Computational Methods for Estimating Rail Life

Chase Carlton Holland

Thesis submitted to the faculty of the Virginia Polytechnic Institute and State University in partial fulfillment of the requirements for the degree of

Master of Science
In
Mechanical Engineering

Robert L. West
Mehdi Ahmadian
Norman E. Dowling

February 20th, 2012
Blacksburg, VA

Keywords: Finite element analysis, fatigue, fracture, response surface, railroad track, detail fracture

Computational Methods for Estimating Rail Life

Chase Holland

ABSTRACT

In American rail operations, rails fail due to the combined effects of rail wear due to repetitive wheel contact and the growth of surface and sub-surface cracks and flaws. Rail maintenance includes frequent uncoupled wear and ultrasonic inspections that determine the amount of wear that the rail has undergone and the presence of cracks and flaws. A rail is removed from service when its wear reaches a pre-determined wear limit or a flaw is detected in its cross section. In rail research, the life of a rail is typically estimated using fracture mechanic or fatigue methods and an assumed flaw geometry. Multiple models ranging from complex elastic-plastic finite element models to simplified representations of a beam on an elastic foundation have been developed to predict the life of a rail. The majority of rail failure models do not incorporate rail wear into their analysis, and assume an unworn rail geometry. In order to account for rail wear, certain models adopt simplified rail geometries that uncouple rail wear into top-wear and side-wear.

This thesis presents a rail failure model that describes the combined effects of rail wear and crack growth through the development of a functional relationship between input variables describing the geometry, loading, and material properties of a given rail and output variables describing the life characteristics of the rail. This relationship takes the form of multiple response surfaces estimating the desired output variables. Finite element models incorporating worn rail profiles and an assumed crack geometry corresponding to a detail fracture are combined to determine the state of stress and strain at the assumed flaw. Strain-life fatigue methods and fracture mechanic concepts are used to develop the output variables necessary to describe the life of the rail using the finite element model results. The goals of this research are to predict the remaining fatigue life and estimate the crack-growth rate of the rail based on the minimum number of geometry, loading, and material property independent variables. The outputs developed to describe the rail's remaining life are intended to be used for the decision making for rail removal.

Table of Contents

List of Figures

List of Tables

Nomenclature

A	Crack cross-sectional area
A	Rail cross-sectional area
A_H	Rail head cross-sectional area
a	Semi-major crack dimension
a_c	Critical crack size
a_i	Initial crack size
a_f	Final crack size
b	Semi-minor crack dimension
b	Strain life constant
C	Paris law crack-growth constant
C_o	Walker crack-growth constant
c	Strain life constant
da/dN	Crack-growth rate
E	Material elastic modulus
e	Nominal strain
H'	Ramberg Osgood strain hardening coefficient
h	Overall height of rail
I_{xx}	Rail horizontal area moment of inertia
I_{yy}	Rail vertical area moment of inertia
K	Stress intensity
K_C	Critical stress intensity (fracture toughness)
K_{IC}	Mode I critical stress intensity (fracture toughness)
k_t	Theoretical stress intensity factor
k_ε	Local strain concentration factor
k_σ	Local stress concentration factor
L/V	Lateral to vertical wheel load ratio
m	Paris law crack-growth exponent
M_G	Stress gradient magnification factor
M_S	Crack shape magnification factor
M_1	Finite cross section magnification factor
MGT	Million gross tons
N	Number of load cycles
N	Isoparametric interpolation function
n	Response surface nodal values
N^*	Morrow strain life correction
n'	Ramberg Osgood cyclic hardening exponent
N_f	Fatigue life
N_g	Isoparametric global interpolation matrix
$n(\bar{\bar{\xi}})$	Response surface output variable
p	Ravaee and Hassani material constant
P_{LAT}	Lateral wheel load
P_{MAX}	Maximum vertical wheel load
q	Ravaee and Hassani material constant
R	Cyclic loading stress ratio
R	Track curvature radius
S	Nominal stress
T	Rail temperature
x_o	Reverse bending location for a beam on an elastic foundation

x_T	Response surface transformed independent variable
y	Crack horizontal location dimension in rail head
z^*	Crack vertical location dimension in rail head
α	Thermal expansion coefficient
α	Ramberg Osgood yield offset coefficient
γ	Walker material constant
ε	Local strain
ε_a	Load cycle local strain amplitude
ε_e	Elastic strain
ε_f'	Strain life constants
ε_p	Plastic strain
ε_t	Total strain
Γ	Gamma function
Δh	Head loss rail wear
$\overline{\Delta K}$	Equivalent $R=0$ stress intensity
ΔS	Stress range
ΔT	Rail temperature gradient
Δw	Gauge face loss wear
μ	T-location scale distribution location parameter (mean)
v	T-location scale distribution shape parameter
v	Material Poisson's ratio
ξ	Parametric natural parameter
$\overline{\xi}$	Response surface independent coded variable vector
ξ_x	Response surface independent coded variable
σ	T-location scale distribution scale parameter (standard deviation)
σ	Local stress
σ	Rail bending stress
σ_f'	Strain life constants
σ_m	Load cycle mean local stress
σ_{max}	Load cycle maximum local stress
σ_{min}	Load cycle minimum local stress
σ_R	Rail residual stress
σ_T	Rail thermal stress
σ_{UT}	Material ultimate tensile strength (also denoted S_{UT})
σ_z	Normal stress in z-direction (denoted S_{33} in Abaqus)
σ_o	Material yield strength
σ_1	Max principle stress
σ_2	Mid principle stress
σ_3	Min principle stress
$\overline{\sigma}$	Von Mises equivalent stress
$°C$	Rail curvature in degrees

Chapter 1. Introduction

1.1 Overview and Statement of Need

For many years, American railroads have relied on previous experience and frequent wear and ultrasonic inspections to ensure the safety of their rail network. Rail wear is inspected using a Track Geometry car equipped with lasers capable of "reading" the profile of the worn rail head. Rails are taken out of service when the rail wear exceeds a pre-determined amount or if a subsurface flaw is detected in the rail. It is desired to develop a scientific method to determine when a rail should be taken out of service instead of using wear limits based on previous railroad experience. This is primarily motivated by a desire to reduce operating costs associated with rail inspections and replacement. In addition, current Association of American Railroads (AAR) regulations mandate that rail be immediately replaced if any size flaw is detected [2].

If it can be shown that a detected crack will take a given number of load cycles to grow to a critical size where failure is expected during the next load cycle, the inspection interval can be determined to assure that the rail is inspected again before failure occurs. In this way, a rail can be considered safe even if a crack is present. This approach is considered damage tolerant analysis. While there has been significant work relating to the initiation and propagation of cracks in rails, the majority of the work does not include rail wear into the study. These works are summarized in Chapter 2, and of these works, only one addresses the effects of rail wear [15]. Therefore, a rail failure model that attempts to predict the remaining life of a rail with section properties characterized by the loss of cross-section due to rail wear and the propagation of subsurface cracks is described in the remainder of this thesis. The rail failure model will characterize the structural capability and remaining life of the rail as a function of rail's remaining cross section for worn profiles obtained from a real Track Geometry car given a detailed fracture at the detectable size limit for the flaw detection equipment. The model takes the form of multiple response surfaces that estimate the remaining life and crack-growth rate of the given rail.

1.2 Concept for Solution

The relationship between rail wear and the life of the rail or the growth rate of a crack growing in the cross-section of the rail is currently not well defined. In order to address the effects of rail wear, a functional relationship between rail geometry and the life and crack-growth rate of the rail is developed in the form of multiple response surfaces. A response surface describes output variables of interest as functions of multiple input variables. In order to develop the functional relationships desired, the state of stress and strain for the rail is developed using the Finite Element Method (FEM). Once the state of stress and strain in the rail is resolved, the life of the rail can be predicted using fatigue and fracture mechanics

methods. As in fracture mechanics, a crack geometry representing a detailed fracture is assumed for the rail, and finite element models are used to determine the stress and strain at the crack edge.

The remaining life of a given worn rail is estimated with two primary output variables. These two variables are the rail's remaining fatigue life and crack-growth rate. The rail's remaining fatigue life is represented as a single output variable, and the crack-growth rate is represented using an exponential equation with two constants. Three response surfaces are generated to represent these variables. The first response surface's output is the rail's remaining fatigue life. The first crack-growth-rate constant is the output variable of the second response surface, and the second crack-growth-rate constant is the output variable of the third response surface. These three output variables will provide the information necessary to make an informed decision on when to remove a rail from service. The fatigue life output variable will provide an estimate of the remaining days until the rail ultimately fails. The crack-growth-rate equation and constants provide information on how quickly the crack is growing as a function of load cycles in wheel load reversals. The response surfaces' output estimates the rate that the rail's structural capability is decreasing, and can be used to determine a new inspection interval for the rail. The new inspection interval would be determined so that the rail is inspected again before the estimated time of failure. The estimated flaw size at the time of inspection can be calculated from the crack-growth-rate equation. This thesis explains the development of the three response surfaces used to estimate these variables.

While each response surface has a different output variable, all three response surfaces have the same independent or input variables that represent the geometry, loading, and material properties of the rail. Four independent variables are used to represent the geometry of the worn rail. These variables are the cross-sectional area, the overall height, the horizontal area moment of inertia, and the vertical area moment of inertia of the worn rail. The loading of the rail is represented using three independent variables, the maximum vertical wheel load, the curvature of the rail, and the seasonal change in temperature that the rail undergoes during 24 hours. The material of the rail is represented with a single independent variable, the material's ultimate tensile strength.

The response surfaces reference life estimates generated using finite element models and a fatigue post-processor. The finite element models are built using multiple worn rail profiles and the loads applied to a typical rail. The worn rail profiles are representative of typical wear patterns and are provided by data collected by a Track Geometry car. The profiles are divided into top-wear profiles and side-wear profiles, and have wear amounts ranging from severely worn to an unworn geometry. The Track Geometry car uses a laser to scan the rail head profile and plot multiple data points representing the rail head. In order to recreate the profiles in the finite element model, the data is fit using a curve-fitting algorithm. The four

2

independent geometry variables are calculated using numerical techniques to a desired precision from the fitted profile. Once the independent geometry variables are calculated, the fitted profile is imported into the finite element model and used to create a full rail. Multiple load cases including vertical and lateral wheel loads and a thermal load are applied in the finite element model in order to provide the stress results necessary to create a daily load history in the fatigue post-processor. The independent load variables are generated from track data and historical weather data. The independent load variables are used to create the load cases in the finite element model. In addition, multiple sub-surface crack sizes are modeled in each worn rail model to develop the information necessary to estimate the rail's crack-growth rate. These sub-surface cracks represent a common flaw called a detail fracture. Multiple elastic analyses are completed to generate the samples necessary to build the three response surfaces.

After the elastic analyses are completed, the results are imported into the fatigue post-processor in order to estimate the remaining fatigue life of each rail. The fatigue post-processor uses the elastic stress results, Ramberg-Osgood stress-strain law, and Neuber's rule to approximate the plastic stress and strain at each node in the finite element model. A load history is developed using the stress results from the load cases completed in the elastic analysis. The fatigue life in days is calculated for each element in the finite element model that is included in the analysis group using a strain-life algorithm. The life results are post-processed a second time to calculate the estimated crack-growth rate for the rail. This is done using the fatigue life results for the multiple crack sizes modeled in each worn profile. In order to collect the necessary samples required for the generation of the three response surfaces, all the independent variables must be varied during this overall process. The independent geometry variables are varied in the finite element models by building models featuring the geometry of different wear patterns and amounts. The curvature independent load variable is also varied in the finite element model by changing the magnitude of the lateral wheel load. The other independent load variables and the independent material variable are varied in the fatigue post-processor by using multiple materials and changing the magnitude of the loads in the load history definition.

The three response surfaces are generated using the independent variables varied in the finite element models and fatigue post-processor and their respective output variables. Once the response surfaces are created, the results can be examined. A flow diagram of the data and processes required to generate the response surfaces and the process for using the response surfaces to evaluate a worn rail is shown in Figure 1.1.
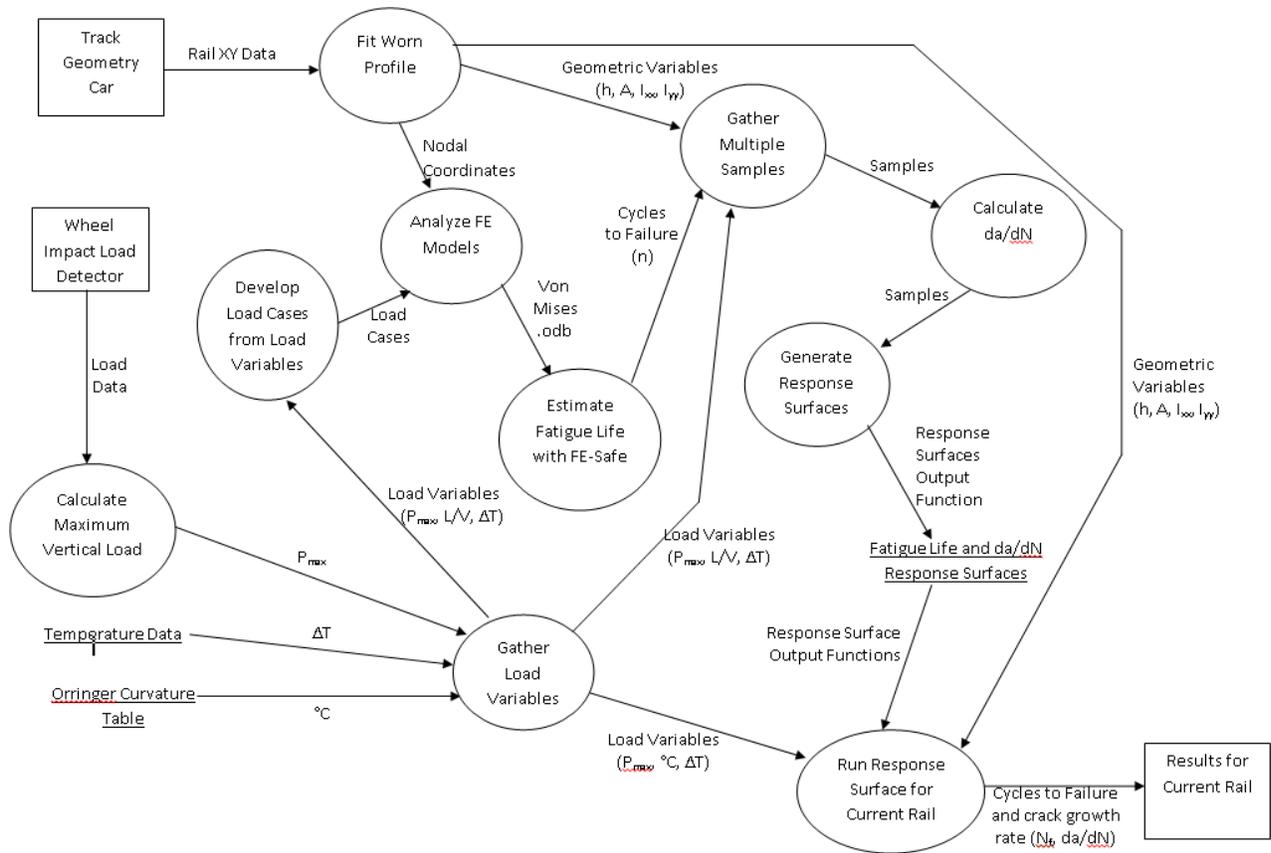
Figure 1.1 Data flow diagram for the generation and use of the response surfaces.

