Nationaal Lucht- en Ruimtevaartlaboratorium
National Aerospace Laboratory NLR

NLR-TP-2002-484

Improvement of the optimisation convergence of B2000/B20PT evaluated for the Ariane 5 LOX fairing design

R.J.C. Creemers and P. Arendsen

Nationaal Lucht- en Ruimtevaartlaboratorium
National Aerospace Laboratory NLR

NLR-TP-2002-484

Improvement of the optimisation convergence of B2000/B20PT evaluated for the Ariane 5 LOX fairing design

R.J.C. Creemers and P. Arendsen

This report is based on a presentation held at the 4th Bi-annual B2000 workshop at Ligerz (Switzerland), 24-25 October 2002.

Customer:	National Aerospace Laboratory NLR
Working Plan number:	S.1.A.4
Owner:	National Aerospace Laboratory NLR
Division:	Structures and Materials
Distribution:	Unlimited
Classification title:	Unclassified
	September 2002

Summary

In the recent past, a composite Ariane 5 LOX line cover was designed and optimised. During the optimisation some specific problems occurred with respect to the convergence of the optimisation process. These problems were associated with distributed/accelerations loads in combination with changing geometry/thickness during the optimisation and with buckling/vibrational modes and loads defined as optimisation constraints. The convergence problems were solved by modifying the input processor, by adding a number of subroutines to the optimisation module and by editing a part of the existing source code. Further, two test cases were developed, showing that the modifications to the source code are important improvements. They resulted in a smoother optimisation process and in some cases a faster convergence.

Contents

28 Figures

(29 pages in total)

List of Abbreviations

CID	Constraint Identification
DV	Design Variable
FEM	Finite Element Model
LOX	Liquid Oxygen
NLR	National Aerospace Laboratory (The Netherlands)

List of Symbols

K	global stiffness matrix
$K_{,x}$	global stiffness gradient
P	global force vector
$P_{,x}$	global force gradient
U	global displacement vector
$U_{,x}$	global displacement gradient
$\vec{v}_i^j$	eigenvector of mode j in optimisation cycle i
x^j	j'th Design Variable

1 Introduction

Nowadays, the Finite Element Method (FEM) is widely used as a solution method for mechanical engineering problems. One of the FEM programmes available at the National Aerospace Laboratory NLR is B2000. This programme has a modular structure, and contains the optimisation module ”B2OPT”, see references 1 and 2. In the recent past several optimisations have been performed with this optimisation module, see for instance references 3 to 5. However, as described in references 4 and 5, certain specific problems occurred during the optimisation of the Ariane 5 LOX line cover. These problems were associated with a distributed pressure load in combination with a changing geometry during the optimisation and with minimum buckling loads defined as optimisation constraints.

This report describes the solution of the above mentioned problems for a wider range of applications by modifying the input processor, by adding a number of subroutines to the optimisation module and by editing part of the existing source code. The aim is to obtain better and faster convergence of the optimisation process.

The report starts with a problem description in chapter 2 where the specific problems encountered during the optimisation of the Ariane LOX line cover are translated to more generic problems. Next, the improvements are presented in chapter 3. The improved optimisation module has been applied to test cases as described in chapter 4 and to the Ariane 5 LOX line cover in chapter 5. Finally, chapter 6 presents the conclusions and recommendations.

2 The optimisation process in B2000: problem definition

In this chapter the problem definition for the optimisation process is presented. First, background information is given and it is shown that the specific problems encountered during the optimisation of the Ariane 5 LOX line cover can be translated to two generic problems in the optimisation module of B2000. The first problem concerns element loads that are changing during the optimisation. The second problem concerns mode-jumping (eigenmode and/or buckling mode) during the optimisation.

2.1 Background

In June 2000, a two-year joint programme, carried out by Dutch Space B.V., Centre of Lightweight Structures, and the National Aerospace Laboratory NLR, was started with the

Fig. 1 The Ariane 5 launch vehicle

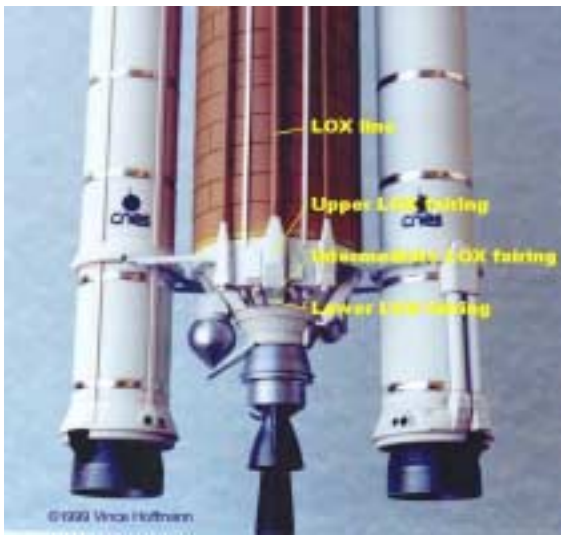

Fig. 2 The covers over the LOX line of Ariane 5

objective to develop, manufacture, and qualify a demonstrator fairing (the upper Liquid OXygen line cover) for the Ariane 5 launch vehicle. Within the joint programme NLR contributed to the preliminary design and was responsible for the design optimisation. The design optimisation was performed with the B2000 code.

The Ariane 5 launch vehicle can be seen in figure 1 and the LOX line covers are shown in figure 2. The outer geometry of the composite redesign of the upper LOX line cover is shown in figure 3 and a FEM model of half the fairing is shown in figure 4. The outer dimensions of the LOX fairing are aproximately 1.4 x 0.7 x 0.5 m. The FEM model shown is a configuration with three hat-stiffeners in the prismatic section and 2 hat-stiffeners in the tapered section (3-2 configuration). A range of configurations was analysed and optimised, and a 4-3 configuration resulted in the optimum design with respect to weight and (recurring) costs. Five different ply thicknesses were defined as design variables in the optimisations. Further, two geometrical variables were defined, i.e., the width and height of the stiffeners. Allowable stresses and a minimum buckling load were applied as design constraints. Initially, the optimisations were performed using estimated material data (Ref. 4). At a later stage B-basis allowables, obtained in

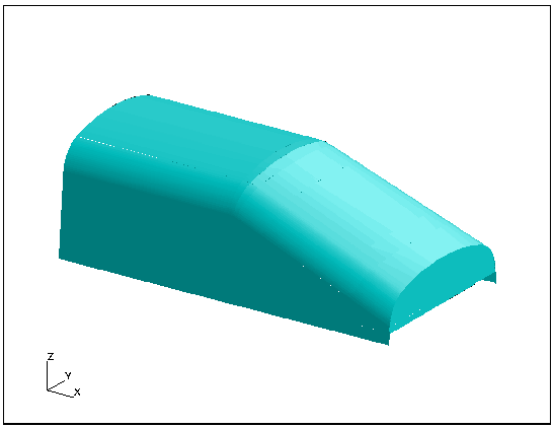

Fig. 3 The outer geometry of the composite fairing

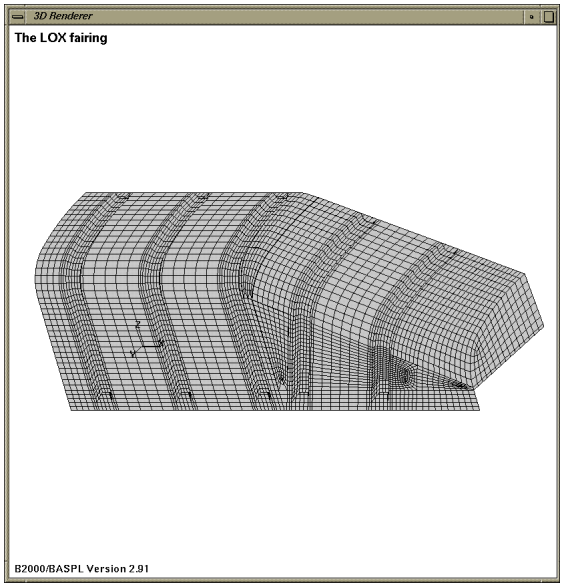

Fig. 4 FEM model of the LOX fairing (3-2 config)

a material test programme, were used (Ref. 5). During these optimisations two specific problems occurred. They are described in the following sections.

