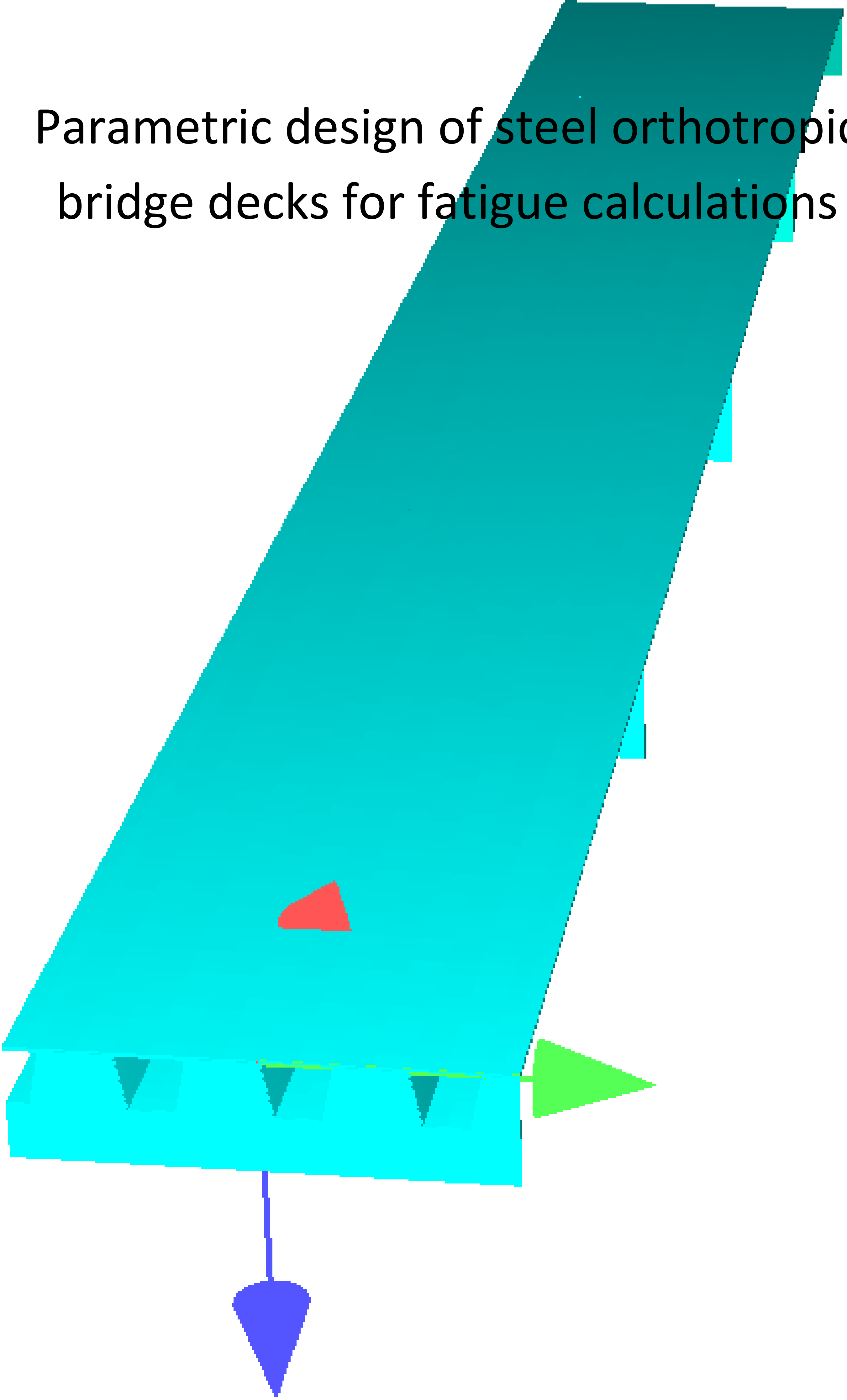
Parametric design of steel orthotropic bridge decks for fatigue calculations



Parametric design of steel orthotropic bridge decks for fatigue calculations

by

Maarten van der Wateren
E: maarten.wateren@gmail.com
T: +31611835813
Student number: 4277945

Company
Royal HaskoningDHV
George Hintzenweg 85
3068 AX Rotterdam

TU Delft supervisor
Prof.dr. M.Velkjkovic
E: m.veljkovic@tudelft.nl
T: +3115278584

Company supervisor
Ir. J. Voermans
E: Jurgen.voermans@rhdhv.com
T: +31 883488350

Company Supervisor
Ir. T. Boeters
E: Ton.boeters@rhdhv.com
T: +31 883489388

Contents

1 Introduction

In partial fulfilment of the requirements for the degree of Master of Science at the TU Delft University of Technology an internship can be done. In cooperation with the engineering firm Royal HaskoningDHV I started an internship of ten weeks. This report is the result of the activities done during the internship.

Currently the transport & planning (transport & logistics) group from Royal HaskoningDHV makes their pre-design for steel orthotropic bridge decks by using rules of thumb and experience. In the final design-stage this is verified whereby the force distribution is checked with the help of 3D finite element software (FEM). In this stage only small design changes can be made and therefore the design cannot be optimized to its full extend. The final design is highly dependent on experienced engineers and the rules of thumb. Since the rules of thumb are safe estimates this might result in an overdesigned design with more material than necessary, making it economically and ecologically unattractive. Furthermore, the fatigue calculation, which is always done in the final design stage, is very extensive and makes doing the calculations very expensive and labour intensive.

This process is then repeated for each separate bridge even though steel orthotropic bridge decks are build up in the same way most of the time. Therefore, a parametric model is desired where dimensions can be altered to fit as many bridges as possible. The model should include all the load calculations as given by the norms in the Eurocode.

To make the most of the ten-week internship some preliminary constraints have been decided on together with the company. SOFiSTik will be used as FEM-software and excel will be used for the pre- and post-processing. Furthermore, the choice is made to make a model on detail level excluding all global loads. Further explanation on the model can be found in chapter 3.

Before the model can be made the structural requirements must be known. These will follow from the Eurocode and will be explained in chapter 2. Here both the strength and fatigue requirements will be considered. An already existing tool will be used for the fatigue requirements. At last this tool will be connected to the model to make it fully automated to go from input parameters to a damage value per detail as output.

The previously mentioned information results in the following (sub)questions:

1. What are the requirements of a bridge in terms of fatigue verification?
2. How can the bridge deck be modelled best for a quick but thorough calculation?
3. How does the program go from a model to the eventual damage results?
4. Can this whole process be automated?

2 Structural requirements

Each structure has requirements that must be met to be considered as safe. These requirements are stated in the Eurocode norms and depend on the type of structure. In the case of a steel orthotropic bridge deck there are strength and fatigue requirements that must be verified. In this report only the fatigue requirements will be considered. An explanation on fatigue in general can be found in this chapter together with the requirements from the norm. These include the load cases as stated in NEN-EN1991-2 and the fatigue details, fatigue locations and detail categories as stated in NEN-EN1993-1-9 and the national annex.

NEN-EN1991-2 Eurocode 1: belastingen op constructies – Deel 2: Verkeersbelasting op bruggen
NEN-EN1993-1-9 Eurocode 3: Ontwerp en verekening van staal constructies -Deel 1-9: vermoeiing

2.1 Fatigue in general

Fatigue is one of the main causes of damage in metallic members together with corrosion and wear. When a member is subjected to repeated cyclic loading, and therefore fluctuating stresses, fatigue may occur. Due to the repetitive stress fluctuations cracks may form at specific locations which can grow progressively. This will result in a loss of resistance in time. The main critical places for fatigue in a structure are connections and joints. Mainly the connections made by means of welding are critical and will be examined in this report.

At the toes of the welds stress concentrations occur due to a change in geometry from the attachments. In the same spots discontinuities occur in the material due to the welding process. These stress concentrations combined with the discontinuities result in fatigue crack propagation which will result in a loss of resistance.

To determine the damage on a structure due to fatigue, equation (2.1) can be used.

$$D_{f,tot} = \sum \frac{n_i}{N_i} \tag{2.1}$$

Here D_f is the fatigue damage done, n_i is the number of occurrences of $\Delta\sigma_i$ and N_i is the amount of cycles needed for failure at stresses of $\Delta\sigma_i$. To determine the N_i a damage accumulation scheme is needed. The simplest form, known as Miner's rule, is shown in Figure 1. When $D_f > 1.0$ the structure fails.

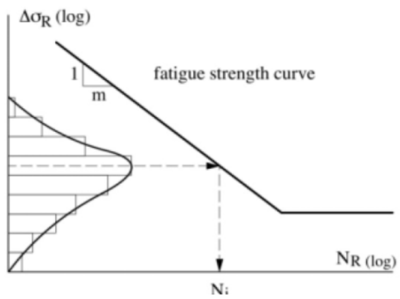


Figure 1 Damage accumulation scheme (EN1993-1-9)

A more advanced scheme includes a constant amplitude fatigue limit ($\Delta\sigma_C$) and a cut-off limit ($\Delta\sigma_L$) as is shown in Figure 2. Above the constant amplitude fatigue limit (CAFL) the scheme is still the same. Between the CAFL and the cut-off limit the slope has changed resulting in a lower damage done.

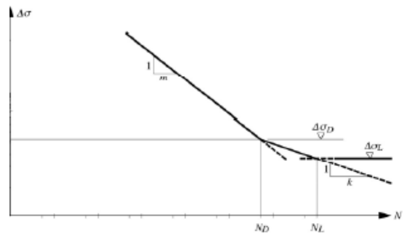


Figure 2 DAS including CAFL (EN1993-1-9)

As shown above the stress range ($\Delta\sigma$) is the main variable needed to calculate the damage of the structure. The stress range can be calculated by using equation (2.2).

$$\Delta\sigma_R = \sigma_{max} - \sigma_{min} \; (including\ signs) \tag{2.2}$$

From the stress range the number of cycles till failure for a stress range can be calculated using equation (2.3).

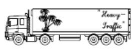
$$N = C * \Delta\sigma_R^{-m} \tag{2.3}$$

Here C is a constant depending on the structural detail and m is the slope coefficient. The results express as straight lines and follow equation when using a logarithmic scale (2.4).

$$\log(\mathrm{N}) = \log(\mathrm{C}) - \mathrm{m} * \log(\Delta\sigma_{\mathrm{R}}) \tag{2.4}$$

2.2 Load cases

In this specific case stresses occur due to traffic driving across the bridge. The Eurocode states which loads must be considered. In the case of a steel orthotropic traffic bridge the fatigue damage will mainly be caused by the moving traffic and load model 4a will be leading for this as is shown in Figure 3. As shown in this figure there are three different wheel types that are considered, namely wheel type A, B and C. The geometrical definition of these wheels can be found in Figure 4. Dispurtion due to the thickness of the deck has to be taken into acount in accordance to the norms.

Type voertuig			Verkeerstype			
Afbeelding van de vrachtwagen	Afstand tussen de assen m	Gelijkwaardige aslast kN	Lange afstand % [a]	Middellange afstand % [a]	Lokaal verkeer % [a]	Wiel-type
	4,5	70 130	20,0	50,0	80,0	A B
	4,20 1,30	70 120 120	5,0	5,0	5,0	A B B
	3,20 5,20 1,30 1,30	70 150 90 90 90	40,0	20,0	5,0	A B C C C
	3,40 6,00 1,80	70 140 90 90	25,0	15,0	5,0	A B C C
	4,80 3,60 4,40 1,30	70 130 90 80 80	10,0	10,0	5,0	A B C C C

[a] Percentage vrachtwagens.

Figure 3 Equivalent standard lorries (NEN-EN 1991-2 tabel NB.6)

When the deck is comprised of an asphalt layer and a steel deck, equation (2.5) can be found according to NEN-EN1991-2 section 4.3.6.

$$x_{tot} = x_{contact} + 2 * t_{asphalt} + t_{deck} \tag{2.5}$$

The combination of different vehicle types mimic a certain type of traffic, short, medium or long distance. This traffic type has to be combined with a certain amount of passing vehicles to obtain a correct assumption of the traffic load. The amount of vehicles passing in a year is called (N_{obs}). An assumption for this is given in Figure 5. This N_{obs} is the number of heavy traffic on 1 lane.

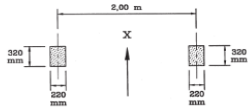
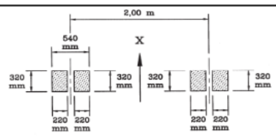
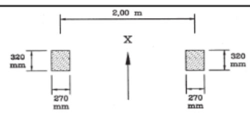
WIEL-/ ASTYPE	OMSCHRIJVING VAN DE GEOMETRIE
A	
B	
C	

Figure 4 Description of wheels and axles (NEN-EN 1991-2 tabel 4.8)

	Verkeerscategorie	$N_{obs,a,ai}$ per jaar en per rijstrook voor zwaar verkeer
1	Autosnelwegen (A-wegen) en wegen met twee of meer rijstroken per rijrichting en met intensief vrachtverkeer	$2,0 \times 10^6$
2	(Auto)wegen met gemiddeld vrachtverkeer (zoals N-wegen)	$0,5 \times 10^6$
3	Wegen met weinig vrachtverkeer	0,125 × 106
4	Wegen met weinig vrachtverkeer en bovendien uitsluitend bestemmingsverkeer	$0,05 \times 10^6$

OPMERKING De aantallen zware voertuigen per jaar en per rijstrook voor zwaar verkeer $N_{obs,a,ai}$ zijn inclusief trend.

Figure 5 Expected heavy traffic per lane per year (NEN-EN1991-2 tabel 4.5)

Since not each driver drives exactly in the center of a lane a frequency ditribution must be used which is shown in Figure 6 in accordance with the Eurocode. This distribution means that 50% of the vehicles (N_{obs}) drive in the center, 18% on either side shifted with 0.1 m and 7% on either side shifted with 0.2 m. These tracks will be refered to in this report as track-2, track-1, track mid, track 1 and track 2.

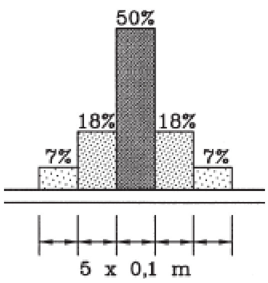


Figure 6 Frequency distribution in transverse direction (NEN-EN 1991-2 figuur 4.6)

2.3 Fatigue details

With all the load cases known the details that might be leading must be determined. To limit the amount of fatigue checks only critical zone 1-3 will be considered as shown in Figure 7. For further simplicity only the case for cross-beams without cut-outs will be considered. This results in the following list of details to be considered.

English descriptions of the details can be found in Appendix A: English detail description.

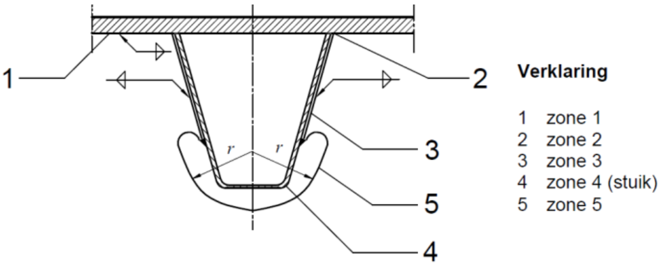
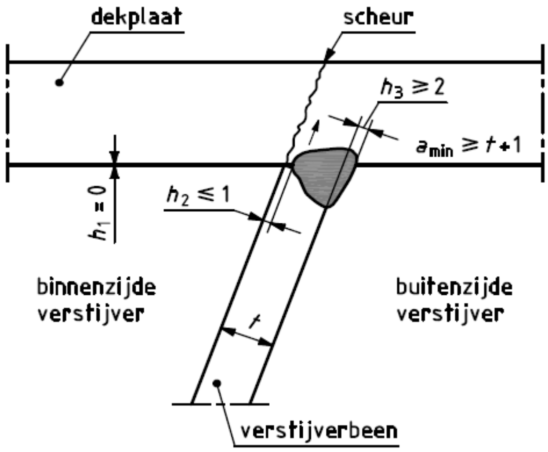
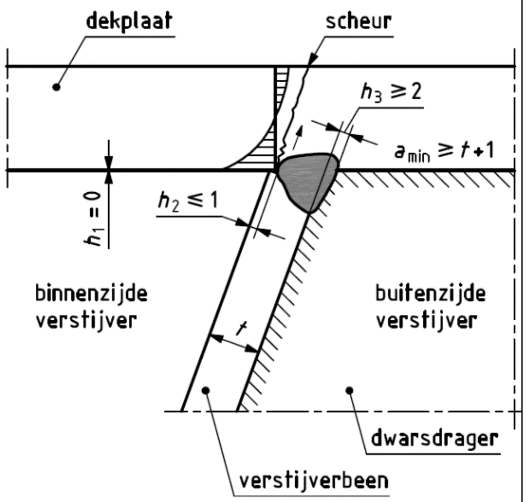


Figure 7 Critical zones fatigue (EN-1993-2 figure 9.1)



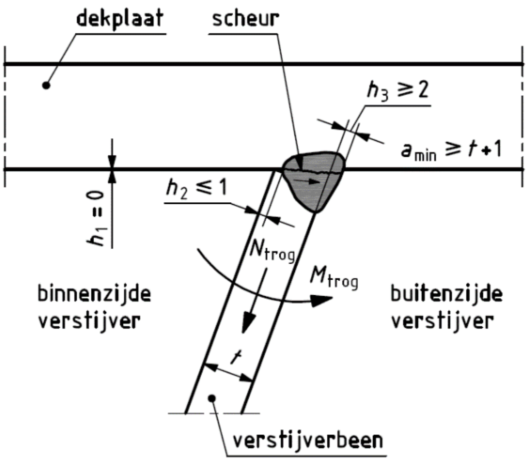
Detail category: 125
Locatie: Scheur in de dekplaat op een locatie tussen de dwarsdragers
Scheurtype: Scheur geïnitieerd vanuit de las tussen de dekplaat en de verstijvers; kan aan beide zijden ontstaan
Scheurgroei: Door de dikte van de dekplaat vanuit de las
Spanningswisseling: Berekend als nominale lokale spanning aan de onderzijde van de dekplaat op het scheurinitiatiepunt, berekend met een 3Dmodel
$\boldsymbol{amin} < t + 1\,mm$
Voorbewerking: Verstijverbeen afschuinen tot een lasopeningshoek van 50°. Bij OP-lassen tot $t \leq 6mm$ geen afschuining
Spleet $h_1 = 0\ mm$; over 10 % van de lengte is $h_1 < 0{,}5\ mm$ toegelaten
MDF $h_2 \leq 1{,}0\ mm$
NDO visueel: 100 %; MT: alle lasaanzetten + 10 % van de laslengte als steekproef te kiezen op basis van de visuele inspectie
Lasgeometrie: De las moet vloeiend aanliggen aan het dek en het verstijverbeen

Figure 8 Fatigue detail 1&2 (in- and outside (NEN-EN1993-2 NB Tabel NB.7 construction detail 1)



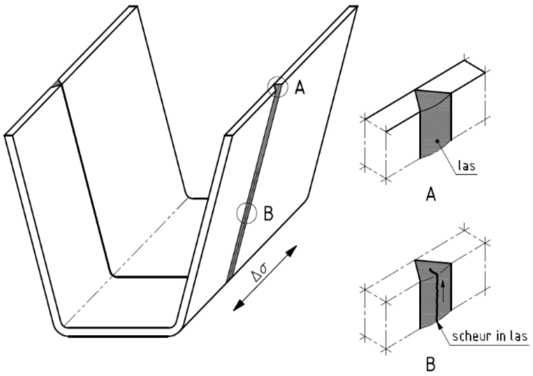
Detail category: 125
Locatie: Dekplaat boven de kruising tussen de verstijverbenen en de dwarsdrager
Scheurtype: Scheur geïnitieerd vanuit de las tussen de dekplaat en de verstijver
Scheurgroei: Door de dikte van de dekplaat vanuit de wortel van de las
Spanningswisseling: Berekend als lokale 'hot spot stress'- spanning aan de onderzijde van de dekplaat op het scheurinitiatiepunt, berekend met een 3D-model of de lokale nominale spanning vermenigvuldigd met een SCF-factor uit een vereenvoudigd 2Dmodel overeenkomstig de aanpak van figuur NB.5
$\boldsymbol{amin} \leq t + 1\,mm$
Voorbewerking: Verstijverbeen afschuinen tot een lasopeningshoek van 50°. Bij OP-lassen tot $t \leq 6\ mm$ geen afschuining
Spleet $h1$ = 0 mm; over 10 % van de lengte is $h_1 \leq 0{,}5\ mm$ toegelaten
MDF $h_2 \leq 1{,}0\ mm$
NDO visueel: 100 %; MT: 100 % waar de las van de verstijver aan de dwarsdrager en dekplaat en de las van de dwarsdrager aan de dekplaat samenkomen (x-y-zaansluiting) en ter plaatse van

Figure 9 Fatigue detail 3 (NEN-EN1993-2 NB Tabel NB.7 construction detail 2)



Detail category: 90 (hand las), 100 (automatisch lasproces)
Locatie: Las tussen het verstijverbeen en de dekplaat op een locatie tussen de dwarsdragers
Scheurtype: Scheur geïnitieerd vanuit de las
Scheurgroei: Door de dikte van de las vanuit de wortel of de teen van de las
Spanningswisseling: Berekend als lokale nominale spanning in het verstijverbeen, berekend met een 3D model
$\boldsymbol{amin} \leq t + 1\,mm$
Voorbewerking: Verstijverbeen afschuinen tot een lasopeningshoek van 50°. Bij OP-lassen tot *t* £ 6 mm geen afschuining
Spleet $h_1 = 0\,mm$; over 10 % van de lengte is $h_2 < 0{,}5\,mm$ toegelaten
MDF $h_2 \leq 1{,}0\,mm$
NDO visueel: 100 %; MT: alle lasaanzetten + 10 % van de laslengte als steekproef te kiezen op basis van de visuele inspectie
Lasgeometrie: De las moet vloeiend aanliggen aan het dek en het verstijverbeen
Lasoverhoogte $h_3 \geq 2$

Figure 10 Fatigue detail 4 (NEN-EN1993-2 NB Tabel NB.7 construction detail 3)



[a] Een hogere categorie is mogelijk volgens detail 1 of detail 5 in tabel 8.3 van EN 1993-1-9 indien eisen worden aangehouden. (Figure 15)

Detail category: 80[a]
Locatie: Fabrieksmatig aangebrachte stompe las bij verlenging van de verstijver
Scheurtype: Vanuit de grondnaad van de las
Scheurgroei: Door de dikte van de las
Spanningswisseling: Berekend als nominale spanning in de bodem van de verstijver
Voorbewerking: Beide einden afgeschuind, V-naad 60°
Toleranties: Uitlijnfout ≤ 0,5 mm
Vooropening: ≥ 4,0 mm
Lassen: Handlas in fabriek
MDF 0 mm; volledige doorlassing
NDO MT en visueel: 100 %; RT: onderzijde verstijver + halve hoogte verstijverbeen vanaf onderzijde verstijver: 100 %
Lasgeometrie: Lassen niet geslepen. Bolling van de las maximaal 20 % van de breedte. Lasaanzetten niet in hoog belaste zones (onderzijde verstijver + halve hoogte verstijverbeen vanaf onderzijde

Figure 11 Fatigue detail 5 (NEN-EN1993-2 NB Tabel NB.7 construction detail 4)

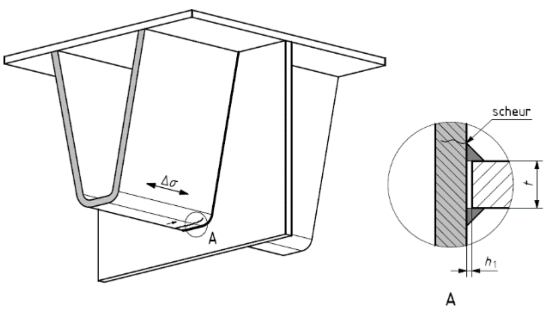


Figure 12 Fatigue detail 6 (NEN-EN1993-2 NB Tabel NB.7 construction detail 7)

Detail category: 125 (mits berekend als hot spot spanning)
Locatie: Verbinding van de verstijver aan de dwarsdrager bij doorgestoken verstijvers zonder uitsparing
Scheurtype: Scheur in de verstijver vanuit de teen van de las
Scheurgroei: Door de dikte van het lijf
Spanningswisselin: Berekend als nominale spanning in de onderkant van de verstijver ter plaatse van de dwarsdrager bij een afstand van onderzijde trog tot onderzijde dwarsdrager $> 500\,mm$. Bij een afstand $\leq 500\,mm$ moet $\Delta\sigma$ zijn berekend als geometrische ‘hot spot stress’-spanning
amin $\geq 5\,mm$ of groter indien noodzakelijk voor de sterkte van de las
Voorbewerking: Spleet $h_1 \leq 1{,}0\,mm$
Lassen: Handlas; lasaanzetten op de koudvervormde delen zijn niet toegelaten
NDO: Visueel en MT 100 %

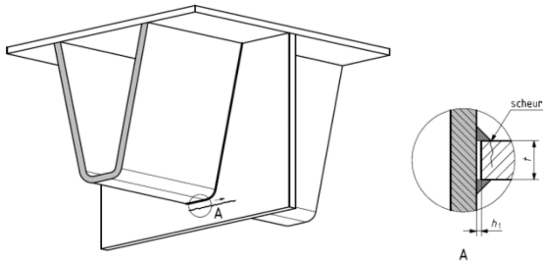


Figure 13 Fatigue detail 7 (NEN-EN1993-2 NB Tabel NB.7 construction detail 9)

Detail category: 80
Locatie: Verbinding van de verstijver aan de dwarsdrager
Scheurtype: Scheur in de dwarsdrager bij doorgestoken verstijver
Scheurgroei: In de lijfplaat van de dwarsdrager vanuit teen van de las
Spanningswisseling: Berekend als nominale spanning in het lijf van de dwarsdrager ter plaatse van de teen van de las door buiging uit het vlak van het lijf van de dwarsdrager als gevolg van de doorbuiging van de verstijver gecombineerd met de vierendeeleffecten in het vlak van de dwarsdrager
***a*min**$\geq$ $5\ mm$, of groter indien noodzakelijk voor sterkte van de las
Voorbewerking: Spleet $h_1 \leq 1{,}0\ mm$
Lassen Handlas; lasaanzetten op de koudvervormde delen zijn niet toegelaten
NDO visueel en MT 100 %

Figure 14 Fatigue detail 8 (EN1993-1-9 Table 8.1)

Detail category: 160
Locatie: Ter plaatse van maximale spanning.
Beschrijving: Platen en stripstaal met randen in leveringstoestand
Eisen: Scherpe randen, oppervlaktefouten en walsfouten zijn verwijderd door slijpen waarbij een gladde overgangszone ontstaat.

