Using Crack Propagation Software with da/dN Lines from ASME Code

In the previous chapter “Using Crack Propagation Software” the da/dN vs ΔK transform was described by a single line. In this example we will be using a new version of this Open Source software that uses lines computed from the ASME code. Specifically a set of da/dN lines each for a different R ratio. As before the stress intensity factor computations follow the recommendations of British Standard BS7910 closely, and the LIFO or Push-Down list material memory model is used to follow the effects of load input history.

The new set of programs are available in a “tar” file at:
https://github.com/pdprop/pdprop2

The Open Source type license that comes with the programs allows you to make software alterations to include any other features that you may wish to add. As usual there is of course **no warranty of any kind**. Please read the license at the top of the program listings and follow the web links to the full license terms.

The steps involved in computing damage or the propagation of a fatigue crack have been outlined in Chapter 8 of this tutorial. The present chapter section will demonstrate the use of the software to make predictions by means of an example.

F.A.Conle
Adjunct Prof
U.Waterloo, 2021

For an overview of the background of this example it would be helpful to look over the following references:

[1] L.Tucker, S.Bussa, “The SAE cumulative fatigue damage test program” SAE Congress SAE paper 750038, Feb. 1975, pp. 198–248.

[2] FD+E SAE Keyhole Specimen Test Load Histories and Crack Initiation and Propagation Results http://fde.uwaterloo.ca/Fde/Loads/Keyhole/keyhole.html

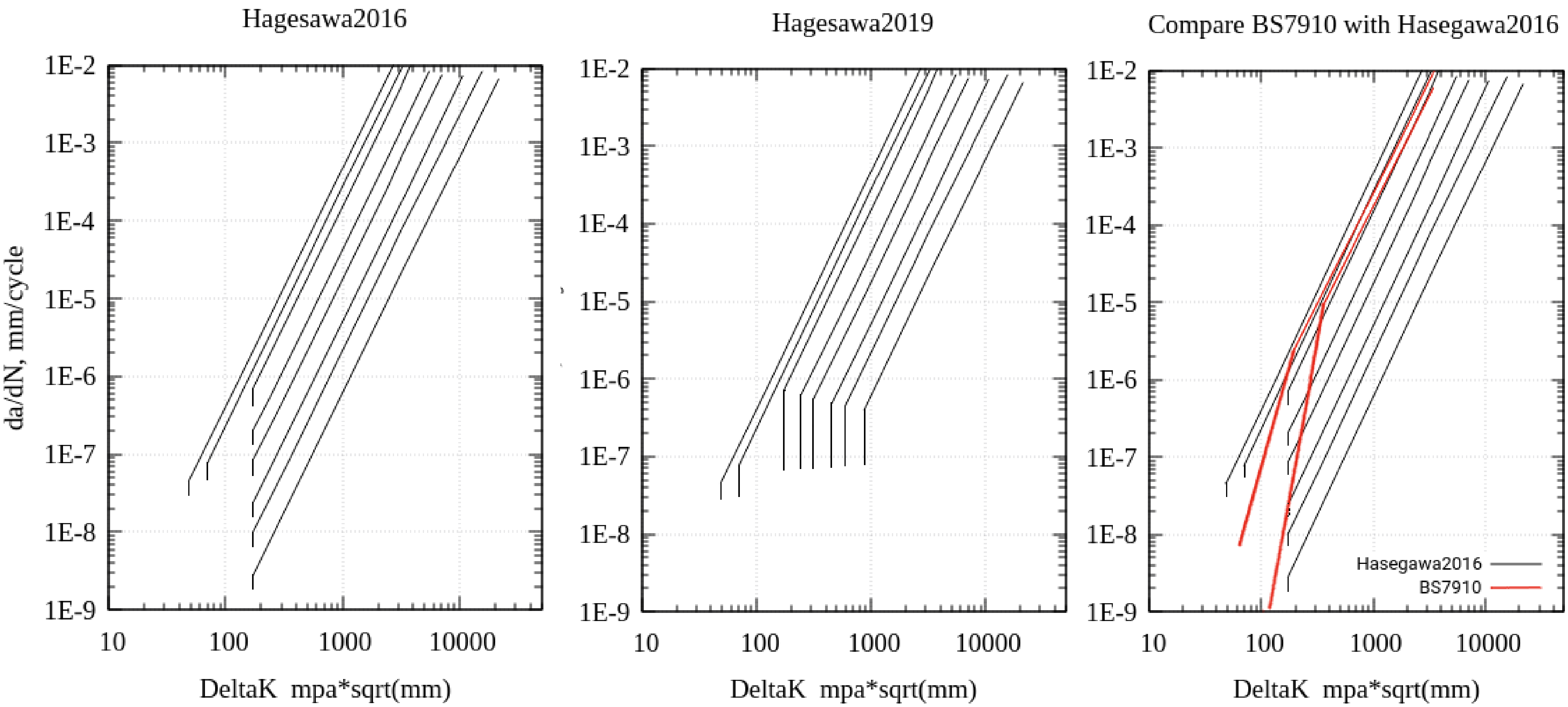
[3] K.Hasegawa, M.Vratislav, Y.Yamaguchi, Y.Li, “Fatigue Crack Growth Rates for Ferritic Steels Under Negative R Ratio”. ASME 2016 Pressure Vessels and Piping Conference, 50350

[4] F.A.Conle, "Crack Initiation and Propagation Predictions for ManTen and RQC-100 Steel Keyhole Notched Specimens Tested by the Fatigue Design & Evaluation Committee of SAE”, SAE paper 2020-01-0191

