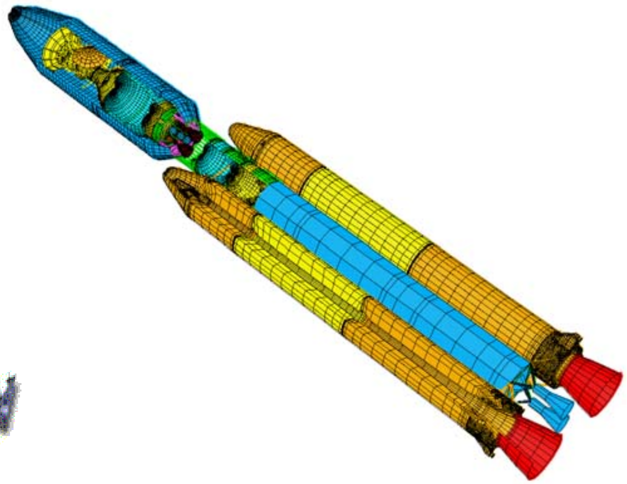
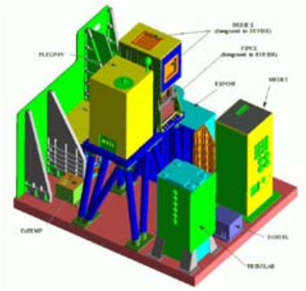
Some Applications of Nastran DMAP in Aerospace Analysis

Paul Blelloch
ATA Engineering, Inc.
paul.blelloch@ata-e.com.com
858-480-2065





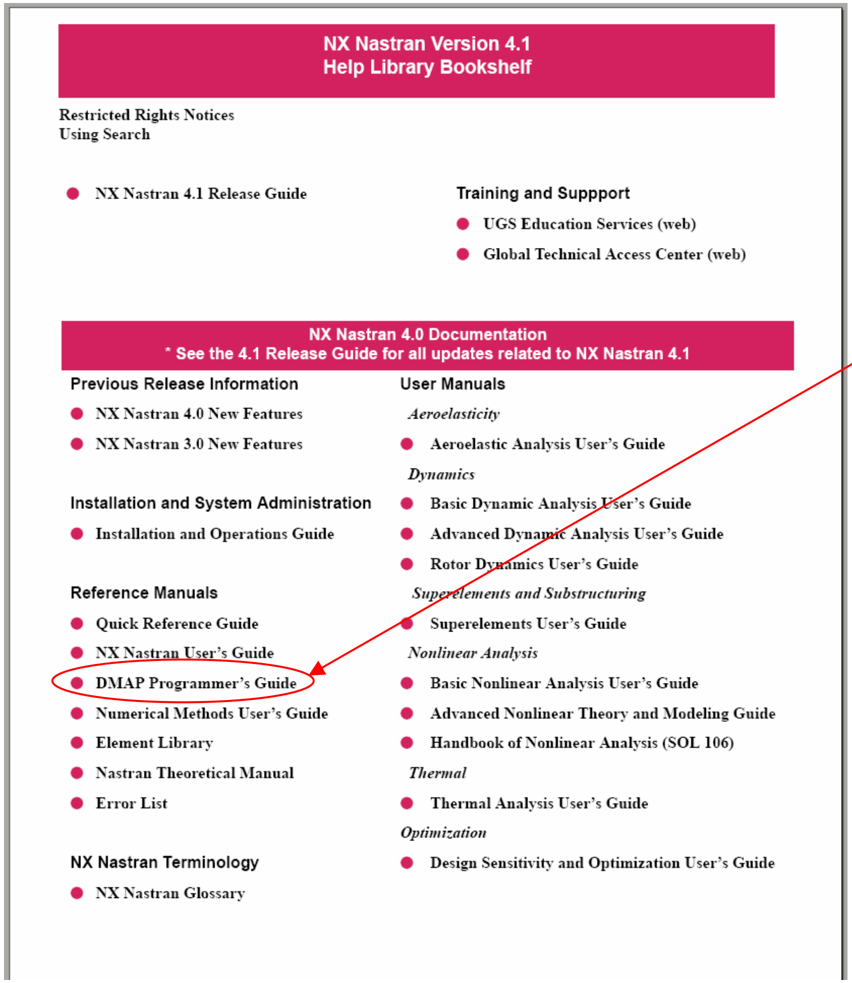
Outline

- What is DMAP?
- Use of DMAP to Select ASET DOF
- Use of DMAP in dynamic analysis
  - Modal transient analysis
  - Fluid mass loading
- Summary/Free stuff!