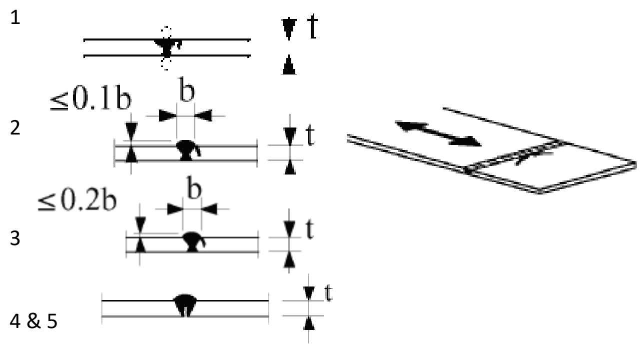


Figure 15 Fatigue detail 9 (EN1993-1-9 Table 8.3)

Detail category: 1) 112, 2) 90, 3) 80, 4) 71, 5) 36
Locatie: Ter plaatse van stompe las
Beschrijving: Stuiklassen in dwarsrichting in platen en strippen.
Eisen: Lasuitloopstukken zijn gebruikt en zijn achteraf verwijderd, plaattranen geslepen in spanningsrichting, gelast aan beide zijden &
1) vlakgeslepen lassen
2) maximale overdikte van 10% van lasbreedte
3) maximale overdikte van 20% van lasbreedte
4) las uitgevoerd aan 1 kant, volledig gecontroleerd door geschikt NDO
5) enkelzijdige las ongecontroleerd

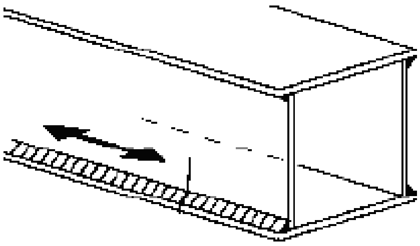


Figure 16 Fatigue detail 10 (EN1993-1-9 Table 8.2)

Detail category: 100
Locatie: Ter plaatse van las in langsrichting
Beschrijving: Handmatig of automatisch of volledig gemechaniseerd gelaste stompe lassen, uitgevoerd aan slechts één kant, in het bijzonder bij kokerprofielen.
Eisen: Tussen flens en lijfplaten is een zeer goede passing vereist. De rand van het lijf is voldoende af te schijnen voor het behoorlijk uitvoeren van een regelmatige penetratie ter plaatse van de laswortel, zonder uitvloeien.

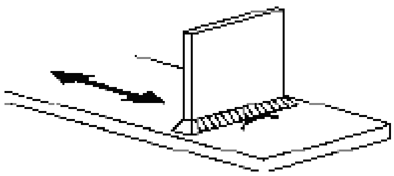


Figure 17 Fatigue detail 11 (EN1993-1-9 Table 8.4)

Detail category: 80
Locatie: Ter plaatse van dwarsdrager en dek verbinding.
Beschrijving: Aangelaste platen in de dwarsrichting gelast op een plaat.
Eisen: De uiteinden van de lassen zijn zorgvuldig geslepen om mogelijk aanwezige randinkartelingen te verwijderen.

2.4 Fatigue locations

Figure 18 shows an overview of the bridge deck including the cross-beams, stiffeners and deck. Here the previously mentioned load cases will "run" accros the deck in X-direction (red).

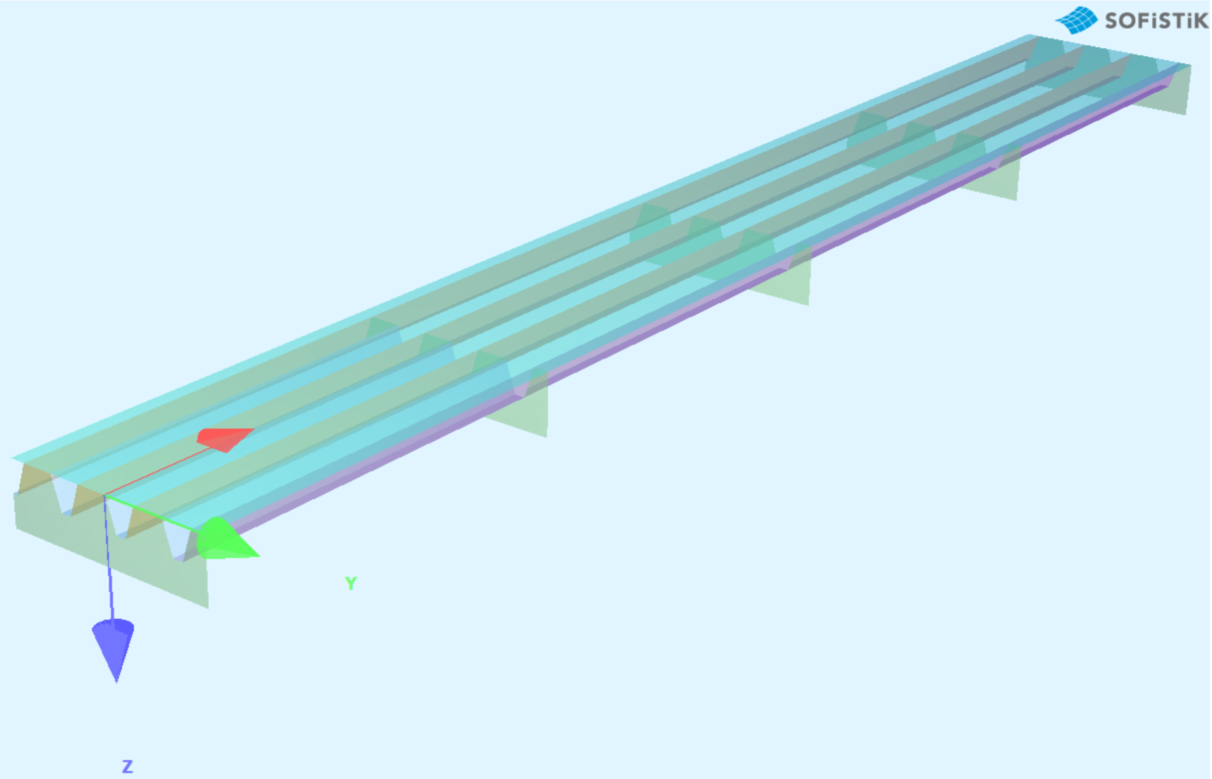


Figure 18 overview of bridge

The necessary stresses will be determined in three different cross sections as shown in Figure 19. Cross section B is not fixed, it's location depends on the butt-weld in the deck and stiffener. Usually this butt-weld lies at a location of three quarters of the cross-beam span (or one quarter depending on the starting point).

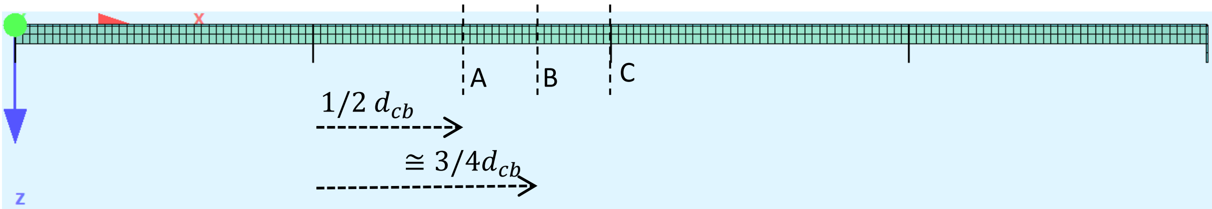


Figure 19 Side-view location for cross sections

Multiple tracks in transverse direction of the model will be analysed and combined using the frequency distribution shown in Figure 6. The location of the center load track depends on the detail that is analysed, therefore multiple load tracks will be analysed and the most damaging combination will be used. The location of the load tracks numbered 1 to 8 can be found in Figure 20.

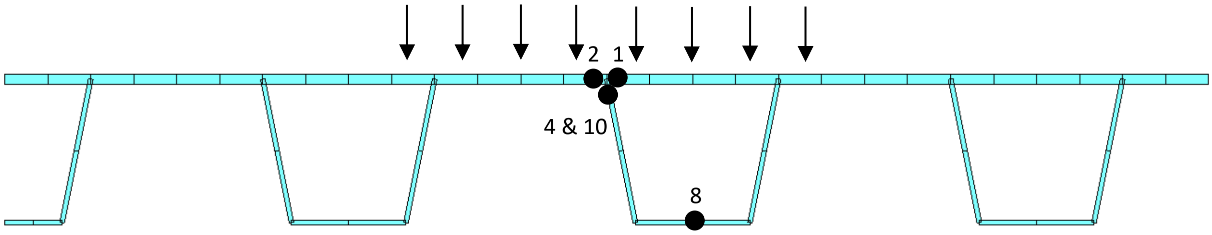


Figure 20 Location for details cross section A

Figure 20 shows cross section A where five details are situated. Details 4 and 10 are at the same location, however detail 4 uses the stress in transverse direction and detail 10 the stress in longitudinal direction. The black arrows represent the centre lines of the different load tracks which are 100 mm apart.

Figure 21 includes 2 details namely the butt-weld details for the deck and stiffeners. These details aren't exactly in the same cross section, usually there is 20-40 cm between the butt-weld locations (in longitudinal direction). This is done to prevent a serious weak spot in the structure.

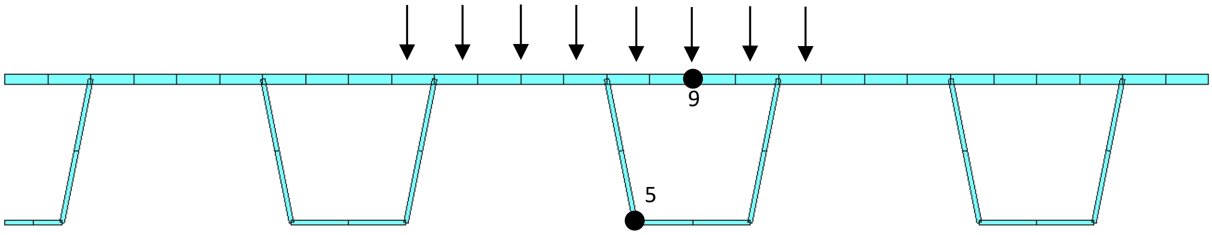


Figure 21 Location for details cross section B

The last 4 details are situated in cross section C shown in Figure 22. Since the stresses for details 3 and 6 must be determined with the hot-spot stress method extra read out points are necessary. Further explanation about the hot-spot stress method can be found in Chapter 3.3.7.

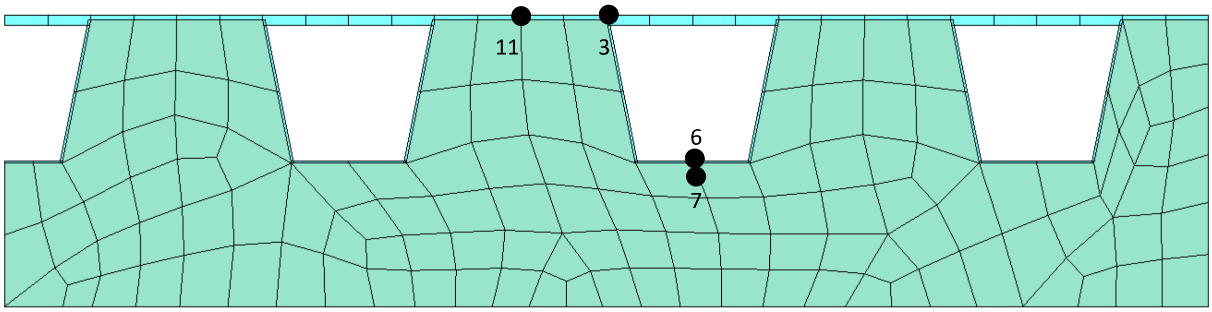


Figure 22 Location for details cross section C

2.5 Detail category

The detail category is the stress range for which a specimen fails after $2 * 10^6$ cycles. With this the rest of the graph can be determined. Figure 23 shows this for different failure categories.
A distinct kink can be seen in the graph at $5 * 10^6$ cycles, this is the constant amplitude fatigue limit (CAFL). There will be no fatigue damage for constant stress ranges that occur below the CAFL. For altering stress ranges damage will occur when below the CAFL.
Stress ranges below the cut-off limit will never result in fatigue damage. Further details about how this graph is made can be found at the start of Chapter 2.0.

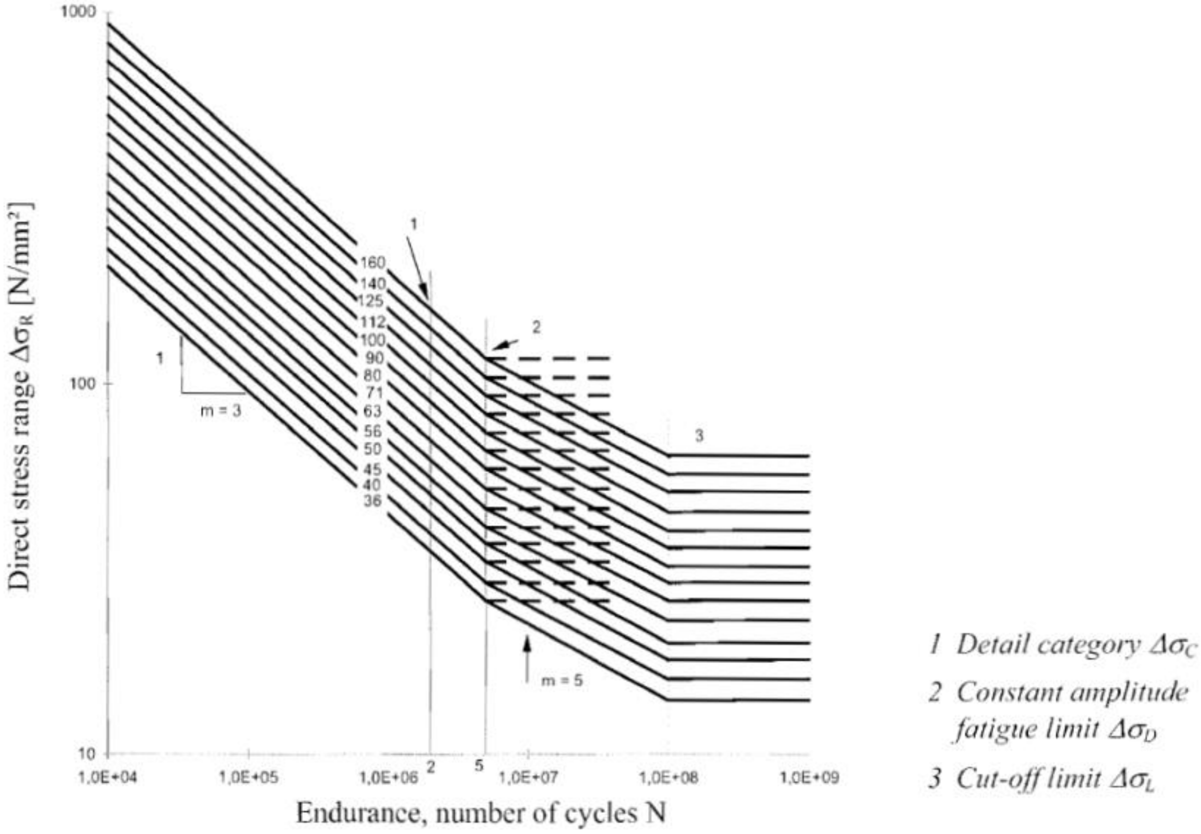


Figure 23 Fatigue strength curves for direct stress ranges (EN1993-1-9 figure 7.1)

2.6 Factors

2.6.1 Dynamic amplification factor

Depending on the location of the vehicle on the deck a dynamic factor should be taken into account if the vehicle is close to an expansion joint. This factor is given by equation (2.6). Here D is the distance between the closest wheel and the expansion joint. The national annex states however that the factor may be taken as a fixed value of 1.15 if a wheel is within 6 meters from a joint. For this model there will be no expansion joint implemented, therefore the factor 1.15 will be used for the whole structure.

$$\Delta\varphi_{\mathrm{fat}} = 1.3\left(1 - \frac{D}{26}\right)\ ;\ \Delta\varphi_{\mathrm{fat}} \geq 1 \tag{2.6}$$

With the previously mentioned criteria many different load configurations must be taken into acount. Three different wheel geometries, five different lorries with different wheel loads and eight “lanes” in transverse direction would result in a minimum of 120 unique load configurations. To use this for each detail would be time consuming so a post calculation will be done to simplify everything. The three wheel geometries will be modelled with a unit load and the eight transverse paths will be used resulting in 24 combinations. These 24 load combinations can be used to make a step model of a moving wheel. When combining these load steps into a graph an influence line is formed. For these 24 different influence lines a post calculation will be done to form the different vehicles stated in Figure 3. More about this post calculation can be found in chapter 5

2.6.2 partial safety factor fatigue strength γ_{MF}

The partial safety factor γ_{Mf} is a factor that is taken over the detail category. It considers the consequences of a possible failure and has 4 different options. It uses one of two methods, the damage tolerant method or the safe-life method.
Choosing either of these methods and combining it with the consequence of failure (low or high) results in the γ_{Mf}-factor as is shown in Table 1.

Table 1 recommended values for partial factors for fatigue strength (NEN-EN 1993-1-9 table NB. 1)

Beoordelingsmethode	Gevolg van het bezwijken	
	Gering gevolg	Groot gevolg
Gebaseerd op het concept van schadetolerant	1,00	1,15
Gebaseerd op het concept van veilige levensduur	1,15	1,35

The national annex NEN-EN1993-NB art 9.3 states that for stiffeners, deck and connection between stiffeners and cross-beams of orthotropic decks the first column (low consequences) can be used. For consequence class 3 (highway bridges) the method of safe life must be used, for lower consequence classes it is allowed to use the damage tolerant method.

3 Model definition

The model has certain requirements that must be met to result in an adequate and correct model. These requirements will be explained starting with requirements for the design of the model. Next the dimensions in longitudinal direction will be explained including boundary conditions. Dimensions in transverse direction and its boundaries will be explained afterwards. Two types of optimization, mesh and load-step optimization, will be explained last.

3.1 Design requirements

The model should include the deck, the stiffeners and the cross-beams including the connections (welds) to include the previously mentioned details. The main girders are not in the scope for this project. The design-tool will only be used for new structures to limit the amount of parameters necessary. All the previously mentioned boundaries and limitations result in a cross section as shown in Figure 24 where the girder will be left out. This would still result in a deck of several tens of meters wide and long. To decrease the size of the model there will be some smart use of symmetry conditions.

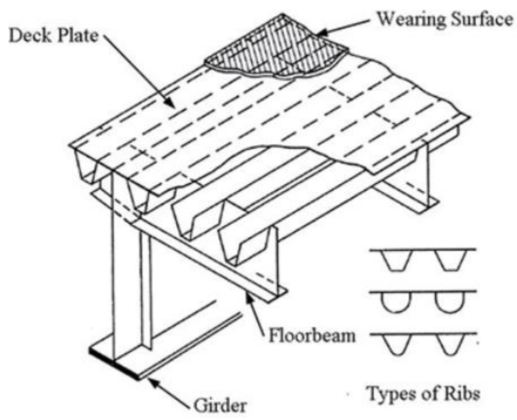


Figure 24 Cross section model

The distance between two wheels on the same axle is 2 meters according to the Eurocode. Due to this the model can be decreased to a width of 2 meters with symmetry conditions at the edge. Using symmetry results in mirroring of the same loads and structural dimensions at the other side of the symmetry axis. This way the load of a car can be modelled using only 2 (front and back) wheels instead of 4 as is shown in Figure 25.

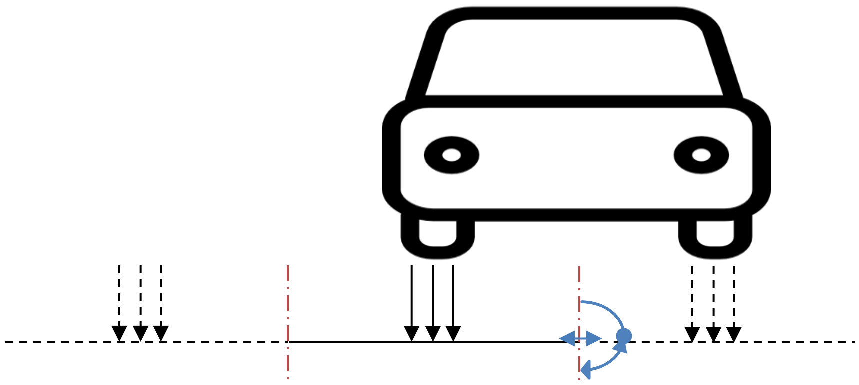


Figure 25 Symmetry conditions

In FEM model's symmetry conditions are formed when fixing the translation and rotation as shown by the blue arrows in Figure 25. How these boundary conditions are implemented will be explained in chapter 3.3.4 after the longitudinal and transverse dimensions have been defined.

3.2 Longitudinal direction

To obtain an alternating stress distribution for the influence lines one span length left of the detail and one span length right is needed. This results in a model with two spans and two extra half spans as shown in Figure 26. The edge conditions at the ends should make sure that there is no kink in the deck, therefore the moment M_y should be fixed. Together with a fixation of U_x this would result in symmetry conditions in longitudinal direction.

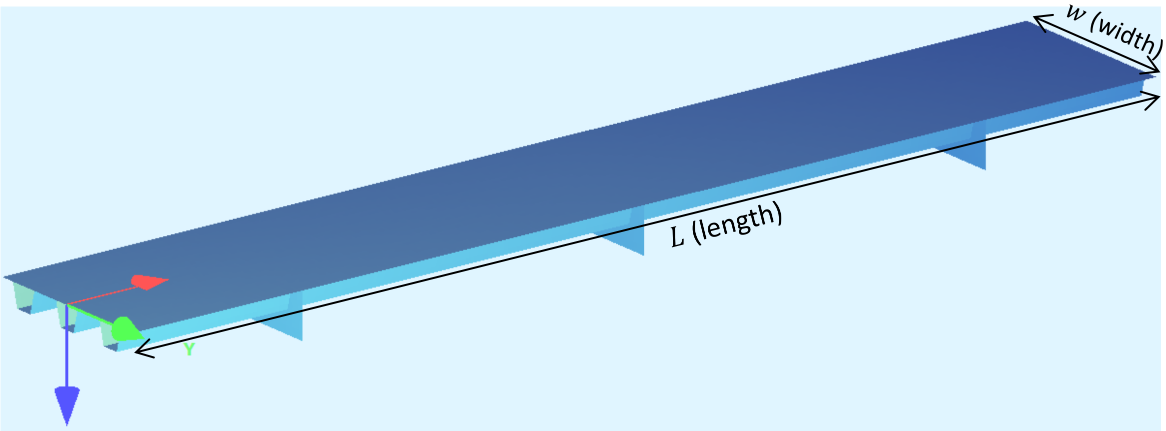


Figure 26 Model with coordinate system

Although this seems correct at first sight some strange results occur in the influence lines (more about how the results have been obtained and analysed in chapter 5. The influence lines show a distinct kink in the stress spectrum as shown for example for detail 5 in Figure 27 including a close-up.

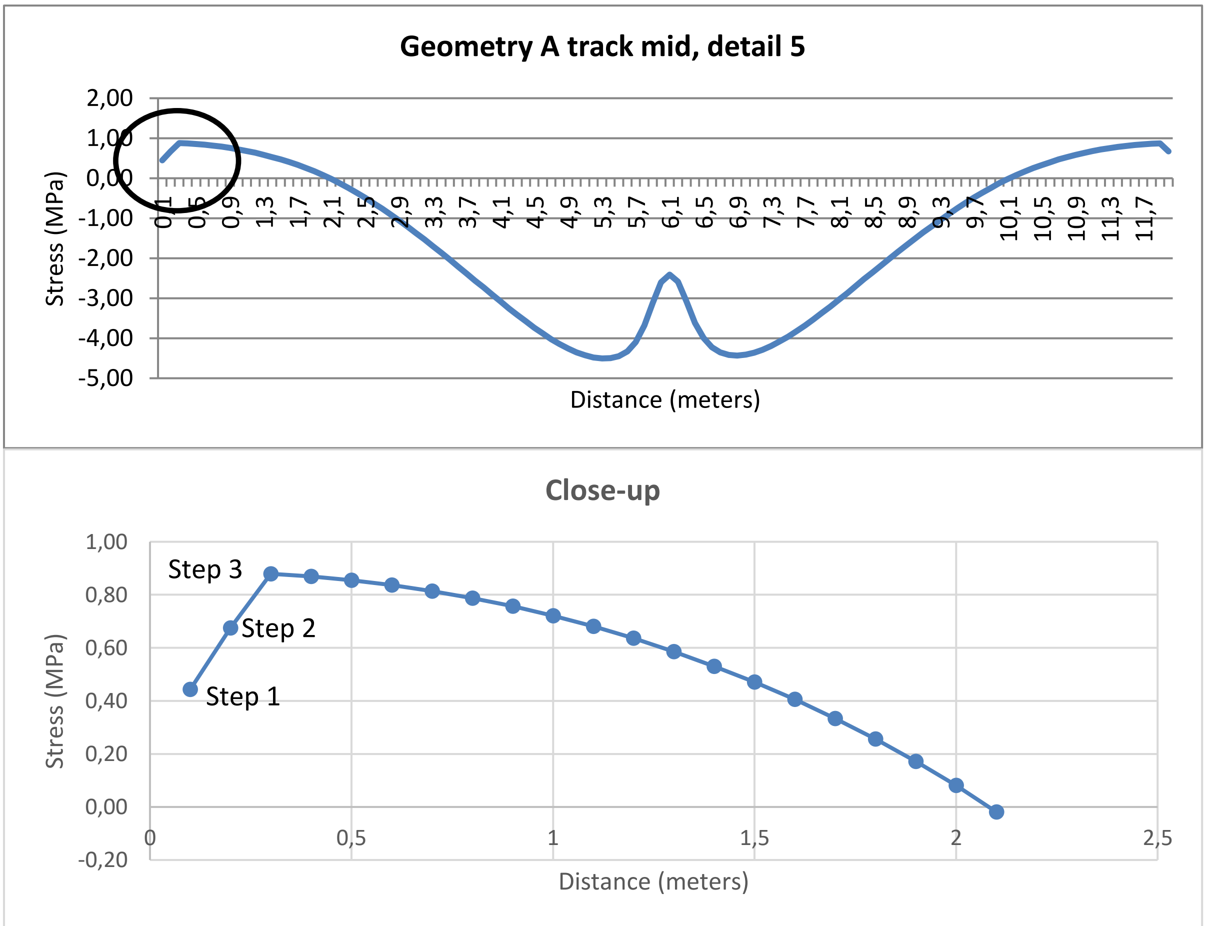


Figure 27 Stress spectrum detail 5, old model

For the first load step the centroid of the area is positioned right at the edge of the model. Therefore only half of the wheel is modelled and due to the symmetry conditions the other half is included as well adding up to one complete wheel load.
The second load step is still close enough to the edge where it is not able to model the whole area of the wheel. Three quarters of the wheel can be modelled and due to symmetry another three quarters of the wheel is modelled. From the third step onwards the whole area can be modelled but due to symmetry a whole extra wheel is modelled. Due to this modification, a problem arises as can be seen in the graphs of Figure 27. Due to the symmetry conditions, the loads are mirrored and for the model it is as if a double load is applied on the same span. This process is shown in Figure 28 where one grey block is the loading area of a wheel. As is shown, for the first load case the modelling is correct, but for the rest a bigger load is modelled than there should be.

