

Minimum weight design of an axially compressed isotropic prismatic panel consisting of a series of cylindrical segments and verification by STAGS

David Bushnell

775 Northampton Dr., Palo Alto, CA, 94303, Ph.D, AIAA Fellow, email: bush@sonic.net

Michael S. Jacoby

Senior Staff Engineer, Lockheed Martin Missiles and Space, Palo Alto, California 94304, AIAA Member

We want to determine the minimum amount of material required to span a distance of width = WIDTH for an axially compressed, unstiffened, prismatic metallic wall or panel of length = LENGTH. The prismatic panel has a complex “corrugated” cross section. The width of the complex “corrugated” panel is composed of a number of “major” cylindrical segments, each of which may be divided into a number of cylindrical “sub-segments”. Each segment and each sub-segment is a sector of a little cylindrical shell the generators of which run in the axial direction. In addition there exists an overall arching over the entire width of the panel. The several little cylindrical segments/sub-segments are joined generator to generator in series. The entire unstiffened “corrugated” panel can be fabricated without any welding. Half of the width of the panel is included in the BIGBOSOR4 model used for optimization, with symmetry and anti-symmetry conditions imposed at its mid-width, WIDTH/2. The GENOPT/BIGBOSOR4 system is used to build the model and to perform the optimization. The minimum weight of the panel is determined in the presence of the following behavioral constraints: 1. The panel shall not buckle locally, 2. The panel shall not buckle in a general mode that is symmetric at the plane of symmetry, 3. The panel shall not buckle in a general mode that is anti-symmetric at the plane of symmetry, 4. Each little cylindrical segment of the panel shall not buckle in a “classical” mode, and 5. The maximum stress in the wall of the panel shall be less than a specified value. The numerical analysis is extended to flat or curved panels of any width with repeating complex cross sections. An optimized cross-section profile with reflected and multiple repeating complex sections is mapped onto a cylindrical surface, and the buckling characteristics and weight of this complexly corrugated cylindrical shell are compared with those of an optimized “equivalent” cylindrical shell with external T-shaped stringers, an “equivalent” optimized cylindrical shell with a truss-core sandwich wall, and an optimized “equivalent” cylindrical shell with uniform corrugations. It is found that the optimized complexly corrugated cylindrical shell weighs less than these other three optimized “equivalent” cylindrical shells that have different wall constructions. Optimized complexly corrugated panels with cylindrical segments the geometry of which varies across its width weigh about 12-14 per cent less than optimized uniformly corrugated panels, such as a standard corrugated panel. The existence of sub-segments does not lead to panels that weigh less than panels without sub-segments. The gradient-based optimizer used in the GENOPT/BIGBOSOR4 system (“ADS” by Vanderplaats) has difficulty finding “global” optimum designs for panels with many cylindrical segments. Several of the optimized designs determined with GENOPT/BIGBOSOR4 are verified by comparison with predictions from the general-purpose computer program STAGS. The agreement between the predictions of GENOPT/BIGBOSOR4 and STAGS qualifies the use of GENOPT/BIGBOSOR4 for optimization in the particular cases studied here. The behavior of optimized configurations with “corners” (discontinuous slope in the “width-wise” direction) between major cylindrical segments are compared with that of configurations in which these “corners” have been smoothed by the introduction of transitional fairing segments. A program called HUGEBOBOR4 is developed to handle cases with up to 2950 segments.

Section 1. INTRODUCTION AND SUMMARY

Introduction:

A GOOGLE search on the string, “corrugated panel buckling” yields 2,240,000 items! A GOOGLE item of particular interest is United States Patent no. 4,358,916, November 16, 1982, entitled “Novel Corrugated Metal Building Structural Unit”, invented by Maurice Lacasse [1]. Figure 1 from [1] shows one of the panel geometries covered by the Lacasse patent. The prismatic Lacasse panel has major waves and minor waves within the major waves. This geometry is analogous to that suggested in this paper: panels with major cylindrical segments and cylindrical sub-segments superposed on each major cylindrical segment. In this paper, in addition to the major cylindrical segments and cylindrical sub-segments, there exists an overall arching over the entire width of the panel (the variable PHIBIG in Fig. 2 and in the bottom two frames of Fig. 3).

Many of the citations resulting from the GOOGLE search on “corrugated panel buckling” involve semi-sandwich and sandwich panels with corrugated cores of varying complexity. Other refinements include corrugated panels with laminated composite walls, sandwich panels with hierarchical corrugated cores and corrugated panels under combined normal pressure, axial compression and in-plane shear loading. The earliest work on buckling of corrugated panels seems to be that by Dean [2].

This paper includes a subset of the results that exist in [3]. Examples of the cross-sections of axially compressed panels investigated here are shown in Figs. 2 and 3. The overall dimensions, LENGTH x WIDTH, are indicated in Figs. 1 and 20.

A computer program system called "GENOPT/BIGBOSOR4" is used to obtain the optimum designs. GENOPT is a system of programs that can optimize anything [4] and BIGBOSOR4 [5], the successor to BOSOR4 [6], can analyze shells of revolution or prismatic shells [7]. The gradient-based optimizer used in GENOPT is called "ADS" [8,9]. The software, ADS, was created by Vanderplaats and his colleagues in the 1980s.

BIGBOSOR4 [5] is so named because it will handle complex shells of revolution with far more segments (295 segments) than will the original BOSOR4 program [6] from which it is derived. In addition, BIGBOSOR4 will handle prismatic shells [7], not just shells of revolution. Also, BIGBOSOR4 will handle shells of revolution that are connected to ground by multiple rings of springs (struts) [20]. Toward the end of this project a new version of BIGBOSOR4 called “HUGEBOSOR4” was created. HUGEBOSOR4 will handle up to 2950 segments.

In GENOPT the optimizer, ADS [8,9], is "hard-wired" in a so-called "0-5-7" mode: the "modified method of steepest descent". Constraint gradients are computed by finite differences of the behaviors of the "perturbed" design and the behaviors at the "current" design, in which the word, "behaviors", means modal vibration or stress or buckling, etc. A "perturbed" design is a design in which the value of one decision variable has been increased by a small amount (usually five per cent) from its "current" value. A constraint-gradient matrix is automatically generated by computation of the change in each behavior caused by the perturbation of each decision variable. The values of the "behavioral" constraints and the constraint-gradient matrix are inputs to the ADS optimizer by means of which a new "current" design is obtained by the modified method of steepest descent. The interaction of the BIGBOSOR4 (or HUGEBOSOR4) computer program with the ADS optimizer is entirely automated. The program user need specify only the number of design iterations for short computer runs

and does not even have to specify that for longer computer runs that involve automated multiple optimization processes starting from different points in design space.

For some of the cases described in this paper the gradient-based ADS software does not easily lead to a “global” optimum design. Design iterations sometimes repeatedly lead to local optima with less favorable objectives (WEIGHTs) that are not in the neighborhood of the “global” optimum. Haftka and his colleagues have developed the “Response Surface” method [10 - 15] that often avoids this deficiency. The “Response Surface” method has not been introduced into GENOPT.

In the previous paragraph "global" is in quotes because the strategy used in GENOPT cannot rigorously determine a true global optimum design. Instead, GENOPT attempts to find a design for which the objective is likely to be near that of a true global optimum by starting from many different points in design space during a single execution of a GENOPT processor called “SUPEROPT” (Figs. 4 and 5). Each new "starting" point is established randomly in a manner consistent with equality and inequality constraints. The GENOPT processor called “AUTOCHANGE” determines each new “starting” point in design space. In Figs. 4 and 5 each new "starting" point corresponds to a "spike" in the plot of objective versus design iterations processed during a single execution of SUPEROPT.

The GENOPT/BIGBOSOR4 system of computer programs has previously been used to obtain optimum designs of several structures that are comprised mainly of shells of revolution [5 and 16 - 21] or prismatic shells [7 and 22 - 24].

In the GENOPT system [4] there are two types of cases: **generic** and **specific**. In this work the generic case is called “**span9**”, and the specific cases have names such as “fold98updown” (Figs. 3, 6 – 12, 20 – 35), 40, “fold98supdwn” (Figs. 13 – 15), “fold98updwnu” (Figs. 16 – 19, 38, 39), “fold93” and “fold94” (Fig. A2), “fold95” and “fold96” (Fig. A3), “fold97” and “fold98” (Figs. A4, A7 and A8), “fold916” (Figs. A9 - A11), “fold94S” (Fig. A6), “wide98” (Fig. A12) and “wide98updown” (Fig. A13). Also, there are specific cases with names such as “narw96updown” (Figs. 46 – 53) and “narw91updown” (Figs. 46, 54 and 55). The specific cases are all members of the generic class, “**span9**”. There are two types of user: the **GENOPT user** and the **END user**. The GENOPT user creates the software by means of which specific cases in the generic class can be solved. The END user sets up and runs specific cases such as “fold98updown”, “fold98supdwn”, “fold98updwnu”, “fold94”, “fold94S”, “fold95”, “wide98”, “narw96updown”, etc. In the work reported here the GENOPT user and the END user are the same person: the first author of this paper.

Summary:

Sections 2 and 3 give general information about GENOPT. **Section 4** contains a statement of the problem, a list and description of the behavioral constraints (e.g. buckling, stress, etc.), identification of variables that can be decision variables during optimization, boundary conditions, loading (**uniform axial compression**), and an example. **Section 5** provides information about the generic case called “span9”. **Section 6** describes how, in general, optimum designs are arrived at. **Section 7** gives numerical results from GENOPT/BIGBOSOR4 for a panel with LENGTH x WIDTH = 100 x 100 inches. **Section 8** describes the effect of changing boundary conditions along the left-hand longitudinal edge from “OLD” to “NEW” with subsequent (post-optimization) extension to panels of any width. **Section 9** introduces sub-segments. **Section 10** treats panels with uniform corrugations. **Section 11** compares predictions for a previously optimized panel from GENOPT/BIGBOSOR4

with those from STAGS. **Section 12** explores design sensitivity of a previously optimized panel. **Section 13** identifies certain difficulties in finding “global” optimum designs. **Section 14** describes how panels with “corners” (discontinuous slopes in the width-wise direction) are smoothed away. **Section 15** describes how to optimize panels that can be fabricated by stamping. **Section 16** describes how an optimized “flat” panel (flat in an average sense) can be mapped onto a “cylindrical” surface (cylindrical in an average sense) and compares the optimized weights of complexly corrugated cylindrical panels with the optimized weights of cylindrical panels with other types of wall construction. **Section 17** establishes the weights of uniformly corrugated panels and complexly corrugated panels over a range of number, NSEG, of major segments spanning WIDTH/2 = 50 inches. **Section 18** extends the analysis described in Section 17 for 100 x 100-inch panels by optimizing cross-section profiles over a range of smaller widths (WIDTH << 100 inches) by first optimizing for various values of WIDTH/2 and then extending the previously optimized cross-section profiles to the total 100-inch width of the panel by reflection and repetition, followed by mapping a much wider “flat” cross-section profile onto a cylindrical surface.

Section 2. INFORMATION ABOUT GENOPT

Information about GENOPT is provided in the papers cited in the Introduction. Also, there are several files provided on the “**shellbuckling.com**” website cited in [3]. The files listed next are contained in the compressed “tar” file, general.info.tar.gz, which is contained in the bigger compressed “tar” file, .../genopt/case/span/**span9.tar.gz**, that is part of the very big compressed “tar” file that can be downloaded from the “Downloads” page of the “**shellbuckling.com**” website [3]. The individual files contained in the “general.info.tar.gz” file are as follows:

FILES GIVING GENERAL INFORMATION ABOUT GENOPT

```
-----
-rw-r--r--  1 dave  staff    5720 Nov 24  2011 genopt.abstract
-rw-r--r--  1 dave  staff    2024 Nov 24  2011 genopt.commands
-rw-r--r--  1 dave  staff    3234 Nov 24  2011 genopt.files
-rw-r--r--  1 dave  staff   93262 Nov 24  2011 genopt.getting.started
-rw-r--r--  1 dave  staff    8909 Nov 24  2011 genopt.information
-rw-r--r--  1 dave  staff     987 Nov 24  2011 genopt.information.files
-rw-r--r--  1 dave  staff    2507 Nov 24  2011 genopt.programs
-rw-r--r--  1 dave  staff   251587 Nov 24  2011 genopt.runstream
-rw-r--r--  1 dave  staff    11183 Nov 24  2011 genopt.summary
-rw-r--r--  1 dave  staff    1077 Nov 24  2011 genopt.variable.roles
-----
```

In addition there exists a file in [3], ...genopt/doc/genopt.news, that contains a log of updates to the GENOPT system.

Section 3. AN EXAMPLE OF A GENOPT/BIGBOSOR4 RUN STREAM

A rather detailed example of a GENOPT/BIGBOSOR4 run stream executed during past work (for a generic case called “tank” [3, 20]) is contained in the file called “**genopt.runstream**” that is one of the files contained in the compressed “tar” file, general.info.tar.gz, as just listed. The run stream involves execution of the various GENOPT processors, which are listed near the beginning of the file, genopt.runstream, and repeated here for

convenience (with the generic name, “tank”, changed here to the generic name, “span9”, that pertains to this paper rather than to the “tank” configuration described in [20]):

gentext (to be executed in a directory called “genoptcase”) The GENOPT user and the GENOPT process, gentext, generate a prompt file, span9.PRO (included as part of the file, .../genopt/case/span/span9.tar.gz, of [3]) and an information file, span9.DEF, that contains a glossary: Table 1 of this paper. (Input data for “gentext” are stored in a file called “span9.INP”.) Skeletal libraries, behavior.new and struct.new, are automatically created by gentext. These skeletal libraries are to be "fleshed out" by the GENOPT user. (See the files, behavior.span9 and struct.span9 that are contained in the compressed tar file, .../genopt/case/span/span9.tar.gz [3].) Before the next GENOPT command, “**genprograms**”, can be given by the GENOPT user, the GENOPT user must, for the generic case called “**span9**” that is the subject of this particular paper, first type the following:

cd .../<working directory>
cp .../genopt/case/span/span9.tar.gz <working directory>
gunzip span9.tar.gz
tar xvf span9.tar
cp span9.INP .../genoptcase/.
cp behavior.span9 .../genoptcase/behavior.new
cp struct.span9 .../genoptcase/struct.new
cp bosdec.span9.hugecyl .../bosdec/sources/bosdec.src
cp .../bigbosor4/sources/addbosor4.src .../bosdec/sources/addbosor4.src
cd .../genoptcase

NOTE: The FORTRAN source file called “bosdec.src” generates a valid input file for BIGBOSOR4. “bosdec.src” must be created by the GENOPT user if it does not already exist for the generic class.

genprograms The GENOPT user (via the GENOPT system) generates (makes) executables: begin, decide, mainsetup, optimize, change, chooseplot, and diplot that pertain to the generic case called “**span9**”.

begin The End user provides starting data for a specific case: For example, one input file for “begin” is called “fold98updown.BEG” (Table 2 in this paper), in which the specific case name is “fold98updown”.

decide The End user chooses decision variables, bounds, linked variables, and inequality constraints: For example, the input file for “decide” is called “fold98updown.DEC” (included in the file, .../genopt/case/span/span9.tar.gz, of [3]).

mainsetup The End user sets up strategy parameters: For example, the input file for “mainsetup” is called “fold98updown.OPT” (included in the file, .../genopt/case/span/span9.tar.gz, of [3]).

optimize The End user performs optimization (Figs. 25a and 25b), or analyzes a fixed design (Figs. 7 - 9 and (for example) the output file, fold98updown.OPM, included in the file, .../genopt/case/span/span9.tar.gz, of [3]), or performs design sensitivity with respect to selected decision variables (Figs. 23a and 23b).

change The End user changes/archives variables: For example, the input file for “change” is called “fold98updown.CHG” (included in the file, .../genopt/case/span/span9.tar.gz, of [3]).

autochange New values for decision variables are randomly computed by GENOPT: (See the "spikes" in the plots in Figs. 4 and 5). Each "spike" represents a new "starting" design created by "autochange".

superopt The End user finds a "global" optimum design via several automated sequential sets of operations: (**n** executions of optimize followed by one execution of autochange, in which **n** is provided by the End user) with the same sequence (...) repeated many times: (Figs. 4 and 5 and A15 – A18).

superduperopt GENOPT automatically executes **superopt** "k" times, in which "k" is provided by the End user. (Figs. 4 and 5 result, respectively, from the first two consecutive executions of **superopt** embedded in a single execution of **superduperopt**.)

chooseplot The End user chooses which variables/margins to plot vs. design iterations: For example, the input file for "chooseplot" is called "fold98updown.CPL" (included in the file, .../genopt/case/span/span9.tar.gz, of [3]).

diplot The End user plots variables vs. design iterations: (Figs. 4 and 5). For example, the "postscript" file for a plot of the objective vs. design iterations for the specific case, "fold98updown", is called "fold98updown.5.ps". The plot appears on the End user's screen via the command: gv fold98updown.5.ps, in which "gv" means "ghost view".

insert The GENOPT user adds parameters to the generic problem.

cleanspec The End user cleans up SPECific case files (e.g. "fold98updown" files).

cleangen The GENOPT user cleans up GENeric case files (e.g. "span9" files).

Section 4. SOME GEOMETRICAL AND OTHER DETAILS

Statement of the problem

We want to determine the minimum amount of metallic (aluminum) material required to span a distance of WIDTH (Tables 1 and 2, Figs. 2, 3, 10, 15, 20) for a prismatic, axially compressed structure (a wall or panel) with a complex "corrugated" cross section such as those displayed in Figs. 2, 3, 6, top frame in Fig. 11, top frame in Fig. 15, Fig. 16, Fig. 18 and Fig. 20. The axial length of the prismatic panel is LENGTH (Tables 1 and 2, Fig. 1 and Fig. 20). The entire width, WIDTH, of the prismatic "corrugated" panel is composed of a number, $2 \times \text{NSEG}$, of so-called "major" segments (Figs. 3 and 6), each of which may be divided into a number, NSUBSEG , of sub-segments (Fig. 13, top and Fig. 15, top and Fig. A6, top). Each segment and each sub-segment is a sector of a little cylindrical shell the generators of which run in the axial (z) direction. Half of the width, $\text{WIDTH}/2$, of the corrugated panel is included in the model used for optimization, with symmetry conditions applied at $x = \text{WIDTH}/2$ in the prebuckling phase of the problem (Fig. 3, bottom and Fig. 6) and both symmetry and anti-symmetry conditions applied in the bifurcation general buckling phase of the problem (Fig. 7, for example). In addition to the boundary condition along the left-hand longitudinal edge, "u,v,w fixed;

rotation free”, as indicated in Fig. 6 (called “**OLD**” boundary condition in Section 4), there are BIGBOSOR4 models in which the boundary condition along the left-hand longitudinal edge is “symmetry/antisymmetry”, as indicated in the top frame of Fig. 9 (called “**NEW**” boundary condition in Section 4). The “**NEW**” boundary condition is more practical than the “**OLD**” because it permits extension of the BIGBOSOR4 model to models with multiple repeating complex cross-section profiles, such as that shown at the top of Fig. 11 and in Fig. 33.

The material is isotropic, with material properties listed near the beginning of Table 2. (The material properties used in this work represent aluminum.) There is only one layer in the wall of the corrugated panel. The thickness is constant within each major segment of the panel. The panel exists in an x,y,z coordinate system in which: x = the width-wise coordinate (in the WIDTH direction, the horizontal coordinate in Fig. 6, for example), y = the vertical coordinate (normal to the x,z plane), and z = the length-wise coordinate (in the LENGTH direction: distance along the axis of the prismatic panel). The uniform axial compression (uniform end shortening) is applied in the z -direction, that is, normal to the plane of the screen in Figs. 2, 3 and 6, for example.

Behavioral Constraints

The corrugated panel is optimized subject to the following "behavioral" constraints:

1. Local buckling that is symmetric about the symmetry plane at $x = \text{WIDTH}/2$. Local buckling is computed in SUBROUTINE BEHX1, which is one of the subroutines in the file, behavior.span9. The FORTRAN file, behavior.span9, is included in the long compressed “tar” file, .../genopt/case/span/**span9.tar.gz**, of [3]. See the local buckling modes shown in Fig. 8, for example. The local buckling modes such as those shown in Fig. 8 are computed by BIGBOSOR4. Figure 8 shows a local buckling mode with the “**OLD**” boundary condition (u,v,w fixed; rotation free) imposed along the left-hand edge. The bottom frame in Fig. 9 shows a local buckling mode with the “**NEW**” boundary condition (“symmetry/antisymmetry”) imposed along the left-hand edge.
2. General buckling that is symmetric about the symmetry plane at $x = \text{WIDTH}/2$. General "symmetric" buckling is computed in SUBROUTINE BEHX2, which is one of the subroutines in the file, behavior.span9. See the buckling modes shown in the bottom two frames of Fig. 7, for example. The symmetric general buckling modes such as those shown in Fig. 7 are computed by BIGBOSOR4. Figure 7 shows general buckling modes with the “**OLD**” boundary condition (u,v,w fixed; rotation free) imposed along the left-hand edge. The middle frame in Fig. 9 shows a general buckling mode with the “**NEW**” boundary condition (“symmetry/antisymmetry”) imposed along the left-hand edge.
3. General buckling that is anti-symmetric about the symmetry plane at $x = \text{WIDTH}/2$. General "anti-symmetric" buckling is computed in SUBROUTINE BEHX3, which is one of the subroutines in the file, behavior.span9 [3]. See the buckling modes shown in the top two frames of Fig. 7, for example. The anti-symmetric general buckling modes such as those shown in Fig. 7 are computed by BIGBOSOR4. During the course of the effort required to produce this paper, it was discovered that, with use of the “**NEW**” boundary conditions, it is necessary to compute **three** so-called “anti-symmetric” buckling load factors, one for each of three models: **a.** anti-symmetry along both left-hand and right-hand longitudinal edges, **b.** anti-symmetry along the left-hand longitudinal edge and symmetry along the right-hand longitudinal edge, and **c.** symmetry along the left-hand longitudinal edge and anti-symmetry along the right-hand longitudinal edge. See Fig. 50 for an example. These three models are all processed in SUBROUTINE BEHX3. The model that yields the lowest

buckling load factor of the three generates the critical “anti-symmetric” buckling load factor (BUCASY in Table 1). This determines the buckling margin containing the string, “BUCASY”.

4. "Classical" buckling of each cylindrical segment of the corrugated panel. If there exist sub-segments this behavioral constraint applies to one of the sub-segments within each major segment. (All of the sub-segments within a given major segment have the same radius/thickness; therefore the “classical” buckling load factor is the same for every sub-segment in that major segment.) "Classical" buckling is computed in SUBROUTINE BEHX4, which is one of the subroutines in the file, behavior.span9 [3]. **The "Classical" buckling formula includes a suitable knockdown factor** computed from interpolation of a table of values that depend on the radius/thickness of each cylindrical segment as given by the curve labeled “A design recommendation” in Fig. 5.18 on page 186 of the 1975 McGraw-Hill book by Brush and Almroth [25]. (For the convenience of the reader a modified version of that figure appears in this paper as Fig. A14.)