Each process depicted in this data flow diagram is explained in the remaining sections of this thesis. The end result of the research completed is a functional series of response surfaces that estimate the remaining life and crack-growth rate of a worn rail given its geometric properties, loading, and material properties.

1.3 Goals and Objectives

The purpose of this research is to develop a model capable of predicting rail failure of worn rails due to fracture. This is completed by generating the three response surfaces introduced in Section 1.2 that predict the life and crack-growth rate of the rail. In order to develop the response surfaces, the geometry must be accurately represented. Instead of modeling every worn profile of interest, the geometry of the worn rail is characterized using section properties including the four geometric independent variables. The next objective is to represent the loading of the rail which is stochastic in nature. This is accomplished using the three independent load variables that represent the probability distributions of each load using deterministic values. The load cases applied in the finite element models are developed from the independent load variables. The third objective is to determine the stress and strain state in the rail. This is accomplished using an assumed crack geometry and finite element models. The fourth objective is to characterize the material properties of the rails of interest. Because the properties

4

necessary for fatigue and fracture calculations are not readily available, these properties are estimated using approximations and the material properties that are known. The last objective is to estimate the response surface output variables. This is achieved by combining strain-life fatigue algorithms and fracture mechanics post-processing of the fatigue results. After the output variables are calculated, the response surfaces can be formulated.

The primary goal of this research is to show that the concept for the solution introduced in Section 1.2 and explained in the remainder of this thesis is a valid approach. A secondary objective of the research is to develop an understanding of the propagation of detail fractures in worn rails, and the loading associated with the growth of this flaw. In order to develop accurate estimates of the remaining life of a worn rail subjected to a detail fracture of a certain size, the modeling technique must correctly capture the true effects of the crack. This process will be used to evaluate the strengths and areas of needed improvement for the current modeling technique. The response surfaces will undergo an evaluation stage in order to determine their accuracy.

1.4 Scope of Thesis

This thesis provides a first attempt for a failure prediction model for rails undergoing wear due to repeated contact with train wheels and the initiation and propagation of subsurface flaws. Countless factors and variables affect the life of rails, and this thesis in no way attempts to characterize all of them. The list of independent variables included in this analysis is developed from multiple discussions with representatives of a major American railroad and examination of research previously completed. It is expected that the scope of this work could be changed after continued analysis and examination of the results produced. In keeping with this approach, a list of assumptions and limitations are placed on the analysis completed. A brief list of some of these assumptions includes:

- Only rail failure due to the propagation of a subsurface flaw termed a detail fracture is considered.
- Only 136RE rails are included in the generation of the first set of response surfaces.
- Only the geometry and independent load variables defined in Section 1.2 are included in the generation of the first set of response surfaces. All other variables are treated as constants model parameters in the analysis or are ignored.
- The previous load history of the rail is represented using a flaw modeled in the rail head equivalent to the minimum detectable flaw size of current detection equipment.

- Because no previous information is known about the relationship between the independent and output variables, the relationship will initially be considered linear to decrease the number of samples needed.

- The minimum number of analyses necessary for the generation of the response surfaces will be completed.

- Post-processing of the finite element elastic stress results using a strain-life algorithm will be used to evaluate the fatigue life of the rail due to the limited material data available for rails in service.

- Linear superposition of loads and scaling of the elastic stress results will be used to generate the 24 hour load history and to vary the independent load variables in the fatigue post-processor.

These assumptions attempt to keep the scope of the thesis reasonable for an initial study, and have been developed from extended research on related resources and discussions with railroad representatives. Other assumptions and limitations of the current analysis method will be explained in the following chapters.

1.5 Thesis Outline

This thesis begins by providing information regarding the current rail maintenance practices typically used by railroads in America. The literature review continues by examining some of the technical research that has been completed regarding rail failure due to fracture and fatigue. In these works, multiple failure models are reviewed using fracture mechanics and fatigue methods. Chapter 3 begins the discussion of the rail failure model introduced in Section 1.2. Chapter 3 addresses the fitting of the worn rail's cross-section and the generation of the four independent geometry variables. Chapter 4 documents the development of the three independent load variables from track and weather data. The development of the finite element model used to generate the stress and strain results necessary for the fatigue post-processing and the data required for the development of the response surfaces is explained in Chapter 5. Chapter 6 and 7 explain the fatigue post-processing of the finite element stress results and the development of the crack-growth rate through additional post-processing of the fatigue results. The construction of the multiple response surfaces necessary to estimate the remaining fatigue life and crack-growth rate of the rail is explained in Chapter 8. Chapter 9 examines the initial results used to create the first generation of the response surfaces and provides discussion relating to the strengths and weaknesses of the current process and representation in the finite element model. Suggested improvements for the next generation of the response surfaces are discussed. Finally, Chapter 10 concludes the thesis and provides an overview of the work completed.

Chapter 2. Literature Review

This thesis presents a method for estimating the remaining life of a section of railroad track using procedures developed from strain-based fatigue theory and linear elastic fracture mechanics. Current railroad practice for track maintenance includes the implementation of rail wear limits and ultrasonic testing for defects including cracks and inclusions. The current wear limits and inspection intervals employed by American railroads are determined using statistical analysis of historical data for track failures. Multiple technical papers address the failure of railroad track due to the initiation and propagation of cracks of various shapes and sizes. While the majority of the papers focus primarily on experimental testing of material specimens and actual rails, a limited number of papers attempt to define simplified analytical models capable of predicting rail failure due to fracture or fatigue. This section will provide a brief overview of the current maintenance practices typically implemented by American railroad companies and those mandated by the AAR. The various technical papers that address the fatigue failure of railroad track and suggested models for predicting this failure will subsequently be discussed.

2.1 Railroad Track Maintenance Practices

Track maintenance is completed through the periodic testing of the rail's cross-section for embedded flaws and cracks as well as periodic inspections of the rail's wear level. The testing of the rail's cross-section for flaws is completed using the Ultrasonic Test car [2]. The worn profile of the rail is examined by a Track Geometry car that uses laser technology to "read" the profile of the worn rail head [2]. By comparing the worn profile to that of an unworn rail, the amount of wear in the vertical and lateral (gauge-face) directions can be determined. The testing for embedded flaws and wear inspections are completed on different inspection intervals that vary substantially for the various types of tracks used in America.

Wear Limits

The wear limits utilized by American railroad companies vary from railroad to railroad, but typically include separate limits for rail head-height and gauge-face loss. These limits are also called top-wear and side-wear limits, and are independent of each other. The values of these limits depend on the weight or type of rail and the amount of traffic that the rail endures. For example, top-wear and side-wear limits for 136RE rail on main track are 10/16^{ths} and 8/16^{ths} of an inch (1.5875 and 1.270 cm) respectively for one major American railroad [1]. The rail type 136RE refers to a rail that weighs approximately 136 lbs. per yard. The values of the implemented wear limits are normally determined from historic track data and past experience for the given railroad. The wear limits implemented by the railroad are enforced using a Track Geometry car capable of scanning the worn rail's profile and superimposing this profile on top of

an unworn rail profile for the given rail weight. The method for measuring the amount of side and top-wear differ from railroad to railroad. Each railroad uses a different rail wear gauge that measures the wear level utilizing a point on the unworn rail geometry as the reference. A typical wear gauge that is currently in use for 132RE rail is shown in Figure 2.1.

Figure 2.1 Typical rail wear measurements.

This top-wear measurement references the top center point of the unworn rail, and the side-wear measurement is referenced from the lower gauge corner of the unworn rail which corresponds to a point on the side of the rail that is 1.4375 inches (3.6513 cm) below the original top of the rail [1]. Therefore, the measurements are always referencing the unworn geometry, and the reference points do not change. American operated track sections are inspected at different intervals depending again on the weight and purpose of the rail. The traffic for a given rail is defined as the summation of static axle loads in million gross tons (MGT) that have accumulated in a given year. Typical rail wear inspection intervals are given in Table 2.1.

Table 2.1 Typical rail wear inspection intervals.

	Heavy (Main)Traffic Line	Medium Traffic Line	Low Traffic Line
Tonnage	15+ MGT	5-15 MGT	0-5 MGT
Yearly Inspections	3	2	1

The inspection intervals provided in Table 2.1 are developed from extensive experience with rail wear rates spanning decades [2].

Ultrasonic Testing
As previously stated, the Ultrasonic Test car is used to determine if cracks and flaws are present in the rail's cross-section. The detection equipment operates by sending ultrasonic pulses into the rail and analyzes the reflected response [3]. The probability of locating a flaw increases with the crack size, represented as the fractured percentage of the rail's head area (%HA). The reliability of the testing procedures for multiple flaw types and sizes are given by the reference AREMA handbook [4]. According the AAR regulations, the effected section of rail must be immediately repaired or replaced if a flaw is discovered in the rail [4]. Therefore, the speed and efficiency of the testing procedure depends significantly on the number of flaws detected and the amount of time necessary for rail section repair or replacement. Current practice utilizes a flaw size of 10% HA as the minimum detectable size for the Ultrasonic Test car [3].

Ultrasonic Testing Frequency
The inspection intervals for the testing of the rail's cross-section for flaws with the Ultrasonic Test car are determined from risk-based statistical analysis of historical rail failure data collected by the given railroad company. By defining the risk associated with rail failure as "the annual rate of service defect occurrences", the accepted level of risk can be developed from historical defect occurrences and inspection intervals [5]. A common level of accepted risk for American operations is 0.066 service defects per mile per year, which is a function of the age of the rail, the type and level of traffic, and the reliability of the ultrasonic inspection equipment [5].

2.2 Rail Testing
While there has been significant amounts of testing of detail fractures in actual rail specimens, Orringer, Tang, Gordon, Jeong, Morris, and Permlamn provide the largest study of external effects on the growth of detail fractures. Their 1988 paper presents the findings of these studies, and has become a common resource for the rail engineer. Orringer et al. encompasses ten years of rail fracture research including

field experiments and crack growth in actual rails, laboratory tests, and the development of analytical models. The purpose of this research is to understand and predict the propagation of detail fractures in rails. The research includes load, material, and geometry sensitivity studies; applied and residual stress mapping; the development of initial crack-growth models; and material testing [7].

The field of fracture mechanics focuses on the study of propagating cracks in engineering materials. Because the failure of the rail typically occurs through the brittle fracture of a rail cross-section containing an interior flaw, Orringer et al. use fracture mechanics to predict and analyze rail failures. The majority of the analytical models developed to predict rail failure operate by assuming a crack geometry and using Linear Elastic Fracture Mechanics (LEFM) to describe the behavior and propagation of the assumed crack. LEFM is valid for the brittle fracture of materials where Small Scale Yielding (SSY) is assumed. Small Scale Yielding assumes that the non-linear effects and plastic deformation near the tip of the crack are contained in a relatively small plastic region when compared to the region dominated by the crack singularity.

The method of LEFM introduces a quantity called the stress intensity factor to describe the stress state at a crack edge. The stress intensity factor is a function of the component geometry, the crack size, and the nominal stress applied to the component assuming linear-elastic material behavior. The stress intensity factor is given by the general equation:

$$K = FS\sqrt{\pi a} \qquad\qquad (2.1)$$

where K is the stress intensity factor, F is a constant defined for the given geometry and loading, S is the nominal applied stress, and a is the relative crack length [6]. Typically, the non-dimensional constant F is defined using empirical relationships for an assumed geometry developed from experimental testing. The fracture toughness (K_c) of a material is defined to describe a material's resistance to cracking and is a function of temperature, component thickness, and mode of failure. Three failure modes describe the method of crack aggravation and include opening, in-plane, and out-of-plane shear. Propagation of the crack for a given failure mode is expected due to brittle fracture when the calculated stress intensity factor is equal to fracture toughness for that failure mode.

