# Rainflow-counting algorithm

From Wikipedia, the free encyclopedia

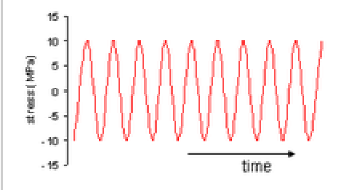

Figure 1: Uniform alternating loading

The **rainflow-counting algorithm** (also known as the "rain-flow counting method") is used in the analysis of fatigue data in order to reduce a spectrum of varying stress into a set of simple stress reversals. Its importance is that it allows the application of Miner's rule in order to assess the fatigue life of a structure subject to complex loading. The algorithm was developed by Tatsuo Endo and M. Matsuishi in 1968.[1] Though there are a number of cycle-counting algorithms for such applications, the rainflow method is the most popular as of 2008.

Downing and Socie created one of the more widely referenced and utilized rainflow cycle-counting algorithms in 1982,[2] which was included as one of many cycle-counting algorithms in ASTM E 1049-85.[3] This algorithm is used in Sandia National Laboratories LIFE2 code[4] for the fatigue analysis of wind turbine components.

Igor Rychlik gave a mathematical definition for the rainflow counting method,[5] thus enabling closed-form computations from the statistical properties of the load signal.

For simple periodic loadings, such as Figure 1, rainflow counting is unnecessary. That sequence clearly has 10 cycles of amplitude 10 MPa and a structure's life can be estimated from a simple application of the relevant S-N curve.

Compare this with Figure 2 which cannot be assessed in terms of simply-described stress reversals.

**Contents** [hide]



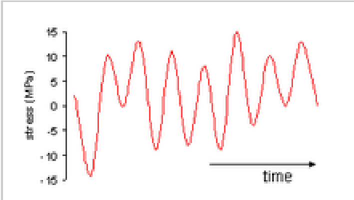

Figure 2: Spectrum loading

## Algorithm [edit]

1. Reduce the time history to a sequence of (tensile) peaks and (compressive) valleys.
2. Imagine that the time history is a template for a rigid sheet (pagoda roof).
3. Turn the sheet clockwise 90° (earliest time to the top).
4. Each *tensile peak* is imagined as a source of water that "drips" down the pagoda.
5. Count the number of half-cycles by looking for terminations in the flow occurring when either:
   - It reaches the end of the time history;
   - It merges with a flow that started at an earlier *tensile peak*; or
   - It flows when an opposite *tensile peak* has greater magnitude.
6. Repeat step 5 for *compressive valleys*.
7. Assign a magnitude to each half-cycle equal to the stress difference between its start and termination.
8. Pair up half-cycles of identical magnitude (but opposite sense) to count the number of complete cycles. Typically, there are some residual half-cycles.

## Example [edit]

- The stress history in Figure 2 is reduced to peaks and valleys in Figure 3.
- Half-cycle (A) starts at tensile peak (1) and terminates opposite a greater tensile stress, peak (2); its magnitude is 16 MPa.
- Half-cycle (B) starts at tensile peak (4) and terminates where it is interrupted by a flow from an earlier peak, (3); its magnitude is 17 MPa.
- Half-cycle (C) starts at tensile peak (5) and terminates at the end of the time history.

- Similar half-cycles are calculated for compressive stresses (Figure 4) and the half-cycles are then matched.

| Stress (MPa) | Whole cycles | Half cycles |
|---|---|---|
| 10 | 2 | 0 |
| 13 | 0 | 1 |
| 16 | 0 | 2 |
| 17 | 0 | 2 |
| 19 | 1 | 0 |
| 20 | 0 | 1 |
| 22 | 0 | 1 |
| 24 | 0 | 1 |
| 27 | 0 | 1 |

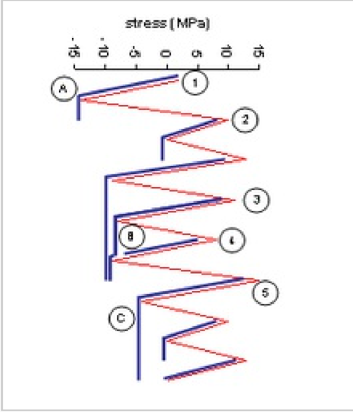

Figure 3: Rainflow analysis for tensile peaks

## Block Loading example [edit]

There are many cases in which a structure will undergo periodic loading. Assume that a specimen is loaded periodically until failure. The number of blocks endured before failure can be determined easily by using the Palmgren-Miner rule of block loading. The actual load history is shown in Figure 5.

