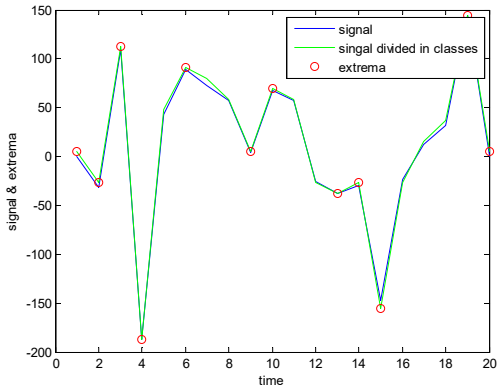
Cycle counting at Investmech

Dr. Michiel Heyns Pr.Eng.
T: +27 12 664-7604
C: +27 82 445-0510
mheyns@investmech.com

Agenda

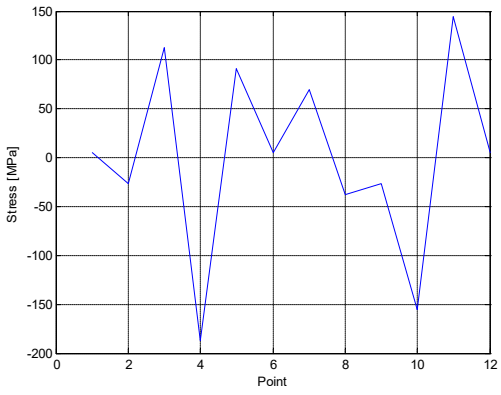
- The rainflow counting method
- Summary of process used to count cycles

Signal to demonstrate rainflow counting method



Graph was constructed using: sig2ext(S,[1:20],32)
S=randn(10,1)*100 where after the endpoints were made 0

The rainflow counting method



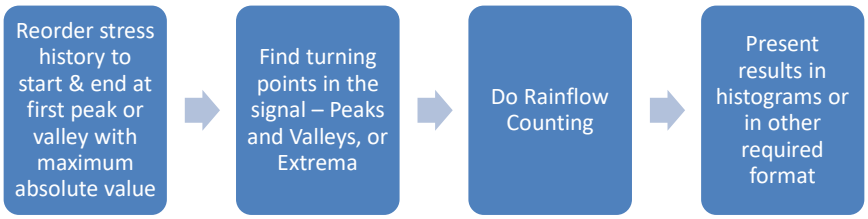
Turning points are:
5.2
-26.9
112.2
-187.4
90.8
5.2
69.4
-37.6
-26.9
-155.3
144.4
5.2

Lecturer demonstrates use in class

INVESTMECH

Cycle counting process

In practice millions of stress ranges need to be analysed, for which manual calculations are impractical. Numerical calculations are then carried out on computers to count the cycles.



INVESTMECH

What we use

- Matlab files stored in folder: RAINFLOW 2009-11-23
  - Golder contains routines downloaded from Mathworks website
  - No person is allowed to change any file in this folder
- Routines in this folder
  - Sig2ext.m – Peak-valley reduction
  - Rainflow.m – Rainflow counting
  - Rfhist.m – Plot histogram of amplitudes
  - Rfmatrix.m – Calculate equivalent of Markov matrix

Note, use 32-bit Matlab for this

Rainflow counting process & Matlab algorithms

Reorder signal

```
ii=find(max(abs(x))==abs(x));
S=[x(ii(1):end)',x(1:ii(1))'];
```

Extrema

- sig2ext.m
- Not necessary if not particularly interested in extrema

Rainflow counting

- rainflow.m
- If not interested in extrema, only do Rainflow and indicate classes

Present results

- rfhist.m
- Rather use histogram

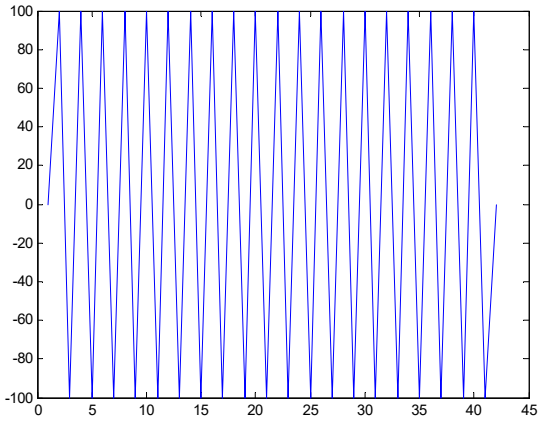
Generic signals for testing

- Fs = 1000 Hz
- t = 0:1/Fs:10 – to build a 10 seconds file
- sig = 100*sin(2*pi*t*2); This is a 2 Hz signal
- So, what do we expect:
  - For first range = 0 to 100, one reversal (½ cycle)
    - Amplitude = 50, Mean = 50
  - Next 38 ranges of 200
    - Amplitude = 100, Mean = 0
  - Last range = -100 to 0
    - Amplitude = 50, Mean = -50