The crack-growth tests in the work of Orringer et al. include 10 sections of rail, all of which contain detail fractures of various sizes. The damaged rails are supplied by various railroads, and are installed in the Facility for Accelerated Service Testing (FAST) track at Transportation Technology Center Inc. (TTCI). TTCI is a research facility funded by the AAR and other American railroads. After installation, the test rails are repeatedly loaded using the FAST train, which averages 54.3 MGTs in a 6 month period, until

failure. Thirty additional rails with detail fractures of various sizes are statically tested to determine the residual strength of the damaged rail as a function of crack size. In order to represent the detail fractures in the test rails, a three-axis ellipse is used. An empirical relationship relating the detail fracture size to its location in the rail head is developed from the test data. In addition, empirical magnification factors used to represent the stress state in Equation 2.1 are developed from the test data to describe the effects of the shape of the crack, the transition from an embedded crack to an edge crack, the stress gradient, and the finite geometry of the rail. The developed stress intensity can be approximated by the equation:

$$K = 2M_S M_1 [\sigma_R + \sigma_T + M_G \sigma] \sqrt{r/\pi} \tag{2.2}$$

where M_S is the shape magnification factor, M_1 is the finite cross-section magnification factor, M_G is the stress gradient magnification factor, σ_R is the residual stress, σ_T is the thermal stress, σ is the bending stress, and r is the crack semi-major dimension [7]. Similar versions of this equation are used in many of the papers addressed in this section, and have been adopted directly by Jeong, Tang, and Orringer in their 1998 paper [15].

The stresses used in Equation 2.2 are developed through a combination of analytical and experimental methods. The bending stress is calculated assuming that the rail is a beam on an elastic foundation with stiffness's in the lateral, vertical, and torsional directions. The wheel load is applied at a point on the rail head at a certain distance, denoted e, from the rail center, and has lateral and vertical magnitudes. The magnitude of the bending stress due to the application of the wheel load varies along the length of the rail due to the effect of multiple wheels that make up a train. The variation in bending stress along the rail is shown in Figure 2.2.
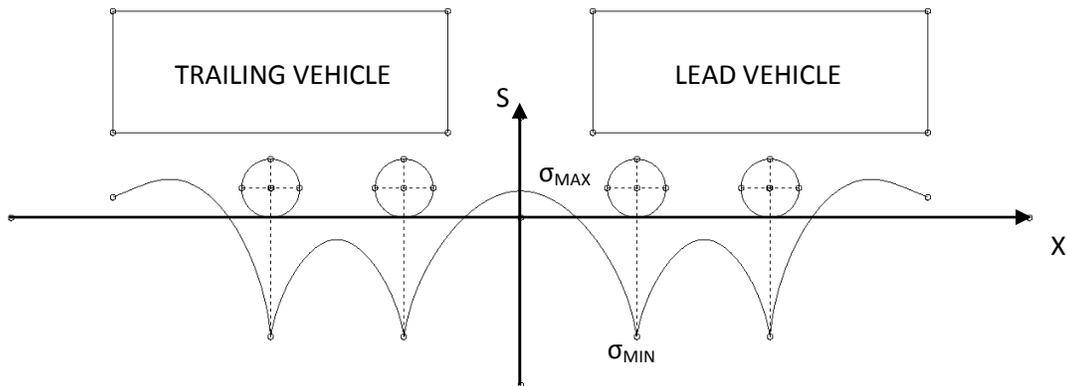
Figure 2.2 Bending stress variation along rail length [7].

At the application of each wheel load, there is a compressive stress in the rail head. These points are shown as negative peaks in Figure 2.2. At the point between the front and rear cars, there is a tensile bending stress in the rail head. This corresponds to the point labeled σ_{MAX} in Figure 2.2. The bending load cycle is defined using the maximum and minimum stresses shown in Figure 2.2. The dynamic effects of the wheel loads are represented using a bi-modal four-parameter Gaussian model that is applied to the FAST data and other available rail data for heavy and light car loads. The collected data suggests that the variation in dynamic load increases as the car weight decreases, and that vehicle speed does not influence the variation of the data. The fitted distributions can be used to determine a dynamic load factor for the vertical component of the wheel load assuming a static wheel load. The lateral component of the wheel load is developed using the vertical magnitude and the lateral-to-vertical load ratio. The lateral-to-vertical load ratios are developed from track data for 100-ton freight cars as a function of curvature in degrees. The thermal stress in the rail is developed from historical temperature data, and the residual stress in the rail is considered due to plastic deformation at the rail's running surface assuming rolling contact as the dominant factor. After the wheel load is removed from the rail head, compressive stresses remain underneath the rail running surface due to the plastic deformation of the contact patch. A simple empirical model for estimating the residual stress as a function of crack size is adopted using measured residual stresses in actual rails. This model is given in Equation 2.3:

$$\sigma_R = 30 - 2.125 \left(\frac{A}{A_H} \times 100 \right) \ for \ 0\%HA \ \leq \left(\frac{A}{A_H} \times 100 \right) \leq 10\%HA$$

$$\sigma_R = 10 - 0.125 \left(\frac{A}{A_H} \times 100 \right) \ for \ \left(\frac{A}{A_H} \times 100 \right) > 10\%HA \tag{2.3}$$

where A is the crack area and A_H is the cross-sectional area of the rail head [7].

The sensitivity study includes vehicle, track, and other factors effects. The vehicular factors include 1) the vehicle type, 2) average axle load, and 3) the dynamic effects of car motions and wheel defects. The track factors include 1) curvature, 2) vertical foundation stiffness, 3) rail section, and 4) rail residual stress. The other factors include 1) thermal tension in continuous welded rail (CWR) due to temperature differential, 2) wheel contact location on the rail running surface, and 3) the detail fracture location in the rail head. From the study, the three most influential factors on the crack-growth life are the temperature differential, the residual stress in the rail head, and the curvature of the rail. The rail section, wheel contact location, and track modulus showed lower significance, and the axle load, vehicle dynamic effects, and flaw location had the lowest effect on the crack-growth life [7]. The results of the sensitivity study were examined during the development of the independent variables used in this thesis.

The representation of the stress intensity and magnification factors given in Equation 2.2, the rail modeling techniques, and the crack-growth conclusions developed in this paper have been adopted by numerous other entities in their exploration of detail fracture growth behavior in rails. This work serves as the foundation for more refined crack analysis models that have been and are currently being developed. A number of these analysis models are explored in the following section.

2.3 Rail Failure Models

Multiple technical papers address the problem of rail failure due to brittle fracture. While there are multiple types of surface and subsurface flaws and cracks, the majority of the papers address a specific type of subsurface crack called the detail fracture. Rice and Rungta state that over a 9 year period from 1974 to 1982, ultrasonic detector cars have found over 938,000 defects in rails on American rail lines. Of these defects, the majority (63 percent) are joint or web and base related. The remaining defects are head related and include surface and subsurface shelling, horizontal and vertical splits, engine burns, transverse defects, and detail fractures. Transverse defects and detail fractures make up about 39 percent of the overall rail-head related flaws [8]. While this information is dated, the number of detail fractures detected with current equipment has only increased. Transverse defects and detail fractures are subsurface flaws that grow through the cross-section of the rail. Detail fractures form in the gauge corner of the rail and are usually associated with side-wear [9]. Detail fractures typically initiate at subsurface longitudinal shelling locations that turn transverse and propagate [9].

These flaws are particularly dangerous because they are hard to detect at smaller sizes and can grow to a critical size quickly. Multiple technical papers provide information on the testing of rail material to achieve the material constants necessary for LEFM analysis. In addition, a few papers present analytical rail models that utilize LEFM methods to represent the propagation of the detail fracture in an effort to predict rail fracture. Each model uses different representations of the crack, geometry, and loading of the rail. These models are explained below.

Rail Loading

The representation of a rail's loading varies from paper to paper, but is usually represented as a combination of vehicle and environmental loading. Andreikiv, Shchur, and Darchuk [10] present a representation of the vehicular wheel load in their 1987 paper as a bending moment distribution by representing the rail as an infinite beam on an elastic foundation. Using the assumption that forces acting on a rail are not influential when greater than 4 meters away from the point of interest, the bending moment distribution for a four-axle car truck is developed using the last two axles and the first two axles of two connected rail cars similar to that given by Orringer et al. in Figure 2.2. This load cycle defined

between the maximum and minimum bending moments from the distribution is used to develop the characteristic stress intensity range for the rail. This paper only addresses the loading of the rail due to the application of bending loads by the train wheels.

Rice and Rugta [8] explore the combined effects of vehicle loads applied by the wheel and residual stresses caused by plastic deformation of the rail at the running surface in their 1987 paper. The wheel induced stresses are developed in a 3-D finite element model by the application of a 30 kips (133.4 kN) concentrated load at a location on the gauge side of a rail, 0.79 inches (20 mm) off center. The gauge side of the rail is side of the rail closest to the center of the track. The residual stresses are developed by measuring the residual stresses in actual rails from tangent track and making the assumption that the measured stresses are representative for all rails laid in tangent track. The mapping of the residual stresses in the rail head reveals areas of high compressive stress at the gauge face, gauge corner, and running surface of the rail. These high compressive stresses force the formation of significant tensile stresses in the interior of the rail that are believed to be significant in the formation of transverse defects. The wheel load induced stresses and the residual stresses are superimposed to generate the overall state of stress in the rail. Using the combined state of stress, an algorithm is developed using an interpolation/extrapolation technique to predict the state of stress in terms of the three principal stresses at any point in the rail head. The combined loading is simplified by only considering the axial component of each stress. This practice is consistent with the assumption that the crack initiates and propagates as a Mode I aggravated failure.

Pelloux, Tang, and Jablonski [11] introduce an experimental model that simulates the growth of a detail fracture using the experimental propagation of a crack in a compact test specimen in their 1990 paper. The compact specimen is cut and machined from an actual rail head. The simulation is completed by matching the stress intensity factor of the experimental compact specimen to that of a rail with an approximated detail fracture. The loading assumed for the rail includes the combined effects of wheel induced bending stresses, residual stresses in the rail head, and longitudinal stresses induced by thermal loads. Using an empirical formula to calculate the residual stress and mechanics of materials approximations to calculate the bending and thermal stresses, the total rail stress is represented as the summation of the axial components of each stress. In order to represent the propagation of the detail fracture using the fracture specimen, the change in the stress intensity at the detail fracture over the change in detail fracture major dimension is set equal to the change in compact specimen stress intensity over the change in crack length. This is given by the equation:

$$\frac{\partial K_R}{\partial r} = \frac{\partial K_{CS}}{\partial a} \qquad (2.4)$$

where K_R is the rail stress intensity factor, K_{CS} is the compact specimen stress intensity factor, r is the detail fracture major dimension, and a is the crack length in the compact specimen. Equation 2.4 is used to determine the compact specimen load history that will generate the rail stress intensity factor.

Tehrani and Saket [12] identify the rail's loading using primary and secondary loads in their 2009 paper. The primary loads include the effects of the wheel load on the rail including the bending, axial, and Hertzian contact stresses developed from rolling contact. The secondary loading includes thermal axial stresses and residual stresses in the rail. The thermal axial stresses are developed through the temperature cycling of the rail, and the residual stresses considered are products of the rail manufacturing processes. A rail failure model is developed using Modified Goodman Stress-life fatigue algorithm and the rail stress state. The rail stress state is determined using a three-dimensional elastic-plastic finite element model of the rail and wheel. While the secondary loading of the rail is discussed in the paper, the finite element model used to determine the rail stresses only explores the stresses developed by a static wheel load. The wheel load is applied as a vertical concentrated force to a reference point at the center of the wheel. The bending, axial, and contact stresses in the rail are developed through frictional contact between the wheel and the rail. The magnitude of the wheel load represents the maximum design load and is equal to 32.9 kips (146.2 kN). The load is then cycled between the maximum design load and a zero state of stress for the fatigue calculations.

Plu, Condeux, Boulanger, and Heyder [13] identify that the true stresses seen in the rail head cannot be determined without considering the interaction between the track and rail vehicles as a system in their 2009 paper. The dynamics of the interaction vary substantially for different vehicle types, vehicle speeds, and track conditions. One example of this variation in loading is the contact points between the vehicle's wheel and the rail. For a given track geometry and rail vehicle, the location of contact between the wheel and the rail will change significantly for different sections of the train due to in-train dynamic loads. This variation is increased further by altering the vehicle's speed, the vehicle's weight, the curvature of the rail, or the rail's foundation stiffness. Therefore, the stress response in the rail head utilized in their crack-growth model is developed using multi-body software paired with three-dimensional finite element models of the track geometry. Two load cases are used to validate the crack-growth model. These load cases correspond to high-speed line with tangent track and freight line with a 4986.9 ft. (1520 m) curve. The load history is created using fully-reversed cyclic loading.

The approach for representing external loads developed in this thesis draws on the ideas explained above. The specific ideas adopted include applying the wheel loads through contact between a deformable wheel and rail in a finite element model. Therefore, the contact stresses and bending stresses are applied

simultaneously, and the geometry of the rail is represented. The wheel load cycles between a maximum and minimum value that can be determined from a distribution similar to that given in Figure 2.2 by Orringer et al. The external loads can be represented by superimposing the cyclic wheel loads and thermal loads. However, the load magnitudes and applications are modified from those presented here to represent the current track conditions as accurately as possible.

Crack Representation

In order to analytically model the propagation of a detail fracture or another interior flaw, an initial representation of the crack geometry must first be assumed. The majority of the crack-growth models assume an elliptical flaw present in the cross-section of the rail. The exact flaw geometry and location vary significantly from paper to paper. For example, Andreikiv, Shchur, and Darchuk [10] assume a biaxial elliptical flaw geometry represented by the equation:

$$\left(\frac{x}{a}\right)^4 + \left(\frac{y}{b}\right)^4 = 1 \qquad\qquad (2.5)$$

where *a* and *b* are the semi-major and semi-minor axis dimensions in the *x* and *y* directions. The crack is located at the center of the rail head at a certain distance below the rail's running surface. This crack geometry is assumed because approximate solutions are available for rails with such cracks loaded in bending. Ravaee and Hassani [14] adopt the same crack representation in their 2007 paper. The crack's vertical location is measured from the rail's running surface to the center of the ellipse and is labeled Z^*. It is argued that this crack shape and location matches detail fractures in actual rail.

Rice and Rungta [8] develop a more detailed representation of the crack geometry. They utilize a biaxial elliptical flaw that can change aspect ratio and tilt in the rail's cross-sectional plane. Assuming an initial crack location in the upper gauge corner of the rail head where the measured tensile residual stresses are at a maximum, the crack growth is approximated incrementally. The incremental crack expansion process involves defining new approximated ellipses after small amounts of growth have occurred around the original ellipse. The growth at different points around the original ellipse is estimated using local stress intensity values. Using this process, the crack is free to change aspect ratios and tilt as it grows.

