FACULTY OF SCIENCE AND TECHNOLOGY

BACHELOR'S THESIS

Study programme / specialisation:	Spring semester 2023
Structural Engineering	Open/~~Confidential~~

Author:
Maja Mydland Hope

Supervisor at UiS:
Fredrik Bjørheim

Thesis title:
Finite Element Analysis and Fatigue Damage Assessment of a Railway Bridge Subjected to Variable Amplitude Loading

Credits (ECTS): 20

Keywords:	Pages:44
Fatigue damage assessment	+ appendix: 54
Palmgren-Miner's Rule	
Nonlinear fatigue damage model	Stavanger, 15.05.2023
Steel railway bridge	

Acknowledgement

This bachelor's thesis concludes my bachelor's degree in Structural Engineering, at the Department of Mechanical and Structural Engineering and Materials Science at the University of Stavanger, Norway.

Firstly, I would like to thank my supervisor, Fredrik Bjørheim, for great help and support during the assignment process. The advice and help with research and guidance for structural analysis and calculations have been crucial for the result and greatly appreciated. I would also like to thank Sudath C. Siriwardane for great support and guidance for the modelling in SAP2000. His dedication and interest to educate is truly inspiring.

The process of this thesis has been very educational and at times, challenging. The knowledge gained from three years of studying at the University of Stavanger have been very useful, but I have also learned a lot that was not included in earlier courses. My knowledge of fatigue prior to this thesis was limited, but after working with the subject during this process, I have gained a broader understanding of the concept, which I will bring with me in further studies and work life.

Abstract

Many of the railway bridges in Norway and the rest of the world are closing in on the end of their predicted service life, and the need for more precise fatigue life assessments and predictions is challenging for the railway authorities. Fatigue is a phenomenon that to this day is a mysterious and complicated process. At the time when many of the oldest bridges were built, fatigue was not particularly considered in design. Fatigue is one of the main reasons for failure in steel structures, and the consequences can potentially be fatal since a fracture is a quick process and without significant visible signs. The Palmgren-Miner's rule, which is the standard method used for fatigue assessments on older railway bridges, have been found to be imprecise, as it fails to properly account for the variable amplitude loading. The rule considers damage to be linear, while research has shown that the process is in fact highly non-linear.

This thesis performs a fatigue life assessment for an actual ageing steel railway bridge based on provided data of loading history. The conventional Miner's rule and one other proposed nonlinear model is utilized to estimate the damage and remaining fatigue life of Saulidelva Bridge on Bergensbanen, Norway. This thesis aims to compare and analyse the results obtained from the two fatigue assessment methods, shedding light on the discrepancies resulting from simplifications in modelling, the choice of partial safety factors and S-N curves, and most important, the incorporation of variable amplitude loading.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

FEM	Finite Element Methods
BCC	Body Centred Cubic
FCC	Face Centred Cubic
HCF	High Cycle Fatigue
LCF	Low Cycle Fatigue
VAL	Variable Amplitude Loading
DNVGL	Det Norske Veritas Germanischer LLoyd
CoV	Coefficient of Variation
LMF	Load Model Freight train
LMP	Load Model Passenger train

1. Introduction

1.1. Background

In the Norwegian railway system, there are over 2300 railway bridges, out of which approximately 1000 are steel bridges. More than 300 out of these bridges are over 100 years old. The oldest bridges were not designed with fatigue in mind, as the knowledge of fatigue in the industry were limited 100 years ago. Additionally, the total weight and lengths of the trains have increased throughout the years [1]. Steel bridges are regularly subjected to traffic loads that may be under their yield strength limit. While a single application may not cause any noticeable damage, the accumulation of structural damage over time can result in the cumulative failure process known as fatigue failure [2]. Fatigue is considered one of the most critical failure modes in steel structures, as is it is a sudden, brittle fracture from a global perspective. However, the cause of formation and failure mechanism is not well understood [3].

The Palmgren-Miner Rule or the linear damage rule is widely used as standard for fatigue design to predict remaining fatigue life [4]. The Palmgren-Miner Rule assumes that damage is accumulated linearly with each cycle until failure [5]. However, the damaging process is not a linear process, and numerous experimental studies have shown that the rule is flawed, as it does not properly consider the effect of the loading sequence [6]. Several alternative models have been developed to predict remaining fatigue life more accurately, but Palmgren-Miner's rule remains the only accepted model for industry standard. To take account for the loading sequence uncertainty, very high safety factors for fatigue design is used [7, 8].

1.2. Objective

 The objective of this bachelor thesis is to perform a finite element analysis and a fatigue life assessment of the ageing railway bridge over Saulidelva, which was built in 1906 and is a part of the Bergensbanen railway. Due to its high exposure to variable amplitude loading of variable amplitude loading from both passenger and freight trains, the bridge has certain elements that theoretically have reached the end of their fatigue lives. For the fatigue assessment, one linear model and one none-linear model is utilized, both applying different factors and S-N curves to compare the effect on results. Aside from estimating the remaining fatigue life, the findings of this thesis will contribute to understanding the impact of geometric simplification in modelling in programmes for Finite Element Methods (FEM) have on the accuracy of remaining fatigue life estimates, and the understanding of the performance of linear and nonlinear fatigue damage models. Nonlinear models are not yet approved in industry standards partly due to lack of empirical validation from multistage loading and from real situations.

2. Theory

2.1. Background

Fatigue is the most common cause of mechanical failure. There is no exact number of how many accidents are caused by fatigue, but several studies implies that about 50-90% of all mechanical failures are due to fatigue[9]. According to the American Society of Civil Engineers (ASCE) as much as 90% of all service failure in metallic components can be linked to fatigue [10]. For bridges, it can be found that 38,3% cases of damage are related to fatigue [11]. The main causes of fatigue failure are reported to be poor maintenance, fabrication failure and design deficiencies according to a study performed by Battelle Columbus Laboratories in 1983. The report concluded that the number of fatigue related accidents could be drastically reduced by better use of fatigue analysis models and technology [12, 13]. Fatigue is often associated with brittle failure, even for normally ductile materials, in which the final fracture happens rapid and with none or little plastic deformation [14].

Fatigue damage occurs when the structure is subjected to cyclic loading and unloading. The dynamic and fluctuating stresses caused from this action will over time result in failure. Failure can occur at a stress level lower than the static load bearing capacity [15]. These high cyclic stresses will over time lead to crack initiation. The cracks will grow under stress from alternating loads, and fracture will eventually occur. Fatigue failure is mainly caused by tension stress, but compressive stresses, bending, and torsional stress also may contribute to failure [14]. Compressive stresses contribute to crack initiation, but tensile stress is required for crack propagation. The cracks in the material will grow due to the variation of stress, not the absolute level of stress [16].

2.1.1. Fatigue Impact

Knowledge of fatigue is important to avoid major accidents and loss of life. A wide range of structures and infrastructures are subjected to the risk of fatigue failure. Aircrafts, bridges, offshore structures, cranes and pipelines are some examples of structures relevant for fatigue [12]. There are several examples throughout the history of fatal accidents caused by fatigue.

The most infamous accident caused by fatigue failure in Norway is the offshore platform Alexander L. Kielland in 1980 where 123 people died. A crack was initiated from the welding in one of the diagonal braces. Before the final fracture, the crack was over 1 meter long [15]. The crack expanded as ductile fibrous tearing, before failure took place as a brittle fracture [17]. As a result of this, the load exceeded the bearing capacity of the remaining braces, and the entire column was ripped apart from the platform.

It took less than 20 minutes from the fracture occurred until the platform had capsized [18]. This accident is to this day one of the worst offshore oil rig accidents in the history [19].

Another example is the Eschede Train Disaster in Germany in 1998. The train travelled at 200 km/h when it collided with a bridge. The force of the following train cars caused the bridge to collapse on top of several cars. The tragedy cost over 100 people their lives [20]. A steel axle on the first car of the train failed due to a fatigue crack. The crack was developed due to a deformed shape of the steel wheel, which again caused vibrations at high speeds. The tire fell off the train and got stuck through the floor of the first car. After followed a tragic series of chain reactions; the axle passed a switch which derailed one of the first cars, which slammed into the supports of an overpassing bridge [21].

Many railway bridges in Europe were built in the early 1900s. Due to the lack of knowledge about fatigue when many of the steel bridges were built, several fatigue damages have been found in the structures. This also includes bridges which has not yet reached the end of their predicted lifetime. Throughout the years, the amount of traffic has also increased on the railways. This is also a contributing factor to fatigue damage; if the loading exceeds the service limit [1, 22].

2.1.2. Fatigue History

The term "fatigue" was used as early as in 1854 to describe the phenomenon of failure resulting from cyclic loading. Several years before this, around 1840, engineers had started to notice the pattern of material failure after a certain period of service. To better understand this phenomenon, simple tests were performed on the components that were known to fail. In the years 1860-1870, a technologist in the railway system in Germany, August Wöhler, conducted the first systematic laboratory tests on axles subjected to alternating loads considerably lower than the static strength of the railway axle [23].After analysing his results, Wöhler presented his findings in a report in 1870, concluding that *"Material can be induced to fail by many repetitions of stresses, all of which lower than the static strength. The stress amplitudes are decisive for the destruction of the cohesion of the material" [24]*. Wöhler also established the foundation for the S-N diagrams in the form of tables, with the results subsequently plotted as curves by his successor Spangenberg [23].

Fatigue remained a puzzling phenomenon until the 1900s, as fractures occurred without any visible warning. However, numerous studies and tests were conducted during the 20th century, leading to the discovery that repeated load cycles initiate a fatigue mechanism in the material. The mechanism begins with a microcrack, which subsequently grows until fracture of the material [23]. To gain a deeper understanding of fatigue, Ewing and Humphrey investigated slip lines and slip bands, which can lead to

the development of microcracks [25]. Additionally, the relationship between stress and number of stress cycles, as presented by Wöhler, was plotted as a log-linear relationship by Basquin in 1910 and is still widely utilized as the common S-N curves today [26].

The study of fatigue has been a significant area of research in material science and engineering due to its complex nature and severe implications in structural failures. The linear cumulative damage model for variable amplitude loading, which is widely used to assess the fatigue life of materials, was first proposed by Palmgren [27] in 1924. Later, Miner [28] developed a criterion for linear cumulative fatigue damage in 1945., which is now commonly referred to as the Palmgren-Miner Rule. Despite these advancements, the increasing number of fatigue related accidents has led to continued efforts to understand the mechanism of fatigue and fracture mechanics. As a result, many alternative models for fatigue damage assessments have been proposed in the later years. Nonetheless, fatigue remains a challenging concept to comprehend and necessitates further research to expand the understanding of the topic [9].

2.2. Mechanisms of Fatigue

The mechanism of fatigue is still not fully understood as is it a complicated and confusing process. However, knowledge of the fatigue process is essential concerning the fatigue life and crack growth. [12] Fatigue is defined as a damage process in the material due to cycles of stress. The level of stress can be under the yield limit of the material, but the damage can still be accumulated by repeated stress application. If the material is subjected to cyclic loading of a certain level over a longer period, fatigue failure will eventually occur. The mechanism of fatigue is usually divided into three phases: crack initiation, crack growth and fracture [15].

2.2.1. Crack Initiation

Crack initiation begins on the surface of the material, typically close to a notch, at a point with high stress concentration, or where a defect is present. When loading is imposed, crystallographic slip planes in the material slides, and some grains in the material is subjected to plastic deformation. The planes will not slide back when loading is reversed due to cyclic strain hardening. On the other hand, nearby planes will slide in opposite direction. This mechanism happens on a microscopic level, and eventually results in a microcrack. Typically a component experiencing high cycle fatigue (HCF) will use up most of its lifetime in the crack initiation stage [15]. For a component with a polished surface and no material defects, about 95% of its lifetime is spent in the crack initiation phase. If the component has rivets, welds, abrupt changes in geometry of material defects, the crack initiation phase would be shortened[29].

2.2.2. Crack Propagation

After crack initiation, the microcrack will grow to from spanning over a few grains, to reach several grain boundaries. The crack will grow in a perpendicular direction of the largest tensile principal stress. After each load cycle, the crack front will have progressed slightly. The distance of advancement for each cycle is believed to be equal to the distance between two striations [15]. Striations is the term of ridges formed near the crack surface and is only visible on a microscopic level. During the crack propagation there is a plastic deformation at the crack tips, even though the maximum stress level of each cycle is below the yield strength. There is an amplification of the stress at this point so that the local stress level exceeds the yield strength [14].

2.2.3. Fracture

The final phase, fracture, occurs when the crack has grown to such a size that the remaining ligament of the cross section cannot withstand the applied load, or when the local stresses and strains at the crack front impose a brittle fracture [15]. Generally, there are two possible failure modes for metal: ductile and brittle failure. For fatigue, and especially HCF, brittle fracture is most relevant. Brittle fracture occurs without little to no noticeable plastic deformation.

2.2.4. Fatigue factors

The fatigue process is influenced by the following conditions:

- External loading cycles
- Geometry of structure
- Material properties
- Fabrication quality and residual stresses
- Environment of surroundings

When an **external loading** is applied, it can lead to various effects such as bending, torsion, and normal stresses, which can result in stress conditions near the crack initiation sites. Normal stresses act as main agents for crack initiation. The crack planes will be moved directly apart by normal stress. Shear stress due to torsion, will cause crack planes to shear to extension. The essential factors to consider in cyclic loading are the range of stress, mean stress, force variance, and number of cycles. Generally, the frequency of loading doesn't significantly affect the fatigue process – unless the structure is in a corrosive environment [14, 15].

In the design process, it's crucial to consider the **geometry of the structure** early on to ensure its fatigue durability. Areas subjected to high stress concentrations are typically in notches. Notches refer to abrupt changes in the geometry, such as edges, welds, or threads. At a notch, the stress can be magnified locally by a factor between 3 and 8 compared to the average nominal stress in the rest of the component. The increase in stress at notches is usually due to reduced cross-sections and local disturbances caused by the notch itself. To reduce the stress in these areas the structure's dimensions can be increased or the notch radius can be improved. However, increasing the item's dimensions may lead to higher weight and costs while decreasing its fatigue resistance. Hence, optimizing the notch geometry is always preferable [15, 30].

For the material properties and surface effects, yield and tensile strength, modulus of elasticity, are all factors influencing the fatigue strength of a structure. In addition to this, specific tests are performed to determine fatigue life [15]. For many fatigue fractures, the crack initiation takes place at the surface of the component, especially at points susceptible to higher stress concentrations. Therefore, it has been observed that the fatigue life is sensitive to the condition of the surface of the component. Markings and scratches can be introduced on the component from machining, and this may affect the fatigue life. Fatigue life increases as the roughness of the surface decreases [31].

If the **environment surrounding a structure** differs from the typical air conditions, it can alter its fatigue behaviour. The primary types of fatigue failure that can be affected by the environment are thermal and corrosion fatigue. Fluctuating thermal stresses at higher temperatures can cause thermal fatigue since dislocation motion is more manageable at high temperatures and yield strength decreases. Corrosion failure is a result of cyclic stresses and chemical reactions that reduce the fatigue life. Corrosion can form small pits that act as crack initiation sites, and the crack propagation process is enhanced due to corrosion [14]. The formation of intrusions and extrusions is also facilitated by corrosion [32]. Protective coatings, careful material selection, and reducing the corrosiveness of the environment are some methods to reduce corrosion damage [14]. Humidity levels also have a significant impact on fatigue life. Studies suggest that humidity can affect crack initiation. The fatigue capacity is better in a vacuum than in air, and areas with dry air have better fatigue resistance than humid areas or those near the sea [33].

2.2.5. High Cycle Fatigue

In high cycle fatigue, a relatively large number of cycles is required to fracture. The stress levels are lower, and deformations will be completely elastic. The components exposed to high cycle fatigue are expected to achieve long lifetimes. High cycle is considered for fatigue lives higher than about 10^4 cycles. Low Cycle fatigue (LCF) is considered if the number of cycles is below 10^4 cycles- This approach of high cycle fatigue leads to determining an "endurance limit" that indicates the stress amplitude level at which a material is anticipated to have an unlimited fatigue life, assuming no pre-existing flaws [34].

2.3.   Fracture Mechanics

Generally, there are two techniques used for estimating remaining fatigue life: fracture mechanics and fatigue damage accumulation. Fracture mechanics is a theoretical method using equations for fatigue crack growth rate. This method assumes that a crack already is present, and the component is currently in the crack growth phase. However, in most cases it is desired to calculate the remaining fatigue life before the crack growth phase [29]. Therefore, the method of fatigue damage accumulation is usually more favourable, and is the method utilized in this thesis.

2.4.   Fatigue Damage Accumulation

Fatigue damage accumulation can be utilized when the current damage state is unknown and is based on load-time histories and S-N curves. For the calculation of fatigue damage accumulation, Palmgren-Miner's rule is the only accepted method in most industry standards [7, 8]. In recent years, several nonlinear models have been proposed but none of them are accepted in standards partly due to complexity of the models, lack of empirical data for real life situations, and consistency with existing standards.

2.4.1. S-N Curves

Laboratory tests are used to determine the fatigue properties of a material. For instance, rotating-bending tests. Here the specimen I subjected to alternating tension and compression of equal magnitude, while bending and torsion is imposed. The tests are usually performed at a relatively high maximum stress, close to the yield limit. The tests are repeated many times on several specimens at decreasing stress amplitude. Each time the number of cycles is counted and plotted in a diagram as nominal stress versus the logarithm of the number of cycles to failure [14].

This is known as the S-N curves. The S-N curves for steel, which is relevant for this thesis, show that a higher maximum stress level, reduces the number of cycles to failure. Some ferrous materials, such as steel, have a fatigue limit. Here, the S-N curve becomes horizontal at higher N values, below which failure will never occur for an infinite number of cycles under constant amplitude loading. Most nonferrous alloys, like aluminium, do not have a lower fatigue limit, and here the S-N curve will continue downwards. Steel has Body centred cubic (BCC) microstructure, while aluminium has a Face centred cubic (FCC) microstructure. BCC structures typically lead to harder and less ductile materials. The fatigue strength, which is the stress level that causes failure at a specific number of cycles, can be determined from the S-N curve. Conversely, the fatigue life is the number of cycles to failure at a particular stress level [14].

Unfortunately, there will always be some scatter in the S-N curves, which can lead to uncertainties in fatigue design. Some parameters are impossible to keep equal for every specimen, such as fabrication, metallurgical variables, and specimen alignment in the test apparatus [14].

2.4.2. Palmgren-Miner's Rule

The Palmgren-Miner's rule or the linear damage rule is widely used as the standard for fatigue damage accumulation under cyclic loading. The rule assumes that fatigue damage is equal to accumulated cycle ratio. The Palmgren-Miners Rule is expressed by equation (1).

$$D = \sum \frac{n_i}{N_i} \leq 1$$

(1)

Where D is total damage, and D = 1 corresponds to failure. N_i is the number of cycles to failure at the i-the constant stress level cycle, and n_i is the number of applied cycles to the structure. The number of cycles to failure is according to the relevant S-N curve [35].

Miner's rule considers fatigue damage accumulation mechanism to be linear, although the process is highly nonlinear. Despite it being a widely used rule for design, it does not account for the sequence of the load, the load level independence or load interaction effects due to crack slip plasticity [36]. The loading sequence is an important factor to consider in fatigue life assessments and will give higher accuracy in fatigue life prediction. Loading sequence effect is essential in structures subjected to variable amplitude loading (VAL), which is relevant for the railway bridge considered in this thesis. The condition of the applied loads, the geometry and material properties of the component, the environment and microstructure of the component all affects the magnitude of the loading sequence effect [37].

As the Palmgren-Miner's rule fails to consider these factors, experimental tests have shown that there are deviations between the actual lifetime and predicted lifetime according to Palmgren-Miners Rule. The rule has shown to often be conservative for low-to-high loading sequences, and non-conservative for high-to-low loading sequences [35]. Additionally, Schijve also mentions the rules lack of ability to consider the damage caused from stress below the endurance limit [29].

This leads to high design safety factors in combination with a S-N curves with low probability of failure in the Eurocode and DNVGL [38]. DNV assumes a coefficient of variation (CoV) of 0,3 for Palmgren-Miner's rule, where CoV is defined as standard deviation divided by mean value. The result is a conservative design [39].

2.4.3. Alternative Models

Several alternative nonlinear models have been proposed to overcome the shortcomings of Miners rule. The aim is to maintain the simplicity of the Miners rule, while accounting for variable amplitude loading. An example is the double linear damage rule (DLDR) proposed by Manson, which applies two linear damage rules categorized as Phase I and Phase II [40]. A model only based on the common S-N curves, while incorporating a concept of damage transfer to consider loading sequence was proposed by Aeran in 2017. The damage model only depends on the parameter N_f, found in the S-N curves, and the concept of transfer depends on stress amplitudes related to the curves. The model shows better agreement with experimental results than earlier proposed models [41].

This thesis employs a nonlinear damage function recently developed by Fredrik Bjørheim, which relies solely on the S-N curve and does not require parameter adjustments. The foundation for this model is a theory of S-N fatigue damage envelope proposed by Pavlou [38]. In this theory, the isodamage lines are not straight lines, but curved lines converging both at ultimate stress point, and at the knee point. The S-N curve has a damage state of D=1, whereas both axes have a damage state of D=0. Following, one should assume that no damage can be accumulated when the number of cycles is equal to zero, and vice versa if the stress amplitude is below the fatigue limit [42].

The proposed damage function is presented in equation (2).

$$D = \left(\frac{n}{N_f}\right)^{q(\sigma,m)}$$

(2)

Here, the exponent q(σ,m) is a parameter affected by stress amplitude and material properties expressed as equation (3).

$$q(\sigma,m) = \frac{a(S_u - S_e)}{S_i - S_e}$$

(3)

For multistage loading, total damage can be calculated by equation (4).

$$D_{tot} = \left(\left(\left(\left(\cdots\left(\left(\left(\frac{n_1}{N_{f1}}\right)^{\frac{q(\sigma_1,m)}{q(\sigma_2,m)}} + \frac{n_2}{N_{f2}}\right)^{\frac{q(\sigma_2,m)}{q(\sigma_3,m)}} + \frac{n_3}{N_{f3}}\right)^{\frac{q(\sigma_3,m)}{q(\sigma_4,m)}} + \cdots\right.\right.\right.\right.$$

(4)

$$\left.\left. + \frac{n_{k-1}}{N_{f(k-1)}}\right)^{\frac{q(\sigma_{k-1},m)}{q(\sigma_k,m)}} + \frac{n_k}{N_{f(k)}}\right)^{q(\sigma_k,m)}$$

3. Bridge Specifics - Saulidelva Bridge

The railway bridge over Saulidelva is located on Bergensbanen in Norway, on the section between Trolldalen and Ål. The bridge was built in 1906 and is a single-track truss bridge. The speed limit over the bridge is 110 km/h for passenger trains, and 100 km/h for freight trains. Train passings per year is estimated to be 3219 passenger trains, and 4480 for freight trains. The data is extracted from a report provided by Bane NOR, from 2018. The numbers are based on the average of the maximum number of trains in the years 2011, 2014 and 2015 [43].

Figure 3.1: Saulidelva bridge. Used with permission from Bane NOR[43]

3.1. Model of Bridge Structure

Saulidelva bridge is a single span riveted truss bridge with a length of 24m and width of 3m. the total height of the steel structure is 2,7 m. The main span consists of two support walls on each side, diagonal braces in the bottom, and transverse and longitudinal beams lying on main truss. Main sections of bridge shown in Figure 3.2. The cross sections consist of simple and built-up L-profiles and I-profiles [43]. Material properties are given in Table 3.1.

Table 3.1: Material Properties [43]

Material Parameters	Value
Unity weight γ	77 [kN/m^3]
Yield strength f_y	220 [N/mm^2]
Tensile strength f_u	350 [N/mm^2]
Modulus of elasticity (Young's modulus) E	2.1 $*10^5$ [N/mm^2]
Shear modulus G	0.8$*10^5$ [N/mm^2]
Poisson's ratio ν	0.3
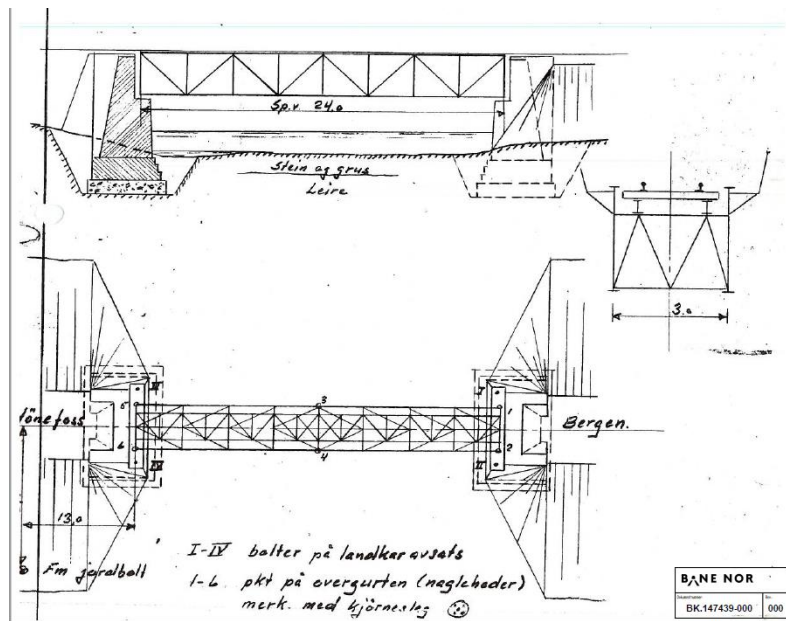
Figure 3.2: Sections of bridge. Used with permission from Bane NOR[43].

The bridge is simply supported in all corners of the model. Location of the nodes illustrated in Figure 3.3.