5. Maximum allowable stress. The stress is that for the complex corrugated panel in a membrane prebuckled state, that is, uniform axial compression (uniform end shortening) in the z coordinate direction. The maximum stress in the cross-section of the prismatic panel is computed in SUBROUTINE BEHX5, which is one of the subroutines in the file, behavior.span9 [3]. There is no development of Poisson-ratio-induced hoop compression; it is assumed that the axially compressed panel is free to expand in the hoop direction.

Decision Variable Candidates

The problem variables, established by the GENOPT user, are listed in Tables 1 – 3. The decision variable candidates (variables with Role Type No. 1 listed in Table 1 and indicated in Figs. 2 and 3) are:

1. The thickness, THICK, assumed to be constant over each major segment, but can be different in each major segment (THICK(i), $i = 1, 2, \dots$ NSEG). See the middle frame of Fig. 3. There are NSEG major segments over the half-width, WIDTH/2, of the panel. All of the results presented for specific cases in this paper are for panels in which the thickness of the entire WIDTH of the panel is uniform, that is, the decision variables, THICK(j), $j = 2, 3, \dots$, NSEG, are linked to THICK(1) with the linking constant equal to 1.0.
2. The projected width, SUBWID, of each major segment (SUBWID(i), $i = 1, 2, \dots$ NSEG). This is the width of the major panel segment projected onto the x, z plane, that is, the "x-width" of the major segment. (See Fig. 2 and the bottom frame of Fig. 3.)
3. The half-angles, PHISEG(i), $i = 1, 2, \dots$ NSEG, (degrees) of the major cylindrical corrugation segments. (See Fig. 2 and the bottom frame of Fig. 3.) "Half-angle" (in degrees) means half the angle subtended by a single cylindrical segment of the complex corrugated panel. The major segments are joined width-wise longitudinal edge-to-edge (generator-to-generator as shown in Figs. 2 and 3).
4. The half-angles, PHISUB(i), $i = 1, 2, \dots$ NSEG, (degrees), of the cylindrical sub-segments, if any. (See the top frame of Fig. 13, for example.) The longitudinal edges of each cylindrical sub-segment have the same y-coordinates of the major cylindrical segment that would exist if there were no sub-segments. PHISUB(i) is half the angle spanned by a sub-corrugation in the ith major segment, as indicated for Major Segment No. 3, for example, in the top frame of Fig. A6. All the sub-corrugations within a given major segment have the same

value of PHISUB. PHISUB(i), $i=1$, NSEG can be different in different major segments. The thicknesses of all sub-corrugations in the i th major segment are the same and equal to THICK(i).

5. The distances, YPLATE, that the junctions of the major segments and the two longitudinal edges of the model, the leftmost edge at $x = 0$ and the rightmost edge at $x = \text{WIDTH}/2$, are raised above the x,z plane **as if the overall arching angle, PHIBIG (defined next), were zero**. If PHIBIG is greater than zero YPLATE are the vertical distances above the overall arch at the x -locations of the junctions of the major segments and at the left-most and right-most edges of the panel, as shown in Fig. 2. YPLATE is a vector with NSEG+1 components, in which NSEG is the number of major segments over the panel half-width, $\text{WIDTH}/2$. (See Fig. 2 and the top frame of Fig. 3, for example.)

6. An overall arching half-angle, PHIBIG (degrees), that spans the distance, $\text{WIDTH}/2$, of the corrugated panel. The top of the arch (or bottom of the "valley") is located at $x = \text{WIDTH}/2$. (See Fig. 2.) Figure 3 shows starting designs of corrugated panels with three values of PHIBIG: Top frame: PHIBIG = 0.1 degree (essentially zero), Middle frame: PHIBIG = 10 degrees and Bottom frame: PHIBIG = 60 degrees.

The entire corrugated panel is raised above the x,z plane by a y -distance equal to YPLATE(1). The purpose of YPLATE(1) is to avoid zero, near-zero, and negative values of the decision variable candidates, YPLATE(2), YPLATE(3), ... , YPLATE(NSEG+1). YPLATE(1) should NEVER be a decision variable.

Boundary Conditions

There are two different boundary conditions used during the course of this work. The first option, described in the next paragraph and called "**OLD**", was used during the initial phase of the work and is now considered to be superseded by the second, more practical, option, "**NEW**", described two paragraphs hence.

OLD: The edges all around the perimeter of the complete corrugated panel of width = WIDTH (Tables 1 and 2) are simply supported. NOTE: At the left-hand longitudinal edge the three displacement components, USTAR, VSTAR, WSTAR (called u,v,w on the left-hand side of Fig. 6), are constrained and the rotation about the generator is free. USTAR, VSTAR, WSTAR are the vertical (y -direction), axial (z -direction), and width-wise (x -direction) displacement components, respectively. The two curved edges at $z = 0$ and $z = \text{LENGTH}$ are simply supported in the "classical" sense, that is, buckling is anti-symmetric about those two edges. Another way of expressing this is: buckling nodal lines occur along the two edges at $z = 0$ and $z = \text{LENGTH}$.

NEW: In order to extend the applicability of the model to panels of any width with repeating complex cross sections, such as that shown in the top frame of Fig. 11 and that shown in the top frame of Fig. 33, a version of the "span9" "bosdec" software was generated with symmetry/anti-symmetry conditions applied along both the left-hand and right-hand generators (the two longitudinal edges of the panel), such as is indicated in the top frame of Fig. 9. This modified version of the "span9" software was originally called "bosdec.span9.leftedge". The file, "bosdec.span9.leftedge", has been superseded by the two files, "bosdec.span9.hugecyl" and "bosdec.span9.smoothing". These two newer files are included in the larger compressed "tar" file, ...genopt/case/span/**span9.tar.gz** [3]. ("bosdec" is a FORTRAN subroutine that generates valid input files for BIGBOSOR4. Both files, "bosdec.span9.hugecyl" and "bosdec.span9.smoothing", also generate valid input files for STAGS.) **NOTE:** In the case of anti-symmetric general buckling (called "BUCASY" in Table 1), with use of the NEW boundary conditions (symmetry or anti-symmetry along both the left-hand and right-hand

longitudinal edges as indicated in the top frame of Fig. 9), the anti-symmetric buckling load factor, BUKASY, is computed **as the minimum of three general buckling load factors**: **1.** that computed with anti-symmetry conditions applied along the left-hand edge and anti-symmetry along the right-hand edge, **2.** that computed with anti-symmetry conditions applied along the left-hand edge and symmetry along the right-hand edge, and **3.** that computed with symmetry conditions applied along the left-hand edge and anti-symmetry along the right-hand edge. Figure 50 shows an example of these three “anti-symmetric” buckling models and buckling modes and buckling load factors.

An Example (“OLD” boundary conditions)

Typical values of design variables for an **optimized design** of a corrugated panel originally with 5 uniform “convex surface up” major segments over WIDTH/2 and no sub-segments (the specific case called “fold95” shown in Appendix 3 in the top two frames of Fig. A3) are listed in the following table (output abstracted from the entire *.OPM file [3], in which “*” represents the name of the specific case, “fold95”):

OPTIMIZED VALUES OF DESIGN VARIABLES FOR THE BEST FEASIBLE DESIGN

VAR. CURRENT

NO.	VALUE	DEFINITION
1	6.382E-02	wall thickness of the major segment: THICK(1)
2	6.382E-02	wall thickness of the major segment: THICK(2)
3	6.382E-02	wall thickness of the major segment: THICK(3)
4	6.382E-02	wall thickness of the major segment: THICK(4)
5	6.382E-02	wall thickness of the major segment: THICK(5)
6	4.408E+00	projected width(x-width) of sub-plate: SUBWID(1)
7	5.648E+00	projected width(x-width) of sub-plate: SUBWID(2)
8	1.276E+01	projected width(x-width) of sub-plate: SUBWID(3)
9	1.319E+01	projected width(x-width) of sub-plate: SUBWID(4)
10	1.354E+01	projected width(x-width) of sub-plate: SUBWID(5)
11	3.409E+01	half-angle(deg.) of major corrugation: PHISEG(1)
12	3.511E+01	half-angle(deg.) of major corrugation: PHISEG(2)
13	7.877E+01	half-angle(deg.) of major corrugation: PHISEG(3)
14	8.447E+01	half-angle(deg.) of major corrugation: PHISEG(4)
15	8.397E+01	half-angle(deg.) of major corrugation: PHISEG(5)
16	7.000E+01	half-angle (deg.) of sub-corrugation: PHISUB(1)
17	7.000E+01	half-angle (deg.) of sub-corrugation: PHISUB(2)
18	7.000E+01	half-angle (deg.) of sub-corrugation: PHISUB(3)
19	7.000E+01	half-angle (deg.) of sub-corrugation: PHISUB(4)
20	7.000E+01	half-angle (deg.) of sub-corrugation: PHISUB(5)
21	3.000E+01	vertical y above (x,y,z) origin if PHIBIG=0: YPLATE(1)
22	2.301E+01	vertical y above (x,y,z) origin if PHIBIG=0: YPLATE(2)
23	2.096E+01	vertical y above (x,y,z) origin if PHIBIG=0: YPLATE(3)
24	1.986E+01	vertical y above (x,y,z) origin if PHIBIG=0: YPLATE(4)
25	2.022E+01	vertical y above (x,y,z) origin if PHIBIG=0: YPLATE(5)

26 2.056E+01 vertical y above (x,y,z) origin if PHIBIG=0:YPLATE(6)

27 8.866E+00 half-angle (deg.) of overall arching: PHIBIG

Note that sub-segment half-angles, PHISUB(i), i=1,5, are listed above (variable numbers 16-20) even though in this particular case there are no sub-segments. The wall thicknesses of all 5 major segments are the same in this particular example because THICK(2), THICK(3), THICK(4) and THICK(5) are all linked to THICK(1) via the GENOPT processor, DECIDE. (This linking of THICK(2) through THICK(NSEG) holds for all of the specific cases described in this paper. Hence, the **entire paper pertains to corrugated panels of uniform thickness.**)

Typical **behavioral constraints** (from which the behavioral design margins are computed) for the optimized “fold95” design just listed are as follows:

***** RESULTS FOR LOAD SET NO. 1 *****

PARAMETERS THAT DESCRIBE BEHAVIOR(e.g. stress,buckling load)

BEH. CURRENT

NO.	VALUE	DEFINITION
1	2.011E+00	local buckling load factor: LOCBUK(1)
2	1.503E+00	symmetric general buckling: BUKSYM(1)
3	1.512E+00	antisymmetric general buckling: BUKASY(1)
4	1.009E+00	classical buckling load factor: CYLBUK(1,1)
5	1.554E+00	classical buckling load factor: CYLBUK(1,2)
6	1.084E+00	classical buckling load factor: CYLBUK(1,3)
7	1.054E+00	classical buckling load factor: CYLBUK(1,4)
8	1.022E+00	classical buckling load factor: CYLBUK(1,5)
9	2.181E+04	maximum effective stress: STRESS(1)

The first array index in the behavioral variables, LOCBUK, BUKSYM, BUKASY, CYLBUK and STRESS, is the load case number. **In this work there exists only a single load case: axial loading** (loading in the LENGTH direction in Figs. 1 and 20). The second array index in the behavioral variable, CYLBUK, is the major segment number. The “classical” buckling load factors, CYLBUK(1, j), j = 1,NSEG, each include a knockdown factor that depends on the radius/thickness ratio of each cylindrical segment in the corrugated panel. This knockdown factor is interpolated from a table of knockdown factors versus radius/thickness ratio as shown for the curve labeled “A design recommendation” in Fig. 5.18 on page 186 of the 1975 McGraw-Hill book by Brush and Almroth [25]. (See Fig. A14 in this paper for a modified version of that Brush and Almroth figure.) The table of knockdown factors versus radius/thickness is listed near the beginning of SUBROUTINE BEHX4 in the file, behavior.span9, which is included in the compressed “tar” file, ...genopt/case/span/**span9.tar.gz** [3].

Typical **design margins** for the “fold95” optimized design are:

***** RESULTS FOR LOAD SET NO. 1 *****

MARGINS CORRESPONDING TO CURRENT DESIGN (F.S.= FACTOR OF SAFETY)

MARGIN CURRENT

NO.	VALUE	DEFINITION
1	9.540E-02	$6.05 - 0.10 * V(6) - 0.10 * V(7) - 0.10 * V(8) - 0.10 * V(9) - 0.10 * V(10) - 1$
2	4.600E-03	$-3.95 + 0.10 * V(6) + 0.10 * V(7) + 0.10 * V(8) + 0.10 * V(9) + 0.10 * V(10) - 1$
3	5.380E-03	$(LOCBUK(1) / LOCBUKA(1)) / LOCBUKF(1) - 1; F.S. = 2.00$

```

4 2.171E-03 (BUKSYM(1)/BUKSYMA(1))/BUKSYMF(1)-1;F.S.=1.50
5 8.285E-03 (BUKASY(1)/BUKASYA(1))/BUKASYF(1)-1;F.S.=1.50
6 8.958E-03 (CYLBUK(1,1)/CYLBUKA(1,1))/CYLBUKF(1,1)-1;F.S.=1.0
7 5.542E-01 (CYLBUK(1,2)/CYLBUKA(1,2))/CYLBUKF(1,2)-1;F.S.=1.0
8 8.370E-02 (CYLBUK(1,3)/CYLBUKA(1,3))/CYLBUKF(1,3)-1;F.S.=1.0
9 5.350E-02 (CYLBUK(1,4)/CYLBUKA(1,4))/CYLBUKF(1,4)-1;F.S.=1.0
10 2.183E-02 (CYLBUK(1,5)/CYLBUKA(1,5))/CYLBUKF(1,5)-1;F.S.=1.0
11 2.057E+00 (STRESSA(1)/STRESS(1))/STRESSF(1)-1; F.S.= 1.50

```

The critical margins (margins near zero) are listed in **bold face**. Margins 2 – 11 correspond to Behaviors 1 – 9 in the list with the heading, “**PARAMETERS THAT DESCRIBE BEHAVIOR**” In the lists of “behavior” and “margins” the following conventions apply:

Behavioral variable names (edited from Table 1):

LOCBUK means local buckling load factor.

BUKSYM means symmetric general buckling (buckling mode is symmetric about the symmetry plane).

BUKASY means anti-symmetric general buckling (buckling mode is anti-symmetric about the symmetry plane).

CYLBUK means knocked down “classical” buckling load factor: buckling of each cylindrical segment from classical theory including the appropriate knockdown factor from the curve in Fig. A14 labeled “A design recommendation”; the knocked-down **buckling stress** = $K \times 0.605 \times E \times t/r$. (K = knockdown factor computed from interpolation along the curve labeled “A design recommendation” in Fig. A14.)

STRESS means maximum effective stress.

"A", "F", and "F.S.":

An "A" added to a behavioral variable name means "allowable"

An "F" added to a behavioral variable name means "factor of safety"

"F.S." means "factor of safety". The factors of safety for CYLBUK(i,j) = 1.0 because a conservative knockdown factor **K** is already included in the “Classical” buckling formula.

Indices, i,j:

LOCBUK(i), BUKSYM(i), BUKASY(i) = linear buckling of various types: i = load case number

CYLBUK(i,j) = “classical” buckling of cylindrical shell: i = load case; j = cylindrical major segment number

STRESS(i) = maximum effective stress: i = load case number

The format of the definitions of the margins listed above is generated automatically by the GENTEXT processor of GENOPT.

Two Inequality Constraints

The first two “fold95” margins listed above (associated with the heading, “**MARGINS CORRESPONDING TO CURRENT DESIGN**”):

```
1 9.540E-02 6.05-0.10*V(6)-0.10*V(7)-0.10*V(8)-0.10*V(9)-0.10*V(10)-1
2 4.600E-03 -3.95+0.10*V(6)+0.10*V(7)+0.10*V(8)+0.10*V(9)+0.10*V(10)-1
```

require an explanation. These two margins arise from two inequality constraints generated by the End user in the GENOPT processor called “DECIDE”. The variables, V(6), V(7), V(8), V(9), V(10), are respectively the projected widths, SUBWID(1), SUBWID(2), SUBWID(3), SUBWID(4) and SUBWID(5), of the 5 major segments listed above under the heading, “**VALUES OF DESIGN VARIABLES FOR THE BEST FEASIBLE DESIGN**”. These 5 variables are decision variables, and the sum of them should always be very close to the half-width, WIDTH/2, of the corrugated panel. In order to accomplish this, two inequality constraints are introduced during execution of the GENOPT processor called "DECIDE":

1. The overall half-width, WIDTH/2=50 inches, of the corrugated panel should be less than 50.5 inches.
2. The overall half-width, WIDTH/2=50 inches, of the corrugated panel should be greater than 49.5 inches.

In other words,

$$50.5 > V(6) + V(7) + V(8) + V(9) + V(10)$$

and

$$49.5 < V(6) + V(7) + V(8) + V(9) + V(10)$$

During the development of this application it was found that the numerical behavior was improved by multiplying both sides of the two inequality constraints by 0.1. Therefore, the two new (equivalent) inequality constraints are:

$$5.05 > 0.1*V(6) + 0.1*V(7) + 0.1*V(8) + 0.1*V(9) + 0.1*V(10)$$

$$4.95 < 0.1*V(6) + 0.1*V(7) + 0.1*V(8) + 0.1*V(9) + 0.1*V(10)$$

Item No. 603 of the file, ../genopt/execute/URPROMPT.DAT (a permanent GENOPT file, independent of the generic case, that contains prompting phrases and "help" paragraphs for the End user) includes the following lines:

603.1 Identify the type of inequality expression (1 or 2)

603.2

The inequality relations have either of the two forms:

Type 1 is: $1.0 > f(v1,v2,v3,...)$

Type 2 is: $1.0 < f(v1,v2,v3,...)$

in which the expression $f(v1,v2,v3,...)$ has the form:

$f(v1,v2,v3,...) = C0 + C1*v1**D1 + C2*v2**D2 + C3*v3**D3 +$
+etc (up to max. of 45 terms).

In this application we always use inequality constraints of Type 2: $1.0 < f(v1,v2,v3,...)$. Therefore, we multiply both sides of the first inequality constraint by -1.0, obtaining

$$-5.05 < -0.1*V(6) -0.1*V(7) -0.1*V(8) -0.1*V(9) -0.1*V(10)$$

Adding 5.05 to both sides of this first inequality gives:

$$0.0 < 5.05 -0.1*V(6) -0.1*V(7) -0.1*V(8) -0.1*V(9) -0.1*V(10)$$

and then adding 1.0 to both sides of this inequality gives:

$$1.0 < 6.05 -0.1*V(6) -0.1*V(7) -0.1*V(8) -0.1*V(9) -0.1*V(10)$$

which is in the proper “Type 2” form identified above, that is: $1.0 < f(v1,v2,v3,...)$.

In an analogous manner, subtracting 4.95 from both sides of the second inequality listed above and then adding 1.0 to both sides of this second inequality gives:

$$1.0 < -3.95 +0.1*V(6) +0.1*V(7) +0.1*V(8) +0.1*V(9) +0.1*V(10)$$

These two inequalities, valid for the particular specific case called “fold95”, are:

$$\begin{aligned} 1.0 &< 6.05 -0.1*V(6) -0.1*V(7) -0.1*V(8) -0.1*V(9) -0.1*V(10) \\ 1.0 &< -3.95 +0.1*V(6) +0.1*V(7) +0.1*V(8) +0.1*V(9) +0.1*V(10) \end{aligned}$$

They correspond to the first two margins listed above in the section entitled “**MARGINS CORRESPONDING TO CURRENT DESIGN (F.S.= FACTOR OF SAFETY)**”.

The application of the two inequality relationships usually leads to a small gap in the optimized panel, such as that indicated and explained in Fig. 6 and shown in Fig. 10, Fig. 15, and Fig. 20b. The second inequality leads to the more critical of the two corresponding design margins because an optimized panel of $WIDTH/2 = 49.5$ inches weighs less than an equivalent optimized panel of $WIDTH/2 = 50.5$ inches.

Loading

In this problem there is **only one load case: overall axial compression**. The axial stress resultant in each of the NSEG segments of the corrugated panel is computed as if the entire corrugated panel is subjected to uniform end shortening. In other words, the axial membrane resultant, N_z , at any point across the width of the corrugated panel is proportional to the wall thickness of each major segment. What is asked for in the GENOPT processor called “BEGIN” is the TOTAL axial load applied to the corrugated panel over the entire panel width, WIDTH. For example, in Table 2 (the input file for “BEGIN” for the specific case called “fold98updown”) the relevant record is as follows:

```
-200000.0      $ total axial load (e.g. lb): TOTLOD( 1)
```

The axial load is negative for compression, units = lb, for example. It is assumed in this “span9” model that the applied axial compression does not induce any width-wise loading. In other words, the corrugated panel is free to expand in the width direction when the axial compression (uniform end shortening) is applied.

Section 5. INFORMATION ABOUT THE GENERIC CASE CALLED "span9"

In the GENOPT universe there are two types of cases:

(1) A **generic** case (called "span9" in this paper)

(2) **Specific** cases that fit within the generic set. These specific cases are here called “fold98updown”, “fold98supdwn”, “fold98updwnu”, “fold91”, “fold92”, “fold93”, “fold94”, “fold94S”, “fold95”, “fold96”, “fold97”, “fold98”, “fold916”, “wide98” and “wide98updown”. Also, there are specific cases with names such as “narw96updown” and “narw91updown”. The numeral “9” in the name denotes that these specific cases are members of the generic set, **span9**. The second numeral in the names denotes the number of major cylindrical segments over half the panel width: WIDTH/2. The letter “s” or “S” in the names, “fold98supdwn” and “fold94S”, denotes the presence of sub-segments in the model. The ending letter “u” in the name, “fold98updwnu” denotes “uniform major segments”. The string, “updown” or “updwn” in the names, fold98updown, fold98supdwn, fold98updwnu and wide98updown, denotes major segments with alternating “convex surface up” and “convex surface down” configurations such as that shown in Figs. 2, 6 and 16, for example. The string, “narw” in the names, narw96updown and narw91updown, indicates “narrow panels”.

Corresponding to each of the two classes of case, **generic** and **specific**, there are possibly different users. The role of the **GENOPT user** is to create the software for setting up the GENERIC environment ("span9"). The **End user** exercises the GENERIC environment, "span9", for SPECIFIC cases, such as the cases called "fold94", "fold94S", "fold95", etc. In the work reported here the GENOPT user and End user are the same person, the first author.