Sig2ext.m

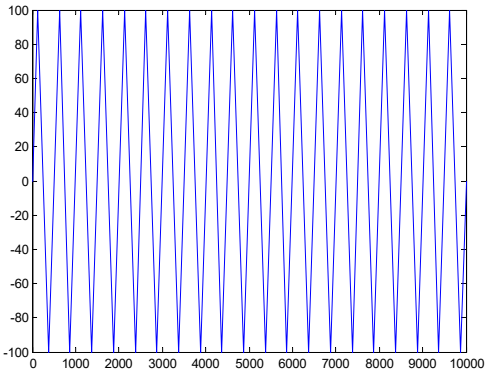
- Options:
  - [ext]=sig2ext(sig)
  - [ext,exttime]=sig2ext(sig)
  - [ext,exttime]=sig2ext(sig,dt)
  - [ext,exttime]=sig2ext(sig,dt,clsn)
- Let us test these functions one by one

[ext]=sig2ext(sig)



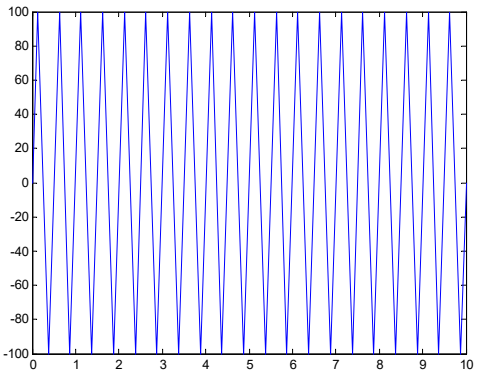
As shown, the turning points are at 100 and -100 and 0 for the end points
The horizontal axis is the reversal in consecutive order
This output can be used for strain life, stress life and other types of analyses

[ext,exttime]=sig2ext(sig)



In this function, exttime contains a vector with the point in the original sig where the peak or valley is
Plot(exttime, ext) gives the figure on the left
Scaling with time step make it is possible to see time of turning points

[ext,exttime]=sig2ext(sig, dt)



In this case the time step dt = 1/Fs
Note the horizontal axis that is now indicating the time in seconds where the turning point is for plot(exttime,ext) command

[ext,exttime]=sig2ext(sig, dt, clsn)

- clsn
  - a number of classes of SIG (division is performed before searching of extrema)
  - no CLSN means no division into classes
  - Therefore, for counting turning points, the range of the signal is first divided by the classes to determine the stress intervals used for determination of turning points
  - A random signal will demonstrate this effect the best
  - **According to BS 7608, $clsn \geq 32$**
    - **Use 40 at Investmech**

Peak-Valley reduction at Investmech

- Shall use the command:
  - [ext,exttime]=sig2ext(sig,dt,clsn)
  - Remember:
    - dt is the time step and is $dt = \frac{1}{F_s}$ where F_s is the sampling frequency of the signal in Hz
    - $clsn \geq 32$ according to BS 7608, use 40
    - ext is the vector with turning points
    - exttime is the time at which the turning point occurs

Peak-valley reduction or finding extrema

- **[ext,exttime]=sig2ext(sig,dt,clsn);**
- clsn >= 32, Investmech use 40

Rainflow counting with Matlab

- Syntax
  - c = rainflow(x)
  - c = rainflow(x,fs)
  - c = rainflow(x,t)
  - c = rainflow(xt)
  - c = rainflow(___,'ext') % to indicate input consist of already determined extrema

For amplitudes:
rf(1,:)'
For mean values:
rf(2,:)'
For cycles:
rf(3,:)'

Peak-valley reduction or finding extrema

- **[ext,exttime]=sig2ext(sig,dt,clsn);**
- clsn >= 32 according to BS 7608, Investmech use 40