DMAP

- Direct Matrix Abstraction Program
- Fundamental to the way Nastran works
  - All standard solution sequences written in DMAP
  - Solution sequences can be “ALTERed” by adding user defined DMAP
  - Not true for new nonlinear sequences (SOL 601/701) because these are running other codes
- NASA identified DMAP as one of the key features of NASTRAN early on
  - Not originally designed as a “user friendly” programming language
- Community of DMAP programmers has developed over time
  - nastrandmap on Yahoo Groups
- MSC and NX Nastran DMAP has changed since original COSMIC version
  - Fundamentals are the same
- Some codes with NASTRAN name are not based on DMAP
  - What defines NASTRAN?
- DMAP is a very rich and powerful language, but not for the “faint of heart”
- You can use DMAP alters without knowing DMAP
  - Some rudimentary knowledge is helpful (or entertaining)

DMAP Documentation


NX Nastran Version 4.1
Help Library Bookshelf

Restricted Rights Notices
Using Search

- NX Nastran 4.1 Release Guide

Training and Suppport
- UGS Education Services (web)
- Global Technical Access Center (web)

NX Nastran 4.0 Documentation
* See the 4.1 Release Guide for all updates related to NX Nastran 4.1

Previous Release Information
- NX Nastran 4.0 New Features
- NX Nastran 3.0 New Features

Installation and System Administration
- Installation and Operations Guide

Reference Manuals
- Quick Reference Guide
- NX Nastran User's Guide
- DMAP Programmer's Guide
- Numerical Methods User's Guide
- Element Library
- Nastran Theoretical Manual
- Error List

NX Nastran Terminology
- NX Nastran Glossary

User Manuals

Aeroelasticity
- Aeroelastic Analysis User's Guide

Dynamics
- Basic Dynamic Analysis User's Guide
- Advanced Dynamic Analysis User's Guide
- Rotor Dynamics User's Guide

Superelements and Substructuring
- Superelements User's Guide

Nonlinear Analysis
- Basic Nonlinear Analysis User's Guide
- Advanced Nonlinear Theory and Modeling Guide
- Handbook of Nonlinear Analysis (SOL 106)

Thermal
- Thermal Analysis User's Guide

Optimization
- Design Sensitivity and Optimization User's Guide

Detailed descriptions of all DMAP modules
- 100's of modules

DMAP Syntax

MODULE Input_Datablocks,…/Output_Datablocks,…/Parameters/…

Examples:

ADD A,B/C/ALPHA/BETA/IOPT $\left[C\right]=\alpha\left[A\right]\left(+,\otimes,\div\right)\beta\left[B\right]$

MPYAD A,B,C/X/T/SIGNAB/SIGNC/PREC/FORM $\left[X\right]=\pm\left[A\right]^{(T)}\left[B\right]\pm\left[C\right]$

SOLVE A,B,SIL,USET,PARTVEC/X/SYM/SIGN/SETNAME $\left[X\right]=\pm\left[A\right]^{-1}\left[B\right]$

MATPRN A,B,C,D,E//

MESSAGE //P1/P2/…/PN

OUTPUT4 A,B,C,D,E//ITAPE/IUNIT//BIGMAT/DIGITS

DMAP Offers Common Logic Constructs

- IF THEN, ELSE, ENDIF

```
IF ( SEID>0 ) THEN $
   ...
ELSE $
   ...
ENDIF $
```

- DO WHILE, ENDDO

```
DO WHILE ( I<N ) $
   ...
ENDDO $
```

DMAP Controlled by Parameters

- Parameters values are defined by user on PARAM cards in bulk data or case control
  - Real, complex, integer, logical or character
- Use parameters to control execution of DMAP
  - Character values (‘YES’ or ‘NO’) where possible
  - Set up reasonable default values
  - Clearly comment use of parameters
- Echo back values of parameters using MESSAGE module
- User interaction with DMAP should be through parameters as much as possible