Material: “ManTen” Steel

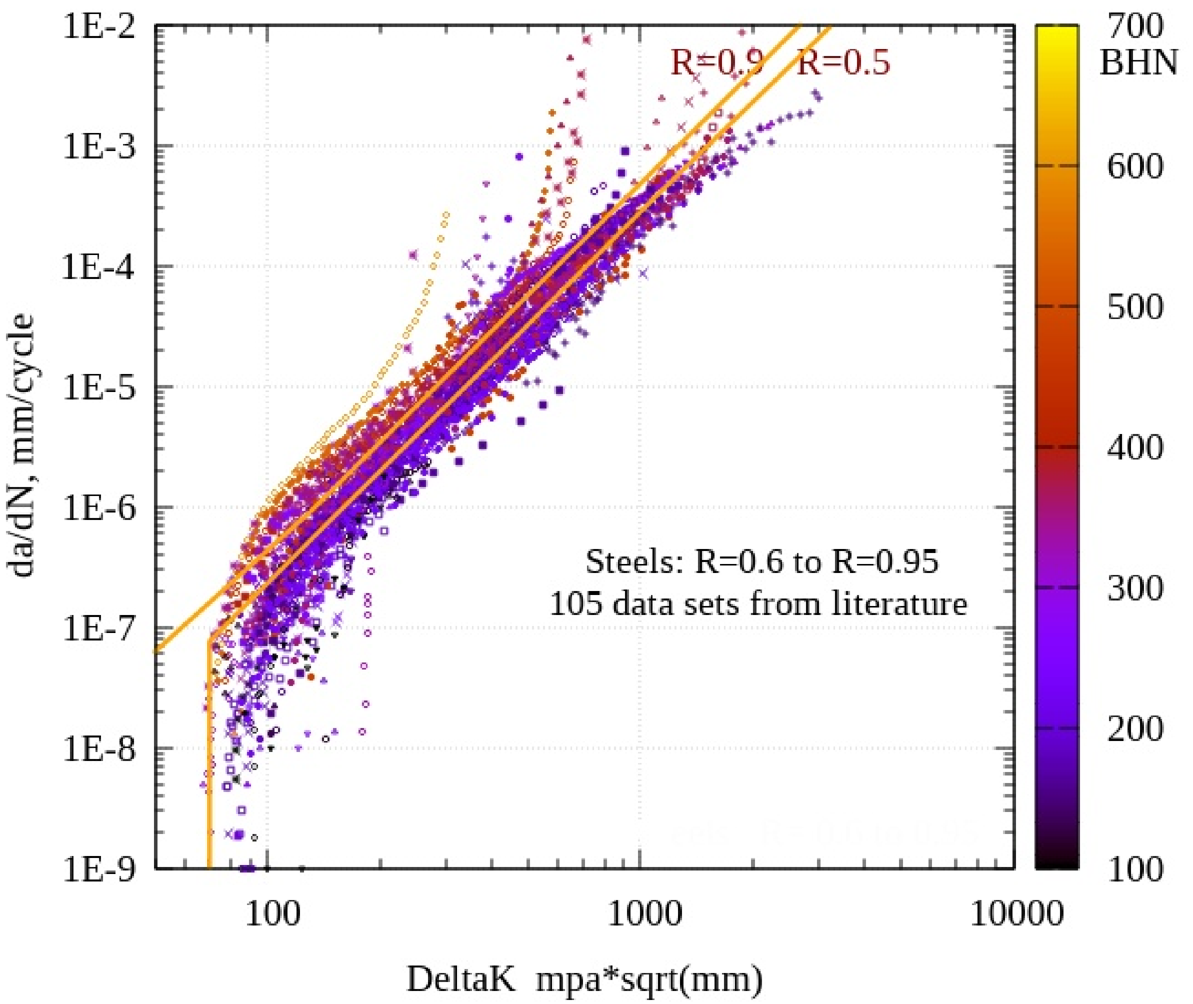
Previous versions of the crack propagation model used a single da/dN curve.

Significant Change: A multi-R ratio da/dN vs ΔK Model



These curves are set by specifying their name in the environment file pdprop.env

The da/dN lines, derived by Hasegawa et al, match observations of constant amplitude data in the literature.



A program user can also specify their own single or set of lines by setting the flag

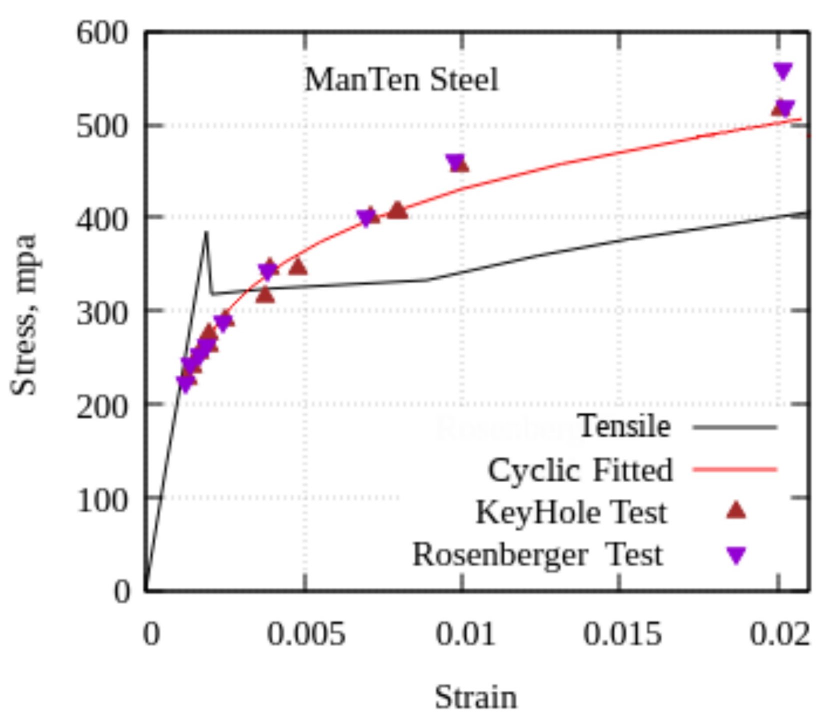
#DADN= USER

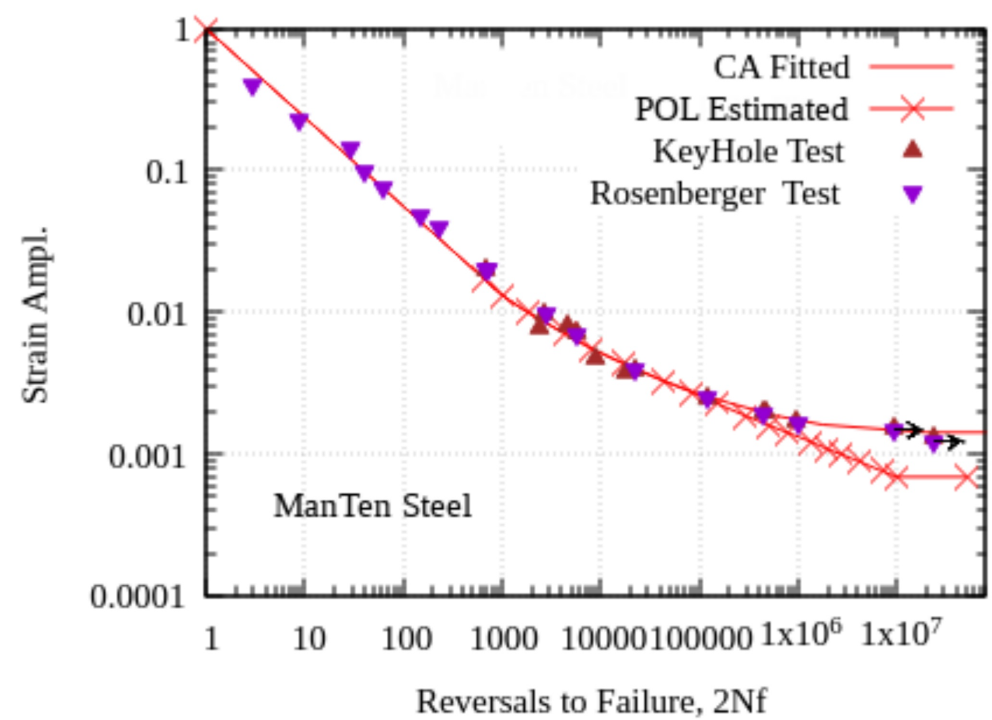
in the pdprop.env file and supplying the da/dN digital data for each line in seperate files.

