StressAlyzer

By Paul S. Steif
Department of Mechanical Engineering
Carnegie Mellon University

August 2002

Overview of Program

StressAlyzer is a set of computer modules running on Windows-based PCs, which help students develop expertise in analyzing very fundamental problems in Mechanics of Materials.

Each module is devoted to a single topic, such axial loading, shear force and bending moment diagrams, etc. Each module typically begins with very simple problems; subsequent problems become progressively more complex. The total number of problems ranges from 20 to 40, depending on the module. For most problems, randomly generated versions are offered. This allows a user to try the same type of problem multiple times.

There are brief reviews of the relevant theory, either at the beginning of a module or interspersed between problems. However, it is best to use StressAlyzer in conjunction with the other activities of a typical Mechanics of Materials course, including lectures, a textbook and so forth. StressAlyzer can be used, for example, as required homework assignments, as extra credit assignments, or as review before examinations.

Each module typically requires 2 to 5 hours to complete. By keeping a log file, the program tracks the user's successful completion of problems. The user can quit after completing any number of problems, and then call up the log file and resume work at a later time. The log file can be submitted for grading either electronically as an email attachment, or by handing in a diskette.

Content of Modules

StressAlyzer consists of 6 modules:

Axial.ctb: axial loading

Twist.ctb: torsional loading

Bend.ctb: shear force and bending moment diagrams

Deflect.ctb: beam deflections using method of superposition

Load3D.ctb: calculating stresses in tension, torsion, and bending, but in a 3-D context

Trans.ctb: stress transformations

For the User of StressAlyzer

The screen area of the monitor should be set at 800 x 600 pixels or higher.

To begin any module, double click on ctwinx.exe and then open one of the modules listed above (ending in .ctb). Maximize the window containing the module you have opened.

StressAlyzer keeps track of your progress by maintaining a log file. The log file associated with the module axial.ctb must be called axial.log, and so forth for the other modules.

Shortly after opening a module, you will be offered two choices: (1) to start a new log file, or (2) to connect to an existing log file. The first time using a module, you must start a new log file. When starting a new log file, you will first be asked for your name (in the form, for example, Smith_Mary). This will not be the name of the log file, although this name will be encoded in the log file. After entering your name, a standard Windows dialog box will ask you to name the file. You must use the requested file name (e.g., axial.log, or as appropriate to the module), but you can put the file in any directory. When you quit the module, your progress thus far is recorded in the log file. When you want to resume work on the module, choose to connect to an existing log file. With a standard dialog box, you can browse and then open the appropriate log file. You will then be asked to enter your name, which will be compared with the name originally encoded in the file. Of course, you can always choose to start a new log file, although there is no advantage to doing so.

If more than one person is using StressAlyzer on the same computer, each person should be sure to set up a log file in a separate folder. Otherwise, StressAlyzer will simply overwrite the old log file with the new log file.

StressAlyzer also creates a backup log file named, for example, axial.bkp. If the log file is destroyed or lost, then, if the back up file is still present, you may be able to retrieve your work by renaming the backup file. For example, axial.bkp would be renamed axial.log. Then, you restart StressAlyzer, open the module axial.ctb and connect to this axial.log file.

You may do problems in any order, and you may choose to do any subset of the problems. However, the problems have been devised and ordered to enable you to build your understanding gradually from simpler ideas to more complex ones. Thus, users who are new to mechanics of materials would probably be best served by going through the problems in order. Of course, you may have been directed by an instructor to solve a particular set of problems.

If you get a problem correct on the first try, StressAlyzer will move you on to the next type of problem. Otherwise, when a problem is eventually solved, StressAlyzer will give you a new variant of the same type of problem. To jump to a new type of problem, use the Navigation Menu. There, you can choose to jump to the Next (Type of) Problem, to the Previous (Type of) Problem, or to a problem number of your choice. In addition, you can choose to View Progress. This will display all problems in the module. Those that are uncompleted are shown in red; those that are completed are shown in black. You may click on any problem to receive a problem of that type.

Additional Comments on Using StressAlyzer

Many problems require the user to enter a numerical answer. Generally, the user can type any characters or numbers. Expressions that contain only numbers (e.g., 125.4), or scientific notation (e.g., 1.3E6), or mathematically valid combinations of numbers and the arithmetic symbols +,-,*,/,^, are evaluated according to normal mathematical rules. Expressions that include invalid combinations of symbols, or any letters besides E or e, are evaluated to zero. When such an expression is entered, StressAlyzer will proceed as if the user entered 0.