The following files pertain to the GENERIC case called "span9". The files whose names are listed next are contained in the compressed "tar" file, **span9.tar.gz**, which is contained in the directory, .../genopt/case/span, which is part of the very big compressed “tar” file that can be downloaded from the “Downloads” page of the “**shellbuckling.com**” website [3]. Some of the files contained in the “**span9.tar.gz**” file are the following:

FILES RELATED TO THE GENOPT USER'S GENERIC CASE = “span9” PHASE OF THIS PROJECT

Jul 15 06:28	behavior.span9	(“fleshed-out” version of behavior.new)
Sep 23 16:32	behavior.span9.huge	(behavior.new used with HUGEBSOR4)
Jul 11 16:03	behavior.spanexp	(experimental version: behavior.span9)
Jul 15 06:26	bosdec.span9	(generates valid BIGBSOR4 input file)
Sep 15 23:12	bosdec.span9.hugecyl	(generates optimized corrugated cyl.)
Sep 23 16:32	bosdec.span9.hugecyl.huge	(to be used with HUGEBSOR4)
Jul 16 07:02	bosdec.span9.leftedge	(version of bosdec with left-hand sym.)
Sep 15 23:01	bosdec.span9.smoothing	(bosdec with extra smoothing segments)
Sep 23 16:33	bosdec.span9.smoothing.huge	(to be used with HUGEBSOR4)
Jul 11 16:07	bosdec.spanexp	(experimental version of bosdec.span9)

Sep 11 07:53	span.readme	(information about "span9")
May 16 20:15	span.runstream	(sample run stream)
May 16 06:53	span9.DEF	(general information & span9 glossary)
May 16 06:53	span9.INP	(valid span9 input file for GENTEXT)
May 16 06:53	span9.PRO	(prompting file for span9 cases)
May 25 06:47	span9.glossary	(span9 variable names, definitions)
Sep 16 06:50	span9.tar.gz	(the entire span9 compressed tar file)
Jul 11 16:06	spanexp.DEF	(info about experimental span9)
Jul 11 16:04	spanexp.INP	(input file for experimental span9)
Jun 30 2012	struct.span9	("fleshed-out" version of struct.new)
Jul 11 16:03	struct.spanexp	("fleshed-out" experimental version)

Skeletal forms of the files, **behavior.span9** and **struct.span9**, are created automatically by GENOPT during the GENOPT user's long interactive session associated with the GENOPT command, GENTEXT. It is the responsibility of the GENOPT user to "flesh out" these skeletal libraries (originally called behavior.new and struct.new) so that they compute any "behaviors" (vibration, buckling, stress, etc.) that may constrain the design during optimization cycles. The "fleshed out" versions are archived in the files, **behavior.span9** and **struct.span9** (included in the file, ../genopt/case/span/span9.tar.gz, of [3]). The FORTRAN file, **behavior.span9**, contains several subroutines, some written entirely by GENTEXT and others written partly by GENTEXT (the skeletal version) and partly by the GENOPT user (the "fleshed-out" version). The file, **behavior.span9**, includes SUBROUTINES BEHX1 (computes local buckling with the use of BIGBOSOR4), BEHX2 (computes general symmetric buckling with the use of BIGBOSOR4), BEHX3 (computes general anti-symmetric buckling with the use of BIGBOSOR4), BEHX4 (computes knocked-down "classical" buckling of every cylindrical segment from the formula, $K \times 0.605 \times E \times t/r$, in which K is a knockdown factor interpolated from the curve labeled "A design recommendation" plotted in Fig. A14), BEHX5 (computes the maximum effective stress in the axially compressed corrugated panel), and OBJECT (computes the objective function, which in the "span9" formulation is the weight of the entire panel of width, WIDTH, and length, LENGTH).

In the generic "span9" formulation, the FORTRAN subroutine called **struct.span9** is almost entirely automatically written by the GENOPT processor, GENTEXT. Only three lines, having to do with opening, rewinding and closing files, are added by the GENOPT user.

The file, **behavior.span9.huge**, is to be used in connection with a new version of BIGBOSOR4 called "HUGEBOBOR4". Details about this new version of BIGBOSOR4 with examples are given in Appendix 6.

The file, **bosdec.span9** (included in the file, ../genopt/case/span/span9.tar.gz, of [3]), is entirely written by the GENOPT user. The purpose of the FORTRAN code, **bosdec.span9**, is to generate a valid input file for BIGBOSOR4, which is the shell-of-revolution/prismatic shell analyzer that computes the various vibration, buckling, and stress behaviors that might constrain the design during optimization cycles. If a GENOPT user wants to optimize something that does not involve or use BIGBOSOR4, then the "bosdec" and "addbosor4" files play no role. However, dummy "bosdec" and "addbosor4" libraries must be provided in order that a successful compilation will follow via the GENOPT user's command, "genprograms". (The file, **bosdec.span9**, has been superseded by the two files, **bosdec.span9.hugecyl** and **bosdec.span9.smoothing**.)

The two files, **bosdec.span9.hugecyl.huge** and **bosdec.span9.smoothing.huge**, are to be used in connection with a new version of BIGBOSOR4 called "HUGEBOBOR4". Details about this new version of BIGBOSOR4

with examples are given in Appendix 6. The results given in Fig. 52 were generated with the use of **bosdec.span9.smoothing.huge** and **HUGEBOBOR4**.

The three files, **span9.INP**, **span9.PRO** and **span9.DEF** (included in the file, .../genopt/case/span/span9.tar.gz, of [3]), are created automatically by the GENOPT processor, GENTEXT. **span9.INP** contains an image of the GENOPT user's long interactive GENTEXT session; **span9.PRO** is a prompting file created automatically by GENTEXT from the input words, phrases, and "help" paragraphs created by the GENOPT user during his/her long interactive GENTEXT session; **span9.DEF** is a file created automatically by GENTEXT. It contains general information about GENOPT and a glossary of variable names, definitions and properties established by the GENOPT user during his/her interactive GENTEXT session. The file, **span9.glossary** (Table 1 in this paper), is the "glossary" part of the file, span9.DEF.

The files containing the string, "**spanexp**", represent "experimental" versions of the analogous files without the letters "exp" in the names. In this "experiment" separate behavioral constraints were established for local and general buckling with $n = 1$ and $n = 2$ and $n = 3$ axial half-waves over the reduced axial length, FACLEN x LENGTH, for local buckling and over the entire axial length, LENGTH, for general buckling. The first author thought that by means of this "**spanexp**" device optimization cycles would lead more easily to "global" optimum designs of smaller weight because there would be less "zigzag" behavior from optimization cycle to cycle than usually occurs without this refinement. (See Figs. 25a and 25b, for examples of "zigzag" behavior.) The thinking was that the "zigzag" behavior might be caused by alternating designs in which the most critical buckling margins correspond to alternating numbers of axial half-waves. However, this appears not to be the case. The "zigzag" behavior persists, and the experiment was therefore deemed a failure. Without the "**spanexp**" experiment the design margins relating to local and general buckling have the form listed above and repeated here for convenience:

MARGINS CORRESPONDING TO CURRENT DESIGN (F.S.= FACTOR OF SAFETY)

MARGIN NO.	CURRENT VALUE	DEFINITION
3	5.380E-03	(LOCBUK(1)/LOCBUKA(1))/LOCBUKF(1)-1; F.S.=2.00
4	2.171E-03	(BUKSYM(1)/BUKSYMA(1))/BUKSYMF(1)-1; F.S.=1.50
5	8.285E-03	(BUKASY(1)/BUKASYA(1))/BUKASYF(1)-1; F.S.=1.50

With the "**spanexp**" experiment there are three design margins relating to each type of buckling, and these design margins have the following form:

MARGIN NO.	CURRENT VALUE	DEFINITION
3	2.423E-03	(LOCBUK(1,1)/LOCBUKA(1,1))/LOCBUKF(1,1)-1; F.S.=2.00
4	4.885E-02	(LOCBUK(1,2)/LOCBUKA(1,2))/LOCBUKF(1,2)-1; F.S.=2.00
5	1.977E-01	(LOCBUK(1,3)/LOCBUKA(1,3))/LOCBUKF(1,3)-1; F.S.=2.00
6	-4.203E-02	(BUKSYM(1,1)/BUKSYMA(1,1))/BUKSYMF(1,1)-1; F.S.=1.50
7	5.596E-02	(BUKSYM(1,2)/BUKSYMA(1,2))/BUKSYMF(1,2)-1; F.S.=1.50
8	2.990E-01	(BUKSYM(1,3)/BUKSYMA(1,3))/BUKSYMF(1,3)-1; F.S.=1.50
9	-3.428E-02	(BUKASY(1,1)/BUKASYA(1,1))/BUKASYF(1,1)-1; F.S.=1.50
10	-1.756E-02	(BUKASY(1,2)/BUKASYA(1,2))/BUKASYF(1,2)-1; F.S.=1.50
11	7.550E-02	(BUKASY(1,3)/BUKASYA(1,3))/BUKASYF(1,3)-1; F.S.=1.50

With the “**spanexp**” experiment the arrays for LOCBUK, BUKSYM and BUKASY are two dimensional, with the second array index referring to the number of axial half-waves over the appropriate axial length included in the model, either the reduced axial length, $FACLEN \times LENGTH$, for LOCBUK or the entire axial length, $LENGTH$, for BUCSYM and BUCASY.

The file, **bosdec.span9.leftedge**, contains a version of **bosdec.span9** in which symmetry/anti-symmetry conditions are applied to both left-hand and right-hand longitudinal edges of the “span9” model (“**NEW**” boundary conditions defined in Section 4), rather than the “u,v,w held; rotation free” condition applied to the left-hand edge and symmetry/antisymmetry applied to the right-hand edge (“**OLD**” boundary conditions defined in Section 4). Figures 9 – 12, for example, are derived from the **bosdec.span9.leftedge** model rather than the **bosdec.span9** model. The purpose of the **bosdec.span9.leftedge** model is to permit the extension of the various “fold” models to BIGBOSOR4 models of wide panels with repeating cross sections, such as the wide panel model displayed at the top of Figs. 11 and the cylindrical shell model displayed in Fig. 33, for example. (The file, **bosdec.span9.leftedge**, has been superseded by the two files, **bosdec.span9.hugecyl** and **bosdec.span9.smoothing**.)

The “wide panel” (“leftedge”) capability described in the previous paragraph was expanded to permit the analysis of a huge cylindrical shell of radius $RCYL$. The wall of this huge cylindrical shell is a previously optimized corrugation cross-section profile with repeating geometry as described in the last paragraph. The “flat” wide panel (“flat” = flat in an average sense) such as that shown at the top of Fig. 11 is mapped onto a cylindrical surface. An example is shown in Figs. 33 and 34. SUBROUTINE BOSDEC2 was created and modified to incorporate this new capability into the “span9” software. The new subroutine bosdec2 is included in the files called “**bosdec.span9.hugecyl**” and **bosdec.span9.smoothing**. Also included in these two files is FORTRAN coding to generate valid input files for STAGS. This coding (labeled “STAGS stuff”) produces a file called *.inp3, in which “*” represents the specific case name. The STAGS model shown in the top frame of Fig. 42 was generated from FORTRAN coding included in the file, **bosdec.span9.hugecyl**, for example.

The file, **bosdec.span9.smoothing**, “smooths” away the “corners” visible in Fig. 9 (top), for example, by inserting in the BIGBOSOR4 model an additional segment between each pair of adjacent major segments, such as depicted in Fig. 28. Smoothing the corners greatly reduces large local bending strains that would occur at corners during fabrication. This added “span9” capability is activated only if there are no sub-segments. If the “corner” angle between two adjacent segments is less than 20 degrees there is no extra “smoothing” segment inserted. If there are two internal adjacent major segments with the same “convex up” or “convex down” orientation, smoothing does not occur between any of the major segments. “Smoothing” edge segments are also added at the left-hand and right-hand longitudinal edges. Their purpose is to permit post-optimization reflection and extension of the half-width model used for optimization without the introduction of cusps in the reflected and extended cross-section profile. The bottom frame in Fig. 31 shows the cross section of an extended cross section profile in which a “smoothing” edge segment prevents the formation of a cusp at the mid-width of the extended model. The panel weight is computed as if there were no smoothing. The FORTRAN files, **bosdec.span9.hugecyl** and **bosdec.span9.smoothing**, are based on symmetry/antisymmetry applied along both the left-hand and right-hand longitudinal edges (“**NEW**” boundary conditions defined in Section 4). When “smoothing” is present the “x-coordinate” (called “**RADIUS**” in many of the figures) is shifted so that the left-hand end of the panel cross-section profile corresponds to a non-zero value of x (non-zero **RADIUS**). The file, **bosdec.span9.smoothing**, sets $RSMOOTH=1.0$, in which $RSMOOTH$ is the radius of each of the added little cylindrical segments that smooth the transitions from one major segment to the next. Figures 27 – 31 demonstrate the “smoothing” geometry and capability.

Section 6. FINDING OPTIMUM DESIGNS

The following points pertain to optimization of specific cases that belong to the generic **span9** system:

1. The GENOPT processor called SUPERDUPEROPT may be executed. SUPERDUPEROPT is a script by means of which several sequential executions of SUPEROPT are automatically conducted. The End user is asked to provide two input data:

- a. The number of sequential executions of OPTIMIZE for each execution of AUTOCHANGE.
- b. The number of sequential executions of SUPEROPT to be conducted during SUPERDUPEROPT.

Each execution of SUPEROPT requires from one to six hours of computer time, depending on the number of segments/sub-segments, on the first author's very fast workstation. In many of the specific cases reported here there existed up to six automated sequential executions of SUPEROPT following the command, SUPERDUPEROPT.

2. In this study, GENOPT often has difficulty finding a "global" optimum design. The SUPEROPT process often converges several times to objectives (WEIGHTs) that are significantly higher than that corresponding to the "best" design, that is, a design with the lowest objective (WEIGHT) that is either "FEASIBLE" or "ALMOST FEASIBLE". See Fig. A8, for example, in which results for the case called "fold98" are shown. The optimized design displayed in the middle frame of Fig. A8 was obtained after the first execution of SUPEROPT. No better (smaller weight) design was obtained during the following 5 automated executions of SUPEROPT during the long execution of SUPERDUPEROPT, even though the bottom frame of Fig. A8, obtained via a different (now forgotten) sequence of computer runs, demonstrates that a better (smaller weight) design exists. The same holds for the case, "fold916", results for which are shown in Figs. A9 – A11. The same holds for the case, "wide98", results for which are shown in Fig. A12.

3. The quality of the optimized "global" optimum design often depends on the End user's specification of lower and upper bounds of decision variables he or she provides during execution of the GENOPT processor, DECIDE. For example, if, in the specific case called "fold98" (NSEG=8, Fig. A8), lower bounds of SUBWID(i), i=1, NSEG are set to 5.0 inches instead of 1.0 inch and if the lower bounds of PHISEG(i), i = 1, NSEG are set to 10.0 degrees instead of 5.0 degrees and if the upper bounds of YPLATE(i), i = 1, NSEG are set to 50.0 inches instead of 100.0 inches and if the lower bound of PHIBIG is set to 10.0 degrees instead of 1.0 degree, the optimized design obtained from SUPERDUPEROPT has an objective (WEIGHT) of 104.5 lb instead of 111.8 lb, as is indicated the middle frame of Fig. A8. An analogous result is obtained for the case, "wide98", results for which are shown in Fig. A12. With new lower and upper bounds specified as just described for the case, "fold98", the new optimized weight for the case, "wide98", is 192.2 lb rather than 200.5 lb as indicated in the middle frame of Fig. A12. In general, restricting the permitted range of the decision variables is usually beneficial, provided, of course, that the "global" optimum design exists within that restricted range.

4. Sometimes a good "global" optimum design can be obtained by first executing SUPERDUPEROPT with use of the so-called "852" strategy (described in Section 7) followed by an execution of SUPEROPT with use of the so-called "1532" strategy (described in Appendix 5). The "852" strategy is associated with the use of 5

iterations per execution of OPTIMIZE, and the “1532” strategy is associated with the use of 15 iterations per execution of OPTIMIZE. More details are provided in Appendix 5.

Plots can be obtained of the objective, variables, and design margins versus design iterations corresponding to each execution of SUPEROPT. Examples of such plots are given in Figs 4 and 5, for example. Also, similar such plots can be obtained corresponding to a short series of executions of OPTIMIZE. Examples of such plots are given in Figs. 25a (objective versus design iterations) and Fig. 25b (margins versus design iterations).

Section 7. NUMERICAL RESULTS FROM GENOPT/BIGBOSOR4 FOR A 100 X 100-INCH PANEL

Most of the results described in this section were obtained with use of the “OLD” boundary conditions (Section 4). Figures 2 – 10, 13 – 17, 23 – 25, A2 – A11 and Tables 2 and 3 pertain to this section. In all the specific cases with the string, “fold”, as part of the case name, the overall dimensions of the “corrugated” panel are LENGTH x WIDTH = 100 x 100 inches, and the total axial load is -200000 lb (Table 2, for example). Extensive results from all of these “fold” cases are contained in several files that in turn are included in the file, **span9.tar.gz**, of [3]. These “fold” files are as follows:

----- FILES RELATED TO THE END USER'S SPECIFIC CASES = “fold*” PHASE OF THIS PROJECT

May 25 06:39	fold91.tar	(one major “convex up” segment; Fig. A1)
Jun 10 12:26	fold916.tar	(16 major “convex up” segments)
Jun 19 12:03	fold916supdn.tar	(16 “up-down” segments & “up-down” sub-segments)
Jun 9 13:03	fold916ud.tar	(16 “up-down” segs., small, fixed “down” segs.)
Jul 12 05:06	fold916updown.tar	(16 “up-down” segs., no sub-segments)
May 24 21:17	fold92.tar	(two major “convex-up” segments)
May 24 14:53	fold93.tar	(three major “convex-up” segments; Fig. A2)
May 24 14:53	fold94.tar	(four major “convex-up” segments; Fig. A2)
May 24 14:53	fold94S.tar	(4 “up” segs., each with “up” sub-segs, Fig. A6)
Jun 4 02:50	fold94supdown.tar	(4 “up-down” segs., each with sub-segments)
Jun 4 03:12	fold94updown.tar	(4 “up-down” segs., no sub-segments)
May 24 14:53	fold95.tar	(five major “convex-up” segments; Fig. A3)
May 24 14:53	fold96.tar	(six major “convex-up” segments; Fig. A3)
May 24 14:53	fold97.tar	(7 major “convex-up” segments; Figs. A4, A7)
May 27 15:40	fold98.tar	(eight major “convex-up” segments; Fig. A4)
Jun 27 15:08	fold98alt.tar	(8 “up-down” segs., small, fixed “down” segs.)
Jul 18 14:18	fold98supdown.tar	(8 “up-down” segs., each w 10 “up-down” sub-segs)
Jul 24 12:42	fold98updown.tar	(8 “up-down” segs., no sub-segs. Figs.5-9)
Jul 17 17:38	fold98updownu.tar	(8 “up-down” uniform segs., no sub-segs. Fig.16)
Jul 12 04:13	test3.tar	(results for specific case based on spanexp)

Figures 4 and 5 display, for the specific case called “fold98updown”, the evolution of the objective during the first and second executions of SUPEROPT launched automatically by a single execution of SUPERDUPEROPT. The starting design (Iteration No. 0) is shown in the middle frame of Fig. 3, and the optimized panel cross section is shown in Figs. 2 and 6 – 8. The prismatic “corrugated” panel has eight major segments over half its width, WIDTH/2, with alternating “convex surface up” and “convex surface down”,

hence the string, “updown”, in the name of the case. Figure 5 shows the evolution of the objective during the second execution of SUPEROPT. There were four additional executions of SUPEROPT during the single execution of SUPERDUPEROPT for optimization of the “fold98updown” configuration. The plots of objective versus design iteration for those additional 4 executions of SUPEROPT (not shown in this paper) resemble that shown in Fig. 5, and the “best” design did not change from that existing at the end of the second execution of SUPEROPT. With use of the “**OLD**” boundary conditions, the best FEASIBLE design weighs 92.69 lb and the best ALMOST FEASIBLE design weighs 90.75 lb. In GENOPT the convention is established that a design is called “FEASIBLE” if the most critical design margin is greater than -0.01 , and a design is called “ALMOST FEASIBLE” if the most critical design margin is greater than -0.05 .

Figure 6 shows half the cross-section of the final optimized “**fold98updown**” design obtained with use of the “**OLD**” boundary conditions. The value of the objective (weight of the entire panel of WIDTH = 100 inches and LENGTH = 100 inches) is 92.69 lb. The presence of the small gap at the symmetry plane is explained in Fig. 6. This gap shows up more clearly in the STAGS finite element model of the same optimized panel displayed in Fig. 20b. The string, “852”, in the title of Fig. 6 means the following three items:

- Item 1:** There are 8 executions of the GENOPT processor called OPTIMIZE for each execution of the GENOPT processor called AUTOCHANGE during each execution of SUPEROPT during the long SUPERDUPEROPT process.
- Item 2:** The strategy parameter called IMOVE, specified in MAINSETUP by the End user, is set equal to 5.
- Item 3:** The strategy parameter called IAUTOOF, specified in MAINSETUP by the End user, is set equal to 2.

The “852” strategy is, in the particular cases described in this paper, associated with the specification of 5 design iterations for each execution of the GENOPT processor, OPTIMIZE.

The functions of these strategy parameters are explained in the permanent GENOPT prompting file called “URPROMPT.DAT”. URPROMPT.DAT is independent of the generic case. This “URPROMPT” file is located in the directory, ...genopt/execute [3].

The top two frames of Fig. 7 show the two most critical **general** buckling modes of the optimized “**fold98updown**” panel that are anti-symmetric about the panel mid-width symmetry plane, and the bottom two frames of Fig. 7 show the two most critical **general** buckling modes that are symmetric about the panel mid-width symmetry plane. The predictions correspond to use of the “**OLD**” boundary conditions. These four plots are obtained from executions of BIGBOSOR4/bosorplot with use of the valid BIGBOSOR4 input files produced automatically by GENOPT/span9: fold98updown.BEHX3 (top two frames in Fig. 7) and fold98updown.BEHX2 (bottom two frames in Fig. 7). All of the most critical general buckling load factors are very close to 1.5, which is consistent with the specification of a factor of safety of 1.5 for general buckling as listed in Table 2. One can see from the margins listed in Table 3 for the optimized design that the margins corresponding to BUKSYM (general symmetric buckling) and BUKASY (general anti-symmetric buckling) are both critical (near zero):

MARGINS CORRESPONDING TO THE DESIGN (F.S.= FACTOR OF SAFETY)				
MAR.	CURRENT			
NO.	VALUE	DEFINITION		
4	-2.035E-03	(BUKSYM(1)/BUKSYMA(1)) / BUKSYMF(1)-1; F.S.= 1.50		
5	-6.481E-04	(BUKASY(1)/BUKASYA(1)) / BUKASYF(1)-1; F.S.= 1.50		

The top two frames of Fig. 8 show, for the optimized specific case called “**fold98updown**”, two of the most critical **local** buckling modes. They correspond to $n = 1$ axial half-wave (top frame) and $n = 2$ axial half-waves (second frame) over the **reduced axial length**, $FACLEN \times LENGTH = 0.3 \times LENGTH = 30$ inches, in which $FACLEN$ is specified by the End user as listed in Table 2. These 2 plots are obtained from executions of BIGBOSOR4/bosorplot with use of the valid BIGBOSOR4 input file produced automatically by GENOPT/span9: fold98updown.BEHX1 (top two frames in Fig. 8). The first two **local** buckling load factors are very close to 2.0, which is consistent with the specification of a factor of safety of 2.0 for local buckling as listed in Table 2. One can see from the margins listed in Table 3 for the optimized design that the margin corresponding to LOCBUK (local buckling) is critical (near zero):

MARGINS CORRESPONDING TO THE DESIGN (F.S.= FACTOR OF SAFETY)			
MAR. NO.	CURRENT VALUE	DEFINITION	
3	1.838E-03	(LOCBUK(1)/LOCBUKA(1)) / LOCBUKF(1)-1; F.S.= 2.00	

The top two frames in Fig. 8 are termed “**local**” buckling because a **reduced axial length**, $FACLEN \times LENGTH$, is specified in the BIGBOSOR4 model. This strategy of using a reduced axial length in the BIGBOSOR4 model of local buckling is employed to permit an axial half-wavelength search that includes buckling modes with very short axial half-wavelengths without having to cover a huge range in the number of axial half-waves “**n**” in order to find the minimum local buckling load factor with respect to **n**, thus decreasing the computer time required to obtain optimum designs. Also, different factors of safety are used for local and general buckling. Therefore, it would not make sense to include long-axial-wavelength buckling in the search for a critical local buckling mode. The length reduction factor, $FACLEN$, is provided by the End user, as listed near the top of Table 2.