2.2 Element loads

For the dimensioning of the LOX fairing, acoustic loads, blast wave loads, and aerodynamic loads are approximated by static distributed pressure loads. In B2000, distributed pressure loads are converted by the input processor 'b2ip' to nodal loads. With these nodal loads the structure is in a state of static equilibrium:

$$K \cdot U = P \tag{1}$$

where K is the global stiffness matrix, U the global displacement vector, and P the global force vector. With P and K known, the resulting displacements and stresses can be calculated. However, if the nodal forces are not adjusted during the optimisation, the change of geometry in the optimisation causes a non-uniform pressure distribution on the LOX fairing (illustrated in figure 5), which in reality is not the case. The vector containing the nodal loads in the FEM calculations is no longer correct, so incorrect displacements/stresses and buckling loads will be obtained. Further, the optimisation uses the derivative of equation 1 with respect to each design variable to calculate the displacement/stress gradients in each optimisation step (Maxi-cycle) for

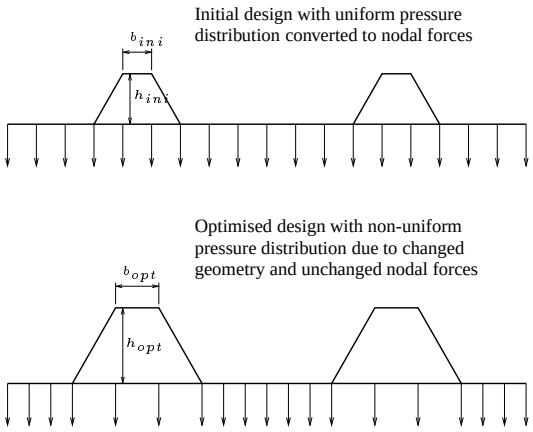

Fig. 5 The nodal loads due to the internal overpressure on the LOX fairing (before and after optimisation)

the determination of search directions:

$$K_{,x} \cdot U + K \cdot U_{,x} = P_{,x} \tag{2}$$

Usually, the right hand term in equation 2 is equal to zero ($P_{,x} = 0$: no change of the global force vector due to change of a design variable), but clearly, this is not true anymore. The nodal forces depend on the geometrical variables (stiffener width and height) and therefore the global force gradient will no longer be equal to zero ($P_{,x} \neq 0$), which influences the search direction of the optimisation.

In references 4 and 5 this problem was solved by restarting the optimisation after a certain number of cycles or after a certain amount of change of the base of the stiffener. When the optimisation is restarted with the changed geometry as initial geometry, the nodal force vector is recalculated for the uniform pressure distribution and therefore is correct again. Figure 6 shows the influence of neglecting the change of nodal forces during the optimisation of the 4-2 configuration. For the optimisation without restarts, Maxi-cycles 1 to 4 all represent infeasible designs. The first feasible design is found at Maxi-cycle 5. Here, the geometrical design variables both reached their maximum values and the above-discussed effect (non-uniform distribution of the pressure) is relatively large. The optimisation without the restart finds an optimum that is close to the first feasible design. In the modified procedure, the base of the stiffener changed more than 15% with respect to the initial design in Maxi-cycles 2 and 4, and here the optimisation was restarted. In that case, the first feasible design is found in Maxi-cycle 3. As can

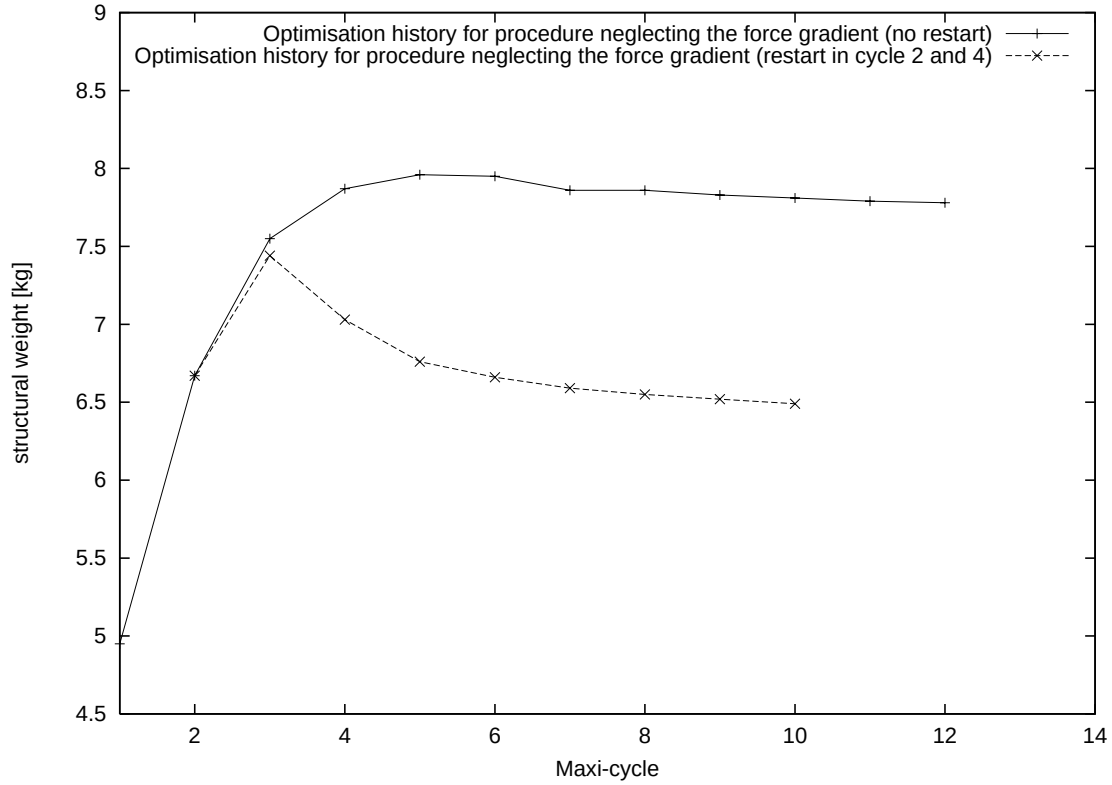

Fig. 6 The optimisation history for the 4-2 configuration

be seen, the optimisation converges faster. The result is a completely different design with a much lower weight. Although the procedure does not take into account the change of the nodal forces at each step, it is comparable to an optimisation procedure which does modify the nodal forces at each step, because when the change of the pressure distribution becomes too large (e.g. 15%) due to the change of the geometrical variables, the nodal forces are recalculated and the pressure distribution is uniform again. This shows clearly that, in general, an optimisation with distributed loads may find the wrong optimum design (and possibly even an infeasible design) when the change of nodal forces due to the changing geometry is neglected. In the particular case of the LOX fairing, the uniform pressure distribution is already an approximation of the acoustic, blast wave, and aerodynamic loads on the fairing (and therefore not exactly equal to the actual loads), but the same arguments still hold. The optimisation may result in an overly conservative design or in an infeasible design.

There are several ways to solve the above presented problem (restarts, linking the design variables to the nodal forces, etc.), but ideally it would be solved by the optimisation itself by recalculating the nodal forces at each optimisation step. This should be done for a number of cases:

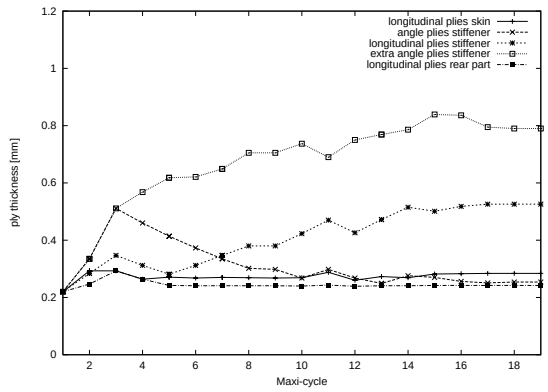

Fig. 7 Ply thicknesses during the optimisation (4-3 config)

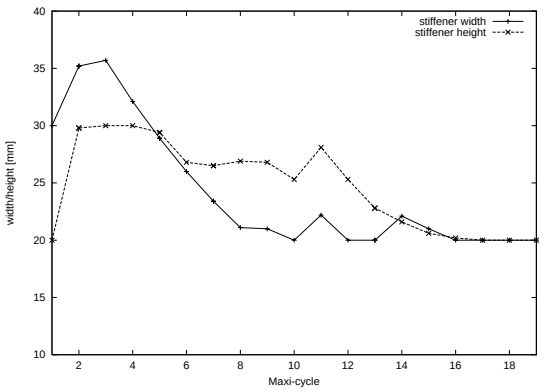

Fig. 8 Stiffener width and height during the optimisation (4-3 config)