DMAP Can be Used to “Alter” Solution Sequences

- Most common use of DMAP
- Take an existing solution sequence (e.g. modal transient) and modify some part of its operation
- DMAP is “inserted” in a specified location in the solution sequence
- Three ways to alter solution sequence
  - Alter line number; insert at a specified line number
  - String based alter; insert after a specified string
  - MALTER; insert after limited predefined locations
- DMAP is the language, DMAP alter (or alter) is set of DMAP code that modifies a solution sequence

Using a Line or String Based Alter

- Line based alter

```
COMPILE PHASE1DR LIST
ALTER 152 $
MATPRN KGG// $
```

- String based alter

```
COMPILE PHASE1DR LIST
ALTER 'CALL PHASE1A' $
MATPRN KGG// $
```

- String based alters are much more robust from version to version

Using a MALTER Based Alter

- MALTER will search through entire solution sequence for a string
  - Designed to look for specific “MALTER” strings
  - Does not require a COMPILE card
  - COMPILE options set on a “COMPILER” card
- Very coarse ability to alter solution sequence
- I have not used MALTER much
  - Old habits die hard
  - “Real DMAPers” don’t do MALTER ;-)

```
SOL 101
COMPILER LIST
MALTER ‘MALTER:AFTER SUPERELEMENT STIFFNESS GENERATION’
MATPRN KJJZ// $
```

USERDMAP Provides Simple Means to Develop User DMAPs

- Very simple 1 line solution sequence that just runs PREFACE SUBDMAP
  - Reads Case Control and Bulk Data and generates standard datablocks (GEOMi, CASECC, etc.)
  - You can access datablocks already on the database if this is a restart run
- Following are equivalent:

```
ALTER 2
ALTER 'CALL PREFACE'
MALTER '$USERDMAP - AFTER CALL PREFACE'
```

- SOL 100 and SOL USERDMAP are equivalent

ATA

Summary of DMAP

- DMAP is a matrix oriented programming language
- All “standard” Nastran is based on DMAP
- To use DMAP effectively you need to understand:
  - Structure of solution sequences
  - Nastran data blocks and parameters
  - DMAP documentation
- DMAP alters are easy to use IFF:
  - String based to make robust
  - Use parameters to control
  - Are well commented

Outline

- What is DMAP?
- Use of DMAP to Select ASET DOF
- Use of DMAP in dynamic analysis
  - Modal transient analysis
  - Fluid mass loading
- Summary/Free stuff!

Why is Selecting ASET DOF Important?

- ASET DOF used for Guyan Reduction
  - Master DOF in I-DEAS
- Accuracy of reduction very sensitive to choice of DOF
- Reduced matrices used in comparing test and analysis models (TAMs)

$$[M_{AA}] = \begin{bmatrix} I \\ G_{OA} \end{bmatrix}^T [M_{FF}] \begin{bmatrix} I \\ G_{OA} \end{bmatrix}$$

$$[G_{OA}] = -[K_{OO}]^{-1}[K_{OO}]$$

DOF Selected Using Iterative Residual Kinetic Energy Algorithm

- Good Guyan reduction results in orthogonal modes up to frequency range of interest

$$\varepsilon = \left| [I] - [\Phi_A]^T [M_{AA}][\Phi_A] \right|$$

- Define Residual Kinetic Energy as grid point KE associated with error in expanded mode

$$RKE = [M_{OO}]\left([\Phi_O] - [G_{OA}][\Phi_A]\right) \otimes \left([\Phi_O] - [G_{OA}][\Phi_A]\right)$$

- Iterative Residual Kinetic Energy (IRKE) starts with small ASET and adds DOF one or more at a time based on RKE
- Error calculated at each step. Algorithm stopped when error reaches acceptable level