Jablonski, Tang, and Pelloux [11] simply represent the detail fracture as a penny shaped crack in an infinite body subjected to the axial stresses present at a typical location for the center of a detail fracture in a rail. As mentioned in the previous section, the axial stresses in the rail are a combination of wheel induced bending, residual, and thermal stresses. It is assumed that the residual stresses stay constant as the crack grows.

Because Tehrani and Saket [12] use Modified Goodman Stress-life failure criteria, no crack is represented in the rail model. Damage is defined as the ratio of applied load cycles over the rail fatigue life. Therefore, the crack initiates at a damage value of 1. The fatigue life is calculated using the Modified Goodman fracture line and the load cycle defined in the previous section. The location of crack initiation is determined using the maximum von Mises stress location from the elastic-plastic finite element model's results. The maximum von Mises stress, and therefore the crack initiation location, is subsurface at the point of contact between the rail head and wheel.

Plu, Bondeux, Boulanger, and Heyder [13] represent multiple surface and subsurface cracks of different sizes in their analytical model. Finite element models are used to predict the stress intensity range for each crack geometry. The crack locations of interest include the rail head surface at the gauge corner, the rail head surface slightly offset from the rail center plane, and two subsurface locations at the same lateral locations as the two surface cracks. By varying these crack sizes the crack-growth-rate curve can be approximated for each crack type.

In each resource, a different representation of the detail fracture is utilized. In some situations, simplified models are used because analytical and empirical solutions for similar problems are readily available. In the resources presented here, the biaxial elliptical crack is the most commonly used representation of the detail fracture. When developed in a finite element model, the stress intensity at the crack edge can be predicted for the complicated crack and rail geometry. This approach has been adopted in this thesis. While there is not an agreement on the exact location of the detail fracture, the majority of the resources suggest the upper gauge corner of the rail head as the most probable location.

Life Estimations and Results

The majority of the models introduced in the Rail Loading and Crack Representation sections of this chapter utilize a fracture mechanics approach to estimate the fatigue failure of the rail. Each of these models makes the assumption that Mode I is the dominant fracture mode in the rail. This assumption simplifies the analysis significantly, but is supported by the high axial stresses present in the rail under the applied wheel and thermal loads for each model. Using this assumption, fracture mechanics can be used with the idealized geometry and load representations to estimate the fatigue life of the rail for each model.

Using the constant amplitude load cycle defined previously, Andreikiv, Shchur, and Darchuk [10] develop a crack-growth-rate equation using the stress intensity equation for their representation of the detail fracture. For the elliptical flaw centered laterally in the rail head under bending loads, the maximum stress intensity is located at the top of the ellipse. By inverting and integrating the equation with respect to the crack size, the number of cycles for the crack to grow from an initial detectable size to

critical is estimated. For the time period in which this paper was written, the detectable crack size corresponds to a flaw approximately 20% of the overall rail head area (20% HA).

The crack-growth rate is developed incrementally using the local approximated stress intensities at twelve equally spaced locations around the crack edge in Rice and Rungta's [8] analytical model. The crack-growth rate at each point is calculated. After incremental growth, the distorted ellipse is used to define a new ellipse that best represents the distorted geometry. Using an initial flaw corresponding to 2% HA, several preliminary analyses were completed. The crack-growth rates at the points around the crack edge calculated in the analyses remained relatively constant during the incremental increases in crack size. According to Rice and Rungta, this agrees well with cracks growing in actual rail. However, the majority of analytical crack-growth models predict increasing growth rates with increasing crack sizes. The model also predicted rail failure at 20% HA crack sizes. While actual data include failures at smaller crack sizes, failure typically occurs at crack sizes reaching 50% HA [2].

The compact tension specimens used in Jablonski, Tang, and Pelloux's [11] crack-growth simulation are pre-cracked to from a 0.669 inch (17.78 mm) initial crack prior to testing. During testing, five load spectrums representing stress intensities in rails corresponding to general freight, general freight on rail with a 5 degree curve, Transportation Technology Center Inc. FAST train, coal trains with 100 ton loaded cars, and coal trains with 125 ton loaded cars are applied to the test specimens, and the crack growth measured after each applied load spectrum. The crack-growth rate is calculated from experimental crack size versus applied load spectrum plot where each spectrum represents the loads applied for an entire train. Because the load spectrums used in the tests are different time lengths, the calculated crack-growth rates are rescaled in terms of MGT. After rescaling, the crack-growth rates of the 5 experimental rails can be compared on the same scale. When compared, the results from actual FAST trains and the crack-growth results from the load spectrum representing the FAST train agree well at lower crack sizes. Once the crack size reaches 50% HA the experimental and actual growth rates diverge.

Before completing their crack-growth analysis, Ravaee and Hassani [14] determine the plane strain fracture toughness for the rail steel of interest using compact specimen fracture tests. The critical crack size is calculated using the stress intensity given by the equation and the experimental fracture toughness:

$$K_{Ic} = \frac{2}{\pi} M_S M_1 \sigma \sqrt{\pi a_c} \tag{2.6}$$

where K_{Ic} is the fracture toughness of the rail material, M_S is an empirical factor to account to the crack's elliptical shape, M_1 is an empirical factor to account for the finite dimensions of the rail head, σ is the longitudinal stress in the rail head due to the superposition of bending, thermal and residual stresses, and

a_c is the crack semi-major dimension. The number of cycles to grow a crack from an initial size to the critical size calculated using Equation 2.6 is calculated using the form of Paris's law given in the equation:

$$\frac{da}{dN} = C\frac{\Delta K^p}{(1-R)^q}$$
(2.7)

where C, p, and q are material constants developed previously. By inverting and integrating Equation 2.7 with respect to a, the equation for the number of cycles is determined as:

$$N = \frac{1}{C}\int_{a_i}^{a_f}\frac{(1-R)^q}{\left(G(a)\Delta\sigma\sqrt{\pi a}\right)^p}$$
(2.8)

where $G=2M_sM_1/\pi$. Therefore, the remaining life of the rail is very dependent on the minimum crack size that is detectable.

Instead of calculating a crack-growth rate for an assumed flaw, Tehrani and Saket [12] quantify the damage under loading accumulated by the rail using the von Mises stress developed in their elastic-plastic finite element model and Modified Goodman fatigue criteria. The damage is quantified as the ratio of applied load cycles to the Goodman fatigue life. Using this criterion, the effects of various factors on the accumulation of damage can be studied. The two factors of interest in their paper are the magnitude of the vertical wheel load and the hardness of the rail steel. As expected, the accumulated damage increases both with increased vertical load and increased material hardness, and both relationships have steep exponential increasing trends. It is hypothesized that the increase in material hardness decreases the contact area between the wheel and the rail and therefore increases contact stresses in the rail head.

Plu, Bondeux, Boulanger, and Heyder [13] also calculate the crack-growth rate for the given crack geometries and loads applied in their model. While the majority of analytical models, excluding Tehrani and Saket [12], utilize a fracture mechanics approach, the differences in loading, crack representation, and equations used provide significantly different crack-growth approximations. In each paper it is argued that the crack-growth approximation method presented agrees well with cracks growing in actual rail. This may be true for specific situations, but few models are capable of correctly predicting crack-growth behavior in a variety of loading and cracked conditions. This is largely due to the many assumptions utilized in the analysis.

2.4 Detail Fracture Growth in Worn Rail

Jeong, Tang, and Orringer [15] explore the combined effects of crack growth and rail wear in their 1998 paper funded by the Volpe National Transportation System Center. The purpose of this paper is to

determine vertical and gauge-face wear limits using a fracture mechanics approach. The work of Jeong et al. builds on the research documented in Orringer et al., and adopts many of the assumptions and approximations explained in Section 2.2. In addition, many of the equations and assumptions used by Jeong et al. are similar to those seen in the works explained in Section 2.3. In order to include rail wear into the failure analysis, the geometry of the rail is substantially simplified. Two failure modes are considered in the analysis: 1) rail failure due to the brittle fracture of a rail containing a detail fracture, and 2) permanent plastic bending. Through analysis it is shown that rail fracture is the dominant failure mode. This section will outline the rail fracture method of analysis developed by Jeong et al.

In order to incorporate the rail wear into the analysis, Jeong et al. [15] utilizes a simplified geometric representation of the rail. The simplified geometry is an I-beam with rectangular sections representing the head, web, and base of the actual rail. Gauge-face and vertical wear are uncoupled and are divided into two analysis groups that are analyzed separately. The idealized wear patterns are shown in Figure 2.3.

a) Vertical (top) wear   b) Gauge-face (side) wear
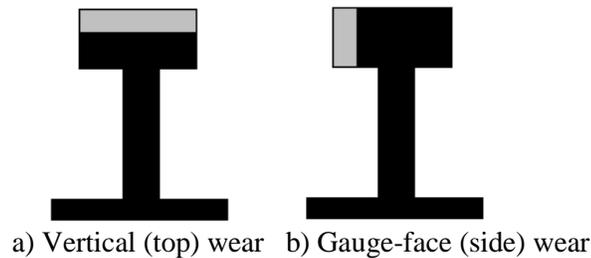
Figure 2.3 Idealized top and side wear patterns [15].

The amount of wear is quantified using increments of $1/16^{ths}$ of an inch. The loss in head-height for a top-wear profile is defined as Δh and the loss in gauge-width for a side-wear profile is defined as Δw. Section properties for various rail sections are given in Appendix A as defined by Jeong et al.

Like the majority of the rail failure models previously discussed, Jeong et al. [15] utilize a biaxial ellipse to represent the detail fracture in their crack-growth model. The location of the detail fracture is adopted directly from the empirical relationships developed by Orringer et al. The location of the detail fracture is shown in Figure 2.4.
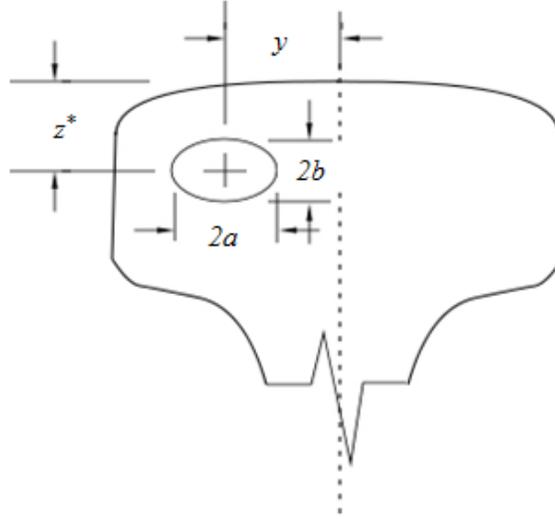
Figure 2.4 Detail fracture location in rail head [15].

The location of the detail fracture is described by empirically determined dimensions y and z^* given by the Equations 2.9 and 2.10.

$$y = 1.1874 - 2.9523\left(\frac{A}{A_H}\right) + 3.4306\left(\frac{A}{A_H}\right)^2 \tag{2.9}$$

$$z^* = 0.6213 + 1.7580\left(\frac{A}{A_H}\right) - 1.7933\left(\frac{A}{A_H}\right)^2 \tag{2.10}$$

where A is the crack area, and A_H is the cross-sectional area of the rail head. The crack aspect ratio is treated as a constant and defined as 0.7. Jeong et al. state that this assumption agrees with examined detail fractures in actual rails. Using the assumed aspect ratio, the crack area, A, is given by the equation:

$$A = 0.7\pi a^2 \tag{2.11}$$

where a is the semi-major crack dimension. The semi-minor dimension is labeled b. The stress intensity at the crack edge is given by Equation 2.2 as developed by Orringer et al. with the crack dimension a substituted in for r. Rail bending, thermal, and residual stresses are developed using the same techniques provided by Orringer et al. The bending stress at the detail fracture is determined by treating the rail as a beam on a continuous vertical, lateral, and torsional elastic foundation. The magnitude of the applied wheel load is adopted directly from the work of Orringer et al. The longitudinal thermal stress is determined assuming a characteristic temperature differential and thermal expansion coefficient. The material properties and seasonal temperature differentials are also adopted directly from Orringer et al.

The residual stresses are calculated as a function of crack size using the empirical relationships developed by Orringer et al. and given in Equation 2.3.

The estimated side-wear and top-wear limits are developed separately using two sets of calculations. In the first set of calculations, Equation 2.2 is set equal to the Mode I fracture toughness of the rail material, and the critical crack size and the amount of wear are varied. When the wear is plotted as a function of critical crack size this creates a decreasing curve. In the second set of calculations, two assumptions are made. The first assumption is that the initial crack size is 5% HA, and the second assumption is that the inspection interval is greater than 20 MGT. Therefore, it is concluded that the rail wear becomes critical when the crack grows from the initial size to critical in 20 MGT. Multiple crack-growth calculations are completed for various amounts of wear. When the wear is plotted as a function of the crack size at 20 MGT, this creates an increasing curve. The wear limit and the critical crack size are determined as the point of intersection between the curve from the first set of calculations and the curve from the second set of calculations. This intersection of the two plots is shown in Figure 2.5.
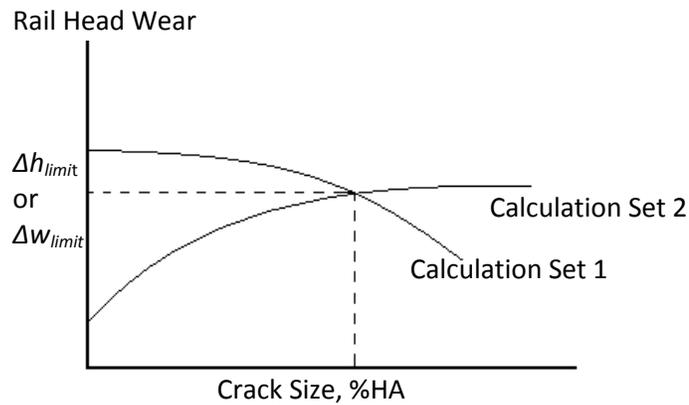
Figure 2.5 Wear limit estimation based on fracture mechanics [15].

