

Table 34 A possible **run stream** for obtaining an optimum design and other information. This information appears in the file called \*.DEF, in which "\*" represents the generic case name, for example, "equivellipse", in this application of GENOPT. A list of the equivellipse.DEF file appears in Table a2 of the appendix. Note: **The preferred method to obtain optimum designs is to use SUPEROPT rather than multiple executions of OPTIMIZE**, as is listed in the first part of this table.

=====

A typical runstream is:

```
C      GENOPTLOG      (activate command set)
C      BEGIN          (provide starting design, loads, etc.)
C      DECIDE         (choose decision variables and bounds)
C      MAINSETUP      (choose print option and analysis type)
C      OPTIMIZE       (launch batch run for n design iterations)
C      OPTIMIZE       (launch batch run for n design iterations)
C      OPTIMIZE       (launch batch run for n design iterations)
C      OPTIMIZE       (launch batch run for n design iterations)
C      OPTIMIZE       (launch batch run for n design iterations)
C      CHANGE         (change some variables for new starting pt)
C      OPTIMIZE       (launch batch run for n design iterations)
C      OPTIMIZE       (launch batch run for n design iterations)
C      OPTIMIZE       (launch batch run for n design iterations)
C      OPTIMIZE       (launch batch run for n design iterations)
C      OPTIMIZE       (launch batch run for n design iterations)
C      CHOOSEPLOT     (choose which variables to plot)
C      DIPLOT         (plot variables v. iterations)
C      CHOOSEPLOT     (choose additional variables to plot)
C      DIPLOT         (plot more variables v design iterations)
C      CLEANSPEC      (delete extraneous files for specific case)
```

```
C  IMPORTANT:  YOU MUST ALWAYS GIVE THE COMMAND "OPTIMIZE"
C              SEVERAL TIMES IN SUCCESSION IN ORDER TO OBTAIN
C              CONVERGENCE! AN EXPLANATION OF WHY YOU MUST DO
C              THIS IS GIVEN ON P 580-582 OF THE PAPER "PANDA2,
C              PROGRAM FOR MINIMUM WEIGHT DESIGN OF STIFFENED,
C              COMPOSITE LOCALLY BUCKLED PANELS", Computers and
C              Structures, Vol. 25, No. 4, pp 469-605 (1987).
```

```
C  Due to introduction of a "global" optimizer, SUPEROPT,
C  described in the paper, Bushnell, D., "Recent enhancements to
C  PANDA2", AIAA paper 96-1337-CP, Proc. 37th AIAA SDM Meeting,
C  April 1996 pp. 126-182, in particular, pp. 127-130, you can
C  now use the runstream
```

```
C      BEGIN          (provide starting design, loads, etc.)
C      DECIDE         (choose decision variables and bounds)
```

```
C      MAINSETUP      (choose print option and analysis type)
C      SUPEROPT       (launch batch run for "global" optimization)
C      CHOOSEPLOT     (choose which variables to plot)
C      DIPLOT         (plot variables v. iterations)
```

C "Global" is in quotes because SUPEROPT does its best to find  
C a true global optimum design. The user is strongly urged to  
C execute SUPEROPT/CHOOSEPLOT several times in succession in  
C order to determine an optimum that is essentially just as  
C good as the theoretical true global optimum. Each execution  
C of the series,

```
C
C      SUPEROPT
C      CHOOSEPLOT
```

C does the following:

C 1. SUPEROPT executes many sets of the two processors,  
C OPTIMIZE and AUTOCHANGE (AUTOCHANGE gets a new random  
C "starting" design), in which each set does the following:

```
C      OPTIMIZE          (perform k design iterations)
C      OPTIMIZE          (perform k design iterations)
C      OPTIMIZE          (perform k design iterations)
C      OPTIMIZE          (perform k design iterations)
C      OPTIMIZE          (perform k design iterations)
C      AUTOCHANGE        (get new starting design randomly)
```

C SUPEROPT keeps repeating the above sequence until the  
C total number of design iterations reaches about 470.  
C The number of OPTIMIZEs per AUTOCHANGE is user-provided.

C 2. CHOOSEPLOT allows the user to plot stuff and resets the  
C total number of design iterations from SUPEROPT to zero.  
C After each execution of SUPEROPT the user MUST execute  
C CHOOSEPLOT: before the next execution of SUPEROPT the  
C total number of design iterations MUST be reset to zero.

=====