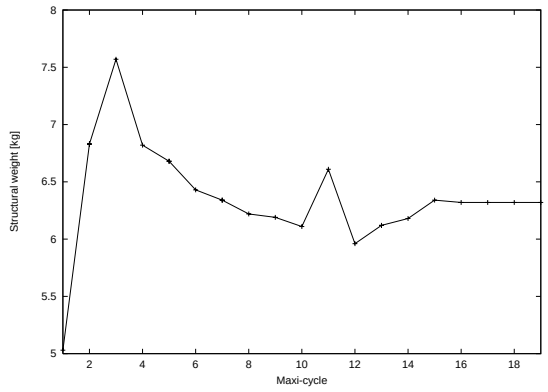

Fig. 9 Structural weight during the optimisation (4-3 config)

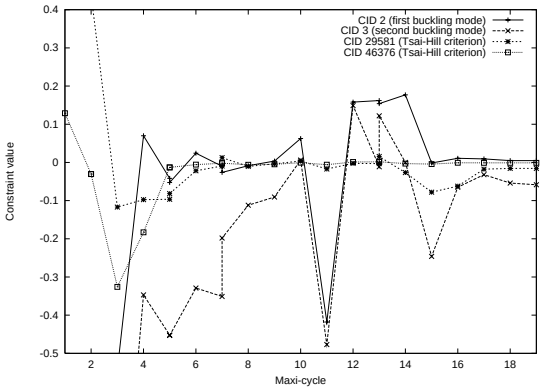

Fig. 10 Constraint values during the optimisation (4-3 config)

- distributed (line or surface) loads in combination with changing geometry;
- acceleration loads in combination with changing geometry;
- acceleration loads in combination with changing thickness;

or in general, any element load that changes during the optimisation due to the change of a design variable.

2.3 Mode-jumping

The procedure with restarts, as presented in the previous section, was applied to the optimisation of the 4-3 configuration of the LOX fairing as well. The optimisation was restarted in Maxi-cycles 2, 5, 7 and 13. The history results of the optimisation can be seen in figures 7 to 10. In these figures ply thicknesses, width and height of the stiffeners, structural weight, and certain relevant constraints of the LOX fairing are shown for each optimisation cycle. The definition of a constraint within B2000 is such, that a positive constraint value indicates a violated constraint

(and thus the design is infeasible). In figure 10 it can be seen that the initial design was an infeasible design. Several stress constraints were violated and the optimisation started with a search for a feasible design (with a constraint violation minimisation). At Maxi-cycle 3 the first feasible design was found, while the weight had increased. From here the optimisation continued with an objective function (weight) optimisation, resulting in an optimised design with a smaller weight and several critical stress and buckling constraints.

As can be seen in the optimisation history, the process had great difficulties with converging between cycles 8 and 13. The phenomenon causing this is discussed here. The optimisation history shows a large variation of the constraint values that correspond to buckling of the structure (CID 1 is the objective or weight of the structure, CID 2 to 5 are the four buckling constraints associated with the four smallest buckling loads). In general, the buckling constraints vary highly non-linear with the design variable (e.g. buckling of the skin between the stiffeners depends on the thickness of the skin to the power three and the distance between the stiffeners to the power two). Therefore, it is important that the second order derivatives of the buckling constraints with respect to the DV's are correct. For the calculation of the second order derivative, the first order derivatives of two consecutive optimisation cycles are used. However, the first buckling constraint (CID 2) corresponds to the buckling mode with the smallest eigen*value* (and therefore largest constraint value), but it does not correspond to a particular buckling *mode*. Therefore, from one optimisation cycle to another these buckling modes can exchange CID's (hereafter referred to as mode jumping). The second buckling mode (CID 3) in the first cycle might get the smallest eigenvalue in the second cycle due to the changing geometry and thicknesses and therefore it will become the first buckling mode (CID 2) in the second cycle. Now, when the second order derivative of CID 2 is determined, the first order derivatives of two non-equal buckling modes in the two consecutive optimisation cycles are used. Then, the results for the second order derivative will not be correct, thereby misguiding the optimisation. A detailed view of the first three buckling constraints(CID 2, 3 and 4) and buckling modes showed that mode-jumping repeatedly occurred during cycles 8 and 13. The three modes involved are shown in figures 11 to 13. The constraint values corresponding to a particular mode shape are plotted in figure 14. It can be seen that mode shape 3 in cycle 9 (CID 4) gets the smallest eigenvalue (and largest constraint value) in cycle 10 and therefore gets CID 2. Further, mode shape 1 gets CID 3, and mode shape 2 gets CID 4. Therefore, the second order derivatives of these constraints will not be correct. For the optimisation of the LOX fairing the following method was used to overcome this problem. Starting from cycle 13, the relative variations of the DV's were allowed to be only small (5% in stead of 10%). Then, for the optimisation a correct second order derivative is less important compared to the first order derivative, and a relatively

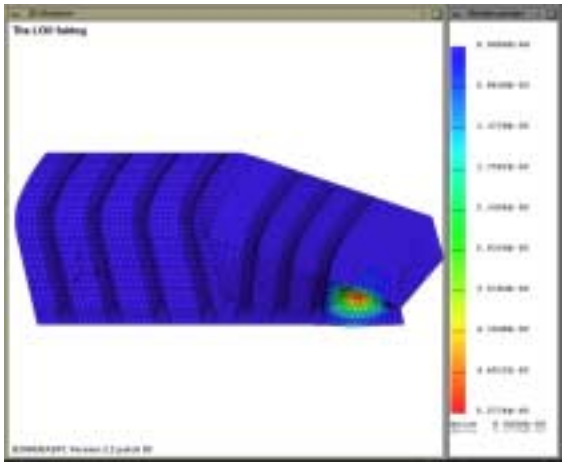

Fig. 11 Mode shape 1: skin buckling lower section

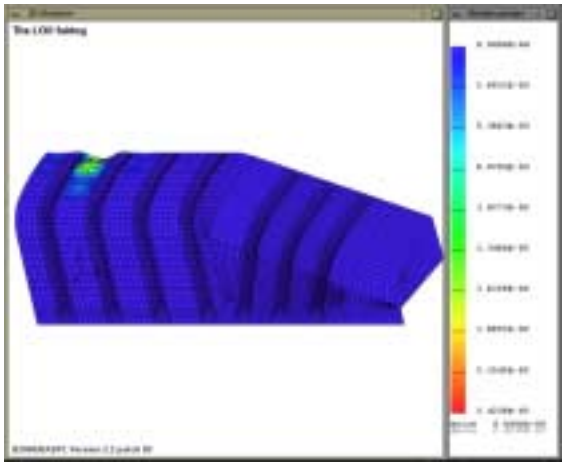

Fig. 12 Mode shape 2: skin buckling upper section

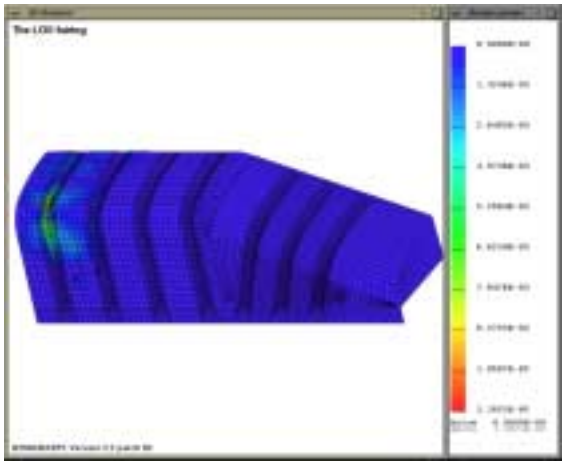

Fig. 13 Mode shape 3: stiffener buckling upper section

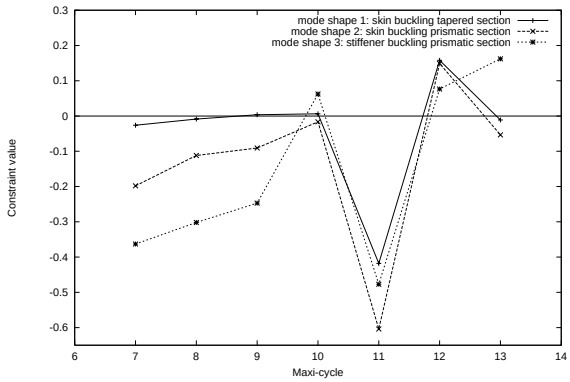

Fig. 14 Constraint values during the optimisation (4-3 config)

quick convergence was the result, see figures 7 to 10.

In the above presented solution the second order derivatives can still be incorrect. Only their influence on the optimisation process has been decreased by allowing smaller variations of the design variables (with the possible disadvantage of necessitating more steps to find the optimum design). To get correct second order derivatives for each CID, mode tracking would have to be incorporated in the optimisation. This should be done for the following constraint types:

- linear buckling load constraint
- linear buckling mode constraint
- vibrational eigenfrequency constraint

- vibrational eigenmode constraint

This will generally result in a smoother and faster optimisation process.