This procedure is repeated for both types of rail wear to develop the top-wear and side-wear limits. When compared to typical wear limits implemented by American railroads, the wear limits developed using the fracture mechanics approach allow more wear [15].

In a 2003 follow up report, Jeong [16] explores the applications of the analytical crack-growth model developed by Orringer et al. and used by Jeong et al. to rail defect management with an emphasis on developing inspection intervals and risk management. In order to develop an inspection interval program, it is assumed that 0.1 service defects per mile is the target occurrence. This is known as the defect rate. A self-adaptive inspection program that is a function of the tested defect rate is developed utilizing the crack-growth model, a Weibull statistical model estimating the rate of internal defect development, and

the probability of defect detection as a function of defect size. When compared to constant testing intervals, the self-adaptive inspection program requires more inspections per year, but decreases the defect rate significantly.

2.5 Conclusion

American railroads currently prevent track failure by means of track maintenance programs including rail wear inspection and defect detection using ultrasonic sensors. Uncoupled top-wear and side-wear limits and wear inspection intervals are determined from decades of experience and empirical data collected by the railroads. The ultrasonic testing schedule is developed from risk analysis statistical models. In order to provide an understanding of the crack-growth behavior of the detail fracture, multiple researchers have completed rail testing and the development of crack-growth models utilizing linear elastic fracture mechanics. Using these methods, the stress intensity at the assumed geometric representation of the detail fracture can be described, and the number of cycles for a crack to grow from an initial size to critical can be estimated. It is desired to validate these methods by correlating the crack-growth results with detail fractures growing in actual rails. Orringer et al. provide extensive information regarding the growth of detail fractures including full-scale crack-growth experiments in actual rails, laboratory tests, material property development, and analysis method development. The conclusions developed from the experimental stages and the analysis methods developed serve as a foundation for the majority of crack propagation models discussed in this section. The work of Jeong et al. incorporates wear into the crack-growth analysis in an effort to develop wear limits based on a fracture mechanics approach. In order to quantify the amount of wear that the rail has suffered, vertical head-height loss and gauge-face side loss are decoupled, and a simplified block geometry is used to represent the rail.

The limitations of the work of Jeong et al. including the decoupling of the side and top-wear and the block representation of the worn geometry can be addressed by the development of finite element models representing true worn geometries. By assuming a crack geometry similar to that shown in Figure 2.4 and incorporating that geometry into the finite element model, the stress and strain state at the crack edge can be estimated with greater accuracy. In addition, the effect of wheel-induced contact and bending stresses can be examined by modeling the loading of the rail as an interaction with a wheel as completed by Rice and Rungta. This loading can be superimposed with thermal loading of the rail to include all external loads. These improvements can lead to accurate estimated life values for the rail and eliminate the need to develop wear limits as are calculated in Figure 2.5.

23

Chapter 3. Geometric Response Variable Development

In order to represent the geometry of the worn rail in the response surfaces, four geometric independent variables are defined. These variables are the height of the worn rail (h), the total cross-sectional area of the worn rail (A), and the area moments of inertia about the vertical and horizontal axis through the point formed by the intersection of the lines tangent to the slope of each rail foot (I_{xx} and I_{yy}) as shown in Figure 3.1. This point is named the web origin.
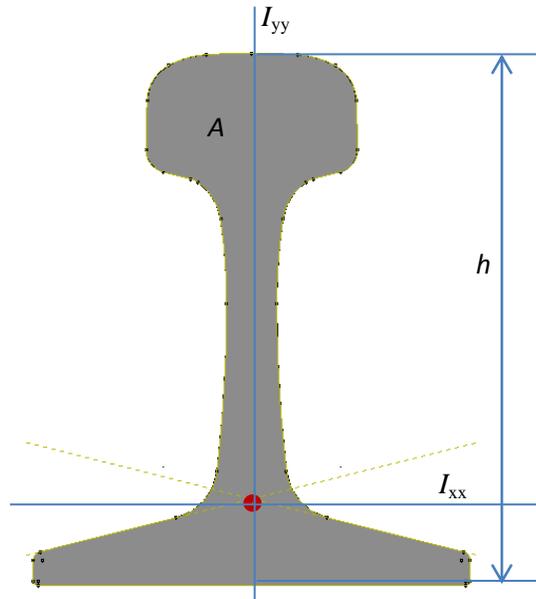
Figure 3.1 136RE rail geometry used to develop the geometric independent variables.

These four section properties must be obtained for each worn rail profile that is evaluated in the response surfaces. There are multiple types of rails currently in service, and each rail type belongs to a larger group that combines rails of similar cross-sectional dimensions. The 136RE rail group is most commonly used in current operations, and therefore the first iteration of the response surfaces will focus on evaluating rails that belong to this group. In order to obtain the four geometric independent variables, the output from the Track Geometry car is curve-fit using multiple isoparametric curve elements in MATLAB, and the section properties are calculated approximately from the fitted worn profile [18]. In addition, certain worn profiles must be recreated in the Abaqus Finite Element (FE) model in order to develop the samples necessary to build the response surfaces. The profile fitting process solves for curve elemental nodal values that are used to update the FE model with the new worn profile. The process developed for fitting the worn profile and calculating the section properties is explained in this chapter.

3.1 Importing Track Geometry Car Data

The Track Geometry cars uses a laser to determine the profile of the rail as the car passes over the section of track of interest. The profile can be outputted as X-Y coordinates and read into Microsoft Excel or another spreadsheet editor. The data consists of approximately 200 points representing the rail-head profile for each side of the track, the left or right rail. The origin from which the data points are originally referenced is located at the center of the track at ground level. Before the curve fit of the data is calculated, the data for each side of the track is transformed by changing the data origin to the point shown in Figure 3.1. This point is an appropriate origin because the base and web of the rail do not undergo wear; therefore, this point's location remains constant. In addition, the profile data is corrected for any cant or lean of the rail, making it easy to superimpose the worn rail head profile onto an existing unworn geometry. An example of the corrected data set for a virtually unworn section of 136RE rail is shown in Figure 3.2.
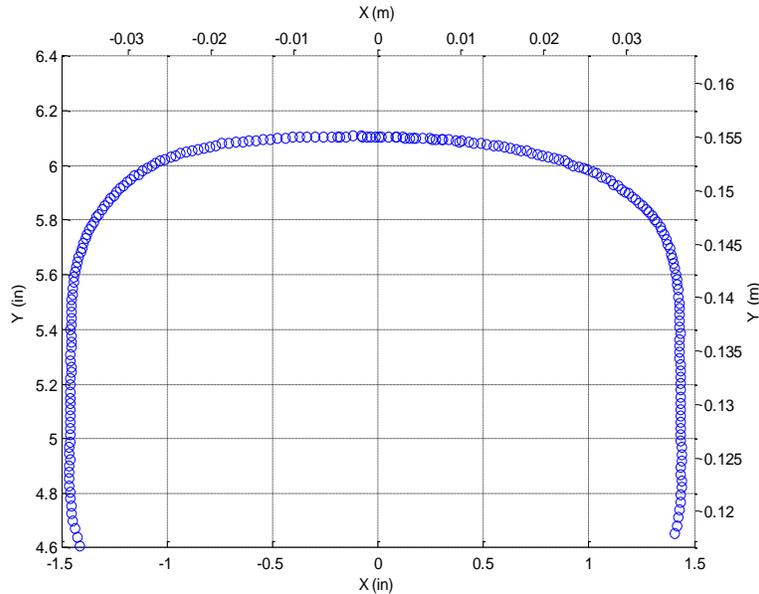
Figure 3.2 Corrected Track Geometry car X-Y data for unworn 136RE rail section.

After transformation, the data is ready to be fit as explained in the following section.

3.2 Fitting Worn Rail Head

The transformed Track Geometry car data is fit using four isoparametric cubic curve elements and the least squares method. The curve element definition is adopted from Finite Element Methods and is that of a four noded, C^0 continuous, isoparametric curve element. The element definition for X and Y in terms of the natural parameter ξ ranges from -1 to 1 for a given element is:

$$x(\xi) = N_1 X_1 + N_2 X_2 + N_3 X_3 + N_4 X_4$$

25

$$y(\xi) = N_1 Y_1 + N_2 Y_2 + N_3 Y_3 + N_4 Y_4 \tag{3.1}$$

In Equation 3.1, N refers to the four interpolation or shape functions, and X and Y are the desired nodal curve coordinates [17]. In order to use this element, the interpolation functions must first be derived in terms of ξ. For a four noded cubic curve element, the evenly spaced nodes are located at ξ equals -1, -1/3, 1/3, and 1. The interpolation function vector N is defined by the equation:

$$N = [1 \quad \xi \quad \xi^2 \quad \xi^3][A]^{-1} \tag{3.2}$$

where the 4x4 matrix A is defined by evaluating the vector $[1 \quad \xi \quad \xi^2 \quad \xi^3]$ at the four nodal values of ξ [17]. After evaluating A, Equation 3.2 becomes:

$$N = [1 \quad \xi \quad \xi^2 \quad \xi^3]\begin{bmatrix} 1 & -1 & 1 & -1 \\ 1 & -\frac{1}{3} & \frac{1}{9} & -\frac{1}{27} \\ 1 & \frac{1}{3} & \frac{1}{9} & \frac{1}{27} \\ 1 & 1 & 1 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} \frac{-9\xi^3+9\xi^2-1}{16} \\ \frac{27\xi^3-9\xi^2-27\xi+9}{16} \\ \frac{-27\xi^3-9\xi^2+27\xi+9}{16} \\ \frac{9\xi^3+9\xi^2-\xi-1}{16} \end{bmatrix} \tag{3.3}$$

yielding the four interpolation functions. These interpolation functions are plotted in Figure 3.3.
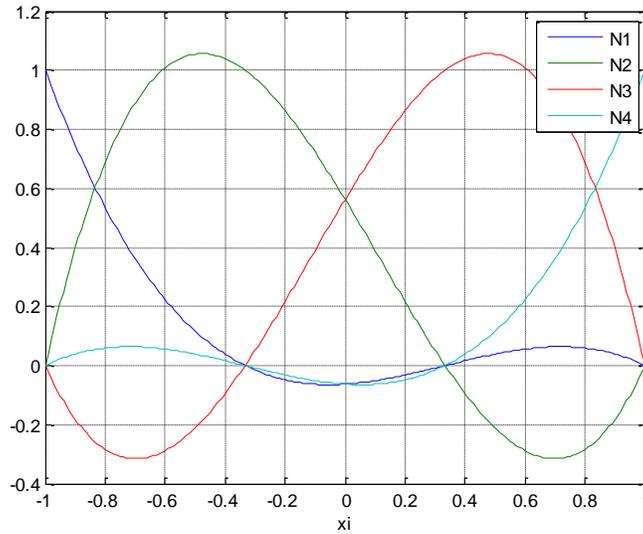
Figure 3.3 Isoparametric C^0 continuous cubic interpolation functions.

Each interpolation function is characterized as having a value of one at its given node and a value of zero at the other three nodes. For example, interpolation function N_1 has a value of one when ξ equals -1, but has a value of zero at ξ equals -1/3, 1/3, and 1.

Once the necessary interpolation functions are defined, the worn profile's X-Y data is broken into four elemental sets of X and Y vectors. The element break points are shown in Figure 3.4 where each color represents the data for one of the four elements.
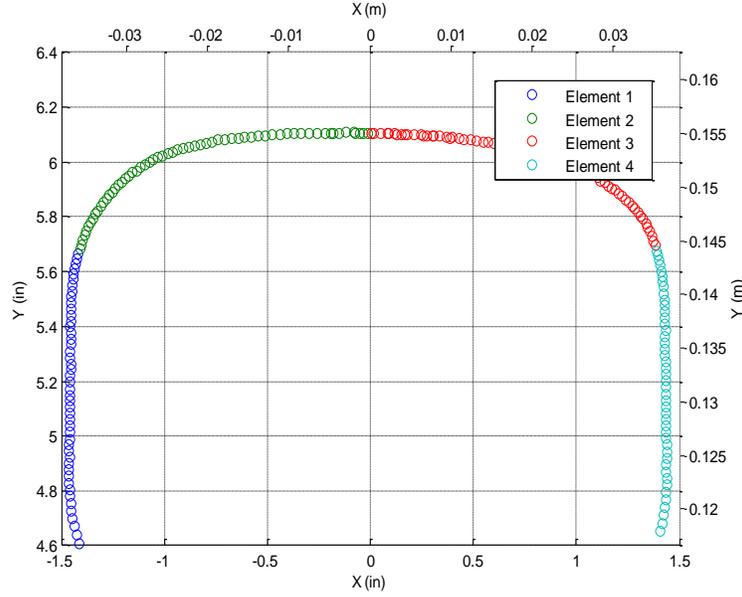
Figure 3.4 Element breaks for rail profile data.

After the data has been divided into four curve elemental sets, the ζ value for each data point in the element X and Y vectors is calculated using its position out of the overall number of data points for that element. If there are Q data points in the element X and Y vectors, the elemental interpolation matrix, N_e, is defined as a matrix with Q rows and 4 columns where each row is generated by evaluating the four interpolation functions at each data point's ζ value. Therefore, for the first row of the elemental interpolation matrix, the first column is N_1 evaluated at the current ζ value, the second column is N_2 evaluated at the current ζ, and so on. The elemental interpolation matrices for the four elements are assembled into a single global interpolation matrix, N_g. Because each curve element features four nodes and there are three nodes where two elements share the same node, there are thirteen total nodal coordinates to solve for. The nodal x-coordinates and y-coordinates are calculated separately using the least squares equations:

$$\{X\} = [B]^{-1}[N_g]^T\{x\}$$

$$\{Y\} = [B]^{-1}[N_g]^T\{y\} \ where \ [B] = [N_g]^T[N_g]. \tag{3.4}$$

27

In Equation 3.4, the lowercase vectors x and y represent the Track Geometry car measurements of the rail profile data, and the uppercase vectors X and Y represent the nodal vectors of interest [17].