- Node 101 – zero translation.
- Node 102 – translation in x-direction.
- Node 103 – translation in y-direction.
- Node 104 – translation in x- and y-direction.
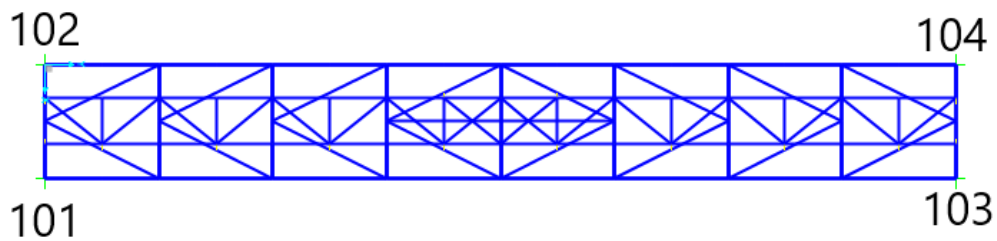
Figure 3.3: Identification of nodes in bridge model.

3.1.1. Simplifications

The model designed in SAP2000 for analysis is based on node coordinates and geometry of model from report provided by Bane NOR. A slight horizontal curvature of the bridge is neglected in the model, as the contribution is minimal. Some elements are considered not to affect the load distribution as they are used only for stabilizing longitudinal beams against torsion and are not included in model. The self-weight of the steel bridge, weight of the railway tracks and wooden railroad ties is not included. For the built-up cross sections, some simplifications were made, and parts that were not considered to be critical were neglected. It is assumed that the all the elements of the bridge are connected at the centreline of the cross sections.

4. Methodology

4.1. Approach

In this thesis a finite element analysis was performed of the railway bridge using the FEM software SAP2000. The model of the railway bridge was subjected to variable amplitude loading (VAL) representing the historic railway traffic. The different load models used for both passenger trains and freight trains are presented in section 4.3. The result of the analysis is used to calculate stress ranges for each model which were used for the fatigue assessment. For the fatigue assessment 2 S-N based methods were utilized; the Palmgren-Miner rule and a nonlinear fatigue model proposed by Fredrik Bjørheim. The stress-life approach or the S-N methods is suitable for structures and components subjected to high cycle fatigue and variable amplitude loading. Results from both assessments were compared and discussed regarding strengths and weaknesses of both assessment models.

4.2. Eurocode Standards

Several standards are relevant for fatigue design. In this thesis, 3 Eurocode standards were considered. In the standards it is provided guidance and regulations for structural design to withstand fatigue, design of bridges, and traffic loads on bridges.

4.2.1. NS-EN 1993 – 1-9 – Design of steel structures - Fatigue

The NS-EN-1993-1-9 outlines necessary procedures and criteria for evaluating fatigue in steel structures and their constituent parts. It is applicable to all types of structural steel, provided that the steel has the appropriate corrosion protection and is maintained to meet the required service life. This means that the NS-EN-1993-1-9 covers structures in normal atmospheric conditions, with only mild corrosion[44].

NS-EN 1993-1-9 contains a table of 10 detail categories, table 8.1, where the number of the category represents stress range at 2 million cycles with constant amplitude. For riveted bridge connections detail category 71 is considered conservative and a safe lower boundary. When the remaining fatigue life cannot be demonstrated with this curve, detail category 85 can be utilized as it gives a better fatigue strength for stress ranges between 20 and 80 MPa [45]. These two detail categories were utilized in this thesis to calculate total damage according to Palmgren-Miner's rule and the nonlinear fatigue damage model in this thesis.

Each of the detail categories corresponds to a S-N curve shown in figure 7.1 in the NS-EN 1993 1-9 that is calculated from test data from a representative test specimen subjected to various stress levels.

Section 7.1 (3) describes how to calculate the capacity N_R for VAL according to the stress ranges and is expressed by equation (5) and (6).

$$\Delta\sigma_R^m N_R = \Delta\sigma_C^m * 2 * 10^6 \; with \; m = 3 \; for \; N \leq 5 * 10^6 \tag{5}$$

$$\Delta\sigma_R^m N_R = \Delta\sigma_D^m * 5 * 10^6 \; with \; m = 5 \; for \; 5 * 10^6 \leq N \leq 10^8 \tag{6}$$

Where $\Delta\sigma_C$ corresponds to the relevant detail category, $\Delta\sigma_R$ is the actual stress range, and $\Delta\sigma_D$ corresponds to the fatigue limit under constant stress level. $\Delta\sigma_D$ is given by equation (7).

$$\Delta\sigma_D = 0,737 * \Delta\sigma_C \tag{7}$$

4.2.2. NS-EN 1991-2 – Traffic loads on bridges

NS-EN 1991-2 defines the traffic loads that must be considered for various design limit states when designing bridges. The code provides the imposed loads for different categories of standardized vehicles or trains, depending on the type of bridge and traffic [44].

Five load models for railway bridges are defined in NS-EN 1991-2[46]:
- Load model 71(and Load Model SW/0 for continuous bridges) – represent normal rail traffic on mainline railways. This model is relevant for this thesis.
- Load Model SW/2 represent heavy loads.
- Load Model HSLM represent loading from passenger trains at speeds exceeding 200km/h.
- Load Model "unloaded train" to represent effect of unloaded train.

The Annex D of the NS-EN 1991-2 describes the basis for fatigue assessment of railway bridges and contains the load models Type 5-8 for freight trains described in section 4.3.1.
Part D.1 of the Annex describes assumptions for fatigue actions. D.1 (2) states that a dynamic factor Φ should be included to take account for average effect for assumed more than 100 years life of structure.

4.2.3. NS-EN 1993-2 – Steel Bridges

This section of the Eurocode specifies the regulations and design principles for steel bridges and the components of composite bridges. Included in this section are also the recommended load models for design limit states and techniques outlined in NS-EN 1991-2[44].

4.3. Load Models

Bane NOR have performed a fatigue damage assessment using a combination of self-defined load models developed through a collaboration with NTNU [43] and load models found in the NS-EN 1991-2[46] to represent the historical load on the bridge. The self-defined load models consisted of 4 train sets for freight trains, labelled LMF1-LMF4, and 5 train sets for passenger trains labelled LMP1-LMP4. For the load models form the NS-EN 1991-2, Type 5-8 were used. The load models are further described in section **Feil! Fant ikke referansekilden.** and **Feil! Fant ikke referansekilden.**. The historical load was divided into 4 time periods, P1-P4, with corresponding models. Each period is described in Table 4.1.

Table 4.1: Description of time periods

	Period	Years	Train
P1	1906-1930	24	LMF2
			LMP2
P2	1930-1960	30	LMF3
			LMP3
P3	1960-1985	25	LMF4
			LMP4
P4	1985-2023	38	Type 5
			Type 6
			Type 7
			Type 8
			LMP5

4.3.1. General Assumptions

The load models LMF1 and LMP1, were not relevant for this analysis as they represent the railway traffic on the bridge pre. 1900, and the bridge was built in 1906. On the specific railway, the maximum speed limit of freight trains is set 100 km/h and maximum speed limit for passenger trains is 110 km/h [43]. Therefore, the speed of freight train Type 7 was adjusted from120 km/h to 100 km/h, and speed of passenger trains LMP4 and LMP5 was adjusted from respectively 130 km/h and 160 km/h to 110 km/h.

Bane NOR has defined desired total weight for each type of train. Following the number of wagons should be adjusted for both passenger trains and freight trains. The freight trains total weight should be approximately, but bigger than, 750 tonnes, and equally for each passenger train, the total weight should be approximately, but bigger than, 225 tonnes. In Table 4.2 the number of passings per year, the adjusted number of wagons, total weight, and speed in km/h is given for each load model.

Table 4.2: General data for load models

Train	Frequency	Wagons	Total Weight (tonnes)	Speed(km/h)
LMF2	4480	26	750	70
LMF3	4480	21	777	70
LMF4	4480	18	756	80
Type 5	949,76	5	825	80
Type 6	1630,72	11	759	100
Type 7	1084,16	7	780	100
Type 8	815,36	14	780	100
LMP2	3219	4	254	90
LMP3	3219	3	266	100
LMP4	3219	3	270	110
LMP5	3219	2	244	110

4.3.2. Load models from NS-EN 1991-2

In NS-EN 1991-2 Annex D.3 (1)- standard and light traffic mixes, the load models Type 5 - 8 are described. The load models were used to represent railway traffic from 1985-2023, and future traffic. According to NS-EN 1991-2 Annex D.3 table D.1 for standard traffic mix with axles ≤ 225kN, the distribution of the train models Type 5-8 in the time period is as given in Table 4.3. The total number of passings each year for freight trains is 4480, and the total number is distributed on the train models Type 5-Type 8.

Table 4.3: Distribution for Type 5 – Type 8

Train	Distribution	Passings per year
Type 5	21,20 %	949,76
Type 6	36,40 %	1630,72
Type 7	24,20 %	1084,16
Type 8	18,20 %	815,36

4.4. Modelling and Structural Analysis in SAP2000

The Saulidelva bridge was modelled and analysed in the FEM-programme SAP2000. This is a powerful and user-friendly tool for structural engineering developed by Computers and Structures Inc. The software can be used for modelling of everything from easy 2D structures to complex 3D geometry.

A 3D model of the bridge based on given data of nodes and elements provided by Bane NOR was designed in SAP200. Firstly, the outline of the model was drawn in a grid structure. The material properties listed in Table 3.1 were applicable throughout the entire bridge structure and assigned to each member in SAP2000. In total the bridge consists of 32 different cross sections that was designed and assigned to the appropriate members of the truss. Figure 4.1 of the model shows the different cross sections of the bridge represented by different colours.
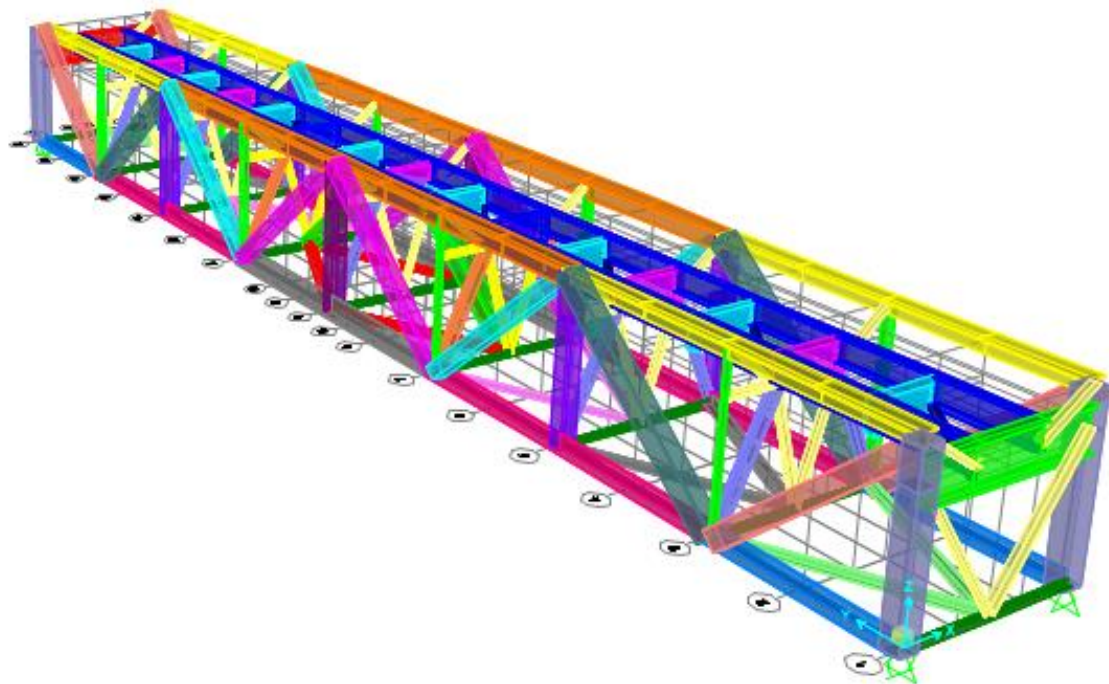
Figure 4.1: Bridge model showing cross sections.

Two versions of the model were analysed. First the bridge was modelled and analysed with some of the complicated built-up sections simplified. As designing the simplified cross sections directly in the software was faster and easier, it was interesting to see the effects this simplification would have on the results. Thereafter all the cross sections were redefined in SAP2000 as "general" cross sections with the accurate section properties and analysed once more. The simplified model is hereafter referred to as Case A, and the accurate model is referred to as Case B. Tables of net area, torsional constant, and second moment of area about z-z axis and y-y axis for the respective cross sections in Case A and Case B are found in the Appendix A.

18

The 11 load models described in section 4.3 were defined as vehicles in the software, along with the paths that the load was applied to. Figure 4.2 shows path 1 and path 2 marked in red and yellow.
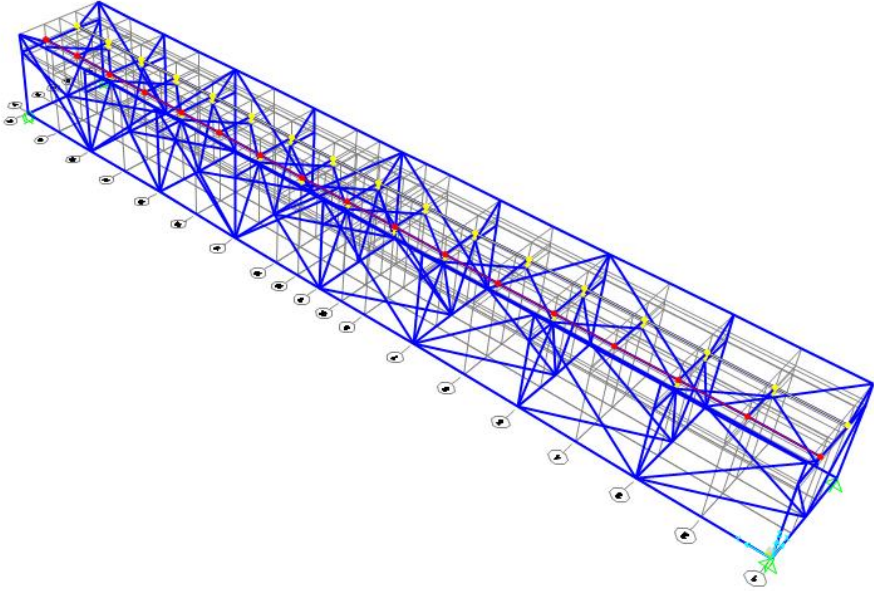
Figure 4.2: Bridge model showing paths.

For both Case A and Case B, each of the load cases were applied to the load paths with a scale factor 0.5. A scale factor of 0.5 indicates that 50% of the total weight of the load case were applied to each path.

The analysis for each model returned graphs of forces and stress envelopes with maximum and minimum stress for each member of the bridge, so that the most critical member could be identified and evaluated.

4.5.   Fatigue Calculations

An evaluation of the analyses determined the most critical member and point of the bridge. Only the member generally subjected to the highest stress range for all the train sets were chosen for fatigue calculations. For nearly all the load models, the most critical members were the main girder in the middle of the span, referred to as frame 111-1, shown in Figure 4.3. The level of stress is assumed to be highest at the top of the member, and the damage were calculated in this point, demonstrated in Figure 4.4. It should be emphasized that the cross section of frame 111-1 is not one of the cross sections that were simplified in Case A but was defined with accurate section properties for both cases.
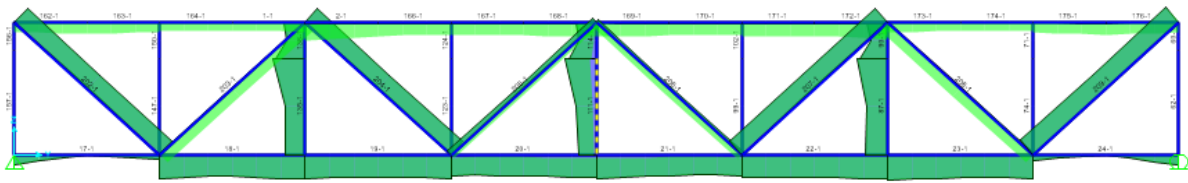
Figure 4.3: Stress diagram of main girders.
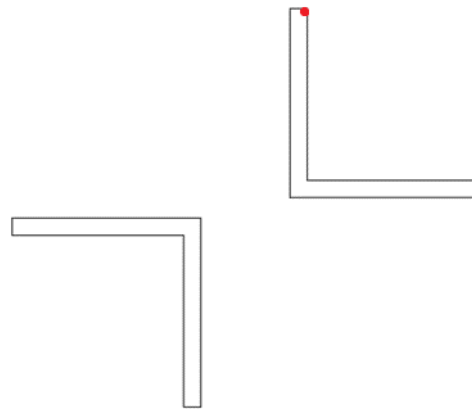
Figure 4.4: Cross section of frame 111-1.

Tables of axial forces and moment from the relevant point of the member for each train model were extracted from SAP2000 to use for further calculations. All load-time histories for Case A and Case B can be found in the Appendix B.

4.5.1. Rainflow Cycle Counting

For determining number of stress cycles, the rainflow counting command in the programming platform MATLAB were utilized. Rainflow counting method is used for determining the number of fatigue cycles in a load-time history. Rainflow counting is a widely accepted counting method for both DNV and NS-standards[7, 8]. For this thesis, the ASTM standard E1049 procedure for rainflow counting was followed. Cycle counting for real life variable amplitude loading can be difficult to consider as it often appears to have a random nature. The method of rainflow counting makes it easier to determine the number of cycles and reduces the data for time history to the minimum required data to preserve necessary information for damage [47].

A load-time series consists of peaks and valleys, and the load range is the difference between a peak and the adjacent valley. often, load-time datasets contain very small oscillations that does not contribute to damage. These small oscillations are typically filtered out and neglected as it will take up a large amount of data storage and increase the analysis duration [47].

The output data contains the number of cycles for each stress range, sorted in order of decreasing stress range. The cycle count data will consist of half cycles and full cycles, assuming no stress ranges are exactly equal. Normally, the results are presented in histograms, where the cycles are grouped into "bins". Figure 4.5 and Figure 4.6 respectively shows the load-time history and histogram of grouped bins The size of the bins is appropriately specified, and the stress range is usually set to the upper end of the bin [47]. However, bins were not used further in the calculation of this thesis as the output data were of relatively small size.
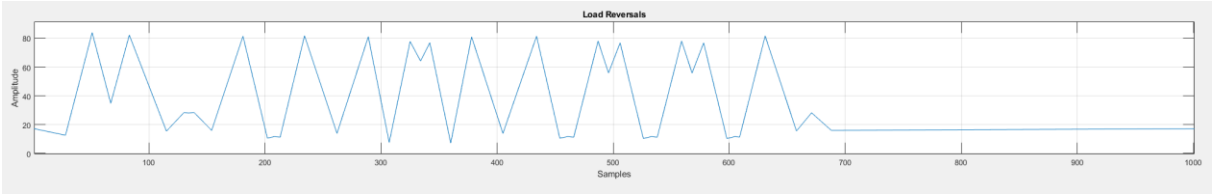
Figure 4.5: Load reversals for load model Type 6.
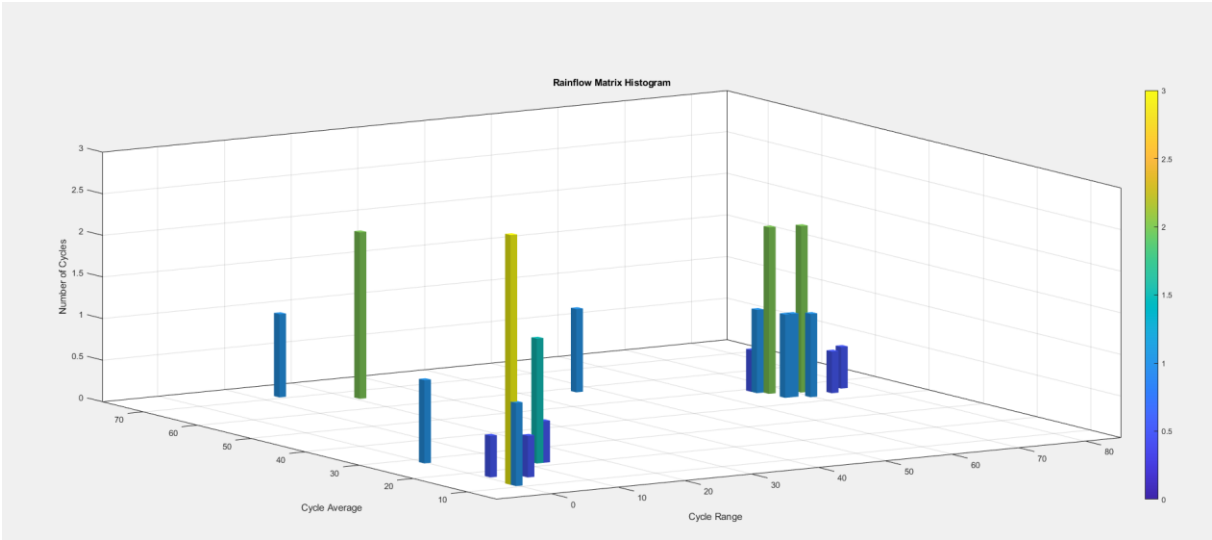
Figure 4.6: Bins for stress ranges load model Type 6.

4.6. Palmgren-Miners Rule

The cycle count and stress range for Case A and Case B were extracted from the rainflow matrix in MATLAB to calculate the damage for each model. The damage was calculated using two different S-N curves and three different partial factors for comparison of the results. The NS-EN 1993-1-9 does not include a specific detail category for riveted connections, but a study conducted by Taras & Greiner, states that S-N curve 71, $\Delta\sigma_C$=71 MPa, from the NS-EN 1993-1-9 could be utilized as it is considered a lower conservative limit for all riveted connections[45]. Damage calculation with the S-N curve $\Delta\sigma_C$=85 MPa were also considered, as it gives a more detailed categorization. This curve gives a higher fatigue strength for stress ranges between 20 MPa and 80 MPa [43, 45].

Partial factor for fatigue strength of γ_{Mf}=1.0, γ_{Mf}=1.35 and γ_{Mf}=2.0 were considered for calculations. The partial factor γ_{Mf}=2.0 is the recommended value for high consequence of failure for safe life method in the national annex of NS-EN 1993-1-9, table NA.3.1. Partial factor of γ_{Mf}=1.35 is the recommended value in table 3.1 in the Eurocode.

4.7. Non-Linear Fatigue Damage Model

The calculations for the nonlinear fatigue damage model were performed in MATLAB. Only the model for Case B were considered, and only for $\Delta\sigma_C$=85 MPa and γ_{Mf}=1.0 were used for the data input. The data was separated into four sets, each dataset representing the four time periods and the respective load models. A single dataset consisted of the relevant stress ranges, number of cycles, and capacity for the relevant load models in the actual period. One data set represented the trains passings for one day, the number of times each data set runs is corresponding to the number of days in the relevant period. It was assumed that passenger trains pass the bridge, at daytime and freight trains at night. Based on the total number of passing trains for a year, it was assumed that nine passenger trains pass every day, and 12 freight trains every night. The sequence for the datasets were organised accordingly.

Thereafter, MATLAB calculated the accumulated damage for each cycle. This was executed with a for-loop going through each line of the dataset. Another for-loop repeated the process for each day, until the end of the period, or when damage D=1. If damage has not occurred by the end of the period, the loop continued to the next period. The output was a matrix showing accumulated damage over time. MATLAB script can be found in Appendix D.

22

4.7.1. Cases for Time Period 4

For the nonlinear fatigue model, three different cases for the fourth period were considered. As there are four different types of freight trains running in the same period, the order of the freight trains was different for each case. For Case 1 and 2, the order of the trains was reversed. For the Case 3, the sequence was arranged to alternate every other day. This was done to represent a more random load history. Following, Case 3 represents a span of two days in Period 4, while Case 1 and Case 2 represents one day. The sequence in each case for Period 4 are presented in Table 4.4 below.

Table 4.4: Sequence for Case 1-3 in Period 4

Case 1	9xLMP5 - 3xType 5 - 4xType 6 - 3xType 7 - 2xType 8
Case 2	9xLMP5 - 2xType 8 - 3xType 7 - 4xType 6 - 3xType 5
Case 3	9xLMP5 - 3xType 5 - 4xType 6 - 3xType 7 - 2xType 8 - 9xLMP5 - 2xType 8 - 3xType 7 - 4xType 6 -3xType 5

5. Calculations and Results

In this chapter, the calculations and results for both the Palmgren-Miner's rule and the non-linear fatigue damage model is presented. Hand calculations were mainly based on formulas and tables form the NS-EN 1993 1-9, NS-EN 1991-2, and NS-EN 1993.2.

5.1. Palmgren- Miner's Rule

5.1.1. Calculations

Palmgren-Miner's rule is defined by equation (1).

Here, n is the number of cycles for each stress range, and N is the capacity according to the relevant S-N curve and stress range.

Firstly, the dynamic factor to account for average effect for over 100 years of life of the structure must be calculated and is expressed in NS-EN 1991-2 D.1 (2) by equation (8) as follows:

$$\Phi = 1 + \frac{1}{2}\left(\varphi' + \frac{1}{2}\varphi''\right) \tag{8}$$

Where φ' and φ'' respectively is expressed by equation (9) and (10) respectively.

$$\varphi' = \frac{K}{1 - K + K^4} \tag{9}$$

$$\varphi'' = 0{,}56e^{-\frac{L^2}{100}} \tag{10}$$

Where K is equal to:

$$K = \frac{v}{47{,}16L^{0{,}408}} \; for \; L > 20m \tag{11}$$

For the critical members considered in this thesis, L = 24m according to table 6.2, Case 5.1 in NS-EN 1991-2, stating that the determinant length for main girders should be equal to the span in main girder direction. The dynamic factor for each load model is calculated in

Table 5.1.