3 The optimisation process in B2000: improvements

The optimisation convergence problems discussed in the previous chapter have been solved by modifying the input processor, by adding a number of subroutines to the optimisation module and by editing some of the existing source code. Here, these improvements are discussed.

3.1 Element loads

As said before, in B2000 element loads are converted by the input processor ’b2ip’ to nodal loads. These nodal loads are saved on database and the original element loads (line loads, pressure distribution, acceleration loads, etc.) are discarded and not saved on database. This is correct for most analysis types and saves storage capacity and computational effort (the element loads are converted to nodal loads only once). However, in case of an optimisation process with changing geometry/thickness in combination with distributed/accelerations loads these nodal loads may have to be adjusted from one cycle to another (see section 2.2 as well). Therefore, the input processor ’b2ip’ has been modified, a procedure for adjusting the nodal loads during the optimisation has been developed, and modifications to the sensitivity analysis to ensure the correct calculation of derivatives in case of element loads depending on design variables are made. These features are described here.

3.1.1 Modified b2ip processor

The modification of b2ip consists of the additional storage of elemental loads. The specification of these loads (type, coordinates system, etc.) is stored in a data set ”ELOAD.branch-number.0.0.load-case”. Based upon the element loads, nodal forces are calculated as before and added to possibly already existing nodal forces (BFRC data set). The storage of this data is used later on in the B2OPT processor to distinguish between nodal forces coming from a nodal specification and nodal forces derived from element loads. For compatibility reasons, the element loads are NOT separated from the ’pure’ nodal forces.

3.1.2 Nodal force updates

At the beginning of an optimisation run within module ’opram’, nodal forces are calculated from the elemental loads only, as defined in the initial ELOAD data set, and stored in a data set ”BFRC-E.branch-number.maxi-cycle.0.load-case”. In subsequent optimisation runs these nodal

forces are subtracted from the total nodal forces. The element loads are for each optimisation run converted to nodal forces, which are added to the total. Notice that the nodal forces derived from the element loads differ from one optimisation step to another, so the element loads of the previous step is to be used within the subtraction. Matters are a bit complicated, since the first model to analyse in B2OPT is not always the same as the model given by 'b2ip', due to design variable offsets. Both nodal forces and element loads could themselves vary as a function of the design variables, but in principle this subtraction/addition is all that is needed to enable adjusted element loads as the result of varying geometry.

3.1.3 Sensitivity analysis adjustment

The calculation of force derivatives $P_{,x}$ is done using a semi-analytical approach. This approach is also used in the calculation of derivatives of stiffness matrices $K_{,x}$, as implemented in routine 'opfdep.F'. The calculation of nodal forces due to element loads (routine 'opforc.F') has been developed to enable the improvements in the previous subsection. The combination of these two routines enabled the quick development of a routine 'opfdforc.F' which calculates the derivative of nodal forces due to element loads.

3.2 Mode-jumping

As shown in section 2.3, mode jumping during the optimisation may cause bad convergence of the optimisation process when no mode-tracking is incorporated. The mode jumps cause errors in the calculation of the second derivatives of the buckling/eigenfrequency constraints with respect to the design variables. Therefore, mode-tracking has been incorporated in order to use the first order derivatives of a certain mode *shape* in the calculation, instead of the first order derivatives belonging to certain *CID number*.

Mode-tracking can be divided in two separate actions, that have to be performed during optimisation. First, a list or dataset will be created, containing for each mode of the current optimisation cycle the corresponding mode in the previous cycle. Next, this list will be used to adapt the optimisation such, that the second order derivatives of the constraints associated with these modes are calculated with the correct first order derivatives of both cycles. These two steps are presented below.

Although mode-tracking has been implemented in the optimisation, it is still possible to perform an optimisation without mode-tracking. It can be switched of by the command "MODETRACK OFF" (default is "MODETRACK ON").

3.2.1 Mode order determination

In the buckling/eigenfrequency calculations, the eigenvectors of each mode are determined. In each optimisation cycle, these eigenvectors are saved on database in the datasets "BMOD.GLOB.maxi-cycle.load-case.mode-number" (linear buckling modes) and/or "VMOD.GLOB.maxi-cycle.0.mode-number" (vibrational eigenmodes). The eigenvector of a certain mode is always perpendicular to the eigenvector of any other mode. Then, the scalar product of the eigenvector of a certain mode with the eigenvector of any other mode will be equal to zero, but also, the scalar product of the eigenvector of a certain mode with itself will be non-zero. This feature is used to find similar mode shapes in two consecutive cycles. For every load case, the scalar product can be calculated for the eigenvector of mode 'j' in optimisation cycle 'i' with the eigenvector of mode 'k' in optimisation cycle 'i-1' (the previous optimisation cycle):

$$scpr_{i,j,k} = \frac{\vec{v}_i^j \cdot \vec{v}_{i-1}^k}{\left|\vec{v}_i^j\right|} \tag{3}$$

in which $scpr$ is the normalised result of the scalar product. When the result of the scalar product is zero (or close to zero) the mode 'j' in the current cycle is not similar to mode 'k' in the previous cycle, when the result is one (or close to one) two similar modes have been found. Using this, for each mode in the current cycle, the similar mode in the previous cycle can be found. For instance, buckling mode 1 in cycle 5 is similar to buckling mode 3 in cycle 4 (the previous cycle) when the normalised result of the scalar product for the eigenvector of mode 1 in cycle 5 ($\vec{v}_5^1$) with mode 3 in cycle 4 ($\vec{v}_4^3$) is (close to) one. As a consequence, the normalised result of the scalar product for the eigenvector of mode 1 in cycle 5 with any other mode in cycle 4 will be (close to) zero, as these modes cannot be similar as well. When no similar modes in the previous cycle can be found for a certain mode, apparently a new mode has been found. Due to the changing geometry or thickness of the structure a mode appears in the current cycle, which was not present in the structure in the previous cycle.

The above presented principle has been implemented as follows. The subroutine "oplstmodor.F" has been added to the structural response control module "opsrcm.F". A flowchart of "oplstmodor.F" can be seen in figure 15. In the current cycle 'i', the subroutine calculates the scalar product for the eigenvector of mode 'j' with the eigenvector of each mode 'k' in the previous cycle 'i-1'. In addition, the maximum normalised scalar product ('scprmax') and the associated mode are determined. Next, this maximum value is compared to some preset values. If

