis (convex quadratic optimization)

We consider the problem of finding

$$x = \arg\min \frac{1}{2} x^T G x + x^T d \tag{B.65}$$

$$\text{s.t.} \quad Ax \geq b; \tag{B.66}$$

here $x \in \mathbb{R}^N$ represents the displacement vector, $G \in\in \mathbb{R}^{N\times N}$ represents the (positive semidefinite) stiffness matrix, $d \in \mathbb{R}^N$ represents the load vector, and $Ax \geq b$ represents the tension-compression conditions as equivalent inequalities with $A \in \mathbb{R}^{M\times N}$. Note that we consider a system with N degrees of freedom and M inequality constraints.

The problem (B.65) represents a convex quadratic constrained optimization problem.

Sestra has two algorithms for the solution of (B.65): an interior-point algorithm [10] (invoked with the INTP command) and an Uzawa algorithm [13] (invoked with the UZAW command).

The most commonly used method in Sestra is the interior-point method, which we consider in the following. The purpose here is merely to provide sufficient information for the user to interpret the stopping criterion and the residuals calculated by the algorithm (and printed to the Sestra.MLG file) at each iteration. For additional details see e.g. [10].

The (first-order) Karush-Kuhn-Tucker (KKT) conditions associated with (B.65) are

$$Gx - A^T\lambda + d = 0 \tag{B.67}$$

$$Ax - b \geq 0, \tag{B.68}$$

$$(Ax - b)_i \lambda_i = 0, \quad 1 \leq i \leq M \tag{B.69}$$

$$\lambda \geq 0. \tag{B.70}$$

By inserting the *slack variable* $y = Ax - b$ we obtain

$$Gx - A^T\lambda + d = 0, \tag{B.71}$$

$$Ax - y - b = 0, \tag{B.72}$$

$$\Lambda y = 0, \tag{B.73}$$

$$y \geq 0, \tag{B.74}$$

$$\lambda \geq 0, \tag{B.75}$$

where $\Lambda = \text{diag}(\lambda) \in \mathbb{R}^{M\times M}$. Because we consider a convex objective function with convex contraints, the KKT conditions (B.71)–(B.75) are not only necessary but also sufficient: the solution to (B.71)–(B.75) is the solution to (B.65). The interior-point algorithm in Sestra attemts at solving (B.71)–(B.75) iteratively.

At each iteration of this algorithm, three residual quantities are computed (and printed to Sestra.MLG). When all of these quantities are smaller than the tolerance specified on the INTP command the algorithm stops. The residual quantities are directly related to (B.71)–(B.75): At iteration k we define the "primary residual" and "dual residual"

$$r_p^k = Ax^k - y^k - b, \tag{B.76}$$

$$r_d^k = Gx^k - A^T\lambda^k + d, \tag{B.77}$$

respectively; we also define the "duality gap" $\mu^k = \sum_{i=1}^M \lambda_i^k y_i^k$. The quantites printed on Sestra.MLG are

$$\frac{\|r_p^k\|_2}{\|x^k\|_2 + \|y^k\|_2}, \qquad \frac{\|r_d^k\|_2}{\|d^k\|_2 + \|\lambda^k\|_2}, \qquad \min\Big(\frac{\mu^k}{\|y^k\|_2}, \frac{\mu^k}{\|\lambda^k\|_2}\Big), \tag{B.78}$$

and the algorithm stops when the maximum of these quantities reaches below the user-specified tolerance.

C The THTS and THQS finite elements

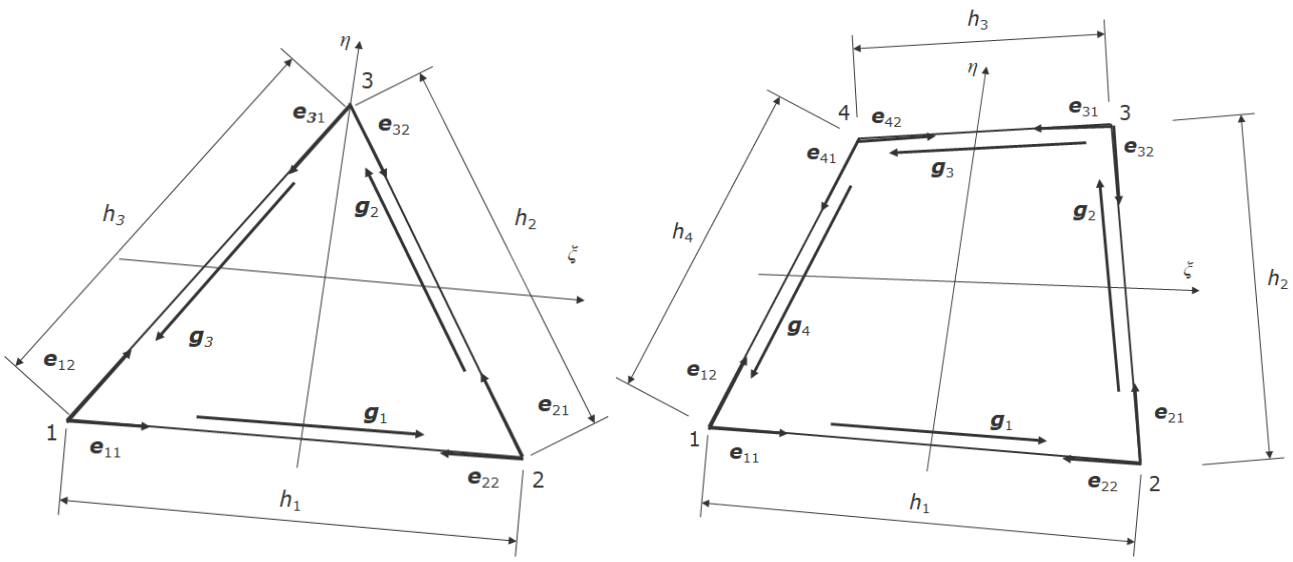


Figure C.1: Element node numbering, and geometric and kinematic data for (a) the 3-node, linear, triangular element THTS, and (b) the 4-node, bilinear, quadrilateral element THQS.