Table 5.1 Dynamic factor for load models

Train	v (m/s)	L (m)	K	φ'	φ''	Dyn. factor Φ
LMF2	19,44	24	0,1127	0,1270	0,0018	1,0640
LMF3	19,44	24	0,1127	0,1270	0,0018	1,0639
LMF4	22,22	24	0,1288	0,1478	0,0018	1,0744
Type 5	22,22	24	0,1288	0,1478	0,0018	1,0744
Type 6	27,78	24	0,1611	0,1918	0,0018	1,0964
Type 7	27,78	24	0,1611	0,1918	0,0018	1,0964
Type 8	27,78	24	0,1611	0,1918	0,0018	1,0964
LMP2	25,00	24	0,1450	0,1694	0,0018	1,0852
LMP3	27,78	24	0,1611	0,1918	0,0018	1,0964
LMP4	30,55	24	0,1771	0,2150	0,0018	1,1079
LMP5	30,55	24	0,1771	0,2150	0,0018	1,1079

For calculation of the stress ranges, the cross-sectional properties of frame 111-1 listed in Table 5.2 is necessary. The axial force and bending moment for each load model were given in tables from the FEM analysis.

Table 5.2 Cross-sectional properties frame 111-1

A (Net Area)	3800 mm^2
I$_y$	14500000 mm^4
I$_z$	10610000 mm^4
y$_{max}$	116 mm
z$_{max}$	36 mm

Axial stress is calculated by (12).

$$\sigma_{axial} = \frac{P}{A} \tag{12}$$

Bending stress about y-y axis and z-z axis is calculated by (13) and (14) respectively.

$$\sigma_{b,y} = \frac{My_{max}}{I_y} \tag{13}$$

$$\sigma_{b,z} = \frac{Mz_{max}}{I_z} \tag{14}$$

The total stress for each stress range is the sum of the contribution from axial force and bending times the dynamic factor expressed by equation (15).

$$\sigma_{total} * \Phi = \left(\sigma_{axial} + \sigma_{b,y} + \sigma_{b,z}\right) * \Phi \qquad (15)$$

The number of cycles for every stress range for each load model was then analysed using rainflow cycle counting in MATLAB.

As stated in chapter 4.5.2, $\Delta\sigma_C$=71 MPa and $\Delta\sigma_C$=85 MPa will be utilized for the fatigue calculations. According to equation (7), $\Delta\sigma_D$ is calculated as follows.

$$\Delta\sigma_D = \Delta\sigma_C * 0{,}737 = 71 * 0{,}737 = 52{,}33 \; MPa \; for \; \Delta\sigma_C = 71MPa$$
$$\Delta\sigma_D = \Delta\sigma_C * 0{,}737 = 85 * 0{,}737 = 62{,}65 \; MPa \; for \; \Delta\sigma_C = 85MPa$$

$\Delta\sigma_D$ is used to calculate the lower fatigue limit $\Delta\sigma_L$ for VAL according to NS-EN 1993. 1-9 7.1 (3):
$$\Delta\sigma_L = \Delta\sigma_D * 0{,}549 = 52{,}33 * 0{,}549 = 28{,}73 \; MPa \; for \; \Delta\sigma_C = 71MPa$$
$$\Delta\sigma_L = \Delta\sigma_D * 0{,}549 = 62{,}65 * 0{,}549 = 34{,}39 \; MPa \; for \; \Delta\sigma_C = 85MPa$$

If the stress range included the partial factor is below the lower fatigue limit, $\Delta\sigma_L$, the contribution is neglected.

Table NA.3.1 in NS-EN 1993 1-9 states that for structures with high consequence in case of fatigue failure and considering safe life method, γ_{Mf} = 2,0 should be considered. For comparison calculation using the recommended partial factor from the NS-EN 1993-1-9 Table 3.1, γ_{Mf} = 1.35, and a partial factor of γ_{Mf} = 1.0 was also considered. Partial factor γ_{Ff} = 1,0 were found in NA.9.3(1) in NS-EN 1993-2. $\Delta\sigma_R$ is calculated by equation (16).

$$\Delta\sigma_R = \Delta\sigma_i * \gamma_{Mf} * \gamma_{Ff} \qquad (16)$$

Where $\Delta\sigma_i$ is the stress ranges from analysis.

$\Delta\sigma_R$ was used to calculate N_R from NS-EN 1993. 1-9 7.1 (3) which is expressed by equation (5) and (6).

Here following conditions applies:

- If $\Delta\sigma_R \geq \Delta\sigma_D$; N_{RA} is suitable.

- If $\Delta\sigma_L \leq \Delta\sigma_R \leq \Delta\sigma_D$; N_{RB} is suitable.

- If $\Delta\sigma_R \leq \Delta\sigma_L$; $N_R = \infty$; No fatigue.

The total number of cycles for each stress range, n, can be calculated by multiplying the number of cycles by the number of trains for each year for the relevant load model. Data for the number of trains per year is given in table 4.2.

5.1.2. Results

The current total damage results per 2023 for Case A and Case B with the respective S-N curves and partial factors are presented in Table 5.3 and Table 5.4.

Table 5.3 Current total damage results for Case A

Δσc=85		Δσc=71	
γ_{mf}	D	γ_{mf}	D
1,00	0,547454591	1,00	1,00444576
1,35	1,509155079	1,35	2,6668093
2,00	5,213773095	2,00	9,14376553

Table 5.4 Current total damage results for Case B

Δσc=85		Δσc=71	
γ_{mf}	D	γ_{mf}	D
1,00	0,51781447	1,00	0,9569643
1,35	1,41393402	1,35	2,51904281
2,00	4,89746513	2,00	8,54910718

Additionally, it was calculated when damage is estimated to occur according to Miner's rule for Case B, S-N curve 71 and 85, and partial factor γ_{Mf} = 1.0. Results presented in Table 5.5. Detailed results for both current damage and final fracture are found in the Appendix C.

Table 5.5 Results for expected failure for Case B

S-N Curve	Lifetime (years)	Year
71	119	2025
85	160	2066

5.2. Non-Linear Fatigue Damage Model

The stress ranges, number of cycles, and capacity calculated for the Case B for Palmgren-Miner's rule described in chapter 5.1.1, were used for input in MATLAB. The output was current damage given the time parameters. Current damage per 2023 is presented in Table 5.6. Additionally, by changing parameters in the code, the approximate time of failure for each case was estimated. As the current damage D does not directly give information about remaining lifetime, it is important to simulate until D=1 to be able to evaluate time of failure. The results for estimated time of failure are presented in Table 5.7. In the Appendix E, the graphs of damage for Case 1-3 can be found.

Table 5.6 Current damage per 2023

Case	Damage
Case 1	4.70989240487073e-10
Case 2	4.70978099401385e-10
Case 3	4.70983674034695e-10

Table 5.7 Time of failure for Case 1-3

	Times (days)	Damage	Lifetime (years)	Year
Case 1	59515	0.999389244964660	163,05	2069
	59516	1.00030302550666		
Case 2	59515	0.999390562995743	163,05	2069
	59516	1.00030434545460		
Case 3	59516	0.999389908243754	163,06	2069
	59517	1.00121827960186		

6. Discussion

6.1. Case A and B

The results display interesting differences and similarities for the two fatigue calculation models and the different cases. Firstly, the results for current damage according to Palmgren-Miner's rule for Case A and Case B shows that simplifying the cross sections lead to reduction in the fatigue capacity for the cross section being assessed. By simplifying the cross sections, moment of inertia is also reduced which affects the resistance to bending. Consequently, the stiffness of the model decreased, and the applied forces is less evenly distributed among the members. Through the process of modelling the structure, it was learned that defining every cross section as "general cross sections" with accurate section properties, but without the correct geometry, could be done easily by importing tables of cross section data. However, the visual inaccuracy that follows may be problematic as all "general" cross sections are considered as rectangles by the programme. By default, the points that are measured are located at the corners of the cross section. The frame 111-1, which was assessed here, did in fact not exist in the corners of the general rectangle, and the location of the points had to be adjusted. Alternatively, every cross section would have to be modelled and defined with correct geometry and properties in SAP2000, which would have been a time-consuming process. Therefore, it was worth to investigate whether simplifying cross sections would have a significant effect on the result.

6.2. S-N Curves

For the damage results of Case A and B, the choice of S-N curves has a large impact. By shifting from S-N curve 71 to S-N curve 85, the damage D is reduced by a factor between 0,5 and 0,6. This is true for both case A and Case B and is in line with expected outcome. S-N curve 71 is generally considered a conservative lower boundary for all riveted connections. However, the S—N curve 71 have historically displayed increasingly scatter of test data results for the high and very high cycle ranges [45] This is highly relevant for this situation, an ageing railway bridge which is subjected to high cycle fatigue. Additionally, for old railway bridges some of the cross sections are custom fabricated. Following, using the same S-N curve for all riveted connections is equal to assuming that all of them behaves the same regardless of the differences in stress states around notches, which is a rather broad classification [45]. Calculations for S-N curve 85 gives a higher fatigue strength and possibly more representative results.

6.3.   Partial Factor γ_{Mf}

The use of partial factors also naturally affects the damage. By using the partial factor γ_{Mf}=2.0 recommended for safe life method, and the conservative S-N curve $\Delta\sigma_C$=71 MPa, Miner's rule gives that the element have exceeded almost 9 times its predicted lifetime. If one were presented this number for current damage without context, it can be viewed as a rather unsettling result, considering how many trains that still are passing the bridge daily. However, γ_{Mf}=2.0 and S-N curve 71 is recommended for design as it is a considered a lower conservative boundary. When designing for fatigue in structures with high consequence of failure, one should always consider the most conservative values for safety. It should be mentioned that when using partial factor γ_{Mf}=1.0, the results should still be considered safe as the design S-N curves found in the standards are already conservative.

6.4.   Linear vs. Nonlinear Model

Comparing the results for current damage for Case B using $\Delta\sigma c$ = 85 and γ_{mf} = 1.0, and results for the nonlinear model, there is a significant deviation in the results. While the linear model concludes that the current damage about 0.52, about half of its total lifetime, the nonlinear model shows that the current damage is minimal for all 3 cases. In the paper proposing the nonlinear model, it was stated that for D < 1, the model is not capable of giving a good fatigue damage estimation. However, the model should provide a good estimation of remaining fatigue life[38]. Therefore, the results of current damage for the nonlinear model may not be representative for the true damage, and it is important to simulate until D=1 to get a good estimation on remaining lifetime. Generally, Palmgren-Miner's Rule is conservative for low to high loading amplitudes, but better accuracy for loading histories with high stress levels [48, 49].

When it comes to lifetime estimation, both models showed more agreement when the same parameters were utilized. There is a deviation of tree years between the linear and the nonlinear model, which is a relatively small deviation. The reason for the similar answers could be that the loading-time history for the bridge were relatively simple and both models were able to accurately capture the critical features of the loading history. Additionally, the stress states that the nonlinear might have been able to account for, where Palmgren-Miner's rule did not, did not have a significant impact on predicted lifetime. Simplifications and assumptions that affected both models might also have contributed to similar results. Worth noting is that both models were simple to follow, and provided conservative results, which is preferable.

6.5. Case 1-3

For Case 1-3, the sequence of the trains displayed minimal effect for the results. Case 3, where the sequence of the trains was alternating every other day, calculations indicated damage only one day later than case 1 and 2. The reason for this is that the only difference between the 3 cases is Period 4. Among the load models, the variation in stress ranges is relatively small. In actual life, the trains are passing the bridge at a much more random sequence, and to assume that passenger trains pass the bridge at daytime and freight trains at night, is a broad generalisation. For future research it would be interesting to do a simulation where the trains in each time period pass the bridge in a random sequence, which is a more realistic representation of the loading history.

7. Conclusion

Regarding Case A and B, defining the cross sections with accurate section properties can be argued to be favourable, as it gives a significant higher fatigue strength. The procedure of defining the cross sections as "general" is simple and have a significant effect on the fatigue strength. As long as the points of measurements are specified correctly, Case B would give a more representative result and should be utilized.

The most conservative partial factors and S-N curves may not provide the most realistic representation of damage when calculating damage in existing structures. As mentioned, γ_{Mf}= 1.0 should still be considered safe as the design S-N curves are already conservative. However, in fatigue design one should consider the more conservative values and S-N curves to guarantee the safety.

For the linear vs. nonlinear model, as the loading history for this bridge were relatively simple, both models estimated similar lifetimes, but a significant deviation in estimated current damage. Palmgren-Miner's rule indicates remaining lifetime directly from current damage, but the nonlinear model must be simulated until damage. With a more complicated loading history, the linear and nonlinear models, as well as Cases 1-3, may show a more significant variation in results.

References

[1] "Evaluering av utmattingsberegninger - KU-027973-000_000_001," Johs Holt, Technical report - evaluation 2018.

[2] J. W. Fisher, *Fatigue and fracture in steel bridges. Case studies.* 1984.

[3] T. Guo, D. M. Frangopol, and Y. Chen, "Fatigue reliability assessment of steel bridge details integrating weigh-in-motion data and probabilistic finite element analysis," *Computers & Structures,* vol. 112, pp. 245-257, 2012.

[4] A. Fernández-Canteli, S. Blasón, J. Correia, and A. De Jesus, "A probabilistic interpretation of the Miner number for fatigue life prediction," *Frattura ed Integrita Strutturale,* vol. 8, no. 30, pp. 327-339, 2014.

[5] J. Kauzlarich, "The Palmgren-Miner rule derived," in *Tribology Series*, vol. 14: Elsevier, 1989, pp. 175-179.

[6] N. D. Adasooriya, "Fatigue reliability assessment of ageing railway truss bridges: Rationality of probabilistic stress-life approach," *Case Studies in Structural Engineering,* vol. 6, pp. 1-10, 2016.

[7] DNVGL-RP-C203, "Fatigue design of offshore steel structures," *Norwegian University of Science and Technology,* 2016.

[8] *Design of Steel Structures - Fatigue*, NS-EN-1993-1-9, 2010.

[9] R. I. Stephens, A. Fatemi, R. R. Stephens, and H. O. Fuchs, *Metal Fatigue in Engineering*, 2. ed. New York: John Wiley & Sons Inc., 2001.

[10] ASCE, "Committee on fatigue and fracture reliability of the committee on structural safety and reliability of the structural division. Fatigue reliability 1–4," *J. Struct. Div.,* vol. 108, no. ST1, pp. 3-88, 1982.

[11] G. Ersdal, J. V. Sharp, and A. Stacey, *Ageing and Life Extension of Offshore Structures: The Challenge of Managing Structural Integrity*. John Wiley & Sons, 2019.

[12] X. W. Ye, Y. H. Su, and J. P. Han, "A State-of-the-Art Review on Fatigue Life Assessment of Steel Bridges," *Mathematical Problems in Engineering,* vol. 14, p. 13, 2014, doi: https://doi.org/10.1155/2014/956473.

[13] R. Reed, J. Smith, and B. Christ, "Economic effects of fracture in the United States. Part 1. A synopsis of the September 30, 1982 report to NBS by Battelle Columbus Laboratories," National Measurement Lab., Washington, DC (USA). Center for Materials Science, 1983.

[14] W. D. Callister and D. G. Rethwisch, *Material Science & Engineering: An introduction*, 10. ed. Hoboken, NJ, USA: Wiley, 2020.

[15] T. Lassen and N. Recho, *Fatigue Life Analyses of Welded Structures*, 1. ed. London: ISTE, 2006.

[16] J. H. Kim, T. Chau-Dinh, G. Zi, and J. S. Kong, "The effect of compression stresses, stress level and stress order on fatigue crack growth of multiple site damage," *Fatigue & Fracture of Engineering Materials & Structures,* vol. 35, no. 10, pp. 903-917, 2012.

[17] E. J. France, "The Alexander L. Kielland Disaster Revisited: A Review by an Experienced Welding Engineer of the Catastrophic North Sea Platform Collapse," *Journal of Failure Analysis and Prevention,* vol. 19, pp. 875-881, 2019, doi: https://doi.org/10.1007/s11668-019-00680-4.

[18] P.-K. Foss, "The Office of the Auditor General´s investigation of the authorities work on the Alexander L. Kielland accident," The Office of the Auditor General of Norway, Investigation 09.03.21, 2021. [Online]. Available: https://www.riksrevisjonen.no/globalassets/reports/en-2020-2021/alexander-l-kielland-accident.pdf

[19] T. Moan. "The Alexander L. Kielland accident." https://www.ptil.no/contentassets/b8b5e5e0932f4d08a0c1a4d76c934e5f/alexander-l.-kielland-ulykken---torgeir-moan.pdf (accessed 15.02.23.

[20] M. Brumsen, "Case description: the ICE train accident near Eschede," *European business ethics cases in context: The morality of corporate decision making,* pp. 157-168, 2011.

[21] element.com. "5 Disasters Caused by Material fatigue and What We Learned From Them." https://www.element.com/nucleus/2016/5-disasters-caused-by-material-fatigue-and-what-we-learned-from-them (accessed 22.02.2023.

[22] T. Siwowski, "Fatigue assessment of existing riveted truss bridges: case study," *Bulletin of the Polish Academy of Sciences,* vol. 63, no. 1, pp. 125-133, 2015, doi: 10.1515/bpasts-2015-0014.

[23] W. Shütz, "A history of fatigue," *Engineering Fracture Mechanics,* vol. 54, no. 2, pp. 263-300, 1996, doi: https://doi.org/10.1016/0013-7944(95)00178-6.

[24] A. Wöhler, *Über die Festigkeits-Versuche mit Eisen und Stahl*, 1. ed. Berlin: Ernst & Korn, 1870.

[25] J. A. Ewing and J. Humfrey, "VI. The fracture of metals under repeated alternations of stress," *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character,* vol. 200, no. 321-330, pp. 241-250, 1903.

[26] O. Basquin, "The exponential law of endurance tests," in *Proc Am Soc Test Mater*, 1910, vol. 10, pp. 625-630.

[27] A. Palmgren, "Die lebensdauer von kugellagern," *Verfahrenstechinik,* vol. 68, pp. 339-341, 1924.

[28] M. A. Miner, "Cumulative damage in fatigue," 1945.

[29] J. Schijve, *Fatigue of Structures and Materials*, 2. ed. Delft, The Netherlands: Springer, 2008.

[30] O. Skoglund and J. Leander, "The impact of local geometry on the fatigue life of a welded structural detail," *ce/papers,* vol. 3, no. 3-4, pp. 641-646, 2019.

[31] A. Fatemi and N. Sanaei, "Analysis of the effect of surface roughness on fatigue performance of powder bed fusion additive manufactured metals," *Theoretical and Applied Fracture Mechanics,* vol. 108, 2020, doi: https://doi.org/10.1016/j.tafmec.2020.102638.

[32] R. E. Smallman and A. H. W. Ngan, *Modern Physical Metallurgy*, 8. ed. Elsevier Ltd., 2014.

[33] K. Tokaji, H. N. Ko, M. Nakajima, and H. Itoga, "Effects of humidity on crack initiation mechanism and associated S-N characteristics in very high strength steels," *Material Science and Engineering,* vol. 345, no. 1-2, pp. 197-206, 2003, doi: https://doi.org/10.1016/S0921-5093(02)00460-4.

[34] F. M. Shuaeib, K. Y. Benyounis, and M. S. J. Hashmi, "Material Behavior and Performance in Environments of Extreme Pressure and Temperatures," in *Materials Science and Materials Engineering*. Elsevier, 2017.

[35] K. Hectors and W. De Waele, "Cumulative Damage and Life Prediction Models for High-Cycle Fatigue of Metals: A Review," *Metals,* vol. 11, no. 2, p. 204, 2021. [Online]. Available: https://www.mdpi.com/2075-4701/11/2/204.

[36] A. Fatemi and L. Yang, "Cumulative fatigue damage and life prediction theories: a survey of the state of the art for homogenous materials," *International Journal of Fatigue,* vol. 20, no. 1, pp. 9-34, 1998, doi: https://doi.org/10.1016/S0142-1123(97)00081-9.

[37] K. A. Zakaria, S. Abdullah, and M. Ghazali, "A Review of the Loading Sequence Effects on the Fatigue Life Behaviour of Metallic Materials," *Journal of Engineering Science & Technology Review,* vol. 9, no. 5, 2016.

[38] F. Bjørheim, D. G. Pavlou, and S. C. Siriwardane, "Nonlinear fatigue life prediction model based on the theory of the S-N fatigue damage envelope," *Fatigue & Fracture of Engineering Materials & Structures,* Online vol. 45, no. 5, pp. 1480-1493, 2022, doi: https://doi.org/10.1111/ffe.13680.

[39] I. Lotsberg, *Fatigue design of marine structures*. Cambridge University Press, 2016.

[40] S. S. Manson, A. J. Nachtigall, C. R. Ensign, and J. C. Freche, "Further Investigation of a Relation for Cumulative Fatigue Damage in Bending," *Journal of Engineering for Industry,* vol. 87, no. 1, pp. 25-35, 1965, doi: 10.1115/1.3670753.

[41] A. Aeran, S. C. Siriwardane, O. Mikkelsen, and I. Langen, "A new nonlinear fatigue damage model based only on S-N curve parameters," *International Journal of Fatigue,* vol. 103, pp. 227-341, 2017, doi: https://doi.org/10.1016/j.ijfatigue.2017.06.017.

[42] D. Pavlou, "A deterministic algorithm for nonlinear, fatigue-based structural health monitoring," *Computer-Aided Civil and Infrastructure Engineering,* vol. 37, no. 7, pp. 809-831, 2022.

[43] "Beregningsrapport - Utmatting, Saulidelva - KU-027943-000_000," Johs Holt, Technical report 23.11.2018, 2018.

[44] M. Al-Emrani and M. Aygül, "Fatigue design of steel and composite bridges," Chalmers University of Technology, 2014.

[45] A. Taras and R. Greiner, "Development and Application of a Fatigue Class Catalogue for Riveted Componenta," *Structural Engineering International,* Online vol. 20, no. 1, pp. 91-103, 2018, doi: https://doi.org/10.2749/101686610791555810.

[46] *Laster på Konstruksjoner - Del 2: Trafikklast på bruer*, NS-1991-2, 2010.

[47] T. L. Anderson, *Fracture mechanics: fundamentals and applications*. CRC press, 2017.

[48] D. Pavlou, "Fatigue design challenges: Recent linear and nonlinear models," in *IOP Conference Series: Materials Science and Engineering*, 2019, vol. 700, no. 1: IOP Publishing, p. 012028.

[49] K. Rege and D. G. Pavlou, "A one-parameter nonlinear fatigue damage accumulation model," *International Journal of Fatigue,* vol. 98, pp. 234-246, 2017.