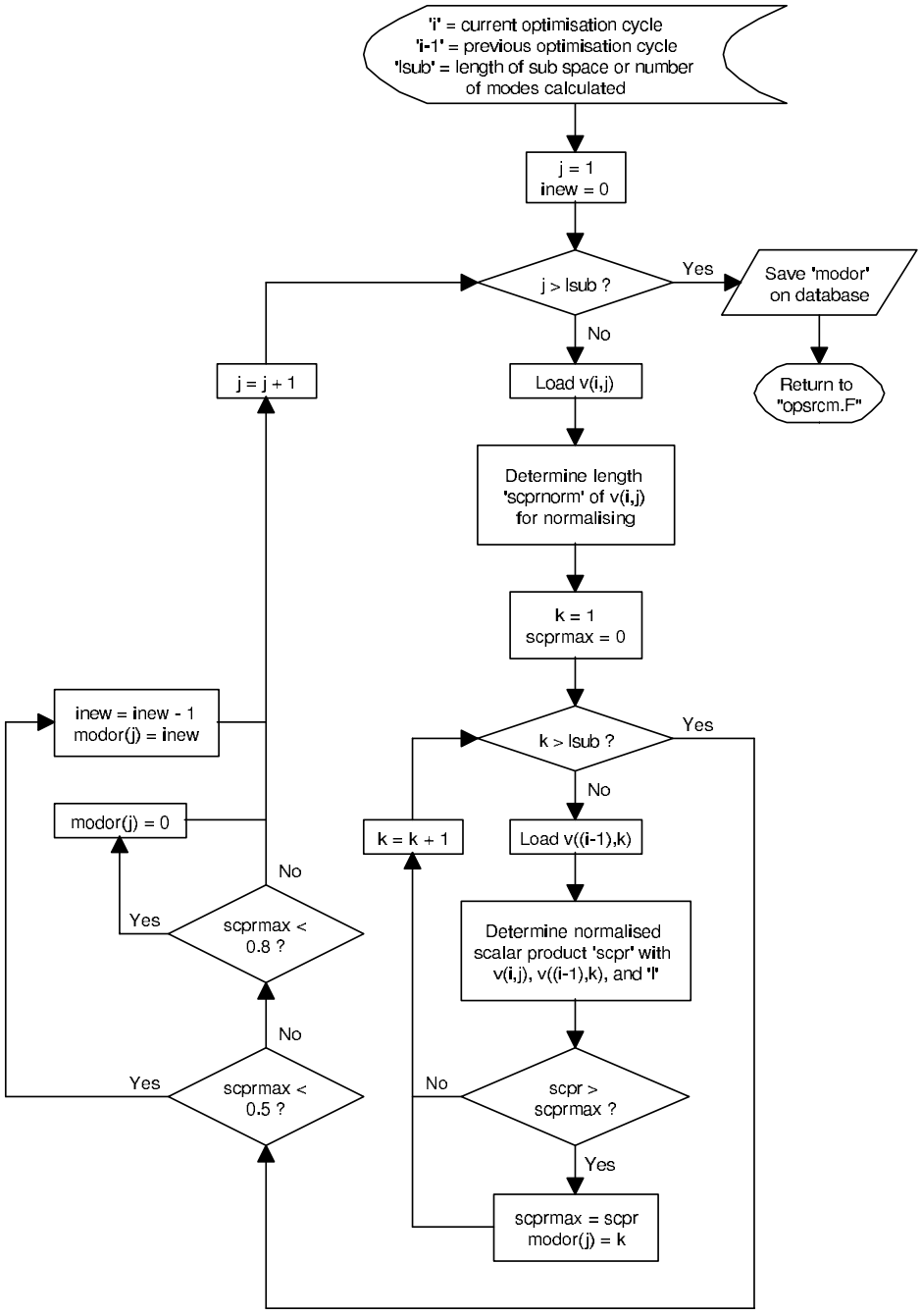

Fig. 15 The subroutine "oplstmodor.F"

'scprmax' is close to zero (e.g. lower than 0.5) the current mode 'j' does not correspond to any mode 'k' in the previous cycle, so the current mode 'j' is a new mode. If 'scprmax' is close to one (e.g. higher than 0.8) the current mode 'j' corresponds to the mode 'k' associated with this maximum scalar product, so the corresponding mode has been found. For any values of

```
Data set "BMODOR.LACY.4.0.1", idiv      0, type "I   ", size       10
   print from       1 to      10 step       1

     1     5     2     3     4     0    -1     8     9    -2
```

Fig. 16 The dataset BMODOR.LACY for Maxi-cycle 4 and load case 1

'scprmax' in between these preset values ($0.5 < scprmax < 0.8$) one cannot be confident whether a new mode or a corresponding mode has been found. This procedure is applied to each mode 'j' and the information is saved on database. The dataset containing the buckling mode order is called "BMODOR.LACY.maxi-cycle.0.load-case". The dataset containing the vibrational mode order is called "VMODOR.LACY.maxi-cycle". In figure 16 an example is shown of the dataset "BMODOR.LACY" for Maxi-cycle 4 and load case 1. The data set shows that buckling mode 1 in the current Maxi-cycle (cycle 4) corresponds to buckling mode 1 in the previous Maxi-cycle (cycle 3), so no mode-jumping occurs for the first buckling mode. However, mode 2 in cycle 4 correspond to mode 5 in cycle 3. Mode 3 in cycle 4 corresponds to Mode 2 in cycle 3, etc. When a new mode is found, this is indicated by a negative number, so -1 on place 7 in the data set means that the seventh mode in cycle 4 is the first buckling mode for which no similar mode in cycle 3 could be found. It is the first new mode in cycle 4. Similarly, the tenth mode in cycle 4 is the second new buckling mode. Further, for the sixth mode in cycle 4 the corresponding mode in cycle 3 could not be determined with high enough confidence. It might be a new mode as well, but one cannot be sure. This is indicated with a zero and it can be caused by a number of reasons. As said before, it may be an indication of a new mode after all. Another possibility is that the vibration or buckling analysis has not reached convergence in either one of the two consecutive cycles and therefore the eigenvectors of the modes have not been determined with enough accuracy. However, the most common reason is a large change of design variables from one cycle to another causing (small) variations in the eigenvectors of a particular mode in the two consecutive cycles. This is especially the case, when geometrical variables have been applied changing the nodal coordinates of the model during the optimisation. This may affect the eigenvectors as well. An example is shown in section 4.2

3.2.2 Calculation of second order derivatives

The optimisation process has been adapted such, that in each cycle 'i' a list is generated which contains for each mode 'j' the corresponding mode 'k' in the previous cycle. Now this list has to be used for the determination of the correct second order derivatives of the buckling/eigenfrequency constraints. This is done by the subroutine "opsetmodor.F" and the control module "opcamcm.F". This is described here.

```
Data set "NACON.4", idiv      0, type "I   ", size       12
   print from       1 to      12 step       1

     1     2     3    -4    -5     0     0     0     0     0
     0     0
```

Fig. 17 The dataset NACON for Maxi-cycle 4

The second order derivatives are calculated in ”opcamcm.F”, but this is done only for certain constraints. In the dataset ”NACON.maxi-cycle” the near active constraints are stored for each optimisation cycle, see for instance figure 17. One can see twelve numbers in the dataset, one number for each constraint (so apparently twelve constraints have been defined). A zero indicates that the constraint is not active, a number indicates what active constraint it is. Further, for the positive numbers both the first order derivatives and the second order derivatives (Hessian matrix) have to be calculated, for the negative numbers only the first order derivatives have to be calculated (to reduce storage). So -5 on the sixth place in ”NACON.4” means that in cycle 4 the sixth constraint (CID 6) is the fifth active constraint and that only the first order derivative has to be determined. The way the second order derivatives are determined (for constraints one, two, and three) depends on the constraint information available from the previous cycle, which in this case would be indicated by ”NACON.3”, see figure 18. Again, a positive number in ”NACON.3” indicates that a Hessian is available, a negative number indicates that only a first order derivative is available, and a zero indicates that the constraint was not active in cycle 3. However, in the previous chapter it has been shown that buckling constraints may exchange CID number from one cycle to another. Therefore, a dataset ”NACALT” will be constructed internally, in which the values of NACON.3 associated with buckling constraints also exchange places according to the mode orders of ”BMODOR.4.0.1”. Next, ”NACALT” instead of ”NACON.3” is used by ”opcamcm.F” to find the information needed for determining the second order derivatives (or Hessian matrix).

The construction of ”NACALT” is performed in each cycle. In each cycle the dataset ”NACON” of the previous cycle is copied to the dataset ”NACALT” by ”opcamcm.F”. Next, the subroutine ”opsetmodor.F” is called. The only purpose it has is to exchange places of the values associated with a buckling constraint. A flowchart can be seen in figure 19. The programme goes through

```
Data set "NACON.3", idiv      0, type "I   ", size       12
   print from       1 to      12 step       1

     1     2     3    -4     0     0     0     0     0     0
     0     0
```