The buckling mode and load factor associated with $n = 5$ axial half-waves over the complete axial length, $LENGTH = 100$ inches, is shown in the third frame in Fig. 8. This buckling mode very much resembles the local buckling mode in the second frame of Fig. 8 even though it corresponds to buckling with 5 axial half-waves over the complete axial length, $LENGTH = 100$ inches, and not over the reduced axial length, $FACLEN \times LENGTH = 30$ inches. It is plotted in Fig. 8 because it corresponds to a local minimum buckling load factor in the set, buckling load factors(**n**), $n = 1, 2, \dots, 80$ as listed below (bold face entry corresponding to $n = 5$ axial half-waves over $LENGTH$). This buckling mode is a kind of combination of local and general buckling. The buckling load factor, 1.8498, is less than 2.0, the factor of safety used for local buckling, and greater than 1.5, the factor of safety used for general buckling. A question arises: is the design safe even though the factor of safety for this particular buckling mode is less than 2.0? We do not know the answer to this question, but our intuition tells us that the factors of safety used for general and local buckling and the knockdown factors used for “classical” buckling are more than conservative enough to produce safe optimum designs in the generic **span9** application.

***** EIGENVALUES AND MODE SHAPES FROM BIGBOSOR4 MODEL OF “**fold98updown**” *****
 EIGENVALUE (**n** AXIAL HALF WAVES OVER THE FULL AXIAL LENGTH = 100 INCHES)
 (Buckling mode is symmetric at the symmetry plane at $x = WIDTH/2 = 50$ inches.). The buckling modes corresponding to $n = 1$ and $n = 2$ are shown in the bottom two frames of Fig. 7; the buckling modes corresponding to $n = 5$ and $n = 41$ are shown in the bottom two frames of Fig. 8.

```
=====
1.5026E+00( 1); 1.4969E+00( 2); 1.8910E+00( 3); 1.9157E+00( 4); 1.8498E+00( 5);
1.9398E+00( 6); 2.0915E+00( 7); 2.1875E+00( 8); 2.2330E+00( 9); 2.2632E+00( 10);
```

2.2862E+00(11); 2.3032E+00(12); 2.3146E+00(13); 2.3226E+00(14); 2.3314E+00(15);
 2.3432E+00(16); 2.3573E+00(17); 2.3715E+00(18); 2.3846E+00(19); 2.3964E+00(20);

 2.4072E+00(21); 2.4171E+00(22); 2.4262E+00(23); 2.4347E+00(24); 2.4426E+00(25);
 2.4502E+00(26); 2.4575E+00(27); 2.4646E+00(28); 2.4716E+00(29); 2.4784E+00(30);
 2.4852E+00(31); 2.4919E+00(32); 2.4986E+00(33); 2.5051E+00(34); 2.5116E+00(35);
 2.5180E+00(36); 2.5243E+00(37); 2.5306E+00(38); 2.5368E+00(39); 2.5480E+00(40);

2.5481E+00(41); 2.5486E+00(42); 2.5496E+00(43); 2.5509E+00(44); 2.5527E+00(45);
 2.5548E+00(46); 2.5572E+00(47); 2.5599E+00(48); 2.5629E+00(49); 2.5661E+00(50);
 2.5697E+00(51); 2.5734E+00(52); 2.5774E+00(53); 2.5815E+00(54); 2.5856E+00(55);
 2.5896E+00(56); 2.5927E+00(57); 2.5938E+00(58); 2.5918E+00(59); 2.5876E+00(60);

 2.5826E+00(61); 2.5777E+00(62); 2.5733E+00(63); 2.5697E+00(64); 2.5669E+00(65);
 2.5650E+00(66); 2.5640E+00(67); 2.5639E+00(68); 2.5649E+00(69); 2.5667E+00(70);
 2.5695E+00(71); 2.5732E+00(72); 2.5776E+00(73); 2.5802E+00(74); 2.5772E+00(75);
 2.5745E+00(76); 2.5731E+00(77); 2.5732E+00(78); 2.5747E+00(79); 2.5777E+00(80);
 =====

The first two (bold face) entries in the list above correspond to general buckling with mode shapes and eigenvalues as given in the bottom two frames of Fig. 7. The **boldface** entry above corresponding to $n = 41$ axial half-waves over the complete length, LENGTH = 100 inches, closely represents a “classical” buckling mode of a cylindrical shell. The buckling mode shape is plotted in the bottom frame of Fig. 8. The buckling load factor from BIGBOSOR4, 2.5481, is slightly lower than the “classical” buckling load factor, which is derived from the formula for buckling stress (buckling stress = $0.605 \times E \times t/r$) = 2.5753 for Poisson ratio $\nu = 0.3$, wall thickness t and local radius, r , of cylindrical segment number 3, probably because there remains a small buckling modal displacement at the boundaries of Major Segment No. 3.

Notice that in the list above the buckling load factors from BIGBOSOR4 for high numbers of axial half-waves, $40 < n < 80$, approach the “classical” buckling load factor of Major Segment No. 3: 2.5753. The “classical” buckling load factor is independent of the number n of axial half-waves. (The “classical” buckling formula is: buckling stress = $0.605 \times E \times t/r$).

The buckling margins corresponding to the knocked-down “classical” buckling load factors, CYLBUK(1,j), $j = 1, 2, \dots, \text{NEG}$, are listed in Table 3 and repeated here (with critical and near-critical margins in **bold face**), as follows:

MARGINS CORRESPONDING TO THE DESIGN (F.S.= FACTOR OF SAFETY)

MAR. NO.	CURRENT VALUE	DEFINITION
6	9.946E-02	(CYLBUK(1,1)/CYLBUKA(1,1)) / CYLBUKF(1,1)-1; F.S.= 1.
7	1.326E+00	(CYLBUK(1,2)/CYLBUKA(1,2)) / CYLBUKF(1,2)-1; F.S.= 1.
8	7.281E-03	(CYLBUK(1,3)/CYLBUKA(1,3)) / CYLBUKF(1,3)-1; F.S.= 1.
9	2.600E+00	(CYLBUK(1,4)/CYLBUKA(1,4)) / CYLBUKF(1,4)-1; F.S.= 1.
10	3.630E-01	(CYLBUK(1,5)/CYLBUKA(1,5)) / CYLBUKF(1,5)-1; F.S.= 1.
11	5.077E-01	(CYLBUK(1,6)/CYLBUKA(1,6)) / CYLBUKF(1,6)-1; F.S.= 1.
12	1.476E+00	(CYLBUK(1,7)/CYLBUKA(1,7)) / CYLBUKF(1,7)-1; F.S.= 1.
13	1.500E+00	(CYLBUK(1,8)/CYLBUKA(1,8)) / CYLBUKF(1,8)-1; F.S.= 1.

According to the listed output for the optimized “fold98updown” panel contained in the file called “fold98updown.OPM” (included in the file, fold98updown.tar, which is included in the file, .../genopt/case/span/**span9.tar.gz**, of [3]), the radius/thickness ratio of Major Segment No. 1 is 101.63, which is associated with an interpolated knockdown factor of 0.39837 computed by the span9 software from the curve labeled “A design recommendation” given in Fig. 5.18 on page 186 of the Brush and Almroth textbook [25], reproduced in modified form in this paper as Fig. A14. The product of the buckling load factor, 2.7598, and the interpolated knockdown factor, 0.39837 equals 1.0994, not far above the critical value, 1.0. A similar result holds for the “knocked down” buckling load factor of Major Segment No. 3: radius/thickness of Major Segment No. 3 is 108.91, which is associated with an interpolated knockdown factor, 0.39109. The product of the “classical” buckling load factor, 2.5753, and knockdown factor, $0.39109 = 1.0071$, very close to the critical value, 1.0.

Section 8. OPTIMIZATION OF A PANEL WITH THE BOUNDARY CONDITION ALONG THE LEFT-HAND LONGITUDINAL EDGE CHANGED FROM “U,V,W HELD; ROTATION FREE” (Fig. 6) TO “SYMMETRY/ANTISYMMETRY” (Fig. 9) AND EXTENSION TO WIDE PANELS (Fig. 11)

The results described in this section were obtained with use of the “NEW” boundary conditions (Section 4). Figures 9 – 12 pertain to this section. The specific name of the case remains “fold98updown”, and there are still eight major segments over half the panel width with alternating “convex surface up”/“convex surface down” configuration. The results in this section were produced with use of the version of “bosdec” (GENOPT-user-written subroutine that generates a valid input file for BIGBOSOR4) called “**bosdec.span9.hugecyl**” (no smoothing), which is described at the end of Section 5.

The top frame in Fig. 9 shows the new optimized panel cross section profile obtained by an execution of SUPEROPT with the previously optimized profile displayed in Fig. 6 as a starting design. The optimized weight of the re-optimized entire panel of WIDTH = 100 inches increases from 92.69 lb (Fig. 6) to 97.16 lb. The middle frame in Fig. 9 shows the critical symmetric general buckling mode and buckling load factor, and the bottom frame shows the critical local buckling mode and load factor. The optimized weight, 97.16 lb, is heavier than that previously determined, 92.69 lb, because the “NEW” boundary conditions are less restrictive than the “OLD” boundary conditions.

The new “symmetry/antisymmetry” boundary condition along the left-hand longitudinal edge permits the extension of the analysis by BIGBOSOR4 of a previously optimized panel of half-width, WIDTH/2 = 50 inches, to a panel that includes the full width, WIDTH = 100 inches (Fig. 10), and to a wide panel that includes an arbitrary number of repeating, previously optimized, cross section profiles, each of WIDTH = 100 inches (Figs. 11 and 12).

Figure 10 shows three general buckling modes obtained from the BIGBOSOR4 model of the previously optimized panel cross section. All three general buckling modes correspond to symmetry conditions being applied along both the left hand and right hand longitudinal edges. The critical (lowest) general buckling load factor (eigenvalue = 1.4741) corresponds to buckling with $n = 1$ axial half-wave that is antisymmetric about the mid-width of the panel. The second eigenvalue for $n = 1$ axial half wave (1.49909) corresponds to wide-column buckling of the entire panel and agrees very well with the buckling load factor (1.4979) and mode shape shown in the middle frame of the previous figure, as it should.

Figures 11 and 12 show buckling of a very wide panel with five repeating cross section profiles, each of the five cross section profiles the same as the cross section profile of WIDTH = 100 inches displayed in Fig. 10. General buckling predictions from BIGBOSOR4 for $n = 1$ axial half-wave over the axial length, LENGTH = 100 inches are shown in Fig. 11 and the same with $n = 2$ axial half-waves are shown in Fig. 12. The buckling load factors agree well with those in Figs. 9 and 10, as they should. With such a wide panel there are many, many eigenvalues clustered near the critical value.

It would be of interest to compute buckling load factors and mode shapes for a corrugated cylindrical panel with the same repeating cross section profile but with different overall average radii of curvature. (However, see Section 16.) Also, it would be of interest to compute buckling load factors for the same elaborate cylindrical panel with various kinds of initial geometric imperfections.

Section 9. OPTIMIZATION OF A PANEL WITH BOTH ALTERNATING “CONVEX UP”/ “CONVEX DOWN” MAJOR SEGMENTS AND ALTERNATING “CONVEX UP”/ “CONVEX DOWN” SUB-SEGMENTS

The results described in this section were obtained with use of both the “OLD” and “NEW” boundary conditions (Section 4). Figures 13 – 15 pertain to this section. The specific name of the specific case is “fold98supdwn”, the letter “s” in the name signifying the presence of sub-segments. There are 10 sub-segments in each major segment. Each sub-segment in a given major segment (Major Segment No. i) has the same half-angle, PHISUB(i). Sub-segments in different major segments may have different PHISUB(j). As with the specific case, “fold98updown”, the wall thickness is uniform over the entire panel.

From Fig. 13 it is seen that the optimized profiles with left-hand longitudinal edge boundary condition “u,v,w held; rotation free” and with left-hand longitudinal edge boundary condition “symmetry” have sub-segments with small amplitudes. The two profiles optimized with different boundary conditions are similar except that Major Segment No. 1 has different slopes at the left-hand longitudinal edge. The optimized weight of the panel with left-hand “u,v,w held; rotation free” is 94.68 lb compared to 92.69 lb for the optimized weight of the panel without any sub-segments. The optimized weight of the panel with left-hand symmetry, 98.56 lb, is fairly close to the optimized weight of the analogous panel without any sub-segments (top frame of Fig. 9; WEIGHT=97.16 lb). Notice that the overall profiles of the optimized panels with sub-segments are quite different from those without any sub-segments, which are displayed in Figs. 6 and 9.

The top three frames in Figure 14 show the optimized profiles of the panel with left-hand “u,v,w held; rotation free”. The top frame is the same as the middle frame in Fig. 13. The middle two frames show general and local buckling modes that are symmetric at the panel mid-width symmetry plane. The bottom frame shows the local buckling mode that corresponds to the lowest local buckling load factor, 1.1768, **for the same optimized profile** but with the “weaker” symmetry boundary condition along the left-hand longitudinal edge. The buckling load factor, 1.1768, is significantly smaller than the minimum required local buckling load factor, 2.0, and local buckling occurs only in the neighborhood of the left-hand edge. This too-small local buckling load factor and corresponding mode shape are avoided primarily by the slope of Major Segment No. 1 being significantly different from zero along the left-hand longitudinal edge of the optimized panel with symmetry imposed along the left-hand edge, as is demonstrated for the optimized profile displayed in the bottom frame of Fig. 13.

Figure 15 shows the starting design of the entire width of the panel with sub-segments (top frame), the smallest general buckling load factor and corresponding mode shape for the profile optimized with the left-hand edge condition set at “u,v,w held; rotation free” but with that same profile analyzed with the left-hand edge condition set at “symmetry” (middle frame), and general buckling of the re-optimized panel with left-hand symmetry. The increase in weight from 94.68 lb for the “u,v,w held; rotation free” optimized panel to 98.56 lb for the “symmetry” panel is required to raise the local buckling load factor shown in the bottom frame of Fig. 14 from 1.1768 to the required value of 2.0 and the general buckling load factor shown in the middle frame of Fig. 15 to be raised from 1.2855 to close to the minimum required value of 1.5.

Section 10. OPTIMIZATION OF PANELS WITH UNIFORM ALTERNATING “CONVEX UP”/ “CONVEX DOWN” MAJOR SEGMENTS, NO SUB-SEGMENTS

The results described in this section were obtained with use of the “NEW” boundary conditions (Section 4). Figures 16 – 19 pertain to this section. The specific name of the case is “**fold98updownu**”, the final letter “u” in the name signifying “uniform” alternating “convex up”/ “convex down” major segments, no sub-segments. The starting design is similar (but not the same as) the starting design displayed in the bottom frame of Fig. 3. There are eight major segments over half the width, $WIDTH/2 = 50$ inches, of the panel. The purpose of this section is to demonstrate that permitting the geometry of each major segment to differ (as has been done until now) leads to optimized panels with somewhat lower weight than the more ordinary corrugated panels with uniform alternating “convex up”/ “convex down” segments, which is the subject of this section.

The top frame in Fig. 16 shows the starting design and the bottom frame shows the optimized design. Because of the large initial value of the overall arching angle, $PHIBIG = 60$ degrees, the starting major segments are not actually uniform although each of the six interior cylindrical segments subtend 180 degrees [$2 \times PHISEG(k)$, $k = 2, 3, \dots, 7 = 180$ degrees], and each of the two cylindrical segments nearest the two longitudinal edges subtend 90 degrees [$2 \times PHISEG(j)$, $j = 1$ and $j = 8$ equal 90 degrees]. $PHISEG(2)$, $PHISEG(3)$, ..., $PHISEG(7)$ are each linked to $PHISEG(1)$ with the linking constant equal to 2.0, that is, $PHISEG(k) = 2.0 \times PHISEG(1)$, $k = 2, 3, \dots, 7$. $PHISEG(8)$ is linked to $PHISEG(1)$ with the linking constant equal to 1.0. The major cylindrical segments are not initially uniform because their radii vary along the half-width, $WIDTH/2$, of the panel because of the large starting value of $PHIBIG$. At the optimum design the major segments are still not exactly uniform because of the optimized non-zero value, $PHIBIG = 16.08$ degrees. However, they are nearly uniform. The optimized weight of the entire panel of width, $WIDTH = 100$ inches is 117.8 lb, significantly greater than the optimized weight, 97.16 lb, of the panel with non-uniform major segments shown in the top frame of Fig. 9.

Figure 17 shows the most critical general and local buckling modes and load factors. In this case local buckling is not critical because the smallest buckling load factors corresponding to local buckling, 2.7180 for $n = 1$ axial half-wave over the reduced axial length $FACLEN \times LENGTH = 30$ inches and 2.6595 for $n = 2$ axial half-waves, are significantly greater than the minimum allowable value, 2.0.

Figure 18 shows the starting and optimized cross sections of two wide “uniform” panels. These two BIGBOSOR4 wide panel models are analogous to the wide panel model displayed in the top frame of Fig. 11. The top two frames in Fig. 18 show the starting and optimized wide panels in which the overall arching angle, $PHIBIG$, is one of the decision variables. The bottom two frames in Fig. 18 show the starting and optimized wide panels in which the overall arching angle is not a decision variable and is assigned a very small initial

value, PHIBIG = 0.1 degree (essentially zero). The optimized weights per WIDTH of the two wide panels are almost the same, 117.8 lb/WIDTH for the panel with optimized PHIBIG = 16.08 degrees and 118.1 lb/WIDTH for the panel with PHIBIG = 0.1 degree (essentially zero).

The top three frames of Fig. 19 show the most critical general and local buckling modes and load factors of the wide panel with optimized overall arching angle, PHIBIG 16.08 degrees. The bottom frame shows the most critical general buckling mode and load factor of the optimized “flat” corrugated wide panel (PHIBIG=0.1 deg).

Section 11. COMPARISON OF PREDICTIONS FROM GENOPT/BIGBOSOR4 WITH THOSE FROM STAGS FOR THE OPTIMIZED 100 x 100-inch COMPLEX CORRUGATED PANEL (the specific case called “fold98updown”)

The results described in this section were obtained with the use of the “OLD” boundary conditions (Section 4). Figures 20 – 22 pertain to this section. The STAGS models were constructed and processed by the second author of this paper, who developed the special pre- and post-processors for STAGS known as “StagsViewer”.

Figure 20a is the STAGS model of half the width of the optimized “fold98updown” panel the cross section profile of which is displayed in Figs. 2 and 6. Buckling predictions from this STAGS model are displayed in Figs. 21 and 22. Figure 21 shows buckling load factors and corresponding buckling modes that are symmetric about the symmetry plane at $x = \text{WIDTH}/2 = 50$ inches. Figure 22 shows buckling load factors and corresponding buckling modes that are anti-symmetric about the symmetry plane at $x = \text{WIDTH}/2 = 50$ inches. There is excellent agreement between the predictions of STAGS and the predictions of BIGBOSOR4 for the optimized panel.

About STAGS and its application to the optimized span9 configuration called “fold98updown”

Version 6.1 of the STAGS computer program [26-29] was used to validate the design of the optimized configuration called “fold98updown” (eight major alternating “convex up” and “convex down” segments over the half-width, $\text{WIDTH}/2 = 50$ inches, of the complex corrugated panel shown in Figs. 2 and 6). Figure 20a shows the STAGS finite element model used for computation of the buckling modes and load factors. Figure 20b shows a STAGS model of the entire panel of width, $\text{WIDTH}=100$ inches. This “whole-width” STAGS model was not used to compute buckling loads in the final version of this paper. The presence of the very narrow longitudinal gap at the panel mid-width is explained in Fig. 6. The width of the narrow gap is $2 \times (50.0 - 49.48378) = 1.03244$ inch. It is caused by the presence of the second inequality margin listed in Table 3:

2 -1.622E-03 -3.95+0.10*V(9)+0.10*V(10)+0.10*V(11)+0.10*V(12)+0.10*V(13)+...-1.0
The gap is slightly greater than $2 \times (50 - 49.5) = 1.0$ inch because the second inequality margin, Margin No. 2 in Table 3, reproduced above, is slightly negative.

STAGS buckling prediction:

Figure 21 shows several STAGS buckling modes, from the Fig. 20a “half-width” model, corresponding to general buckling that is symmetric at the mid-width symmetry plane, and Fig. 22 shows several STAGS buckling modes corresponding to general buckling that is anti-symmetric at the mid-width symmetry plane. Half the width, $\text{WIDTH}/2$, of the panel is included in the STAGS model, as is shown in Fig. 20a. There is extremely good agreement between the predictions from STAGS and the predictions from BIGBOSOR4.

About STAGS and StagsViewer:

The capabilities and properties of the STAGS computer program [26 – 29] are summarized in Appendix 1. The “official” version of STAGS, developed by Almroth, Brogan, Rankin and others during the past 40 years, has recently been considerably improved by Michael Jacoby, who created STAGS pre- and post-processing code called “**StagsViewer**”. The StagsViewer application is written entirely in the C computer language, is intended for LINUX and UNIX based systems with graphics hardware acceleration, and employs the OpenMotif and OpenGL standards. The construction of finite element models is accomplished via a simple general purpose easy-to-use scripting language built into the StagsViewer code that contains many advanced features (such as surface patch intersections) and that simplifies the specification of many commonly used modeling techniques. Through interaction with the GUI, a user of StagsViewer can build up complex models in an intuitive way.