Appendix A

Section Properties for Case A and B

Case A

Table H.1: General Cross Setions

Section Name	Cross section	Area (m^2)	Tors. Const.	I$_{33, z}$ (m^4)	I$_{22, y}$(m^4)
BremsFag Diag	General	0,002856	1,371E-07	0,000002367	0,000006556
BremsFag Stag1	General	0,005712	2,742E-07	0,000013	0,00001
BremsFag Stag2	General	0,006942	3,911E-07	0,000028	0,000028
Diag 2-3	General	0,013216	8,634E-07	0,000072	0,000029
Diag 3-6	General	0,013884	7,821E-07	0,000063	0,000063
Diag 6-7	General	0,0084	0,00000028	0,000025	0,000025
Diag 7-10	General	0,009984	4,792E-07	0,000031	0,000031
Fag 1-2	General	0,013884	7,821E-07	0,000055	0,000086
Fag 3-4/7-8	General	0,0021	0,00000007	0,000002427	0,000002427
Fag 5-6	General	0,0042	0,00000014	0,000011	0,000019
Fag 9-10	General	0,0042	0,00000014	0,000011	0,000019
LB	General	0,012684	4,073E-07	0,000319	0,000015
LB-Diag	General	0,0015	0,00000005	8,898E-07	8,898E-07
OG 2-6	General	0,016208	0,000002953	0,00007	0,00008
OG 6-10	General	0,022448	0,000007241	0,000198	0,000118
Tverr Diag 2_1	General	0,004992	0,000000606	0,000005677	0,000011
Tverr Diag1_1	General	0,003	2,533E-07	0,00000178	0,00000346
Tverr Diag1_2	General	0,003552	4,332E-07	0,000002063	0,000004174
Tverr Diag2_2	General	0,006942	9,872E-07	0,000013	0,000024
Tverr Ende 1	General	0,015116	5,327E-07	0,000908	0,000014
Tverr Ende 2	General	0,017664	0,000000629	0,001102	0,000026
Tverr Stag	General	0,003	2,533E-07	0,00000178	0,00000346
Tverr1	General	0,006	1,093E-07	0,000258	0,000007212
Tverr2	General	0,007104	0,000000188	0,000306	0,000008705
Tverr3	General	0,006	1,093E-07	0,0003	0,000007212
UG 2-4	General	0,012128	0,000001324	0,000054	0,000041
UG 4-8	General	0,016208	0,000002953	0,00007	0,00008
UG 8-10	General	0,022448	0,000007424	0,000198	0,00012
VFag 1-3	General	0,0042	3,533E-07	0,000004854	0,00000894
VFag 3-5	General	0,003471	1,955E-07	0,000006487	0,000006487
VFag 5-7	General	0,0021	0,00000007	0,000002427	0,000002427
VFag 7-9	General	0,004992	0,000000606	0,000005677	0,000011

Case B

Table A.2: Simplified Cross Sections

Section Name	Cross section	Area (m²⁾	Tors. Const.	I₃₃, z (m⁴)	I₂₂, y(m⁴)
BremsFag Diag	Angle	0,002856	0,000000132	0,000002367	0,000006556
BremsFag Stag1	General	0,005712	2,742E-07	0,000013	0,00001
BremsFag Stag2	General	0,006942	3,911E-07	0,000028	0,000028
Diag 2-3	General	0,013216	8,634E-07	0,000072	0,000029
Diag 3-6	General	0,013388	7,821E-07	0,000063	0,000063
Diag 6-7	General	0,0084	0,00000028	0,000025	0,000025
Diag 7-10	General	0,009984	4,792E-07	0,000031	0,000031
Fag 1-2	General	0,013884	7,821E-07	0,000055	0,000086
Fag 3-4/7-8	Angle	0,0021	6,755E-08	0,000002427	0,000002427
Fag 5-6	General	0,0042	0,00000014	0,000011	0,000019
Fag 9-10	General	0,0042	0,00000014	0,000011	0,000019
LB	I/Wide Flange	0,00868	3,889E-07	0,000229	0,000015
LB-Diag	Angle	0,0015	4,755E-08	8,898E-07	8,898E-07
OG 2-6	Tee	0,014024	0,00000292	0,000068	0,000085
OG 6-10	Tee	0,020264	0,000007354	0,000197	0,000125
Tverr Diag 2_1	Double Angle	0,004992	2,295E-07	0,000005677	0,000011
Tverr Diag1_1	Double Angle	0,003	9,51E-08	0,00000178	0,00000346
Tverr Diag1_2	Double Angle	0,003552	1,603E-07	0,000002063	0,000004174
Tverr Diag2_2	Double Angle	0,006942	3,771E-07	0,000013	0,000024
Tverr Ende 1	I/Wide Flange	0,01164	5,158E-07	0,000654	0,000013
Tverr Ende 2	I/Wide Flange	0,01296	0,000000609	0,000783	0,000025
Tverr Stag	Double Angle	0,003	9,51E-08	0,00000178	0,00000346
Tverr1	General	0,006	1,093E-07	0,000258	0,000007212
Tverr2	General	0,007014	0,000000188	0,000306	0,000008705
Tverr3	General	0,006	1,093E-07	0,0003	0,000007212
UG 2-4	Tee	0,00972	0,000001278	0,000052	0,00004
UG 4-8	Tee	0,014024	0,00000292	0,000068	0,000085
UG 8-10	Tee	0,020264	0,000007354	0,000197	0,000125
VFag 1-3	Double Angle	0,0042	1,351E-07	0,000004854	0,00000894
VFag 3-5	Angle	0,003471	1,885E-07	0,000006487	0,000006487
VFag 5-7	Angle	0,0021	6,755E-08	0,000002427	0,000002427

VFag 7-9	Double Angle	0,004992	2,295E-07	0,000005677	0,000011

Appendix B
Load-time Histories

Case B: General cross sections
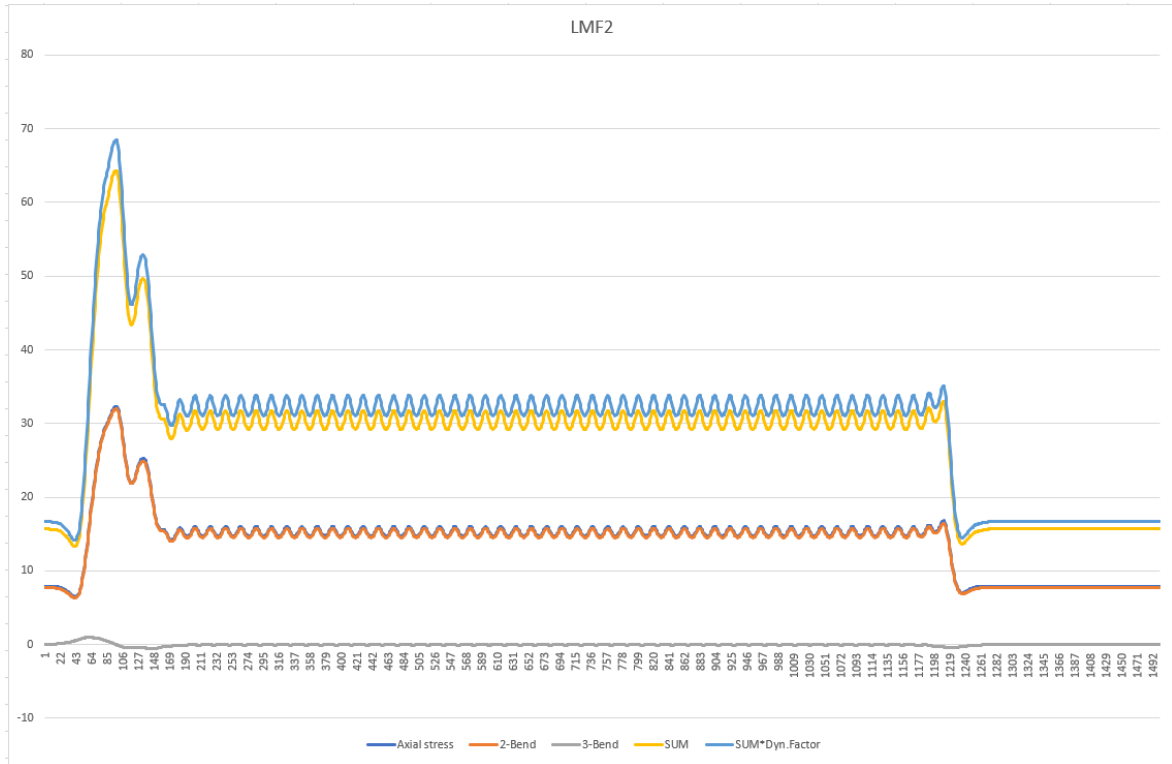
Figure B.1: LMF2
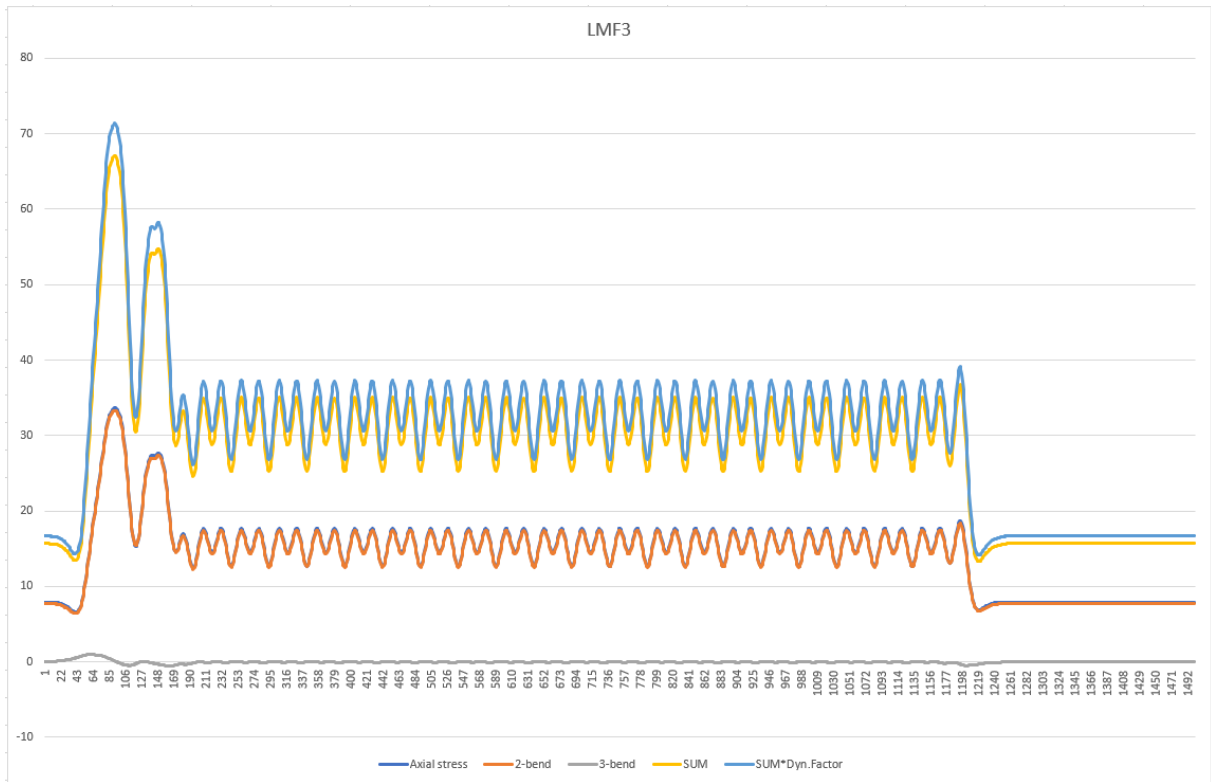
Figure B.2: LMF3
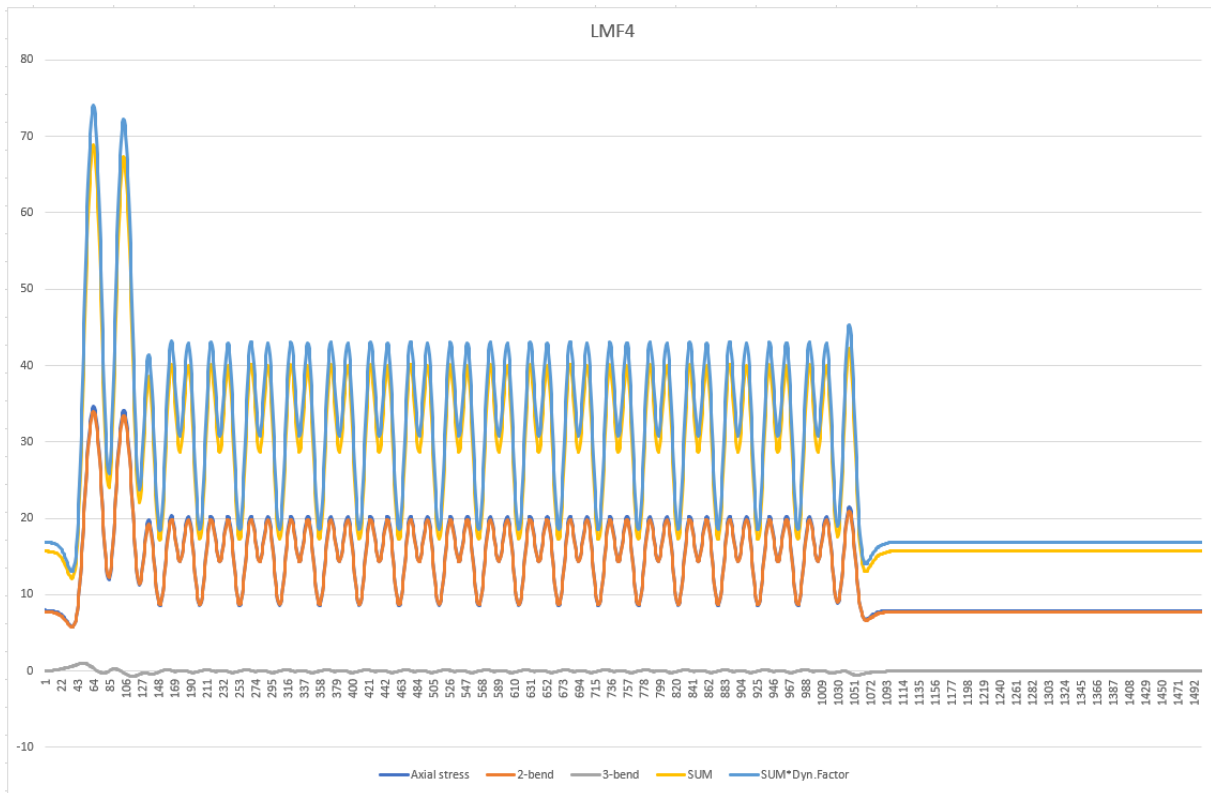
Figure B.3: LMF4
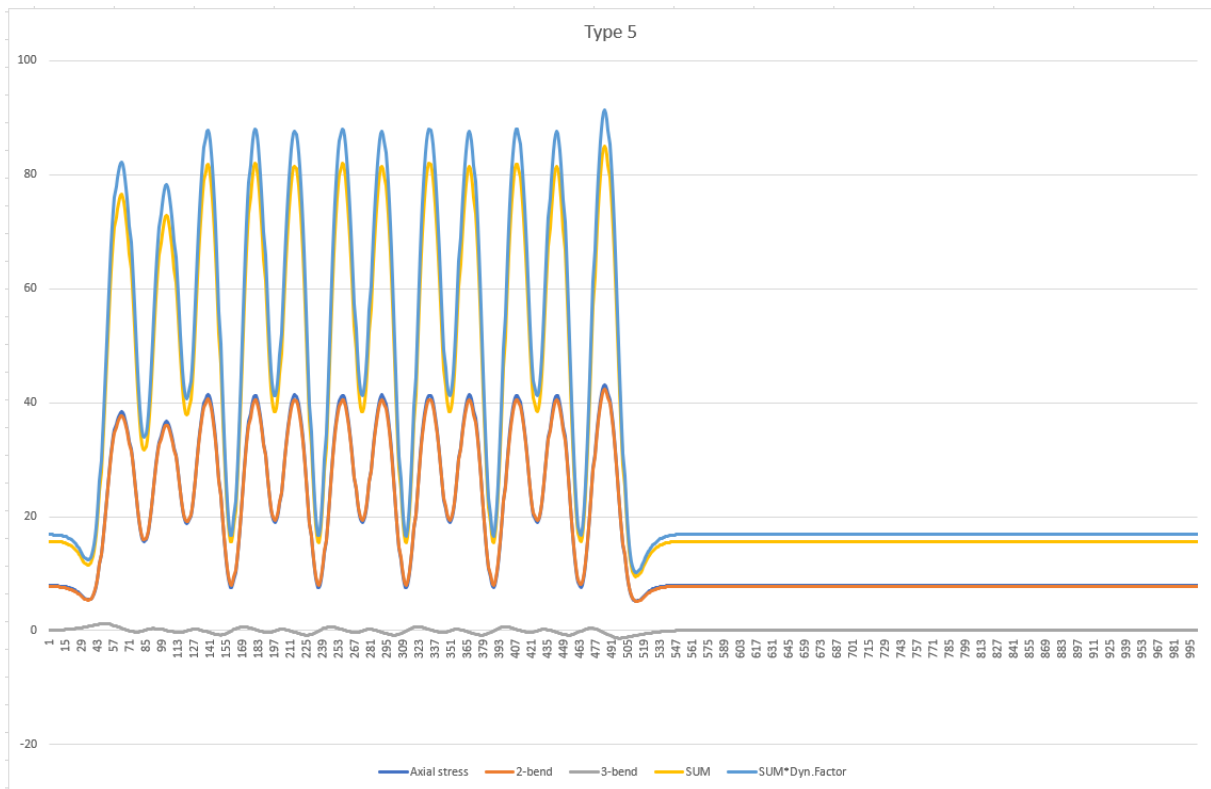
Figure B.4: Type 5
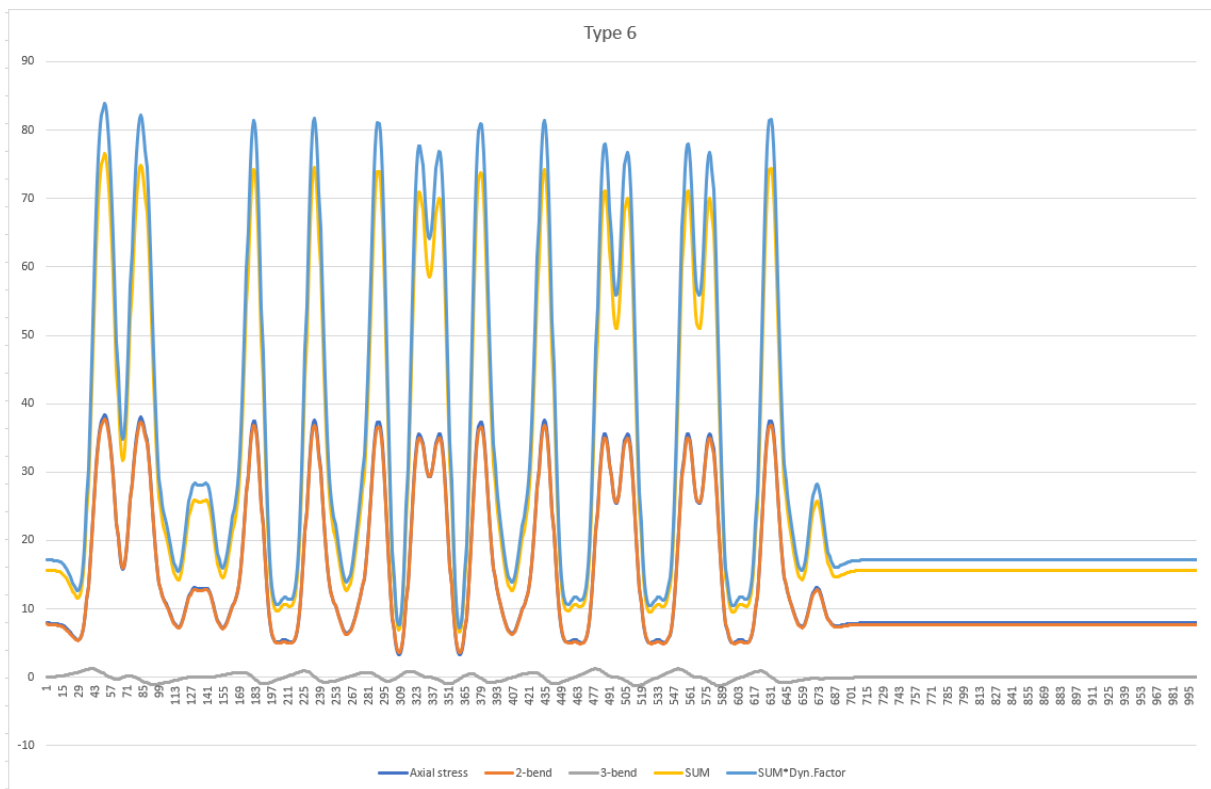
Figure B.5: Type 6
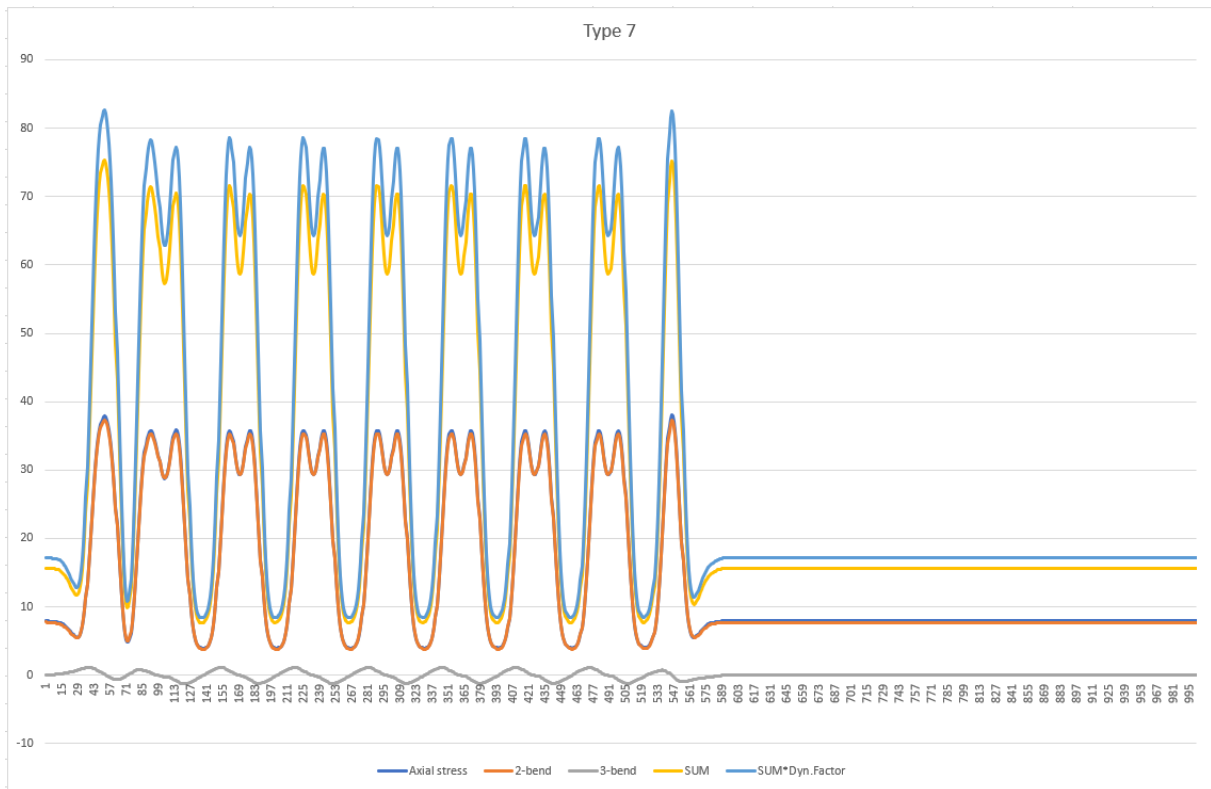
Figure B.6: Type 7
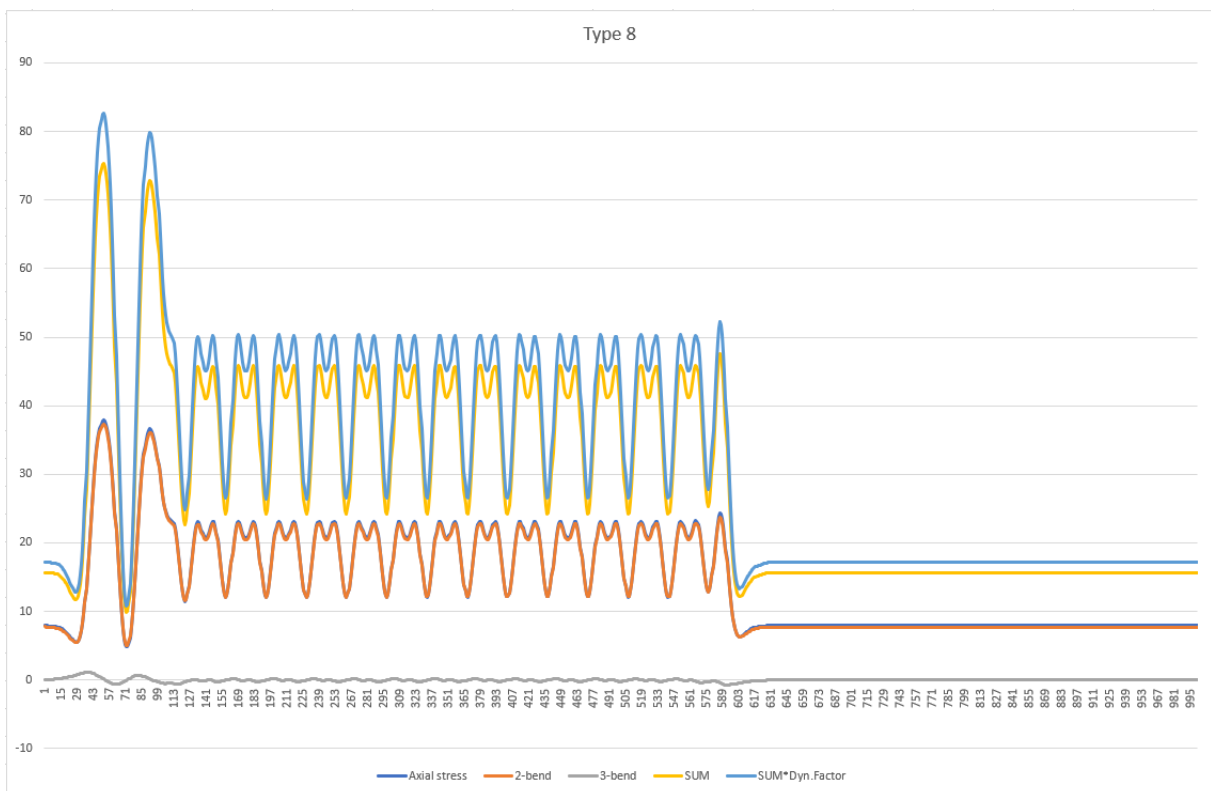
Figure B.7: Type 8
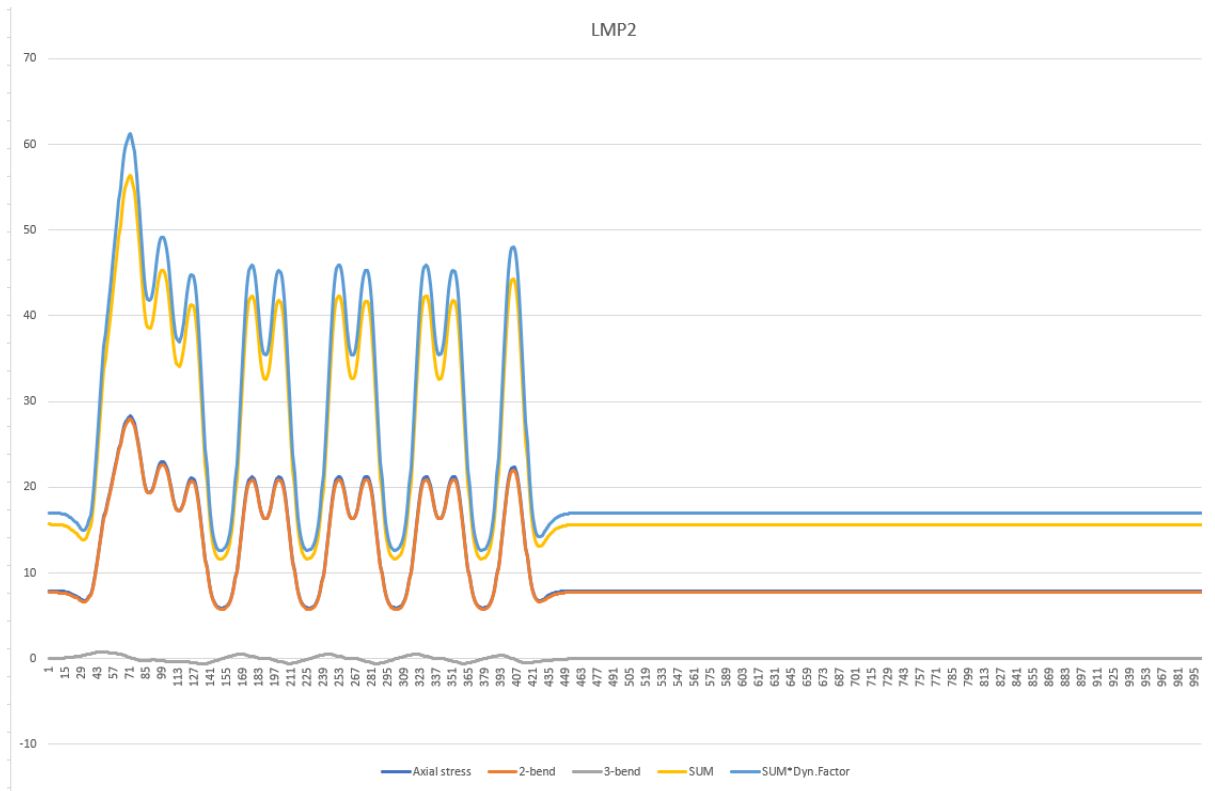
Figure B.8: LMP2
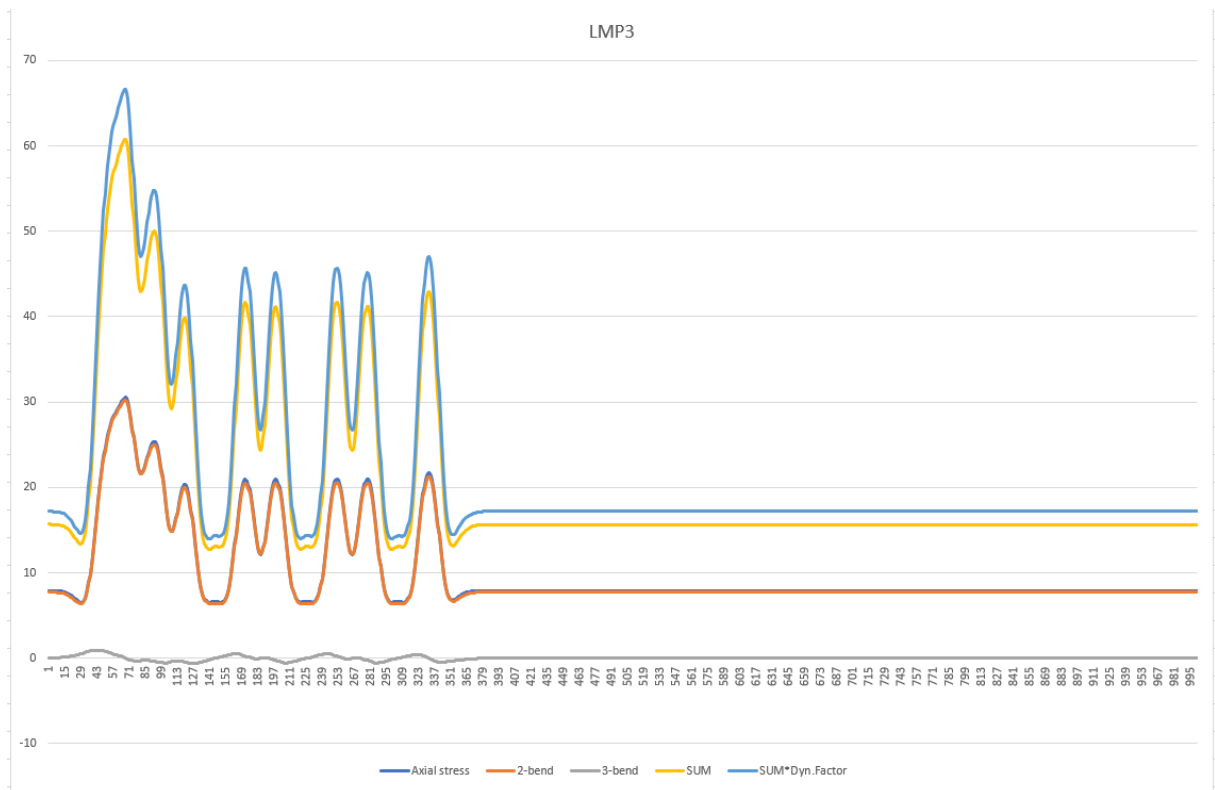
Figure B.9: LMP3