Before the nodal coordinates are calculated, the global interpolation matrix used in the calculation of the nodal x-coordinates is partitioned to specify the x-value of the node connecting curve element 2 and element 3. This node represents the center of the rail, and is given an x-value of 0 to assure this function. The calculated fit is superimposed onto the Track Geometry car data and shown in Figure 3.5.
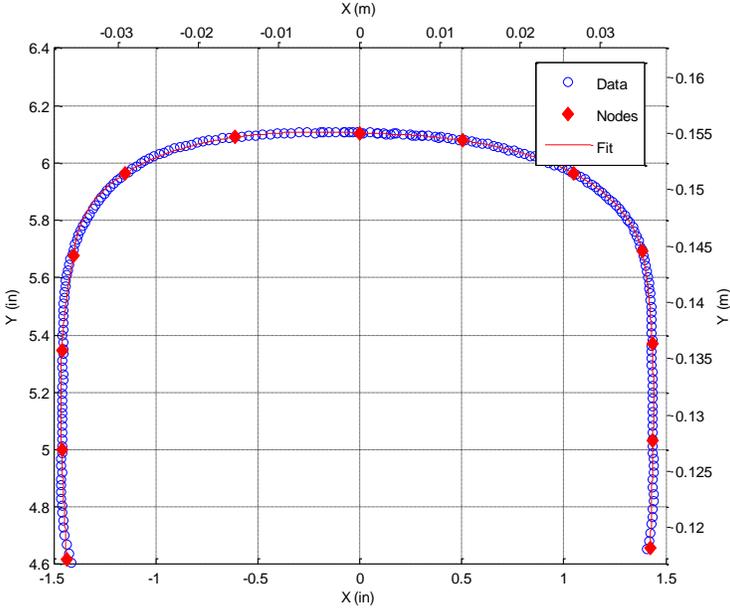
Figure 3.5 Curve fit for unworn 136RE rail head section.

Once complete, the fit is evaluated by calculating the error residuals and other statistical properties such as the mean error, standard deviation, and signal-to-noise ratio using built in MATLAB functions. The X and Y residual plots for the unworn rail profile fit are shown in Figure 3.6.
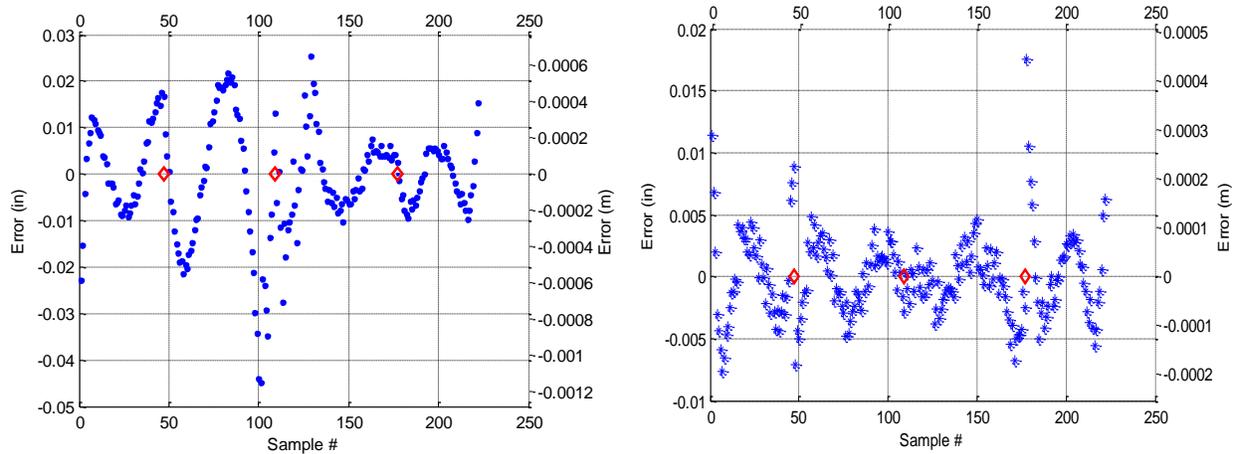
Figure 3.6 Error plots for unworn 136RE rail. Left plot is X residual, right plot is Y residual.

In Figure 3.6, the element breaks are marked with red diamonds. Ideally, the mean error is zero for a good fit, suggesting random scatter as the source of error with no bias. While Figure 3.6 shows that the error isn't purely random, the low magnitude of the error, as well as the zero mean trend, suggest that the curve fit represents the data well.

3.3 Fitting Unworn Rail Web

In order to connect the fitted worn rail head to the remaining rail section, the left-side upper web of a 136RE rail is fit using a single element identical in definition to those used to fit the rail head. The data required to complete this fit is taken by querying multiple points from an engineering model of an unworn 136RE rail. The interpolation matrix for this element is partitioned in order to maintain that the first node of the fitted element remains constant and on the cross section of the 136RE rail's profile. Therefore, only three nodal values are calculated using the least-squares method. Once the fit is complete, it is mirrored across the Y-axis in order to obtain a fit for the right upper web. The two fitted web elements are shown in Figure 3.7 with the data used to derive them.
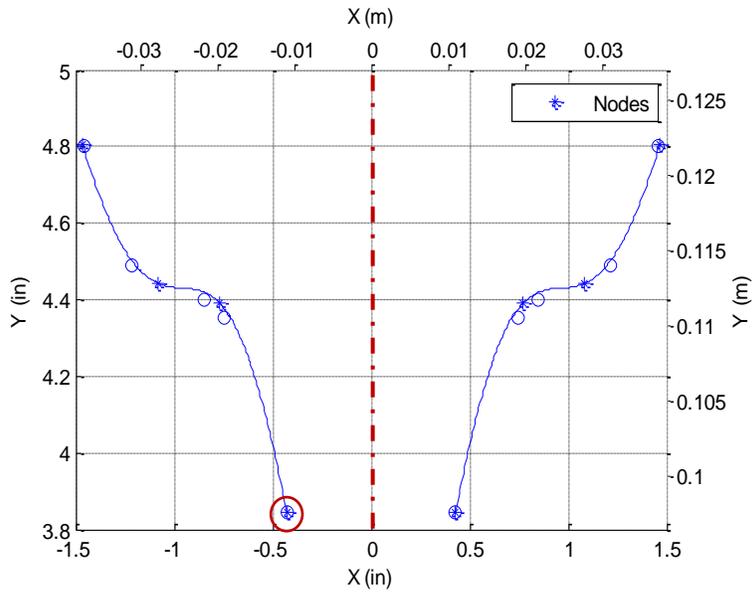
Figure 3.7 Curve fit for upper web of unworn 136RE rail.

The point circled in red in Figure 3.7 is the node that is partitioned out of the interpolation matrix during the fit calculation. Its coordinates are taken directly from the unworn 136RE rail's dimensions. This allows the fitted web element to connect to the remaining web and base section of the rail without any discontinuity. The dotted red line represents the mirrored axis used to develop the right side's fit. The fit statistics including mean error, standard deviation, and signal-to-noise ratio are also calculated for the web elements to evaluate the quality of the fit using built in MATLAB functions.

3.4 Intersecting Fitted Profiles

Once the fits for the worn rail head and the upper web are complete, the overall cross-section of the rail head and upper web is created by calculating the intersection points between the two fits. The x and y coordinates of the intersection between the first rail head element and the left web element is calculated, as well as the intersection between the fourth rail head element and the right web element. The nodes at the end of the intersecting elements are moved to the point of intersection in order to close the profile and trim the remaining lines. Figure 3.8 illustrates the intersecting fits and the completed profile.
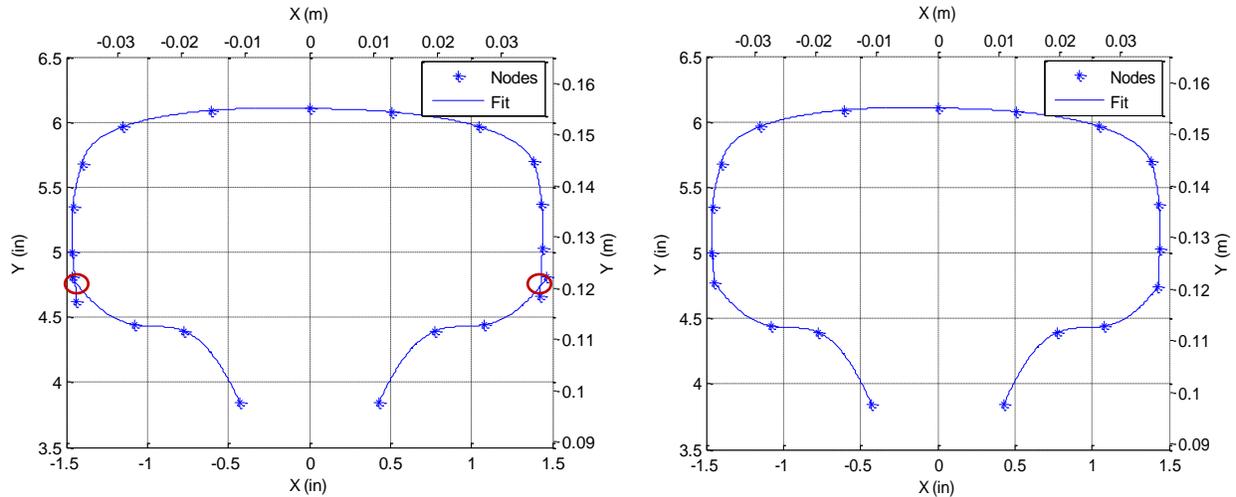
Figure 3.8 Intersection rail head and web fits and the resulting closed cross section.

In Figure 3.8, the intersection points between the rail head and web are shown in the left plot circled in red. Once the two intersection points are determined, the intersecting elements are trimmed and their end nodes translated to the point of intersection as shown in the right plot. This process provides a closed profile that can be used to calculate the section properties of the worn rail, and the nineteen curve element nodes that are imported into the Abaqus model to update the worn geometry of the rail.

3.5 Approximate Calculation of Section Properties

Once the overall cross-section of the worn rail head and web is complete, the four geometric independent variables used in the response surfaces (h, A, I_{xx}, I_{yy}) can be calculated approximately using a scan-line algorithm. The scan-line is a horizontal line at a constant y-value that intersects the rail profile in at least two places. The scan line begins at the smallest y-value—the y-value at node one—and increases in value by a delta of 0.1 inches (0.00254 meters) until it reaches a y-value equal to the overall height of the rail. An example of the scan-line at a specific y value is shown in Figure 3.9.
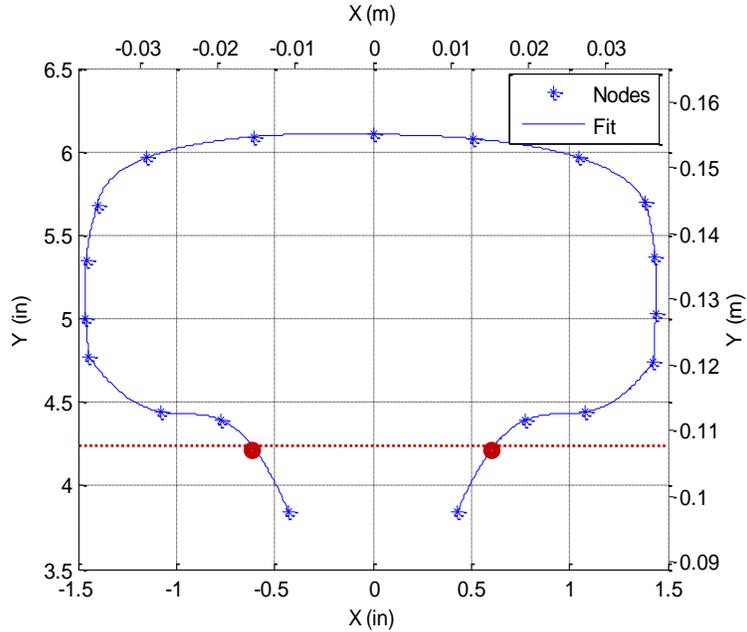
Figure 3.9 Scan-line intersecting the web elements.

The rail profile is divided into two sections: the left side is composed of the left web curve element and curve elements one and two from the rail head, and the right side is composed of the right web curve element and curve elements three and four from the rail head. The developed MATLAB section properties code evaluates which curve elements are intersected by the scan-line at its current y-value, and calculates the x-coordinate for the intersection. If one of the curve elements that compose the left side of the rail and one of the curve elements that compose the right side are intersected by the current scan-line, the scan-line area is determined by the equation:

$$A_e = \Delta y(x_R - x_L) \tag{3.5}$$

where A_e is the scan-line area, Δy is the scan-line delta value of 0.1 inches (0.00254 meters), x_R is the right intersection x-coordinate, and x_L is the left intersection x-coordinate. In Figure 3.9 the scan line intersects the left web curve element and the right web curve element at the points shown in red to form a scan-line area. The overall area of the rail head and upper web cross-section (A) is determined by summing the scan-line areas of all the scan lines. For certain worn profiles, it may be possible for the scan-line to intersect a specific curve element more than once, or for the scan line to only intersect curve elements on the left side of the rail or the right side of the rail. In these cases, additional scan-line areas are defined using the two intersection points in the current curve element. These scan-line areas are added to the overall area of the cross-section.