The StagsViewer **pre-processor** produces a STAGS model in what in STAGS jargon is called “Element Unit” input data format. The “official” version of STAGS can also generate finite element models in what in STAGS jargon is called “Shell Unit” input data format. In many cases of interest in the advanced nonlinear analysis of shells, use of the STAGS Shell Unit input data format simplifies the construction of segmented models.

In addition to finite element pre-processing, the StagsViewer can also **post-process** STAGS results in a manner similar to that of many of the large commercially available tools, such as Patran, I-deas, and others. All of the figures in this paper pertaining to STAGS finite element results were produced with use of the post-processor of StagsViewer. The StagsViewer post-processing files are generated by the STAGS processor call “STAPL”. The post-processor of StagsViewer allows for easy interactive visualization of deflected shapes, continuous and discrete color contouring of stress, strain and other result quantities, buckling and vibration mode shape animation, load step animation, and the interrogation of results. The results can be exported as encapsulated postscript or jpeg “screen shots”. One-button creation of GIF-based animations is also readily accomplished.

The STAGS finite element analyses described in this paper were performed on an Apple Macbook Pro running the Mac OSX 10.5.8 operating system, with a 2.5 Ghz Intel Core 2 Duo processor with 4 Gb of RAM memory and an NVIDIA GeForce 8600M GT graphics card.

For the most part, the details of the STAGS finite element model faithfully represent the modeling idealizations employed in the GENOPT/BIGBOSOR4 model. Only the standard features in STAGS need be used; the development of STAGS user-written subroutines was not required for the specific cases explored in the work leading to this paper.

Section 12. DESIGN SENSITIVITY OF THE OPTIMIZED CONFIGURATION CALLED “fold98updown”

The results in this section were obtained with the use of the “**OLD**” boundary conditions (Section 4). Figures 23a and 23b show a sample “design sensitivity” plots corresponding to the optimized design of the specific case called “fold98updown”. (See Figs. 2 and 6.) The “design sensitivity” plots are generated by the GENOPT/BIGBOSOR4 model from analyses in which a user-selected decision variable varies over a user-selected range while all other decision variables are held constant. A typical characteristic of these plots, which ordinarily pertain to a previously optimized design, is that several design margins become critical or nearly critical at the value of the user-selected decision variable that corresponds to its optimized value.

In this case variations of the design margins with respect to variations of the decision variables, YPLATE(8) (Fig. 23a) and YPLATE(9) (Fig. 23b), are plotted. Notice that the local buckling margin (LOCBUK) is especially sensitive to small changes in YPLATE(8) and YPLATE(9) in the neighborhoods of the optimized values of YPLATE(8) and YPLATE(9). Also, the symmetric general buckling margin (BUCSYM) is quite sensitive to small changes in YPLATE(8) and YPLATE(9) in the neighborhoods of the optimized values of YPLATE(8) and YPLATE(9). These properties often make it very difficult to find a “global” optimum design.

Another example of design sensitivity of a previously optimized corrugated panel (specific case = “fold916”) is described in Appendix 3. (See Figs. A9 – A11.)

Section 13 DIFFICULTY FINDING “GLOBAL” OPTIMUM DESIGNS

The results described in this section were obtained with use of the “**OLD**” boundary conditions (Section 4). Figures 23 – 26 and Figs. A8, A9, A12, A15–A17 pertain to this section. Perhaps because of the behavior shown in Figs. 23a and 23b, that is, the high sensitivity of the optimum designs to changes in certain of the decision variables, unique “global” optimum designs are often difficult to find. In the previous sentence “global” is in quotes because the strategy used in GENOPT cannot rigorously determine a true global optimum design. Instead GENOPT attempts to find a design for which the objective is likely to be near that of a true global optimum by starting from many different points in design space during a single execution of SUPEROPT or SUPERDUPEROPT (Figs. 4 and 5 and the top two frames in Fig. A11). For the complex corrugated panel configurations GENOPT often arrives at quite different “global” optimum designs and objectives (panel weights) when different strategies are specified by the End user in MAINSETUP.

Typical input data for MAINSETUP (a *.OPT file) are as follows:

```

n          $ Do you want a tutorial session and tutorial output?
0          $ Choose an analysis you DON'T want (1, 2,...), IBEHAV
0          $ NPRINT= output index (0=GOOD, 1=ok, 2=debug, 3=too much)
1          $ Choose type of analysis (1=opt., 2=fixed, 3=sensit.) ITYPE
5          $ How many design iterations in this run (3 to 25)?
n          $ Take "shortcuts" for perturbed designs (Y or N)?
2          $ Choose 1 or 2 or 3 or 4 or 5 for IDESIGN
5          $ Choose 1 or 2 or 3 or 4 or 5 for move limits, IMOVE
n          $ Do you want default (RATIO=10) for initial move limit jump?
100000    $ Provide a value for the "move limit jump" ratio, RATIO
y          $ Do you want the default perturbation (dx/x = 0.05)?
n          $ Do you want to have dx/x modified by GENOPT?
n          $ Do you want to reset total iterations to zero (Type H)?
2          $ Choose IAUTOFF= 1 or 2 or 3 or 4 or 5 or 6 to change X(i)

```

“Choose an analysis you DON’T want”, IBEHAV = 0, means “do all the analyses, that is, run all the SUBROUTINES BEHX_i, i = 1, 2, ..., number of behaviors”. (number of behaviors = 5 in the present application of GENOPT: BEHX₁ = LOCBUK, BEHX₂ = BUCSYM, BEHX₃ = BUCASY, BEHX₄ = CYLBUK, BEHX₅ = STRESS). The index for “type of analysis”, ITYPE = 1, means “do optimization”. “How many design iterations in this run” means “how many iterations **for each execution of OPTIMIZE**”. “Take ‘shortcuts’ for perturbed designs” is explained in the permanent GENOPT prompting file, URPROMPT.DAT. The index IDESIGN = 2 means “accept designs that are either FEASIBLE or ALMOST FEASIBLE”. Move limit, IMOVE = 5, means “permit decision variables to change by as much as 5 per cent during a given design

iteration”. The remaining 6 input data for MAINSETUP are explained in the permanent GENOPT prompting file, URPROMPT.DAT.

Figure 24 shows two quite different “global” optimum designs of the “fold98updown” configuration arrived at with the use of different sequences of computer executions. The weights (objectives) of these two “global” optimum designs are within one per cent of each other, yet the optimized configurations are quite different from each other. Figure A8 shows another example of different “global” optimum designs arrived at with the use of different strategies in MAINSETUP. In the Fig. A8 example both the optimum designs and especially the optimum weights are significantly different from each other. Figures A9 and A12 show yet additional examples.

Figure 25a shows plots of the objective versus design iterations obtained in three cases, (A), (B), (C), from four sequential executions of the GENOPT processor, “OPTIMIZE”, with 5 design iterations specified for each execution of OPTIMIZE, and in one case, (D), from one execution of OPTIMIZE with 25 design iterations specified for that single execution of OPTIMIZE. The four different optimization strategies, (A – D) are specified by the End user in the GENOPT processor, MAINSETUP, the input data file for which are listed in a file called “fold98updown.OPT” [3 and listed above]. The strategy parameter called “IMOVE” governs the move limits of the decision variables during design iterations: IMOVE = 1 means the move limit of any decision variable is a maximum of 10 per cent of that decision variable per iteration; IMOVE = 5 means the move limits are 5 per cent per iteration; IMOVE = 4 means the move limits are 2 per cent per iteration; IMOVE = 3 means the move limits are 1 per cent per iteration. The strategy parameter, IAUTOOF, governs how a new “starting” design (Fig. 4) is arrived at during execution of the GENOPT processor called “AUTOCHANGE”, as described in the permanent GENOPT prompting file called “URPROMPT.DAT” [3]. IAUTOOF is not active during “OPTIMIZE” executions and therefore has no effect on the results displayed in Figs. 25a and 25b. During each execution of “OPTIMIZE” the “window” of permitted excursions of the decision variables is shrunk by a certain percentage from design iteration to design iteration. For example, Fig. 25(D) implies convergence to an objective (panel weight) close to 94 lb. However, this is a false impression. The appearance of convergence exists only because, after so many design iterations during that one execution of “OPTIMIZE”, the “window” of permitted excursions of the decision variables has been shrunk to a tiny size.

It is therefore recommended that GENOPT users generally specify 5 iterations per execution of OPTIMIZE, and execute OPTIMIZE multiple times either “by hand” via the command, “OPTIMIZE”, or automatically via the command “SUPEROPT” or the command “SUPERDUPEROPT”, rather than specify a large number of design iterations for each execution of SUPEROPT. Design space is more completely explored by the strategy of using only 5 iterations per execution of OPTIMIZE because consequently there are more intermediate “starting” designs (spikes in Figs. 4 and 5, for example) produced by the GENOPT processor called “AUTOCHANGE” than would exist during an execution of SUPEROPT or SUPERDUPEROPT with the specification of more design iterations per execution of OPTIMIZE. However, see Appendix 5 for a contrary optimization strategy.

The starting design in Figs. 25a and 25b is the FEASIBLE optimum design shown in Figs. 2 and 6.

Figure 25b shows the evolution of several of the most critical design margins during the same computations described in connection with Fig. 25a. The four different strategies, to which each frame in Figs. 25a and 25b correspond, lead to four different “global” optimum designs.

During the course of this project it was found that the best optimization strategy seemed to be the following:

1. For “fold9” cases with fewer than about 10 major segments over half the panel width, $WIDTH/2$, it seems best to start with an execution of SUPERDUPEROPT with 3 automatic successive executions of SUPEROPT with use of the “611” strategy, meaning: 6 executions of OPTIMIZE for each execution of AUTOCHANGE, $IMOVE = 1$ and $IAUTOOF = 1$. Use 5 design iterations per execution of OPTIMIZE. Possibly it is best to eliminate this step for “fold9” cases with more than 10 major segments over the panel half-width, $WIDTH/2$, although including this step can do no harm; only time may be wasted.
2. Execute SUPERDUPEROPT with 3 automatic successive executions of SUPEROPT again, this time with use of the “852” strategy, meaning: 8 executions of OPTIMIZE for each execution of AUTOCHANGE, $IMOVE = 5$ and $IAUTOOF = 2$. Use 5 design iterations per execution of OPTIMIZE.
3. Execute SUPERDUPEROPT with 3 automatic successive executions of SUPEROPT again, this time with use of the “1532” strategy, meaning: 15 executions of OPTIMIZE for each execution of AUTOCHANGE, $IMOVE = 3$ and $IAUTOOF = 2$. Use 15 design iterations per execution of OPTIMIZE.

Figure 26 shows a comparison of objective versus design iterations from the “611” and “852” strategies. Both strategies yield “zigzag” results, but in this particular case the “852” strategy produces a significantly lighter-weight “ALMOST FEASIBLE” corrugated panel than the “611” strategy.

More on the difficulty of finding “global” optimum designs is written in Appendices 3 and 4, in particular the discussion associated with Figs. A8 – A12, and in the discussion in Appendix 5 associated with Figs. A15 – A18. Appendix 5 provides a recommended strategy that sometimes works for determining “global” optimum designs in the specific “fold9” cases with more than about 10 major segments over the half-width of the panel. These “fold9” cases involve many decision variables. **The more decision variables the harder it seems to be to find a “global” optimum design.**

This difficulty in finding “global” optimum designs with use of the gradient-based method (ADS) implemented in GENOPT is definitely a weakness in the GENOPT software. Perhaps the “Response Surface Method” of Haftka and his colleagues [10 – 15] should be introduced into GENOPT.

Section 14. “SMOOTHING” AWAY THE “CORNERS”

The results described in this section were obtained with the use of the “NEW” boundary conditions (Section 4). Figures 27 – 31 pertain to this section. As described at the bottom of Section 5, the FORTRAN subroutine called **“bosdec.span9.smoothing”** is used to generate BIGBOSOR4 models in which the “corners” between adjacent major segments of the panel cross-section profile are eliminated by the introduction of additional transitional small cylindrical segments with radius, $RSMOOTH$, between major segments. **Smoothing the corners greatly reduces large local bending strains that would occur at corners during fabrication.** Also, new “edge” segments are added at the left-hand and right-hand longitudinal edges in order to ensure that the slope of the cross section in the x-direction (width direction) is zero at the two planes of symmetry, a strategy that enables the extension to BIGBOSOR4 models of practical wide panels with cross sections that consist of multiple profiles of width, $WIDTH$, analogous to that shown in the top frame of Fig. 11 (for which there is no smoothing), but without the introduction of cusps. Figures 27 and 28 demonstrate this strategy, and Fig. 29 gives a comparison of the panel cross-section profiles without and with smoothing for the specific case called “fold98updown” and with $RSMOOTH = 2.0$ inches.

Figure 30 shows the influence of RSMOOTH on the critical buckling load factor corresponding to the type of buckling mode displayed at the bottom of the figure. Figure 31 demonstrates that “smoothed” panel cross-section profiles can be optimized, and the behavior of the optimized “smoothed” profiles of width = WIDTH/2 can subsequently be predicted by BIGBOSOR4 for wide panels with repeating sections each of width = WIDTH without the introduction of cusps between adjacent sections of WIDTH=100+ inches. In the bottom frame of Fig. 31 the string, “100+ inches”, means that narrow cylindrical “smoothing” segments are attached to the left-hand side of Major Segment No. 1 and to the right-hand side of Major Segment No. NSEG, as demonstrated in the top frame of Fig. 31. Therefore, the width of the “smoothed” panel cross-section profile is slightly larger than WIDTH, which is specified by the End user as 100 inches in this case. Compare the weight, WEIGHT = 95.93 lb, of the optimized cross-section profile with smoothing present during optimization cycles (middle frame of Fig. 31) with the weight, WEIGHT = 97.16 lb, of the optimized cross-section profile without smoothing during optimization cycles (top frame of Fig. 31). If the optimized smoothed design is analyzed with smoothing absent, that non-smoothed “equivalent” (“corners” present and the values of the decision variables equal to those of the optimized smoothed design) is no longer FEASIBLE: the design margin for symmetric general buckling (BUKSYM) has a value equal to -0.104 while all the other design margins remain positive.

Section 15. OPTIMIZATION OF A PANEL THAT CAN BE FABRICATED BY STAMPING

The results described in this section were obtained with the use of the “NEW” boundary conditions (Section 4). Figure 32 pertains to this section. A panel cross-section profile with no segments that “double back” can be fabricated by stamping. The top frame of Fig. 32 shows one segment, Major Segment No. 4, that “doubles back”: there are two non-neighboring “axial stations” that have the same x-coordinate (horizontal direction). (“Axial station” is a label automatically provided by the BIGBOSOR4 plot routine. This label is unfortunate for the geometries of interest here. The label, “Axial station”, in the present application should be interpreted as “coordinate normal to the x-z plane”).

In the GENOPT processor, DECIDE, the upper bounds of the four decision variables, PHISEG(2), PHISEG(3), PHISEG(4) and PHISEG(5), were reduced from 85 degrees to 60 degrees, and the panel was re-optimized. The re-optimized panel cross-section profile is shown in the middle frame of Fig. 32. There seems to be a tiny amount of “doubling back” in Major Segment No. 4 where it joins Major Segment No. 5. However, “smoothing” the transition from Segment 4 to Segment 5 eliminates this tiny amount of “doubling back”, as demonstrated in the bottom frame of Fig. 32.

Section 16. BUCKLING OF AN OPTIMIZED “FLAT” PANEL MAPPED ONTO A CYLINDER

The results described in this section were obtained with the use of the “NEW” boundary conditions (Section 4). Figs 33 – 40 pertain to this section. Except for the configurations shown in Figs. 38 and 39 (uniform corrugations), the optimized panel cross-section profile used for this analysis is that displayed in the top frame of Fig. 11. The panel shown in Fig. 11 is flat in an average sense. There is no further optimization of the cross-section profile from that displayed in the top frame of Fig. 9. Figure 33 shows 90 degrees of the cylindrical shell of average radius, RCYL, and the same “flat” profile reproduced from the top frame in Fig. 11. The (r, z) coordinates of the cylindrical shell are indicated in Fig. 33. In the mapping of the “flat” surface into a cylindrical surface it is assumed that the reference surface of the cylindrical shell does not stretch or shrink from that for the “flat” panel. Therefore, the cylindrical arc length of WIDTH = 100 inches indicated at the top of

Fig. 33 is the same length as that of the first of five repeating sections shown for the “flat” panel at the bottom of Fig. 33. The mapping equations are:

$$\begin{aligned}\text{Theta} &= x/\text{RCYL} \\ r &= (\text{RCYL} + Z - \text{ZAVE})\sin(\text{Theta}) \\ z &= (\text{RCYL} + Z - \text{ZAVE})\cos(\text{Theta})\end{aligned}$$

in which Theta is identified in the top frame of Fig. 33, and x, Z and ZAVE are identified in the bottom frame of Fig. 33. ZAVE is the distance from the x-z plane to what will become the reference surface of the cylindrical shell, that is, ZAVE is an average vertical distance of a module in the “flat” panel profile above the x-z plane. Z is the distance from the x-z plane to any point that lies on the local reference surface within any of the major segments of the “flat” corrugated panel cross-section profile. The mapping scheme means that any segments that lie mostly outside the reference surface of the cylindrical shell, that is, $Z - \text{ZAVE} > 0$, will be stretched slightly in the circumferential direction from their shapes in the “flat” panel, and any segments that lie mostly inside this cylindrical reference surface, that is, $Z - \text{ZAVE} < 0$, will be shrunk slightly in the circumferential direction. The FORTRAN software used to create the BIGBOSOR4 model shown in Fig. 33 is called SUBROUTINE BOSDEC2, and this subroutine is part of the file called **bosdec.span9.hugecyl** or part of the file called **bosdec.span9.smoothing** (also, part of the two analogous most recent files to be used in connection with HUGEBOOR4: **bosdec.span9.hugecyl.huge** and **bosdec.span9.smoothing.huge**, respectively).

Figure 34 shows buckling modes and load factors for the “cylindrical” (in an average sense) and “flat” (in an average sense) complex corrugated panels called “fold98updown”. Explanations are given in the figure and in its caption. The buckling modes and load factors of the “cylindrical” and “flat” profiles are similar. Ning and Pellegrino [30] have found that the buckling loads of axially compressed “wavy” (uniformly corrugated) cylindrical shells are insensitive to initial imperfections. It is likely that the buckling loads of axially compressed complexly corrugated cylindrical shells of the type shown in Figs. 33 and 34 will also be insensitive to imperfections. In order for one to prove this analytically, one would have to create STAGS models. Imperfections in the shapes of one or more buckling modes would have to be introduced into these STAGS models, which would then have to be run in a nonlinear static equilibrium branch of the program, perhaps combined with intermittent nonlinear dynamic analyses in order to trace the nonlinear load path to collapse.

Figure 35 shows buckling modes and load factors of complexly corrugated complete (360-degree) cylindrical shells. The local cross section profiles, each of WIDTH = 100 inches, are the same as the undeformed images in Fig. 10 and for each “flat” segment of WIDTH = 100 inches (flat in an average sense) shown in the top frame of Fig. 11 except that these originally average “flat” segments, laid out around the circumference of the cylindrical surface, are now bent to conform to the average cylindrical surfaces with radii RCYL = 95.493 inches (left side) and RCYL = 286.48 inches (right side). The image on the left-hand side of Fig. 35 corresponds to MMM = 6 (total “cylindrical” arc length = 6 x WIDTH = 600 inches) and the image on the right-hand side corresponds to MMM = 18 (total “cylindrical” arc length = 18 x WIDTH = 1800 inches). The values of RCYL are established so that mapping the “flat” panels with arc lengths of 600 inches and 1800 inches onto cylindrical surfaces of average radii 95.493 and 286.48 inches, respectively, creates complete (360-degree) cylindrical shells. The sensitivity of the buckling load factor to the amplitude of the average “flattened” circumferential portions of the average cylindrical arc is very small: buckling load factor = 1.4163 on the left (where the amplitude of average flattening is rather large), and buckling load factor = 1.4579 on the right (where the amplitude of average flattening is rather small). This result is in keeping with that determined by Ning and Pellegrino in [30] for uniformly corrugated axially compressed cylindrical shells.

It is beneficial to compare the optimized weight of 90 degrees of the cylindrical corrugated panel (485.8 lb indicated near the top of Fig. 33) with the optimized weights of different kinds of uniformly axially compressed aluminum cylindrical shells:

Example 1: a cylindrical shell with external T-shaped stringers (Fig. 36)

Example 2: a cylindrical shell with a truss-core sandwich wall (Figs. 37a, 37b and A19)

Example 3: a cylindrical shell with **uniform** “convex up”/”convex down” corrugations. (Figs. 38 and 39)

Example 1: an aluminum cylindrical shell with aluminum external T-shaped stringers (a case called “pandafold”)

The computer program PANDA2 [31] was used to obtain an optimized design of the cylindrical shell with external T-shaped stringers.

Figure 36 shows the geometry and buckling modes for the optimized configuration. Less than 90 degrees of the optimized cylindrical shell are included in this post-optimized very large BIGBOSOR4 model because the upper limit of the maximum allowed number of segments (295 segments) corresponds to a significantly smaller part of the circumference of the huge cylindrical shell. However, the PANDA2 model is for a cylindrical panel that subtends 180 degrees, and all the buckling modes shown in Fig. 36 indicate that the much smaller sector displayed in Fig. 36 is wide enough to capture all the critical buckling modes with good accuracy. In the GENOPT model of the panel of WIDTH = 100 inches a uniform total axial compression of 200000 lb is applied, which corresponds to a compressive axial resultant of 2000 lb per circumferential inch. Therefore, the axial resultant in the PANDA2 model (resultant normal to the plane of the paper) is -2000 lb/in. In the PANDA2 model the factor of safety for general buckling was set to 1.5. The factor of safety for local buckling was set to 1.0. Figure 36 demonstrates that for the optimized design there are three critical buckling margins: general buckling (top frame), lateral-torsional buckling (middle frame) and local buckling (bottom frame). According to PANDA2 the optimized weight of 90 degrees of the externally T-stringer stiffened cylindrical shell is 658 lb, significantly greater than the optimized weight of 90 degrees of the equivalent complex corrugated cylindrical shell: 485.8 lb. (The optimized weight of 90 degrees of the T-stringer-stiffened cylindrical shell should actually be more than 658 lb because setting the factor of safety for local buckling (1.0 in this case) to a value less than that for general buckling (1.5 in this case) while not computing local post-buckling of the skin/stringer cross section leads to an unconservative prediction of general buckling because the effective average axial stiffness of a locally post-buckled skin/stringer cross section is significantly less than that of an unbuckled cross section, leading to lower general buckling load factors.)

Example 2: a cylindrical shell with an aluminum truss-core sandwich wall (a case called “isotruss2/pandaiso”)

The GENOPT/BIGBOSOR4/trusscomp [24] software was used to obtain an optimized design of the cylindrical shell with a truss-core sandwich wall. The generic case is called “trusscomp” [24] and the specific case is called “isotruss2” (truss-core sandwich wall made of isotropic material).