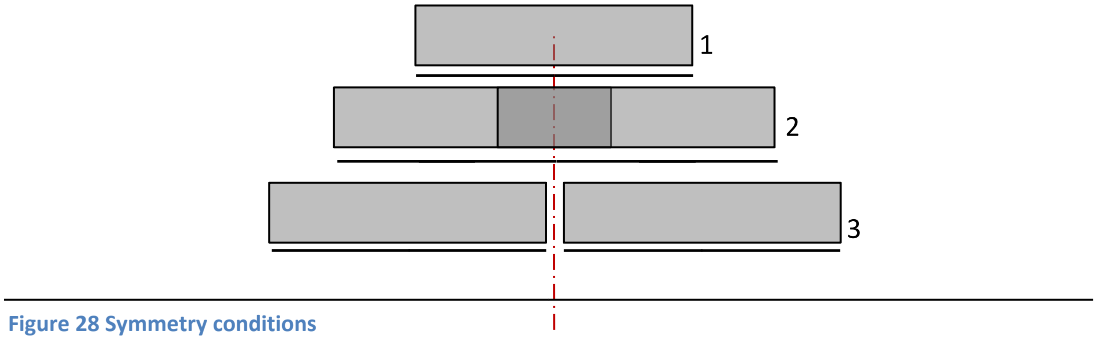


Figure 28 Symmetry conditions

As long as the symmetry conditions are applied this is a problem that cannot be solved. The applied loading has a certain influence length, at a certain distance from the read-out point the stress change is so low that doubling it due to symmetry has no consequences. Since the stress range (range from peak-to-peak) is of interest for the fatigue calculations the location of the first peak needs to be determined. In Figure 27 it seems that the stress will start to decrease before X=0 and after X=12. As a verification, the model has been extended with two extra half spans. The influence line shown in Figure 29 shows the results. As expected no kinks are found thus this model will be used.

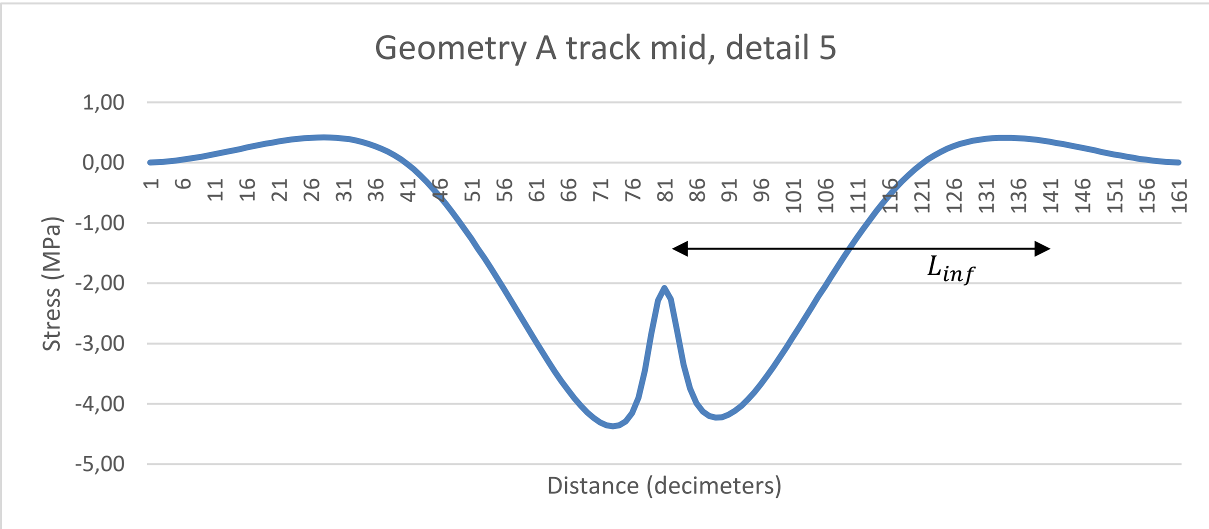


Figure 29 Stress spectrum detail 5, new model

3.3 Transverse direction

The use of symmetry in transverse direction as explained earlier will be used. To make sure the model is conservative the width cannot be more than the distance between the wheels of the vehicles (2m). If it is larger it would mean that the wheels are more than two meters apart resulting in lower stresses and thus an underestimation of the damage. On the other hand, if the width of the model is far below two meters the resulting stresses will be significantly higher than would be realistic. Therefore, a model must be made with a width as close as possible but always less than two meters to be conservative.

With the most common width of 300 mm and a centre to centre distance of 600 mm the model will look like Figure 30.

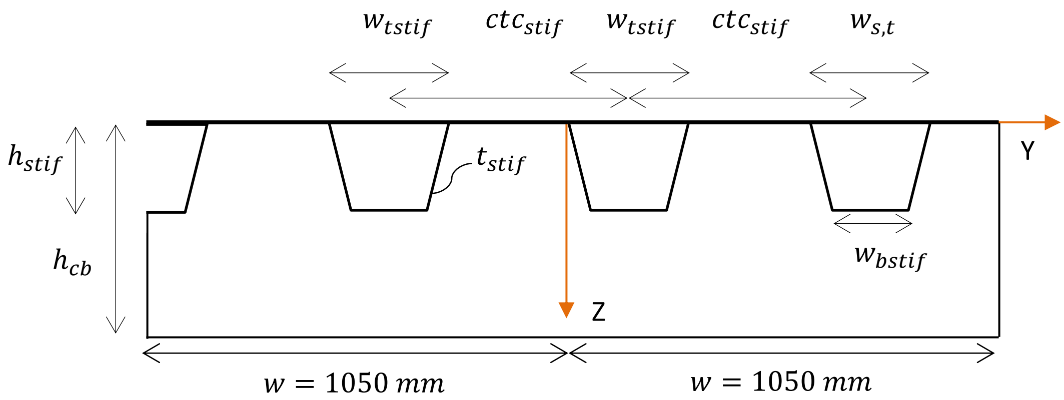


Figure 30 Cross section A-A

The distance from the centre of the model to the sides is $1050\ mm$ in this case. To ensure a minimum axle width of $2000\ mm$ the load tracks must be place $50\ mm$ of centre as is shown in Figure 20.

3.4 Boundary conditions

Three sets of boundary conditions can be considered. These are the vertical boundary conditions at the bottom of the crossbeams (black), the horizontal boundary conditions at the transverse edges (red) and the horizontal boundary conditions at the longitudinal edges (orange) as shown in Figure 31. Each of these boundary conditions will be explained below.

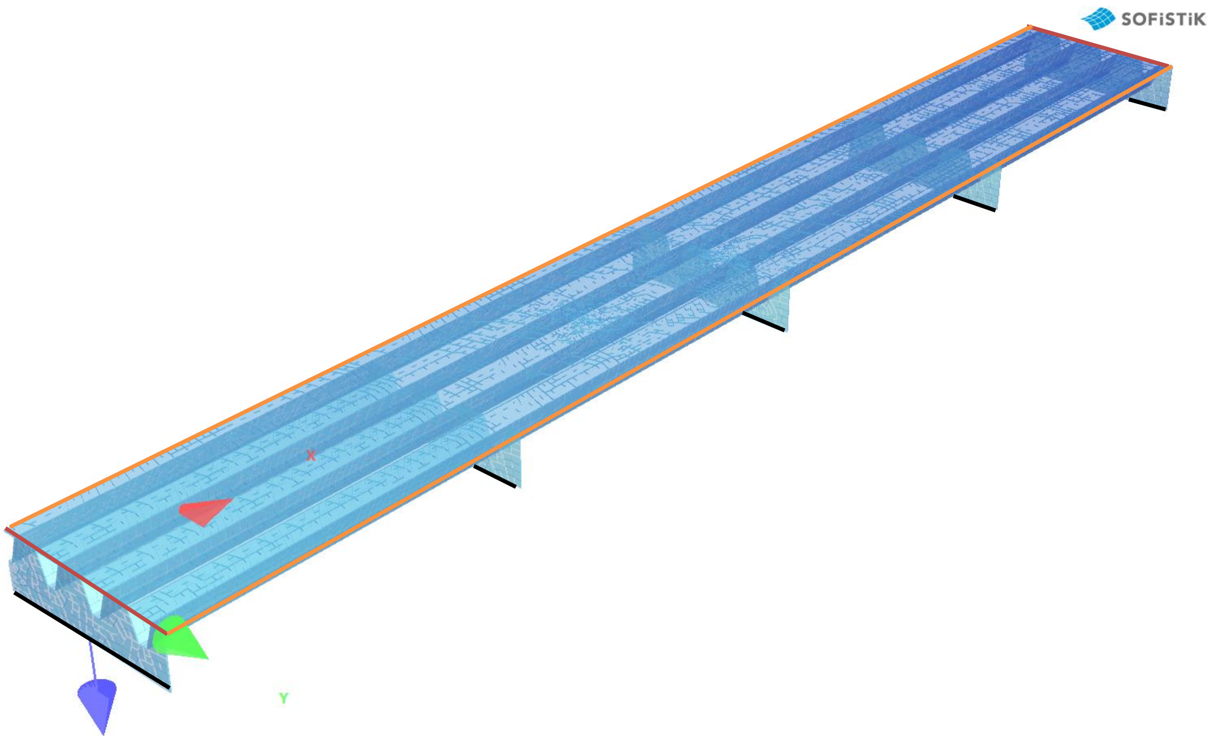


Figure 31 Boundary edges

3.4.1 Transverse edges

The transverse edge boundaries, shown by the red lines in Figure 31, haven't been fixed for rotation or translation at all. In the first model as shown in chapter 3.3.2 these edges had mirror symmetry boundary conditions. Since the model has been changed to include 4 complete spans this mirror symmetry is no longer needed. This free edge represents an edge of the bridge, using mirror symmetry would represent a longer bridge. Even though this might be the case for certain bridges it will not change the damage outcome since the stresses occurring due to loads at this edge are to low to result in damage. Therefore the simpler solution to keep the edges free has been chosen.

3.4.2 Longitudinal edges

The longitudinal edge boundaries, shown by the orange lines in Figure 31, have been fixed in rotation around the X-axis on either side and its translation in Y-direction has been fixed on one side. Some example calculations have been conducted to investigate the effects of the boundary conditions. The starting point was to model symmetry conditions (fixed translation in Y-direction and rotation around X-axis) on both sides. Damage calculations have been done to see whether releasing either of these fixations would lead to a higher or lower damage per result. The results of these calculations have been gathered in Table 2.

Table 2 Longitduinal boundary condition results

	fixed translation		free translation		Translation fixed	Translation free	fixed rotation	free rotation
	fixed rotation	free rotation	fixed rotation	free rotation	Difference fixed-free rotation	Difference fixed-free rotation	Difference fixed-free translation	Difference fixed-free translation
Detail 1	24,47	24,78	25,31	25,24	-1,29%	0,30%	3,34%	1,80%
Detail 2	21,10	21,22	27,51	30,82	-0,59%	-12,02%	23,32%	31,15%
Detail 3	51,87	51,70	59,32	45,79	0,33%	22,81%	12,55%	-12,92%
Detail 4	194,61	194,73	186,88	212,54	-0,06%	-13,73%	-4,14%	8,38%
Detail 5	30,34	30,43	30,97	30,19	-0,31%	2,52%	2,04%	-0,81%
Detail 6	2,84	2,85	2,92	2,93	-0,34%	-0,52%	2,71%	2,89%
Detail 7	2,74	2,75	2,85	2,86	-0,28%	-0,29%	3,71%	3,72%
Detail 8	393,92	394,53	395,30	395,72	-0,15%	-0,11%	0,35%	0,30%
Detail 9	0,00	0,00	0,00	0,00	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!
Detail 10	4,82	4,91	7,93	5,56	-1,72%	29,83%	39,15%	11,79%
Detail 11	10,58	10,55	11,22	10,46	0,32%	6,75%	5,66%	-0,86%

In this table the coloured cells represent the actual damage in the material. The first conclusion that can be drawn from these results is that freeing the translation results in higher damages for all details. This can be explained due to the fact that transverse contraction is prohibited by the fixation causing part of the load being carried by these supports.

Deciding on whether to fix the rotation or not is a bit more difficult since it depends on the detail if the damage is higher for fixed or free rotation. However, freeing the support would create a model without symmetry conditions and would therefore be unrealistic since only half of the car would be considered. Therefore the choice has been made to keep rotation fixed on either side.

To make sure the model isn't unstable and free to move in Y-direction one of the sides has a fixed translation in this direction. This fixation in Y-direction has been set on the negative Y side which ensures mirror conditions on this side. The reason for this choice is that at this location only half of a stiffener has been modelled. Creating full mirror conditions on this side ensures that the extra load carrying capacity of the stiffeners is considered.

3.4.3 Crossbeams

Different choices for the supports at the bottom of the crossbeams can be considered as well. The translation in Z-direction must be fixed in any case as it is assumed that the crossbeams deflect little to not at all. The other considerations are whether to fix the rotation around the Y-axis and fixing the translation in X-direction.

Figure 32 shows the results from the different situations.

	Configuration 1							
	Clamped Rotation	Free Rotation	Clamped Rotation					
	Fixed Translation	Free Translation	Free Translation				Max conf 1	Max conf 2
detail 1	25,31360824	25,40009833	25,40009833	0,34%		detail 1	25,40009833	30,77436427
detail 2	27,51352039	27,81673524	27,810387	1,09%		detail 2	27,81673524	30,7416325
detail 3	59,31575461	59,41021303	59,41021303	0,16%		detail 3	59,41021303	97,15809337
detail 4	186,8787233	189,2748353	189,1762109	1,27%		detail 4	189,2748353	310,2536459
detail 5	30,96974376	32,19886723	32,12148203	3,82%		detail 5	32,19886723	28,12854721
detail 6	2,919342926	1,784235265	1,67240118	-63,62%		detail 6	2,919342926	2,728957835
detail 7	2,850001784	0	0,00000			detail 7	2,850001784	2,806723653
detail 8	395,2962047	404,04171	403,3865909	2,16%		detail 8	404,04171	401,2804159
detail 9	0	0	0			detail 9	0	0
detail 10	7,926827587	8,081894382	8,065814521	1,92%		detail 10	8,081894382	32,5356147
detail 11	11,21694794	11,42265596	11,30780225	1,80%		detail 11	11,42265596	0
	Configuration 2							
	Clamped Rotation	Free Rotation	Clamped Rotation					
	Fixed Translation	Free Translation	Free Translation					
detail 1	30,63780097	30,77436427	30,77436427	0,44%				
detail 2	30,568264	30,7416325	30,74129931	0,56%				
detail 3	97,08582464	97,15809337	97,14578051	0,07%				
detail 4	306,7586232	310,2536459	309,9905347	1,13%				
detail 5	26,95975651	28,12854721	28,0510322	4,16%				
detail 6	2,728957835	1,520388616	1,407356884	-79,49%				
detail 7	2,806723653	0	0,00000	-100,00%				
detail 8	391,5505878	401,2804159	400,7003663	2,42%				
detail 9	0	0	0					
detail 10	31,99406169	32,5356147	32,49430275	1,66%				
detail 11	0	0	0					

Figure 32 Crossbeam supports

The biggest damage values of all the combinations have been circled by a black boxes. Most of them can be found for boundary conditions with free rotation and free translation, even though they do not differ much from the damage values with other boundary conditions. The exceptions are detail 6 and 7 which occur in the stiffener to crossbeam connection. If the translation and rotation is set free the produce significantly lower damage values compared to fixed rotational and translational boundary conditions.

Since these details produces significantly lower damage values for free boundary conditions the choice has been made to fix both the translation and rotation of the crossbeams

3.5 Mesh optimization

By making use of symmetry conditions the size of a model becomes smaller and therefore less elements need to be evaluated. Another way to minimize the number of elements and therefore the calculation time is to use mesh refinements around the locations of the details. This way the rest of the model can have bigger elements without drastically decreasing the accuracy of results. Another reason for mesh refinement is to be sure that applied loads are distributed correctly. Especially close to the details load distribution and transfer is important.

These refinements can be implemented in multiple ways. In this case a standard mesh size is defined for the whole model. After that, mesh refinements are defined along structural lines including a spread zone from the refined mesh to the standard mesh. These refinements can be seen in Figure 33 and Figure 34. The dark blue lines represent the cross beams which can't be seen in a top view.

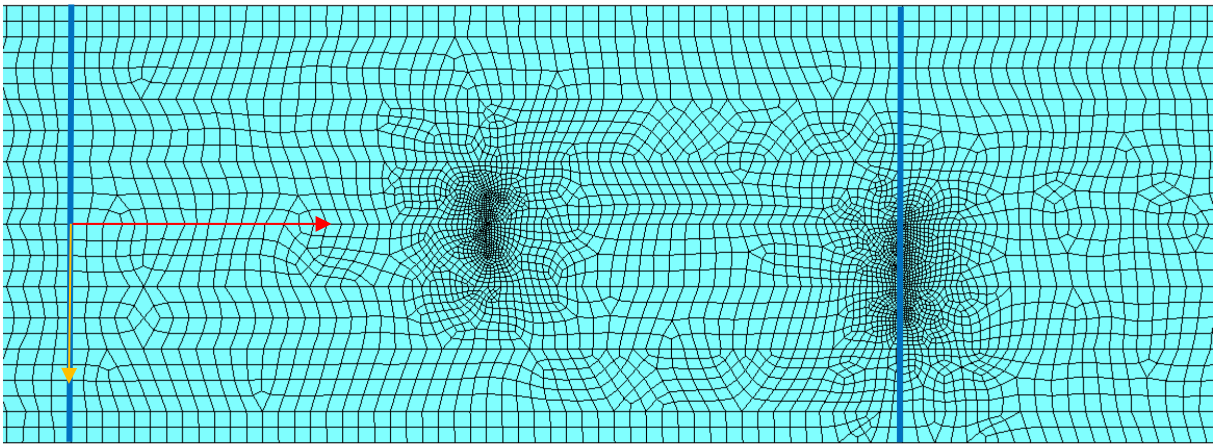


Figure 33 Mesh refinement, top view

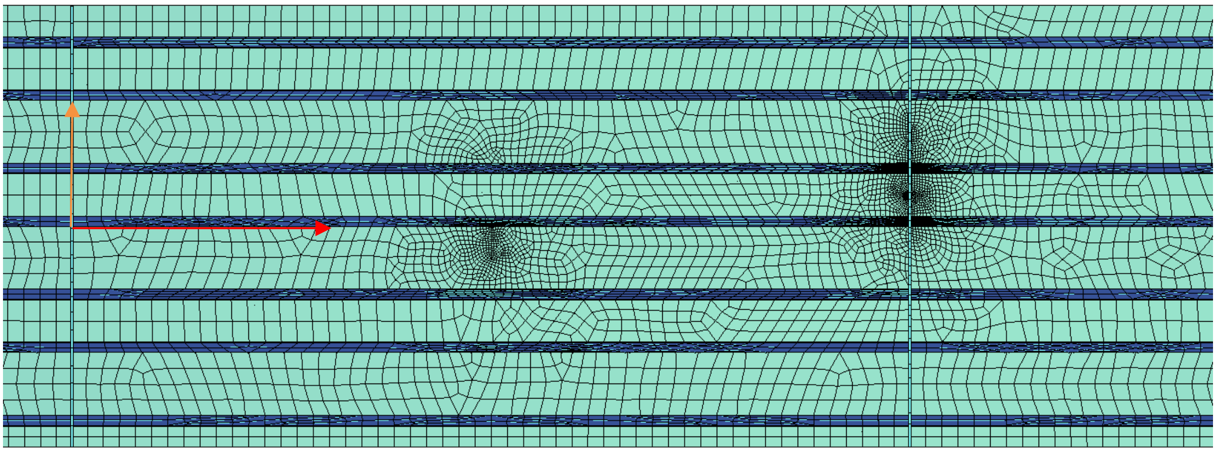


Figure 34 Mesh refinement, bottom view

3.6 Load step optimization

Influence lines are necessary to obtain stress ranges which are needed to calculate the fatigue damage to the structure. An influence line gives a specific result at a point in a structure caused by a load placed at any point along the structure. Placing a point load at any point along the structure would result in many load increments causing a long calculation time. The larger a load increment the quicker the calculation, however accuracy will decrease. Making some smart choices in the load increments however can result in a quick calculation without losing accuracy.

The first load increment on the edge of the model will cause no significant stresses in any of the details sine the load is directly transferred to the cross-beam. The first increment can therefore be skipped and just set on 0. Another simplification lies in the fact that damage is calculated due to stress range which is depending on stress peaks. The first stress peak for the details will lie in the first span between cross-beams but always after the first half span. This means that the increments in the first half span aren't decisive for the stress range and therefore increments can be rather big. Further increase in load steps is not possible because the influence lines for the details differ to much from each other.

To obtain an accurate influence line the step size refinement must be between 0.5 and 0.1 meters. Setting it even lower would result in an enormous calculation time. To make sure the bigger step size is chosen in such a way it fits every possible model it can be made dependent of the span from cross-beam to cross beam, for instance $\frac{cross-beam\ span}{4}$ which is taken for $ld_{inc_{std}}$. For a cross beam span of 4 meters this results in 123 load steps per track per wheel geometry as shown in equation (3.1).

$$\begin{aligned} \boldsymbol{ld_{inc_t}} &= \frac{2 * \frac{\boldsymbol{br_{hoh_{cb}}}}{2}}{\boldsymbol{ld_{inc_{std}}}} + \frac{\boldsymbol{br_L} - \boldsymbol{br_{hoh_{cb}}}}{\boldsymbol{ld_{inc_{ref}}}} - \mathbf{1} \\ &= \frac{\left(2 * \frac{4}{2}\right)}{1} + \frac{(4 * 4) - 4}{0.1} - 1 = 4 + 120 - 1 = 123 \end{aligned} \tag{3.1}$$

A "−1" is added in this equation to overcome double counting of steps. The 123 steps calculated here will be used for all examples in this report.

With 8 tracks and 3 wheel geometries this gives 2952 different load cases to be analysed. A total of 3864 load cases would have been analysed without the above-mentioned refinements as shown in equation (3.2).

$$\begin{gathered} ld_{inc_t} = \frac{br_L}{0.1} + 1 = \frac{16}{0.1} + 1 = 161 \\ 161 * 8 * 3 = 3864 \end{gathered} \tag{3.2}$$

This means that the refinements result in a decrease in calculation time of approximately 25%.

3.7 Export results

As stated before, the fatigue calculation will be done in an already existing tool. This tool requires an input of the influence lines of the stresses due to the wheel loads in a unit of MPa. Due to the demands given by the Eurocode not all stresses can be obtained directly from the model. The nominal stresses are exported from the modelling software directly. For the hot spots a post calculation is needed as explained below.

3.7.1 Hot spot stress (detail 3 & 6)

A special approach is needed for the hot spot stress. When retrieving the stresses at the exact location of the details a singularity will be found. Therefore, an extrapolation will be done from two points near the desired location as is shown in Figure 35. The location of these two points have a set value of half plate thickness and one and a half plate thickness. Structural points will be defined to read out stresses at these locations. The extrapolation will be done in the post calculation in excel. This determination method for hot spot stresses comes from the Norwegian verification documents (DNV) and is used as a national annex in Norway. The extrapolation for this case is shown in equation (3.3).

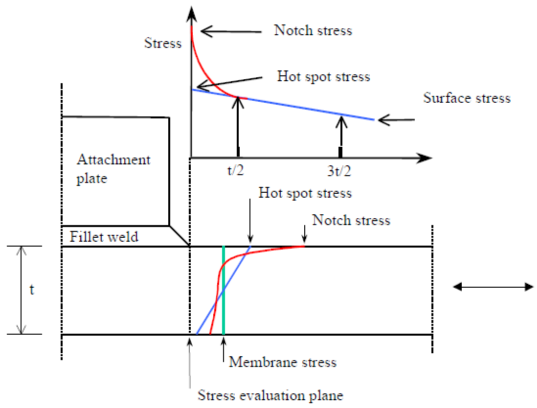


Figure 35 Schematic stress distribution (DNVGL-RP-0005 figure 4-2)

$$f(x) = y_1 + \left(\frac{x - x_1}{x_2 - x_1}\right) * (y_2 - y_1) \tag{3.3}$$

$$point.1 + \left(\frac{0 - 0.5}{1.5 - 0.5}\right)(point.2 - point.1) = point.1 - \frac{1}{2}(point.2 - point.1)$$

4 The model

With the structural requirements and model definitions determined the model can be made. The model will be made using the text editor "teddy" of FEM-software SOFiSTikFirst. The format for the variables will be explained first. After that each module used in teddy will be explained shortly. The following modules have been used: Template, Aqua, Sofimshc, Sofiload, Ase and Results. An explanation of the program commands can be found in Appendix B: SOFiSTik.