The height (h) of the worn rail is specified as the maximum y-coordinate from the worn rail-head data. The area moment of inertia about the X-axis is calculated using the equation:

$$I_{xx} = \sum_{i=1}^{N}(y_i^2 \Delta A_i) \tag{3.6}$$

where N is the number of scan-lines used in the area calculation, y_i is the vertical distance from the origin to the current scan-line, and ΔA_i is the scan-line area as defined in Equation 3.5. In order to calculate the area moment of inertia about the Y-axis, a new elemental area is defined. Each scan-line area is broken into smaller square elements of width Δx equal to 0.1 inches (0.00254 meters). Each element is defined by its area, $\Delta x \Delta y$, and its x and y coordinates from the origin. The area moment of inertia about the Y-axis (I_{yy}) is calculated using the equation:

$$I_{yy} = \sum_{i=1}^{N} \sum_{j=1}^{M} (x_j^2 \Delta A_{ij}) \tag{3.7}$$

where N is the number of scan lines used, M is the changing number of elements for each scan line, x_j is the horizontal distance from the element to the web origin, and ΔA_{ij} is the elemental area equal to $\Delta x \Delta y$.

In order to obtain the cross-sectional area of the full rail, the approximate area of the rail head and upper web as defined above is added to the known area of the remaining rail. The value of the height is unchanged. The area moments of inertia of the full rail are calculated using the known moments of inertia of the remaining rail about its centroid and the parallel axis theorem. Therefore, the total moment of inertia about either axis through the web origin is calculated using the equation:

$$I_{xx} = I_{xx}' + I_C + Ar^2 \tag{3.8}$$

where I_{xx}' is the area moment of inertia of the rail head and upper web about the axis through the web origin, I_C is the area moment of inertia of the remaining rail about its centroid, A is the area of the remaining rail, and r is the perpendicular distance from the centroid of the remaining rail area to the web origin.

3.6 Conclusion

The geometry of each worn profile evaluated in the response surfaces is represented using the geometric independent variables h, A, I_{xx}, and I_{yy}. These four variables are the only geometric inputs for the response surfaces and must represent any worn profile of interest. In order to obtain these section properties, the overall cross-section of the rail head and upper web is developed through least squares curve fitting techniques using multiple isoparametric cubic curve elements. After the cross-section is determined, the four section properties for the rail head and upper web cross-section are calculated approximately using a scan-line algorithm. The section properties for the complete cross-section of the

33

rail are calculated using the calculated variables from the partial cross-section and the known variables for the remaining rail. The section properties for the complete rail become the geometric independent variables used in the response surfaces.

The MATLAB code utilized to develop the profile fits and calculate the geometric independent variables also outputs the nineteen nodal curve coordinates used to represent the worn profile in the FE model. In order to develop the response surfaces, the nodal coordinates from multiple worn profiles must be imported into Abaqus to create the samples needed. These samples must represent the different wear patterns and wear severity seen in typical rail operation. By using partitioned matrices in the fitting of the worn profile, the FE model can be updated quickly without having to rebuild the base of the rail. The MATLAB code developed to fit the worn rail profiles and calculate the necessary section properties is found in Appendix B.1 and Appendix B.2.

Chapter 4. Load Response Variable Development

The loading for a given rail is developed as three separate loads—the vertical wheel load, the lateral wheel load, and the rail temperature load. These loads are represented in the response surfaces using three independent load variables, the maximum vertical wheel load (P_{MAX}), the curvature of the rail in degrees ($°C$), and the temperature delta for the rail ($\varDelta T$). The maximum vertical wheel load is developed from the Wheel Impact Load Detector (WILD) maximum load data for the given section of track. Each section of track has different amounts and types of traffic which results in different maximum load distributions. The curvature of the rail induces lateral wheel loads, and can be related to the vertical load through the lateral-to-vertical wheel-load ratio. The lateral-to-vertical wheel-load ratio ranges from about 0.025 to 0.5 for rails with degrees of curvature up to 14 degrees. The lateral-to-vertical wheel-load ratio increases as the curvature radius of the rail decreases, and a higher ratio indicates a larger lateral load. The wheel load history is built using multiple repetitions of the vertical and lateral wheel loads. The temperature delta or range for the rail represents the difference between the maximum and minimum temperatures that the rail sees in a 24-hour period. Each temperature induces a constant longitudinal stress in the rail that cycles from the maximum temperature to minimum temperature over the course of a day. The development of the three independent load variables used in the response surfaces and to develop the FE models is explained in this chapter.

4.1 Maximum Vertical Wheel Load Definition

The maximum vertical wheel load is the load applied to the rail by the wheel in the vertical or Y direction. The value of this independent load variable is determined using the WILD data for the section of track of interest. The Wheel Impact Load Detector is a sensor that measures the load magnitude applied at a specific point along a rail for a given month. This sensor outputs the speed of the train, the wheel weight, the overall peak load applied by the wheel as it passes over the point of interest, and the load ratio [2]. The peak load is a dynamic load that can be represented by scaling the static wheel load (weight) by the calculated load ratio. The output from the WILD sensor is read into Microsoft Excel and manipulated in MATLAB. The peak load distribution is used to develop the maximum vertical load variable. An example of the peak load distribution created in MATLAB from the WILD data is shown in Figure 4.1.

35
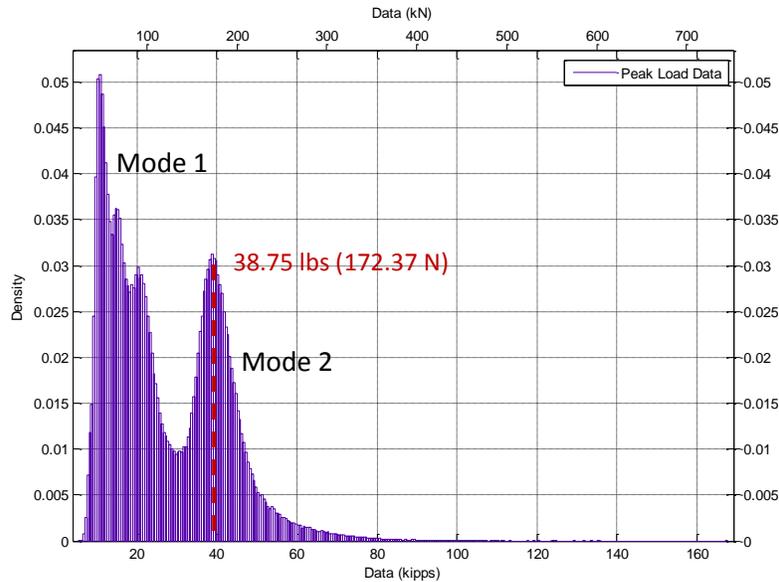
Figure 4.1 Peak Load Distribution from WILD data.

The load distribution in Figure 4.1 is characterized by two primary peaks and a long tail. Peak 1, called model 1, is the distribution of unloaded freight cars and peak 2, called mode 2, is the distribution of loaded freight cars and locomotives. Because of the low magnitude of loading, the loading attributed to the mode 1 distribution does not significantly affect the fatigue life of the rail [7]. Therefore, only mode 2 is considered when developing the maximum vertical load variable.

In order to develop a representative load for the mode 2 peak load distribution, this distribution is fit using MATLAB's statistics toolbox. Before the second peak of the distribution can be fit, it must be generated by itself. To do this, we examine the second peak and make the assumption that it is symmetric about a center axis or mean value. The mean value of the mode 2 distribution is evaluated using the peak load distribution and the distribution fitting GUI in MATLAB (dfittool). This value is also shown in Figure 4.1. Once this value is obtained, a new vector of peak load values can be defined using this value as a minimum. The distribution of this vector represents the right half of the overall mode 2 distribution. The full mode 2 distribution is developed by mirroring the half distribution across its axis of symmetry at the mean value. The half distribution and full distribution for mode 2 are shown in Figure 4.2.

36
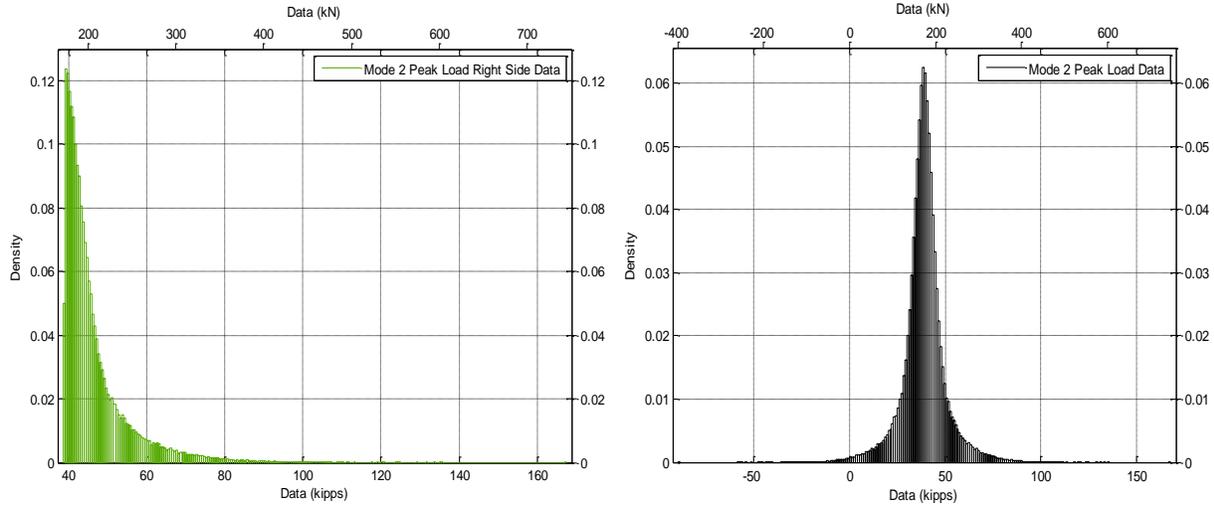
Figure 4.2 Mode 2 half distribution shown in left plot and full distribution shown in right plot.

After the full singular distribution for mode 2 has been developed, it is fit in the MATLAB dfittool GUI using the t-location scale distribution. The t-location scale distribution has a probability density function given by the equation:

$$P(x) = \frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sigma\sqrt{\nu\pi}\Gamma\left(\frac{\nu}{2}\right)}\left[\frac{\nu+\left(\frac{x-\mu}{\sigma}\right)^2}{\nu}\right]^{-\left(\frac{\nu+1}{2}\right)} \tag{4.1}$$

where Γ is the gamma function, μ is the location parameter or mean, σ is the scale parameter or standard deviation, and ν is the shape parameter or degrees of freedom of the distribution [18]. The gamma function is given by the equation:

$$\Gamma(x) = \int_0^\infty e^{-t}t^{x-1}dt \tag{4.2}$$

and interpolates the factorial function for the integer x [18]. The distribution fitting GUI outputs the values for the three t-location scale parameters and estimates the error associated with each parameter. Figure 4.3 illustrates the t-location scale fit of the mode 2 peak load distribution.
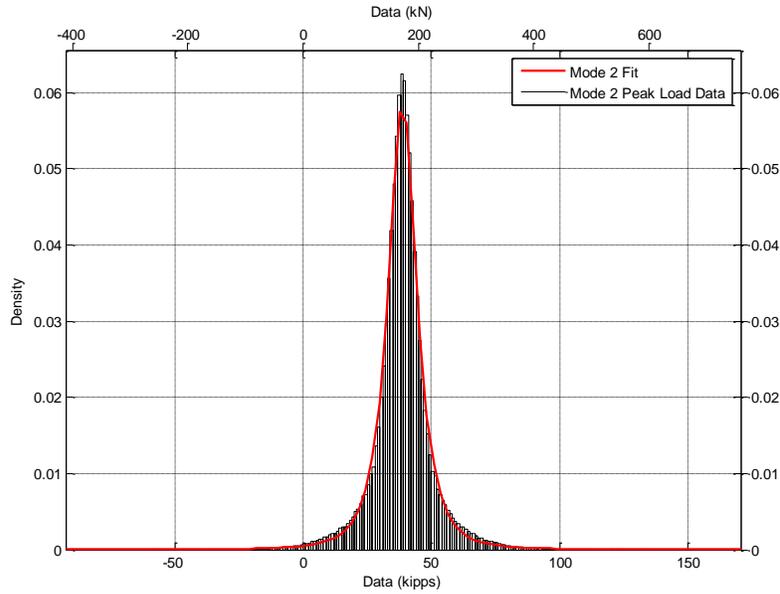
Figure 4.3 Distribution fit of mode 2 peak load data.

Of the distributions available in MATLAB, the t-location scale distribution provides the best fit for the mode 2 peak load data. The best fit is defined as the distribution that fits the shape and magnitude of the data with the lowest estimated error for the distribution parameters. The estimated values for the three t-location scale parameters and the estimated error are shown in Table 4.1.

Table 4.1 Estimated parameters and error for mode 2 peak load distribution fit.

Parameter	Estimate (kips)	Std. Error (kips)	Estimate (kN)	Std. Error (kN)
μ	38.75	0.0183549	172.37	0.0816466
σ	6.06968	0.0215982	27.00	0.0960735
ν	2.20173	0.0153633	9.80	0.0683393

After the parameters associated with the t-location scale distribution are estimated for the mode 2 distribution, the maximum vertical wheel load is calculated in using a defined equation based on the distribution fit parameters. In order to account for the tail of the distribution above the mean, the maximum vertical wheel load is defined using the equation:

$$P_{MAX} = \mu + 3\sigma \qquad (4.3)$$

where μ and σ are the mean and standard deviation of the distribution as given in the fit of the mode 2 peak load distribution. For the fitted data shown in Table 4.1, this produces a maximum vertical load equal to 57.0 kips (253.55 kN). This method is a conservative approach to representing a single instance

38

of the vertical load applied by a locomotive or car wheel for a given month. The MATLAB code used to develop and fit the mode 2 peak load distribution is found in Appendix C.

4.2 Curvature Definition

The curvature of the rail causes lateral loading, and is directly related to the lateral-to-vertical wheel-load ratio. The lateral-to-vertical wheel-load ratio is the ratio of the lateral wheel load to the vertical wheel load, and is used to calculate the magnitude of gauge face loading applied to the rail by a locomotive or car wheel. The lateral-to-vertical wheel-load ratio increases as the degree of curvature increases and the radius of curvature decreases. The *L/V* ratio for tangent track ranges from 0.025 to 0.05 [7]. Table 4.2 provides *L/V* ratios for rails as a function of track curvature given in degrees as developed by Orringer et al. [7].