Rainflow countiing

- **rf = rainflow(ext,dt,'ext');**
- **[c,hist,edges,rmm,idx] = rainflow(ext,exttime,'ext'); % Signal-processing TB**

Use this indication if the signal is extrema determined by sig2ext

Present results: Rainflow.m from the Rainflow algorithms

- Histogram in Rainflow folder:
  - [no,xo] = rfhist(rf,x,rfflag)
    - x = bins in the analysis
      - Use $x = clsn \geq 32$ for analysis. Investmech uses $x = clsn = 40$
      - However, less can be used because this is for presentation of results
    - rfflag = 'ampl' for amplitude, 'mean' for mean value, 'freq' for frequency, 'period' for time period of extracted cycles.
    - no = vector of extracted cycles
    - xo = vector of bin locations

Present results: Rainflow.m from Signal Processing toolbox

- Histogram
  - histogram('BinEdges',edges','BinCounts',sum(hist,2))
- Or
  - Calculate the mean of the bin:
    - cc=diff(edges)/2+edges(1:end-1); % Find the mean of the bin.
  - Plot the values or write to table
    - plot(cc,sum(hist,2));
    - xlabel('Range');ylabel('No of cycles');

Commands to use

Peak-valley reduction or finding extrema

- **[ext,exttime]=sig2ext(sig,dt,clsn);**
- clsn = 40

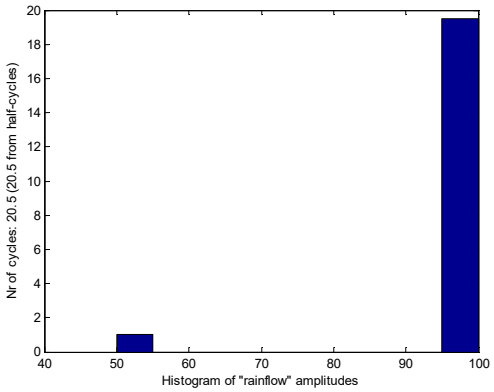
Rainflow countiing

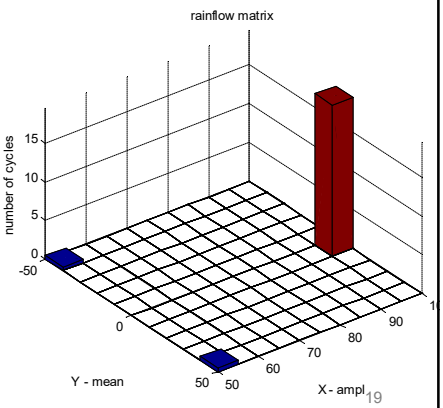
- **rf = rainflow(ext,dt,'ext');**

Rainflow countiing

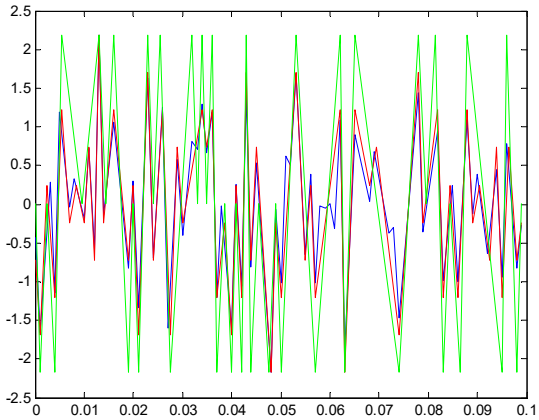
- **[no,xo] = rfhist(rf,x,rfflag);**
- Rfflag = 'ampl' for amplitude, = 'mean' for mean values, 'freq' for frequency, 'period' for time period of extracted cycles.

Histogram of counted cycles





Effect of clsn on results: [ext,exttime]=sig2ext(sig, dt, clsn)



sig=randn(1,100);
Blue line = [ext,exttime]=sig2ext(sig,dt)
Red line = [ext,exttime]=sig2ext(sig,dt,10)
Green line = [ext,exttime]=sig2ext(sig,dt,3)
Clearly, the less classes are used, the more small signals are ignored and lower turning points increased to next level
So how much is required?
>32 CLASSES ACCORDING TO BS 7608