Fig. 18 The dataset NACON for Maxi-cycle 3

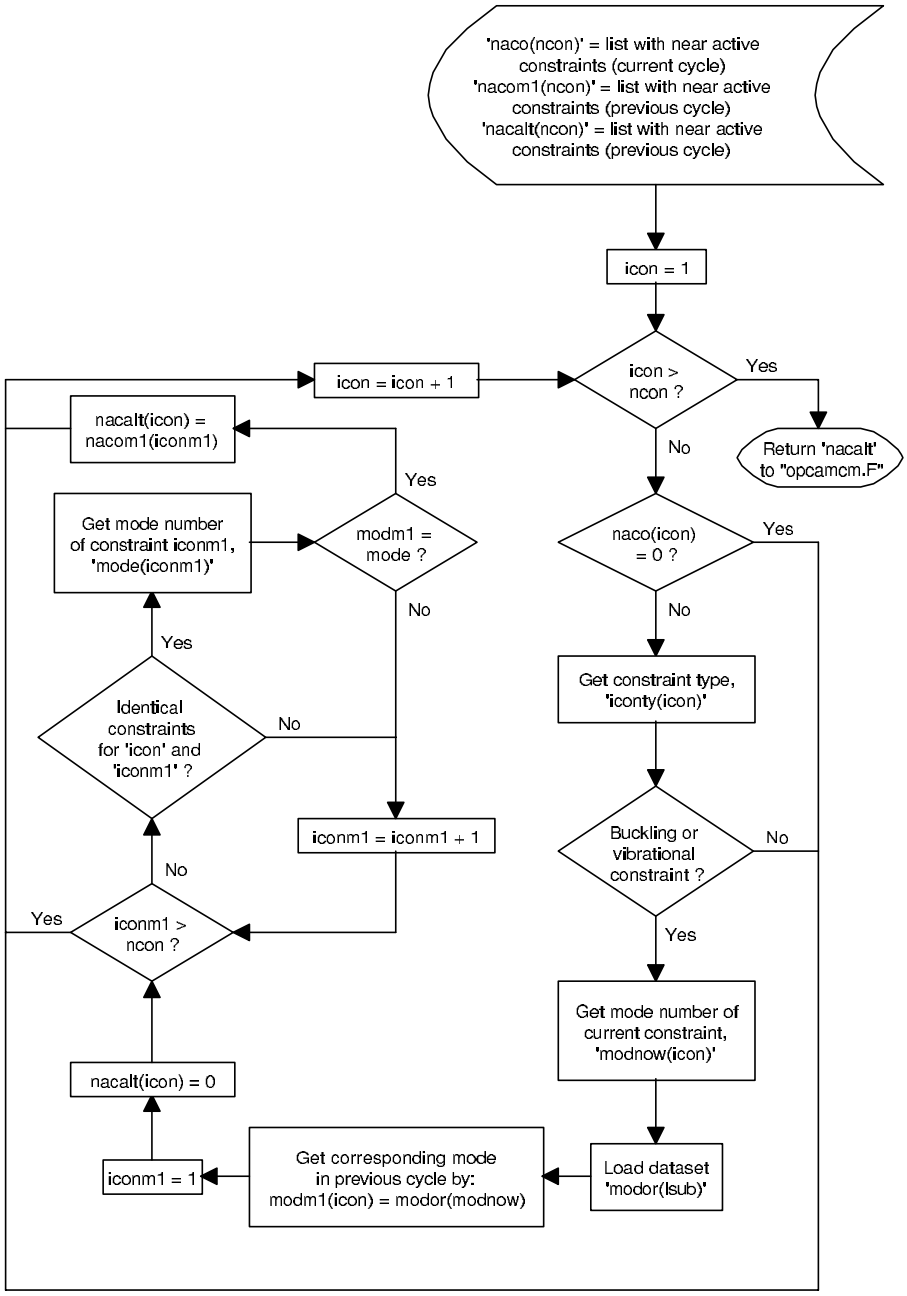

Fig. 19 The subroutine "opsetmodor.F"

the list of near active constraints (the outer loop) in the current optimisation cycle, ’naco(icon)’. When an active buckling/vibrational constraint is found, the mode number associated with this constraint is read from database, ’modnow(icon)’. With the dataset ’modor’ (”BMODOR.LACY” or ”VMODOR.LACY”) the corresponding mode in the previous cycle can be determined, ’modm1(icon)’. Next, a search is started (the inner loop) to find an identical constraint in the list

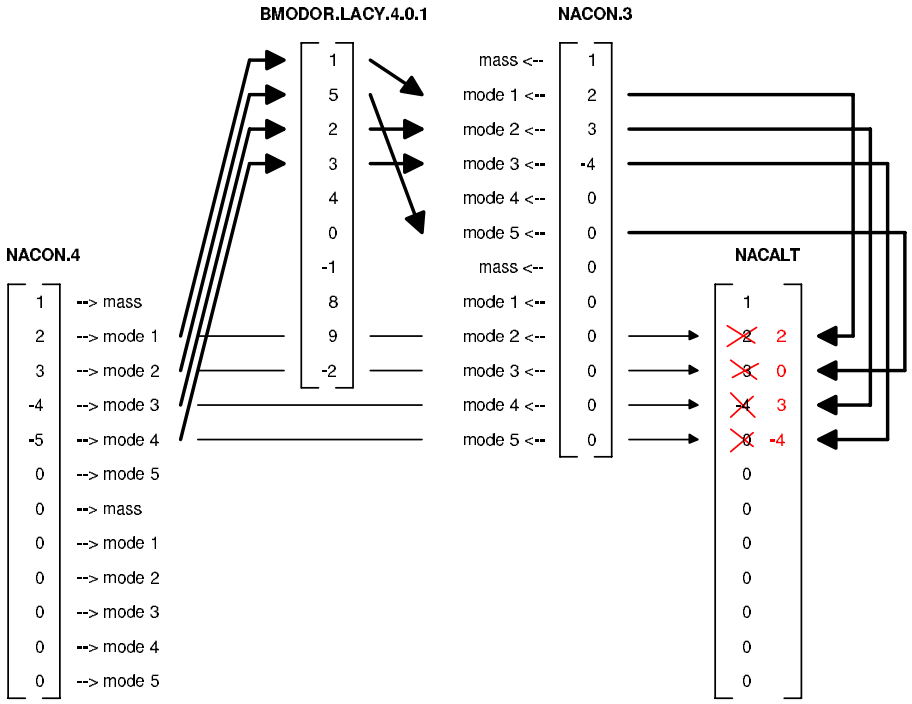

Fig. 20 A graphical example of the subroutine "opsetmodor.F"

of near active constraints in the previous optimisation cycle, ’nacom1(iconm1)’. Identical means that the constraints are of the same type and load case, that they are calculated according to the same method with the same scaling factor and upper/lower limits, etc. Further, the mode number of this constraint ’mode(iconm1)’ has be equal to ’modm1(icon)’. If all these conditions are satisfied, the constraint ’nacom1(iconm1)’ in the previous cycle corresponds to the constraint ’naco(icon)’ in the current cycle. Then, the number associated to ’nacom1(iconm1)’ is stored in the altered dataset ’nacalt(icon)’. If no corresponding constraint could be found (because for instance the mode in the current cycle is a new mode and therefore it does not have a corresponding mode in the previous cycle), a zero is stored in ’nacalt(icon)’. As said before, a zero indicates that no first or second order derivative is available (for this constraint in the previous cycle), which is true, because no corresponding constraint could be found.

An example of the process is displayed graphically in figure 20. The datasets of figures 16 to 18 have been used in this example again. Further, the constraint type (mass or linear buckling load constraint) is indicated for each constraint. The first and seventh constraint (CID 1 and 7) are mass constraints. CID 2 to 6 and 8 to 12 are linear buckling load constraints for which the mode numbers are shown. Although the constraint types of CID 1 to 6 (and the modes involved) are equal to those of CID 7 to 12, they are not the same due to different allowables (upper and lower

limits respectively) and only CID 1 to 5 are active. Now look for instance at CID 3. In cycle 4 it is the third active constraint (see ”NACON.4”) and it is associated to a linear buckling load constraint (load case 1) imposed on mode 2. as can be seen in ”BMODOR.LACY.4.0.1”, mode 2 in cycle 4 corresponds to mode 5 in the previous cycle (cycle 3). For mode 5 an identical constraint has been specified as for mode 2. This is CID 6 in ”NACON.3” and it can be seen that the constraint was not active in cycle 3, so no first or second order derivatives are available. The result is a zero being stored in ”NACALT” for CID 3. Now, for the calculation of the second order derivatives of CID 3 in cycle 4 by ”opcamcm.F” no constraint information of the previous cycle is available, but to have no information will (in most cases) be better than to have wrong information, as would be the case when ”NACON.3” would have been used.

4 Test cases for the improved optimisation process

For both improvements (adapting element loads and mode-tracking during optimisation) test cases have been developed. These are described here.

4.1 Element loads

A single 4-node element test has been developed. This test consists of a square element, height and width equal to 1.0. The element is clamped at one side (with Poisson contraction freedom) and loaded in five different ways:

- normal face loading on the element (value 4.0),
- gravity loading normal to the element (value 4000.0),
- in plane line loading on the edge opposite to the clamping side, normal to cross section (value 2.0).
- nodal loading normal to the surface opposite to the clamping side (value 1.0 per node).
- in plane nodal loading opposite to the clamping side normal to cross section (value 1.0 per node).

The material of the plate is isotropic with a Young’s modulus of 1000.0, a density of 1.0 and a thickness of 0.001. The element loads are converted to nodal loads and the free nodes get an equal share of these loads. As design variable (DV) the width of the element is chosen, starting with value 1.0 such, that if the design variable value doubles, the width of the plate doubles. If the width of the plate doubles, its stiffness doubles too. For the first three load cases this increase in stiffness is matched by the doubling of the nodal forces, since the area on which the element loads act doubles. This implies that these displacements do NOT change as a function of the design variable.