The simulation will also compute the crack initiation life if a “fitted” axial specimen fatigue file is available.   e.g.:
https://fde.uwaterloo.ca/Fde/Materials/Steel/Sae1300/keyholeManTen-str-2Nf_fitted.html
or for Variable Amplitude(VA) test simulations an approximated fitted file:
https://fde.uwaterloo.ca/Fde/Materials/Steel/Sae1300/keyholeManTen-str-2NfPOL_fitted.html

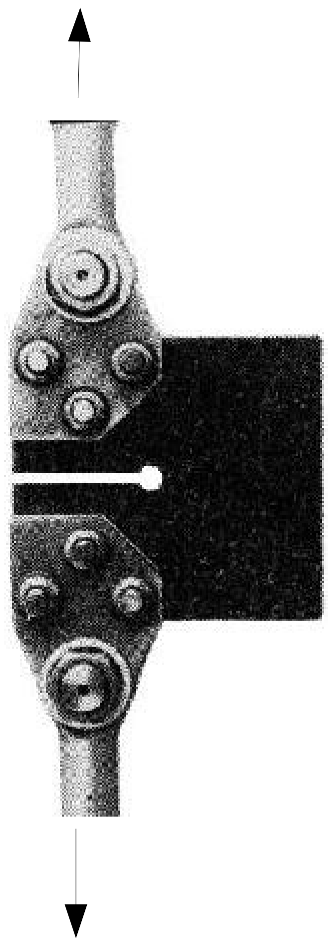
Most of the axial specimen test information appears to have originated in Ref.:
P.C.Rosenberger, "Fatigue Behavior of Smooth and Notched Specimens
# of Man-Ten Steel," MSc. Thesis, T.A.M. Dept., U.of Illinios, 1968
-as can be seen in the two figures below.

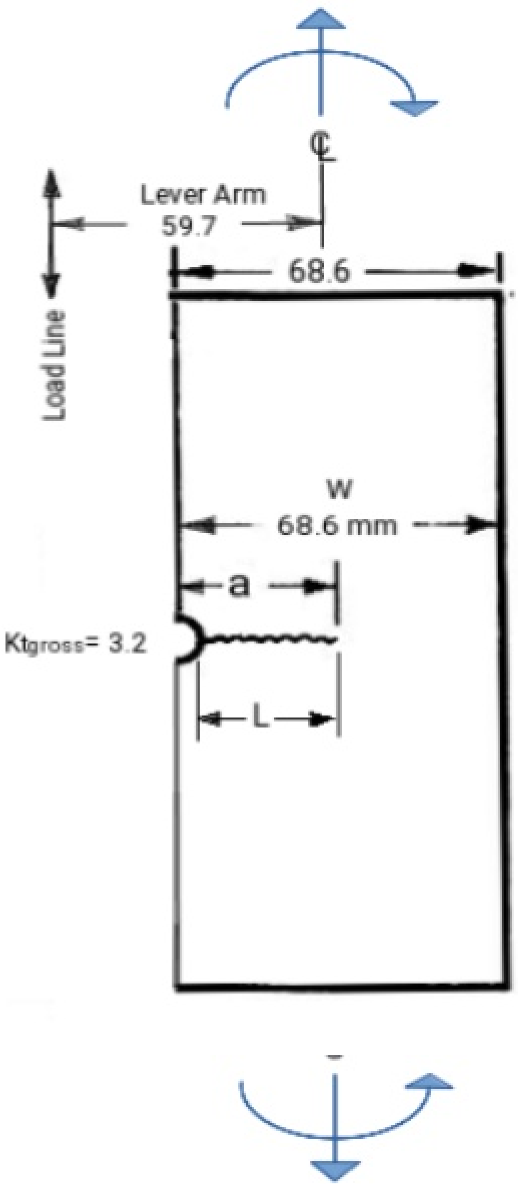
The tensile test data file, used for FAD diagrams, is in the text file at:
https://fde.uwaterloo.ca/Fde/Materials/Steel/Sae1300/keyhManTenTens.txt