43
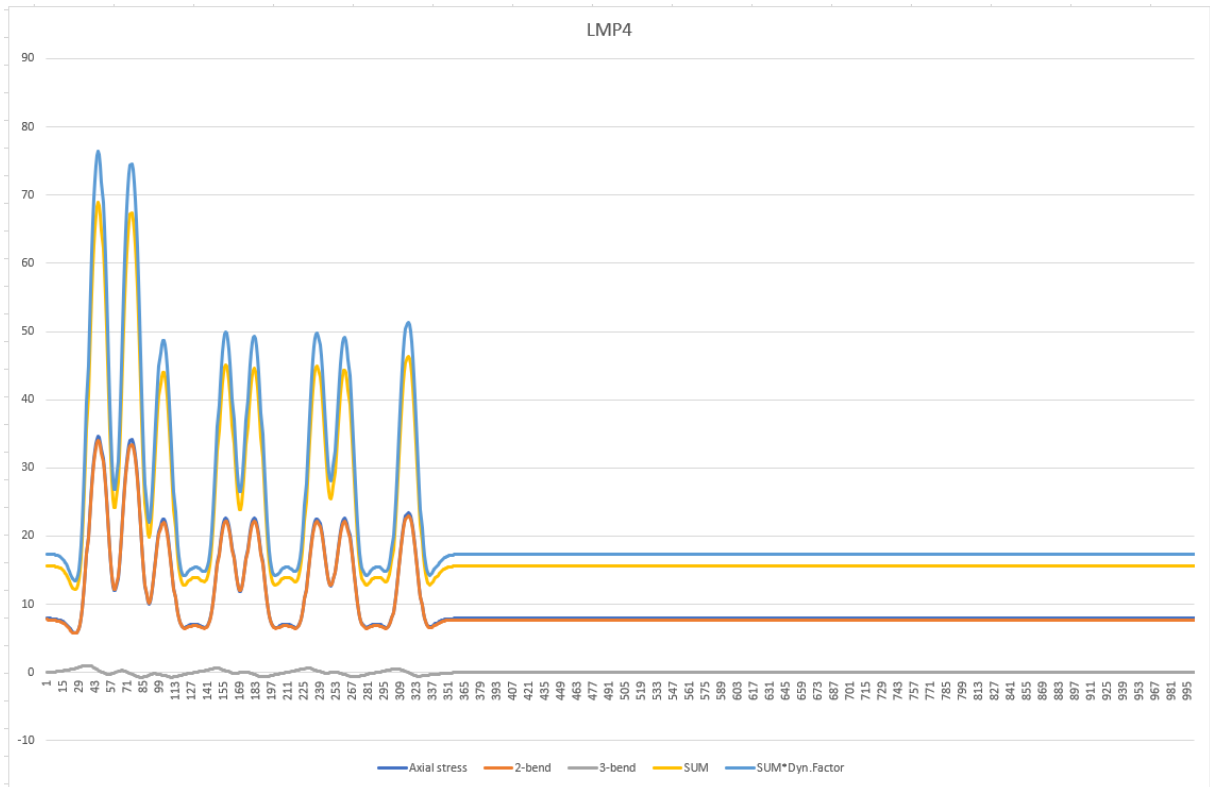
Figure B.10: LMP4
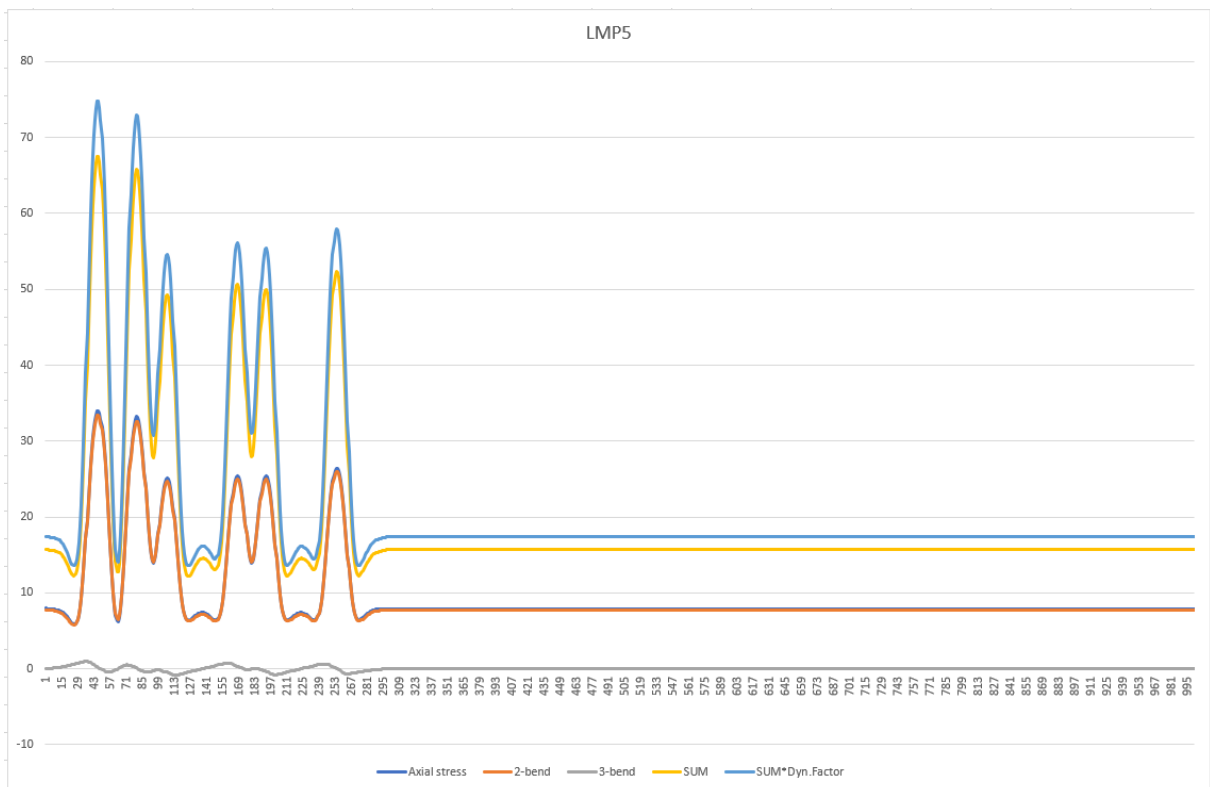
Figure B.11: LMP5

Appendix C
Calculations Palmgren-Miner's Rule

Case A

Table C.1.1: $\Delta\sigma c = 71$, $\gamma_{mf} = 1$

Train	Years in Period	Damage/year	Total Damage
LMF2	24	0,001056419	0,025354056
LMF3	30	0,001212133	0,036363997
LMF4	25	0,00212303	0,053075754
Type 5	38	0,004174777	0,158641535
Type 6	38	0,008330334	0,316552678
Type 7	38	0,005002521	0,190095786
Type 8	38	0,000820136	0,031165165
LMP2	24	0,000738105	0,017714515
LMP3	30	0,000864023	0,0259207
LMP4	25	0,001982477	0,049561934
LMP5	38	0,00263157	0,099999645
		SUM:	1,004445764

Table C.1.2: $\Delta\sigma c = 71$, $\gamma_{mf} = 1.35$

Train	Years in Period	Damage/year	Total Damage
LMF2	24	0,002599187	0,062380485
LMF3	30	0,003133551	0,094006533
LMF4	25	0,008314082	0,207852046
Type 5	38	0,010594263	0,402581982
Type 6	38	0,020628741	0,783892159
Type 7	38	0,012308077	0,467706919
Type 8	38	0,002406271	0,091438287
LMP2	24	0,002600544	0,062413049
LMP3	30	0,00248637	0,074591106
LMP4	25	0,005889576	0,14723939
LMP5	38	0,007176509	0,272707345
		SUM:	2,666809302

Table C.1.3: $\Delta\sigma c = 71$, $\gamma_{mf} = 2.0$

ø	Years in Period	Damage/year	Total Damage
LMF2	24	0,008451352	0,202832447
LMF3	30	0,010759879	0,322796366
LMF4	25	0,037611245	0,940281129
Type 5	38	0,034447636	1,309010154
Type 6	38	0,067323213	2,558282082
Type 7	38	0,040143086	1,525437266
Type 8	38	0,009134287	0,347102893
LMP2	24	0,00975184	0,234044169
LMP3	30	0,009468996	0,284069872
LMP4	25	0,020460386	0,511509642
LMP5	38	0,02390525	0,908399506
		SUM:	9,143765526

Table C.1.4: $\Delta\sigma c = 85$, $\gamma_{mf} = 1.0$

Train	Years in Period	Damage/year	Total Damage
LMF2	24	0,000478828	0,011491864
LMF3	30	0,000602104	0,018063129
LMF4	25	0,001075143	0,02687857
Type 5	38	0,002303956	0,087550323
Type 6	38	0,004812087	0,182859299
Type 7	38	0,00291546	0,110787499
Type 8	38	0,00047098	0,017897233
LMP2	24	0,000197715	0,004745148
LMP3	30	0,000290572	0,008717165
LMP4	25	0,001011334	0,025283359
LMP5	38	0,0013995	0,053181003
		SUM:	0,547454591

Table C.1.5: Δσc = 85, γ_mf = 1.35

Train	Years in Period	Damage/year	Total Damage
LMF2	24	0,001514802	0,03635524
LMF3	30	0,001799583	0,05398748
LMF4	25	0,004323765	0,108094128
Type 5	38	0,006142197	0,233403479
Type 6	38	0,0119887	0,455570587
Type 7	38	0,007173126	0,272578793
Type 8	38	0,001328739	0,050492087
LMP2	24	0,001278956	0,030694936
LMP3	30	0,001308309	0,039249261
LMP4	25	0,003099249	0,077481217
LMP5	38	0,003980207	0,151247872
		SUM:	1,509155079

Table C.1.6: Δσc = 85, γ_mf = 2.0

Train	Years in Period	Damage/year	Total Damage
LMF2	24	0,004925433	0,118210403
LMF3	30	0,006090338	0,182710143
LMF4	25	0,018844811	0,471120279
Type 5	38	0,020076023	0,762888879
Type 6	38	0,039149475	1,487680045
Type 7	38	0,023323684	0,886299991
Type 8	38	0,004980691	0,189266245
LMP2	24	0,005683356	0,136400541
LMP3	30	0,005468388	0,164051643
LMP4	25	0,011709912	0,292747812
LMP5	38	0,013747292	0,522397113
		SUM:	5,213773095

Case B

Table C.2.1: $\Delta\sigma c = 71$, $\gamma_{mf} = 1.0$

Train	Years in Period	Damage/year	Total damage
LMF2	24	0,002163117	0,05191482
LMF3	30	0,001172272	0,035168152
LMF4	25	0,001884983	0,047124571
Type 5	40	0,003667871	0,146714842
Type 6	40	0,007758026	0,310321044
Type 7	40	0,004737442	0,189497685
Type 8	40	0,000760403	0,030416106
LMP2	24	0,000679672	0,016312116
LMP3	30	0,000807447	0,024223396
LMP4	25	0,001818305	0,045457621
LMP5	40	0,002464775	0,098590981
		SUM:	0,995741333

Table C.2.2: $\Delta\sigma c = 71$, $\gamma_{mf} = 1.33$

Train	Years in Period	Damage/year	Total damage
LMF2	24	0,005440554	0,130573306
LMF3	30	0,003002702	0,090081068
LMF4	25	0,006676765	0,166919128
Type 5	40	0,009435695	0,377427783
Type 6	40	0,019195246	0,767809844
Type 7	40	0,011655884	0,466235366
Type 8	40	0,002081213	0,083248522
LMP2	24	0,002455823	0,058939753
LMP3	30	0,002339162	0,070174861
LMP4	25	0,00544059	0,13601474
LMP5	40	0,006746171	0,269846858
		SUM:	2,617271231

Table C.2.3: Δσc = 71, γ$_{mf}$ = 2.0

Train	Years in Period	Damage/year	Total damage
LMF2	24	0,018150423	0,435610158
LMF3	30	0,010223657	0,306709715
LMF4	25	0,028428512	0,710712805
Type 5	40	0,030712397	1,228495865
Type 6	40	0,062625934	2,505037378
Type 7	40	0,037910516	1,516420639
Type 8	40	0,007566782	0,302671264
LMP2	24	0,009327821	0,223867708
LMP3	30	0,008950805	0,268524147
LMP4	25	0,019094011	0,47735027
LMP5	40	0,022403644	0,89614578
		SUM:	8,871545729

Table C.2.4: Δσc = 85, γ$_{mf}$ = 1.0

Train	Years in Period	Damage/year	Total damage
LMF2	24	0,000999784	0,02399482
LMF3	30	0,000569472	0,017084152
LMF4	25	0,000958174	0,023954357
Type 5	40	0,002036131	0,081445249
Type 6	40	0,004486345	0,179453809
Type 7	40	0,002760973	0,110438927
Type 8	40	0,000443161	0,017726455
LMP2	24	0,000179482	0,004307558
LMP3	30	0,000261499	0,007844968
LMP4	25	0,000929183	0,023229576
LMP5	40	0,001257574	0,05030297
		SUM:	0,53978284

Table C.2.5: Δσc = 85, γ_mf = 1.35

Train	Years in Period	Damage/year	Total damage
LMF2	24	0,003149874	0,07559698
LMF3	30	0,001729098	0,051872943
LMF4	25	0,002952941	0,073823532
Type 5	40	0,005467196	0,218687852
Type 6	40	0,011186957	0,447478264
Type 7	40	0,006793029	0,271721174
Type 8	40	0,001101988	0,04407952
LMP2	24	0,001203974	0,028895386
LMP3	30	0,001230217	0,036906499
LMP4	25	0,002884333	0,072108329
LMP5	40	0,003733207	0,149328295
		SUM:	1,470498775

Table C.2.6: Δσc = 85, γ_mf = 2.0

Train	Years in Period	Damage/year	Total damage
LMF2	24	0,010429086	0,250298075
LMF3	30	0,005809381	0,174281422
LMF4	25	0,014427693	0,36069233
Type 5	40	0,017899132	0,71596529
Type 6	40	0,03646096	1,458438392
Type 7	40	0,022087785	0,883511414
Type 8	40	0,004155663	0,166226522
LMP2	24	0,005436238	0,130469717
LMP3	30	0,005176632	0,155298947
LMP4	25	0,010923577	0,273089435
LMP5	40	0,012905281	0,516211229
		SUM:	5,084482773