Figure 37a shows the geometry and buckling modes for the optimized “isotruss2” configuration from the BIGBOSOR4 models for general (top two frames) and local (bottom frame) instability. As with Example 1, significantly less than 90 degrees of the optimized cylindrical shell are included in the BIGBOSOR4 model for

general instability because the upper limit of the maximum allowed number of segments (295 segments) corresponds to a significantly smaller sector of the huge cylindrical shell. (The BIGBOSOR4 model shown in the top two frames of Fig. 37a has 276 segments.) The isotruss2 design margins corresponding to the optimized design are as follows:

MARGIN NO.	CURRENT VALUE	DEFINITION
1	4.352E-01	1+0.45*V(1)-1.00*V(2) -1.
2	-5.632E-03	(LOCBUK(1)/LOCBUKA(1)) / LOCBUKF(1)-1; F.S.= 1.50
3	-7.148E-03	(GENBUK(1)/GENBUKA(1)) / GENBUKF(1)-1; F.S.= 1.50
4	5.704E+01	(STRM1A(1,6)/STRM1(1,6)) / STRM1F(1,6)-1; F.S.= 1.00
4	1.151E+02	(STRM2A(1,6)/STRM2(1,6)) / STRM2F(1,6)-1; F.S.= 1.00

In the list of margins “LOCBUK” means “local buckling”; “GENBUK” means “general buckling”; “STRM” means “maximum stress”. The two stress margins are far from critical because a very high allowable (1000000 psi) was set for stress so that only buckling governs the evolution of the design during optimization cycles.

The optimized objective is:

CURRENT VALUE OF THE OBJECTIVE FUNCTION:

VAR. NO.	CURRENT VALUE	DEFINITION
	1.159E-02	weight/area of the truss-core sandwich wall: WEIGHT

The accuracy of the general buckling mode and load factor shown in the top two frames of Fig. 37a is somewhat doubtful because only a relatively small sector of the circumference of the cylindrical shell could be included in the BIGBOSOR4 model, which is limited to less than 295 shell segments. Therefore, a much bigger “BIGBOSOR4” program was developed. This much larger version of BIGBOSOR4 is called “HUGEBOSOR4”. It is described in Appendix 6, and two examples of very large “HUGEBOSOR4” models are given there. As of this writing HUGEBOSOR4 does not supersede BIGBOSOR4. It may do so at some time in the future. As of this writing HUGEBOSOR4 is supposed to be used in connection with GENOPT.

Figure 37b shows the results from a STAGS general buckling model of the optimized truss-core sandwich cylindrical shell analogous to the BIGBOSOR4 models displayed in the top two frames of Fig. 37a. There is good agreement between the BIGBOSOR4 prediction of the critical buckling mode shape and buckling load factor (1.4893 from the BIGBOSOR4 model shown in the middle frame of Fig. 37a) and the STAGS prediction (1.5022 shown in Fig. 37b).

The computer program, PANDA2 [31] was used to compute design margins and objective of the optimized design identified between the middle and bottom frames in Fig. 37a. In the PANDA2 model 90 degrees of the cylindrical shell were included in the model. The PANDA2 case is called “pandaiso” (meaning “**panda2-isotruss2**”). The design margins analogous to those of the optimized isotruss2 configuration given in Fig. 37a are listed in the PANDA2 file, pandaiso.OPM, as follows:

MARGINS FOR CURRENT DESIGN: LOAD CASE NO. 1, SUBCASE NO. 1		
MAR. NO.	MARGIN VALUE	DEFINITION
1	-6.32E-02	Local buckling from discrete model-1.,M=90 axial halfwaves;FS=1.5

```

2  1.17E+02 local wide-column bucking, discrete model(m=1  axial halfwav);FS=1
3  -1.77E-01 Local buckling from Koiter theory,M=90  axial halfwaves;FS=1.5
4  5.75E+01 eff.stress:matl=1,STR,Dseg=3,node=6,layer=1,z=-0.0178; MID.;FS=1.
5  -6.96E-02 Hi-m skin-stringer, discrete model-1.,M=87  axial halfwaves;FS=1.5
6  5.76E+01 eff.stress:matl=1,STR,Iseg=3,at:n=11,layer=1,z=0.0178;-MID.;FS=1.
7  1.04E+05 buckling marg.   skin   Iseg.(width-wise wide col.)MID.;FS=1.5
8  4.60E+06 buckling marg. stringer Iseg.(width-wise wide col.)MID.;FS=1.5
9  1.04E+05 buckling marg.   skin   Iseg.(width-wise wide col.)NOPO;FS=1.5
10 4.60E+06 buckling marg. stringer Iseg.(width-wise wide col.)NOPO;FS=1.5
11 -5.15E-02 buck.(DONL);simp-support general buck;M=1;N=6;slope=0.;FS=1.5
12 5.85E+02 (Max.allowable ave.axial strain)/(ave.axial strain) -1; FS=1.
13 7.81E-01 0.45  *(Stringer spacing, b)/(Stringer base width, b2)-1;FS=1.
14 2.64E-01 (Str. base width, b2)/(0.2  *(Str. spacing, b))-1; FS=1.
15 -5.65E-02 buck.(SAND);simp-support general buck;M=1;N=6;slope=0.;FS=1.5

```

and the objective listed in the pandaiso.OPM file is given by:

CURRENT VALUE OF THE OBJECTIVE FUNCTION:

VAR.	STR/	SEG.	LAYER	CURRENT	DEFINITION
NO.	RNG	NO.	NO.	VALUE	
		0	0	5.793E+02	WEIGHT OF THE ENTIRE PANEL
TOTAL WEIGHT OF SKIN					= 5.7932E+02
TOTAL WEIGHT OF SUBSTIFFENERS					= 0.0000E+00
TOTAL WEIGHT OF STRINGERS					= 0.0000E+00
TOTAL WEIGHT OF RINGS					= 0.0000E+00
SPECIFIC WEIGHT (WEIGHT/AREA) OF STIFFENED PANEL=					1.1586E-02

Several of the design margins from PANDA2 are negative because the analysis in PANDA2 is more conservative than that in GENOPT/BIGBOSOR4/trusscomp, especially with regard to the “Koiter” model of local buckling (Margin No. 3) and because a conservative knockdown factor is used in PANDA2 to compensate for neglect of the effect of transverse shear deformation in the PANDA2 analysis of general instability, which is based on “smeared” wall properties (Margins 11 and 15). However, the design according to PANDA2 is not too far from FEASIBLE, and the computed specific weight, 0.011586 lb/in², agrees with that computed in the isotruss2 case by GENOPT/BIGBOSOR4/trusscomp, as it should.

Example 3: a cylindrical shell with uniform “convex up”/“convex down” corrugations (“fold98updwnu”)

Figures 38 and 39 pertain to the specific case called “fold98updwnu”, which is the optimized panel with a **uniform “convex up”/“convex down” configuration** such as that displayed in the bottom frame of Fig. 18. Figure 38 shows BIGBOSOR4 models of 90 degrees of a cylindrical shell with average radius, R_{CYL} = 318 inches and with five repeating sections, each with an arc length equal to WIDTH = 100 inches, that is, with MMM = 5 in SUBROUTINE BOSDEC2 of the file, **bosdec.span9.hugecyl**. The upper left-hand frame shows results for the uniformly corrugated panel optimized with the factor of safety for local buckling set equal to 2.0 (as was done for many of the other cases explored in this paper). The other three frames in Fig. 38 pertain to a new uniformly corrugated panel optimized with the factor of safety for local buckling set equal to 1.5 instead of 2.0. This was done so as not to penalize unduly the optimized weight of the uniformly corrugated cylindrical shell. The new optimized weight of this less conservatively designed uniformly corrugated panel is 564.5 lb

compared to 590.5 lb for the panel with the factor of safety for local buckling equal to 2.0. Even with the factor of safety for local buckling reduced from 2.0 to 1.5, the optimized weight, 564.5 lb, of the new uniformly corrugated panel is significantly greater than the optimized weight of the panel with non-uniform (complex) corrugations: 485.8 lb (Fig. 33). Figure 39 shows buckling modes corresponding to the lowest eight eigenvalues for modes with one axial half-wave over the axial length, $LENGTH = 100$ inches. The BIGBOSOR4 model subtends 180 degrees of the cylindrical shell with average radius equal to 318 inches, and there are 10 sections, each of arc length equal to $WIDTH = 100$ inches, that is, $MMM = 10$ in SUBROUTINE BOSDEC2 of the file, **bosdec.span9.hugecyl**. It would be interesting to determine with STAGS the imperfection sensitivity of this cylindrical shell with a series of imperfections with shapes similar to those displayed in Fig. 39.

Figure 40 shows the buckling mode corresponding to the 11th eigenvalue from the optimized “**fold98updown**” configuration for which other results appear in Fig. 11 and in Figs. 33 – 35. Only one “combined” short- and long-wavelength eigenvector could be found that is analogous to one of the modes displayed in Fig. 39. A search over the 50 eigenmodes corresponding to the lowest 50 eigenvalues, for which $n = 1$ axial half-wave over $LENGTH = 100$ inches, comes up with only the 11th mode that has this characteristic. This mode is analogous to the mode shown in the top left frame of Fig. 39. (In order to get convergence of the lowest 50 eigenvalues from BIGBOSOR4’s SUBROUTINE EBAND2, SUBROUTINE EBAND2 had to be modified to permit twice the number of inverse power iterations before giving up. The index called “IMAX” in SUBROUTINE EBAND2 was changed from 10 to 20. This modification of SUBROUTINE EBAND2 has been made permanent, that is, the BIGBOSOR4 computer program was permanently updated.)

Figures 41 and 42 show plots from BIGBOSOR4 and STAGS, respectively. The optimized “flat” panel of width equal to $3 \times WIDTH$ is mapped onto a cylindrical surface the average radius of which is $RCYL = 318$ inches, and the file, **bosdec.span9.hugecyl** [3], is used (no smoothing) to produce both the BIGBOSOR4 model shown in the top frame of Fig. 41 and the STAGS model shown in the top frame of Fig. 42. There is good agreement between the BIGBOSOR4 prediction of buckling shown in the bottom frame of Fig. 41 and the STAGS prediction of buckling shown in the bottom frame of Fig. 42.

Section 17. OPTIMIZED WEIGHTS OF UNIFORMLY CORRUGATED PANELS AND COMPLEXLY CORRUGATED PANELS VERSUS NUMBER OF MAJOR SEGMENTS IN $WIDTH/2=50$ INCHES

Figure 43 pertains to this section. The results described in this section were obtained with the use of the “**NEW**” boundary conditions (Section 4). The following comments apply:

1. The curve connecting the small solid round data points corresponds to optimized **uniformly corrugated** panels of the type shown in the bottom frame of Fig. 16 except with $PHIBIG = 0$ degrees instead of $PHIBEG = 16.08$ degrees.
2. The rest of the data points in Fig. 43 correspond to optimized **complexly corrugated** panels of the type shown in the top frame of Fig. 9 (no smoothing) or the type shown in the bottom frame in Fig. 29 (yes smoothing).
3. The main purpose of Fig. 43 is to demonstrate that the **complexly corrugated** panels weigh somewhat less than the **uniformly corrugated** panels.

4. The data points for weights of optimized **uniformly corrugated** panels probably represent global optimum designs because there exist only two decision variables: THICK(1) and PHISEG(1). THICK(i) and PHISEG(i), $i=2, 3, \dots, NSEG$ are linked to THICK(1) and PHISEG(1), respectively. Neither SUBWID(i), $i=1, 2, \dots, NSEG$ nor YPLATE(i), $i=1, 2, \dots, NSEG+1$, are decision variables. There are no sub-segments. The overall arching angle, PHIBIG, is not a decision variable and is fixed at 0.1 degree (essentially zero). The optimized uniformly corrugated panels are analogous to that shown in the bottom two frames of Fig. 18.

5. For the optimized **complexly corrugated** panels for which there are many more decision variables, it is likely, especially for the panels with eight or more major segments within the half-width, $WIDTH/2 = 50$ inches, that many of the data points do not represent global optimum designs **but are upper bounds for global optimum designs**. However, it is probable that the weights of the optimized panels are reasonably close to the weights of global optimum designs, perhaps within two to 10 per cent of those of global optimum designs. Perhaps an indication that the weight of an optimum design may be very close to the weight of the global optimum design is that one or more of the design margins corresponding to CYLBUK(1,i), $i=1, 2, \dots, NSEG$, are critical in addition to critical margins for local buckling (LOCBUK), symmetrical general buckling (BUKSYM) and anti-symmetrical general buckling (BUKASY).

6. The smallest weight of all the optimized **complexly corrugated** panels (93.15 lb for a 100-inch x 100-inch panel with $NSEG=7$: the “smoothed” version of the specific case called “fold97updown”) is about 13 per cent less than the smallest weight of all the optimized **uniformly corrugated** panels (106.5 lb for a 100 x 100 inch panel with $NSEG=10$). The use of various optimization strategies with the “corners” version of the specific case, fold97updown, leads to “global” optima with objectives in the approximate range $94 \text{ lb} < \text{WEIGHT} < 98 \text{ lb}$.

7. For the **uniformly corrugated** panels the factor of safety for both local and general buckling is 1.5. In many of the cases involving **complexly corrugated** panels discussed in this paper different factors of safety (f.s.) are used for local and general buckling: f.s. = 2.0 for local buckling and f.s. = 1.5 for general buckling. In Fig. 43 data points for optimized complexly corrugated panels with f.s. = 1.5 for local buckling are included in order to obtain a fair comparison of weights of optimized complexly corrugated panels and optimized uniformly corrugated panels.

8. For several values of $NSEG$ there are four data points corresponding to an optimized **complexly corrugated** panel:

- a. optimized panel with “corners” between major segments and with a local buckling factor of safety = 2.0
- b. optimized panel with “corners” between major segments and with a local buckling factor of safety = 1.5
- c. optimized panel with smoothing between major segments and with a local buckling factor of safety = 2.0
- d. optimized panel with smoothing between major segments and with a local buckling factor of safety = 1.5

9. Often the best way to obtain an optimized design under Condition (d.) is to use the optimized design obtained under Condition (b.) as a starting design, and execute “OPTIMIZE” several times in succession with the number of design iterations set to 15 and the “move limit” index, IMOVE, set to 3, which greatly restricts the amount each decision variable is permitted to “move” during a single design iteration. Similarly, the optimized design under Condition (c.) can usually be obtained with use of the optimized design obtained under Condition (a.) as a starting design. In addition, (a.) can be found starting from (b.), and (c.) can be found starting from (d.). A strategy for obtaining “global” optimum designs with some reliability is described in Appendix 5.

10. The weights of optimized designs of **complexly corrugated** panels are much less sensitive to the number of segments, NSEG, within WIDTH/2 than is so for the weights of optimized designs of **uniformly corrugated** panels.

Section 18. MORE ON OPTIMIZED WEIGHTS OF COMPLEXLY CORRUGATED PANELS VERSUS THE NUMBER OF MAJOR SEGMENTS SPANNING A WIDTH OF 50 INCHES

Figure 44 – 55 and Tables 4 - 6 pertain to this section. The results described in this section were obtained with the use of the “NEW” boundary conditions (Section 4). The following comments apply:

1. Panels of length, LENGTH = 100 inches that are narrower than WIDTH = 100 inches are first optimized. Then the widths of these optimized “narrow” panels are expanded by reflection of the optimized cross section as demonstrated for example in Fig. 10 and by repetition of the original plus reflected cross section as demonstrated for example in Fig. 11. The expanded widths are then mapped onto a cylindrical surface as described in Section 16 and as demonstrated in the top frame of Fig. 41 (repeated in the top frame of Fig. 44) and as demonstrated in the bottom frame of Fig. 44. For example, the bottom frame of Fig. 44 shows an optimized expanded panel cross section with the narrower width, WIDTH = 50 inches rather than WIDTH = 100 inches as shown in the top frame of Fig. 44. (The panel width, WIDTH, is specified in the input file for the GENOPT processor, BEGIN, as listed, for example, near the top of Table 2.) The total applied load is proportional to WIDTH. Hence, the total applied load, defined as TOTLOD in Table 2, would be –100000 lb corresponding to WIDTH = 50 inches rather than –200000 lb corresponding to WIDTH = 100 inches as listed in Table 2. Corresponding to the top frame in Fig. 44, the BIGBOSOR4 model used in the optimization phase of the analysis has eight major segments leading to the optimized cross section indicated on the left-hand portion of the top frame in Fig. 44. (This is the same optimized half-width cross section as that displayed in the top frame of Fig. 9.) Corresponding to the bottom frame in Fig. 44, the BIGBOSOR4 model used in the narrow-panel optimization phase of the analysis has four major segments over WIDTH/2 = 25 inches leading to the optimized cross section indicated as the left-most four segments in the middle part of the bottom frame in Fig. 44. These optimized four major segments are reflected to form an optimized cross-section of WIDTH = 50 inches (total of 8 major segments over WIDTH = 50 inches) and repeated to form an optimized cross section consisting of a total of 16 major segments spanning 2 x WIDTH = 100 inches. This 16-segment configuration is repeated three times and then cylindrically mapped to form the cylindrical panel shown in the bottom frame of Fig. 44. Therefore, the panel cross section shown in the bottom frame of Fig. 44 belongs to the same type of specific case as the panel cross section shown in the top frame of Fig. 44. Hence, it is appropriate to compare the optimized weight of the 16-segment section of (old WIDTH) = 100 inches (top frame in Fig. 44) with the optimized weight of the 16-segment section of 2 x (new WIDTH) = 100 inches (bottom frame in Fig. 44).

2. The 4-segment half-width model used for optimization of the panel cross section shown in the bottom frame of Fig. 44 has **about half the number of decision variables** as the 8-segment half-width model used for optimization of the panel cross section shown in the top frame of Fig. 44. Therefore, it is more likely that a “global” optimum design of this model with many fewer degrees of freedom will be found than is the case for the 8-segment half-width model shown in the top frame of Fig. 44. This speculation is borne out by the results displayed in Fig. 46, in which the optimized weights of the narrower panels are generally lower than those for the panels with WIDTH = 100 inches.

3. Smoothing, as described in Section 14, is introduced into the model with “corners” shown in the bottom frame of Fig. 44. The “smoothed” configuration is re-optimized. The re-optimized smoothed configuration is shown in the bottom frame of Fig. 45.

4. Figure 46 shows optimized weights obtained from optimizing narrower panels over a range of number of major segments spanning a panel half-width of 50 inches (solid data points in Fig. 46). The generally lower optimized weights obtained for the panels with fewer decision variables per width = 100 inches imply that the new optimization models are probably closer to “global” optimum designs than the optimized weights of the old optimization models (open circle data points in Fig. 46). However, note that the optimized panel cross sections corresponding to the new optimization models are different from those of the old optimization models. Therefore, the corresponding lower weights may simply indicate better optimum designs rather than optimum designs that are closer to a theoretical “global” optimum design.

5. Various strategies were used to obtain the optimum designs the weights of which are given by the solid circular data points in Fig. 46. To obtain many of these data points an optimum design of the panel cross section without any smoothing (bosdec.span9.hugecyl software) was first obtained by the execution of SUPERDUPEROPT with three consecutive executions of SUPEROPT and with use of the “611” strategy, followed by the execution of SUPERDUPEROPT with three consecutive executions of SUPEROPT and with use of the “852” strategy, followed by the execution of SUPEROPT with use of the “1532” strategy. With use of the optimized cross section with “corners” (no smoothing) as a starting design, one can then obtain the optimized **smoothed** cross section via the bosdec.span9.smoothing software and an execution of SUPERDUPEROPT with two consecutive executions of SUPEROPT and with use of either the “852” strategy or the “1532” strategy. Because of the long sequence of computer executions associated with the determination of each optimum design, one can appreciate that each optimum design, represented in Fig. 46 by a solid circular data point, requires significant computer time as well as calendar time.

6. Table 4 lists the optimized design of the specific case, **narw96updown**, with “corners” (discontinuous slopes in the circumferential direction) between adjacent segments. Table 5 lists the re-optimized design of the same specific case with the “corners” eliminated by the introduction of transitional and edge smoothing segments as described in the discussion in Section 14 associated with Figs. 27 – 32.

7. Figure 47 shows the optimized and extended and cylindrically mapped profile of the specific case, **narw96updown**, with “corners” (no smoothing). The first two buckling modes are not realistic for a complete (360-degree) cylindrical shell because buckling occurs in the neighborhoods of the right-hand and left-hand longitudinal edges. The third buckling mode and buckling load factor (eigenvalue) would also be critical for a 360-degree cylindrical shell.

8. Figure 48 shows a STAGS model of the specific case, **narw96updown**, with “corners” (no smoothing). There is very good agreement between the prediction of STAGS and that of BIGBOSOR4, as seen from the text associated with the lower frame in Fig. 48.

9. Table 5 and Figs. 49 – 53 pertain to the optimized **smoothed narw9xupdown** configurations, $x = 4, 6, 8, 10, 12, 14, 16$ (solid round data points in Fig. 46). The lowest weight plotted in Fig. 46 is 91.16 lb corresponding to “Number of major segments spanning 50 inches” equals 12. This smallest weight of the optimized complexly corrugated panels (91.16 lb) is about 14 per cent less than the smallest weight of the optimized uniformly corrugated panels (106.5 lb at “Number of major segments spanning 50 inches” = 10). The optimized weight of

the smoothed panel (91.16 lb) is slightly more than the optimized weight, 91.04 lb, of the analogous panel with “corners” (no smoothing, see Table 4). Figure 49 shows the optimized smoothed 6-segment cross section profile and critical symmetric buckling mode (top frame), the same smoothed optimized profile extended and mapped onto a cylindrical surface with average radius, $RCYL = 318$ inches (middle frame), and the critical buckling mode and load factor from BIGBOSOR4 of the extended and mapped smoothed model (bottom frame). Figure 50 shows the three anti-symmetric buckling modes and load factors computed in SUBROUTINE BEHX3, the minimum load factor of which is used to generate the design margin that contains the name, BUCASY (anti-symmetric buckling, Table 1). Figure 51 shows the STAGS model that is equivalent to the GENOPT/BIGBOSOR4 model featured in Fig. 49. The buckling load factor predicted by STAGS, 1.476, is very close to that predicted by GENOPT/BIGBOSOR4, 1.472. Figure 52 shows the model and buckling prediction obtained from GENOPT/HUGEBOSSOR4 for the optimized cross-section profile reflected, repeated and cylindrically mapped over 360 degrees. The bosdec.span9.smoothing.huge software and HUGEBOSSOR4 were used to generate the images in Fig. 52. Figure 53 shows the corresponding STAGS model and prediction of buckling. Again, there is good agreement between the predictions of STAGS and HUGEBOSSOR4.