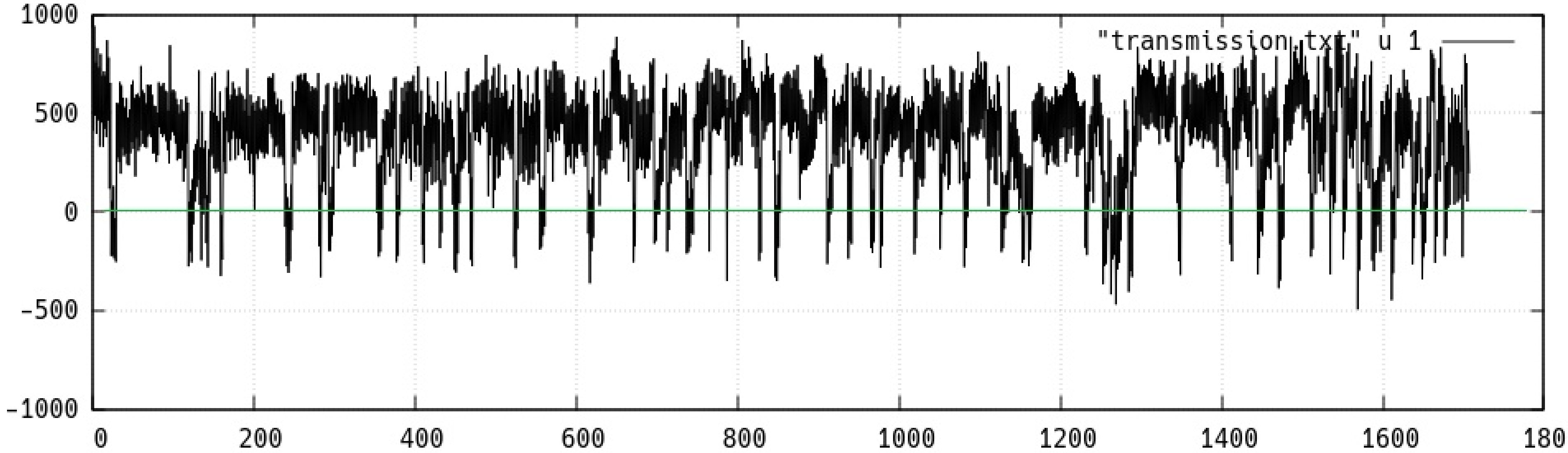


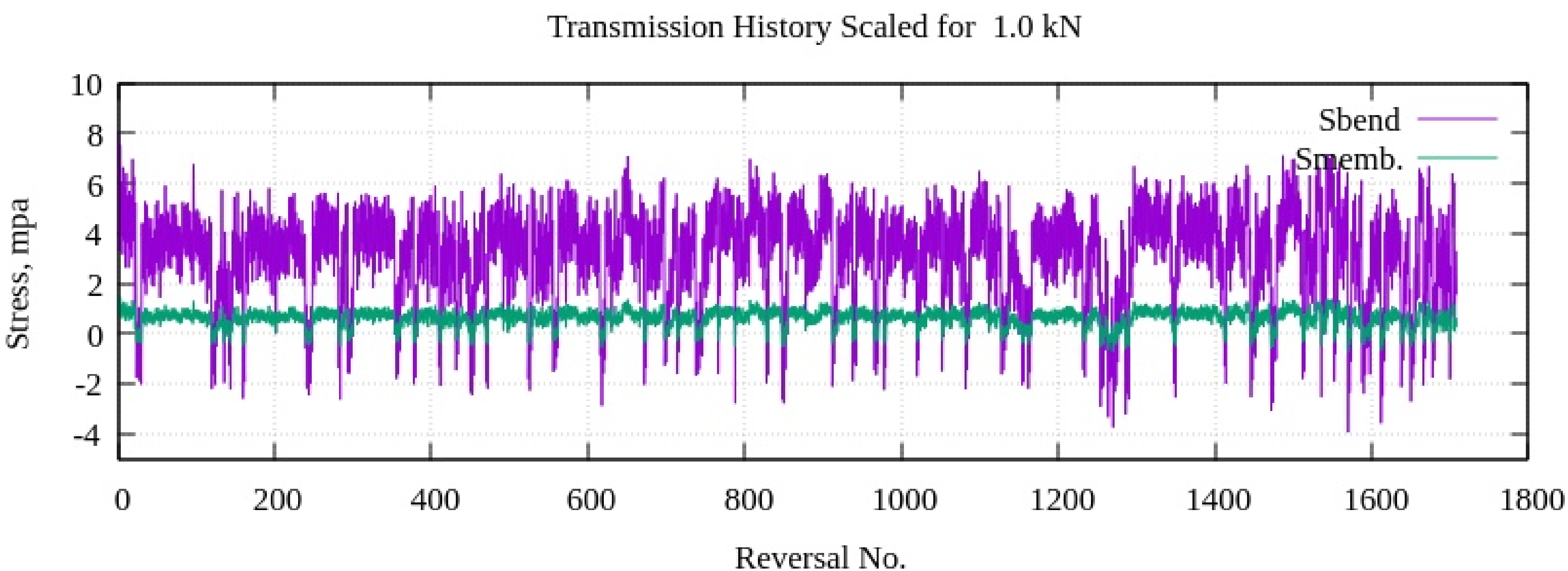
Geometry: The “Keyhole” specimen can be approximated as a plate with axial and in-plane bending loads





Loads: In this example we will be using the Transmission history described in Ref.[2]





In our example calculations we will use the FDE Transmission history file.
The beginning of this file looks like this:

```
#Keyhole 1000 N history. filename: transm1000N.txt
#
#no.  Sm    Sb
  1  1.530  7.990
  2  0.629  3.287
  3  1.443  7.534
  4  0.744  3.887
  5  1.159  6.055
  6  0.603  3.151
  7  1.271  6.639
  8  0.711  3.711
  9  1.121  5.855
 10  0.505  2.639
 11  1.225  6.399
 12  0.692  3.615
 13  1.055  5.511
 14  0.504  2.631
etc.
```

This file will be specified in the command line as “standard input”
General format: ./plateEdgeFlawIPB scaleFactor <loadHistory >outputFile
e.g.:

```
./plateEdgeFlawIPB  15.6   < transm1000N.txt    > TM3-15.6kNout
```

(We will be going thru all the steps in more detail below)

Steps in Detail

1. Make a folder to contain the project. Into that folder download the "tar" file at:
   (a)go to : https://github.com/pdprop/pdprop2/blob/master/cleanPdprop2.tar.gz
   (b) click on the "Download" button

2. Extract the files in the compressed tar file:

   tar xvf cleanPdprop2.tar.gz

   (Reminder: these Open Source come with **no warranty of any kind**. Please read the license at the top of the program listings and follow the web links to the full license terms.)

   Check of files: ls -1

   should show:

   CleanPdprop2

   cleanPdprop2.tar.gz

3. CleanPdprop2 is the folder of all the programs without any compiled (executable) programs, thus we need to enter the folder and compile

   cd CleanPdprop2
   ls

   shows:

```
Test/CleanPdprop2> ls
0_README             decimate.f    makeRepSaefcalc2  PlateEdgeFlawIPB   replaceline.f
Allcompile           delete1arg.f  pdrain.f          PlateLongSurfFlaw  RodSurfFlaw
convert2MPa_mm.f     getFADs.f     PipeInSurfFlaw    PlateSurfFlaw      saefcalc2
convertParis2table.f hilo2.f       PlateEdgeFlaw     PlateThruFlaw      saefcalc2.f
```

The green items are executables. The blue items are Folders and the black, in this case, are fortran programs that need to be compiled.

Setup cont'd

4. We can now compile all the programs and make scripts executable by running the script Allcompile (you can check and edit this script):

   ./Allcompile

   The script will show each file operation, so scroll back its output to look for errors.

5. If all things have been compiled the user can now get rid of the crack propagation types that are not needed.
   In this example only the folder PlateEdgeFlawIPB will be needed so we can remove the other folders that will not be used to run simulations. In this case we can:

   rm -r PipeInSurfFlaw PlateEdgeFlaw PlateLongSurfFlaw
   rm -r PlateSurfFlaw PlateThruFlaw RodSurfFlaw