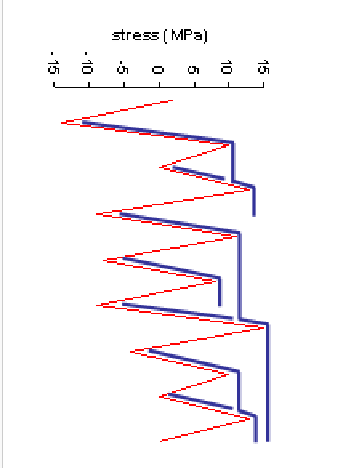

Figure 4: Rainflow analysis for compressive valleys

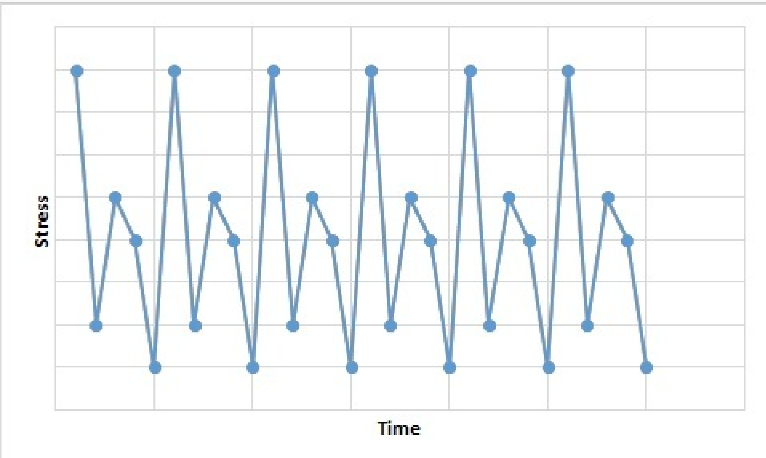

Figure 5: Periodic Loading with Time

If all of the similar loads are grouped together, it forms a series of block loads as shown Figure 6.

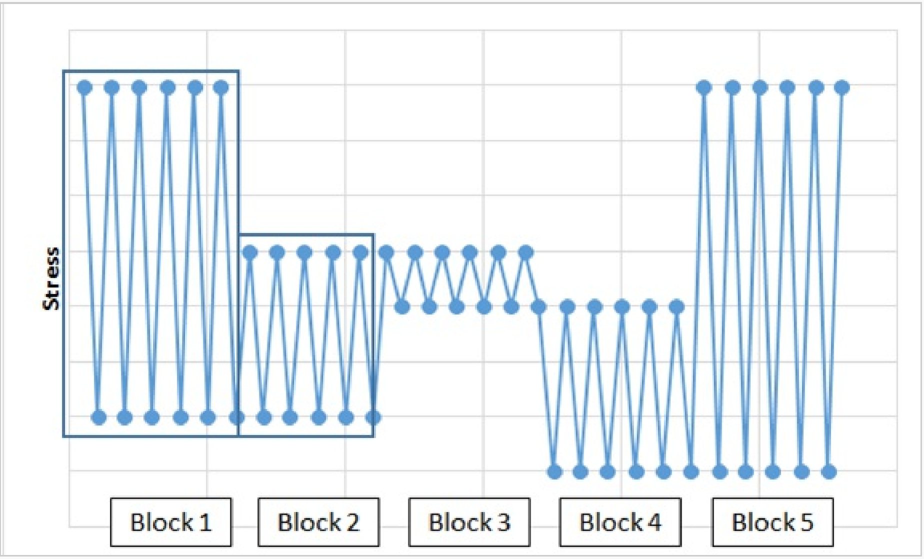

Figure 6: Periodic Loading Rearranged into Blocks

The Palmgren-Miner rule can be expressed as

$$B(\frac{N_1}{N_{f1}} + \frac{N_2}{N_{f2}} + ... + \frac{N_k}{N_{fk}}) = 1$$

where,

*B* = number of blocks

$N_k$ = number of cycles of loading condition, k

$N_{fk}$ =number of cycles to failure for loading condition, k

In this example, each $N_k$=1 because there is one instance of each load for every period of loading. To find $N_f$ (number of loads to failure) for each load the Goodman-Basquin relation can be used

$$N_f = \frac{1}{2}(\frac{\sigma_a}{\sigma'_f}\frac{1}{1-\frac{\sigma_m}{\sigma_u}})^{\frac{1}{b}}$$

where,

$\sigma_a$ = stress amplitude

$\sigma'_f$ = fatigue strength coefficient (material property)

$\sigma_m$ = mean stress

$\sigma_{ult}$ = ultimate stress (material property)

*b* = fatigue strength exponent (material property)

## Assumptions [edit]

There are two key assumptions made in in order to rearrange the loads into blocks. These assumptions may affect the validity of the procedure depending on the situation.

- The loads are independent.
- The order of loading does not matter.

## References [edit]


1. ^ Matsuishi, M. & Endo, T. (1968) Fatigue of metals subjected to varying stress, *Japan Soc. Mech. Engineering.*
2. ^ Downing, S.D., Socie, D.F. (1982). Simple rainflow counting algorithms. International Journal of Fatigue, Volume 4, Issue 1, January, 31-40.
3. ^ ASTM E 1049-85. (Reapproved 2005). "Standard practices for cycle counting in fatigue analysis". ASTM International.
4. ^ Schluter, L. (1991). Programmer's Guide for LIFE2's Rainflow Counting Algorithm. Sandia Report SAND90-2260.
5. ^ Rychlik, I. (1987) A New Definition of the Rainflow Cycle Counting Method, *Int. J. Fatigue 9:2, 119-121.*


## External links [edit]

- StoFlo freeware rainflow cycle counting Excel template
- Matlab Central rainflow counting method
- WAFO. Wave Analysis for Fatigue and Oceanography (Matlab)

- GAC freeware rainflow cycle counting
- Vibrationdata Rainflow Tutorials & Matlab scripts

Categories: Materials science | Elasticity (physics) | Fracture mechanics