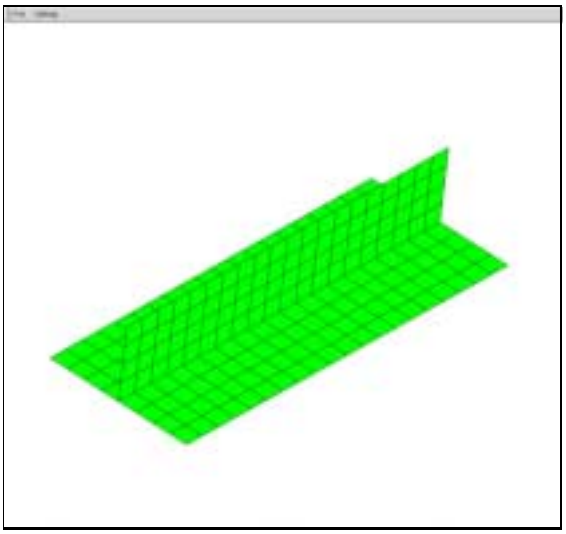

Fig. 21 The FEM model of the blade stiffener

The bending displacement for load cases 1,2 and 4 of the free node in the first cycle (DV=1.0) equals 5346606.38407939300, while in a following cycle (DV=0.9523803591) this displacement equals to 5346606.38478032593 for load cases 1 and 2 and 5613940.20076606888 for load case 4. The in plane displacement for load cases 3 and 5 of the free node in the first cycle (DV=1.0) equals 2.0, while in a following cycle (DV=0.9523803591) this displacement equals to 2.0 for load case 3 and 2.1 for load case 5. As one can see clearly from the results, the displacements do not change in case of element loading and they do change linearly with respect to the inverse of the design variable in case of nodal loading. As an extra check, the gradient of the displacement is checked for the first cycle. For the first three load cases the values of these gradients are 6 orders of magnitude smaller than the corresponding displacements, so they are indeed practically zero. For the last two load cases these gradients equal the displacements almost exactly, including a minus sign of course. From this test it is concluded that the implementation of the element loads is correct with respect to varying geometries.

4.2 Mode-jumping

To test mode-tracking during the optimisation, it has to be ensured that mode jumps occur from one cycle to another. Therefore, a FEM model of a blade stiffened skin strip has been developed, see figure 21. The structure is loaded in axial compression. During the optimisation, the height and thickness of the stiffener and the thickness of the skin are design variables. The objective is minimum weight of the structure. A minimum buckling load is imposed on the first five buckling modes.

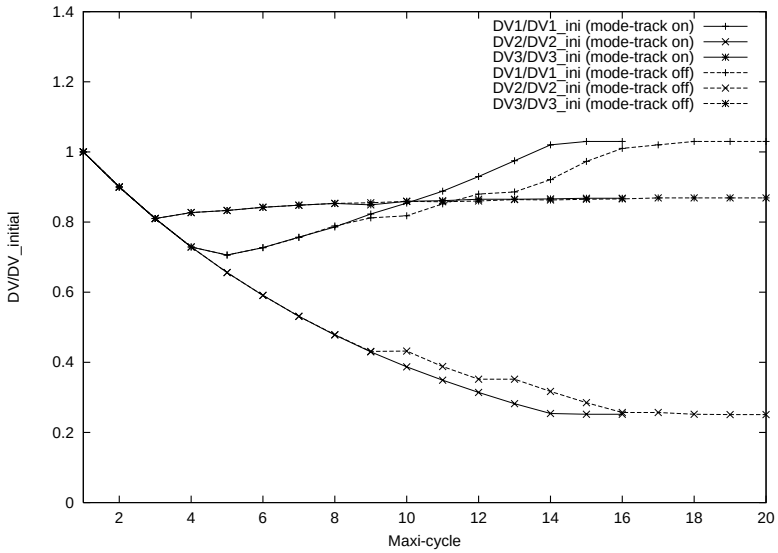

Fig. 22 The optimisation history for the blade stiffener

Two optimisations have been performed on this model, one without mode-tracking and one with mode-tracking. The results can be seen in figure 22. The dashed line is the old optimisation process without mode-tracking and the solid line is the new optimisation with mode-tracking. DV 1 is the stiffener height, DV 2 is the stiffener thickness, and DV 3 is the skin thickness. It can be seen that the optimisation with mode-tracking converges faster and that the behaviour is smoother. The first mode jump of a buckling constraint which is active occurs in cycle 6. The influence on the optimisation is negligible. However, the second mode jump for an active buckling constraint occurs in cycle 9 and the influence on the optimisation is clearly visible. The search direction (for minimum weight) is completely different for the optimisation without mode-tracking, so in cycle 10 quite a different design is obtained. In the next couple of cycles the optimisation recuperates because no mode jumps occur, but every time a new mode jump occurs for an active buckling constraint the optimisation searches in the wrong direction leading to extra optimisation cycles needed for convergence. The test case clearly shows the benefits of mode-tracking, because the computing effort involved with mode-tracking is much smaller than the computing effort for an extra optimisation cycle.

In figure 23 the dataset ”BMODOR.LACY” is shown for cycle 4. For the sixth buckling mode, it

```
Data set "BMODOR.LACY.4.0.1", idiv      0, type "I   ", size       10
   print from       1 to      10 step       1

     1     2     5     3     4     0     0     8     9    10
```