Appendix D
Nonlinear Fatigue Damage Model: MATLAB Script

Case 1

```
clc
clear

%Assumptions:

%The script simply runs a matrix named "Data" in a loop. Hence, it is
%important to ensure that the correct sequence is added by cyclically going
%through the matrix.

%Ds is a parameter which saves the damage state per time the script has
%gone through the matrix.
%Dt is a parameter which saves the damage during the running of the matrix.

%a=3 is used, as the damage concept depends upon the stress amplitude, not
%the range, whereas (Si-Se) can be divided or multiplied by 2 to go from
%range to amp, or amp to range respectively.

a=3;

Su=350; %Ultimate tensile strength
Se=34.392105; %SigmaL (lower fatigue limit for variable amplitude)


%Data input, "cycles, range, capacity" columns from excel. The length
%depends upon the

DataP1=[0.5 46.226096098071 22854197.8721825
0.5 48.5867765945837    17816058.6895933
0.5 35.4641611398813    85991121.7073122
0.5 46.226096098071 22854197.8721825
0.5 48.5867765945837    17816058.6895933
0.5 35.4641611398813    85991121.7073122
0.5 46.226096098071 22854197.8721825
0.5 48.5867765945837    17816058.6895933
0.5 35.4641611398813    85991121.7073122
0.5 46.226096098071 22854197.8721825
0.5 48.5867765945837    17816058.6895933
0.5 35.4641611398813    85991121.7073122
0.5 46.226096098071 22854197.8721825
0.5 48.5867765945837    17816058.6895933
0.5 35.4641611398813    85991121.7073122
0.5 46.226096098071 22854197.8721825
0.5 48.5867765945837    17816058.6895933
0.5 35.4641611398813    85991121.7073122
0.5 46.226096098071 22854197.8721825
0.5 48.5867765945837    17816058.6895933
0.5 35.4641611398813    85991121.7073122
0.5 46.226096098071 22854197.8721825
0.5 48.5867765945837    17816058.6895933
0.5 35.4641611398813    85991121.7073122
0.5 46.226096098071 22854197.8721825
0.5 48.5867765945837    17816058.6895933
```

```
0.5 35.4641611398813      85991121.7073122
0.5 57.1380858972918       7920883.65398595
0.5 57.2939509937503       7813726.59327422
0.5 54.2590854887917      10257457.4453844
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918       7920883.65398595
0.5 57.2939509937503       7813726.59327422
0.5 54.2590854887917      10257457.4453844
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918       7920883.65398595
0.5 57.2939509937503       7813726.59327422
0.5 54.2590854887917      10257457.4453844
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918       7920883.65398595
0.5 57.2939509937503       7813726.59327422
0.5 54.2590854887917      10257457.4453844
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918       7920883.65398595
0.5 57.2939509937503       7813726.59327422
0.5 54.2590854887917      10257457.4453844
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918       7920883.65398595
0.5 57.2939509937503       7813726.59327422
0.5 54.2590854887917      10257457.4453844
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918       7920883.65398595
0.5 57.2939509937503       7813726.59327422
0.5 54.2590854887917      10257457.4453844
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918       7920883.65398595
0.5 57.2939509937503       7813726.59327422
0.5 54.2590854887917      10257457.4453844
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918       7920883.65398595
0.5 57.2939509937503       7813726.59327422
0.5 54.2590854887917      10257457.4453844
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918       7920883.65398595
0.5 57.2939509937503       7813726.59327422
0.5 54.2590854887917      10257457.4453844
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918       7920883.65398595
0.5 57.2939509937503       7813726.59327422
0.5 54.2590854887917      10257457.4453844
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918       7920883.65398595
0.5 57.2939509937503       7813726.59327422
0.5 54.2590854887917      10257457.4453844
0.5 53.935068489208 10569292.0733368
];

neqp=0;

DataP2=[0.5 51.9506764991708      12748116.8545779
0.5 52.6703800807873      11900626.1007275
0.5 51.9506764991708      12748116.8545779
0.5 52.6703800807873      11900626.1007275
0.5 51.9506764991708      12748116.8545779
0.5 52.6703800807873      11900626.1007275
0.5 51.9506764991708      12748116.8545779
```

```
0.5 52.6703800807873      11900626.1007275
0.5 51.9506764991708      12748116.8545779
0.5 52.6703800807873      11900626.1007275
0.5 51.9506764991708      12748116.8545779
0.5 52.6703800807873      11900626.1007275
0.5 51.9506764991708      12748116.8545779
0.5 52.6703800807873      11900626.1007275
0.5 51.9506764991708      12748116.8545779
0.5 52.6703800807873      11900626.1007275
0.5 51.9506764991708      12748116.8545779
0.5 52.6703800807873      11900626.1007275
0.5 57.1380858972918      7920883.65398595
0.5 57.2939509937503      7813726.59327422
0.5 57.1380858972918      7920883.65398595
0.5 57.2939509937503      7813726.59327422
0.5 57.1380858972918      7920883.65398595
0.5 57.2939509937503      7813726.59327422
0.5 57.1380858972918      7920883.65398595
0.5 57.2939509937503      7813726.59327422
0.5 57.1380858972918      7920883.65398595
0.5 57.2939509937503      7813726.59327422
0.5 57.1380858972918      7920883.65398595
0.5 57.2939509937503      7813726.59327422
0.5 57.1380858972918      7920883.65398595
0.5 57.2939509937503      7813726.59327422
0.5 57.1380858972918      7920883.65398595
0.5 57.2939509937503      7813726.59327422
0.5 57.1380858972918      7920883.65398595
0.5 57.2939509937503      7813726.59327422
0.5 57.1380858972918      7920883.65398595
0.5 57.2939509937503      7813726.59327422
0.5 57.1380858972918      7920883.65398595
0.5 57.2939509937503      7813726.59327422
0.5 57.1380858972918      7920883.65398595
0.5 57.2939509937503      7813726.59327422
];

DataP3=[1   47.8116211975847    19307885.8309833
1   35.5793299130182      84608354.6836839
1   35.7772253228728      82294117.3424372
0.5 62.9109099849569      4932974.50996256
0.5 62.2396435175844      5164955.56470239
0.5 37.2253481913988      67485151.7631408
0.5 37.1342025123284      68317436.9421162
1   47.8116211975847      19307885.8309833
1   35.5793299130182      84608354.6836839
1   35.7772253228728      82294117.3424372
0.5 62.9109099849569      4932974.50996256
0.5 62.2396435175844      5164955.56470239
0.5 37.2253481913988      67485151.7631408
0.5 37.1342025123284      68317436.9421162
1   47.8116211975847      19307885.8309833
1   35.5793299130182      84608354.6836839
1   35.7772253228728      82294117.3424372
0.5 62.9109099849569      4932974.50996256
0.5 62.2396435175844      5164955.56470239
0.5 37.2253481913988      67485151.7631408
0.5 37.1342025123284      68317436.9421162
1   47.8116211975847      19307885.8309833
1   35.5793299130182      84608354.6836839
1   35.7772253228728      82294117.3424372
```

```
0.5  62.9109099849569        4932974.50996256
0.5  62.2396435175844        5164955.56470239
0.5  37.2253481913988        67485151.7631408
0.5  37.1342025123284        68317436.9421162
1    47.8116211975847        19307885.8309833
1    35.5793299130182        84608354.6836839
1    35.7772253228728        82294117.3424372
0.5  62.9109099849569        4932974.50996256
0.5  62.2396435175844        5164955.56470239
0.5  37.2253481913988        67485151.7631408
0.5  37.1342025123284        68317436.9421162
1    47.8116211975847        19307885.8309833
1    35.5793299130182        84608354.6836839
1    35.7772253228728        82294117.3424372
0.5  62.9109099849569        4932974.50996256
0.5  62.2396435175844        5164955.56470239
0.5  37.2253481913988        67485151.7631408
0.5  37.1342025123284        68317436.9421162
1    47.8116211975847        19307885.8309833
1    35.5793299130182        84608354.6836839
1    35.7772253228728        82294117.3424372
0.5  62.9109099849569        4932974.50996256
0.5  62.2396435175844        5164955.56470239
0.5  37.2253481913988        67485151.7631408
0.5  37.1342025123284        68317436.9421162
1    47.8116211975847        19307885.8309833
1    35.5793299130182        84608354.6836839
1    35.7772253228728        82294117.3424372
0.5  62.9109099849569        4932974.50996256
0.5  62.2396435175844        5164955.56470239
0.5  37.2253481913988        67485151.7631408
0.5  37.1342025123284        68317436.9421162
1    47.8116211975847        19307885.8309833
1    35.5793299130182        84608354.6836839
1    35.7772253228728        82294117.3424372
0.5  62.9109099849569        4932974.50996256
0.5  62.2396435175844        5164955.56470239
0.5  37.2253481913988        67485151.7631408
0.5  37.1342025123284        68317436.9421162
1    46.4623533962284        22279018.4615184
0.5  61.0258912730212        5699429.914408
0.5  60.0989586913725        6152722.61388507
1    46.4623533962284        22279018.4615184
0.5  61.0258912730212        5699429.914408
0.5  60.0989586913725        6152722.61388507
1    46.4623533962284        22279018.4615184
0.5  61.0258912730212        5699429.914408
0.5  60.0989586913725        6152722.61388507
1    46.4623533962284        22279018.4615184
0.5  61.0258912730212        5699429.914408
0.5  60.0989586913725        6152722.61388507
1    46.4623533962284        22279018.4615184
0.5  61.0258912730212        5699429.914408
0.5  60.0989586913725        6152722.61388507
1    46.4623533962284        22279018.4615184
0.5  61.0258912730212        5699429.914408
0.5  60.0989586913725        6152722.61388507
1    46.4623533962284        22279018.4615184
0.5  61.0258912730212        5699429.914408
0.5  60.0989586913725        6152722.61388507
1    46.4623533962284        22279018.4615184
```

```
0.5 61.0258912730212      5699429.914408
0.5 60.0989586913725      6152722.61388507
1   46.4623533962284      22279018.4615184
0.5 61.0258912730212      5699429.914408
0.5 60.0989586913725      6152722.61388507
1   46.4623533962284      22279018.4615184
0.5 61.0258912730212      5699429.914408
0.5 60.0989586913725      6152722.61388507
1   46.4623533962284      22279018.4615184
0.5 61.0258912730212      5699429.914408
0.5 60.0989586913725      6152722.61388507
1   46.4623533962284      22279018.4615184
0.5 61.0258912730212      5699429.914408
0.5 60.0989586913725      6152722.61388507

];


DataP4=[1    58.8994561685063      6805274.50157922
0.5 61.2569759516093      5592735.82058889
1   42.5447678796853      34607583.8297449
1   44.3684368962769      28056366.6506288
0.5 61.3098074314208      5568680.63795527
1   58.8994561685063      6805274.50157922
0.5 61.2569759516093      5592735.82058889
1   42.5447678796853      34607583.8297449
1   44.3684368962769      28056366.6506288
0.5 61.3098074314208      5568680.63795527
1   58.8994561685063      6805274.50157922
0.5 61.2569759516093      5592735.82058889
1   42.5447678796853      34607583.8297449
1   44.3684368962769      28056366.6506288
0.5 61.3098074314208      5568680.63795527
1   58.8994561685063      6805274.50157922
0.5 61.2569759516093      5592735.82058889
1   42.5447678796853      34607583.8297449
1   44.3684368962769      28056366.6506288
0.5 61.3098074314208      5568680.63795527
1   58.8994561685063      6805274.50157922
0.5 61.2569759516093      5592735.82058889
1   42.5447678796853      34607583.8297449
1   44.3684368962769      28056366.6506288
0.5 61.3098074314208      5568680.63795527
1   58.8994561685063      6805274.50157922
0.5 61.2569759516093      5592735.82058889
1   42.5447678796853      34607583.8297449
1   44.3684368962769      28056366.6506288
0.5 61.3098074314208      5568680.63795527
1   58.8994561685063      6805274.50157922
0.5 61.2569759516093      5592735.82058889
1   42.5447678796853      34607583.8297449
1   44.3684368962769      28056366.6506288
0.5 61.3098074314208      5568680.63795527
1   58.8994561685063      6805274.50157922
0.5 61.2569759516093      5592735.82058889
1   42.5447678796853      34607583.8297449
1   44.3684368962769      28056366.6506288
0.5 61.3098074314208      5568680.63795527
1   58.8994561685063      6805274.50157922
0.5 61.2569759516093      5592735.82058889
1   42.5447678796853      34607583.8297449
```

```
1    44.3684368962769     28056366.6506288
0.5  61.3098074314208     5568680.63795527
1    37.6707189603938     63589077.904494
1    48.190968177313 18559822.3285534
1    71.2286443522726     3398777.51493037
1    46.3071101844495     22654979.5606621
1    46.3271704976026     22605972.4046332
1    71.3866290926499     3376262.05765601
1    46.3431695038957     22566978.151031
1    71.3959542695662     3374939.28820999
1    46.3551019631063     22537947.8029306
1    71.3414627670503     3382678.65928567
1    71.4432777365421     3368237.12966494
0.5  78.9548831493413     2495455.2567017
0.5  81.1547967766676     2297969.12418088
1    37.6707189603938     63589077.904494
1    48.190968177313 18559822.3285534
1    71.2286443522726     3398777.51493037
1    46.3071101844495     22654979.5606621
1    46.3271704976026     22605972.4046332
1    71.3866290926499     3376262.05765601
1    46.3431695038957     22566978.151031
1    71.3959542695662     3374939.28820999
1    46.3551019631063     22537947.8029306
1    71.3414627670503     3382678.65928567
1    71.4432777365421     3368237.12966494
0.5  78.9548831493413     2495455.2567017
0.5  81.1547967766676     2297969.12418088
1    37.6707189603938     63589077.904494
1    48.190968177313 18559822.3285534
1    71.2286443522726     3398777.51493037
1    46.3071101844495     22654979.5606621
1    46.3271704976026     22605972.4046332
1    71.3866290926499     3376262.05765601
1    46.3431695038957     22566978.151031
1    71.3959542695662     3374939.28820999
1    46.3551019631063     22537947.8029306
1    71.3414627670503     3382678.65928567
1    71.4432777365421     3368237.12966494
0.5  78.9548831493413     2495455.2567017
0.5  81.1547967766676     2297969.12418088
1    47.3934577932878     20174839.7522824
1    65.8866035381933     4294329.97286048
0.5  71.2342409398877     3397976.49305574
1    67.1055430588824     4064541.73187302
1    71.0149642525944     3429550.12322718
1    70.1192327783428     3562667.61641143
1    66.9790220184167     4087618.60186114
1    67.3382455269314     4022549.36369605
1    67.5821154403866     3979160.22929321
1    71.0030973799226     3431269.97088661
0.5  76.6526406684916     2727127.29665858
0.5  74.3295921440053     2990897.4568502
0.5  65.9482943402822     4282289.95321429
1    47.3934577932878     20174839.7522824
1    65.8866035381933     4294329.97286048
0.5  71.2342409398877     3397976.49305574
1    67.1055430588824     4064541.73187302
1    71.0149642525944     3429550.12322718
1    70.1192327783428     3562667.61641143
1    66.9790220184167     4087618.60186114
```

```
1    67.3382455269314     4022549.36369605
1    67.5821154403866     3979160.22929321
1    71.0030973799226     3431269.97088661
0.5  76.6526406684916     2727127.29665858
0.5  74.3295921440053     2990897.4568502
0.5  65.9482943402822     4282289.95321429
1    47.3934577932878     20174839.7522824
1    65.8866035381933     4294329.97286048
0.5  71.2342409398877     3397976.49305574
1    67.1055430588824     4064541.73187302
1    71.0149642525944     3429550.12322718
1    70.1192327783428     3562667.61641143
1    66.9790220184167     4087618.60186114
1    67.3382455269314     4022549.36369605
1    67.5821154403866     3979160.22929321
1    71.0030973799226     3431269.97088661
0.5  76.6526406684916     2727127.29665858
0.5  74.3295921440053     2990897.4568502
0.5  65.9482943402822     4282289.95321429
1    47.3934577932878     20174839.7522824
1    65.8866035381933     4294329.97286048
0.5  71.2342409398877     3397976.49305574
1    67.1055430588824     4064541.73187302
1    71.0149642525944     3429550.12322718
1    70.1192327783428     3562667.61641143
1    66.9790220184167     4087618.60186114
1    67.3382455269314     4022549.36369605
1    67.5821154403866     3979160.22929321
1    71.0030973799226     3431269.97088661
0.5  76.6526406684916     2727127.29665858
0.5  74.3295921440053     2990897.4568502
0.5  65.9482943402822     4282289.95321429
0.5  69.8693165191763     3601034.54942235
1    67.5008688741871     3993545.9280625
1    70.0636715908721     3571150.03191316
1    69.9908116468448     3582314.27215596
1    70.0253589607856     3577014.83675995
1    69.9969715746962     3581368.5945754
1    70.0913605565667     3566919.4508751
1    70.1121545528832     3563746.74073921
0.5  74.2565992936875     2999726.11860515
0.5  74.085132494472 3020602.56651052
0.5  71.0922847019632     3418372.28345572
0.5  69.8693165191763     3601034.54942235
1    67.5008688741871     3993545.9280625
1    70.0636715908721     3571150.03191316
1    69.9908116468448     3582314.27215596
1    70.0253589607856     3577014.83675995
1    69.9969715746962     3581368.5945754
1    70.0913605565667     3566919.4508751
1    70.1121545528832     3563746.74073921
0.5  74.2565992936875     2999726.11860515
0.5  74.085132494472 3020602.56651052
0.5  71.0922847019632     3418372.28345572
0.5  69.8693165191763     3601034.54942235
1    67.5008688741871     3993545.9280625
1    70.0636715908721     3571150.03191316
1    69.9908116468448     3582314.27215596
1    70.0253589607856     3577014.83675995
1    69.9969715746962     3581368.5945754
1    70.0913605565667     3566919.4508751
```

```matlab
1   70.1121545528832     3563746.74073921
0.5 74.2565992936875     2999726.11860515
0.5 74.085132494472 3020602.56651052
0.5 71.0922847019632     3418372.28345572
0.5 69.8693165191763     3601034.54942235
0.5 71.8482693639623     3311599.7265311
0.5 69.0406853137464     3732256.57591286
0.5 66.500126263035 4176566.13857651
0.5 69.8693165191763     3601034.54942235
0.5 71.8482693639623     3311599.7265311
0.5 69.0406853137464     3732256.57591286
0.5 66.500126263035 4176566.13857651
];


timesP1=8760;

length1=length(DataP1);

disp('Train P1')

for j=1:1:timesP1

for i=1:1:length1
    k=i;
    if i<length1
        D=((DataP1(k,1)/DataP1(k,3))+neqp)^(a*(Su-Se)/(DataP1(k,2)-Se));

        neqp=(DataP1(k,1)/DataP1(k,3)+neqp)^((DataP1(k+1,2)-
Se)/(DataP1(k,2)-Se));
    else
        D=((DataP1(k,1)/DataP1(k,3))+neqp)^(a*(Su-Se)/(DataP1(k,2)-Se));

        neqps=neqp;

        neqp=(DataP1(k,1)/DataP1(k,3)+neqp)^((DataP1(1,2)-Se)/(DataP1(k,2)-
Se));
    end

    Dt(i,1)=j;
    Dt(i,2)=D;

end

    Ds(j,1)=j;
    Ds(j,2)=D;
    Ds(j,3)=1;

    if D>1
        return
    else
    end

end

neqp=(DataP1(length1,1)/DataP1(length1,3)+neqps)^((DataP2(1,2)-
Se)/(DataP1(length1,2)-Se));
```

```matlab
timesP=timesP1;
timesP2=10950;

length2=length(DataP2);

disp('Train P2')

for j=1:1:timesP2

for i=1:1:length2
    k=i;
    if i<length2
        D=((DataP2(k,1)/DataP2(k,3))+neqp)^(a*(Su-Se)/(DataP2(k,2)-Se));

        neqp=(DataP2(k,1)/DataP2(k,3)+neqp)^((DataP2(k+1,2)-
Se)/(DataP2(k,2)-Se));
    else
        D=((DataP2(k,1)/DataP2(k,3))+neqp)^(a*(Su-Se)/(DataP2(k,2)-Se));

        neqps=neqp;

        neqp=(DataP2(k,1)/DataP2(k,3)+neqp)^((DataP2(1,2)-Se)/(DataP2(k,2)-
Se));
    end

    Dt(i,1)=j;
    Dt(i,2)=D;

end

    Ds(timesP+j,1)=j+timesP;
    Ds(timesP+j,2)=D;
    Ds(timesP+j,3)=2;

    if D>1
        return
    else
    end

end

neqp=(DataP2(length2,1)/DataP2(length2,3)+neqps)^((DataP3(1,2)-
Se)/(DataP2(length2,2)-Se));

timesP=timesP1+timesP2;
timesP3=9125;

length3=length(DataP3);

disp('Train P3')

for j=1:1:timesP3

for i=1:1:length3
    k=i;
    if i<length3
        D=((DataP3(k,1)/DataP3(k,3))+neqp)^(a*(Su-Se)/(DataP3(k,2)-Se));
```

```matlab
            neqp=(DataP3(k,1)/DataP3(k,3)+neqp)^((DataP3(k+1,2)-
Se)/(DataP3(k,2)-Se));
    else
        D=((DataP3(k,1)/DataP3(k,3))+neqp)^(a*(Su-Se)/(DataP3(k,2)-Se));

        neqps=neqp;

        neqp=(DataP3(k,1)/DataP3(k,3)+neqp)^((DataP3(1,2)-Se)/(DataP3(k,2)-
Se));
    end

    Dt(i,1)=j;
    Dt(i,2)=D;

end

    Ds(timesP+j,1)=j+timesP;
    Ds(timesP+j,2)=D;
    Ds(timesP+j,3)=3;

    if D>1
        return
    else
    end

end

neqp=(DataP3(length3,1)/DataP3(length3,3)+neqps)^((DataP4(1,2)-
Se)/(DataP3(length3,2)-Se));

timesP=timesP1+timesP2+timesP3;
timesP4=13870;

length4=length(DataP4);

disp('Train P4')

for j=1:1:timesP4

for i=1:1:length4
    k=i;
    if i<length4
        D=((DataP4(k,1)/DataP4(k,3))+neqp)^(a*(Su-Se)/(DataP4(k,2)-Se));

        neqp=(DataP4(k,1)/DataP4(k,3)+neqp)^((DataP4(k+1,2)-
Se)/(DataP4(k,2)-Se));
    else
        D=((DataP4(k,1)/DataP4(k,3))+neqp)^(a*(Su-Se)/(DataP4(k,2)-Se));

        neqps=neqp;

        neqp=(DataP4(k,1)/DataP4(k,3)+neqp)^((DataP4(1,2)-Se)/(DataP4(k,2)-
Se));
    end

    Dt(i,1)=j;
```

```matlab
        Dt(i,2)=D;

end

    Ds(timesP+j,1)=j+timesP;
    Ds(timesP+j,2)=D;
    Ds(timesP+j,3)=4;

    if D>1
        return
    else
    end

end

neqp=(DataP4(length4,1)/DataP4(length4,3)+neqps)^((DataP4(1,2)-
Se)/(DataP4(length4,2)-Se));

plot(Ds(:,1),Ds(:,2))

%C(:,1:2)
```