4.1 Parameter format

Within each company there are many people who use programs and each person has its own preferences on how to call certain parameters. This preference might not be clear to other people than the maker who try to read the code. A fixed format for parameters is therefore required to be used by everyone. Below is the parameter format that Royal HaskoningDHV wants to implement.

Each parameter consists of at least 2 arguments. The first argument will define the global part it is assigned to (bridge, load, model, material). The second argument assigns the dimension (width, thickness, number, location) and the third argument assigns the parts (stiffener, deck, wheel track, mesh). Table 3 gives an overview of the used parameters in the model.

Table 3 parameter overview

First argument		second argument		third argument		Extra
Bridge	br	width	w	stiffener	.stif	**t**op / **b**ottom
		height	h	deck	deck	
		thickness	t	cross-beam	cb	
		centre to centre	ctc	girder	gir	
		number	n	asphalt	asph	
		elongation weld	elong			
		location	loc			
Loads	ld	selfweight	g	wheel load	wheel	
		pointload	F	wheel track	.track	**x** / **y**
		lineload	p	increments	.inc	**t**otal / **s**td / **r**ef
		area load	q	loadstp	.stp	**x** / **y**
		moment	M			
		offset	offset			
		name	name			
		traffic type	tt			
		number	n			
		standard	std			
		refinement	ref			
		location	loc			
Materials	mt	yieldstrength	Fy	steel	ste	
Model	md	standard	std	mesh	msh	
		refinement	ref	direction	.dir	**x** / **y**
		number	n			
		start refinement	refstart			
		end refinement	refend			

4.2 Template

Program template is the first module and has only one purpose, assigning values to variables. These are the parameters that are used in the rest of the program and can be altered by the user to produce models with different dimensions. The input parameters are shown in Figure 36.

```
+PROG TEMPLATE urs:1
HEAD Input parameters
STO#mt_Fy_ste 355 $Steel trength
STO#br_t_deck 0.018 $Deck thickness
STO#br_t_asph 0.06 $Asphalt thickness
STO#br_w_tstif 0.3 $Stiffener top width
STO#br_w_bstif 0.1 $Stiffener bottom width
STO#br_h_stif 0.3 $Stiffener height
STO#br_t_stif 0.008 $Stiffener thickness
STO#br_ctc_stif 0.6 $Stiffener center-to-center distance
STO#br_h_cb 0.6 $Cross-beam height
STO#br_t_cb 0.016 $Cross-beam thickness
STO#br_ctc_cb 4 $Cross-beam center-to-center distance
STO#br_elong_stif 7.2 $Distance to stiffener butt-weld from crossbeam
STO#br_elong_deck 6.8 $Distance to deck butt-weld from crossbeam
STO#md_std_msh 0.075 $standard mesh size
STO#md_ref_msh 0.01 $mesh refinement at details
STO#ld_std_xstep 1 $Loading longitudinal step size
STO#ld_ref_xstep 0.1 $Loading longitudinal step size refinement
STO#ld_n_ystep 8 $Loading number of transverse steps
STO#ld_std_ystep 0.1 $Loading transverse step size
STO#br_cutout_cb 1 $Model with crossbeam cut-out
STO#ld_offset_y 0 $Offset of load tracks in transverse direction
END
```

Figure 36 SoFiSTik template module

4.3 Aqua

In this module, the material and design code are defined, furthermore the units in which the input needs to be given is chosen. Extra variables that depend on the input parameters or fixed variables are stored here as well. These include for instance the number and location of stiffeners, number of cross beam spans and the load steps. The variables could be defined in any module but the choice has been made to define most of them in the module aqua to create some order in the document. Figure 37 shows the aqua module.

```
+PROG AQUA urs:14.14
HEAD Material definition - parameters
CTRL REST 0
NORM DIN EN1992-2004            $defines default design code
STEE NO 1 TYPE S #mt_Fy_ste      $defines material properties, material 1 = S355
UNIT TYPE 0                          $defines unit, in this case Meters, kN, sec

!*! Values depending on width top stiffener
STO#br_n_stif 3
STO#br_loc_stif -1*#br_ctc_stif,0,1*#br_ctc_stif

!*! further parameters
STO#br_n_cb 5                          $number of crossbeam spans
STO#br_n_span #br_n_cb-1               $number of spans (crossbeam to crossbeam)
STO#ld_name_wheel 'A','B','C'          $geometries used in results
STO#br_L #br_n_span*#br_ctc_cb         $stores length of model
STO#br_w 3.5*#br_ctc_stif      $stores width of model
STO#br_loc_cb 0,#br_ctc_cb,2*#br_ctc_cb,3*#br_ctc_cb,4*#br_ctc_cb  $determines cross beam location
STO#ld_refstart_xdir #ld_std_xstep*2
STO#ld_refend_xdir #br_L-#ld_refstart_xdir
STO#off #ld_offset_y

!*! track steps in Y direction
STO#ld_stp  -3.5*#ld_std_ystep+#off, -2.5*#ld_std_ystep+#off, -1.5*#ld_std_ystep+#off, -0.5*#ld_std_ystep+#off,
STO#ld_name_track '-4','-3','-2','-1','1','2','3','4'

!*! load increments
STO#ld_Fz 50
LET#ld_n_rinc (#ld_refend_xdir-#ld_refstart_xdir)/#ld_ref_xstep
LET#ld_n_sinc (#br_L-#ld_refend_xdir+#ld_refstart_xdir)/#ld_std_xstep
STO#ld_n_tinc #ld_n_rinc+#ld_n_sinc-1

LOOP#i #ld_n_tinc
  IF (#i<#ld_n_sinc/2)
    STO#ld_xstp(#i) #ld_std_xstep*#i+#ld_std_xstep
  ELSEIF (#i<#ld_n_rinc+#ld_n_sinc/2)
    STO#ld_xstp(#i) #ld_refstart_xdir+#ld_ref_xstep*(#i-#ld_n_sinc/2+1)
  ELSE
    STO#ld_xstp(#i) #ld_refend_xdir+#ld_std_xstep*(#i-#ld_n_rinc-#ld_n_sinc/2+1)
  ENDIF
ENDLOOP
END
```

Figure 37 SoFiSTik aqua module

In this first part the norm, steel and units are defined first. After that the values which depend on the top width of the stiffener and the stiffener location are defined. Next are some fixed parameters that are stored and the location of the load tracks are given.

The number of load increments in longitudinal direction is defined in the last part and follows the explanation of the load step optimization of chapter 3.3.6. Depending on the size of the model the number of load steps can differ. With the help of multiple if-statements a string (#ld_xstp) is made which consists of the x-coordinates of the centre of each load area. This string will later be used in the sofiload module.

4.4 Sofimshc

The next module is the Sofimshc module and creates the model with the help of the previously defined parameters. This module is split up in 3 sections in which structural points are defined, then with these points structural lines are made and last structural areas are made from the structural lines. Before this is done the type of mesh (beam & plate elements) and the standard mesh size is defined.

To keep track of which point, line or area is where exactly an overview map is needed. Figure 38 shows an overview of the model with a side-view and the first cross-beam plane. Below is the list of numbers and what they are used for.

- 0-99 for result details
- 100-999 for parts in the cross-beam plane
- 1000-1999 for parts in between the cross-beam planes (stiffeners)
- 2000-2999 for parts used for the overall dimensions (deck)
- 3000-… for the mesh refinements

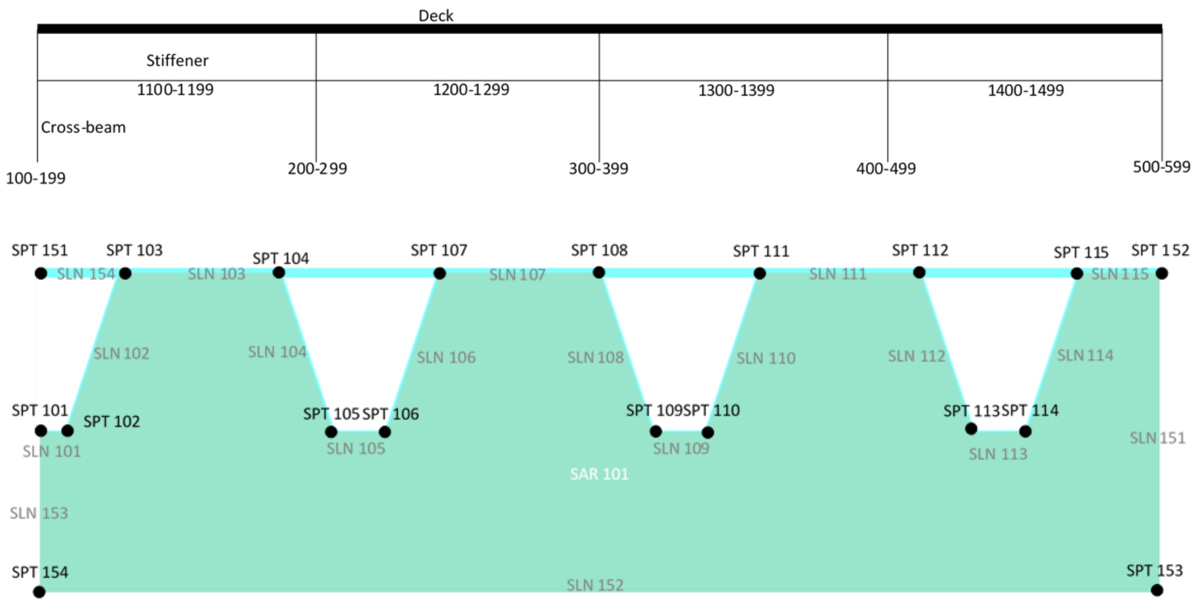


Figure 38 SoFiSTik sofimshc node overview

4.4.1 Section 1

In the first part, shown in Figure 39, mesh parameters are defined together with the structural points to extract data at the detail locations later on. A structural point is defined by using coordinates X, Y and Z. These are made dependent of the previously defined variables to make sure they change when the model dimensions are changed.

```
+PROG SOFIMSHC urs:14.15
HEAD Meshing - element definition     $sorted by parts
SYST 3D GDIV 10000000 GDIR POSZ        $defines the global system used (3D, 1 group=100000 element
CTRL OPT MESH 66;              CTRL OPT hmin #md_std_msh; $makes the mesh, mesh65=meshing of beamquadr
CTRL OPT TOLG #md_ref_msh/10
CTRL OPT PROG 1.2
ECHO VAL NO
CTRL DELN      0
CTRL WARN 10750


!*! Structural points

$results
SPT NO 1  X 3/2*#br_ctc_cb          Y 0+#md_ref_msh/2       Z 0                         $Detail 1
SPT NO 2  X 3/2*#br_ctc_cb          Y 0-#md_ref_msh/2       Z 0                         $Detail 2
SPT NO 3  X 2*#br_ctc_cb            Y +#br_t_deck/2         Z 0                         $Detail 3.1
SPT NO 4  X 2*#br_ctc_cb            Y 0+3/2*#br_t_deck      Z 0                         $Detail 3.2
SPT NO 5  X 3/2*#br_ctc_cb          y 0                     z 0                         $Detail 4 &
SPT NO 6  X #br_elong_stif  Y #br_w_tstif/2-#br_w_bstif/2   Z #br_h_stif                $Detail 5 (s
SPT NO 7  X 2*#br_ctc_cb-#br_t_stif/2      Y #br_w_tstif/2  Z #br_h_stif                $Detail 6.1
SPT NO 8  X 2*#br_ctc_cb-3/2*#br_t_stif    Y #br_w_tstif/2  Z #br_h_stif                $Detail 6.2
SPT NO 9  X 2*#br_ctc_cb            Y #br_w_tstif/2         Z #br_h_stif+#md_ref_msh    $Detail 7 (c
SPT NO 10 X 3/2*#br_ctc_cb          Y #br_w_tstif/2         Z #br_h_stif                $Detail 8 (s
SPT NO 11 X #br_elong_deck          Y #br_w_tstif/2         Z 0                         $detail 9
SPT NO 12 X 2*#br_ctc_cb            Y -#br_w_tstif/2        Z 0                         $detail 11
SLN NO 3005 NPA 7 NPE 8 SDIV #br_t_stif/2                                               $a help line
```

Figure 39 SoFiSTik sofimshc module section 1 part 1

Figure 40 shows the second part of the first section. Here the structural points needed for the stiffeners, cross-beams, deck and the mesh refinement are defined.

For the cross-beams the points defined for the stiffeners will be re-used. Only 4 extra points are needed to be able to define the cross-beams. The points from the cross-beams can be re-used to model the deck. Three sets of mesh refinement are needed which refines a mesh along a certain line, hence the 6 structural points.

```
$stiffener
LOOP#j #br_n_cb
 SPT NO 101+#j*100 X #br_loc_cb(#j) Y -0.5*#br_w                        Z #br_h_stif
 SPT NO 102+#j*100 X #br_loc_cb(#j) Y -0.5*#br_w+#br_w_bstif/2          Z #br_h_stif
 SPT NO 103+#j*100 X #br_loc_cb(#j) Y -0.5*#br_w+#br_w_tstif/2          Z 0
 SPT NO 116+#j*100 X #br_loc_cb(#j) Y +0.5*#br_w                        Z 0
ENDLOOP
LOOP#i #br_n_stif
 LOOP#j #br_n_cb
  SPT NO 104+#i*4+#j*100 X #br_loc_cb(#j) Y #br_loc_stif(#i)                                    Z 0
  SPT NO 105+#i*4+#j*100 X #br_loc_cb(#j) Y +0.5*#br_w_tstif-0.5*#br_w_bstif+#br_loc_stif(#i)   Z #br_h_stif
  SPT NO 106+#i*4+#j*100 X #br_loc_cb(#j) Y +0.5*#br_w_tstif+0.5*#br_w_bstif+#br_loc_stif(#i)   Z #br_h_stif
  SPT NO 107+#i*4+#j*100 X #br_loc_cb(#j) Y #br_w_tstif+#br_loc_stif(#i)                        Z 0
 ENDLOOP
ENDLOOP

$cross beams
LOOP#j #br_n_cb
  SPT NO 151+#j*100 X #br_loc_cb(#j) Y -#br_w/2        Z 0
  SPT NO 152+#j*100 X #br_loc_cb(#j) Y #br_w/2         Z 0
  SPT NO 153+#j*100 X #br_loc_cb(#j) Y #br_w/2         Z #br_h_cb
  SPT NO 154+#j*100 X #br_loc_cb(#j) Y -#br_w/2        Z #br_h_cb
ENDLOOP

$mesh refinement
SPT NO 3001 X 2*#br_ctc_cb        Y -#br_w_tstif/2       Z 0
SPT NO 3002 X 2*#br_ctc_cb        Y 1.5*#br_w_tstif      Z 0
SPT NO 3003 X 2*#br_ctc_cb        Y 0                    Z #br_h_stif
SPT NO 3004 X 2*#br_ctc_cb        Y #br_w_tstif          Z #br_h_stif
SPT NO 3005 X 1.5*#br_ctc_cb      Y -#br_w_tstif/2       z 0
SPT NO 3006 X 1.5*#br_ctc_cb      Y #br_w_tstif/2        z 0
```

Figure 40 SoFiSTik sofimshc module section 1 part 2

4.4.2 Section 2

The second section of the sofimshc module is the defining of structural lines as is shown in Figure 41. First of in Figure 41 is the defining of structural lines for the stiffeners. Be aware that the loops depend on the number of stiffeners in the model and the number of spans. This part will make all the structural lines that run between the cross-beams hence the numbers 1101-1105.

The second part defines the structural lines in the cross beam plane. This also includes the lines needed for the stiffeners (101-104) and the lines only needed for the cross-beams (151-154). Here the loops run as many times as there are cross-beams in the model (#br_n_cb).

For the deck the structural points at the top corners of the first and last cross-beams are used. Only two lines (in X-direction) need to be defined since the structural line 154 of the crossbeams can be re-used.

The mesh refinement uses the predefined points from Figure 40. Refinement occurs due to the “SDIV #md_ref_msh” at the and of the structural line definition which stands for the mesh density used for a subdivision.

```
!*! Structural lines
$stiffener

LOOP#j #br_n_span
 SLN NO 1101+#j*100 NPA 101+#j*100 NPE 201+#j*100 FIX PYMX
 SLN NO 1102+#j*100 NPA 102+#j*100 NPE 202+#j*100
 SLN NO 1103+#j*100 NPA 103+#j*100 NPE 203+#j*100
ENDLOOP
LOOP#i #br_n_stif
 LOOP#j #br_n_span
  SLN NO 1104+#i*4+#j*100 NPA 104+#i*4+#j*100 NPE 204+#i*4+#j*100
  SLN NO 1105+#i*4+#j*100 NPA 105+#i*4+#j*100 NPE 205+#i*4+#j*100
  SLN NO 1106+#i*4+#j*100 NPA 106+#i*4+#j*100 NPE 206+#i*4+#j*100
  SLN NO 1107+#i*4+#j*100 NPA 107+#i*4+#j*100 NPE 207+#i*4+#j*100
 ENDLOOP
ENDLOOP

$cross beams

LOOP#j #br_n_cb
 SLN NO 151+#j*100 NPA 152+#j*100 NPE 153+#j*100
 SLN NO 152+#j*100 NPA 153+#j*100 NPE 154+#j*100 FIX PZMY
 SLN NO 153+#j*100 NPA 154+#j*100 NPE 151+#j*100 FIX PY
 SLN NO 154+#j*100 NPA 151+#j*100 NPE 152+#j*100
ENDLOOP
LOOP#j #br_n_cb
 SLN NO 113+#j*100 NPA 113+#j*100 NPE 114+#j*100
 SLN NO 114+#j*100 NPA 114+#j*100 NPE 115+#j*100
 SLN NO 115+#j*100 NPA 115+#j*100 NPE 152+#j*100
 LOOP#i #br_n_stif
  SLN NO 101+#i*4+#j*100 NPA 101+#i*4+#j*100 NPE 102+#i*4+#j*100
  SLN NO 102+#i*4+#j*100 NPA 102+#i*4+#j*100 NPE 103+#i*4+#j*100
  SLN NO 103+#i*4+#j*100 NPA 103+#i*4+#j*100 NPE 104+#i*4+#j*100
  SLN NO 104+#i*4+#j*100 NPA 104+#i*4+#j*100 NPE 105+#i*4+#j*100
 ENDLOOP
ENDLOOP

$deck
LOOP#j #br_n_span
 SLN NO 1151+#j*100 NPA 151+#j*100 NPE 251+#j*100 FIX PYMX
 SLN NO 1152+#j*100 NPA 152+#j*100 NPE 252+#j*100 FIX MX
ENDLOOP

$mesh refinement
SLN NO 3001 NPA 3001 NPE 3002 SDIV #md_ref_msh
SLN NO 3002 NPA 3003 NPE 3004 SDIV #md_ref_msh
SLN NO 3003 NPA 3005 NPE 3006 SDIV #md_ref_msh
```

Figure 41 SoFiSTik sofimshc section 2 part 1

4.4.3 Section 3

The structural areas or plates (SAR) are defined in the third section by combining structural area boundaries (SARB) which refer to the structural lines and by assigning a material and thickness as is shown in figure Figure 42. It can be seen that the structural areas of the stiffeners are made from cross-beam to cross-beam (it loops #br_n_span times) making it very easy to add an extra cross-beam if required for an analysis.

For the second structural area in the loop (SAR NO 1103+..) an extra distinction for the group is made. This side plate of the stiffener has a different group for each span which is necessary to read out the results correctly.

```
!*! Structural areas
$stiffeners
LOOP#i #br_n_stif
 LOOP#j #br_n_span
  SAR NO 1101+#i*3+#j*100 GRP 1 MNO 1 T 1000*#br_t_stif
   SARB TYPE OUT 101+#i*4+#j*100
   SARB TYPE OUT 201+#i*4+#j*100
   SARB TYPE OUT 1101+#i*4+#j*100
   SARB TYPE OUT 1102+#i*4+#j*100

  SAR NO 1102+#i*3+#j*100 GRP 2 MNO 1 T 1000*#br_t_stif
   SARB TYPE OUT 102+#i*4+#j*100
   SARB TYPE OUT 202+#i*4+#j*100
   SARB TYPE OUT 1102+#i*4+#j*100
   SARB TYPE OUT 1103+#i*4+#j*100

  SAR NO 1103+#i*3+#j*100 GRP 3+#j*10 MNO 1 T 1000*#br_t_stif
   SARB TYPE OUT 104+#i*4+#j*100
   SARB TYPE OUT 204+#i*4+#j*100
   SARB TYPE OUT 1104+#i*4+#j*100
   SARB TYPE OUT 1105+#i*4+#j*100
 ENDLOOP
ENDLOOP
LOOP#i 2
 LOOP#j #br_n_span
  SAR NO 1110+#i+#j*100 GRP 3 MNO 1 T 1000*#br_t_stif
   SARB TYPE OUT 113+#i+#j*100
   SARB TYPE OUT 213+#i+#j*100
   SARB TYPE OUT 1113+#i+#j*100
   SARB TYPE OUT 1114+#i+#j*100
 ENDLOOP
ENDLOOP
```

Figure 42 SoFiSTik sofimshc section 3 part 1

Figure 43 shows the creation of cross-beams and the deck. The deck is also split up in different groups for the same reason as the stiffener plate. This is done specifically for detail 11.

```
$cross beams
LOOP#j #br_n_cb
 SAR NO 101+#j*100 GRP 4 MNO 1 T 1000*#br_t_cb
  LOOP#i 15
   SARB TYPE OUT 101+#j*100+#i
  ENDLOOP
  LOOP#i 3
   SARB TYPE OUT 151+#j*100+#i
  ENDLOOP
ENDLOOP

$deck
LOOP#j #br_n_span
SAR NO 2000+#j*100 GRP 5+#j MNO 1 T 1000*#br_t_deck MCTL REGM
 SARB TYPE OUT 154+#j*100
 SARB TYPE OUT 254+#j*100
 SARB TYPE OUT 1151+#j*100
 SARB TYPE OUT 1152+#j*100
ENDLOOP
END
```

Figure 43 SoFiSTik sofimshc section 3 part 2

4.5 Sofiload

The next step in SOFiSTik is to define the different loads. This will consist of 3 different wheels running on 8 tracks with 123 steps each as explained in chapter 3.3.6. The string #ld_xstep defined in the aqua module is used for all the steps. Figure 45 shows the code that defines the loads for wheel geometry A (until the dashed black line).

First of a loop is initiated to loop through the different wheel tracks defined earlier. This sets the load step in Y-direction. Next the size of the area is defined with #Xax and #Yax which takes into account the dispersion in the deck following equation (2.5). After that a loop is initiated to loop through the earlier defined load increments in the X-direction. With the load increments in both X and Y-direction the origin of each load step is known. From there the load area (LAR) is defined using the #Xax and #Yax as shown in Figure 44. In the last step of this loop the area is assigned to a load case and the load and load direction are assigned. In this case this is a load in global Z-direction (PZZ) and the load equals 50kN divided by the area.

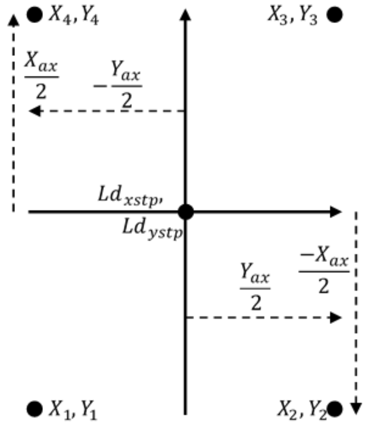


Figure 44 Load area explanation

The loads for geometry B and C are defined below the dashed black line. There are two different options for wheel geometry B, one where the two load areas of which geometry B is comprised do not overlap and one where they do overlap. The two load areas might overlap due to the dispersion in the deck, which happens if half the width of one wheel is bigger than 0.16 meters. When this happens only one load area will be modelled with the outer dimensions of the two wheels combined and applying the dispersion formula afterwards. The moment the dispersed wheel areas do not meet the load is split along two load areas as stated in the norm.

Load area for geometry C is comprised the same way as A, only the dimensions have been altered.

```
+PROG SOFILOAD urs:14.16
HEAD Load defenition fatigue
ECHO VAL NO
CTRL WARN 79

LOOP#D #ld_n_ystep
  $geometry A
  LET#Xax 0.32+2*#br_t_asph+#br_t_deck; LET#Yax 0.22+2*#br_t_asph+#br_t_deck
  LOOP#i #ld_n_tinc
      LAR NO 10000+#D*200+#i+1  X1 -#Xax/2+#ld_xstp(#i) Y1 -#Yax/2+#ld_ystp(#D)
                                X2 -#Xax/2+#ld_xstp(#i) Y2 #Yax/2+#ld_ystp(#D)
                                X3 #Xax/2+#ld_xstp(#i) Y3 #Yax/2+#ld_ystp(#D)
                                X4 #Xax/2+#ld_xstp(#i) Y4 -#Yax/2+#ld_ystp(#D)
      LC NO 10000+#D*200+#i+1 ; AREA REF LAR NO 10000+#D*200+#i+1 TYPE PZZ P1 #ld_Fz/(#Xax*#Yax)
  ENDLOOP
  $Geometry B
  LET#Xax 0.32+2*#br_t_asph+#br_t_deck; LET#Yax 0.22+2*#br_t_asph+#br_t_deck
  IF (#Yax/2<0.16)
    LOOP#i #ld_n_tinc
        LAR NO 11000+#D*200+#i+1  X1 -#Xax/2+#ld_xstp(#i) Y1 -0.16-#Yax/2+#ld_ystp(#D)   X2 -#Xax/2+#ld_xstp(#i) Y2 -0.16+#Ya
        LAR NO 12000+#D*200+#i+1  X1 -#Xax/2+#ld_xstp(#i) Y1 +0.16-#Yax/2+#ld_ystp(#D)   X2 -#Xax/2+#ld_xstp(#i) Y2 +0.16+#Ya
        LC NO 11000+#D*200+#i+1 ; AREA REF LAR NO 11000+#D*200+#i+1 TYPE PZZ P1 #ld_Fz/(2*#Xax*#Yax); AREA REF LAR NO 12000+#
    ENDLOOP
  ELSE
    LET#Xax 0.32+2*#br_t_asph+#br_t_deck; LET#Yax 0.54+2*#br_t_asph+#br_t_deck
    LOOP#i #ld_n_tinc
        LAR NO 11000+#D*200+#i+1  X1 -#Xax/2+#ld_xstp(#i) Y1 -#Yax/2+#ld_ystp(#D)   X2 -#Xax/2+#ld_xstp(#i) Y2 #Yax/2+#ld_yst
        LC NO 11000+#D*200+#i+1 ; AREA REF LAR NO 11000+#D*200+#i+1 TYPE PZZ P1 #ld_Fz/(#Xax*#Yax)
    ENDLOOP
  ENDIF
  $Geometry C
  LET#Xax=0.32+2*#br_t_asph+#br_t_deck; LET#Yax=0.27+2*#br_t_asph+#br_t_deck
  LOOP#i #ld_n_tinc
      LAR NO 13000+#D*200+#i+1  X1 -#Xax/2+#ld_xstp(#i) Y1 -#Yax/2+#ld_ystp(#D)   X2 -#Xax/2+#ld_xstp(#i) Y2 #Yax/2+#ld_ystp(
      LC NO 12000+#D*200+#i+1 ; AREA REF LAR NO 13000+#D*200+#i+1 TYPE PZZ P1 #ld_Fz/(#Xax*#Yax)
  ENDLOOP
ENDLOOP
```

Figure 45 SoFiSTik sofiload part 1

4.6 Ase

Ase is the analysis module of SOFiSTik. The load cases that should be analysed must be given here. In this case this is fairly simple since all load cases need to be examined as is, however when load combinations are needed this part gets tricky. As could have been noticed in the other modules there is a line of code that states ECHO VAL NO. This tells the program that no extra unnecessary output is created and stored. Doing this decreases the calculation time significantly.

```
+PROG ASE urs:14.17
HEAD Analisys fatigue
LC All
END
```

Figure 46 SoFiSTik ase

4.7 Results

The results module produces the excel workbooks needed for the model analysis. This collects data produced by the analysis and exports it into an excel workbook. The way to produce results for detail 3 is shown in Figure 47 as an example.