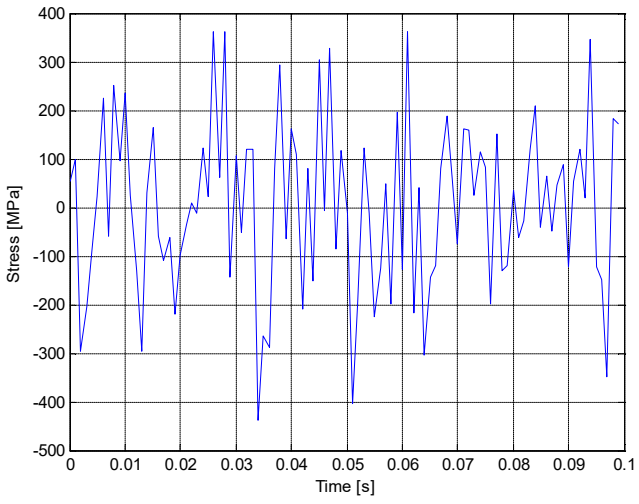
Example: Downloaded m-file if no signal processing toolbox

- Type in the following Matlab Commands:

```
Fs=1000;
dt=1/Fs;
Stress=200*randn(1,100);
[ext,exttime]=sig2ext(Stress,1/Fs,32);
rf=rainflow(ext,'ext');
rfhist(rf,32,'ampl'); %This will plot the histogram
[no,xo]=rfhist(rf,32,'ampl'); %To get histogram results
rfmatrix(rf); %To plot the stress matrix
```

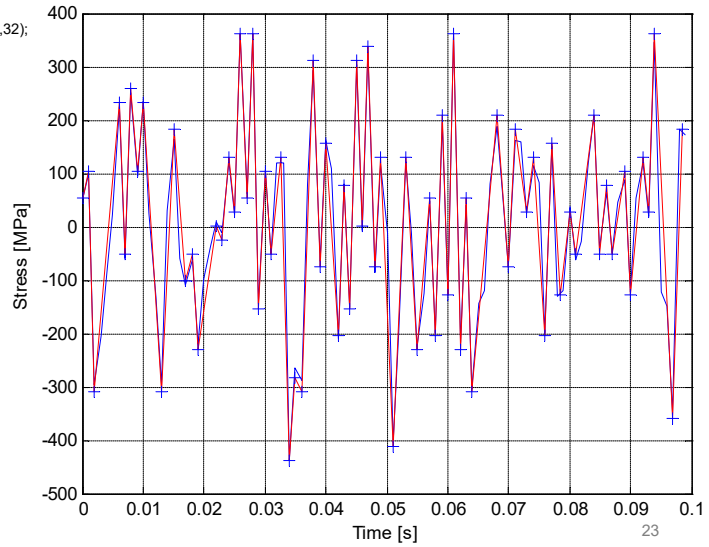

Example

```
Fs=1000;
dt=1/Fs;
Stress=200*randn(1,100);
t=(0:length(Stress)-1)*dt;
plot(t,Stress)
grid;
xlabel('Time [s]');
ylabel('Stress [MPa]');
```



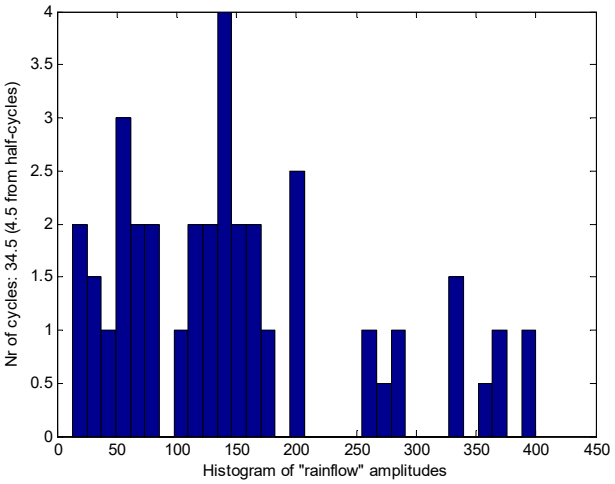
INVESTMECH

```
[ext,exttime]=sig2ext(Stress,1/Fs,32);
hold on
plot(exttime,ext,'r')
plot(exttime,ext,'+')
```