Case 2

```
clc
clear

%Assumptions:

%The script simply runs a matrix named "Data" in a loop. Hence, it is
%important to ensure that the correct sequence is added by cyclically going
%through the matrix.

%Ds is a parameter which saves the damage state per time the script has
%gone through the matrix.
%Dt is a parameter which saves the damage during the running of the matrix.

%a=3 is used, as the damage concept depends upon the stress amplitude, not
%the range, whereas (Si-Se) can be divided or multiplied by 2 to go from
%range to amp, or amp to range respectively.

a=3;

Su=350; %Ultimate tensile strength
Se=34.392105; %SigmaL (lower fatigue limit for variable amplitude)


%Data input, "cycles, range, capacity" columns from excel. The length
%depends upon the

DataP1=[0.5 46.226096098071 22854197.8721825
0.5 48.5867765945837    17816058.6895933
0.5 35.4641611398813    85991121.7073122
0.5 46.226096098071 22854197.8721825
0.5 48.5867765945837    17816058.6895933
0.5 35.4641611398813    85991121.7073122
0.5 46.226096098071 22854197.8721825
0.5 48.5867765945837    17816058.6895933
0.5 35.4641611398813    85991121.7073122
0.5 46.226096098071 22854197.8721825
0.5 48.5867765945837    17816058.6895933
0.5 35.4641611398813    85991121.7073122
0.5 46.226096098071 22854197.8721825
0.5 48.5867765945837    17816058.6895933
0.5 35.4641611398813    85991121.7073122
0.5 46.226096098071 22854197.8721825
0.5 48.5867765945837    17816058.6895933
0.5 35.4641611398813    85991121.7073122
0.5 46.226096098071 22854197.8721825
0.5 48.5867765945837    17816058.6895933
0.5 35.4641611398813    85991121.7073122
0.5 46.226096098071 22854197.8721825
0.5 48.5867765945837    17816058.6895933
0.5 35.4641611398813    85991121.7073122
0.5 46.226096098071 22854197.8721825
0.5 48.5867765945837    17816058.6895933
0.5 35.4641611398813    85991121.7073122
0.5 57.1380858972918    7920883.65398595
0.5 57.2939509937503    7813726.59327422
0.5 54.2590854887917    10257457.4453844
```

```
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918    7920883.65398595
0.5 57.2939509937503    7813726.59327422
0.5 54.2590854887917    10257457.4453844
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918    7920883.65398595
0.5 57.2939509937503    7813726.59327422
0.5 54.2590854887917    10257457.4453844
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918    7920883.65398595
0.5 57.2939509937503    7813726.59327422
0.5 54.2590854887917    10257457.4453844
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918    7920883.65398595
0.5 57.2939509937503    7813726.59327422
0.5 54.2590854887917    10257457.4453844
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918    7920883.65398595
0.5 57.2939509937503    7813726.59327422
0.5 54.2590854887917    10257457.4453844
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918    7920883.65398595
0.5 57.2939509937503    7813726.59327422
0.5 54.2590854887917    10257457.4453844
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918    7920883.65398595
0.5 57.2939509937503    7813726.59327422
0.5 54.2590854887917    10257457.4453844
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918    7920883.65398595
0.5 57.2939509937503    7813726.59327422
0.5 54.2590854887917    10257457.4453844
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918    7920883.65398595
0.5 57.2939509937503    7813726.59327422
0.5 54.2590854887917    10257457.4453844
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918    7920883.65398595
0.5 57.2939509937503    7813726.59327422
0.5 54.2590854887917    10257457.4453844
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918    7920883.65398595
0.5 57.2939509937503    7813726.59327422
0.5 54.2590854887917    10257457.4453844
0.5 53.935068489208 10569292.0733368
];

neqp=0;

DataP2=[0.5 51.9506764991708    12748116.8545779
0.5 52.6703800807873    11900626.1007275
0.5 51.9506764991708    12748116.8545779
0.5 52.6703800807873    11900626.1007275
0.5 51.9506764991708    12748116.8545779
0.5 52.6703800807873    11900626.1007275
0.5 51.9506764991708    12748116.8545779
0.5 52.6703800807873    11900626.1007275
0.5 51.9506764991708    12748116.8545779
0.5 52.6703800807873    11900626.1007275
0.5 51.9506764991708    12748116.8545779
```

```
0.5 52.6703800807873      11900626.1007275
0.5 51.9506764991708      12748116.8545779
0.5 52.6703800807873      11900626.1007275
0.5 51.9506764991708      12748116.8545779
0.5 52.6703800807873      11900626.1007275
0.5 51.9506764991708      12748116.8545779
0.5 52.6703800807873      11900626.1007275
0.5 57.1380858972918      7920883.65398595
0.5 57.2939509937503      7813726.59327422
0.5 57.1380858972918      7920883.65398595
0.5 57.2939509937503      7813726.59327422
0.5 57.1380858972918      7920883.65398595
0.5 57.2939509937503      7813726.59327422
0.5 57.1380858972918      7920883.65398595
0.5 57.2939509937503      7813726.59327422
0.5 57.1380858972918      7920883.65398595
0.5 57.2939509937503      7813726.59327422
0.5 57.1380858972918      7920883.65398595
0.5 57.2939509937503      7813726.59327422
0.5 57.1380858972918      7920883.65398595
0.5 57.2939509937503      7813726.59327422
0.5 57.1380858972918      7920883.65398595
0.5 57.2939509937503      7813726.59327422
0.5 57.1380858972918      7920883.65398595
0.5 57.2939509937503      7813726.59327422
0.5 57.1380858972918      7920883.65398595
0.5 57.2939509937503      7813726.59327422
0.5 57.1380858972918      7920883.65398595
0.5 57.2939509937503      7813726.59327422
];

DataP3=[1    47.8116211975847     19307885.8309833
1    35.5793299130182      84608354.6836839
1    35.7772253228728      82294117.3424372
0.5 62.9109099849569      4932974.50996256
0.5 62.2396435175844      5164955.56470239
0.5 37.2253481913988      67485151.7631408
0.5 37.1342025123284      68317436.9421162
1    47.8116211975847      19307885.8309833
1    35.5793299130182      84608354.6836839
1    35.7772253228728      82294117.3424372
0.5 62.9109099849569      4932974.50996256
0.5 62.2396435175844      5164955.56470239
0.5 37.2253481913988      67485151.7631408
0.5 37.1342025123284      68317436.9421162
1    47.8116211975847      19307885.8309833
1    35.5793299130182      84608354.6836839
1    35.7772253228728      82294117.3424372
0.5 62.9109099849569      4932974.50996256
0.5 62.2396435175844      5164955.56470239
0.5 37.2253481913988      67485151.7631408
0.5 37.1342025123284      68317436.9421162
1    47.8116211975847      19307885.8309833
1    35.5793299130182      84608354.6836839
1    35.7772253228728      82294117.3424372
0.5 62.9109099849569      4932974.50996256
0.5 62.2396435175844      5164955.56470239
0.5 37.2253481913988      67485151.7631408
0.5 37.1342025123284      68317436.9421162
```

```
1    47.8116211975847    19307885.8309833
1    35.5793299130182    84608354.6836839
1    35.7772253228728    82294117.3424372
0.5  62.9109099849569    4932974.50996256
0.5  62.2396435175844    5164955.56470239
0.5  37.2253481913988    67485151.7631408
0.5  37.1342025123284    68317436.9421162
1    47.8116211975847    19307885.8309833
1    35.5793299130182    84608354.6836839
1    35.7772253228728    82294117.3424372
0.5  62.9109099849569    4932974.50996256
0.5  62.2396435175844    5164955.56470239
0.5  37.2253481913988    67485151.7631408
0.5  37.1342025123284    68317436.9421162
1    47.8116211975847    19307885.8309833
1    35.5793299130182    84608354.6836839
1    35.7772253228728    82294117.3424372
0.5  62.9109099849569    4932974.50996256
0.5  62.2396435175844    5164955.56470239
0.5  37.2253481913988    67485151.7631408
0.5  37.1342025123284    68317436.9421162
1    47.8116211975847    19307885.8309833
1    35.5793299130182    84608354.6836839
1    35.7772253228728    82294117.3424372
0.5  62.9109099849569    4932974.50996256
0.5  62.2396435175844    5164955.56470239
0.5  37.2253481913988    67485151.7631408
0.5  37.1342025123284    68317436.9421162
1    47.8116211975847    19307885.8309833
1    35.5793299130182    84608354.6836839
1    35.7772253228728    82294117.3424372
0.5  62.9109099849569    4932974.50996256
0.5  62.2396435175844    5164955.56470239
0.5  37.2253481913988    67485151.7631408
0.5  37.1342025123284    68317436.9421162
1    46.4623533962284    22279018.4615184
0.5  61.0258912730212    5699429.914408
0.5  60.0989586913725    6152722.61388507
1    46.4623533962284    22279018.4615184
0.5  61.0258912730212    5699429.914408
0.5  60.0989586913725    6152722.61388507
1    46.4623533962284    22279018.4615184
0.5  61.0258912730212    5699429.914408
0.5  60.0989586913725    6152722.61388507
1    46.4623533962284    22279018.4615184
0.5  61.0258912730212    5699429.914408
0.5  60.0989586913725    6152722.61388507
1    46.4623533962284    22279018.4615184
0.5  61.0258912730212    5699429.914408
0.5  60.0989586913725    6152722.61388507
1    46.4623533962284    22279018.4615184
0.5  61.0258912730212    5699429.914408
0.5  60.0989586913725    6152722.61388507
1    46.4623533962284    22279018.4615184
0.5  61.0258912730212    5699429.914408
0.5  60.0989586913725    6152722.61388507
1    46.4623533962284    22279018.4615184
0.5  61.0258912730212    5699429.914408
0.5  60.0989586913725    6152722.61388507
1    46.4623533962284    22279018.4615184
0.5  61.0258912730212    5699429.914408
```

```
0.5 60.0989586913725      6152722.61388507
1   46.4623533962284      22279018.4615184
0.5 61.0258912730212      5699429.914408
0.5 60.0989586913725      6152722.61388507
1   46.4623533962284      22279018.4615184
0.5 61.0258912730212      5699429.914408
0.5 60.0989586913725      6152722.61388507
1   46.4623533962284      22279018.4615184
0.5 61.0258912730212      5699429.914408
0.5 60.0989586913725      6152722.61388507

];


DataP4=[1   58.8994561685063      6805274.50157922
0.5 61.2569759516093      5592735.82058889
1   42.5447678796853      34607583.8297449
1   44.3684368962769      28056366.6506288
0.5 61.3098074314208      5568680.63795527
1   58.8994561685063      6805274.50157922
0.5 61.2569759516093      5592735.82058889
1   42.5447678796853      34607583.8297449
1   44.3684368962769      28056366.6506288
0.5 61.3098074314208      5568680.63795527
1   58.8994561685063      6805274.50157922
0.5 61.2569759516093      5592735.82058889
1   42.5447678796853      34607583.8297449
1   44.3684368962769      28056366.6506288
0.5 61.3098074314208      5568680.63795527
1   58.8994561685063      6805274.50157922
0.5 61.2569759516093      5592735.82058889
1   42.5447678796853      34607583.8297449
1   44.3684368962769      28056366.6506288
0.5 61.3098074314208      5568680.63795527
1   58.8994561685063      6805274.50157922
0.5 61.2569759516093      5592735.82058889
1   42.5447678796853      34607583.8297449
1   44.3684368962769      28056366.6506288
0.5 61.3098074314208      5568680.63795527
1   58.8994561685063      6805274.50157922
0.5 61.2569759516093      5592735.82058889
1   42.5447678796853      34607583.8297449
1   44.3684368962769      28056366.6506288
0.5 61.3098074314208      5568680.63795527
1   58.8994561685063      6805274.50157922
0.5 61.2569759516093      5592735.82058889
1   42.5447678796853      34607583.8297449
1   44.3684368962769      28056366.6506288
0.5 61.3098074314208      5568680.63795527
1   58.8994561685063      6805274.50157922
0.5 61.2569759516093      5592735.82058889
1   42.5447678796853      34607583.8297449
1   44.3684368962769      28056366.6506288
0.5 61.3098074314208      5568680.63795527
1   58.8994561685063      6805274.50157922
0.5 61.2569759516093      5592735.82058889
1   42.5447678796853      34607583.8297449
1   44.3684368962769      28056366.6506288
0.5 61.3098074314208      5568680.63795527
0.5 69.8693165191763      3601034.54942235
0.5 71.8482693639623      3311599.7265311
```

```
0.5  69.0406853137464      3732256.57591286
0.5  66.500126263035  4176566.13857651
0.5  69.8693165191763      3601034.54942235
0.5  71.8482693639623      3311599.7265311
0.5  69.0406853137464      3732256.57591286
0.5  66.500126263035  4176566.13857651
0.5  69.8693165191763      3601034.54942235
1    67.5008688741871      3993545.9280625
1    70.0636715908721      3571150.03191316
1    69.9908116468448      3582314.27215596
1    70.0253589607856      3577014.83675995
1    69.9969715746962      3581368.5945754
1    70.0913605565667      3566919.4508751
1    70.1121545528832      3563746.74073921
0.5  74.2565992936875      2999726.11860515
0.5  74.085132494472  3020602.56651052
0.5  71.0922847019632      3418372.28345572
0.5  69.8693165191763      3601034.54942235
1    67.5008688741871      3993545.9280625
1    70.0636715908721      3571150.03191316
1    69.9908116468448      3582314.27215596
1    70.0253589607856      3577014.83675995
1    69.9969715746962      3581368.5945754
1    70.0913605565667      3566919.4508751
1    70.1121545528832      3563746.74073921
0.5  74.2565992936875      2999726.11860515
0.5  74.085132494472  3020602.56651052
0.5  71.0922847019632      3418372.28345572
0.5  69.8693165191763      3601034.54942235
1    67.5008688741871      3993545.9280625
1    70.0636715908721      3571150.03191316
1    69.9908116468448      3582314.27215596
1    70.0253589607856      3577014.83675995
1    69.9969715746962      3581368.5945754
1    70.0913605565667      3566919.4508751
1    70.1121545528832      3563746.74073921
0.5  74.2565992936875      2999726.11860515
0.5  74.085132494472  3020602.56651052
0.5  71.0922847019632      3418372.28345572
1    47.3934577932878      20174839.7522824
1    65.8866035381933      4294329.97286048
0.5  71.2342409398877      3397976.49305574
1    67.1055430588824      4064541.73187302
1    71.0149642525944      3429550.12322718
1    70.1192327783428      3562667.61641143
1    66.9790220184167      4087618.60186114
1    67.3382455269314      4022549.36369605
1    67.5821154403866      3979160.22929321
1    71.0030973799226      3431269.97088661
0.5  76.6526406684916      2727127.29665858
0.5  74.3295921440053      2990897.4568502
0.5  65.9482943402822      4282289.95321429
1    47.3934577932878      20174839.7522824
1    65.8866035381933      4294329.97286048
0.5  71.2342409398877      3397976.49305574
1    67.1055430588824      4064541.73187302
1    71.0149642525944      3429550.12322718
1    70.1192327783428      3562667.61641143
1    66.9790220184167      4087618.60186114
1    67.3382455269314      4022549.36369605
1    67.5821154403866      3979160.22929321
```

```
1    71.0030973799226       3431269.97088661
0.5  76.6526406684916       2727127.29665858
0.5  74.3295921440053       2990897.4568502
0.5  65.9482943402822       4282289.95321429
1    47.3934577932878       20174839.7522824
1    65.8866035381933       4294329.97286048
0.5  71.2342409398877       3397976.49305574
1    67.1055430588824       4064541.73187302
1    71.0149642525944       3429550.12322718
1    70.1192327783428       3562667.61641143
1    66.9790220184167       4087618.60186114
1    67.3382455269314       4022549.36369605
1    67.5821154403866       3979160.22929321
1    71.0030973799226       3431269.97088661
0.5  76.6526406684916       2727127.29665858
0.5  74.3295921440053       2990897.4568502
0.5  65.9482943402822       4282289.95321429
1    47.3934577932878       20174839.7522824
1    65.8866035381933       4294329.97286048
0.5  71.2342409398877       3397976.49305574
1    67.1055430588824       4064541.73187302
1    71.0149642525944       3429550.12322718
1    70.1192327783428       3562667.61641143
1    66.9790220184167       4087618.60186114
1    67.3382455269314       4022549.36369605
1    67.5821154403866       3979160.22929321
1    71.0030973799226       3431269.97088661
0.5  76.6526406684916       2727127.29665858
0.5  74.3295921440053       2990897.4568502
0.5  65.9482943402822       4282289.95321429
1    37.6707189603938       63589077.904494
1    48.190968177313 18559822.3285534
1    71.2286443522726       3398777.51493037
1    46.3071101844495       22654979.5606621
1    46.3271704976026       22605972.4046332
1    71.3866290926499       3376262.05765601
1    46.3431695038957       22566978.151031
1    71.3959542695662       3374939.28820999
1    46.3551019631063       22537947.8029306
1    71.3414627670503       3382678.65928567
1    71.4432777365421       3368237.12966494
0.5  78.9548831493413       2495455.2567017
0.5  81.1547967766676       2297969.12418088
1    37.6707189603938       63589077.904494
1    48.190968177313 18559822.3285534
1    71.2286443522726       3398777.51493037
1    46.3071101844495       22654979.5606621
1    46.3271704976026       22605972.4046332
1    71.3866290926499       3376262.05765601
1    46.3431695038957       22566978.151031
1    71.3959542695662       3374939.28820999
1    46.3551019631063       22537947.8029306
1    71.3414627670503       3382678.65928567
1    71.4432777365421       3368237.12966494
0.5  78.9548831493413       2495455.2567017
0.5  81.1547967766676       2297969.12418088
1    37.6707189603938       63589077.904494
1    48.190968177313 18559822.3285534
1    71.2286443522726       3398777.51493037
1    46.3071101844495       22654979.5606621
1    46.3271704976026       22605972.4046332
```

```
1   71.3866290926499    3376262.05765601
1   46.3431695038957    22566978.151031
1   71.3959542695662    3374939.28820999
1   46.3551019631063    22537947.8029306
1   71.3414627670503    3382678.65928567
1   71.4432777365421    3368237.12966494
0.5 78.9548831493413    2495455.2567017
0.5 81.1547967766676    2297969.12418088
];


timesP1=8760;

length1=length(DataP1);

disp('Train P1')

for j=1:1:timesP1

for i=1:1:length1
    k=i;
    if i<length1
        D=((DataP1(k,1)/DataP1(k,3))+neqp)^(a*(Su-Se)/(DataP1(k,2)-Se));

        neqp=(DataP1(k,1)/DataP1(k,3)+neqp)^((DataP1(k+1,2)-
Se)/(DataP1(k,2)-Se));
    else
        D=((DataP1(k,1)/DataP1(k,3))+neqp)^(a*(Su-Se)/(DataP1(k,2)-Se));

        neqps=neqp;

        neqp=(DataP1(k,1)/DataP1(k,3)+neqp)^((DataP1(1,2)-Se)/(DataP1(k,2)-
Se));
    end

    Dt(i,1)=j;
    Dt(i,2)=D;

end

    Ds(j,1)=j;
    Ds(j,2)=D;
    Ds(j,3)=1;

    if D>1
        return
    else
    end

end

neqp=(DataP1(length1,1)/DataP1(length1,3)+neqps)^((DataP2(1,2)-
Se)/(DataP1(length1,2)-Se));

timesP=timesP1;
timesP2=10950;
```

```matlab
length2=length(DataP2);

disp('Train P2')

for j=1:1:timesP2

for i=1:1:length2
    k=i;
    if i<length2
        D=((DataP2(k,1)/DataP2(k,3))+neqp)^(a*(Su-Se)/(DataP2(k,2)-Se));

        neqp=(DataP2(k,1)/DataP2(k,3)+neqp)^((DataP2(k+1,2)-
Se)/(DataP2(k,2)-Se));
    else
        D=((DataP2(k,1)/DataP2(k,3))+neqp)^(a*(Su-Se)/(DataP2(k,2)-Se));

        neqps=neqp;

        neqp=(DataP2(k,1)/DataP2(k,3)+neqp)^((DataP2(1,2)-Se)/(DataP2(k,2)-
Se));
    end

    Dt(i,1)=j;
    Dt(i,2)=D;

end

    Ds(timesP+j,1)=j+timesP;
    Ds(timesP+j,2)=D;
    Ds(timesP+j,3)=2;

    if D>1
        return
    else
    end

end

neqp=(DataP2(length2,1)/DataP2(length2,3)+neqps)^((DataP3(1,2)-
Se)/(DataP2(length2,2)-Se));

timesP=timesP1+timesP2;
timesP3=9125;

length3=length(DataP3);

disp('Train P3')

for j=1:1:timesP3

for i=1:1:length3
    k=i;
    if i<length3
        D=((DataP3(k,1)/DataP3(k,3))+neqp)^(a*(Su-Se)/(DataP3(k,2)-Se));

        neqp=(DataP3(k,1)/DataP3(k,3)+neqp)^((DataP3(k+1,2)-
Se)/(DataP3(k,2)-Se));
```

```matlab
    else
        D=((DataP3(k,1)/DataP3(k,3))+neqp)^(a*(Su-Se)/(DataP3(k,2)-Se));

        neqps=neqp;

        neqp=(DataP3(k,1)/DataP3(k,3)+neqp)^((DataP3(1,2)-Se)/(DataP3(k,2)-
Se));
    end

    Dt(i,1)=j;
    Dt(i,2)=D;

end

    Ds(timesP+j,1)=j+timesP;
    Ds(timesP+j,2)=D;
    Ds(timesP+j,3)=3;

    if D>1
        return
    else
    end

end

neqp=(DataP3(length3,1)/DataP3(length3,3)+neqps)^((DataP4(1,2)-
Se)/(DataP3(length3,2)-Se));

timesP=timesP1+timesP2+timesP3;
timesP4=13870;

length4=length(DataP4);

disp('Train P4')

for j=1:1:timesP4

for i=1:1:length4
    k=i;
    if i<length4
        D=((DataP4(k,1)/DataP4(k,3))+neqp)^(a*(Su-Se)/(DataP4(k,2)-Se));

        neqp=(DataP4(k,1)/DataP4(k,3)+neqp)^((DataP4(k+1,2)-
Se)/(DataP4(k,2)-Se));
    else
        D=((DataP4(k,1)/DataP4(k,3))+neqp)^(a*(Su-Se)/(DataP4(k,2)-Se));

        neqps=neqp;

        neqp=(DataP4(k,1)/DataP4(k,3)+neqp)^((DataP4(1,2)-Se)/(DataP4(k,2)-
Se));
    end

    Dt(i,1)=j;
    Dt(i,2)=D;

end
```

```matlab
        Ds(timesP+j,1)=j+timesP;
        Ds(timesP+j,2)=D;
        Ds(timesP+j,3)=4;

        if D>1
            return
        else
        end

end

neqp=(DataP4(length4,1)/DataP4(length4,3)+neqps)^((DataP4(1,2)-
Se)/(DataP4(length4,2)-Se));

plot(Ds(:,1),Ds(:,2))

%C(:,1:2)
```