Once more loops have been used, in this case to loop through the different wheel tracks and after that the number of wheel geometries. The next argument creates a new "picture" or in this a figure. After that a filter is needed to tell the program for which point data should be retrieved. In this case two filters have been used. The first filter requests the results of structural point 5. Since this structural point is both used for the stiffener and the deck another filter is needed to tell the program that it needs the results of group one which corresponds with the stiffeners. The next line defines the different loads which should be exported. To make sure that the data is exported to an .xlsx file the next line is used. This also states the name of the worksheet (WS) which uses the loops to create different names for each sheet. To finish the output the program needs to know which results should be exported, in this case the stresses in local Y-direction (SYL) have been requested.

```
+PROG RESULTS urs:14.22
HEAD Results detail 3

!#!detail 3 cross section A location
LOOP#T #ld_n_ystep      $number of tracks (#ld_name_track defined in sofiloads)
  LOOP#G 3              $number of wheel geometries
    PICT SC DEFA W DEFA H DEFA TITL "Geo #ld_name_wheel(#G) track #ld_name_track(#T)"  SPLT NO
    FILT NAME "quad_nfo.nr" TYPE EQ OPTI YES VAL1 5
    FILT NAME "quad_nfo.ng" TYPE EQ OPTI YES VAL1 1
    LC   NO (10001+1000*#G+200*#T  10000+1000*#G+200*#T+#ld_n_tinc  1)
    XLSX NAME "Export detail 3 CS_A (S).xlsx" WS "Geo #ld_name_wheel(#G) track #ld_name_track(#T)" ROW 1 COL 1 CLNM  YES TIME
    QUAD TYPE SYL STYP NODE REPR DLST
  ENDLOOP
ENDLOOP

END
```

Figure 47 SoFiSTik results

To make it easier to switch certain results on or of each detail has its own results module. By changing the "+" in the first line into a "-" the module is not considered.

5 Model analysis

With the results from the SOFiSTik model the fatigue damage can be determined. The process to do so will be explained step-by-step in the next few paragraphs. Figure 48 shows a calculation flow chart of the necessary steps.

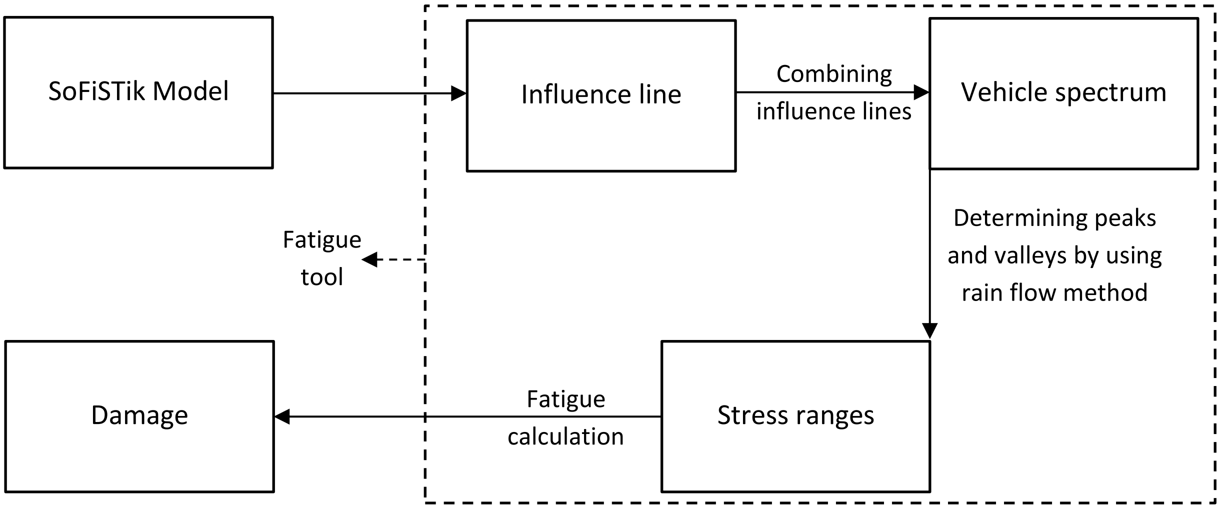


Figure 48 Calculation flow chart

5.1 Influence lines

SOFiSTik has an export module which can give requested values for each detail. This is exported in excel workbooks with separate sheets per detail with different sheets for each wheel geometry and track. These exports are with units as stated in chapter 3.0. An example of an export is shown in Figure 49. This consists of the Load case, load, node number, node group and the requested result, in this case a stress in MPa. The load has been set on a value of 50 kN.

LC: 10001-10123				
LC	LC-title	NR	NG	sxl [MPa]
10001	sum_PZ= 50.0(	10	1	-1,99
10002	sum_PZ= 50.0(	10	1	-3,26
10003	sum_PZ= 50.0(	10	1	-3,31
10004	sum_PZ= 50.0(	10	1	-3,35
10005	sum_PZ= 50.0(	10	1	-3,38
10006	sum_PZ= 50.0(	10	1	-3,38
10007	sum_PZ= 50.0(	10	1	-3,37
10008	sum_PZ= 50.0(	10	1	-3,33
10009	sum_PZ= 50.0(	10	1	-3,28

Sheet1 | Geo A track -2 | Geo B track -2 | Geo C track -2 | Geo A track -1 | Geo B track -1

Figure 49 Example excel export (detail 7)

An already existing tool will be used for the fatigue calculation to obtain correct stress ranges and calculate the damage for a certain wheel track. This tool requires an influence line with an equal step size of 0.1. Not all load steps are the same size however due to the load step optimization, thus an interpolation is necessary. This interpolation is done in excel, more explanation about this procedure can be found in chapter 6.6.1.

Figure 50 shows an example of an influence line after interpolation. This is an influence line of detail 8 which is in cross section A in in the bottom of the stiffener (Figure 20). When the load is directly above the cross-section tension stresses are highest, stresses are 0 when above the cross-beams and compression stresses occur when the load is above the spans next to the main span which is all exactly as expected.

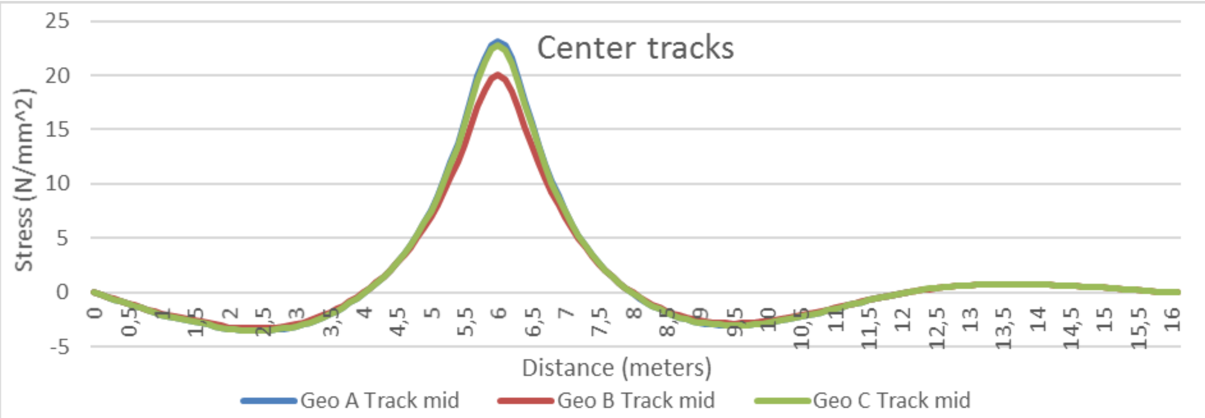


Figure 50 Example excel influence line (detail 7)

5.2 Vehicle spectrum

Next the influence lines need to be combined to a vehicle spectrum which represents the lorries stated in chapter 2.0. The procedure will be explained with an example. Vehicle type 3 will be analysed for track 5 for long distance traffic. The following parameters belong to this lorry:

Table 4 lorry parameters

Axle distance [dm]	0	32	52	13	13
Wheel geometry	a	b	c	c	c
Axle load [kN]	70	150	90	90	90

The influence lines as shown in Figure 50 belong to a load of 50 kN. This needs to be factorized to obtain the correct axle load. A multiplication with 0.7 is necessary for wheel A, 1.5 for wheel B and 0.9 for wheel C. The results are shown in Figure 51.

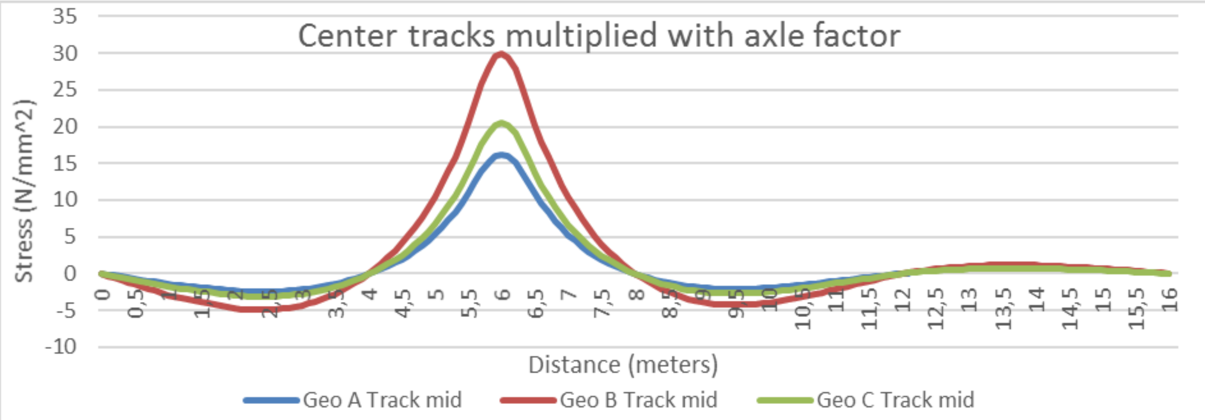


Figure 51 Example excel influence line factorized (detail 7)

Next the influence lines need to be superposed in accordance with the axle distance to form the lorry. The results can be found in Figure 52.

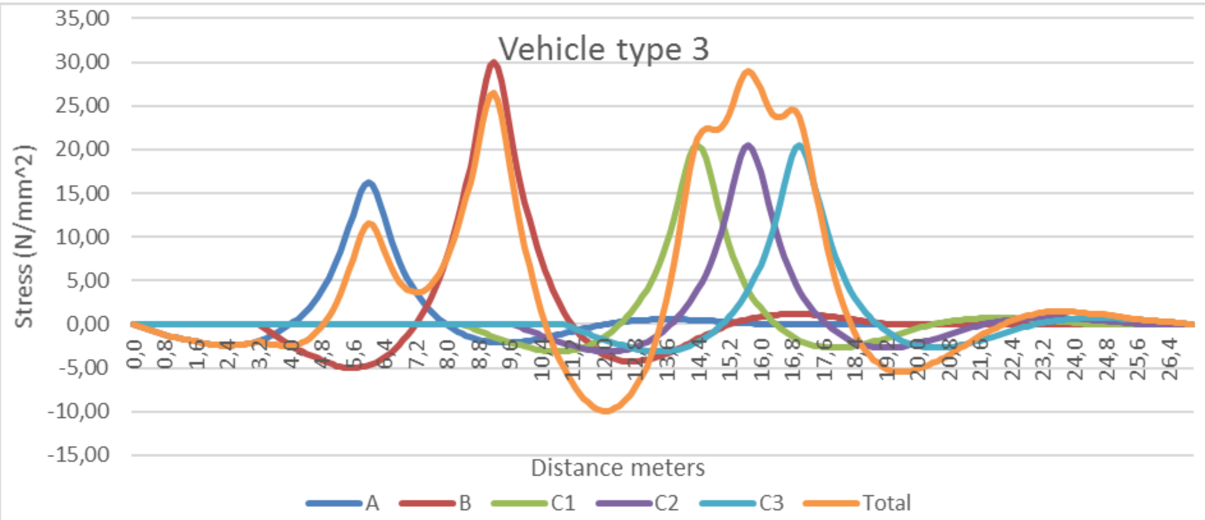


Figure 52 Spectrum vehicle type 3 (detail 7)

The orange line is the resulting influence line for this vehicle. This shows the stresses occurring in the structure if the first wheel of the lorry is at a distance X from the start of the model. The maximum stress is 29 $\frac{N}{mm^2}$ at X=15.7 which is the exact moment that the second "C" wheel is above the detail location.

5.3 Stress range

The stress ranges, or so-called fatigue ripples, can be determined from the vehicle spectrum in multiple ways, one of which is the rain-flow counting method. This method consists of multiple steps and will be explained below.

1. Rearrange the graph to start with the highest peak.
   The maximum has been determined previously thus the graph can be rearranged as shown in Figure 53.

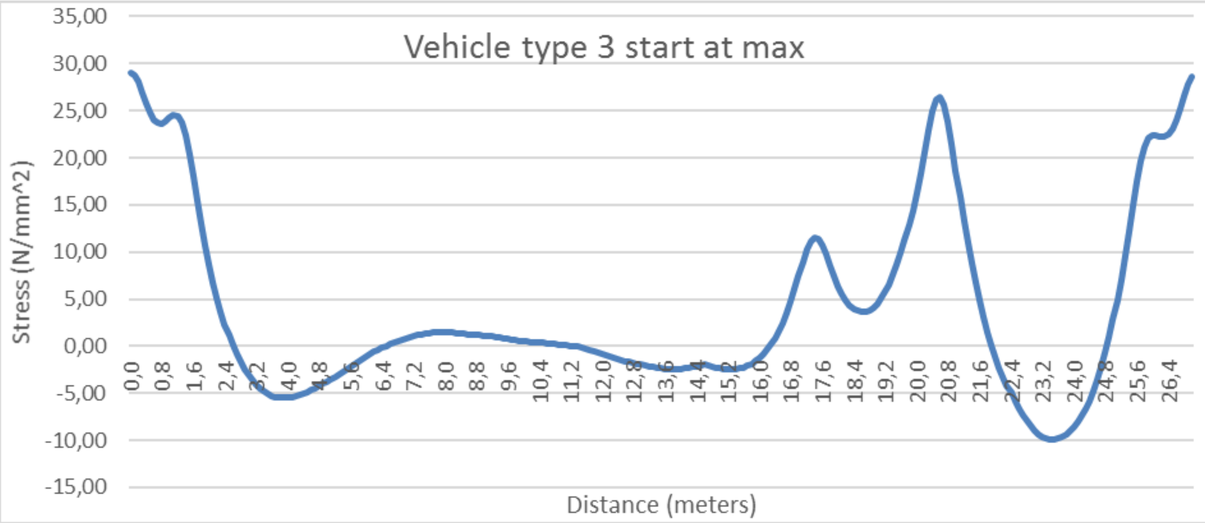


Figure 53 Spectrum vehicle type 3 start at max (detail 7)

2. Find all the peaks and valleys and create a new graph from this.
   With the help of excel a simple IF() statement can be used to see whether the next value is smaller than the current value. The moment it returns a false value instead of a true value a valley has been found. The moment it returns a true value after a false value a peak has been found. These peaks and valleys arranged after each other results in Figure 54 and has a close resemblance to Figure 53.

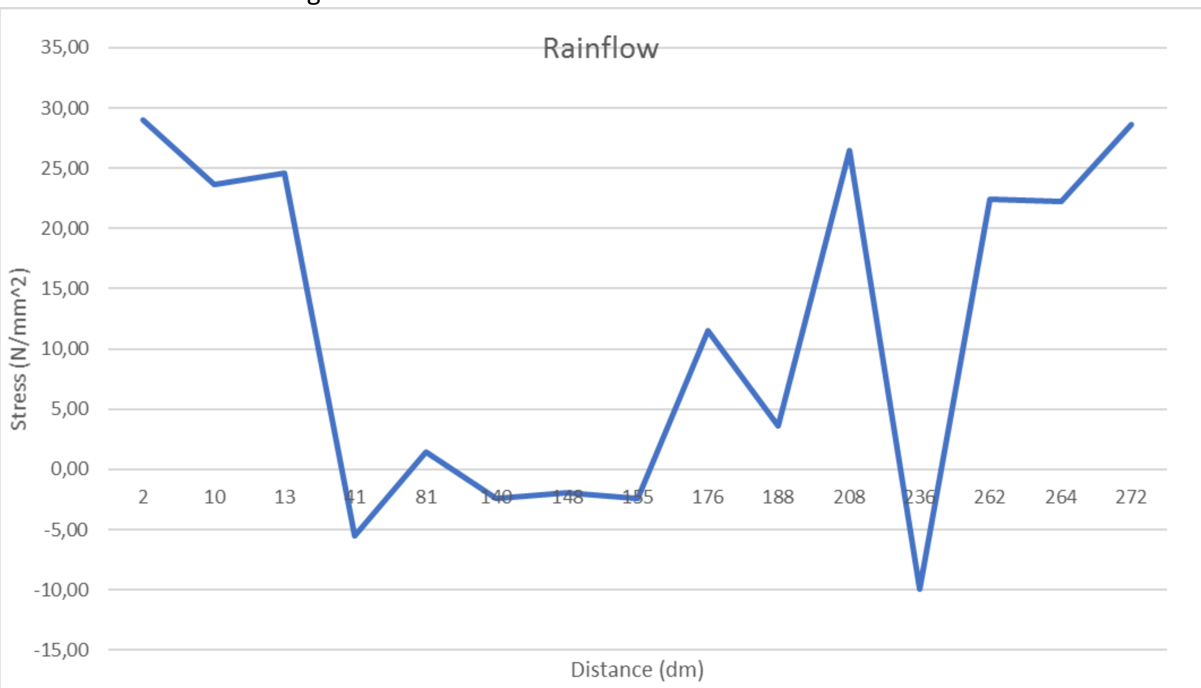
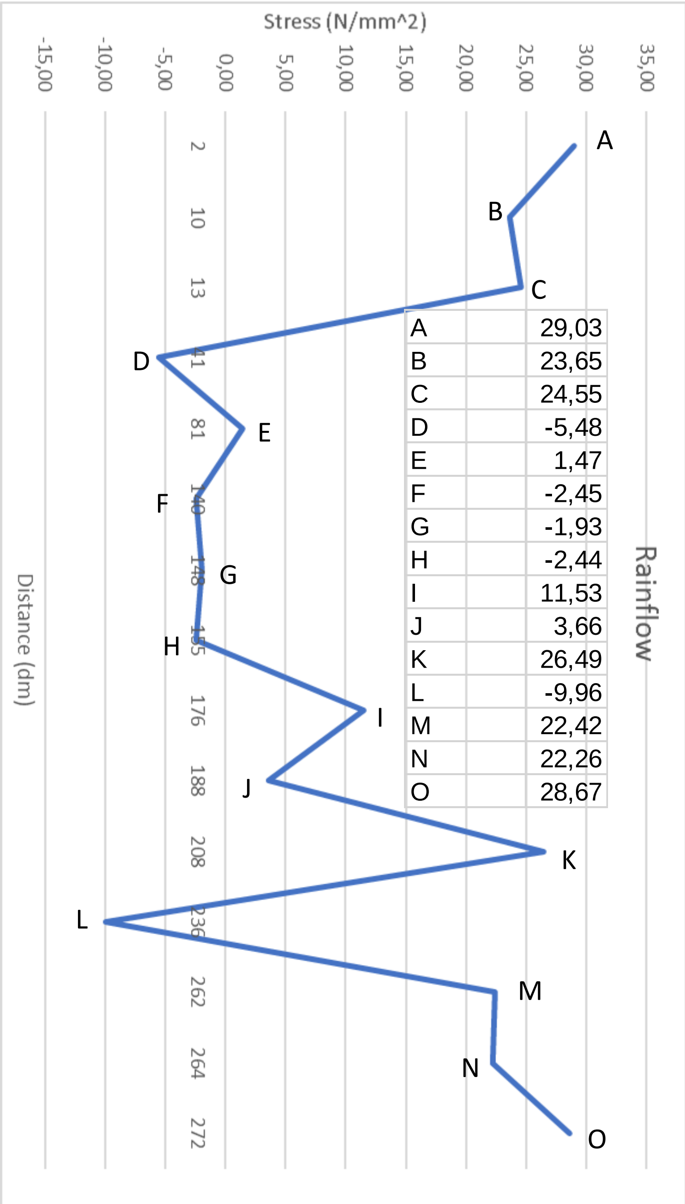


Figure 54 Rain-flow diagram vehicle type 3 (detail 7)

3. Rotate the graph clockwise 90° and count the stress ranges.

Each stress range from peak to valley has been numbered and gathered in Figure 56

4. Sea each peak/valley as a source of water and let water drip down (see Figure 56)



A	29,03
B	23,65
C	24,55
D	-5,48
E	1,47
F	-2,45
G	-1,93
H	-2,44
I	11,53
J	3,66
K	26,49
L	-9,96
M	22,42
N	22,26
O	28,67

Figure 56 stress ranges vehicle type 3 (detail 7)

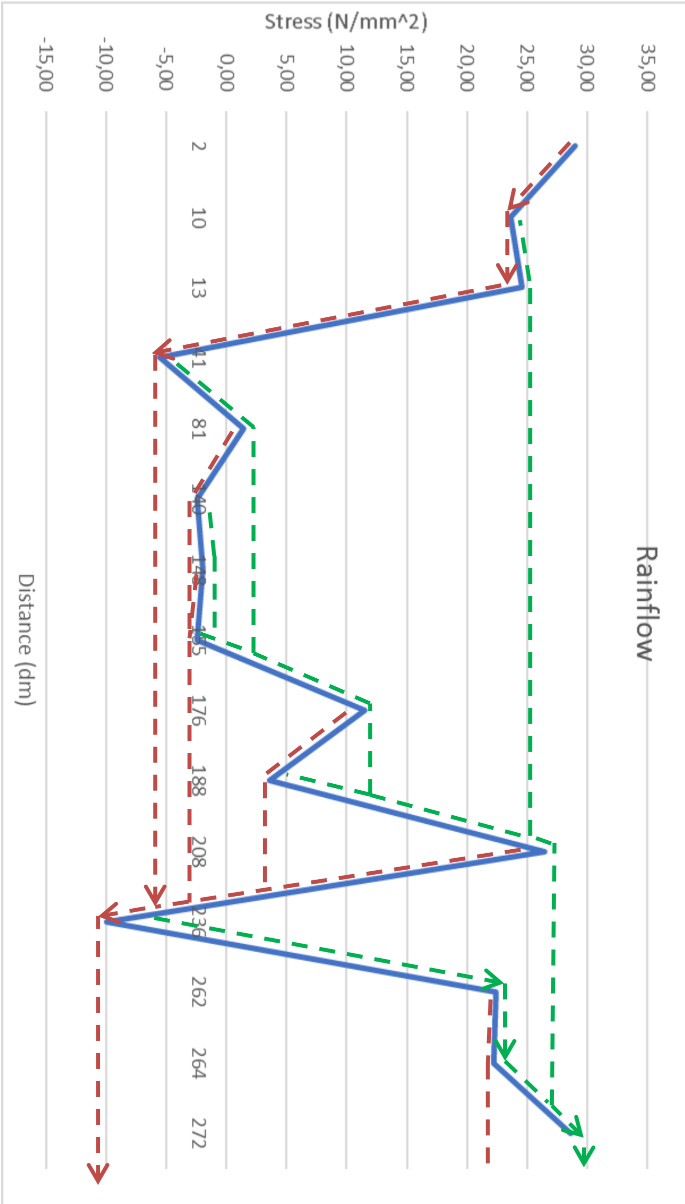


Figure 56 Rain flow for stress ranges (detail 7)

For each valley in the graph the water will drip down to the right (green lines) for each peak the water will drip down to the left (red lines). In case of the red lines the highest peak must be found first and water starts dripping down from there. This results in a line of water running from A through B and D to L with a stress range of $29.03 - -9.96 = 38.99\ MPa$ as is shown by the red arrows. The next biggest range will start from the next peak, in this case K and will run until it finds the already existing flow. The results in a stress range running from K to D before it merges with the already existing flow of water. This process is repeated until there are no peaks left as a water source. Then the same process is done starting from the valleys as is shown by the green lines.

5. Count the number of occurrences of each stress range.

Since the bigger the stress range the bigger the damage the task is to find the highest stress ranges possible without using the same line twice. An overview can be found in Table 5 and is in accordance with the lines shown in Figure 56.