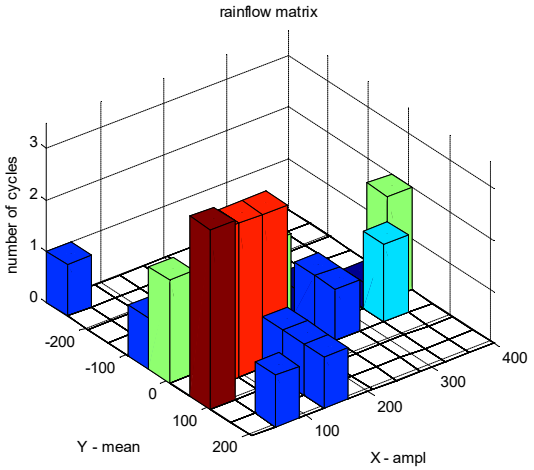


INVESTMECH

```
rf=rainflow(ext,'ext');
rfhist
```

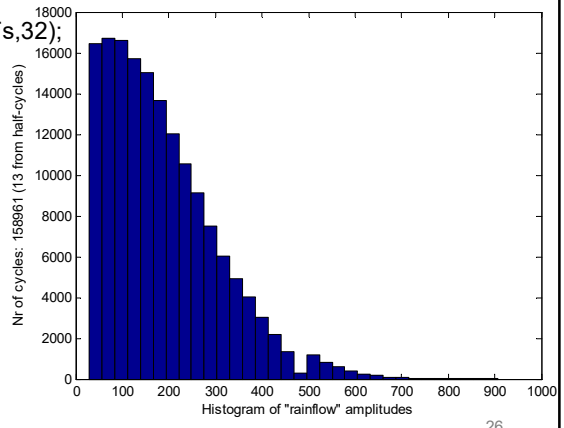


rfmatrix(rf);



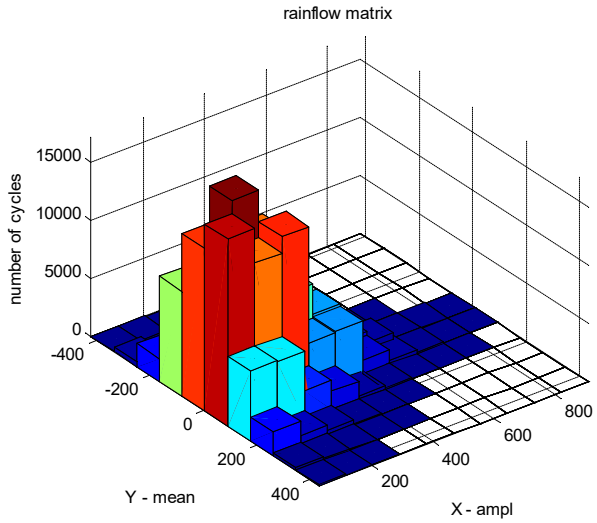
For 500000 points things look different

```
Fs=1000;
dt=1/Fs;
Stress=200*randn(1,500000);
[ext,exttime]=sig2ext(Stress,1/Fs,32);
rf=rainflow(ext,'ext');
rfhist(rf,32,'ampl')
```



For 500,000 points the rfmatrix looks more realistic

rfmatrix(rf)



If you have Signal Processing Toolbox

- Signal processing toolbox has Rainflow algorithm
- Outputs different to downloaded version
- To analyse and present:
  - [c,hist,edges,rmm,idx] = rainflow(ext,exttime,'ext'); % Signal-processing TB
  - histogram('BinEdges',edges','BinCounts',sum(hist,2))
  - xlabel('Stress Range')
  - ylabel('Cycle Counts')
- Or use:
  - [c,hist,edges,rmm,idx] = rainflow(sig2ext(200*randn(1,500000),1,50));histogram('BinEdges',edges','BinCounts',sum(hist,2));
  - cc=diff(edges)/2+edges(1:end-1); % Find the mean of the bin.
  - plot(cc,sum(hist,2));
  - xlabel('Range');ylabel('No of cycles');
- To remove non-zero cycles rows:
  - range=diff(edges)/2+edges(1:end-1);
  - ii = find(sum(hist,2)~=0);
  - rangefinal=range(ii);cycles=sum(hist(ii,:),2);
  - [rangefinal cycles]
- To determine which function is active: type help rainflow

Conclusion

- Re-order the signal to start and end with the maximum absolute value
- Cycles counted
- Results is vector with mean, amplitude and number of cycles
- This can now be used for damage calculation using any S-N or Sr-N curve
  - Note for damage calculation using weld stress range dependant Sr-N curves, you can further reduce values by using rfhist as explained
    - Stress range = 2 x stress amplitude
- Remember to do mean stress correction where required
  - Use Goodman mean stress correction at Investmech