ATA

DMAP is ideal tool for implementing IRKE

- Need to deal with full size matrices
  - Possibly millions of DOF
  - Problem would not be feasible in Matlab
- Key to efficient implementation is to avoid solving $[K_{OO}]^{=-1}[K_{OA}]$ at each iteration
- Use ‘U1’ set to limit DOF to choose from
- Write error norm at each step to monitor progress
- Other algorithms implemented in DMAP
  - Effective Independence (EI)
  - Mass Weighted Effective Independence (MWEI)
  - Iterative Guyan Reduction (IGR)
- Part of a package of DMAP alters called TAMKIT

Outline

- What is DMAP?
- Use of DMAP to Select ASET DOF
- Use of DMAP in dynamic analysis
  - Modal transient analysis
  - Fluid mass loading
- Summary/Free stuff!

ATA

A DMAP Example

- Correct “Deficiencies” in Modal Transient Analysis
  - Start from a quasi-static initial condition
  - Include only flexible contribution to displacement outputs
  - Static and Dynamic Uncertainty Factors (SUF and DUF)
  - Conversion of structural to viscous damping

Initial Conditions

- Modal transient does not support initial conditions
  - Initial force treated as a step
  - IC Case Control allows some capability, but not practical
- Input force can be decomposed as follows:

$$[I]\{\ddot{q}(t)\}+[B]\{\dot{q}(t)\}+[\Lambda]\{q(t)\}=[\Phi_A]^T\left(\left(\{f_A(t)\}-\{f_A(t_0)\}\right)+\{f_A(t_0)\}\right)$$

- Quasi-static response defined as follows:

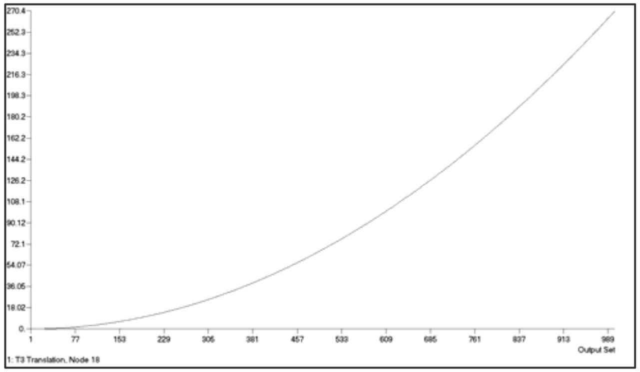
$$\{\ddot{q}_{RB}\}=[\Phi_A]^T_{RB}\{f_A(t_0)\} \qquad \{q_{RB}\}=\{0\}$$

$$\{\ddot{q}_{EL}\}=\{0\} \qquad \{q_{EL}\}=[\Lambda_{EL}]^{-1}[\Phi_A]^T_{EL}\{f_A(t_0)\}$$

- Subtract initial forces from force time history and calculate response
- Add quasi-static response to initial forces back in

Remove Rigid Body Displacements

- Typically not interested in rigid body displacements
  - Rigid body displacements “swamp” flexible displacements
  - Only flexible displacements generate loads
  - Including rigid body displacements results in inaccurate loads
  - Can do this using PARAM,LFREQ, but that also gets rid of rigid body accelerations
- Solution is to simply set rigid body displacements to zero



$$\{q_{RB}(t)\} = \{0\}$$

ATA

Apply SUF and DUF

- Uncertainty factors commonly used in coupled loads analysis
  - Occasionally desire different factors on quasi-static (SUF) vs. dynamic (DUF) parts of response
- Quasi-static response defined as follows:

$$\left\{\ddot{q}_{RB}(t)\right\}_{QS} = \left[\Phi_A\right]_{RB}^{T}\left\{f_A(t)\right\} \qquad \left\{q_{RB}(t)\right\}_{QS} = \left\{0\right\}$$

$$\left\{\ddot{q}_{EL}(t)\right\}_{QS} = \left\{0\right\} \quad \left\{q_{EL}(t)\right\}_{QS} = \left[\Lambda_{EL}\right]^{-1}\left[\Phi_A\right]_{EL}^{T}\left\{f_A(t)\right\}$$