Table 5 stress range overview detail 7

Start	Finish	Stress	Start	Finish	Stress		Stress range	Stress	Occurence
A	L	38,99	L	O	38,63		$\Delta\sigma_1$	38,99	1
K	D	31,97	D	K	31,97		$\Delta\sigma_2$	38,63	1
I	J	7,88	J	I	7,88		$\Delta\sigma_3$	31,97	2
E	F	3,91	F	E	3,91		$\Delta\sigma_4$	7,88	2
C	B	0,90	B	C	0,90		$\Delta\sigma_5$	3,91	2
G	H	0,51	H	G	0,51		$\Delta\sigma_6$	0,90	2
M	N	0,16	M	N	0,16		$\Delta\sigma_7$	0,51	2
							$\Delta\sigma_8$	0,16	2

The different stress ranges have been rearranged to form a list of stress ranges in descending order. Next to this a column is added which sums up the number of occurrences of that specific stress range. The two biggest stress ranges will always result in a single occurrence, all the others will result in 2 occurrences. These occurrences can also be called the number of half-cycles.

5.4 Damage

Before the damage can be calculated a few more steps need to be taken. First of N_i from equation (5.1) must be determined.

$$D_{f,tot} = \sum \frac{n_i}{N_i} \tag{5.1}$$

N_i can be determined using the Eurocode (EN1993-1-9 chapter 7.1). Equation (5.2) is taken from this norm and rewritten.

$$\begin{gathered} N_i = \frac{\Delta\sigma_C^m * 2 * 10^6}{\Delta\sigma_R^m} \; for \; \Delta\sigma_R \geq \Delta\sigma \; with \; m = 3 \\ N_i = \frac{\Delta\sigma^m * 5 * 10^6}{\Delta\sigma_R^m} \; for \; \Delta\sigma > \Delta\sigma_R \geq \Delta\sigma_\mathrm{L} \; with \; m = 5 \end{gathered} \tag{5.2}$$

Here $\Delta\sigma_c$ is the detail category and $\Delta\sigma$ is the constant amplitude fatigue limit (CAFL) as explained in chapter 2.0. Furthermore, the CAFL and cut-off limit can be determined according to equation (5.3).

$$\begin{gathered} \Delta\sigma = \left(\frac{2}{5}\right)^{\frac{1}{3}} * \Delta\sigma_C \\ \Delta\sigma_L = \left(\frac{5}{100}\right)^{\frac{1}{5}} * \Delta\sigma \end{gathered} \tag{5.3}$$

A detail category of 80 will be used for this calculation to make sure some damage will occur. This results in a CAFL and cut-off limit as shown in equation (5.4).

$$\begin{gathered} \Delta\sigma = 0.737 * 80 = 58.94 \\ \Delta\sigma_L = 0.549 * 58.94 = 32.38 \end{gathered} \tag{5.4}$$

All stresses below $\Delta\sigma_L$ will not have to be considered according to the norm. This means many of the stress ranges can be left out. Before any stress ranges can be removed from further calculation the $\Delta\varphi_{fat}$ must be implemented. This is a factor 1.15 as explained in chapter 2.0. This results in the stress ranges as shown in Table 6. In the same table n_i is shown. This follows from the number of vehicles that drive by per year ($2 * 10^6$), the percentage for vehicle type 3 (50%) and the number of full stress cycles. For $\Delta\sigma_1$ and $\Delta\sigma_2$ there is only a half cycle present, the rest has 1 full cycle per vehicle that passes by. The damage $D_{f,i}$ can be determined using equation (5.1).

Table 6 Fatigue damage calculation

Stress range	Stress	Stress(inc. φ)	N_i	n_i	$D_{f,i}$
$\Delta\sigma_1$	38,99	44.84	$1{,}96E + 07$	500.000	0,0255
$\Delta\sigma_2$	38,63	44.42	$2{,}06E + 07$	500.000	0,0243
$\Delta\sigma_3$	31,97	36.77	$5{,}29E + 07$	1.000.000	0,0189

This results in a total damage for vehicle type 3 of 0.0687 per year as is shown in equation (5.5).

$$D_{f,tot} = 0.0255 + 0.0243 + 0.0189 = 0.0687 \tag{5.5}$$

This means that after a period of 14.5 years this structural detail would not be safe according to equation (5.6). Usually structures are made to last for 100 years which is not the case for this calculation. However, a lower detail category has been chosen, the true fatigue damage lies below 0 when using the detail category of 160 which corresponds with detail 7.

$$D_{f,tot}: 0.0687 * n_{years} < 1.0 \; for \; n_{years} = 14.56 \tag{5.6}$$

5.5 Fatigue tool

Calculating the damage for 1 vehicle is a tedious job as can be seen in the previous chapter. From the influence line of the wheels the procedure is always the same, therefore Royal Haskoning DHV made a tool that does the calculation automatically, fatigue_bridge.xlsx. This is an excel workbook with multiple input sheets (calculation, input influence line, input vehicles, input Nobs) some calculation sheets, a python script and two result sheets (export_vehicles, export_damages). The calculation sheet produces data that is necessary for the python script to run properly. The python script is the core of this tool and does the full calculation. In the following paragraphs, the input and result sheets will be explained and the python script will be verified with the previous example (vehicle type 3 for long distance traffic).

5.5.1 Calculation ("berekening")

This is the main sheet where fatigue parameters are stated. For this report only the detail category and the dynamic amplification factor $\left(\Delta\varphi_{fat}\right)$ and $\gamma_{FF}, \gamma_{\mathrm{MF}}$ are of use. Other partial factors can be stated in this sheet as well but these are not of interest for this project.

5.5.2 Input influence line ("invoer invloedslijnen")

Here the influence lines must be given for wheel geometry A, B and C. This must be done with an equal step size of 0.1 meter. There is an option to insert an influence line for an extra traffic lane which is used for oncoming or overtaking traffic. Since only local details are considered in this model this is unnecessary to consider and can therefore be set to zero.

5.5.3 Input vehicles ("invoer voertuigen")

The next step is to give the input for the vehicles. This is done in three main columns namely, axle distance, wheel type and factor unit load. First of the axle distances must be given with a unit of dm. Then the wheel types corresponding to the previously given axles must be stated. And last a factor for the axle load. This depends on the load used in the model to determine the influence lines. In this case a load of 50 kN has been used. This results in the data as shown in Table 7 which corresponds with the demands stated in the norm as shown in chapter 2.0.

Table 7 Input vehycles

Naam	Jaren		% Nobs	As-afstanden [m]										Wieltypen (-)										Factor eenheidslast laag									
	van	tot		1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10
T1L	2000	2099	20%	0,0	4,5									a	b									0,70	1,30								
T2L	2000	2099	5%	0,0	4,2	1,3								a	b	b								0,70	1,20	1,20							
T3L	2000	2099	40%	0,0	3,2	5,2	1,3	1,3						a	b	c	c	c						0,70	1,50	0,90	0,90	0,90					
T4L	2000	2099	25%	0,0	3,4	6,0	1,8							a	b	c	c							0,70	1,40	0,90	0,90						
T5L	2000	2099	10%	0,0	4,8	3,6	4,4	1,3						a	b	c	c	c						0,70	1,30	0,90	0,80	0,80					

To complete the vehicle definition a name should be given and the percentage of appearance from the norm must be used. In this case the names refer to the vehicle row from the norm (T1=Type1) and the last letter refers to the traffic type (L=Long, M=medium, S=short). The years for which the calculation is done follow from the input Nobs and do not have to be given here.

5.5.4 Input Nobs ("invoer Nobs")

Nobs shows the number of vehicles that pass by each year and the number of years that should be included for the fatigue calculation. A reduction or growth in vehicle numbers can also be implemented here but this is not given in the norm.

Table 8 Input Nobs

Begin levensduur		2000	[-]
Eind levensduur		2099	[-]
Aantal jaren		100	[jr]
Referentiejaar		2010	[-]
Wisselingen per jaar		2.000.000	[Nobs]
Reductie per jaar		0,00%	[%]

5.5.5 Python

With all the previously mentioned data the python script can be run. This script follows the steps from chapter 4.0 to 4.0 but uses a loop to calculate the damage for each vehicle given in the “input vehicles”.

5.5.6 Export vehicles

This sheet shows the results of the calculation for the vehicle spectrum and the stress ranges. Figure 57 shows this spectrum and when comparing it to Figure 53 it shows almost the exact same results. The only difference is that the influence line from the fatigue tool skipped the values with a stress of zero, shortening the influence line by 2 dm. This is however only in the excel export and is not present in the further fatigue calculation and therefore causes no problems.

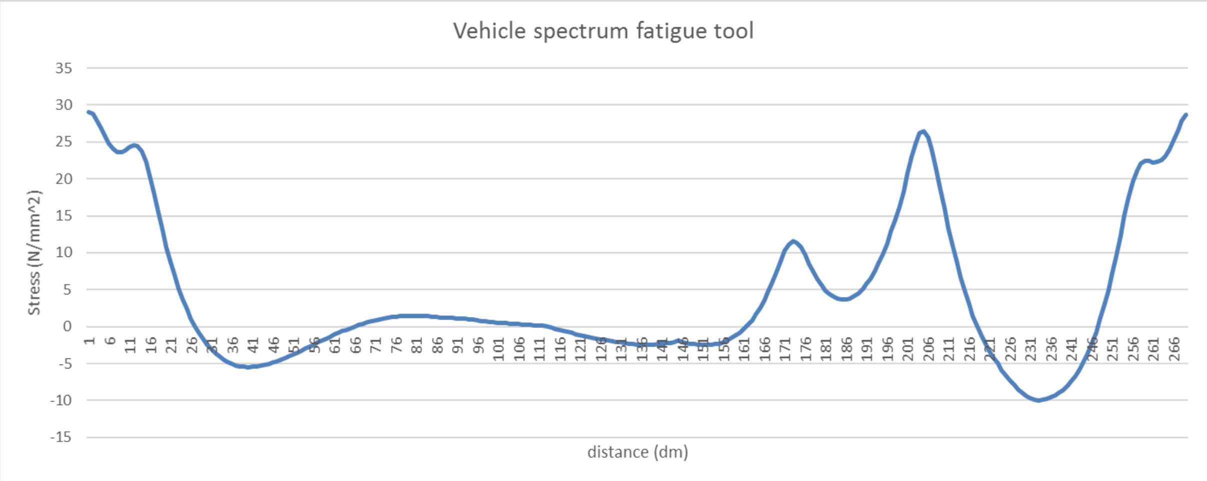


Figure 57 Vehicle spectrum fatigue tool

The fatigue ripples are also given in this sheet and are determined with the rain-flow counting method. They are sorted in descending order and include the number of full cycles present (0,5 meaning a half-cycle). Figure 58 shows the results and they are the same as the stress ranges determined in Table 5. These values do not include the $\Delta\varphi_{fat}$ yet.

(0.16, 1.0)
(0.51, 1.0)
(0.9, 1.0)
(3.91, 1.0)
(7.88, 1.0)
(31.97, 1.0)
(38.63, 0.5)
(38.99, 0.5)

Figure 58 Fatigue ripples

5.5.7 Export damages

Here the final results of the calculation are shown. Table 9 shows a part of the results. Here the damage is given per year for each vehicle. Since the Nobs is the same each year the damage is the same as well. As can be seen here the damage for vehicle T3L is 0.0687 which is the same as from the hand calculation. Vehicles T2 and T4 also result in damage to the structure and vehicle T1 and T5 don't.

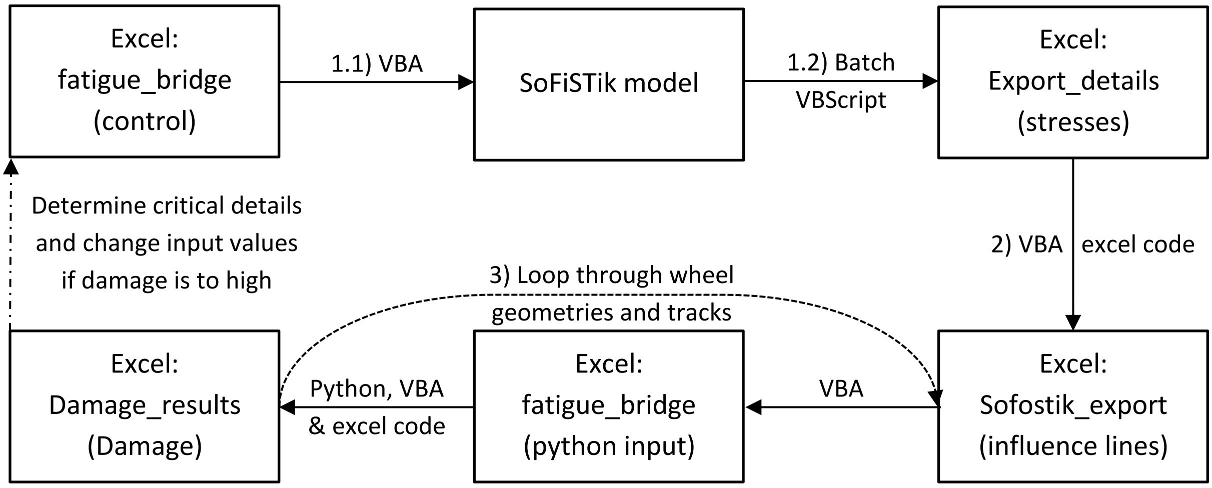
Table 9 Export damages fatigue tool

		2000	2001	2002	2003	2004	2005
T1L A low	0,2	0	0	0	0	0	0
T2L A low	0,05	0,002702	0,002702	0,002702	0,002702	0,002702	0,002702
T3L A low	0,5	0,068667	0,068667	0,068667	0,068667	0,068667	0,068667
T4L A low	0,15	0,007386	0,007386	0,007386	0,007386	0,007386	0,007386
T5L A low	0,1	0	0	0	0	0	0

By doing a summation with the damage results it becomes clear if the structure fails $(D > 1.0)$ or that the structure is safe $(D < 1.0)$.

6 Automated program

In the previous chapter a program is explained that can determine the fatigue damage from an influence line. In this case however there are 24 influence lines for each detail, so a total of 264 influence lines to be considered. Copying three influence lines at a time into the fatigue tool is therefore undesirable. To automate this process macro's will be used in excel which are coded in visual basics (VBA). It is also desirable to have one workbook from which everything is controlled including the FEM-model in SOFiSTik. This will also be done with the help of a macro. A program flow chart of the complete process can be found in Figure 59.



VBA=visual basics (excel macro), Batch=DOS (windows command), VBScript=visual basics (windows command)

Figure 59 Program flow chart

1. First of in this sheet the SOFiSTik program must be made and run to obtain stress results for each of the details.
2. After that the results from SOFiSTik will have to merged and organized in one excel workbook to give an overview of the stresses per step for each detail, wheel geometry and wheel track. This is done in the workbook SOFiSTik_export.
3. Subsequently the previously mentioned python script must be used and looped through to obtain the damage results. First a set of wheel geometries must be copied to the input sheet, then the python script must be run, after that the damage results should be gathered in an excel workbook (Damage_results) and finally this process has to be looped through for each detail and set of wheel tracks.

An explanation on these 3 steps can be found in the following paragraphs. All the macro's and other necessary files can be found in the annex. The fourth and final step is to combine all this together which is explained in the last paragraph of this chapter

6.1 Excel to SOFiSTik

The goal is to be able to run the program from one excel workbook. In this workbook the input must be given and the final damage results should be stated in this workbook as well. The sheet called "control" is the main sheet where the user can give the input values for the model and decide which parts of the program to run. The sheet called "results" is the sheet where all damage results are gathered. This includes the damage values per track and a combined damage value using the frequency distribution. It also gives an overview of the details being considered. The third and final sheet in this workbook is called "explanation" and gives a brief explanation on the processes that the program can do. Each of these sheets is shown in Figure 60 to Figure 62.

Next to this workbook there is one other workbook that needs input which is the fatigue_bridge workbook created by Nick Elbers. Since this workbook is changed and updated every now and then the choice has been made to keep it separate from the main workbook in case a newer version is available. In this workbook the input has to be given in the same way as explained in chapter 5.5.5.

Values that can be changed to create personalized model, recommendations/standard values are given

Values that can be changed but not recommended due to SOFiSTik code or eurocode norms

Input parameters for SOFiSTik model				
Input Parameters	in code	Value	Unit	Value Range
Steel trength	#mt_Fy_ste		Mpa	See EN1993-1-1 Table 3.1
Deck thickness	#br_t_deck		m	0,018-0,024 m for highway bridge
Asphalt thickness	#br_t_asph		m	0,05 or more
Stiffener top width	#br_w_tstif		m	0,2-0,35 m
Stiffener bottom width	#br_w_bstif		m	0<...<#br_w_tstif
Stiffener height	#br_h_stif		m	0<...<#br_h_cb
Stiffener thickness	#br_t_stif		m	0,006-0,008 m
Stiffener center-to-center distance	#br_ctc_stif		m	0,4-0,66 m
Cross-beam height	#br_h_cb		m	#br_h_stif<...
Cross-beam thickness	#br_t_cb		m	0,01-0,016 m
Cross-beam center-to-center distance	#br_ctc_cb		m	3,0 - 5,0 m
Distance to stiffener butt-weld from crossbeam	#br_elong_stif		%	in percentages of cross beam span
Distance to deck butt-weld from crossbeam	#br_elong_deck		%	in percentages of cross beam span

Details	YMf	Categorie		Demands for detail category
detail 1	1,15	125		
detail 2	1,15	125		
detail 3	1,15	125		
detail 4	1,15	100		detail 4 is verbonden met behulp van een automatisch lasproces
detail 5	1,15	112		lassen vlakgeslepen, lasuitloopstukken verwijderd en plaatranden geslepen in spanningsrichting, gelast aan beide zijden
detail 6	1,15	125		
detail 7	1,15	80		
detail 8	1,15	160		
detail 9	1,15	112		lassen vlakgeslepen, lasuitloopstukken verwijderd en plaatranden geslepen in spanningsrichting, gelast aan beide zijden
detail 10	1,15	100		
detail 11	1,15	80		

Make sure you also fill in the fatigue calculation input in the file "fatigue_bridge_v1.7.xlsm" in folder "Damage_results"

standard mesh size	#md_std_msh	0,075	m	0,05-0,1 (for calculation time less is not advised)
mesh refinement at details	#md_ref_msh	0,010	m	0,01-0,05 (for calculation time less is not advised)
Loading longitudinal step size	#ld_std_xstep	0,00	m	Smaller or equal to #br_ctc_cb/4 (round of on 1 decimal)
Loading longitudinal step size refinement	#ld_ref_xstep	0,10	m	0,1-0,5 (round of on 1 decimal) 0,1 for damage calculation!
Loading number of transverse steps	#ld_n_ystep	8	-	8 is standard, less works as well
Loading transverse step size	#ld_std_ystep	0,1	m	0,1 in accordance with Eurocode
Model with crossbeam cut-out	#br_cutout_cb	Yes	-	Yes as standard. If no then detail 6 changes (calculate by hand)

☐ Create and run Teddy file

☐ Results gathering

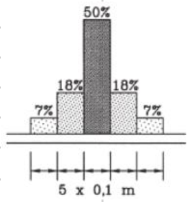
☐ Damage calculation

☐ PDF export

Run program

Figure 60 Main workbook: control sheet

	Track 1	Track 2	Track 3	Track 4	Track 5	Track 6	Track 7	Track 8	Frequency distribution
Detail 1									0,07
Detail 2									0,18
Detail 3									0,5
Detail 4									0,18
Detail 5									0,07
Detail 6									
Detail 7									
Detail 8									
Detail 9									
Detail 10									
Detail 11									



	Frequency distribution implemented								D_max	Center track
Detail 1	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	**0,00**	**No Damage**
Detail 2	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	**0,00**	**No Damage**
Detail 3	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	**0,00**	**No Damage**
Detail 4	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	**0,00**	**No Damage**
Detail 5	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	**0,00**	**No Damage**
Detail 6	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	**0,00**	**No Damage**
Detail 7	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	**0,00**	**No Damage**
Detail 8	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	**0,00**	**No Damage**
Detail 9	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	**0,00**	**No Damage**
Detail 10	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	**0,00**	**No Damage**
Detail 11	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	**0,00**	**No Damage**

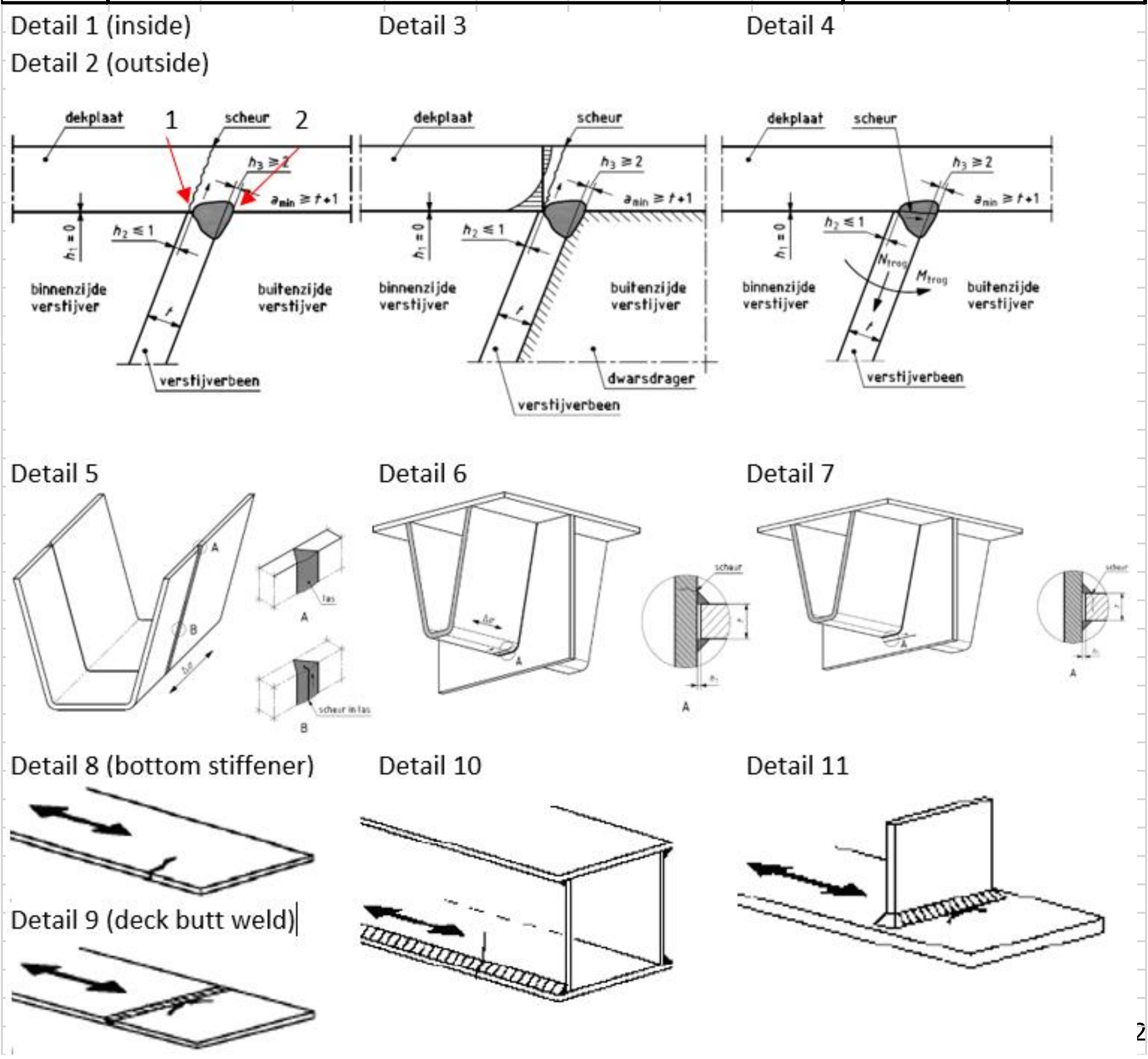


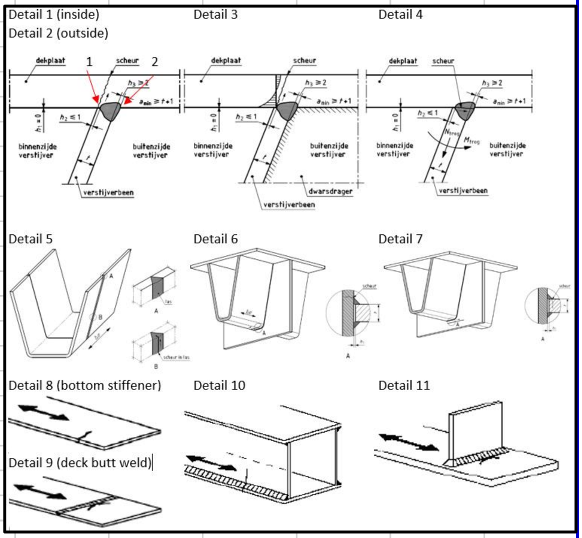
Figure 61 Main workbook: results sheet

Detailed fatigue calculation for an orthotropic bridge deck

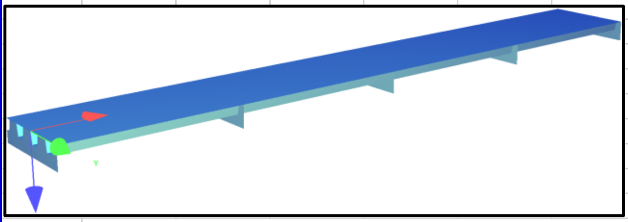
Created by Maarten van der Wateren

This program determines fatigue damage values for 11 details in an orthotropic bridge deck following fatigue load model 4A from the Dutch national annex NEN-EN 1991-2 NB. The details are shown below and consist of all local details in the stiffeners, deck and cross beams (no main girders). These details can be found in NEN-EN 1993-2 NB and NEN-EN 1993-1-9.

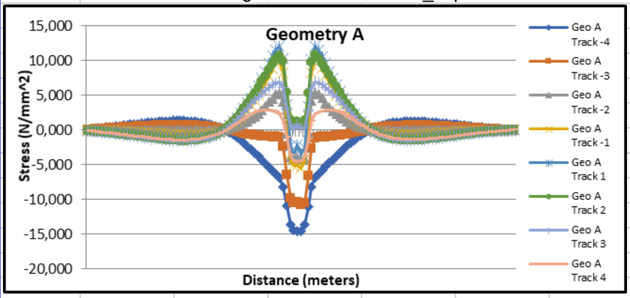
Step 1 Input all project parameters in the "control" sheet in this workbook. Yellow and orange values can be altered by the user. Orange values however are recommended to keep as they are for an optimal model.