10. The solid square data points in Fig. 46 correspond to panels of various widths, WIDTH, in which each model used for optimization has only a **single major segment per WIDTH/2** (specific cases all with the same name, **narw91updown**). The bosdec.span9.smoothing software and BIGBOSOR4 were used to generate the results. It is surprising that, in spite of the limitation of only a single major segment in the model being used for optimization (top part of Fig. 54, for example), which is associated with only three decision variables (bottom of Table 6, for example), the minimum weight of a panel of width 100 inches (92.43 lb), which occurs at 9 segments spanning a width of 50 inches, is only slightly higher than the minimum weight (91.16 lb at 12 segments spanning a width of 50 inches) for the narw9updown cases in which there are multiple major segments per WIDTH/2. A detail: The solid square data point corresponding to “Number of major segments spanning 50 inches” = 6 is significantly higher than the trend in that neighborhood. Multiple attempts were made to find an optimized design with a lower weight for this particular configuration, but the same result was obtained over and over. It is not known why this particular data point does not follow the trend exhibited by its neighbors. Figure 54 shows configurations and predictions for the optimized design with the lowest weight among those represented by the solid square data points in Fig. 46. Fig. 55 shows the corresponding model and predictions from STAGS. Again, there is very good agreement between the predictions of STAGS and GENOPT/BIGBOSOR4 that are listed for general and local buckling in the bottom two frames in Fig. 55.

Section 19. CONCLUSIONS

1. An optimized **complexly** corrugated cylindrical shell weighs somewhat less than an optimized “equivalent” externally T-stringer-stiffened cylindrical shell or an optimized cylindrical shell with a truss-core sandwich wall (Figs. 33 – 37) or an optimized cylindrical shell with **uniform** corrugations (Figs. 38, 39, 46).
2. Predictions from STAGS and from GENOPT/BIGBOSOR4 for optimized corrugated panels (Figs. 20 – 22, 42, 48, 51, 53 and 55) agree well enough to justify the use of GENOPT/BIGBOSOR4 for obtaining optimum designs, provided that GENOPT/BIGBOSOR4 can produce an optimized weight that is reasonably close to that corresponding to a “global” optimum weight.
3. Optimized panels with **uniform** alternating “convex surface up”/ “convex surface down” major cylindrical segments (fold98updown - Figs. 16 – 19, 43 and 46) weigh somewhat more than analogous optimized **complex**

corrugated panels with alternating “convex surface up”/ “convex surface down” major cylindrical segments the geometry of which varies from segment to segment (e.g. fold98updown - Figs. 9 – 12, Fig. 43, Fig. 46). By “geometry” is meant cylindrical segment radius, subtended angle ($2 \times \text{PHISEG}$), and orientation [$\text{YPLATE}(i+1) - \text{YPLATE}(i)$, $i = 1, 2, \dots, \text{NSEG}$].

4. In the two cases in which sub-segments are introduced (fold98supdown – Figs. 13, 14, 15 and fold94S – Fig. A6) it is determined that there is no advantage of having corrugated panels with sub-segments.

5. At least in two cases (fold98updown – Figs. 23a and 23b and fold916 – Fig. A10) some of the buckling margins for an optimized design are very sensitive to changes in one or more of the decision variables for values of those decision variables in the neighborhoods of the optimized values. It is likely that this characteristic makes it difficult to find “global” optimum designs, leading to a “zigzag” behavior of the objective from design iteration to iteration such as that exhibited for the specific case, fold98updown, in Figs. 25 and 26 and for the case, fold916, exhibited in Fig. A11(A).

6. GENOPT/BIGBOSOR4 has difficulty determining “global” optimum designs for corrugated panel configurations with many major cylindrical segments (Fig. 43 and Appendix 5). Optimization cycles from executions of SUPERDUPEROPT with multiple sequential executions of SUPEROPT often converge to designs of panels with weights that can be significantly higher than those of known “global” optimum designs (Figs. A8, A9, A12). It is quite possible that the “response surface method” described in [10 - 15] would lead to optimum designs that weigh less than those corresponding to the data points for complexly corrugated panels plotted in Fig. 43. The data points for complexly corrugated panels plotted in Fig. 43 represent upper bounds of actual global optimum designs.

7. For the corrugated panels with overall dimensions 100 x 100 inches, with all “convex surface up” cylindrical segments and with a relatively large number of major segments the optimized designs display merging of the major segments into fewer approximately cylindrical segments with larger radii. The optimized weights of these “convex surface up” corrugated panels are insensitive to the number of major segments for configurations with four to eight major segments over half the width of the panel (Fig. A5).

8. A modification of the boundary condition along the left-hand longitudinal edge of a panel from “u,v,w held; rotation free” to “symmetry/anti-symmetry” leads to somewhat heavier optimum designs (fold98updown – top frame of Fig. 9 compared to Fig. 6), but permits the post-optimization extension of the analysis by BIGBOSOR4 to wide panels and cylindrical shells with multiple repeating sections of width, WIDTH, (Figs. 10 – 12, Fig 33).

9. “Smoothing” away the “corners” between adjacent major segments and at the left-hand and right-hand longitudinal edges may have a significant influence on the minimum buckling load factor. This is so in the specific case called “fold98updown”, as demonstrated in Fig. 30.

10. One can obtain optimized panel cross-section profiles that can be fabricated by stamping, as demonstrated in Fig. 32.

11. A “HUGEBOBOR4” program has been created to handle up to 2950 shell segments, 10 times the maximum permitted with BIGBOSOR4. HUGEBOBOR4 was needed for analysis of the models shown in Figs. 52, A19 and A20.

12. Optimized panels of length, $LENGTH = 100$ inches, that are narrower than $WIDTH = 100$ inches weigh less **per width** than optimized panels of $WIDTH = 100$ inches perhaps because the optimization phase of the analysis of these narrower panels has significantly fewer decision variables and therefore approaches more closely a theoretical “global” optimum design (solid data points plotted in Fig. 46).

APPENDIX 1

ABOUT STAGS (Structural Analysis of General Shells)

In this paper optimum designs obtained by GENOPT/SPAN9 are evaluated later via STAGS models. **NOTE: STAGS is not used inside the optimization loop.**

STAGS [26 – 29] is a finite element code for the **general-purpose nonlinear analysis of stiffened shell structures of arbitrary shape and complexity**. Its capabilities include stress, stability, vibration, and transient analyses with both material and geometric nonlinearities permitted in all analysis types. STAGS includes enhancements, such as a higher order thick shell element, more advanced nonlinear solution strategies, and more comprehensive post-processing features such as a link with the STAGS postprocessor, STAPL.

Research and development of STAGS by Brogan, Almroth, Rankin, Stanley, Cabiness, Stehlin and others, formerly of the Computational Mechanics Department of the Lockheed Palo Alto Research Laboratory, was under continuous sponsorship from U.S. government agencies and internal Lockheed funding for more than 40 years. During this time particular emphasis was placed on improvement of the capability to solve difficult nonlinear problems such as the prediction of the behavior of axially compressed stiffened panels loaded far into their locally postbuckled states. STAGS has been extensively used worldwide for the evaluation of stiffened panels and shells loaded well into their locally postbuckled states.

A large rotation algorithm that is independent of the finite element library was incorporated into STAGS. With this algorithm there is no artificial stiffening due to large rotations. The finite elements in the STAGS library do not store energy under arbitrary rigid-body motion and the first and second variations of the strain energy are consistent. These properties lead to quadratic convergence during Newton iterations.

Solution control in nonlinear problems includes specification of load levels or use of the **advanced Riks-Crisfield path parameter** that enables traversal of limit points into the post-buckling regime. Two load systems with different histories (Load Sets A and B) can be defined and controlled separately during the solution process. Flexible restart procedures permit switching from one strategy to another during an analysis. This includes shifts from bifurcation buckling to nonlinear collapse analyses and back and shifts from static to transient and transient to static analyses with modified boundary conditions and loading. STAGS provides solutions to the generalized eigenvalue problem for **buckling** and **vibration from a linear or nonlinear stress state**.

Quadric surfaces can be modeled with minimal user input as individual substructures called "**shell units**" in which the analytic geometry is represented exactly. "Shell units" can be connected along edges or internal grid lines with partial or complete compatibility. In this way complex structures can be assembled from relatively simple units. Alternatively, a structure of arbitrary shape can be modeled with use of "element units".

Geometric imperfections can be generated automatically in a variety of ways, thereby permitting imperfection-sensitivity studies to be performed. For example, **imperfections can be generated by superposition of several buckling modes determined from previous STAGS analyses of a given case.**

A variety of material models is available, including both plasticity and creep. STAGS handles isotropic and anisotropic materials, including composites consisting of up to 60 layers of arbitrary orientation. Four plasticity models are available, including isotropic strain hardening, the White Besseling (mechanical sublayer model), kinematic strain hardening, and deformation theory.

Two independent load sets, each composed from simple parts that may be specified with minimal input, define a spatial variation of loading. Any number of point loads, prescribed displacements, line loads, surface tractions, thermal loads, and "live" pressure (hydrostatic pressure which remains normal to the shell surface throughout large deformations) can be combined to make a load set. For transient analysis the user may select from a menu of loading histories, or a general temporal variation may be specified in a user-written subroutine.

Boundary conditions (B.C.) may be imposed either by reference to certain standard conditions or by the use of single- and multi-point constraints. Simple support, symmetry, antisymmetry, clamped, or user-defined B.C. can be defined on a "shell unit" edge. Single-point constraints which allow individual freedoms to be free, fixed, or a prescribed non-zero value may be applied to grid lines and surfaces in "shell units" or "element units". A useful feature for buckling analysis allows these constraints to differ for the prestress and eigenvalue analyses. Lagrangian constraint equations containing up to 100 terms may be defined to impose multi-point constraints.

STAGS has a variety of finite elements suitable for the analysis of stiffened plates and shells. Simple four node quadrilateral plate elements with a cubic lateral displacement field (called "410" and "411" elements) are effective and efficient for the prediction of postbuckling thin shell response. A linear (410) or quadratic (411) membrane interpolation can be selected. For thicker shells in which transverse shear deformation is important, STAGS provides the Assumed Natural Strain (ANS) nine node element (called "480" element). A two node beam element compatible with the four node quadrilateral plate element is provided to simulate stiffeners and beam assemblies. Other finite elements included in STAGS are described in the STAGS literature [26 – 29].

APPENDIX 2

Simulation of the Behavior of s Uniformly Axially Compressed Cylindrical Shell

The results described in this section were obtained with use of the "OLD" boundary conditions (Section 4). A good check on the GENOPT/BIGBOSOR4 models used in this study is provided by a specific case called "fold91" in which there is only one major segment over the half-width, $WIDTH/2$, of the panel with overall dimensions, $LENGTH \times WIDTH = 100 \times 100$ inches. The one major segment represents one quarter of the circumference of a perfect cylindrical shell, as shown in Fig. A1. The predictions from GENOPT/BIGBOSOR4 agree with "classical" buckling theory for a uniformly axially compressed cylindrical shell [25], as is explained in the text included in Fig. A1.

APPENDIX 3

Numerical Results from GENOPT/BIGBOSOR4 for 100 x 100-Inch Panels with All “Convex up” Segments

The results described in this appendix were obtained with use of the “OLD” boundary conditions (Section 4). Figures A2, A3 and A4 show starting designs and optimized designs of panels in which all of the segments are “convex surface up”. The case, fold93, has three major segments over WIDTH/2; the case, fold94, has four major segments over WIDTH/2; the case, fold95, has five major segments over WIDTH/2; the case, fold96, has six major segments over WIDTH/2; the case, fold97, has seven major segments over WIDTH/2; the case, fold98, has eight major segments over WIDTH/2. Notice that in cases with higher than four major segments over WIDTH/2 the optimized configurations demonstrate a merging of some of the major segments into fewer segments with larger radii. Also, notice that the optimized panel weight is not sensitive to the number of major segments in the starting design for configurations with 4 to 8 major segments over WIDTH/2 (Fig. A5). Figures A3 – A5 demonstrate that, in terms of panel weight, it does no good to have more than four major “convex up” segments over WIDTH/2. Perhaps this is true only for the particular overall dimensions of the “corrugated” panels: LENGTH x WIDTH = 100 x 100 inches. Notice that the optimized designs for the configurations, fold95, fold96, fold97, fold98, are somewhat different from that for fold94 even though the optimized weights are almost the same as that for fold94. This result is often typical of optimizations: a given objective (or small variation of objective) may correspond to a variety of quite different designs. Fig. 24 provides another example of this property.

Figure A5 shows the total panel weight as a function of the number, NSEG, of major segments. We see here in graphical form the trend described in the previous paragraph.

Figures A2 – A5 correspond to “corrugated” panels with no sub-segments.

Figure A6 displays the starting design and optimum design of a panel, “fold94S” with four major segments over WIDTH/2 and sub-segments within each major segment, two “convex up” sub-segments in Major Segment No. 1 and four “convex up” sub-segments in each of Major Segments 2, 3 and 4. In this particular case in the optimized design all of the sub-segments have merged into the major segments, indicating that, in this case at least, there is no advantage of introducing sub-segments.

Figure A7 shows symmetric (top) and anti-symmetric (bottom) general buckling modes and load factors from the optimized configuration of the case called “fold97” (seven major segments over WIDTH/2, no sub-segments). The general buckling load factors predicted by BIGBOSOR4 are compared with those predicted by STAGS [26 – 29].

The designs displayed in Figs. A2 – A4 and A6 – A12 are somewhat impractical because all the cylindrical segments are “convex surface up”, producing sharp cusps between them that would be difficult or impossible to fabricate. Figures 2, 3, and 6 represent a design that is more practical because the transitions from segment to segment are significantly smoother, although the width-wise slope of the panel reference surface remains discontinuous at segment junctions.

Figure A8 shows the starting design (top) and two optimized designs (middle and bottom) of the prismatic corrugated panel, “fold98”. The purpose of this figure is to demonstrate that GENOPT often has difficulty in

determining the “global” optimum design. The second optimized design (bottom frame of Fig. A8) weighs about 20 per cent less than the first optimized design (middle frame of Fig. A8). This 20-per-cent difference is reduced to about 12 per cent by a changing of the lower and upper bounds of most of the decision variables as described in Item No. 3 in Section 6. Both designs are somewhat impractical because all the cylindrical segments are “convex surface up”, producing sharp cusps between them that would be difficult or impossible to fabricate.

Figures A9 – A11 show results from the specific case called “fold916” (16 “convex up” major segments over half the panel width, $WIDTH/2 = 50$ inches, as shown in the top frame of Fig. A9). The purpose of these figures is to demonstrate the difficulty that GENOPT/span9/BIGBOSOR4 has in finding a “global” optimum design for configurations with many major segments. The second and third frames in Fig. A9 show optimized panel cross section profiles that each evolve from three successive executions of SUPEROPT launched automatically during a single execution of the GENOPT processor called “SUPERDUPEROPT”. The profile shown in the second frame develops with use of what is called the “852” strategy, which is described in that frame. The profile shown in the third frame develops with use of what is called the “611” strategy. (In a manner analogous to the label, “852”, the label, “611” means “6 executions of OPTIMIZE for each execution of AUTOCHANGE; IMOVE = 1; IAUTO = 1”.) The “611” strategy produces a heavier optimum design (117.6 lb) than the “852” strategy (108.7 lb), mainly because the permitted “move limit” of each decision variable corresponding to IMOVE = 1 (10 per cent per design iteration) is twice that corresponding to IMOVE = 5 (5 per cent per design iteration).

The greater the permitted move limit, the more “zigzag” is the evolution of the objective during design iterations. This “zigzag” characteristic results from high sensitivity of one or more of the design margins (especially the local and general buckling margins in this study) to small changes in one or more of the decision variables in the neighborhoods of the optimum values of those decision variables. Figure 10 demonstrates that the buckling margins in the optimized “611” design are especially sensitive to small changes in the decision variables, YPLATE(9), YPLATE(10), YPLATE(11) and YPLATE(12). The larger the permitted decision variable move limit, the more likely it is that during design iterations the value of the objective will experience “oscillations” of high amplitude as the design “oscillates” from “FEASIBLE” to “NOT FEASIBLE” from design iteration to iteration.

Figures A11 (A,B,C) show three plots of objective versus design iterations during a single execution of SUPEROPT for the specific case, fold916. The “611” strategy has the largest permitted move limit (IMOVE=1 meaning 10 per cent per iteration) and exhibits the most “zigzag” appearance. The “852” strategy has half the permitted move limit (IMOVE=5 meaning 5 per cent per iteration) exhibits a somewhat less “zigzag” appearance. The “2032” strategy has the least permitted move limit (IMOVE=3 meaning 1 per cent per iteration) and is associated with the smoothest appearance. It is emphasized that the smaller the permitted decision variable move limits the more OPTIMIZEs per AUTOCHANGE are required for convergence to a FEASIBLE or ALMOST FEASIBLE optimum design.

Figure A11 (D) shows the three optimized panel cross section profiles corresponding respectively to the “611” strategy (optimized panel WEIGHT=117.6 lb), the “852” strategy (optimized panel WEIGHT=108.7 lb) and the “2032” strategy (optimized WEIGHT=99.22 lb).

The bottom frame in Fig. A9 shows the critical local buckling mode and load factor corresponding to the “611” optimized profile. Local buckling occurs first in a locally slightly flattened region formed by a very shallow

“valley” between two adjacent major “convex up” cylindrical segments. The shallow “valley” plays the role of an imperfection, and the buckling loads of uniformly axially compressed cylindrical sectors are especially sensitive to imperfections. The amplitude of this particular imperfection – a shallow valley between two adjacent major “convex up” cylindrical segments – is governed by the value of YPLATE(j), in which “j” represents the intersection of those two major segments. We see from Fig. A10 that the local buckling margin is especially sensitive to certain of the YPLATE decision variables. Doubtless this sensitivity is related to locally flattened regions between adjacent cylindrical segments, the amplitude of which depends strongly on YPLATE.

We know that there exist optimum designs such as those displayed in the bottom frame in Fig. A2 (91.41 lb), the second frame in Fig. A3 (91.71 lb), the bottom frame in Fig. A3 (92.33 lb), the second frame in Fig. A4 (92.62 lb), and the bottom frame in Fig. A4 (93.07 lb) because the 16 segments shown in the top frame of Fig. A9 could, during optimization cycles, rearrange themselves to form configurations close to those in the optimized panels with the significantly smaller weights displayed in Figs. A2, A3 and A4.

It is emphasized that Fig. A10 demonstrates why it is especially difficult to find a lighter-weight “global” optimum design in the specific case called “fold916” in which the optimization is based on the “611” strategy. In the neighborhoods of the optimized design existing after three consecutive executions of SUPEROPT, that is, the optimized design that weighs 117.6 lb (third frame in Fig. A9), there is extreme sensitivity of the design margins to changes in YPLATE(j), j = 9, 10, 11 and 12 in the immediate neighborhoods of the optimized values of YPLATE(j). That is because relatively small changes of YPLATE(j) give rise to variations in local “valleys” or flattened regions in the panel cross section, flattened regions that act like imperfections.

APPENDIX 4

Numerical Results from GENOPT/BIGBOSOR4 for 100-Inch Long x 200-Inch Wide Panels

The results described in this appendix were obtained with use of the “OLD” boundary conditions (Section 4). Figures A12 and A13 pertain to this section. In the two specific cases with the string, “wide”, as part of the case name, the overall dimensions of the “corrugated” panel are LENGTH x WIDTH = 100 x 200 inches, and the total axial load is -400000 lb. Extensive results from these two “wide” cases are contained in two files that in turn are included in the file, .../genopt/case/span/span9.tar.gz, of [3]. These two “wide” files are as follows:

```
-rw-r--r--  1 dave  staff   849920 May 24 14:53 wide98.tar
-rw-r--r--  1 dave  staff   112640 May 24 14:53 wide98updown.tar
```

Figure A12 shows the starting design (top) and two optimized designs (middle and bottom) of a prismatic corrugated panel that is twice as wide as that for which results are presented in the previous section. The purpose of this figure is to demonstrate that GENOPT often has difficulty in determining the “global” optimum design. The second optimized design weighs about 4 per cent less than the first optimized design. This 4-per-cent difference is reduced to about 1.3 per cent by a changing of the lower and upper bounds of most of the decision variables as described in Item No. 3 in Section 6. Both designs are somewhat impractical because all the cylindrical segments are “convex surface up”, producing sharp cusps between them that would be difficult or impossible to fabricate.

Figure A13 shows the starting design (top) and optimized cross-section (bottom) of a panel in which alternating segments are “convex surface up” and “convex surface down” (specific case name = wide98updown). This is a more practical design than that displayed in the previous figure, although it is significantly heavier (224.6 lb versus 192.2 lb).

The best way to treat wide panels is to use the “**NEW**” boundary conditions (Section 4) to predict the behavior of wide panels by first optimizing a much narrower half-width, then, in a post-optimization analysis phase, to extend that same relatively narrow (width=WIDTH/2) optimized panel cross-section profile by reflection as shown in Fig. 10 and repetition as shown in Figs. 11 and 33.

APPENDIX 5

Difficulty in finding a global “**FEASIBLE**” optimum design for the specific cases such as those called “fold914updown” and “fold913updown”

The results described in this appendix were obtained with use of the “**NEW**” boundary conditions (Section 4). The type of behavior described in this appendix is typical for the optimization of complex corrugated panels with more than about 10 major segments over half the panel width, WIDTH/2 = 50 inches.

Sometimes a good “global” optimum design can be obtained by first executing SUPERDUPEROPT with use of the so-called “852” strategy (described in Section 7 and in Fig. A15) followed by one or more executions of SUPEROPT with use of the so-called “1532” strategy (described below and in Fig. A16). The “852” strategy is associated with the use of 5 iterations per execution of OPTIMIZE, as recommended in Section 13. However, the “1532” strategy is associated with the use of 15 iterations per execution of OPTIMIZE, contrary to the recommendation in Section 13. Why do this? Use of the “1532” strategy is sometimes beneficial in a particular case when previous use of the “852” strategy or the “611” strategy on that same case has produced an optimum “**FEASIBLE**” design that weighs significantly more than an optimum “**ALMOST FEASIBLE**” design or an optimum “**MILDLY UNFEASIBLE**” design. (The design qualities, “**FEASIBLE**”, “**ALMOST FEASIBLE**” and “**MILDLY UNFEASIBLE**”, etc., are defined in Fig. A15.) This condition usually arises when “**FEASIBLE**” designs rarely occur during optimization cycles in a SUPEROPT execution. Rare occurrences of “**FEASIBLE**” designs during optimization cycles (which often happens during optimizations of complex corrugated panels with more than 7 major segments over WIDTH/2, that is, with large numbers of decision variables) may indicate that the objective of the optimized design is far from the “global” optimum objective.