- Dynamic response is total response minus quasi-static

$$\left\{\ddot{q}(t)\right\}_{DYN} = \left\{\ddot{q}(t)\right\} - \left\{\ddot{q}(t)\right\}_{QS} \quad \left\{q(t)\right\}_{DYN} = \left\{q(t)\right\} - \left\{q(t)\right\}_{QS}$$

- Total response recombined with factors

$$\left\{\ddot{q}(t)\right\} = DUF\left\{\ddot{q}(t)\right\}_{DYN} + SUF\left\{\ddot{q}(t)\right\}_{QS}$$

$$\left\{q(t)\right\} = DUF\left\{q(t)\right\}_{DYN} + SUF\left\{q(t)\right\}_{QS}$$

Equivalent Viscous Damping

- Structural (or material or hysteretic) damping is only defined in the frequency domain

$$[K] = [K] + j([K4] + g[K])$$

- Nastran converts structural damping to equivalent viscous at two frequencies (W3 and W4)

$$[B] = \frac{g}{\omega 3}[K] + \frac{1}{\omega 4}[K4]$$

- In modal space we can match damping ratio at all modal resonances

$$[B_{HH}] = g[K_{HH}]^{\frac{1}{2}} + [K_{HH}]^{-\frac{1}{4}}[K4_{HH}][K_{HH}]^{-\frac{1}{4}}$$

Implementation in DMAP

- Use Parameters to Control Solution
  - All parameters have defaults
  - User only needs alter and parameters

```
$     PARAM,NRIG,I          : Number of rigid body modes (default=6)
$     PARAM,AUTOIC,CHAR8    : 'YES' calculate initial conditions (default)
$                             'NO' standard Nastran processing
$     PARAM,EQVDMP,CHAR8    : 'YES' apply equivalent damping algorithm
$                           : 'NO' standard Nastran (default)
$     PARAM,RBDISP,CHAR8    : 'YES' include rigid body displacements
$                             'NO' no rigid body displacements (default)
$     PARAM,NORESV,CHAR8    : 'YES' remove residual vectors from dynamic response (default)
$                             'NO' do not remove residual vectors from dynamic response
$     PARAM,DUF,RS          : Dynamic Uncertainty Factor (default = 1.0)
$     PARAM,SUF,RS          : Static Uncertainty Factor (default = 1.0)
```

- Use string based alter commands
  - Minimize number of locations that are “altered”
  - Try to find “robust” locations for alters
  - Hasn’t broken yet

Enhancements to MFLUID

- Fluid mass matrix based on elemental matrices

$$\left[M_{fluid}\right] = [GEG][MLAM][MCHI]^{-1}[GEG]^{T}$$

- Use non-sparse decomposition of [$MCHI$]
- Calculate fluid rigid body mass

$$\left[M_{fluid}\right]_{RB} = \left[\Phi_{RB}\right]^{T}\left[M_{fluid}\right]\left[\Phi_{RB}\right]$$

- When used with a “Q-set” replaces functionality of PARAM,VMOPT,2 in a competitive version of Nastran

Outline

- What is DMAP?
- Use of DMAP to Select ASET DOF
- Use of DMAP in dynamic analysis
  - Modal transient analysis
  - Fluid mass loading
- Summary/Free stuff!

Summary

- DMAP is a primary differentiator for Nastran vs. other FEM codes
- All NX Nastran (other than advanced nonlinear) is written in DMAP and can be modified using DMAP
- To write DMAP alters you need to understand Nastran data blocks and solution sequences
- Using DMAP is easy if it's written well
  - Use string based alters for robustness
  - Use parameters to control
  - Comment, comment, comment

Free Stuff (DMAP alters) on CD!

- modal_transient
  - Variety of updates to SOL112
- mfluid
  - Calculate fluid mass properties
- readcb
  - Read components in OUTPUT4 format
- unit_conversion
  - Convert units of components
- rcvrcb109
  - Recover component results from direct transient