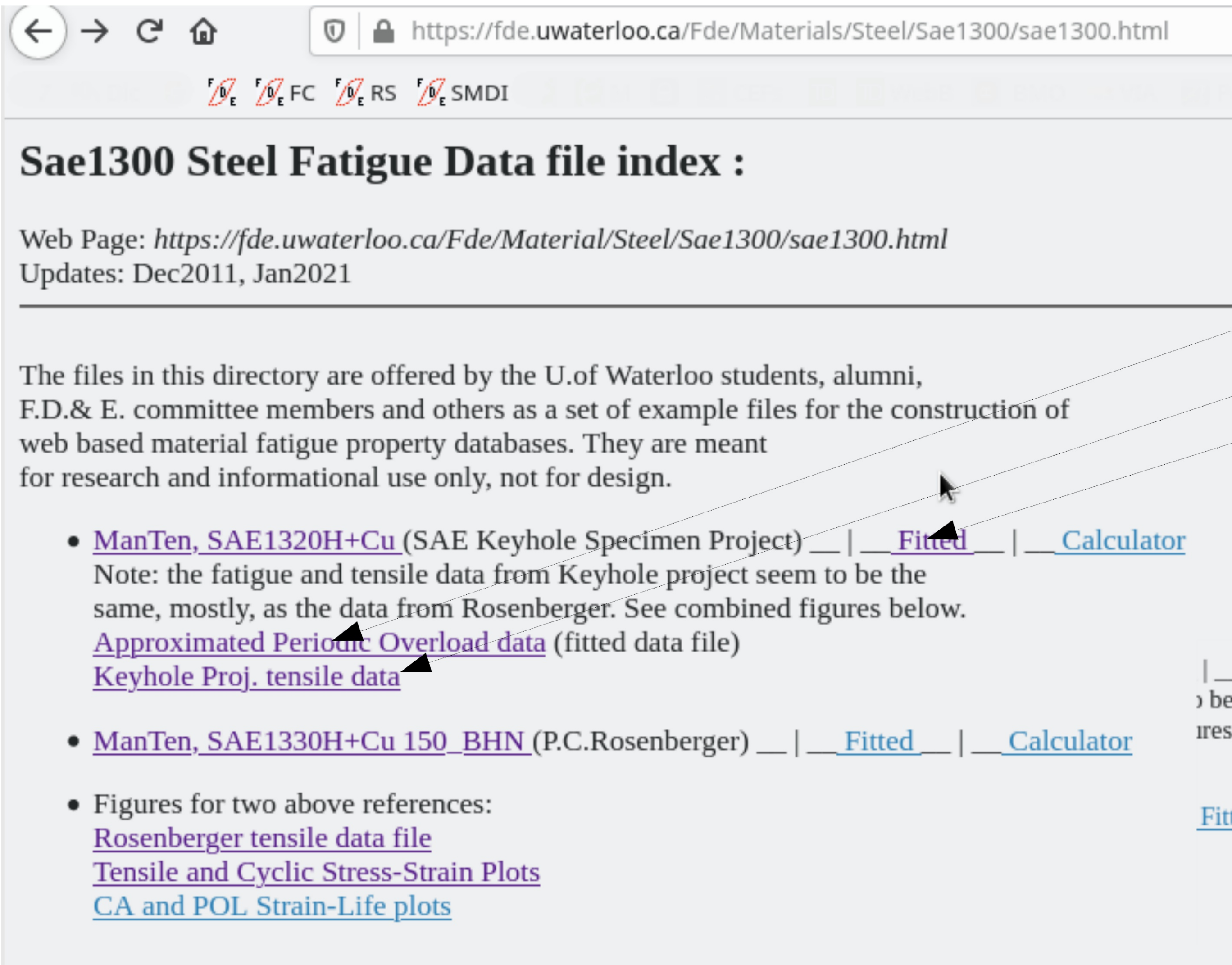
   and the files and folders should now look like this:

```
Test/CleanPdprop2> ls
0_README              decimate      hilo2             replaceline
Allcompile            decimate.f    hilo2.f           replaceline.f
convert2MPa_mm        delete1arg    makeRepSaefcalc2  saefcalc2
convert2MPa_mm.f      delete1arg.f  pdrain            saefcalc2.f
convertParis2table    getFADs       pdrain.f
convertParis2table.f  getFADs.f     PlateEdgeFlawIPB
```

Setup cont’d

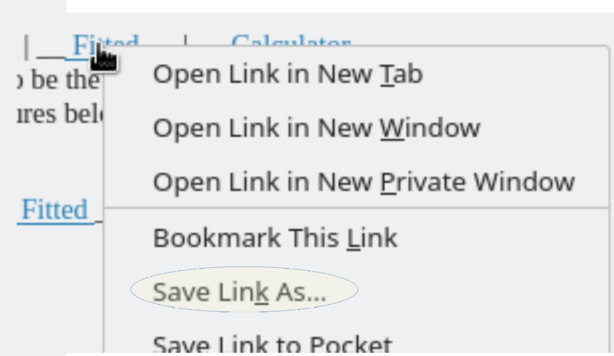
6. We can now enter the PlateEdgeFlawIPB folder and add the material and tensile data files.

(a) In your web browser go to:
https://fde.uwaterloo.ca/Fde/Materials/Steel/Sae1300/sae1300.html



Pick up these files and place in folder PlateEdgeFlawIPB

(Hover cursor over the name, right click, and use “Save link as,,,” option in Firefox for example)



Setup cont'd:

7. Create the load history file you want to run and also place it into folder PlateEdgeFlawIPB

   For our example we will use the prepared F.D.E. Transmission file mentioned on pgs. 8 &9
   A copy of it can be obtained here:

https://fde.uwaterloo.ca/FatigueClass/Chap10Using/FCoursePdprop2/transm1000N.txt

   After downloading the material files and the load history the folder contents should contain these:

```
CleanPdprop2/PlateEdgeFlawIPB> ls
a36+1015.dadn                        loads4rain.out              plateEdgeFAD-IPB.f
a36+1015.dadn.user                   makereport8                 plateEdgeFlawIPB
a36_Mattos_mono_engrSS_FLAT.txt      makereport8.bak             plateEdgeFlawIPB.f
dadnTable                            matfile                     plotAlldadnLines.f
fads.table                           merged_a36_fitted.html      plotFADs
filtExample_2.5.pdf                  merged_a36_fitted.html.user keyhManTenTens.txt
keyholeManTen-str-2Nf_fitted.html    pdprop.env                  setup8
keyholeManTen-str-2NfPOL_fitted.html pdprop.env.pef              setup8.bak
load1.txt                            plateEdgeFAD-IPB            transm1000N.txt
```

The two "fitted" files, keyholeManTen-str-2Nf_fitted.html and keyholeManTen-str-2NfPOL_fitted.html are used for CA and VA crack initiation simulations. The tensile file keyhManTenTens.txt will be used for Failure Analysis Diagram (FAD) simulations.

Setup cont'd:

8. Edit the pdprop.env environment file.

The file pdprop.env provides data file names and other inputs for control of the runtime propagation simulation programs:

plateEdgeFlawIPB.f (peak by peak crack propagation model)
plateEdgeFAD-IPB.f (Failure Analysis Diagram(FAD) computation code)

and also some of the utility programs in the folder above:

pdrain.f for “Rainflow cycle counts”
saefcalc2.f crack initiation life computation
getFADs.f compute FADs from tensile data file

Generally the pdprop.env file has a number of flags and variables already preset for running, in this case: Edge Flaw with **I**n**P**lane **B**ending and axial loads. A similar file is used in all the various propagation programs. Different programs are required to simplify the process of defining the Stress Intensity Factors for differing crack geometries, such as Plate Surface flaw, Plate Long Surface flaw, Plate Thru Crack etc. Thus one only needs to set items such as load file name, da/dN type, tensile data file name, and a few others specific to the user’s project.