Figures A15 – A18 pertain to this appendix. The specific cases, **fold914updown** and **fold913updown**, have “symmetry” edge conditions along both the left-hand and right-hand longitudinal edges (the “**NEW**” boundary conditions identified in Section 4) and, respectively, 14 and 13 alternating “convex up”/“convex down” major segments and no sub-segments over the panel half-width, WIDTH/2 = 50 inches. Fourteen and thirteen are large numbers of major segments. In the specific case, fold914updown, there are 72 decision variable candidates and 44 decision variables: THICK(1), SUBWID(i), i = 1, 2, ..., 14, PHISEG(i), i = 1, 2, ..., 14, YPLATE(i), i = 2, 3, ..., 15, and PHIBIG. The GENOPT/span9/BIGBOSOR4 software usually has a hard time finding a “global” optimum designs for complex corrugated panels of this type. One of the purposes of this appendix is to demonstrate how the End user might search for a global optimum design in such a case. Another is to include a specific case in which this strategy failed to find a global “**FEASIBLE**” design with an optimized objective (weight) that is reasonably close to the optimized objective of an “**ALMOST FEASIBLE**” design. (In GENOPT

a “FEASIBLE” design is a design for which all margins exceed -0.01 . An “ALMOST FEASIBLE” design is a design for which all margins exceed -0.05 .)

The recommended method applicable to the optimization of complex corrugated panels is first to run SUPEROPT or SUPERDUPEROPT with the use of the “852” strategy, that is, 8 executions of OPTIMIZE for each execution of AUTOCHANGE; the index for the maximum permitted “move” of each decision variable during each optimization cycle is set at $IMOVE = 5$ (“move” limit = 5.0 per cent of the current value of each decision variable); and the index for finding new “starting” designs via AUTOCHANGE, $IAUTOOF = 2$. (See the file, ...genopt/execute/URPROMPT.DAT [3], for an explanation of $IAUTOOF$).

Figures A15 – A17 pertain to the specific case called “**fold914updown**”. These three figures show results from a long optimization process in which the “852” strategy is used first, then changed to the “1532” strategy. (The “852” strategy is explained in Fig. A15, and the “1532” strategy is explained in Fig. A16.) This somewhat arcane procedure sometimes leads eventually to an optimum design that, in the opinion of the first author, must be close to the “global” optimum design because there are many occurrences of “ALMOST FEASIBLE” and “FEASIBLE” designs at the lowest WEIGHTs computed during the optimization cycles performed with use of the “1532” strategy.

Figure A18 pertains to the specific case called “**fold913updown**”. The same strategy is used as that described in the previous paragraph. However, in this case the recommended strategy does not work well. Unlike the results in the “fold914updown” case, in the “**fold913updown**” case “ALMOST FEASIBLE” data points seldom occur for panel WEIGHT less than 107 lb, and “FEASIBLE” data points occur only twice for WEIGHT less than 105 lb: the two data points with open circles close to the left-hand side of Fig. A18. This behavior signifies that the strategy will not produce a “global” optimum design.

Run stream details for the specific case called “fold913updown” (13 major segments over WIDTH/2)

Here is the **run stream** used to obtain the “difficult” data points in Fig. 43. (“difficult” means the data points corresponding to more than 8 major segments over the panel half-width, $WIDTH/2$), that is, data points corresponding to optimization cases with the highest number of decision variables.

[First, choose the “bosdec” routine corresponding to the BIGBOSOR4 model with “corners” between adjacent major segments, that is the BIGBOSOR4 model with **NO SMOOTHING**:]

cd .../bosdec/sources

cp bosdec.span9.hugecyl bosdec.src

cd .../genoptcase

genprograms [compile with use of the version of SUBROUTINE BOSDEC with **NO SMOOTHING**]

[Next, find the “global” optimum design with **NO SMOOTHING**:]

begin [establish the starting design, material properties, loading cases, allowables and factors of safety, etc.]

decide [choose decision variables and their bounds, linking relationships and inequality relationships]

mainsetup [use NUMBER OF ITERATIONS = 5; IMOVE = 5; IAUTO = 2; IDESGN=2 (IDESGN=2 means select “ALMOST FEASIBLE” as an acceptable design)]

superduperopt [use 8 OPTIMIZEs for each execution of AUTOCHANGE and 3 executions of SUPEROPT]

[At the end of the long SUPERDUPEROPT execution there will exist three “OPP” files: *.OPP.1, *.OPP.2 and *.OPP.3, in which “*” signifies the specific name of the case. Search each of these three “OPP” files for the best “FEASIBLE” design (the “FEASIBLE” design with the smallest objective), and write down the value of that objective. This you have to do for two reasons:

1. Only the best “ALMOST FEASIBLE” design and objective survive between successive executions of SUPEROPT during the execution of SUPERDUPEROPT because you selected IDESGN = 2 in the interactive MAINSETUP session. The best “FEASIBLE” designs are wiped out, except for that corresponding to the last execution of SUPEROPT, and the objective of that last execution of SUPEROPT may not be the best obtained during the three executions of SUPEROPT during the long SUPERDUPEROPT run.

2. The next execution of SUPERDUPEROPT will write over the three “OPP” files.]

In the specific case called “**fold913updown**” optimized with the “**852**” strategy the three *.OPP.i files, i = (1), (2), (3) yielded, respectively, the following three values of the objective (WEIGHT):

For “ALMOST FEASIBLE” designs: WEIGHT = (1) 103.9 lb, (2) 103.9 lb, (3) 103.9 lb

For “FEASIBLE” designs: WEIGHT = (1) 105.4 lb, (2) 105.4 lb, (3) 105.4 lb

(Often each of the three values of WEIGHT are different from each other.)

[The best “FEASIBLE” design determined from the first execution of SUPERDUPEROPT will probably not be a “global” optimum design with the designation, “FEASIBLE”. In order to find a “FEASIBLE” design with a smaller objective, one must execute SUPERDUPEROPT again with use of the “1532” strategy, as follows:]

mainsetup [use NUMBER OF ITERATIONS = 15; IMOVE = 3; IAUTO = 2; IDESGN=2]

superduperopt [use 15 OPTIMIZEs for each execution of AUTOCHANGE and 3 executions of SUPEROPT]

[As before, at the end of this second long SUPERDUPEROPT execution there will exist three “OPP” files:

*.OPP.1, *.OPP.2 and *.OPP.3, Search each of these three new “OPP” files for the best “FEASIBLE” design.

In the specific case called “**fold913updown**” optimized with the “**1532**” strategy the three *.OPP.i files, i = (1), (2), (3) yielded, respectively, the following three values of the objective (WEIGHT):

For “ALMOST FEASIBLE” designs: WEIGHT = (1) 98.5 lb, (2) 98.5 lb, (3) 98.5 lb

For “FEASIBLE” designs: WEIGHT = (1) **101.1** lb, (2) 106.1 lb, (3) 106.1 lb

In this case the “FEASIBLE” design from the *.OPP.1 file, having the lowest WEIGHT (WEIGHT = 101.1 lb) is the one of interest. It corresponds to the data point with the legend, “WEIGHT of optimized complex corrugated panel: “corners”, factor of safety for local buckling = 1.5” plotted at “Number of major segments” = 13 in Fig. 43.]

[Next, choose the “bosdec” routine corresponding to the BIGBOSOR4 model **WITH SMOOTHING** of the “corners” between adjacent major segments:]

cd ../bosdec/sources

cp bosdec.span9.smoothing bosdec.src

cd .../genoptcase

genprograms [compile with use of the version of SUBROUTINE BOSDEC **WITH SMOOTHING**]

[Next, find the “global” optimum design **WITH SMOOTHING**:]

mainsetup [use NUMBER OF ITERATIONS = 15; IMOVE = 3; IAUTO = 2; IDESGN=2]

superduperopt [use 15 OPTIMIZEs for each execution of AUTOCHANGE and 3 executions of SUPEROPT]

[As before, at the end of this third long SUPERDUPEROPT execution there will exist three “OPP” files: *.OPP.1, *.OPP.2 and *.OPP.3, Search each of these three new “OPP” files for the best “FEASIBLE” design. The result is analogous to that at the end of the first long SUPERDUPEROPT execution: The optimized “FEASIBLE” and “ALMOST FEASIBLE” weights from each of the three *.OPP.i files is the same:
For “ALMOST FEASIBLE” designs: WEIGHT = (1) 100.6 lb, (2) 100.6 lb, (3) 100.6 lb
For “FEASIBLE” designs: WEIGHT = (1) 112.0 lb, (2) 109.6 lb, (3) 109.6 lb

The design at the very first iteration of the first execution of SUPEROPT in this third execution of SUPERDUPEROPT is “MILDLY UNFEASIBLE”. That design is the same as the best “ALMOST FEASIBLE” design obtained at the end of the second execution of SUPERDUPEROPT: the design that weighs 98.5 lb and has “corners” between adjacent major segments. Smoothing these “corners” changes the general buckling load factors as follows:

“corners”: LOCBUK margin = +0.06955; GENSYM margin = -0.02981; GENASY margin = -0.04355

smoothed: LOCBUK margin = -0.02933; GENSYM margin = -0.05480; GENASY margin = -0.05653

Changing from the BIGBOSOR4 model with “corners” to that with added “smoothing” segments at the longitudinal edges and between adjacent major segments decreases the two general buckling margins, GENSYM and GENASY, enough to change the designation of design quality from “ALMOST FEASIBLE” to “MILDLY UNFEASIBLE”. (The most critical margin < -0.05 is the limit for an “ALMOST FEASIBLE” design).

APPENDIX 6

Generation of a “HUGEBOOR4” version of “BIGBOOR4”

The top two frames of Fig. 37a show two general buckling modes of an optimized axially compressed aluminum truss-core sandwich cylindrical shell “equivalent” to the complexly corrugated aluminum cylindrical shell shown in Figs. 33 and 34. The truss-core sandwich cylindrical shell is “equivalent” to the complexly corrugated cylindrical shell in that it has the same material, radius, length, and loading. The purpose of including an analysis of the truss-core sandwich case in this paper is to compare the optimized weights of the two “equivalent” cylindrical shells. (See Example 2 in Section 16.)

Fig. 37a, discussed in connection with Example 2 in Section 16, shows general and local buckling modes of the optimized truss-core sandwich shell obtained with BIGBOOR4. Only a relatively small arc of the cylindrical shell could be included in the BIGBOOR4 model for general buckling because BIGBOOR4 permits models with a maximum of 295 shell segments. A model spanning more of the circumference of the cylindrical shell

would provide a better estimate of the general buckling load factor and mode shape. Therefore, the first author was motivated to generate “**HUGEBOSOR4**”, a version of BIGBOSOR4 that **permits up to 2950 shell segments**. BIGBOSOR4 is retained and HUGEBOSOR4 is added to the “stable” of downloadable computer programs on the “Downloads” page of the shellbuckling.com website [3].

"HUGEBOSOR4" is activated when the user gives the command, "HUGEBOSOR4LOG". The consequently activated command set for "HUGEBOSOR4" is the same as that for "BIGBOSOR4", but the FORTRAN coding in HUGEBOSOR4 has been modified, of course. The modified coding is stored in the files:

```
.../genopt/case/sources/addbosor4.src.huge  
.../genopt/case/sources/b4plot.src.huge  
.../genopt/case/sources/b4util.src.huge  
.../genopt/case/sources/resetup.src.huge
```

In order to run GENOPT with the "HUGEBOSOR4" capability (GENOPT/HUGEBOSOR4) one must first do the following:

```
cp ../genopt/case/sources/addbosor4.src.huge ../bosdec/sources/addbosor4.src  
cp ../genopt/case/sources/b4plot.src.huge ../bosdec/sources/b4plot.src  
cp ../genopt/case/sources/b4util.src.huge ../bosdec/sources/b4util.src  
cp ../genopt/case/sources/resetup.src.huge ../bosdec/sources/resetup.src  
cp behavior.span9.huge ../genoptcase/behavior.new (behavior.span9.huge is stored as part of span9.tar.gz [3])  
cp bosdec.span9.hugecyl.huge ../bosdec/sources/bosdec.src (for "corners"; stored as part of span9.tar.gz [3])  
or  
cp bosdec.span9.smoothing.huge ../bosdec/sources/bosdec.src (for smoothed models; part of span9.tar.gz [3])
```

In order to get stand-alone "HUGEBOSOR4" results (e.g. plots) such as those shown in Figs. A19 and A20, one must go to a working directory where "HUGEBOSOR4" is to be executed and type the command, HUGEBOSOR4LOG, followed by the same consequently activated commands as those that are used in connection with BIGBOSOR4: bigbosorall, bosorplot, etc.

The main role of "HUGEBOSOR4" is to work with GENOPT. In that context valid input files for "HUGEBOSOR4" are generated via the new software, **bosdec.span9.hugecyl.huge** (no smoothing) and **bosdec.span9.smoothing.huge** (yes smoothing). These two FORTRAN files are included in the large compressed “tar” file, ...genopt/case/span/**span9.tar.gz** [3].

Figures A19 and A20 show results from two HUGHBOSOR4 models. Figure A19 shows a HUGEBOSOR4 model of an optimized aluminum truss-core sandwich cylindrical shell sector with 1434 little shell segments. (The individual little shell segments are not visible in Fig. A19 because they are too small to be seen there. See Fig. 37a for a somewhat enlarged view of the previously optimized truss-core sandwich wall construction).

Figure A20 shows a HUGE model of a complexly corrugated cylindrical shell: the specific case called “**fold913updown**” analogous to the much smaller BIGBOSOR4 models of the specific case called “**fold98updown**” displayed in Figs. 33 – 35 and 38 – 42.

ACKNOWLEDGMENT

The authors thank Dr. Frank Weiler for his STAGS analysis of the general buckling of a model of the optimized truss-core sandwich cylindrical shell analogous to the BIGBOSOR4 models shown in the top two frames of Fig. 37a. Mike Jacoby later computed from STAGS the general buckling mode shape and load factor shown in Fig. 37b. Frank Weiler helped Mike Jacoby get the latest version of STAGS “up” on Mike Jacoby’s computer.

REFERENCES

- [1] United States Patent no. 4,358,916, November 16, 1982, entitled “Novel Corrugated Metal Building Structural Unit”, invented by Maurice Lacasse, 1191 Cheman Industriel, Bernieres, Cte Levis, Quebec, Canada, G0S 1C0
- [2] Dean, W. R. The elastic stability of a corrugated plate. Proc. R. Soc., 1926, A111, 144, DOI: 10.1098/rspa.1926.0057
- [3] Bushnell, D. and Bushnell, W.D., Shell buckling website, <http://shellbuckling.com/>, in particular the “GENOPT” page of that website and the file, `.../genopt/case/span9.tar.gz`. In order to download all the files mentioned in this paper go to the “Downloads” page of the website, <http://shellbuckling.com/>, and download a very large compressed “tar” file (about 760 Mbytes) containing the latest versions of the following computer programs/documentation/cases created over many years by David Bushnell: BOSOR4, BIGBOSOR4, HUGEBOBOR4, BOSOR5, PANDA2 and GENOPT, and a small file, “How to Run on LINUX”. Generate the proper directory structure for the six programs; go to the directory, `.../genopt/case/span`; copy the file, `span9.tar.gz` to a working directory; go to that working directory; decompress (gunzip) the file, `span9.tar.gz`; disassemble the resulting file, `span9.tar` (`tar xvf span9.tar`); and uncompress and disassemble the two files, `general.info.tar.gz` and `specificcases.tar.gz`. All of the files referred to in this paper can be obtained by this procedure.
- [4] Bushnell, D., "GENOPT--A program that writes user-friendly optimization code", International Journal of Solids and Structures, Vol. 26, No. 9/10, pp. 2031-380, 1990. The same paper is contained in a bound volume of papers from the International Journal of Solids and Structures published in memory of the late Professor Charles D. Babcock, Jr, formerly with the California Institute of Technology.
- [5] Bushnell, D., "Automated optimum design of shells of revolution with application to ring-stiffened cylindrical shells with wavy walls", AIAA paper 2000-1663, 41st AIAA Structures Meeting, Atlanta, GA, April 2000. Also see Lockheed Martin report, same title, LMMS P534574, November 1999
- [6] Bushnell, D., "BOSOR4: Program for stress, stability, and vibration of complex, branched shells of revolution", in STRUCTURAL ANALYSIS SYSTEMS, Vol. 2, edited by A. Niku-Lari, pp. 25-54, (1986)
- [7] Bushnell, David, "Comparison of a "huge torus" model with a true prismatic model for: 1. an axially compressed simple monocoque cylindrical shell, 2. an axially compressed optimized truss-core sandwich cylindrical shell, and 3. an axially compressed optimized internally ring and stringer stiffened cylindrical shell with a T-stiffened weld land, Unpublished report for NASA Langley Research Center, February 12, 2010 and contained in the file, `...bigbosor4/case/prismatic/prismaticshell.pdf`

AIAA Paper 2014-xxxx, 55th AIAA Structures, Structural Dynamics and Materials Meeting, National Harbor, Maryland, January 13-17, 2014

[8] Vanderplaats, G. N., "ADS--a FORTRAN program for automated design synthesis, Version 2.01", Engineering Design Optimization, Inc, Santa Barbara, CA, January, 1987

[9] Vanderplaats, G. N. and Sugimoto, H., "A general-purpose optimization program for engineering design", Computers and Structures, Vol. 24, pp 13-21, 1986

[10] Haftka, R.T., and Watson, L.T., "Multidisciplinary Design Optimization with Quasiseparable Subsystems," Optimization and Engineering, 6, 9–20, March 2005. | View Publication

[11] Liu, B., Haftka, R.T., and Watson L.T., "Global-Local Structural Optimization Using Response Surfaces of Local Optimization Margins," Structural and Multidisciplinary Optimization, 27(5), 352-359, 2004.

[12] Papila, M., and Haftka, R.T., "Response Surface Approximations: Noise, Error Repair, and Modeling Errors," AIAA Journal 38(12), pp. 2336-2343, 2000.

[13] Hosder, S., Watson, L.T., Grossman, B., Mason, W.H., and Kim, H., "Polynomial Response Surface Approximations for Multidisciplinary Design Optimization of a High Speed Civil Transport," Optimization and Engineering, 2, 431-452, 2001.

[14] Venter, G., Haftka, R.T., and Starnes, J.H., Jr. "Construction of Response Surface Approximation for Design Optimization," AIAA Journal, 36(12), pp. 2242-2249, 1998.

[15] Roux, W.J., Stander, N., and Haftka, R.T., "Response Surface Approximations for Structural Optimization," International Journal for Numerical Methods in Engineering, 42, pp. 517-534, 1998.

[16] Bushnell, D., "Use of GENOPT and BIGBOSOR4 to obtain an optimum design of a deep submergence tank", unpublished report to the DOER company, Alameda, CA, June 30, 2009

[17] Bushnell, D. "Use of GENOPT and BIGBOSOR4 to obtain optimum designs of multi-walled inflatable spherical and cylindrical vacuum chambers", Unpublished report, February, 2011; Also, AIAA Paper 2012-1416 by Bushnell and Rankin, 53rd AIAA Structures, Structural Dynamics and Materials Meeting, Honolulu, Hawaii, April, 2012

[18] Bushnell, D., "SPHERE - Program for minimum weight design of isogrid-stiffened spherical shells under uniform external pressure", Lockheed Report F372046, January, 1990

[19] Bushnell, D., "Optimum design of imperfect isogrid-stiffened ellipsoidal shells...", written and placed in the file ..genopt/case/torisph/sdm50.report.pdf. Also see "A shortened version of the report on minimum weight design of imperfect isogrid-stiffened ellipsoidal shells under uniform external pressure", AIAA Paper 2009-2702, 50th AIAA Structures, Structural Dynamics and Materials Conference, 2009

[20] David Bushnell, Michael S. Jacoby and Charles C. Rankin, "Optimization of Propellant Tanks Supported by Optimized Laminated Composite Tubular Struts", AIAA Paper 2013-1479, 54th AIAA Structures, Structural Dynamics and Materials Meeting, Boston, Massachusetts, April 8-12, 2013.

AIAA Paper 2014-xxxx, 55th AIAA Structures, Structural Dynamics and Materials Meeting, National Harbor, Maryland, January 13-17, 2014

[21] David Bushnell, Michael S. Jacoby and Charles C. Rankin, "Optimization of Propellant Tanks Supported by One or Two Optimized Laminated Composite Skirts", AIAA Paper 2013-1480, 54th AIAA Structures, Structural Dynamics and Materials Meeting, Boston, Massachusetts, April 8-12, 2013.

[22] Bushnell, D., "Use of GENOPT and BIGBOSOR4 to obtain optimum designs of a double-walled inflatable cylindrical vacuum chamber", unpublished report, November, 2010; Also, AIAA Paper 2012-1416 by Bushnell and Rankin, 53rd AIAA Structures, Structural Dynamics and Materials Meeting, Honolulu, Hawaii, April, 2012

[23] Bushnell, D. and Thornburgh, R. P., "Use of GENOPT and BIGBOSOR4 to optimize weld lands in axially compressed stiffened cylindrical shells and evaluation of the optimized designs by STAGS", AIAA Paper 2010-2927, AIAA 51st Structures Meeting, Orlando, Florida, April 2010

[24] Bushnell, D. and Rankin, C., "Use of GENOPT and BIGBOSOR4 to obtain optimum designs of an axially compressed cylindrical shell with a composite truss-core sandwich wall", AIAA Paper 2011-1811, 52nd AIAA Structures Meeting, Denver, CO, April, 2011; See also Andrew E. Lovejoy and Marc R. Schultz, "Evaluation of analysis techniques for fluted-core sandwich cylinders", AIAA Paper 2012-1868, 53rd AIAA Structures, Structural Dynamics and Materials Conference, Honolulu, Hawaii, April 2012

[25] Brush, D.O. and Almroth, B. O., "Buckling of Bars, Plates and Shells", McGraw-Hill, 1975.

[26] B. O. Almroth, F. A. Brogan, "The STAGS Computer Code", NASA CR-2950, NASA Langley Research Center, Hampton, Va.(1978)

[27] C. C. Rankin, P. Stehlin and F. A. Brogan, "Enhancements to the STAGS computer code", NASA CR 4000, NASA Langley Research Center, Hampton, Va, November 1986

[28] Riks, E., Rankin C. C., Brogan F. A., "On the solution of mode jumping phenomena in thin walled shell structures", First ASCE/ASM/SES Mechanics Conference, Charlottesville, VA, June 6-9, 1993, in: Computer Methods in Applied Mechanics and Engineering, Vol.136, 1996.

[29] G. A. Thurston, F. A. Brogan and P. Stehlin, "Postbuckling analysis using a general purpose code", AIAA Journal, 24, (6) (1986) pp. 1013-1020.

[30] Ning, Xi, and Pellegrino, Sergio (2013). Imperfection-Insensitive Axially Loaded Cylindrical Shells. 54th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference, 8 - 11 April 2013, Boston, Massachusetts. Also see Joint Conference of Society of Engineering Science and ASME-AMD Summer Meeting, July 28-31, 2013

[31] Bushnell, D.: "PANDA2 - Program For Minimum Weight Design Of Stiffened, Composite, Locally Buckled Panels," Computers and Structures, Vol. 25, No. 4, pp. 469-605, 1987