The elements are C^0 continuous finite elements and the element bending formulation is based on an element described by Hughes and Tezduyar, see [6].

THTS: A 3-noded flat shell finite element composed from uncoupled bending and membrane action.

- 3 nodes
- 3 $\times$ 5 (6) = 15 (18) degrees of freedom

THQS: A 4-noded flat shell finite element composed from uncoupled bending and membrane action.

- 4 nodes
- 4 $\times$ 5 (6) = 20 (24) degrees of freedom

Common properties:

- linear element sides
- isotropic or anisotropic material data
- deformations considered:
  - strain due to bending – linear through the thickness of the element
  - transverse shear strain due to bending – constant through the thickness of the element
  - in-plane strain due to membrane action – constant through the thickness of the element
- linear element thickness
- element loads:
  - initial strain (temperature load)
  - surface forces
  - line loads
  - line moment load
  - gravitational load
  - general inertia load

Data types used for this element:

Data type	Mandatory	Note
GELMNT1	yes	
GELREF1	yes	
GNODE	yes	
GCOORD	yes	
GELTH	yes	
MISOSEL	yes	**MISOSEL**
MORSSEL	yes	or **MORSSEL** is mandatory.
MTRSEL		Sometimes referred to from **MORSSEL**.
BEUSLO		
BELLO2		
BEISTE		
BGRAV		
BNACCLO		

The following interpretations and restrictions apply to the data types:

`GELREF1` - INTNO and ISPONO are not yet read and thus the number of integration and stress points are constant in the program.

- The element can model linear variation of the thickness. At present the element input data handles one thickness. I.e. the current implementation has constant thickness.

`BELLO2` - The SIDE definition is not used.

`BEUSLO` - Loads are computed as if they act in the middle-plane (for local $z = 0$) of the element.

References

[1] Sesam user manual: Sestra. Technical report, DNV GL - Software, 2014. Program version 8.8-2. 1.1, 2, 2.8.1, 2.8.4, 2.8.5, 2.9.1, 4, A.1, A.4

[2] Nestegård A. and M. Mejlænder-Larsen. Hydrodynamic added mass of a floating vibrating structure. In O. Faltinsen, C. M. Larsen, T. Moan, K. Holden, and N Spidsøe, editors, *Hydroelasticity in Marine Technology*, 1994. 2.8.3

[3] J. Chung and G. M. Hulbert. A time integration algorithm for structural dynamics with improved numerical dissipation: The generalized-α method. *Journal of Applied Mechanics*, 69(371), 1993. 2.1.2, 3.3.2, 4.2, B.1, B.1

[4] S. Erlicher, L. Bonaventura, and O. S. Bursi. The analysis of the generalized-α method for non-linear dynamic problems. *Computational Mechanics*, 28(2):83–104, Mar 2002. B.1

[5] Robert J. Guyan. Reduction of stiffness and mass matrices. 3(2):380–380, 1965. B.5

[6] T.J.R. Hughes and T.E. Tezduyar. Finite elements based upon mindlin plate theory with particular reference to the four-node bilinear isoparametric element. *Journal of Applied Mechanics*, September 1981(3):587–596, 1981. C

[7] G. M. Hulbert and T. J. R. Hughes. An error analysis of truncated starting conditions in step-by-step time integration: consequences for structural dynamics. 15:901–910, 1987. B.1

[8] G. M. Hulbert and I. Jang. Automatic time step control for algorithms for structural mechanics. *Computer Methods in Applied Mechanics and Engineering*, 126:155–178, 1995. B.1, B.1.3, B.1.3, B.1.3

[9] Nathan A. Newmark. A method of computation for structural dynamics. *Journal of the Engineering Mechanics Division, Proceedings of the Americal Society of Civil Engineers*, 1959. 2.1.2, 4.2, B.1, B.1

[10] J. Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer, 1999. 2.1.8, 3.8, B.8

[11] Syvert P. Nørsett. The collocation schemes of hilber and hughes for nonlinear dynamic problems. *Mathematics and*, (4), 1981. 2.1.2, 4.2, 4.2, B.1, B.1

[12] Syvert P. Nørsett. The wilson θ-method as a spline-collocation method for solving dynamic problems. *Mathematics and Computation*, (1), 1981. 2.1.2, 4.2, B.1, B.1, B.1

[13] Y. Saad. *Iterative Methods for Sparse Linear Systems*. SIAM, 2003. B.8

ABOUT DNV GL

DNV GL is a global quality assurance and risk management company. Driven by our purpose of safeguarding life, property and the environment, we enable our customers to advance the safety and sustainability of their business. Operating in more than 100 countries, our professionals are dedicated to helping customers in the maritime, oil & gas, power and renewables and other industries to make the world safer, smarter and greener.

DIGITAL SOLUTIONS

DNV GL is a world-leading provider of digital solutions for managing risk and improving safety and asset performance for ships, pipelines, processing plants, offshore structures, electric grids, smart cities and more. Our open industry platform Veracity, cyber security and software solutions support business-critical activities across many industries, including maritime, energy and healthcare.