It is advisable not to use the Delete key. It can cause text to disappear temporarily. (The backspace key functions as expected.)

In the module bend.ctb, the user must step through a Concise Derivation of the relevant Equations. In addition, the user is given the option of going through a Full Derivation of the equations. Unlike all other problems in StressAlyzer, the user receives no recorded credit for going through the Full Derivations. In addition, the user may be unable to answer a question in the derivation and, thus, be unable to proceed with the derivation. If this happens, the user can always revert to the Concise Derivations or move on to do any of the other problems.

For the Grader of StressAlyzer Log Files

When one or more users have submitted log files, all the log files for a single module may be decoded at once.

Log files are text files. Here is a typical log file, named axial.log, which was generated by the axial module.

1111
top
are
low
may
now
now
tie
low
say
now
try
she
tie
ten
are
the
use
ten
ten
tip
sip
cat
sun
tie
try
say
top
say
not
not
try
was
the
now
two
two
she

two
was
sun
vdntonfjafzjncstfurlhdxlbw
ylcixkkxkyybrhpcsaqyucmxkm
txvyfxlutcfmctszsafbwggtmj
loqfvjojvscjidnpsrpixxielc
rawjggxvipemfgpbjualkbjxrb
uhstfsjuhizdbjzosbfiyexxiy
zdjsupsywhrnqvyxziqovqnrrt
mvtqhfoeqaosdycmsnwciwyvst
qijaxzhreztezbguqavqcsjtuo
xqdahlcyugnqaybrgfecrcrtps

If the log file does not look something like the above, it has probably been corrupted. (Depending on the module, there may not be long strings of characters at the end.)

To decode log files, set up a new folder named Log files in the same folder where ctwinx.exe and axial.ctb, etc. are located.

You will only decode log files from one type of module at a time, although you can store log files from all modules in the same folder named Log files.

Say that you want to decode log files produced by the axial.ctb. Say that one student provides you with a log file; they will probably send it to you as axial.log. Copy that file into the Log files folder and rename it axial1.log. Copy the second student's log file into the Log files folder and rename it axial2.log, and so forth. Of course it makes no difference which student you call 1, 2 and so forth, only that the log files are named successively as axial1.log, axial2.log, etc.

Start the program ctwinx.exe and open up the program decode.ctb. Choose the module which produced the log files you want to decode. In our example here, choose Axial.

Assuming you have put all the files axial1.log, axial2.log, etc into the Log files folder as described above, you may click on the button "Decode Log Files". A dialog box will open which will tell you to "Open up a file to place possible errors from log files." You should click OK and then a standard Windows dialog will then open, allowing you to browse through the file system hierarchy to set up the errors file. You may put the errors in the Log files folder and name it, say, errorsaxial. After doing so, another dialog box will open which will tell you to "Go to the folder Log Files, and open the file axial1.log". You should click OK. A standard Windows dialog will then open, allowing you to locate the folder Log files. You will open the Log files folder and click on axial1.log and then click on Open.

The program will decode the log files. Each time it reads a file, the message "Read in file", followed by the file name, will appear, for example, axial1.log. After all files are read (which could be very fast if there are few files), the message "No more files, or there is too large a gap between file numbers" will appear. You will click on OK. Then, the program will ask you to open

a file to contain the student scores. Click OK and, like before, a standard Windows dialog will then open, this time asking you to define a new file (calling it, say, scoresaxial).

You will be brought back to the very first screen of the decoder program. You may decode another set of log files (say twist.log) if you have them, or you can quit the program.

To look at the scores, use Microsoft Word to open the file you saved which contained the student scores (referred to above as scoresaxial). (Although the file is a text file, the lines are rather long, and may not be properly read by accessories such as Notepad or WordPad; other word processing programs may open it successfully.) The entries on each line are separated by tabs, so you can select the entire contents of the file, and copy and paste them into Microsoft Excel (or other spreadsheet package). (Since the file containing scores consists purely of text separated by tabs, it could also be opened directly in Excel; however, you will be prompted to make some choices by the Text Import Wizard.) Once in Excel, you will find a table. The first column contains the names of the students associated with the log files. Each row contains the scores for the student listed in the first column, and each column describes a different problem. If the cell has a 1 (one), that problem was completed by the student associated with that line. If the cell has a 0 (zero), the problem was not completed.