Case 3

```
clc
clear

%Assumptions:

%The script simply runs a matrix named "Data" in a loop. Hence, it is
%important to ensure that the correct sequence is added by cyclically going
%through the matrix.

%Ds is a parameter which saves the damage state per time the script has
%gone through the matrix.
%Dt is a parameter which saves the damage during the running of the matrix.

%a=3 is used, as the damage concept depends upon the stress amplitude, not
%the range, whereas (Si-Se) can be divided or multiplied by 2 to go from
%range to amp, or amp to range respectively.

a=3;

Su=350; %Ultimate tensile strength
Se=34.392105; %SigmaL (lower fatigue limit for variable amplitude)


%Data input, "cycles, range, capacity" columns from excel. The length
%depends upon the

DataP1=[0.5 46.226096098071 22854197.8721825
0.5 48.5867765945837    17816058.6895933
0.5 35.4641611398813     85991121.7073122
0.5 46.226096098071 22854197.8721825
0.5 48.5867765945837    17816058.6895933
0.5 35.4641611398813     85991121.7073122
0.5 46.226096098071 22854197.8721825
0.5 48.5867765945837    17816058.6895933
0.5 35.4641611398813     85991121.7073122
0.5 46.226096098071 22854197.8721825
0.5 48.5867765945837    17816058.6895933
0.5 35.4641611398813     85991121.7073122
0.5 46.226096098071 22854197.8721825
0.5 48.5867765945837    17816058.6895933
0.5 35.4641611398813     85991121.7073122
0.5 46.226096098071 22854197.8721825
0.5 48.5867765945837    17816058.6895933
0.5 35.4641611398813     85991121.7073122
0.5 46.226096098071 22854197.8721825
0.5 48.5867765945837    17816058.6895933
0.5 35.4641611398813     85991121.7073122
0.5 46.226096098071 22854197.8721825
0.5 48.5867765945837    17816058.6895933
0.5 35.4641611398813     85991121.7073122
0.5 46.226096098071 22854197.8721825
0.5 48.5867765945837    17816058.6895933
0.5 35.4641611398813     85991121.7073122
0.5 57.1380858972918     7920883.65398595
0.5 57.2939509937503     7813726.59327422
0.5 54.2590854887917     10257457.4453844
```

```
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918    7920883.65398595
0.5 57.2939509937503    7813726.59327422
0.5 54.2590854887917    10257457.4453844
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918    7920883.65398595
0.5 57.2939509937503    7813726.59327422
0.5 54.2590854887917    10257457.4453844
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918    7920883.65398595
0.5 57.2939509937503    7813726.59327422
0.5 54.2590854887917    10257457.4453844
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918    7920883.65398595
0.5 57.2939509937503    7813726.59327422
0.5 54.2590854887917    10257457.4453844
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918    7920883.65398595
0.5 57.2939509937503    7813726.59327422
0.5 54.2590854887917    10257457.4453844
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918    7920883.65398595
0.5 57.2939509937503    7813726.59327422
0.5 54.2590854887917    10257457.4453844
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918    7920883.65398595
0.5 57.2939509937503    7813726.59327422
0.5 54.2590854887917    10257457.4453844
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918    7920883.65398595
0.5 57.2939509937503    7813726.59327422
0.5 54.2590854887917    10257457.4453844
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918    7920883.65398595
0.5 57.2939509937503    7813726.59327422
0.5 54.2590854887917    10257457.4453844
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918    7920883.65398595
0.5 57.2939509937503    7813726.59327422
0.5 54.2590854887917    10257457.4453844
0.5 53.935068489208 10569292.0733368
0.5 57.1380858972918    7920883.65398595
0.5 57.2939509937503    7813726.59327422
0.5 54.2590854887917    10257457.4453844
0.5 53.935068489208 10569292.0733368
];

neqp=0;

DataP2=[0.5 51.9506764991708    12748116.8545779
0.5 52.6703800807873    11900626.1007275
0.5 51.9506764991708    12748116.8545779
0.5 52.6703800807873    11900626.1007275
0.5 51.9506764991708    12748116.8545779
0.5 52.6703800807873    11900626.1007275
0.5 51.9506764991708    12748116.8545779
0.5 52.6703800807873    11900626.1007275
0.5 51.9506764991708    12748116.8545779
0.5 52.6703800807873    11900626.1007275
0.5 51.9506764991708    12748116.8545779
```

```
0.5 52.6703800807873        11900626.1007275
0.5 51.9506764991708        12748116.8545779
0.5 52.6703800807873        11900626.1007275
0.5 51.9506764991708        12748116.8545779
0.5 52.6703800807873        11900626.1007275
0.5 51.9506764991708        12748116.8545779
0.5 52.6703800807873        11900626.1007275
0.5 57.1380858972918        7920883.65398595
0.5 57.2939509937503        7813726.59327422
0.5 57.1380858972918        7920883.65398595
0.5 57.2939509937503        7813726.59327422
0.5 57.1380858972918        7920883.65398595
0.5 57.2939509937503        7813726.59327422
0.5 57.1380858972918        7920883.65398595
0.5 57.2939509937503        7813726.59327422
0.5 57.1380858972918        7920883.65398595
0.5 57.2939509937503        7813726.59327422
0.5 57.1380858972918        7920883.65398595
0.5 57.2939509937503        7813726.59327422
0.5 57.1380858972918        7920883.65398595
0.5 57.2939509937503        7813726.59327422
0.5 57.1380858972918        7920883.65398595
0.5 57.2939509937503        7813726.59327422
0.5 57.1380858972918        7920883.65398595
0.5 57.2939509937503        7813726.59327422
0.5 57.1380858972918        7920883.65398595
0.5 57.2939509937503        7813726.59327422
0.5 57.1380858972918        7920883.65398595
0.5 57.2939509937503        7813726.59327422
];

DataP3=[1    47.8116211975847     19307885.8309833
1    35.5793299130182     84608354.6836839
1    35.7772253228728     82294117.3424372
0.5 62.9109099849569     4932974.50996256
0.5 62.2396435175844     5164955.56470239
0.5 37.2253481913988     67485151.7631408
0.5 37.1342025123284     68317436.9421162
1    47.8116211975847     19307885.8309833
1    35.5793299130182     84608354.6836839
1    35.7772253228728     82294117.3424372
0.5 62.9109099849569     4932974.50996256
0.5 62.2396435175844     5164955.56470239
0.5 37.2253481913988     67485151.7631408
0.5 37.1342025123284     68317436.9421162
1    47.8116211975847     19307885.8309833
1    35.5793299130182     84608354.6836839
1    35.7772253228728     82294117.3424372
0.5 62.9109099849569     4932974.50996256
0.5 62.2396435175844     5164955.56470239
0.5 37.2253481913988     67485151.7631408
0.5 37.1342025123284     68317436.9421162
1    47.8116211975847     19307885.8309833
1    35.5793299130182     84608354.6836839
1    35.7772253228728     82294117.3424372
0.5 62.9109099849569     4932974.50996256
0.5 62.2396435175844     5164955.56470239
0.5 37.2253481913988     67485151.7631408
0.5 37.1342025123284     68317436.9421162
```

```
1    47.8116211975847    19307885.8309833
1    35.5793299130182    84608354.6836839
1    35.7772253228728    82294117.3424372
0.5  62.9109099849569    4932974.50996256
0.5  62.2396435175844    5164955.56470239
0.5  37.2253481913988    67485151.7631408
0.5  37.1342025123284    68317436.9421162
1    47.8116211975847    19307885.8309833
1    35.5793299130182    84608354.6836839
1    35.7772253228728    82294117.3424372
0.5  62.9109099849569    4932974.50996256
0.5  62.2396435175844    5164955.56470239
0.5  37.2253481913988    67485151.7631408
0.5  37.1342025123284    68317436.9421162
1    47.8116211975847    19307885.8309833
1    35.5793299130182    84608354.6836839
1    35.7772253228728    82294117.3424372
0.5  62.9109099849569    4932974.50996256
0.5  62.2396435175844    5164955.56470239
0.5  37.2253481913988    67485151.7631408
0.5  37.1342025123284    68317436.9421162
1    47.8116211975847    19307885.8309833
1    35.5793299130182    84608354.6836839
1    35.7772253228728    82294117.3424372
0.5  62.9109099849569    4932974.50996256
0.5  62.2396435175844    5164955.56470239
0.5  37.2253481913988    67485151.7631408
0.5  37.1342025123284    68317436.9421162
1    47.8116211975847    19307885.8309833
1    35.5793299130182    84608354.6836839
1    35.7772253228728    82294117.3424372
0.5  62.9109099849569    4932974.50996256
0.5  62.2396435175844    5164955.56470239
0.5  37.2253481913988    67485151.7631408
0.5  37.1342025123284    68317436.9421162
1    46.4623533962284    22279018.4615184
0.5  61.0258912730212    5699429.914408
0.5  60.0989586913725    6152722.61388507
1    46.4623533962284    22279018.4615184
0.5  61.0258912730212    5699429.914408
0.5  60.0989586913725    6152722.61388507
1    46.4623533962284    22279018.4615184
0.5  61.0258912730212    5699429.914408
0.5  60.0989586913725    6152722.61388507
1    46.4623533962284    22279018.4615184
0.5  61.0258912730212    5699429.914408
0.5  60.0989586913725    6152722.61388507
1    46.4623533962284    22279018.4615184
0.5  61.0258912730212    5699429.914408
0.5  60.0989586913725    6152722.61388507
1    46.4623533962284    22279018.4615184
0.5  61.0258912730212    5699429.914408
0.5  60.0989586913725    6152722.61388507
1    46.4623533962284    22279018.4615184
0.5  61.0258912730212    5699429.914408
0.5  60.0989586913725    6152722.61388507
1    46.4623533962284    22279018.4615184
0.5  61.0258912730212    5699429.914408
0.5  60.0989586913725    6152722.61388507
1    46.4623533962284    22279018.4615184
0.5  61.0258912730212    5699429.914408
```

```
0.5 60.0989586913725     6152722.61388507
1   46.4623533962284     22279018.4615184
0.5 61.0258912730212     5699429.914408
0.5 60.0989586913725     6152722.61388507
1   46.4623533962284     22279018.4615184
0.5 61.0258912730212     5699429.914408
0.5 60.0989586913725     6152722.61388507
1   46.4623533962284     22279018.4615184
0.5 61.0258912730212     5699429.914408
0.5 60.0989586913725     6152722.61388507

];


DataP4=[1    58.8994561685063     6805274.50157922
0.5 61.2569759516093     5592735.82058889
1   42.5447678796853     34607583.8297449
1   44.3684368962769     28056366.6506288
0.5 61.3098074314208     5568680.63795527
1   58.8994561685063     6805274.50157922
0.5 61.2569759516093     5592735.82058889
1   42.5447678796853     34607583.8297449
1   44.3684368962769     28056366.6506288
0.5 61.3098074314208     5568680.63795527
1   58.8994561685063     6805274.50157922
0.5 61.2569759516093     5592735.82058889
1   42.5447678796853     34607583.8297449
1   44.3684368962769     28056366.6506288
0.5 61.3098074314208     5568680.63795527
1   58.8994561685063     6805274.50157922
0.5 61.2569759516093     5592735.82058889
1   42.5447678796853     34607583.8297449
1   44.3684368962769     28056366.6506288
0.5 61.3098074314208     5568680.63795527
1   58.8994561685063     6805274.50157922
0.5 61.2569759516093     5592735.82058889
1   42.5447678796853     34607583.8297449
1   44.3684368962769     28056366.6506288
0.5 61.3098074314208     5568680.63795527
1   58.8994561685063     6805274.50157922
0.5 61.2569759516093     5592735.82058889
1   42.5447678796853     34607583.8297449
1   44.3684368962769     28056366.6506288
0.5 61.3098074314208     5568680.63795527
1   58.8994561685063     6805274.50157922
0.5 61.2569759516093     5592735.82058889
1   42.5447678796853     34607583.8297449
1   44.3684368962769     28056366.6506288
0.5 61.3098074314208     5568680.63795527
1   58.8994561685063     6805274.50157922
0.5 61.2569759516093     5592735.82058889
1   42.5447678796853     34607583.8297449
1   44.3684368962769     28056366.6506288
0.5 61.3098074314208     5568680.63795527
1   58.8994561685063     6805274.50157922
0.5 61.2569759516093     5592735.82058889
1   42.5447678796853     34607583.8297449
1   44.3684368962769     28056366.6506288
0.5 61.3098074314208     5568680.63795527
1   37.6707189603938     63589077.904494
1   48.190968177313 18559822.3285534
```

```
1    71.2286443522726      3398777.51493037
1    46.3071101844495      22654979.5606621
1    46.3271704976026      22605972.4046332
1    71.3866290926499      3376262.05765601
1    46.3431695038957      22566978.151031
1    71.3959542695662      3374939.28820999
1    46.3551019631063      22537947.8029306
1    71.3414627670503      3382678.65928567
1    71.4432777365421      3368237.12966494
0.5  78.9548831493413      2495455.2567017
0.5  81.1547967766676      2297969.12418088
1    37.6707189603938      63589077.904494
1    48.190968177313 18559822.3285534
1    71.2286443522726      3398777.51493037
1    46.3071101844495      22654979.5606621
1    46.3271704976026      22605972.4046332
1    71.3866290926499      3376262.05765601
1    46.3431695038957      22566978.151031
1    71.3959542695662      3374939.28820999
1    46.3551019631063      22537947.8029306
1    71.3414627670503      3382678.65928567
1    71.4432777365421      3368237.12966494
0.5  78.9548831493413      2495455.2567017
0.5  81.1547967766676      2297969.12418088
1    37.6707189603938      63589077.904494
1    48.190968177313 18559822.3285534
1    71.2286443522726      3398777.51493037
1    46.3071101844495      22654979.5606621
1    46.3271704976026      22605972.4046332
1    71.3866290926499      3376262.05765601
1    46.3431695038957      22566978.151031
1    71.3959542695662      3374939.28820999
1    46.3551019631063      22537947.8029306
1    71.3414627670503      3382678.65928567
1    71.4432777365421      3368237.12966494
0.5  78.9548831493413      2495455.2567017
0.5  81.1547967766676      2297969.12418088
1    47.3934577932878      20174839.7522824
1    65.8866035381933      4294329.97286048
0.5  71.2342409398877      3397976.49305574
1    67.1055430588824      4064541.73187302
1    71.0149642525944      3429550.12322718
1    70.1192327783428      3562667.61641143
1    66.9790220184167      4087618.60186114
1    67.3382455269314      4022549.36369605
1    67.5821154403866      3979160.22929321
1    71.0030973799226      3431269.97088661
0.5  76.6526406684916      2727127.29665858
0.5  74.3295921440053      2990897.4568502
0.5  65.9482943402822      4282289.95321429
1    47.3934577932878      20174839.7522824
1    65.8866035381933      4294329.97286048
0.5  71.2342409398877      3397976.49305574
1    67.1055430588824      4064541.73187302
1    71.0149642525944      3429550.12322718
1    70.1192327783428      3562667.61641143
1    66.9790220184167      4087618.60186114
1    67.3382455269314      4022549.36369605
1    67.5821154403866      3979160.22929321
1    71.0030973799226      3431269.97088661
0.5  76.6526406684916      2727127.29665858
```

```
0.5   74.3295921440053       2990897.4568502
0.5   65.9482943402822       4282289.95321429
1     47.3934577932878        20174839.7522824
1     65.8866035381933        4294329.97286048
0.5   71.2342409398877        3397976.49305574
1     67.1055430588824        4064541.73187302
1     71.0149642525944        3429550.12322718
1     70.1192327783428        3562667.61641143
1     66.9790220184167        4087618.60186114
1     67.3382455269314        4022549.36369605
1     67.5821154403866        3979160.22929321
1     71.0030973799226        3431269.97088661
0.5   76.6526406684916        2727127.29665858
0.5   74.3295921440053        2990897.4568502
0.5   65.9482943402822        4282289.95321429
1     47.3934577932878        20174839.7522824
1     65.8866035381933        4294329.97286048
0.5   71.2342409398877        3397976.49305574
1     67.1055430588824        4064541.73187302
1     71.0149642525944        3429550.12322718
1     70.1192327783428        3562667.61641143
1     66.9790220184167        4087618.60186114
1     67.3382455269314        4022549.36369605
1     67.5821154403866        3979160.22929321
1     71.0030973799226        3431269.97088661
0.5   76.6526406684916        2727127.29665858
0.5   74.3295921440053        2990897.4568502
0.5   65.9482943402822        4282289.95321429
0.5   69.8693165191763        3601034.54942235
1     67.5008688741871        3993545.9280625
1     70.0636715908721        3571150.03191316
1     69.9908116468448        3582314.27215596
1     70.0253589607856        3577014.83675995
1     69.9969715746962        3581368.5945754
1     70.0913605565667        3566919.4508751
1     70.1121545528832        3563746.74073921
0.5   74.2565992936875        2999726.11860515
0.5   74.085132494472 3020602.56651052
0.5   71.0922847019632        3418372.28345572
0.5   69.8693165191763        3601034.54942235
1     67.5008688741871        3993545.9280625
1     70.0636715908721        3571150.03191316
1     69.9908116468448        3582314.27215596
1     70.0253589607856        3577014.83675995
1     69.9969715746962        3581368.5945754
1     70.0913605565667        3566919.4508751
1     70.1121545528832        3563746.74073921
0.5   74.2565992936875        2999726.11860515
0.5   74.085132494472 3020602.56651052
0.5   71.0922847019632        3418372.28345572
0.5   69.8693165191763        3601034.54942235
1     67.5008688741871        3993545.9280625
1     70.0636715908721        3571150.03191316
1     69.9908116468448        3582314.27215596
1     70.0253589607856        3577014.83675995
1     69.9969715746962        3581368.5945754
1     70.0913605565667        3566919.4508751
1     70.1121545528832        3563746.74073921
0.5   74.2565992936875        2999726.11860515
0.5   74.085132494472 3020602.56651052
0.5   71.0922847019632        3418372.28345572
```

```
0.5 69.8693165191763      3601034.54942235
0.5 71.8482693639623      3311599.7265311
0.5 69.0406853137464      3732256.57591286
0.5 66.500126263035 4176566.13857651
0.5 69.8693165191763      3601034.54942235
0.5 71.8482693639623      3311599.7265311
0.5 69.0406853137464      3732256.57591286
0.5 66.500126263035 4176566.13857651
1   58.8994561685063      6805274.50157922
0.5 61.2569759516093      5592735.82058889
1   42.5447678796853      34607583.8297449
1   44.3684368962769      28056366.6506288
0.5 61.3098074314208      5568680.63795527
1   58.8994561685063      6805274.50157922
0.5 61.2569759516093      5592735.82058889
1   42.5447678796853      34607583.8297449
1   44.3684368962769      28056366.6506288
0.5 61.3098074314208      5568680.63795527
1   58.8994561685063      6805274.50157922
0.5 61.2569759516093      5592735.82058889
1   42.5447678796853      34607583.8297449
1   44.3684368962769      28056366.6506288
0.5 61.3098074314208      5568680.63795527
1   58.8994561685063      6805274.50157922
0.5 61.2569759516093      5592735.82058889
1   42.5447678796853      34607583.8297449
1   44.3684368962769      28056366.6506288
0.5 61.3098074314208      5568680.63795527
1   58.8994561685063      6805274.50157922
0.5 61.2569759516093      5592735.82058889
1   42.5447678796853      34607583.8297449
1   44.3684368962769      28056366.6506288
0.5 61.3098074314208      5568680.63795527
1   58.8994561685063      6805274.50157922
0.5 61.2569759516093      5592735.82058889
1   42.5447678796853      34607583.8297449
1   44.3684368962769      28056366.6506288
0.5 61.3098074314208      5568680.63795527
1   58.8994561685063      6805274.50157922
0.5 61.2569759516093      5592735.82058889
1   42.5447678796853      34607583.8297449
1   44.3684368962769      28056366.6506288
0.5 61.3098074314208      5568680.63795527
1   58.8994561685063      6805274.50157922
0.5 61.2569759516093      5592735.82058889
1   42.5447678796853      34607583.8297449
1   44.3684368962769      28056366.6506288
0.5 61.3098074314208      5568680.63795527
1   58.8994561685063      6805274.50157922
0.5 61.2569759516093      5592735.82058889
1   42.5447678796853      34607583.8297449
1   44.3684368962769      28056366.6506288
0.5 61.3098074314208      5568680.63795527
0.5 69.8693165191763      3601034.54942235
0.5 71.8482693639623      3311599.7265311
0.5 69.0406853137464      3732256.57591286
0.5 66.500126263035 4176566.13857651
0.5 69.8693165191763      3601034.54942235
0.5 71.8482693639623      3311599.7265311
0.5 69.0406853137464      3732256.57591286
0.5 66.500126263035 4176566.13857651
```

```
0.5 69.8693165191763      3601034.54942235
1   67.5008688741871      3993545.9280625
1   70.0636715908721      3571150.03191316
1   69.9908116468448      3582314.27215596
1   70.0253589607856      3577014.83675995
1   69.9969715746962      3581368.5945754
1   70.0913605565667      3566919.4508751
1   70.1121545528832      3563746.74073921
0.5 74.2565992936875      2999726.11860515
0.5 74.085132494472 3020602.56651052
0.5 71.0922847019632      3418372.28345572
0.5 69.8693165191763      3601034.54942235
1   67.5008688741871      3993545.9280625
1   70.0636715908721      3571150.03191316
1   69.9908116468448      3582314.27215596
1   70.0253589607856      3577014.83675995
1   69.9969715746962      3581368.5945754
1   70.0913605565667      3566919.4508751
1   70.1121545528832      3563746.74073921
0.5 74.2565992936875      2999726.11860515
0.5 74.085132494472 3020602.56651052
0.5 71.0922847019632      3418372.28345572
0.5 69.8693165191763      3601034.54942235
1   67.5008688741871      3993545.9280625
1   70.0636715908721      3571150.03191316
1   69.9908116468448      3582314.27215596
1   70.0253589607856      3577014.83675995
1   69.9969715746962      3581368.5945754
1   70.0913605565667      3566919.4508751
1   70.1121545528832      3563746.74073921
0.5 74.2565992936875      2999726.11860515
0.5 74.085132494472 3020602.56651052
0.5 71.0922847019632      3418372.28345572
1   47.3934577932878      20174839.7522824
1   65.8866035381933      4294329.97286048
0.5 71.2342409398877      3397976.49305574
1   67.1055430588824      4064541.73187302
1   71.0149642525944      3429550.12322718
1   70.1192327783428      3562667.61641143
1   66.9790220184167      4087618.60186114
1   67.3382455269314      4022549.36369605
1   67.5821154403866      3979160.22929321
1   71.0030973799226      3431269.97088661
0.5 76.6526406684916      2727127.29665858
0.5 74.3295921440053      2990897.4568502
0.5 65.9482943402822      4282289.95321429
1   47.3934577932878      20174839.7522824
1   65.8866035381933      4294329.97286048
0.5 71.2342409398877      3397976.49305574
1   67.1055430588824      4064541.73187302
1   71.0149642525944      3429550.12322718
1   70.1192327783428      3562667.61641143
1   66.9790220184167      4087618.60186114
1   67.3382455269314      4022549.36369605
1   67.5821154403866      3979160.22929321
1   71.0030973799226      3431269.97088661
0.5 76.6526406684916      2727127.29665858
0.5 74.3295921440053      2990897.4568502
0.5 65.9482943402822      4282289.95321429
1   47.3934577932878      20174839.7522824
1   65.8866035381933      4294329.97286048
```

```
0.5  71.2342409398877        3397976.49305574
1    67.1055430588824        4064541.73187302
1    71.0149642525944        3429550.12322718
1    70.1192327783428        3562667.61641143
1    66.9790220184167        4087618.60186114
1    67.3382455269314        4022549.36369605
1    67.5821154403866        3979160.22929321
1    71.0030973799226        3431269.97088661
0.5  76.6526406684916        2727127.29665858
0.5  74.3295921440053        2990897.4568502
0.5  65.9482943402822        4282289.95321429
1    47.3934577932878        20174839.7522824
1    65.8866035381933        4294329.97286048
0.5  71.2342409398877        3397976.49305574
1    67.1055430588824        4064541.73187302
1    71.0149642525944        3429550.12322718
1    70.1192327783428        3562667.61641143
1    66.9790220184167        4087618.60186114
1    67.3382455269314        4022549.36369605
1    67.5821154403866        3979160.22929321
1    71.0030973799226        3431269.97088661
0.5  76.6526406684916        2727127.29665858
0.5  74.3295921440053        2990897.4568502
0.5  65.9482943402822        4282289.95321429
1    37.6707189603938        63589077.904494
1    48.190968177313 18559822.3285534
1    71.2286443522726        3398777.51493037
1    46.3071101844495        22654979.5606621
1    46.3271704976026        22605972.4046332
1    71.3866290926499        3376262.05765601
1    46.3431695038957        22566978.151031
1    71.3959542695662        3374939.28820999
1    46.3551019631063        22537947.8029306
1    71.3414627670503        3382678.65928567
1    71.4432777365421        3368237.12966494
0.5  78.9548831493413        2495455.2567017
0.5  81.1547967766676        2297969.12418088
1    37.6707189603938        63589077.904494
1    48.190968177313 18559822.3285534
1    71.2286443522726        3398777.51493037
1    46.3071101844495        22654979.5606621
1    46.3271704976026        22605972.4046332
1    71.3866290926499        3376262.05765601
1    46.3431695038957        22566978.151031
1    71.3959542695662        3374939.28820999
1    46.3551019631063        22537947.8029306
1    71.3414627670503        3382678.65928567
1    71.4432777365421        3368237.12966494
0.5  78.9548831493413        2495455.2567017
0.5  81.1547967766676        2297969.12418088
1    37.6707189603938        63589077.904494
1    48.190968177313 18559822.3285534
1    71.2286443522726        3398777.51493037
1    46.3071101844495        22654979.5606621
1    46.3271704976026        22605972.4046332
1    71.3866290926499        3376262.05765601
1    46.3431695038957        22566978.151031
1    71.3959542695662        3374939.28820999
1    46.3551019631063        22537947.8029306
1    71.3414627670503        3382678.65928567
1    71.4432777365421        3368237.12966494
```

```matlab
0.5 78.9548831493413     2495455.2567017
0.5 81.1547967766676     2297969.12418088
];


timesP1=8760;

length1=length(DataP1);

disp('Train P1')

for j=1:1:timesP1

for i=1:1:length1
    k=i;
    if i<length1
        D=((DataP1(k,1)/DataP1(k,3))+neqp)^(a*(Su-Se)/(DataP1(k,2)-Se));

        neqp=(DataP1(k,1)/DataP1(k,3)+neqp)^((DataP1(k+1,2)-
Se)/(DataP1(k,2)-Se));
    else
        D=((DataP1(k,1)/DataP1(k,3))+neqp)^(a*(Su-Se)/(DataP1(k,2)-Se));

        neqps=neqp;

        neqp=(DataP1(k,1)/DataP1(k,3)+neqp)^((DataP1(1,2)-Se)/(DataP1(k,2)-
Se));
    end

    Dt(i,1)=j;
    Dt(i,2)=D;

end

    Ds(j,1)=j;
    Ds(j,2)=D;
    Ds(j,3)=1;

    if D>1
        return
    else
    end

end

neqp=(DataP1(length1,1)/DataP1(length1,3)+neqps)^((DataP2(1,2)-
Se)/(DataP1(length1,2)-Se));

timesP=timesP1;
timesP2=10950;

length2=length(DataP2);

disp('Train P2')

for j=1:1:timesP2
```

```matlab
for i=1:1:length2
    k=i;
    if i<length2
        D=((DataP2(k,1)/DataP2(k,3))+neqp)^(a*(Su-Se)/(DataP2(k,2)-Se));

        neqp=(DataP2(k,1)/DataP2(k,3)+neqp)^((DataP2(k+1,2)-
Se)/(DataP2(k,2)-Se));
    else
        D=((DataP2(k,1)/DataP2(k,3))+neqp)^(a*(Su-Se)/(DataP2(k,2)-Se));

        neqps=neqp;

        neqp=(DataP2(k,1)/DataP2(k,3)+neqp)^((DataP2(1,2)-Se)/(DataP2(k,2)-
Se));
    end

    Dt(i,1)=j;
    Dt(i,2)=D;

end

    Ds(timesP+j,1)=j+timesP;
    Ds(timesP+j,2)=D;
    Ds(timesP+j,3)=2;

    if D>1
        return
    else
    end

end

neqp=(DataP2(length2,1)/DataP2(length2,3)+neqps)^((DataP3(1,2)-
Se)/(DataP2(length2,2)-Se));

timesP=timesP1+timesP2;
timesP3=9125;

length3=length(DataP3);

disp('Train P3')

for j=1:1:timesP3

for i=1:1:length3
    k=i;
    if i<length3
        D=((DataP3(k,1)/DataP3(k,3))+neqp)^(a*(Su-Se)/(DataP3(k,2)-Se));

        neqp=(DataP3(k,1)/DataP3(k,3)+neqp)^((DataP3(k+1,2)-
Se)/(DataP3(k,2)-Se));
    else
        D=((DataP3(k,1)/DataP3(k,3))+neqp)^(a*(Su-Se)/(DataP3(k,2)-Se));

        neqps=neqp;
```

```
        neqp=(DataP3(k,1)/DataP3(k,3)+neqp)^((DataP3(1,2)-Se)/(DataP3(k,2)-
Se));
    end

    Dt(i,1)=j;
    Dt(i,2)=D;

end

    Ds(timesP+j,1)=j+timesP;
    Ds(timesP+j,2)=D;
    Ds(timesP+j,3)=3;

    if D>1
        return
    else
    end

end

neqp=(DataP3(length3,1)/DataP3(length3,3)+neqps)^((DataP4(1,2)-
Se)/(DataP3(length3,2)-Se));

timesP=timesP1+timesP2+timesP3;
timesP4=6935;

length4=length(DataP4);

disp('Train P4')

for j=1:1:timesP4

for i=1:1:length4
    k=i;
    if i<length4
        D=((DataP4(k,1)/DataP4(k,3))+neqp)^(a*(Su-Se)/(DataP4(k,2)-Se));

        neqp=(DataP4(k,1)/DataP4(k,3)+neqp)^((DataP4(k+1,2)-
Se)/(DataP4(k,2)-Se));
    else
        D=((DataP4(k,1)/DataP4(k,3))+neqp)^(a*(Su-Se)/(DataP4(k,2)-Se));

        neqps=neqp;

        neqp=(DataP4(k,1)/DataP4(k,3)+neqp)^((DataP4(1,2)-Se)/(DataP4(k,2)-
Se));
    end

    Dt(i,1)=j;
    Dt(i,2)=D;

end

    Ds(timesP+j,1)=j+timesP;
    Ds(timesP+j,2)=D;
    Ds(timesP+j,3)=4;
```

```matlab
    if D>1
        return
    else
    end

end

neqp=(DataP4(length4,1)/DataP4(length4,3)+neqps)^((DataP4(1,2)-
Se)/(DataP4(length4,2)-Se));

plot(Ds(:,1),Ds(:,2))

%C(:,1:2)
```

neqp=(DataP4(length4,1)/DataP4(length4,3)+neqps)^((DataP4(1,2)-

Appendix E

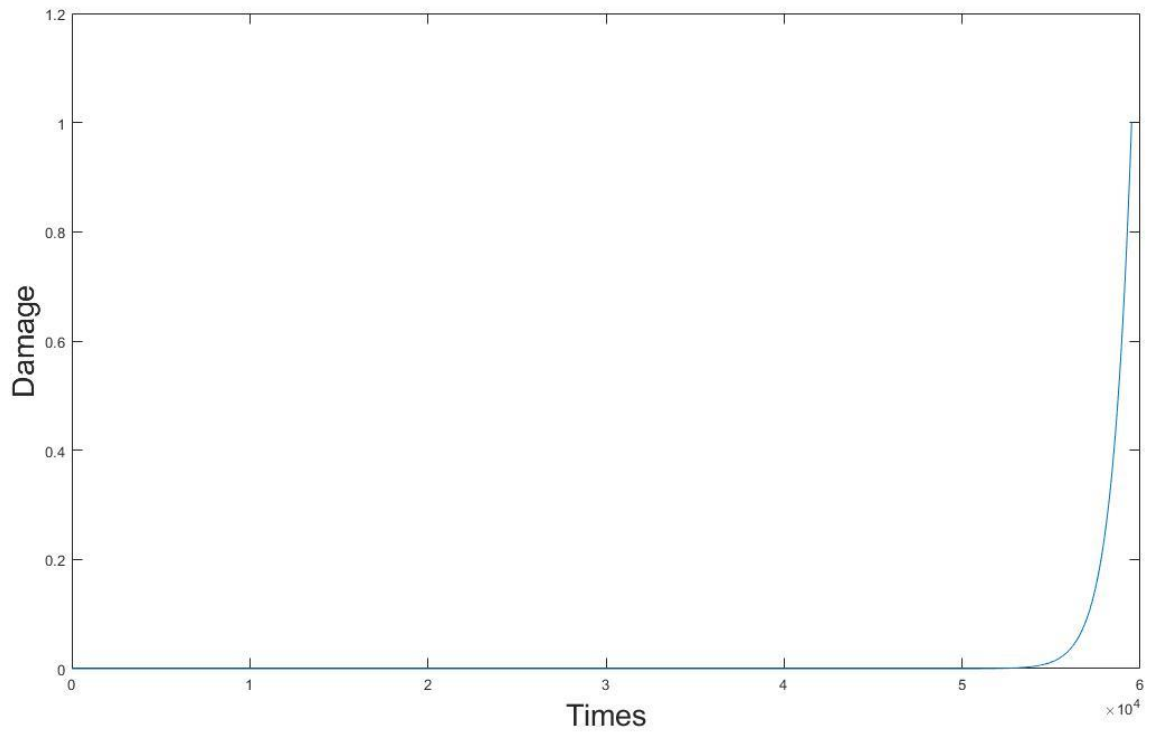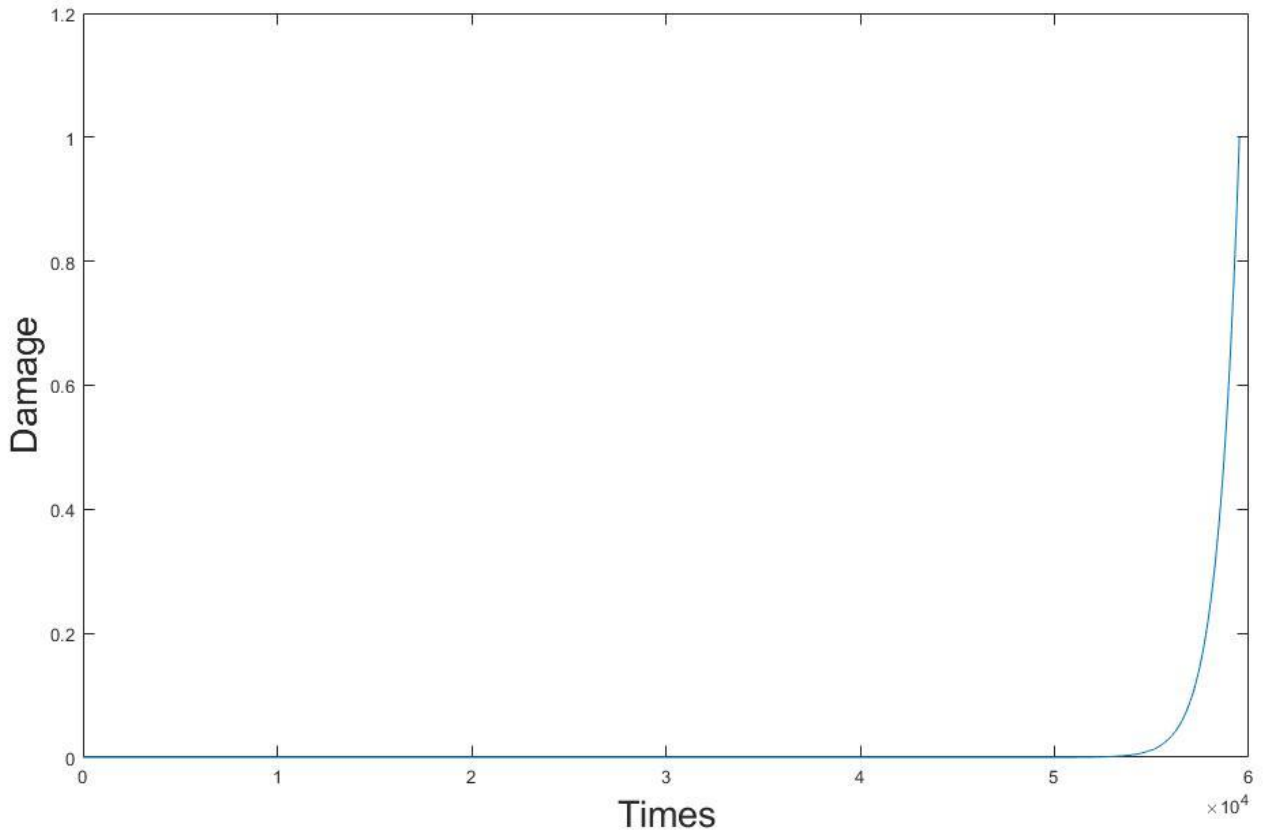## Nonlinear Fatigue Damage Model: MATLAB Plot

Case 1

Figure E.1 : Case 1

Case 2

Figure E.2: Case 2

Case 3

Figure E.3: Case 3