Step 2 Create SOFiSTik model and determine stresses
Done for 3 wheel geometries allong 8 tracks in X-direction (red)
SOFiSTik model and export created in SOFiSTik folder



Step 3 Create influence lines from SOFiSTik model
Influence lines are gathered in SOFiSTik_export



Step 4 Determine fatigue damage per detail per track using a python tool created by Nick Elbers.
Each case is stored in the folder "damage_results" and all damage values are gathered in "Damage_overview".
Damage values are combined using the frequency distribution resulting in a maximum damage value for each detail.
The center track for which the frequency distribution has led to a maximum has been stated in the column.

	Frequency distribution implemented								Center track	MAX
Detail 1	0,00	0,06	0,24	0,65	0,79	0,29	0,09	0,00	Track 5	0,79
Detail 2	0,05	0,08	0,16	0,48	0,29	0,45	0,15	0,07	Track 4	0,48
Detail 3	0,25	0,19	0,38	0,14	0,11	0,20	0,54	0,52	Track 7	0,54
Detail 4	1,12	1,49	2,81	4,06	4,75	4,34	3,77	2,36	Track 5	4,75
Detail 5	4,04	8,72	13,45	15,41	13,27	9,09	4,39	2,25	Track 4	15,41
Detail 6	3,14	1,96	4,27	9,44	14,67	17,06	14,61	9,09	Track 6	17,06
Detail 7	2,56	2,11	3,83	7,00	10,22	11,76	10,30	6,79	Track 6	11,76
Detail 8	0,68	0,44	1,03	1,98	3,06	3,54	3,11	1,95	Track 6	3,54
Detail 9	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	No Damage	0,00
Detail 10	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	No Damage	0,00
Detail 11	0,00	0,00	0,21	0,77	2,09	2,14	0,79	0,22	Track 6	2,14

More information on the steps in between can be found in my internship report (linkje ergens op de schijf??)

Figure 62 Main workbook: explanation sheet

6.1.1 Making the model

First of the SOFiSTik model must be made just as explained in chapter 4. To make the SOFiSTik file the complete text has been copied to excel. To simplify things for the user it is unwanted to change parameters inside the SOFiSTik text, therefore the parameters in SOFiSTik have been linked to the control sheet. The user must give all the necessary parameters in this sheet to make the model. A short description and the name of the parameters is given as can be seen in Figure 60. The 7 (orange) parameters at the bottom of this sheet can be altered for users who understand the program, however this might give errors or conflicts if done incorect. Therefore it is recommended to keep these values as they are and with the help of data validation a warning will be given if the user tries to edit them.

When the user has filled out all the values the macro can be started to create and run teddy. This macro copies and exports the text to a new excel file with a period as delimiter which is necessary for SOFiSTik. After that it saves the exported file as a general ".dat" file which can be opened by teddy.

To give the command to run the model from excel some extra files are needed, one to open the model and one to press "F12" which is the shortkey to run the model. This is done with the help of .vbs file (visual basic script) and .bat file (batch file) respectively. This start the calculation and takes about one hour. As the model runs it can be placed in the background and only after it has finished the next step can be done. The code used for this part can be found in Appendix D: Excel macro "Teddy_export" explanation.

To run all these commands the user only has to check the box next to "create and run teddy file" and press run program.

6.2 SOFiSTik results to SOFiSTik_export

After the teddy file has run and required stresses have been exported the information should be gathered in one clear workbook, called SOFiSTik_export.xlsm. This can be done by checking the box in the control sheet next to “results gathering” and pressing “run profram”.

First off, all the raw data is copied to this workbook in one sheet “SOFiSTik”. An extra sheet is needed to obtain the hot spot stresses for detail 3 and 6 which is called “Hot spot details”. The input parameters are needed for further calculations and are therefore copied to the sheet “Control”. Some calculations are done with the data from the control sheet and are gathered in the sheet “Lists”. All the other sheets are for the influence lines of each detail (1 to 11) which retrieves its data from the previously mentioned tabs. Each of the tabs will be explained briefly in this chapter.

Detail 1	Detail 11	Control	Lists	SOFiSTik	Hot spot details

6.2.1 SOFiSTik

This is where all the results are gathered with the use of a macro called SOFiSTik_export_macro. This macro opens all the export sheets, copies the data from each separate sheet and pastes it into SOFiSTik tab. After it has copied the data it closes the export sheets again. The explanation of this macro can be found in Appendix E: Excel macro “Results_gathering” explanation. An example of this sheet can be seen in Figure 63.

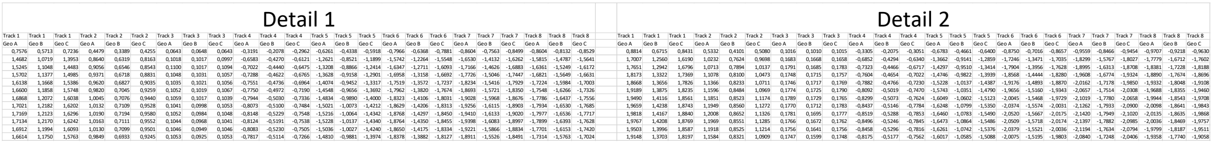


Figure 63 Overview SoFiSTik sheet

6.2.2 Hot spot details

Here the stress results for detail 3 and 6 are calculated. This is done using the extrapolation equation (3.3) as explained in chapter 3.3.7 and looks the same as the SOFiSTik sheet.

6.2.3 Control

Excel can calculate quicker when there are no references outside of its own workbook. Therefore the “control” sheet from the main workbook is copied to the SOFiSTik_export workbook.

6.2.4 Lists

The data from SOFiSTik does not have a constant step size and thus needs an interpolation. Interpolation is done with the help of a macro, see Appendix F: Excel formula "SOFiSTik_export" explanation.

To do so it is necessary to know which step distances have been calculated in SOFiSTik. This can be determined by the input parameters given in the "control" sheet with the help of an excel calculation. Next to this column there is a column with the steps needed for the influence line, values from 0 to the length of the model with a step size of 0.1. The third column determines which of these two columns contain the same steps.

An example of this can be seen in Table 10. Here X_SOFiSTik shows the steps used in the SOFiSTik model, X_python shows the steps needed for the python calculation and Sof_Pyt shows the steps present in both lists.

Table 10 influence line steps

X_SoFiSTiK	X_python	Sof_Pyt
0	0	
1	0,1	
2	0,2	
2,1	0,3	
2,2	0,4	
2,3	0,5	
2,4	0,6	
2,5	0,7	
2,6	0,8	
2,7	0,9	
2,8	1	1
2,9	1,1	
3	1,2	
3,1	1,3	
3,2	1,4	
3,3	1,5	
3,4	1,6	
3,5	1,7	
3,6	1,8	
3,7	1,9	
3,8	2	1
3,9	2,1	1
4	2,2	1

6.2.5 Detail sheets

Here the influence line is formed with the help of the previously mentioned "Lists" sheet. For each step size where the column Sof_Pyt returns 1 the corresponding stress is copied from the SOFiSTik sheet. For each value where the Sof_Pyt column returns nothing a linear interpolation will be done with the nearest two known points. For the point $x = 0$ the stress is fixed at 0 as explained in chapter 3.3.6. An explanation on how the excel code works can be found in Appendix F: Excel formula "SOFiSTik_export" explanation. The detail sheets also include some graphs to have a quick glance at the influence lines. Errors can be found more easily this way. An example of such graph is shown in Figure 64 for geometry A of detail 4.

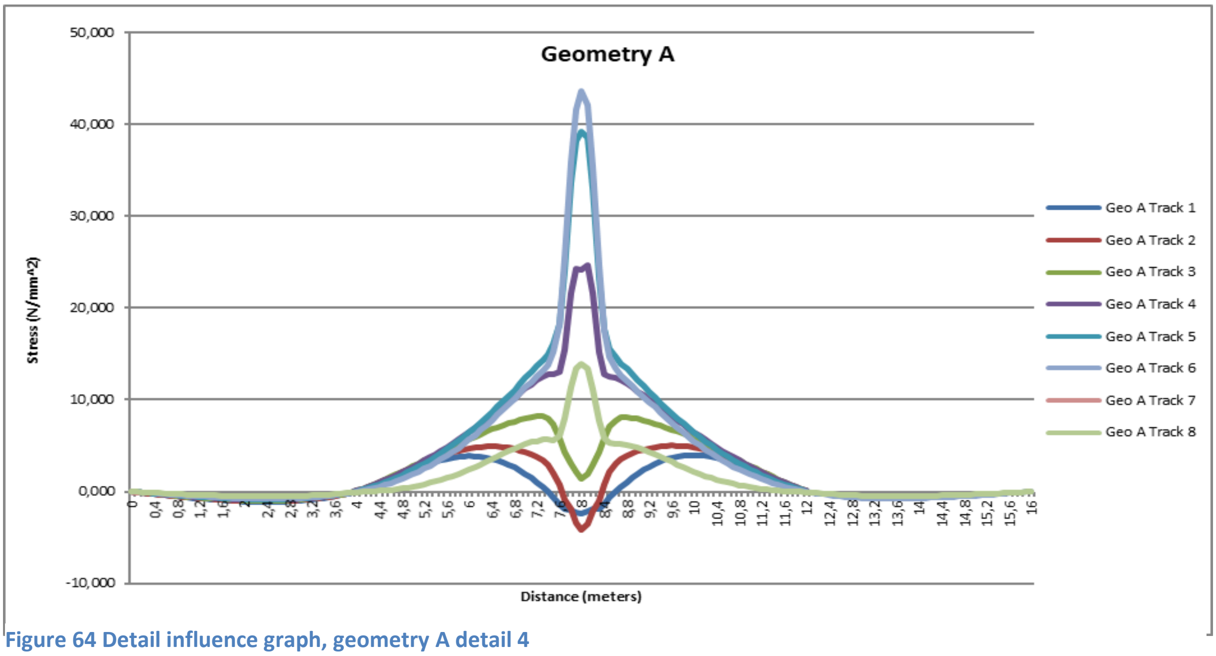


Figure 64 Detail influence graph, geometry A detail 4

6.3 Stress influence lines to damage results

The next step is to export the influence lines to the fatigue workbook in which the python script is implemented and calculate the damage. Then the process as explained in chapter 5.5.5 needs to be followed. The steps are shown in step three of Figure 59 and are listed below in short:

1. Copy influence line to fatigue_bridge
2. Run python script
3. Save fatigue_bridge as detail … track …
4. Copy damage to "results" sheet
5. Repeat for each wheel track and each detail.

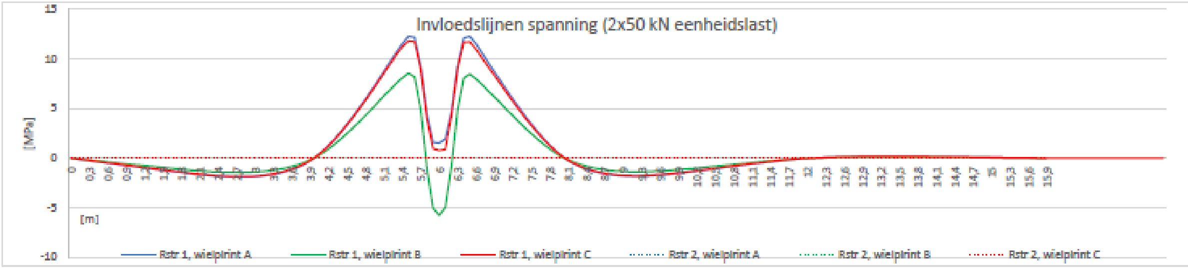
This process is also done in a macro and is as straight forward as the 4 steps mentioned above. The macro code for this process can be found in Appendix G: Excel macro "Fatigue_calculation" explanation.

6.3.1 Damage_results

For each track and each detail a damage calculation is done using the fatigue workbook made by Nick Elbers. After each calculation this workbook is saved in a folder called damage_results. After that all the total damages per detail per track are gathered in the "results" sheet of the main workbook as is shown in Figure 61. Here the damages per track are combined in accordance with the frequency distribution and the maximum damage per detail is highlighted.
Each detail and track is saved as a separate file in case some results in the outcome seem to be strange and a user wants to check the calculation. In this workbook the user can check whether for instance the correct detail category has been used during calculation. The first sheet "calculation" gives an overview and shows the influence line, vehicles, paramters for the S-N curve, Nobs and the damage results per vehicle over the years. An example can be seen in Figure 65.

Spanningen

Permanente belasting	σ_{perm}	0	MPa	druk
Verkeersbelasting (eenheidslast)	F_{TS}	2x50	kN	

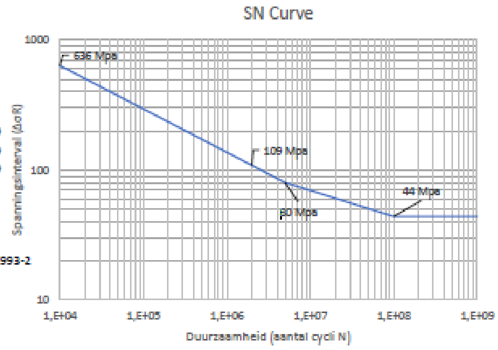


Voertuigen

Naam	Jaren	α	Asconfiguratie [dm]	Wieltype [-]	As-coefficienten licht [-]	As-coefficienten midden	As-coefficienten zwaar
T1	[2000, 2099]	20%	[0, 45]	[a, b]	[0.7, 1.3]	[0.7, 1.3]	[0.7, 1.3]
T2	[2000, 2099]	5%	[0, 42, 13]	[a, b, b]	[0.7, 1.2, 1.2]	[0.7, 1.2, 1.2]	[0.7, 1.2, 1.2]
T3	[2000, 2099]	40%	[0, 32, 52, 13, 13]	[a, b, c, c, c]	[0.7, 1.5, 0.9, 0.9, 0.9]	[0.7, 1.5, 0.9, 0.9, 0.9]	[0.7, 1.5, 0.9, 0.9, 0.9]
T4	[2000, 2099]	25%	[0, 34, 60, 18]	[a, b, c, c]	[0.7, 1.4, 0.9, 0.9]	[0.7, 1.4, 0.9, 0.9]	[0.7, 1.4, 0.9, 0.9]
T5	[2000, 2099]	10%	[0, 48, 36, 44, 13]	[a, b, c, c, c]	[0.7, 1.3, 0.9, 0.8, 0.8]	[0.7, 1.3, 0.9, 0.8, 0.8]	[0.7, 1.3, 0.9, 0.8, 0.8]

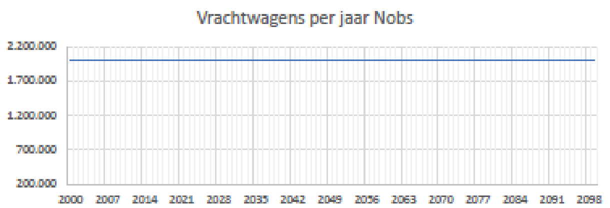
Vermoeiingsparameters

Schadeberekening				
Detailcategorie	$\Delta\sigma_c$	125	MPa	
Trendreductie aslast	γ_T	0,00	%	
Referentiejaar voor lastreductie	T_{ref}	2000	[-]	
Factoren - NEN-EN 1991-1-9				
Partiële factor voor de vermoeiingssterkte $\Delta\sigma_C$, $\Delta\tau_C$	γ_{Mf}	1,15	NEN EN 1991-1-9	
Partiële factor voor de spanningsintervallen	γ_{Ff}	1,00	NEN EN 1991-1-9	
Reductiefactor vermoeiingsterkte door geometrische effecten	k_s	1,00	NEN EN 1991-1-9	
Reductiefactor drukspanning voor niet gelaste verbindingen	$k\sigma_c$	1,00	NEN-EN 1993-2	
Vergrotingsfactor bij voegen (dx = 6m vanaf voeg)	$\Delta\varphi_{fat}$	1,15	NEN-EN 1993-2	
Dynamische correctiefactor door onvlakheden in de weg	φ_{s1}	1,00	NEN 8701	
SN curve				NEN-EN 1993-2
Helling tak 1	m1	3	-	
Helling tak 2	m2	5	-	
Aantal wisselingen minimaal	N_0	1,E+04	[N]	
Aantal wisselingen bij Δσc	N_1	2,E+06	[N]	
Aantal wisselingen knikpunt	N_2	5,E+06	[N]	
Aantal wisselingen afkaplimiet	N_3	1,E+08	[N]	



Optredende wisselingen

Begin levensduur	J1	2000	[-]
Eind levensduur	J2	2099	[-]
Aantal jaren	ΔJ	100	[jr]
Referentiejaar voor Nobs	J_{ref}	2010	[-]
Wisselingen per jaar	N_{obs}	2E+06	[N_{obs}]
Reductie per jaar	P_{nobs}	0,00%	[%]



Resultaten

Totale schades

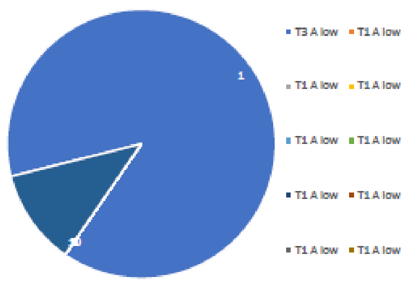
Periode		Jaar	Schadefactor D_{tot}
2000	-	1989	-
2000		2050	0,441
2000	-	2099	0,752

Aantal schadeberekeningen	500
Aantal schadegevallen	100
Percentage onder afkaplimiet	80,0%

Top 10 schades voertuigcombinaties

	Voertuig combinatie	schade	percentage
1	T3 A low	0,865	114,9%
2	T1 A low	0	0,0%
3	T1 A low	0	0,0%
4	T1 A low	0	0,0%
5	T1 A low	0	0,0%
6	T1 A low	0	0,0%
7	T1 A low	0	0,0%
8	T1 A low	0	0,0%
9	T1 A low	0	0,0%
10	T1 A low	0	0,0%
	Overige combinaties		-14,9%

I = periode [2000, 2099]
II = periode
III = periode
A-D = Voertuigcombinatie (zie tabel voertuigcombinatie)



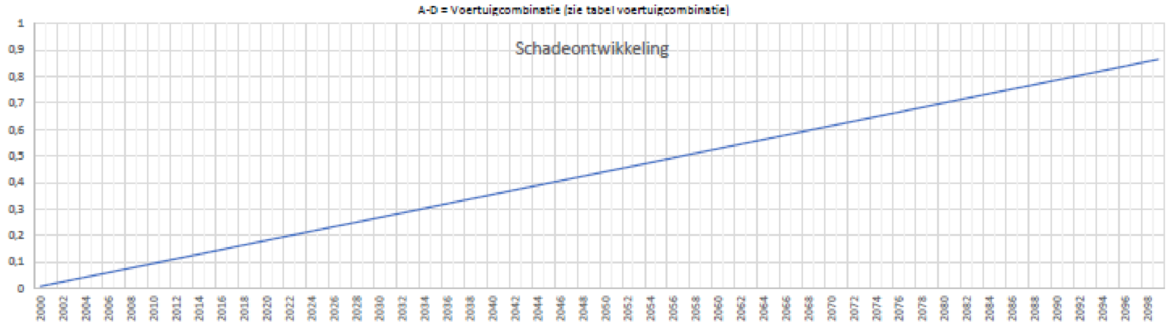


Figure 65 Fatigue_bridge calculation sheet

6.4 Creating the program

The final goal is to combine each step and create a working program. Two types of users can be destinguished, one who runs the program on the laptop they are working on at the moment and the other running the program on a sepperate computer. The first user might want to start the calculation of the model in SOFiSTik and would like to start the post calculation manually. This way excel can still be used while SOFiSTik is running. Whereas the other user just wants to run the complete program and come back after a few hours to see it is finished. To take care of both situations the user is given a few options to decide which parts of the program should or should not be executed.

These options are given in the control pannel shown in Figure 66 which is part of the control sheet shown in Figure 60. Here the user gets 4 options which can be chosen all together or seperatly.
The first option, Create and run Teddy file, creates the model for SOFiSTik and runs it resulting in the stress results being exported in separate excel files.
The second option, Results gathering, takes these excel files and gathers them in the SOFiSTik_export workbook.
The third option is the fatigue calculation using the separate sheet and python script.
These three options coincide with the three paragraphs of this chapter (chapter 6).
The last option is the creation of a PDF-export of the input and results. The macro for this part can be found in Appendix H: Excel macro “PDF_export” explanation.

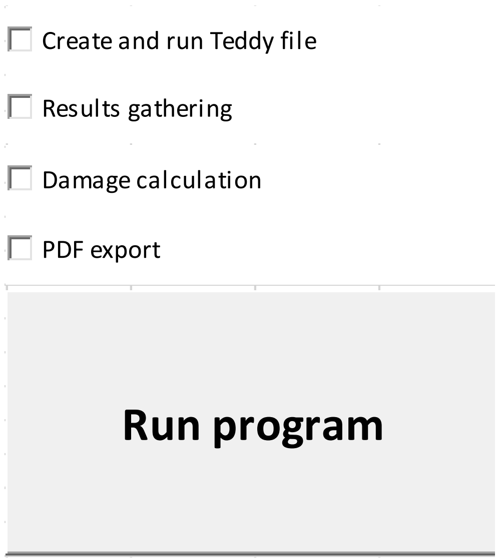


Figure 66 Control panel

By checking the separate boxes and pressing “Run program” a vba code starts and calls the separate macro’s depending on the boxes checked. Extra information is given in the end on the runtime of the separate parts. To do this a timer is included in the code. The code itself can be found in

7 Appendix

7.1 Appendix A: English detail description

Detail 1 **Detail category:** 125 **Location:** crack in deck at a location between cross-beams **Crack type:** Initiated from the weld between deck and stiffener, can crack at both sides of stiffener **Crack growth:** through the thickness of the deck starting at the weld **Stress range:** calculated as a local nominal stress at the bottom of the deck at the rupture initiation point in a 3D model. $a_{\min \leq t} + 1mm$ **Pre-processing:** Stiffener leg chamfering up until a weld angle of 50°. For SAW-welding and $t \leq 6\ mm$ no chamfering needed. **Split:** $h_1 = 0\ mm$; for 10% of the length $h_1 < 0.5\ mm$ is allowed.	**Detail 2** **Detail category:** 125 **Location** deck at junction of stiffener with crossbeam **Crack type:** Initiated from the weld between deck and stiffener **Crack growth:** through the thickness of the deck starting at the weld root **Stress range:** calculated as a local 'hot spot stress' at the bottom of the deck at the crack initiation point in a 3D model or with SCF factor as given in NEN-EN1993-2 F.2 $a_{\min \leq t} + 1mm$ **Pre-processing:** Stiffener leg chamfering up until a weld angle of 50°. For SAW-welding and $t \leq 6\ mm$ no chamfering needed. **Split:** $h_1 = 0\ mm$; for 10% of the length $h_1 < 0.5\ mm$ is allowed.
Detail 3 **Detail category:** 100 (automatic) or 90 (manually welded) **Location:** weld between stiffeners and deck at a location between cross-beams **Crack type:** Initiated from the weld **Crack growth:** through the thickness of the weld starting at the weld root or the weld toe **Stress range:** calculated as a local nominal stress in the stiffener leg in a 3D model $a_{\min \leq t} + 1mm$ **Pre-processing:** Stiffener leg chamfering up until a weld angle of 50°. For SAW-welding and $t \leq 6\ mm$ no chamfering needed. **Split:** $h_1 = 0\ mm$; for 10% of the length $h_1 < 0.5\ mm$ is allowed.	**Detail 4** **Detail category:** 80[a] **Location:** factory made weld at extension of the stiffener **Crack type:** Initiated from the weld surface **Crack growth:** through the thickness of the weld **Stress range:** calculated as a nominal stress at the bottom of the stiffener leg in a 3D model **Pre-processing:** Both sides chamfered, V-seam 60° **Tolerance:** Alignment error $\leq 0.5\ mm$ **Front opening:** $\geq 4.0\ mm$ **Weld:** manual weld in factory
Detail 5 **Detail category:** 80 for $t \leq 12$, 71 for $t \geq 12$, 125[b] **Location:** Connection between stiffener and cross-beam **Crack type:** crack in stiffener starting from the weld toe **Crack growth:** through the thickness of the stiffener **Stress range:** calculated as a nominal stress at the bottom of the stiffener leg in a 3D model. If distance between bottom stiffener and bottom cross-beam is less than 500 mm 'hot spot stress' must be used **Pre-processing:** split $h_1 \leq 1.0\ mm$ **Weld:** manual weld in factory	**Detail 6** **Detail category:** 80 **Location:** Connection between stiffener and cross-beam **Crack type:** crack in cross-beams **Crack growth:** through the thickness of the web of the cross-beam from the weld toe **Stress range:** calculated as a nominal stress at the web of the cross-beam located at the weld toe due to bending out of plane of the web $a_{min} \geq 5\ mm$ **Pre-processing:** split $h_1 \leq 1.0\ mm$ **Weld:** manual weld
Detail 7 **Detail category:** 160 **Location:** in plate material **Crack type:** crack in plate **Crack growth:** through the thickness of the plate **Stress range:** calculated as a nominal stress **Demands:** Sharp edges, surface errors and rolling errors must be removed to obtain a smooth transition zone	**Detail 8** **Detail category:** 1) 112, 2) 90, 3) 80 **Location:** At transverse butt weld **Description:** Transverse splices in plates and flats **Demands:** Weld run-on and run-off pieces to be used and removed, plate edges ground flush in direction of stress, welded from both sides. 1) ground flush to plate surface 2) maximum weld convexity of 10 % of weld with 3) maximum weld convexity of 20 % of weld with
Detail 9 **Detail category:** 100 **Location:** At longitudinal weld **Description:** Manual or automatic or fully mechanized butt welds carried out from one side only, particularly for box girders.	**Detail 10** **Detail category:** 100 **Location:** At transverse attachments **Description:** Cross-beams connection welded to deck plate **Demands:** Ends of welds to be carefully ground to remove any undercut that may be present.

Demands: A very good fit between the flange and web plates is essential. The web edge to be prepared such that the root face is adequate for the achievement of regular root penetration without break-out.	

7.2 Appendix B: SOFiSTik commands

SOFiSTik is a finite element program which can be used both on a visual basis and through a text-editor. When used with visual input it can work together with multiple programs such as Rhino, Revit or AutoCAD. However, for parametric design the text-editor is used. The text-editor that is used is called Teddy and works only for SOFiSTik. This is due to the fact that it uses its own "code" instead of running on python for example.