The next pages will describe what needs to be set by the user in pdprop.env

Setting up the **pdprop.env** environment file

The file is a bit long so we will look at it in sections. It is a text file.

Set the plate thickness, width and the initial crack length.

```
#Keyhole specimen Transmission history example 2021
# This file contains the starting filenames, variables etc
# for the Crack Propagation  programs.   It should be edited by the
# user before each simulation.
#

#TYPE= plate_edge_flaw       #with or without weld using ACTIVATEs:
#ACTIVATE_MmMb=      1   #  Deactivate  = 0
#ACTIVATE_MkmMkb=  0   #  Not used in plate_edge_flaw
#ACTIVATE_fw=            0   #  Not used in plate_edge_flaw
#                              #Other      #TYPE=  options:
#                              # plate_surface_flaw
#...other comments here
#                              # These problem types are used to activate
#                              # appropriate Fw, Mm, Mb, files etc.

# The factors described in this section may be ignored if not applicable to
# the particular problem type described above.
# (All dimensions in mm)
#B= 9.53          # plate (or pipe wall) thickness
#W= 68.58         # plate width
#ri=  00.         # Internal diameter if pipe problem. Ignored if not pipe
#azero= 7.263 # initial crack depth, 1/2 of notch(4.763) + 2.5mm
#czero= 0.0       # initial 1/2 crack width at surface
#L= 00.           # Weld Feature width.  Ignored  if ACTIVATE_MkmMkb= 0 (above)
...more
```

pdprop.env file cont'd:

The history is actually read from standard input so the placement of the name here is just for the report writer script

```
#HISTORYFILE=  transm1000N.txt   # historyFileName
#          # (For keyhole CA hist contained 100 cycles.)
#          # Adjustments to load file variables:
#          # Note that the MEANADD (below) is added AFTER the MAGFACTOR is applied.
#MAGFACTOR_m= 1.0     # Multiply factor on membrane load. Result should be MPa
#MAGFACTOR_b=  1.0    # Multiply factor on bending load term. Result should be MPa
#MEANADD_m=     0.0   # Mean shift in MPa added to membrane stress.
#MEANADD_b=     0.0   # Mean shift in MPa added to bending stress.

#MAXREPS=  10000   # Max no. history repeats in simulation.
#                  # One repetition or application of the load history is
#                  # also called a "block" of cycles.
#
#
#MATERIAL=  keyholeManTen-str-2NfPOL_fitted.html  #File name of material fitted data
#                       If now wanted enter:  none
#                       This file is used to define the cyclic
#                       stress-strain curve, and the Neuber Product curve.
#
#Kt= 3.2             # Ktgross.    The history is in gross stress for cracks.
#                    #Stress Conc. Factor, presently for crack init. calcs only
#                    # It is read by makereport to run initiation
#
```

MAXREPS=
specifies how often the history will be repeated in the simulation.
(set it low initially to prevent very large output file)

#MATERIAL=
specifies the stress-strain-life "fitted" file for crack initiation simulation

#Kt=
specifies the **Gross** stress concentration factor when there is a notch.
Also just for crack initiation at this time.

...more

pdprop.env file cont'd:

#DADN= specifies how you want to define your da/dN curve(s)

```
#DADN= HASEGAWA2019          # Can be "table"  "Paris"  "USER" "BS7910"
#                            # "HASEGAWA2019" "HASEGAWA2016" or "ASME1994"
#                            # If "table" only one dadn file is expected
#                            # and it is specified by #DADN_TABLE= filename below.
#                            # If "USER" copy your prepared dadn files into
#                            # filenames: dadnTable01, dadnTable02, dadnTable03, etc.
#

#DADNFileNum= 1    # Activated only with "USER" above.
#                  #No. of dadn files or curves
#                  # (ignored if NOT "USER" above.)
#                  # Re-run "setup8" when changing this.
#


#DADN_PARIS= 0.0 0.0 0.0 0.0 none     # Kth  a  m  Kc units   (ignored if #DADN= table )
#DADN_TABLE= none              # da/dN digitized da/dN curve file  for material,
#                            including the threshold, and KIc.
#                            See example file:  a36+1015.dadn
#                            If a threshold exists, put in a vertical line
#                            (with two identical X-axis points).
#                            If the threshold needs to be "turned off" then
#                            do NOT put in a vertical line at low da/dN.
#                            (Only active when #DADN= table )
#
```

...more

pdprop.env file cont'd:

```
##FAD Stuff:       (see BS7910 standard for details)

#TensileFile=  keyhManTenTens.txt    #enter "none"  if no FAD

#PmEOL= 100.          #Set these so that Pm+Pb= 0.82*Syield for default.
#PbEOL= 164.
#                     # ManTen 0.82*Sy= 264
#Kmat= 3200.          # KIc See BS7910 Sec. 7.1.5 for details.
#PinJoint= 0          # not used for plateEdgeFlaw.f
#
#BLOCKSKIP=   1.0 percent     # At the end of each block check if the previous
#                        two blocks of cycles had similar damage (crack
#                        extension) within this percentage.  If TRUE then
#                        simply skip the simulation of the next block,
#                        but just add the expected damage.  Continue by
#                        simulating the block after the skip.
#                        A value of 0.0 will disallow skipping blocks.
#SAVELEVEL=  0           #Amount of output saved to disk:
#                     #   3=lots  2=medium  1=minimal
#                     #   0= save #crk=  data into binary direct access file only
#                     #     No #crk= data will be written into the text logfile.
#                     #     Use for large output files with lots of cycles.
```

#BLOCKSKIP= and #SAVELEVEL= are presently not used. They have been found unnecessary.

9. Setup Check using setup8 script

The script setup8 (a text file you can read) is used to read in the pdprop.env descriptors, check them, and do any necessary preparation work priot to the simulation run.
The script will ask you several questions requiring a Y or N reply as it performs its tasks.

When the script is completed scroll up through its output for any warnings or errors that may affect your simulation.

Execute the script with :

```
./setup8
```

Upon running this script in my folder setup8 threw the error message:

```
Checking for Tensile strain-stress  filename in pdprop.env...
found:    #TensileFile= keyhManTenTens.txt
Error: file= keyhManTenTens.txt not found. Check pdprop.env for correct filename.
```

and required that I copy the file to the folder. After which I needed to run setup8 again.

... more

Setup Check cont'd:

The setup8 script will end with the following suggestions of how to start the simulation run and how to create a final report after it ends:

End of setup check. If all is well you can now run commands like:

./plateEdgeFlawIPB scaleFactor <loadHistory >outputFile

eg.: ./plateEdgeFlawIPB 1.0 <load1.txt >plate1.0

eg.: ./plateEdgeFlawIPB 2.0 <load1.txt >plate2.0

Note: if you have imported files from Windows systems you should probably convert the DOS type text files to Linux format with the command:

dos2unix filename

Note2: As your simulation is running you can observe the creation of the local files with the commands:

ls -lt | head

or:

ls -lt | head -20

After running each simulation you can prepare a report with the command:

./makereport5 outputFile

for your simulation output file.