Fig. 23 The dataset BMODOR.LACY in cycle 4 for the blade stiffener

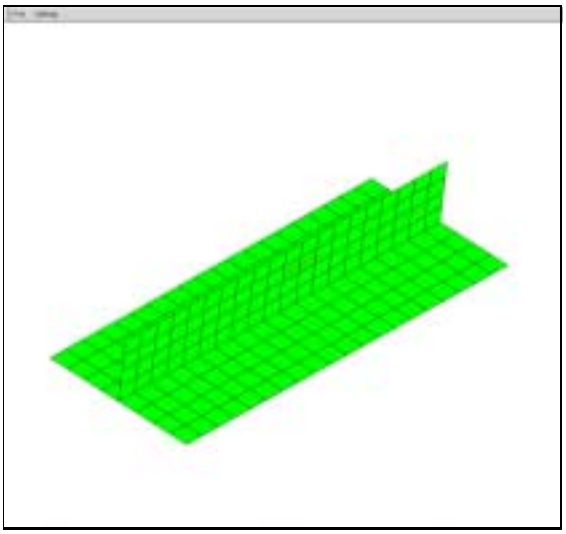

Fig. 24 The FEM model in cycle 3

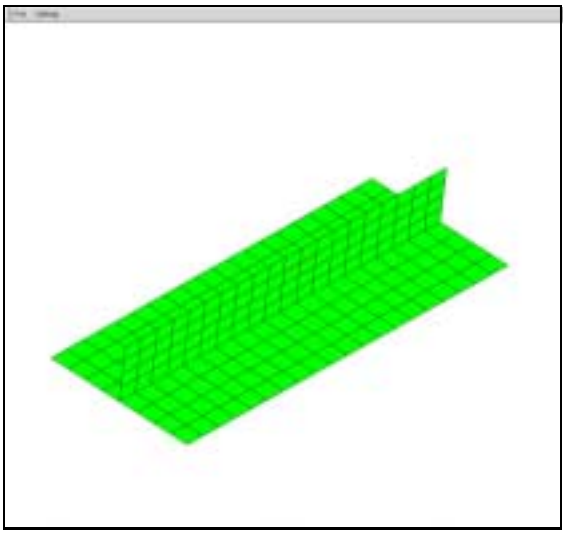

Fig. 25 The FEM model in cycle 4

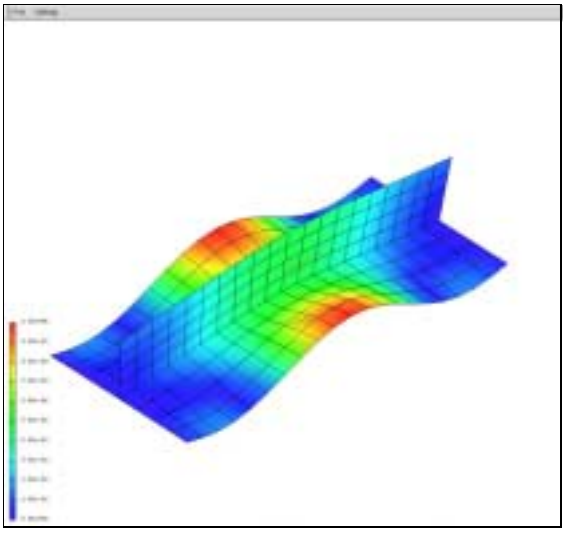

Fig. 26 Mode shape 6, cycle 3

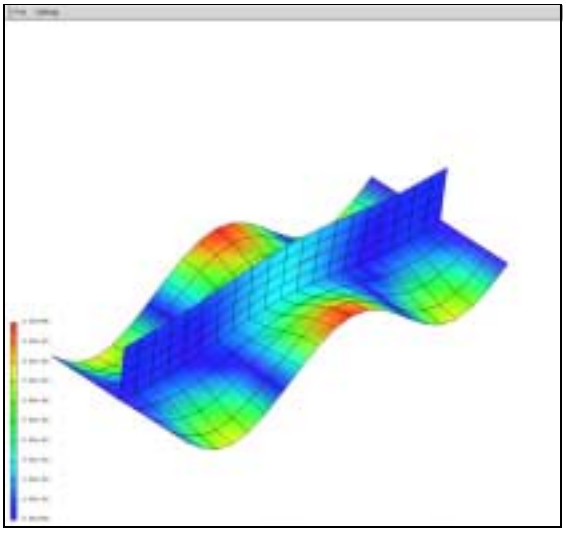

Fig. 27 Mode shape 6, cycle 4

could not be determined with enough confidence whether a corresponding mode in cycle 3 was found or a new mode. As said before, a possible reason for this is the large change of design variables from one cycle to another causing (small) variations in the eigenvectors of a particular mode in the two consecutive cycles, especially when geometrical variables have been applied. This is shown here. In figures 24 and 25, the FEM models of the blade stiffener are shown in cycles 3 and 4. It can be seen that the stiffener height has decreased. Further, also the stiffener thickness has decreased from cycle 3 to 4, while the skin thickness has remained approximately the same (see figure 22 as well). The corresponding buckling modes (mode 6) in both cycles are

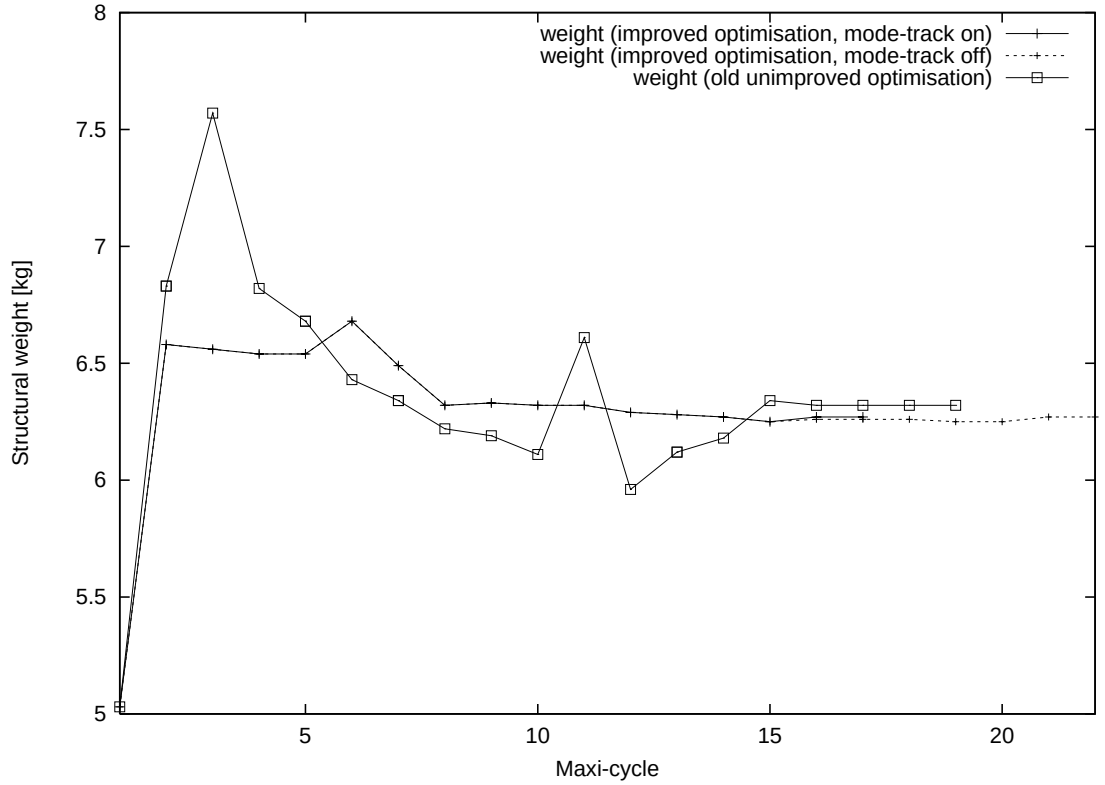

Fig. 28 The structural weight of the LOX line cover (4-3 configuration) for the improved and unimproved optimisation

shown in figures 26 and 27. It can be seen that, although the two modes can be considered similar, the eigenvectors are quite different, resulting in a normalised scalar product below 0.8. Of course, the requirement could be relaxed, but in that case the chance increases of finding a reference mode for a mode that is actually a new mode. Then, for two non-equal modes in the current cycle the same reference mode in the previous cycle could be found, which cannot be true. The value of 0.8 is a compromise between these two features.

5 The improved optimisation process applied to the Ariane 5 LOX line cover

The improved optimisation process incorporating adjusting element loads and/or mode-tracking has been applied to the Ariane 5 LOX line cover of references 4 and 5. In figure 28 the structural weight of the LOX line cover is shown for the improved optimisation with mode-tracking. Further, the old (unimproved) optimisation of chapter 2 is shown as well. A comparison between the improved optimisation with mode-tracking and the old optimisation (without adjusting element loads and mode-tracking), shows that the improved optimisation converges a little bit

faster, but also (and what is more important) that the behaviour is smoother without having to restart the optimisation very often with a new initial design to correct the loads on the cover again. The ply thicknesses and geometric variables are not shown here. It suffices to mention that a slightly different optimum design is found for the improved optimisation. As in many optimisations, there are several local optima to be found for design of the LOX line cover. Due to alterations to the B2000 code (e.g. the implementation of adjusting element loads), the optimisation procedure follows a different optimisation path, leading to a different optimum. Fortunately, the values of the objective (or structural weight) are very close to each other for both local optima. Apparently it is possible that several feasible designs result in more or less the same structural weight for the composite LOX line cover. This offers some flexibility to the designer.

Another optimisation has been performed with the improved B2000 optimisation code in which the mode-tracking feature was switched off. The adjusting element loads are still incorporated in the optimisation process, so no restarts are necessary. These results are also shown in figure 28 and are discussed here briefly. During the optimisation, mode-jumping occurred repeatedly, but until cycle 15 not for buckling constraints that were active in two consecutive optimisation cycles. Therefore, until cycle 15 the optimisation process without mode-tracking is exactly the same as the optimisation process with mode-tracking. From cycle 14 to 15 a mode-jump occurs. The first two buckling modes (with CID 2 and 3), which are active in both cycle 14 and 15, switch places. The optimisation without mode-tracking does not take this into account and a slightly different design is obtained in cycle 16. Because convergence has almost been achieved at cycle 15, the DV change is only small and therefore the effect on the design is not as pronounced as in the old (unimproved) optimisation process (see the example of section 4.2 as well). However, it does lead to an extra number of five optimisation cycles before convergence is achieved compared to the optimisation with mode-tracking.

6 Conclusions and recommendations

The treatment of varying nodal loads due to model changes which affect element loads has been implemented and tested. Further, mode-tracking has been incorporated in the optimisation. As confirmed by figure 28, it can be stated that a better prediction of constraint values and their gradients leads to a smoother optimisation history. Smoother optimisation histories are not always a guarantee for faster convergence, but normally they do shorten the number of needed design cycles (less computing effort).

A future enhancement with respect to mode-tracking is the tracking of a particular mode throughout the entire optimisation process. In that case, different constraint values can be assigned to different mode *shapes* instead of mode *numbers*. This can be helpful, when different constraints apply to different modes (e.g. local vs. global).

7 References

1. Arendsen, P.; *The B2000 Optimization Module: B2OPT,* NLR-TP-94116, National Aerospace Laboratory NLR, Amsterdam, The Netherlands, 1994.
2. Arendsen, P.; *B2OPT Processors and Input Description,* NLR-CR-95179, National Aerospace Laboratory NLR, Amsterdam, The Netherlands, 1995.
3. Arendsen, P.; *Optimization Study of the Window Area for an Ultra High Capacity Aircraft,* NLR-TR-98422, National Aerospace Laboratory NLR, Amsterdam, The Netherlands, 1998.
4. Creemers, R.J.C.; *Design and optimisation of an Ariane LOX line cover,* NLR-TP-2001-150, National Aerospace Laboratory NLR, Amsterdam, The Netherlands, 2001.
5. Creemers, R.J.C.; *Design and optimisation of an Ariane 5 LOX line cover,* NLR-TP-2002-415, National Aerospace Laboratory NLR, Amsterdam, The Netherlands, 2002.