7.2.1.1 Teddy (text-editor)

Teddy uses different modules to produce certain results. Below each module will be explained including its contents to get an understanding of how the program works. A short table with the used commands will also be given. Within every module the commands as shown in Table 11 can be used.

Table 11 General commands SOFiSTik

General	
STO	stores values that will be remembered throughout the whole script.
LET	stores values that will be remembered only in the applied module
LOOP	creates a loop, can assign the loop value to a parameter
IF	creates an if statement
ECHO	Control of output (set NO to decrease calculation time)

7.2.1.2 TEMPLATE

Template is a module which is solely used for the input parameters. This module doesn't do anything else than assigning values to the given parameters with the STO command.

7.2.1.3 AQUA

Aqua is the first module used which determines the material properties and the dimensions of certain cross sections. In this case all the parameters that will be used must be given here. The final goal is that this input can be done via a different program where aqua retrieves the values it needs.

Table 12 AQUA commands SOFiSTik

AQUA	
NORM	defines the design codes that will be used during calculation
STEE	defines the material properties for steel
UNIT	defines the units that are used throughout the script.

7.2.1.4 SOFIMSHC

Sofimshc makes the mesh, and the element definitions. In this case plate elements are used and each side of a plate element has to be given as input here. Together with the boundary conditions this makes the entire model.

Table 13 SOFIMSHC commands SOFiSTik

SOFIMSHC	
SYST	defines the global system used
GDIV	defines the maximum number of elements per group
GDIR	defines the direction of gravity
CTRL OPT mesh	defines mesh properties
CTRL OPT hmin	defines standard mesh size
SPT	defines a structural point by assigning a number and X, Y and Z coordinates.
SLN	defines a structural line by assigning a number and start & end points (SPT)
SAR	defines a structural area, assigns a number, group, material and thickness
SARB	defines the boundaries (SLN) that build up the previously mentioned SAR.

7.2.1.5 SOFILOAD

Sofiload defines the different load cases for the model. Instead of a line load a step load has been used with small increments to obtain relevant results.

Table 14 SOFILOAD commands SOFiSTik

SOFILOAD	
ECHO LOAD	defines the output of the calculation
CTRL WARN	supresses a certain warning during calculation
LAR	defines a load area
LC	defines a load case
AREA	assigns a load area to the previously mentioned load including a force direction and size

7.2.1.6 ASE

Ase is the analyses module that produces the results of the different load cases. This is the part that takes up the most time of the program.

Table 15 ASE commands SOFiSTik

ASE	
LC	analyses a certain load case

7.2.1.7 RESULTS

The results module exports the results into tables. In this case the results are exported to an excel workbook to make it readable for python.

Table 16 Results commands SOFiSTik

RESULTS	
PICT	makes a new picture/table for the information to come
FILT	filters the results, for instance one node
LC	defines which load cases should be analysed
XLSX	exports results to excel, assigns name, worksheet and information about the results
QUAD	the actual results that must be exported (stress/moments/forces)

7.3 Appendix C: EN1993-1-1

Standard and steel grade	Nominal thickness of the element t [mm]			
	$t \leq 40$ mm		40 mm $< t \leq 80$ mm	
	f_y [N/mm^2]	f_u [N/mm^2]	f_y [N/mm^2]	f_u [N/mm^2]
EN 10025-2				
S 235	235	360	215	360
S 275	275	430	255	410
S 355	355	AC2 490 AC2	335	470
S 450	440	550	410	550
EN 10025-3				
S 275 N/NL	275	390	255	370
S 355 N/NL	355	490	335	470
S 420 N/NL	420	520	390	520
S 460 N/NL	460	540	430	540
EN 10025-4				
S 275 M/ML	275	370	255	360
S 355 M/ML	355	470	335	450
S 420 M/ML	420	520	390	500
S 460 M/ML	460	540	430	530
EN 10025-5				
S 235 W	235	360	215	340
S 355 W	355	AC2 490 AC2	335	490
EN 10025-6				
S 460 Q/QL/QL1	460	570	440	550
EN 10210-1				
S 235 H	235	360	215	340
S 275 H	275	430	255	410
S 355 H	355	510	335	490
S 275 NH/NLH	275	390	255	370
S 355 NH/NLH	355	490	335	470
S 420 NH/NLH AC2	420	540	390	520
S 460 NH/NLH	460	560	430	550
EN 10219-1				
S 235 H	235	360		
S 275 H	275	430		
S 355 H	355	510		
S 275 NH/NLH	275	370		
S 355 NH/NLH	355	470		
S 460 NH/NLH	460	550		
S 275 MH/MLH	275	360		
S 355 MH/MLH	355	470		
S 420 MH/MLH	420	500		
S 460 MH/MLH	460	530		

7.4 Appendix D: Excel macro “Teddy_export” explanation

The macro “Teddy_export” copies part of the excel sheet, rearranges it to a file that SOFiSTik can read and executes it.

First of in this macro a verification is done to check whether the folder in which the document is situated doesn’t contain any spaces. If it does it skips to an error message telling the user to change the directory.

The text that must be copied is the text of the whole model as explained in Chapter 4. This text has been placed in an excel column and the input given in excel sheet “control” has been linked to the input parameters in the module “Template”. The rest of the text is an exact copy of the model made in SOFiSTik since it uses the input parameters.

```
Sub Teddy_export()

With Application
.DecimalSeparator = "."
.ThousandsSeparator = ","
.UseSystemSeparators = False
End With

Dim spaces As Integer
spaces = ThisWorkbook.Sheets("Control").Range("J1")
If spaces = 0 Then

Workbooks.Add
ChDir ThisWorkbook.Path
ActiveWorkbook.SaveAs Filename:=ThisWorkbook.Path & "\SOFiSTik" & "\Teddy_file.xlsx", _
FileFormat:=xlOpenXMLWorkbook, CreateBackup:=False
ActiveWindow.Close

Dim X As Workbook, y As Workbook
Dim ws1 As Worksheet, ws2 As Worksheet

Set X = ThisWorkbook
Set y = Workbooks.Open(ThisWorkbook.Path & "\SOFiSTik" & "\Teddy_file.xlsx")

X.Sheets("Control").Range("O:O").Copy: y.Sheets(1).Range("A:A").PasteSpecial xlPasteValues

y.SaveAs Filename:= _
X.Path & "\SOFiSTik" & "\Teddy_file.dat", _
FileFormat:=xlTextPrinter, CreateBackup:=False
y.Close

With Application
.UseSystemSeparators = True
End With

Dim Pathcrnt As String

Pathcrnt = X.Path & "\SOFiSTik"
Call Shell(Pathcrnt & "\Execute_teddy.bat " & Pathcrnt)

Else
MsgBox ("The directory of this file includes spaces and therefore won't work." & _
    "Please change spaces (' ') into underscores ('_').")
End If

End Sub
```

Figure 67 Teddy_export macro

This text is then exported to a new excel file and the delimiter is set on a period if it wasn´t already. After that the file is saved as a .dat file which can be read by SOFiSTik or more specifically teddy.

To open the newly made file a batch file is executed in the last lines of the macro. This batch file opens the new file with the program wps. This is the calculation part which is integrated in SOFiSTik but can be used seperatly. The batch file ends itself by opening a new file to execute the program which is made with VBScript.

BATCH FILE

```
Start "" "C:\Program Files\SOFiSTik\2018\SOFiSTik 2018\wps.exe" %cd%\Teddy_file.dat
wscript %cd%\Execute_teddy.vbs
```

To start the calculation the shortkey {F12} can be pressed. This action can be done with the use of a short script as stated below. This scripts tells the computer to wait until the program “ted” is opened, when it has opened it tells the computer to press {F12} and therefore executes the program.

VBScript

```
Set oShell = CreateObject("WScript.Shell")
WScript.Sleep 3000
oShell.AppActivate "ted"
WScript.Sleep 3000
oShell.SendKeys "{F12}"
```

7.5 Appendix E: Excel macro “Results_gathering” explanation

The Results_gathering macro copies the exports from SOFiSTik to the overview workbook SOFiSTik_export. It is copied from 13 workbooks (D1-D13) and pasted in the sheet SOFiSTik in separate columns.

Before it copies the export data it starts with copying the “Control” sheet of the main workbook. This is done to clarify which dimensions have been used for this specific output data if it is opened later. The data from the control sheet is also needed to interpolate the data points as is explained in chapter 6.6.2-Lists. The lines of code in Figure 68 perform these actions

```
Sub Results_gathering()

Dim D As Workbook, X As Workbook, y As Workbook, D1 As Workbook
Dim ws As Worksheet
Dim version As String, command As String, mydir As String
Dim I As Integer, J As Integer, A As Integer, B As Integer, C As Integer

Set X = ThisWorkbook
Set y = Workbooks.Open(X.Path & "\Sofistik_export.xlsm")
A = ThisWorkbook.Sheets("Control").Range(N_ystep) 'number of wheel tracks

'copy control input to Sofistik_export
Set ws = X.Sheets("Control")
Set ws0 = y.Sheets("Control")
Set ws1 = y.Sheets("SoFiSTik")
Set ws2 = y.Sheets("Lists")
ws.Range("B6:G80").Copy: ws0.Range("A1").PasteSpecial xlPasteValues
```

Figure 68 Results_gathering macro part 1

In the next part, all the detail export workbooks are opened so excel can access them. Then it starts to loop through the workbooks and sheets and copies it to the SOFiSTik sheet. During the loping process it updates the status bar telling the user how far it is in the process. To end this script all the separate workbooks of each detail are closed and the SOFiSTik_export workbook is calculated and saved. This process is done using the lines shown in Figure 69.

```
'Loop through sheets for each detail and copy to Sofistik_export
For J = 1 To 3 * A
    D1.Sheets(J + 1).Range("E4:E204").Copy: ws1.Cells(4, J + 0).PasteSpecial xlPasteValues
    D2.Sheets(J + 1).Range("E4:E204").Copy: ws1.Cells(4, J + 25).PasteSpecial xlPasteValues
    D3.Sheets(J + 1).Range("E4:E204").Copy: ws1.Cells(4, J + 50).PasteSpecial xlPasteValues
    D4.Sheets(J + 1).Range("E4:E204").Copy: ws1.Cells(4, J + 75).PasteSpecial xlPasteValues
    D5.Sheets(J + 1).Range("E4:E204").Copy: ws1.Cells(4, J + 100).PasteSpecial xlPasteValues
    D6.Sheets(J + 1).Range("E4:E204").Copy: ws1.Cells(4, J + 125).PasteSpecial xlPasteValues
    D7.Sheets(J + 1).Range("E4:E204").Copy: ws1.Cells(4, J + 150).PasteSpecial xlPasteValues
    D8.Sheets(J + 1).Range("E4:E204").Copy: ws1.Cells(4, J + 175).PasteSpecial xlPasteValues
    D9.Sheets(J + 1).Range("E4:E204").Copy: ws1.Cells(4, J + 200).PasteSpecial xlPasteValues
    D10.Sheets(J + 1).Range("E4:E204").Copy: ws1.Cells(4, J + 225).PasteSpecial xlPasteValues
    D11.Sheets(J + 1).Range("E4:E204").Copy: ws1.Cells(4, J + 250).PasteSpecial xlPasteValues
    D12.Sheets(J + 1).Range("E4:E204").Copy: ws1.Cells(4, J + 275).PasteSpecial xlPasteValues
    D13.Sheets(J + 1).Range("F4:F204").Copy: ws1.Cells(4, J + 300).PasteSpecial xlPasteValues
    Application.StatusBar = True
    Application.StatusBar = J & " out of 24 done"
Next J

'close all detail sheets
D1.Close: D2.Close: D3.Close: D4.Close: D5.Close: D6.Close: D7.Close: D8.Close: D9.Close: D10.Close: D11.Close: D12.Close: D13.Close
y.Activate
Application.CalculateFull
y.Save
y.Close

End Sub
```

Figure 69 Results_gathering macro part 2

7.6 Appendix F: Excel formula "SOFiSTik_export" explanation

In the excel sheet "SOFiSTik_export" several excel formulas have been used. Each of them will be explained briefly per sheet it is used in.

7.6.1.1 Lists

To obtain the correct influence lines from the exported results an interpolation is needed. This makes it necessary to know the exact X-coordinate used in each step in the SOFiSTik calculation. To determine this a code will be used to create a column with all the steps.

The first step is always 0, the second and third step use the big stepsize (#br_std_xstep) and can be determined by referring to the control sheet.
The next steps are determined by a little bit of code shown below.

```
=ROUND(IF(B4<ref_fin;B4+ref_xstep;IF(B4<t_L;B4+std_xstep;""));1)
```

Each value is rounded off to 1 decimal. The first IF statement checks if the previous step is smaller than the end of the refinement $\left(\#br_L - 2 * \#br_{std_{xstep}}\right)$. If so it takes the previous value and ads one small step. If the previous step is bigger than the end of the refinement it checks whether the previous value is still smaller than the total length of the model. If that is true than it takes the previous value and ads one big step, if it's not true it returns an empty cell.

For the interpolation, it is necessary to know which stress must be interpolated for a certain point and which stress can be copied from the export data. To do so a match function has been used to check which values occur in both the xsteps of the influence line (python) and in the xsteps of the SOFiSTik export.

```
=IFERROR(IF(MATCH(C3;Lists!$B$2:$B$250;0);1;"");"")
```

This match functions searches for the python step in the list of SOFiSTik export steps and if it finds a match it returns a one. This is enough for the interpolation function to determine the influence lines.

7.6.1.2 Details

To determine the influence lines from the SOFiSTik data a rather long formula is used which can be split up in two parts.
=IFERROR(IF(COUNT(Lists!$D$2:$D3)<1;VLOOKUP(std_xstep;$A$9:$Y$209;COLUMN();FALSE)/std_xstep*$A10;Linterp(Lists!$B$3:$B$252;SOFiSTik!A$4:A$254;$A10));"")
The first part is considered when the step being considered is smaller then the first step calculated in SOFiSTik. If this is the case the formula looks for the first value that is calculated by SOFiSTik, divides it by the step size and then multiplies by the actual step. This is actually just a linear interpolation between the first calculated value and 0 and is given by the red part of the formula.

If the step being considered is equal or larger then the first step in SOFiSTik it uses the linear interpolation function. The interpolation function needs three columns, one with the desired steps for which stresses are needed, one with X-coordinates of known stresses and the stress values themselves. With this input, it determines whether a value is known or whether it must interpolate. It searches for the nearest known values when interpolating and uses a linear interpolation pattern. The interpolation function has been made in a macro as is shown in Figure 70.

7.6.1.3 Hot spot details

For the hot spot details a simple formula is used to extrapolate the data obtained from SOFiSTik. It follows the formula as stated in chapter 3.3.7-Hot spot stress (detail 3 & 6). The following function in Excel has been used:

=(SOFiSTik!AY4-SOFiSTik!BX4)/2+SOFiSTik!AY4

```
Public Function Linterp(x As Range, y As Range, x_int As Double) As Variant
     ' linear interpolator / extrapolator
     ' Tbl is a two-column range containing known x, known y, sorted x ascending

    Dim nRow As Long
    Dim iLo As Long, iHi As Long

    nRow = x.Rows.Count
    If nRow < 2 Or x.Columns.Count <> 1 Then
        Linterp = CVErr(xlErrValue)
        Exit Function '--------------------------------------------------------->
    End If

    If x_int < x(1) Then ' x < xmin, extrapolate from first two entries
        iLo = 1
        iHi = 2
    ElseIf x_int > x(nRow) Then ' x > xmax, extrapolate from last two entries
        iLo = nRow - 1
        iHi = nRow
    Else
        iLo = Application.Match(x_int, Application.Index(x, 0), 1)
        If x(iLo) = x_int Then ' x is exact from table
            Linterp = y(iLo)
            Exit Function '----------------------------------------------------->
        Else ' x is between tabulated values, interpolate
            iHi = iLo + 1
        End If
    End If

    Linterp = y(iLo) + (y(iHi) - y(iLo)) _
    * (x_int - x(iLo)) / (x(iHi) - x(iLo))

End Function
```

Figure 70 Linterp visual basics (Modification of Linterp from Nick Elbers from Royal HaskoningDHV)

7.7 Appendix G: Excel macro “Fatigue_calculation” explanation

This macro does the actual fatigue damage calculation and can be split up in several parts. Each part will be explained briefly followed by the part of code that does it all.

The first part sets some parameters and creates names for the different sheets that are needed as is shown in Figure 71.

```
Sub Fatigue_calculation()

Dim w As Workbook, X As Workbook, y As Workbook, z As Workbook, D1, D2, D3, D4, D5, D6, D7, D8, D9 As Workbook
Dim ws1 As Worksheet, ws2 As Worksheet, ws3 As Worksheet, ws4 As Worksheet, ws5 As Worksheet, ws6 As Worksheet
Dim version As String, command As String, mydir As String
Dim I As Integer, J As Integer, A As Integer, B As Integer

'set some values necessary for the program

Set X = ThisWorkbook
A = X.Sheets("Control").Range("N_ystep") 'number of wheel tracks

Set ws = X.Sheets("Control")

Set y = Workbooks.Open(X.Path & "\Sofistik_export.xlsm")
Set z = Workbooks.Open(X.Path & "\damage_results" & "\fatigue_bridge_v1.7.xlsm")
B = z.Sheets("Berekening").Range("Jaren") + 2 'number of years

Set ws2 = z.Sheets("1. invoer invloedslijnen")
Set ws3 = z.Sheets("Berekening")
Set ws6 = X.Sheets("Results")
```

Figure 71 Fatigue_calculation macro part 1

The next part is the actual loop that loops through each different set of tracks per detail location. In this loop the value I represents the detail number and the value J represents the track number. In the first section the influence lines of a certain track and detail are copied to the fatigue_bridge workbook together with the detail category “Delta” and the “gamma” factor.

The next section is to start the python calculation. To do this the correct directory has to be opened and .exe file which is the python script needs to be executed. When the python script is done the macro continues. Some extra worksheets are named, in this case the damage export from the python script is named as “ws4” and the damage sheet in the fatigue_bridge workbook is named “ws5”.

Before copying the damage export to the damage sheet the export is first filtered. Several vehicles with a 0% of occurrence are removed. After this is done the damage export is copied to the damage sheet and the combined damage will be calculated in this sheet.

The final part is the copying of the combined damage to the results sheet in the main workbook to eventually determine the damage including the frequency distribution. After this is done the fatigue_bridge workbook is saved as Detail (I) Track (J) into the damage_results folder. This is shown in Figure 72.

```
'Copy input into fatigue workbook
For I = 1 To 11
    For J = 1 To A

        Set ws1 = y.Sheets(I)


        ws1.Activate
        ws1.Range(Cells(9, 3 * J - 1), Cells(209, 3 * J + 1)).Copy: ws2.Range("B5").PasteSpecial xlPasteValues
        ws2.Range("E5:G2961").FormulaR1C1 = "0"
        ws1.Range("B2").Copy: ws3.Range("Delta").PasteSpecial xlPasteAll
        ws1.Range("B3").Copy: ws3.Range("gamma").PasteSpecial xlPasteAll


        'Python calculation+schadetabel
        z.Save

        mydir = z.Path
        version = z.Worksheets("Berekening").Range("version")
        command = "fatigue_bridge_v" & version & ".exe"

        'run compiled python script
        CreateObject("wscript.shell").Run "cmd.exe /c cd /d " & mydir & " & " & command, 0, True

        Set w = Workbooks.Open(z.Path & "\export_damages_v" & version & ".xlsx")

        Set ws4 = w.Sheets(1)
        Set ws5 = z.Sheets("Schadetabel")

        'Remove rows with beta=0 as given in "berekening"

        ws4.Activate
        With ActiveSheet
            .AutoFilterMode = False
            With Range("B2", Range("B" & Rows.Count).End(xlUp))
                .AutoFilter 1, "0"
                On Error Resume Next
                .Offset(1).SpecialCells(12).EntireRow.Delete
            End With
            .AutoFilterMode = False
        End With

        'Copy results from export_damages to fatigue_bridge
        w.Save
        ws4.Range(Cells(1, 1), Cells(6, B)).Copy: ws5.Range("A20").PasteSpecial xlPasteAll
        w.Cells.ClearContents
        w.Close

        'Copy Damage total to Results sheet

        ws5.Range("D_total").Copy: ws6.Cells(I + 2, J + 3).PasteSpecial xlPasteValues

        z.SaveCopyAs (z.Path & "\Detail " & I & " track " & J & ".xlsm")
```

Figure 72 Fatigue_calculation macro part 2

The last part in the loop is a status bar update. This shows the user how far it is in the calculation showing it the detail and track number out of the total detail and track. When this is done it continues to the next track and/or detail. When the loop is completed it closes the SOFiSTik_export workbook and the Fatigue_bridge workbook and activates the control worksheet where the user has started. This part is shown in Figure 73.

```
        With Application
        .DisplayAlerts = True
        .ScreenUpdating = True
        End With

        Application.StatusBar = True
        Application.StatusBar = "Track " & J & " out of " & A & " Done of detail " & I & " out of 11"

        With Application
        .DisplayAlerts = False
        .ScreenUpdating = False
        End With

    Next J
Next I

y.Close
z.Close
ws.Activate

End Sub
```

Figure 73 Fatigue_calculation macro part 3

7.8 Appendix H: Excel macro “PDF_export” explanation

To create a nice overview for the user the results can be exported to a PDF format which can then be printed if required. This is done using the PDF_export macro and is shown in figure…
This macro creates a PDF-file with the name Detail_fatigue_calculation and adds the date and time the export was made. This way if a user creates a new export in the same folder it won’t overwrite the old export. This is a rather complex macro which consists of a combination of several macro’s found online and is shown in Figure 74.

```
Sub PDF_Export()

'www.contextures.com
'for Excel 2010 and later
Dim wsA As Worksheet
Dim wbA As Workbook
Dim strTime As String
Dim strName As String
Dim strPath As String
Dim strFile As String
Dim strPathFile As String
Dim myFile As Variant
On Error GoTo errHandler

Set wbA = ActiveWorkbook
Set wsA = ActiveSheet
strTime = Format(Now(), "yyyy\-mm\-dd\_hh\.mm")

'get active workbook folder, if saved
strPath = wbA.Path
If strPath = "" Then
  strPath = Application.DefaultFilePath
End If
strPath = strPath & "\"

strName = wbA.Name & "_" & strTime
strName = Replace(strName, ".xlsm", "")

'create default name for savng file
strFile = strName & ".pdf"
strPathFile = strPath & strFile

'export to PDF in current folder
    wbA.ExportAsFixedFormat _
        Type:=xlTypePDF, _
        Filename:=strPathFile, _
        Quality:=xlQualityStandard, _
        IncludeDocProperties:=True, _
        IgnorePrintAreas:=False, _
        OpenAfterPublish:=False
    'confirmation message with file info

ActiveWorkbook.FollowHyperlink (strFile)

exitHandler:
    Exit Sub
errHandler:
    MsgBox "Could not create PDF file"
    Resume exitHandler
End Sub
```

Figure 74 PDF_export macro

7.9 Appendix I: Excel macro "creating_the_program" explanation

This macro calls each task the program has to fulfil as chosen by the user. Part 1, as shown in Figure 75, gives dimensions to some parameters and runs the first two parts options that the user can chose (see chapter 6.6.4).

The macro starts by giving parameter "Starttime" the value of the actual time which is needed to calculate the runtime of each macro later on. Next it minimizes the excel workbook and disables alerts and screen updating in excel to make sure the user isn't bothered by the process.

After this the macro starts looking for the calculations that it has to do. Cells P3, P5, P7 and P9 coincide with the four processes explained in chapter 6.6.4. If the user checks the first box it will create a "1" in cell P3. If this is true this macro will call the Teddy_export which means it will run it.

```
Sub Creating_the_program()

Dim Starttime As Double
Dim SOFiSTik, Results, Fatige, Minutes_elapsed As String

Starttime = Timer
ActiveWindow.WindowState = xlMinimized

With Application
.DisplayAlerts = False
.ScreenUpdating = False
End With

If ActiveSheet.Range("P3") = 1 Then
Call Teddy_export
Application.Wait (Now + TimeValue("0:00:10"))
Else: End If

If ActiveSheet.Range("P5") = 1 Then
'wait till sofistik is finished
Dim I As Integer
thesentence = (ThisWorkbook.Path & "\SOFiSTik\finished.xlsx")
I = 0
Do Until I = 1
    If Dir(thesentence) <> "" Then
        I = 1
    Else
        I = 0
    End If
Loop

SOFiSTik = Format((Timer - Starttime) / 86400, "hh:mm:ss")

Call Results_gathering

Results = Format((Timer - Starttime) / 86400, "hh:mm:ss")
Else
Results = "-"
End If
```

Figure 75 Creating_the_program macro part 1

If P5 is a one the macro calls the Result_gathering macro but before it does it has an extra verification to make sure that SOFiSTik is finished. The last part of the SOFiSTik program exports an excel file called "finished". To verify that this file exists a "Do Until" loop is used. This loop keeps on running until a certain criterium is met, in this case I=1. In the loop an IF-statement is checked every second which looks if the excel file "finished" exists, if it does it returns a one, if it doesn't it returns a 0 and continues the loop. The moment it returns a one it gives the parameter SOFiSTik the time at that moment and starts the Result_gathering.

The second part, shown in Figure 76, runs the fatigue_calculation and PDF_export if the users has checked those boxes. If it has finished all these processes it sets the screen updating and alert display back on. Then it sets the parameter for Minutes_elapsed.

All the parameters are shown at the end of this program in a message box so the user knows how long each part of the program took to calculate.

```
If ActiveSheet.Range("P7") = 1 Then
Call Fatigue_calculation

Fatigue = Format((Timer - Starttime) / 86400, "hh:mm:ss")
Else
Fatigue = "-"
End If

If ActiveSheet.Range("P9") = 1 Then
Call PDF_Export
Else: End If

If ActiveSheet.Range("P3") = 0 Then
SOFiSTik = "-"
Else: End If

With Application
.DisplayAlerts = True
.ScreenUpdating = True
.StatusBar = False
End With

Minutes_elapsed = Format((Timer - Starttime) / 86400, "hh:mm:ss")

ActiveWindow.WindowState = xlMaximized
MsgBox "Calculation finished in " & Minutes_elapsed & vbNewLine &

End Sub
```

Figure 76 Creating_the_program macro part 2