The above script "makereport5" is located in this folder. You can edit it to alter or add features.

Note: If you change stuff in pdprop.env it would be good to re-run setup8

Good luck

Run Time 1

Assuming that setup8 has checked things out OK, we now run an actual crack propagation simulation.
The programs like plateEdgeFlawIPB are quite fast and they write out a great deal of data into a random access file fadInput.rand ; each peak generates -12 floating point numbers. Thus the disk storage can fill up quite fast. Until you are used to the program behavior you should set

#MAXREPS= to a low number in pdprop.env. We have set it to 10000.

(In our case each "REP" is about 1700 reversals. Thus the program would stop after 17,000,000 revs (or 8.5m cycles). In the present example we will create about 3million revs which will occupy about 100Mbytes in fadInput.rand)

The program also creates an text file but it is much smaller.

Running:
> ./plateEdgeFlawIPB 15.6 < transm1000N.txt > transm15.6kN.out

plateEdgeFlawIPB.f vers. 5.2
#Usage: plateEdgeFlawIPB scale <histfile >outfile

#Opening pdprop.env file...
#Opening matfile: Snominal, Stain, Stress Table ...
#filename= matfile
...more

Run Time 2

Text will appear on your command window. You can check for Errors but otherwise ignore it

```
#discrete #dadn for dadnTable01; R= -5.00  discDKinterval=  0.1062581E+02
#discrete #dadn for dadnTable02; R= -3.00  discDKinterval=  0.7805403E+01
#discrete #dadn for dadnTable03; R= -2.00  discDKinterval=  0.5242997E+01
#discrete #dadn for dadnTable04; R= -1.00  discDKinterval=  0.3537514E+01
#discrete #dadn for dadnTable05; R= -0.50  discDKinterval=  0.2753983E+01
#discrete #dadn for dadnTable06; R=  0.00  discDKinterval=  0.1900216E+01
#discrete #dadn for dadnTable07; R=  0.50  discDKinterval=  0.1610485E+01
#discrete #dadn for dadnTable08; R=  0.90  discDKinterval=  0.1340734E+01
#Rhalfway   1  -4.00
#Rhalfway   2  -2.50
#Rhalfway   3  -1.50
#Rhalfway   4  -0.75
#Rhalfway   5  -0.25
#Rhalfway   6   0.25
#Rhalfway   7   0.70
#Rhalfway   8   1.00
#Rhalfway   9   1.00
#Rhalfway  10   1.00
#dadn #Interval for discretizing  K for memory,discDKintervalM=  0.1062581E+02
# Getting Load history file from std.input
#history #Data input completed. nloads=  1708
#history #StotMax=   148.5
#history #StotMin=   -73.6
#history #Where Stot = Smembrane + Sbending

# Eliminating non-reversal points from history...
#history #Elimination window = 2% of (StressMax-StressMin)=   4.44
#history #Any 1/2 cylce smaller than this will be eliminated...
#history #No points needed to be eliminated.
```

Run Time 3

```
#ststot(nstart):    148.511993            1
 #ststot(nstart+1):    61.0895996            2
 #ststot(iput-1):    61.2456055          1707
 #ststot(iput):    29.4371986          1708
 #iendDir:          -1
 #ijoinDir:           1
 #ibeginDir:          -1
#history #BugCheck: GoodPtsFound=    1708 GoodPtsSaved=    1708 should be nearly equal?.
#NLOADSETS=     1708 (loads in each history repetition after filtering
#Wrote TotStress=(Sm+Sb) out for rainflow. nloads=     1708  into file: loads4rain.out

# Opening random access output file:  fadInput.rand ...
#Random Access output file: fadInput.rand   opened.
#Warning: getCracks2: Out of deltaK range, ifile=   6  Kmax,Kmin,deltaK:  0.310E+04 -0.733E+03  0.383E+04  R=  0.00
#Warning: getCracks2: Out of deltaK range, ifile=   6  Kmax,Kmin,deltaK:  0.315E+04 -0.701E+03  0.385E+04  R=  0.00
#Warning: getCracks2: Out of deltaK range, ifile=   6  Kmax,Kmin,deltaK:  0.315E+04 -0.743E+03  0.390E+04  R=  0.00

#Fracture: lobj90.gt. xKmat :  2772085   1624      1 dK=  0.2582073E+04 lobj90=  0.3208995E+04 xKmat=  0.3200E+04
#Last: nrev=    2772085 a=  0.3814059E+02   nblk=       1624 nact=          1 nrecord=    2772085
```

```
CleanPdprop2/PlateEdgeFlawIPB> lst
-rw-r--r-- 1 faconle users 99795060 Jan 14 21:33 fadInput.rand
-rw-r--r-- 1 faconle users     589547 Jan 14 21:33 transm15.6kN.out
-rw-r--r-- 1 faconle users       30793 Jan 14 21:33 loads4rain.out
```

lst is an alias for ls -lt | head -10

End of simulation notice

These files are input for making the report

Make a Report 1

The script makereport8 can be use after a simulation run to make graphs of the crack propagation results and also to compute the crack initiation life.

Information on each peak (tensile or compressive) of the simulation run is contained in the fadInput.rand file. The script will select 1000 points out of the complete set for plotting and other summarization purposes. If needed the user could open and examine the fadInput.rand file to obtain more information. The access format for the file is described in the plateEdgeFlawIPB.f or plateEdge-FAD.f source files. With reference to the simulation output text file

```
./plateEdgeFlawIPB    15.6  < transm1000N.txt  > transm15.6kN.out
```

we can now activate the report script: ./makereport8 transm15.6kN.out

Again there will be a considerable amount of text notifications that appear on the user's monitor. Scan them for any "ERROR" messages.

The script, aside from runing the FAD, Rainflow, and crack initiation programs, will create text summaries and many plots (using gnuplot) first in a web type HTML file. In our case it would be named: transm15.6kN.out.html
This file can be viewed from a web browser. In addition a *.pdf file is created using a converter program called " htmldoc "

(Aside note: In Linux Mint I have had problems with htmldoc output graphs so you may have to get the tar files and build from https://github.com/michaelrsweet/htmldoc/releases)

Make a Report 2

Ok, lets make the example report and show typical output.

```
> ./makereport8  transm15.6kN.out
Preparing input data summary for html page ...
# plateEdgeFAD-IPB.f  vers. 0.70
#Opening pdprop.env file...
# #Opening fads.table  ...

#MAXRECORDS=    2772085 In last rec. Reversal=    2772084
#MAXREVERSALS=    2772084
#MAXBLOCKS=       1623
#NACT=       1708
#plateEdgeFAD-IPB: For plotting:
#   No. Recs with reversals: nitems=        2772084
#   No. of plot Intervals:   nIntervals=       1000
#   No. Recs per Interval:   nrecsPerInt=       2772
#   No. of leftover items:   nremainder=          84
 #scanning remainders, jrecStart,jrecEnd=      2772003     2772085
 #Done with remainders
1001 transm15.6kN.out.dat
doing a tail -250 ...
0.171E+02    2084545       1221         785  1243.  -616.   23.9  -11.8  124.6  -61.8  0.389  0.412
0.494  0.751  0.776  0.931 1.5043 1.0799
fetching line no.= 250 from the data   for label placement...
Fetching filtered history of Sb and Sm ...
#Plotting head and tail of hist...
xrange: 1658:1708
Plotting da/dn table from text results file ...
Starting plot of Kmax and Kmin for direction  a ...
Starting plot of the Factors Mm, Mb, for crack ...

Converting the random access file  to FAD interval file...
Starting plot of the FAD items ...
```

Make a Report 3

As before these are just notifications of events in the script.
Check for errors.
Then view the pdf

```
 Checking if Crack initiation life can be calculated...
found:    #MATERIAL= keyholeManTen-str-2NfPOL_fitted.html
 hilo2 Starts. Usage:  hilo2  <in  >out
#The 1st data line has  1 columns of numbers
#The 1st data line has  2 columns of numbers
 #Done. Scanned        1708  data lines.
 #Total lines=        1711
 #Highs:   1708.00000      148.500000
 #Lows :   1.00000000     -73.5999985
    ../pdrain 2 <loads4rain.out  >transm15.6kN.out.rain.txt
#pdrain vers. 1.2 Starts. Usage:  pdrain ichan <in  >out
# Rainflow count will be on Data column=    2
# First data line found. Read    2 columns.
 #Done max,min hunt. Scanned        1708  data lines.
 #Total lines in file=        1711
#HIGH=  0.1485000E+03 inLine=        1
#LOW=  -0.7360000E+02 inLine=      1570
# Counting will start at DataPt no.=
#NSTART=        1
#STARTVALUE=  0.1485000E+03
#Matrix Max=  0.1502627E+03 Min= -0.7536269E+02 nbins=  64
 #Wrap npointplus1 around to xvalue(1)...
 #Wrap npoint= around to xvalue(1)...
 #pdrain: Last ramp. iphase=3.
Starting Initiation calculations...
../saefcalc2  matfile 3.2 <transm15.6kN.out.rain.txt >transm15.6kN.out.initResults
 #Found History multiply factor:    3.20000005
Deleting temporary files:  loadgp0,1,1b,2,3,4,5,6  temp5,6,7
Done. html file is: transm15.6kN.out.html   Making pdf ...
    htmldoc --webpage --footer ..1 -f transm15.6kN.out.pdf transm15.6kN.out.html
PAGES: 20
BYTES: 836807
```

If you have multiple *.html files you could use a command like:

htmldoc --book -f name.pdf file1.html file2.html ...etc

Done. In Linux you can view pdf with command:

evince transm15.6kN.out.pdf

Example pdf is available at:

https://fde.uwaterloo.ca/FatigueClass/Chap10Using/FCoursePdprop2/transm15.6kN.out.pdf

Cleanup

> lstl (lstl *is an alias for* *ls -lt | head -40*)

```
total 101940
-rw-r--r-- 1 faconle users   836807 Jan 15 13:29 transm15.6kN.out.pdf
-rw-r--r-- 1 faconle users    52632 Jan 15 13:29 transm15.6kN.out.html
-rw-r--r-- 1 faconle users    30533 Jan 15 13:29 transm15.6kN.out.damage.png
-rw-r--r-- 1 faconle users   102664 Jan 15 13:29 transm15.6kN.outss.png
-rw-r--r-- 1 faconle users   543422 Jan 15 13:29 transm15.6kN.out.initResults
-rw-r--r-- 1 faconle users    30793 Jan 15 13:29 transm15.6kN.out.loads4rain.txt
-rw-r--r-- 1 faconle users     3736 Jan 15 13:29 transm15.6kN.out.matfile.txt
-rw-r--r-- 1 faconle users    21418 Jan 15 13:29 transm15.6kN.out.rain.txt
-rw-r--r-- 1 faconle users       70 Jan 15 13:29 transm15.6kN.out.rain.hilo
-rw-r--r-- 1 faconle users    50329 Jan 15 13:29 transm15.6kN.out.FAD.png
-rw-r--r-- 1 faconle users      480 Jan 15 13:29 fad2b_xy
-rw-r--r-- 1 faconle users      648 Jan 15 13:29 fad2a_xy
-rw-r--r-- 1 faconle users       69 Jan 15 13:29 fad1_xy
-rw-r--r-- 1 faconle users   141141 Jan 15 13:29 fadints.out
-rw-r--r-- 1 faconle users    29595 Jan 15 13:29 transm15.6kN.out.factors.png
-rw-r--r-- 1 faconle users    26190 Jan 15 13:29 transm15.6kN.out.Kmax+Kmin90.png
-rw-r--r-- 1 faconle users    32935 Jan 15 13:29 transm15.6kN.out.dadn.png
-rw-r--r-- 1 faconle users    26423 Jan 15 13:29 transm15.6kN.out.stressB.png
-rw-r--r-- 1 faconle users    24359 Jan 15 13:29 transm15.6kN.out.stressA.png
-rw-r--r-- 1 faconle users    60960 Jan 15 13:29 transm15.6kN.out.stress.png
-rw-r--r-- 1 faconle users    62488 Jan 15 13:29 temp.loads
-rw-r--r-- 1 faconle users    21091 Jan 15 13:29 transm15.6kN.out.crk.png
-rw-r--r-- 1 faconle users   141141 Jan 15 13:29 transm15.6kN.out.dat
-rw-r--r-- 1 faconle users   158909 Jan 15 13:29 transm15.6kN.outFAD.out
-rwxr--r-- 1 faconle users    21508 Jan 14 23:10 makereport8
-rw-r--r-- 1 faconle users 99795060 Jan 14 21:33 fadInput.rand
-rw-r--r-- 1 faconle users   589547 Jan 14 21:33 transm15.6kN.out
-rw-r--r-- 1 faconle users    30793 Jan 14 21:33 loads4rain.out
```

As you can see there are a lot of graphs and files created by makereport8 Typically I only **save the pdf file in some other folder** and then do the next run with perhaps another stress level.

You can then delete the non-essential files with

rm transm15.6*

You must also delete the fadInput.rand file before the next run   rm fadInput.rand

All Done. Good Luck with your Simulations !

On the previous page you can see who owns the files in this example. If you have questions or comments you can send them to that owner @gmail.com