Table 4.2 *L/V* ratios as a function of track curvature [7].

Gentle Curves		Medium Curves		Sharp Curves		Very Sharp Curves	
°C	L/V	°C	L/V	°C	L/V	°C	L/V
2.9	0.182	3.7	0.242	8.0	0.409	11.5	0.472
3.1	0.212	4.6	0.302	8.4	0.416	12.0	0.475
3.4	0.222	5.8	0.364	9.5	0.445	12.5	0.478
		7.5	0.406	10.0	0.451	13.0	0.481
				11.0	0.465	13.5	0.485

Table 4.2 can be used to develop an empirical equation that provides the lateral-to-vertical wheel-load ratio as a function of the track curvature independent response variable. This fitted data from Table 4.2 is shown in Figure 4.4.
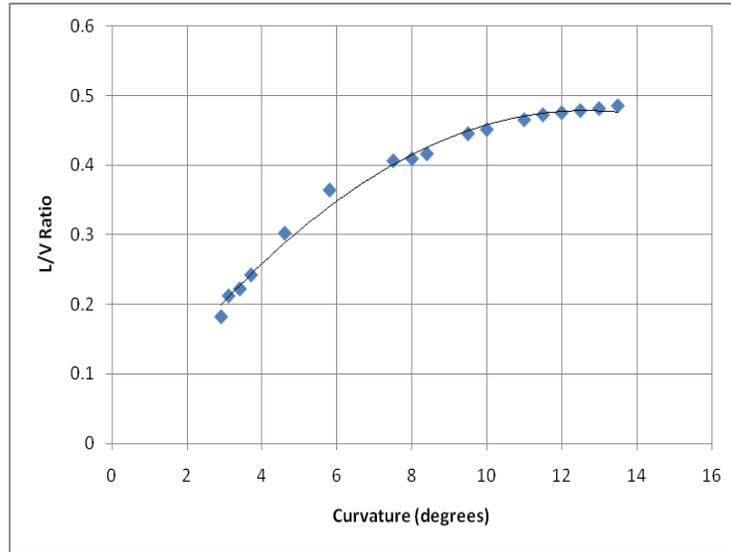
Figure 4.4 *L/V* ratio as a function of track curvature in degrees.

The equation of the fitted line shown in Figure 4.4 is given by the equation:

$$y = -0.003(°C)^2 + 0.0747(°C) + 0.0068 \tag{4.3}$$

In addition, the track curvature in degrees can be related to the curvature radius by the equation:

$$°C = \frac{5730}{R\,(ft)} = \frac{1746.504}{R\,(m)} \tag{4.4}$$

Where $°C$ is the curvature of the rail in degrees, and R is the curve radius in feet or meters. The lateral-to-vertical wheel-load ratios provided in Table 4.2 are representative of 100-ton freight cars with vertical wheel loads averaging 33,000 lbs. (146,791.26 N). Experimental results suggest that the *L/V* ratio varies with car weight, and that lighter cars produce higher *L/V* ratios [7]. Therefore, it is desired to eventually obtain *L/V* ratios from WILD data for a given track set. However, since this data is not readily available, the curvature values and *L/V* ratios from Table 4.2 are used in the first iteration of the response surfaces.

4.3 Temperature Range Development

The temperature for a given section of track fluctuates over the course of 24 hours between a maximum temperature during the middle of the day and a minimum temperature during the night. The constant longitudinal stress induced in the rail is estimated by the equation for a straight bar that is restrained at its ends:

$$\sigma = -\alpha(T - T_{ref})E \tag{4.5}$$

where α is the coefficient of thermal expansion, T is the rail temperature, T_{ref} is the reference temperature, and E is the elastic modulus [19]. This approximation assumes that the reference temperature represents a zero state of stress. Because the rail has significantly different boundary conditions than the generalized case stated above, the thermal stresses induced in the rail by the maximum and minimum rail temperatures are developed in the FE model using temperature loads. The rail installation temperature corresponds to a zero state of stress and varies from 85 to 110 degrees Fahrenheit (29.44 to 43.33 degrees Celsius) [2]. The temperature range independent variable is defined using the equation:

$$\Delta T = T_{MAX} - T_{MIN} \tag{4.6}$$

where T_{MAX} is the maximum temperature reached during the day, and T_{MIN} is the minimum temperature reached during the night. Assuming that the high installation temperature is valid for the current rail, and knowing that a maximum temperature higher than the installation temperature would induce a longitudinal compressional stress in the rail, T_{MAX} in Equation 4.6 is defined as the installation temperature of the rail. Therefore, any minimum temperature less than the installation temperature would induce a longitudinal tensile stress in the rail.

The values for T_{MIN}, and therefore ΔT, are developed using historical temperature records for a given month. Ideally, a minimum temperature monthly distribution would be used to develop a characteristic minimum temperature using a method similar to the development of the maximum vertical wheel load explained in Section 4.1. However, typical historical weather resources only provide daily averages and record high and low temperatures. Therefore, the monthly average of the daily record low temperatures is used to develop the value of T_{MIN} used in Equation 4.6. The temperature delta independent variable is then defined for the given month using this value and the rail installation temperature and Equation 4.6.

4.4 Conclusion
The independent load variables are used to develop the loads applied in the Abaqus FE model and to represent the rail's loading in the response surfaces. The three loads applied in the FE model are the vertical wheel load, the lateral wheel load, and the temperature load. The magnitude of the vertical wheel load is equal to the maximum vertical wheel load variable defined in Section 4.1. The lateral wheel load is developed using the maximum vertical load and the lateral-to-vertical wheel-load ratio calculated from the empirical equation relating to degree of curvature. The lateral wheel load is given by the equation:

$$P_{lateral} = P_{MAX} \left(\frac{L}{V}\right) \tag{4.7}$$

The temperature load is applied at the negative value of the temperature delta variable, and implies a constant tensile stress in the rail. The loads applied in the FE model are cyclic in nature, and the

definition of a cyclic load history from the three FE loads will be addressed in Chapters 5 and 6. The three independent load variables are the only load inputs to the response surfaces, and therefore must be able to represent the variation of loading seen by the rail. The definition of each independent load variable follows strict procedures as defined above which enable consistent representation and development of the rail's loading.

Chapter 5. Abaqus Finite Element Model

The results that are used to develop the dependent or output variables of the response surfaces are developed using the FE model of the rail assembly and a fatigue post-processor. The FE model is created in Abaqus version 6.10-1 [23]. The FE model is used to generate elastic stress results in the rail for the three applied loads—the vertical wheel load, the lateral wheel load, and the thermal load. The three loads applied in the FE model are formulated using the three independent load variables defined in Chapter 4. Because these loads are represented as cyclic loads, the finite element model must represent the maximum and minimum stress states corresponding to the maximum and minimum magnitude for each load. For the two wheel loads, the maximum and minimum load magnitudes are represented by the standard bending and reverse-bending load cases. The temperature load fluctuates between the maximum temperature representing a zero state of stress and the stress state at the minimum temperature.

The geometry of the worn rail is represented in the FE model using the nodal coordinates calculated in Sections 3.2 through 3.4. Parameterized nodal coordinates in the Abaqus part sketcher interface allow the rail geometry of the FE model to be updated quickly and efficiently with a new worn geometry. Three main types of worn rail profiles are evaluated in the response surfaces. These profile types include the unworn rail geometry, side-wear profiles, and top-wear profiles. While there is only one unworn geometry profile, multiple side-wear profiles and top-wear profiles with different levels of wear are evaluated in the FE model to develop the response surfaces. The worn profiles are labeled by the amount of wear that they have undergone in sixteenths of an inch.

An example of a current rail maintenance schedule implemented for a given section of track involves the evaluation of the rail's wear using the Track Geometry car, and the evaluation of defects or cracks present in the rail using the Ultrasonic Test car. The frequency of each inspection varies substantially for different sections of track. According to the specifications of the Ultrasonic Test car, the smallest crack size that the ultrasonic sensor can detect with a suitable confidence interval (65%) is a crack with surfaces area equal to 10 percent of the overall rail head cross-sectional area (10% HA) [3]. Because the past load history of the rail for the current worn profile is unknown, a 10% HA crack is modeled in the rail head to represent a "worst case" scenario where a crack has grown to the detectable size. The remaining cycles to failure or fatigue life of the rail is calculated using the results from this FE model. Additionally, it is important to understand the growth rate of the crack as it increases in size. Therefore, multiple crack sizes ranging from 5% HA to 50% HA are modeled in the rail head for the given worn profile geometry. The results of these models are used to develop an equation for the crack-growth rate as a function of load cycles. The development of the FE model to provide the results necessary for the fatigue calculations is explained in this chapter.

5.1 Parametric Worn Rail Head Geometry

The nodal coordinates calculated in Chapter 3 are imported directly into the Abaqus CAE parametric editor for the rail cross-section geometry. The partitioned interpolation function matrices used to calculate the nodal coordinates specify the x and y values for nodes one and nineteen as well as the x value for node 10. Nodes 1 and 19 are the connecting nodes between the fitted rail head/web cross-section and the rest of the rail. Therefore, it is important for these nodal coordinates to stay constant in order to guarantee continuity between the fitted rail head and upper web and the remaining cross-section of the rail. The geometry of the remaining rail web and base are developed using dimensions provided in the AREMA handbook for 136RE rail sections [4]. The geometry of the web and base in the FE model has been simplified by removing unnecessary geometric features such as fillets and radii in order to make the geometry easier to mesh. The cross-section of the rail head and web are formed by connecting the nodes for one of the six fitted cubic curve elements with a spline curve in the Abaqus sketcher interface. The parameterized rail cross-section is shown in Figure 5.1.
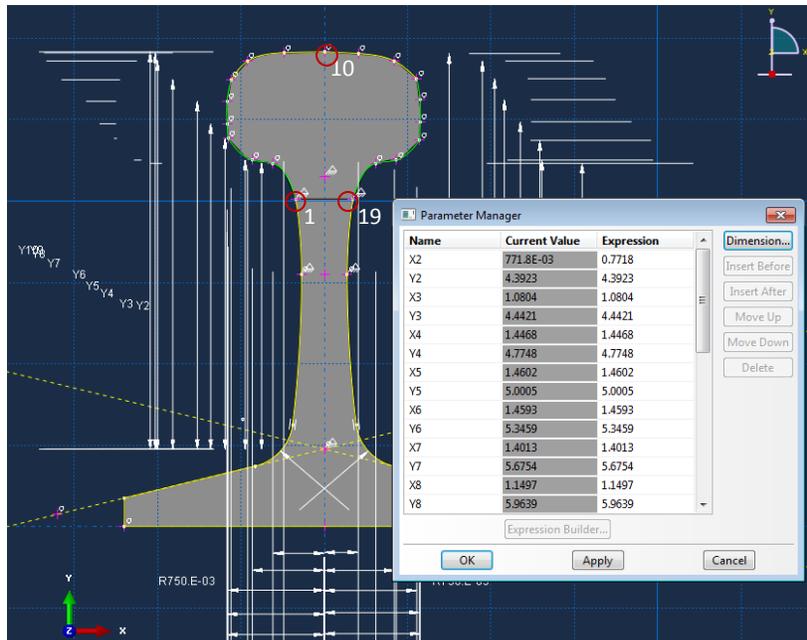
Figure 5.1 Parameterized rail head geometry using nodal coordinates.

The nodal coordinates calculated in Chapter 3 are entered into the sketcher parametric manager shown in Figure 5.1. These coordinates represent the vertical and horizontal dimensions for each node from the origin defined in Figure 3.1. The horizontal or x-coordinate of node 10 is fixed at a value of zero to maintain that it remains on the center vertical axis of the rail. The partitioned curve element nodes 1, 10, and 19 are circled in red and labeled in Figure 5.1. Once the nodal coordinates of the new worn profile are entered, the geometry of the rail is updated automatically. In addition, the partitioning and meshing of

the rail's volume are also updated automatically. This feature allows for rapid development of multiple models representing different worn geometries in a short period of time with decreased user effort.

5.2 Parametric Crack Geometry

The most common subsurface flaw responsible for track failure due to fracture is the detail fracture [7]. Therefore, the geometry and location of the crack represented in the FE model to account for the worst case load history of the rail is that of an approximated detail fracture. The detail fracture is represented as a biaxial elliptical crack in the x-y plane of the rail head. The shape and size definition of the crack in the FE model is identical to the approximation used by Jeong et al. and shown in Figure 2.4. However, the empirical relationship for the location of the flaw shown in Figure 2.4 is not used in the FE model. It is concluded from studying multiple detail fractures of various sizes and the resources introduced in Chapter 2 that the crack does not grow uniformly as it propagates. The crack tends to grow downward and towards the field side of the rail in order to encompass a large portion of the rail head cross-sectional area before penetration of the rail head's outer surface. This trend is shown in Figure 5.2.
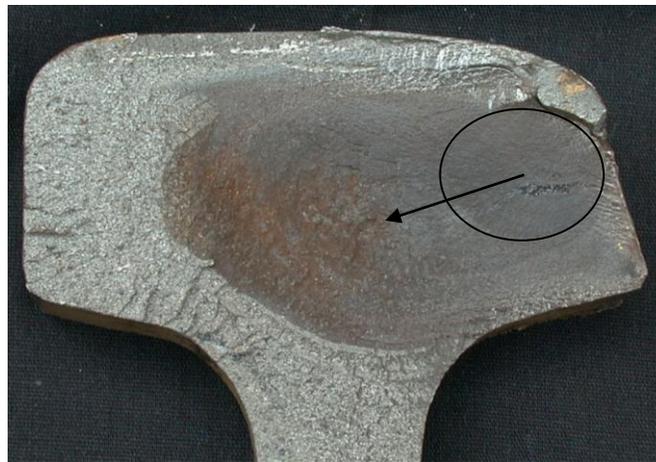
Figure 5.2 Detail fracture in rail head.

The detail fracture is represented in the FE model as a surface partition on the cross-section of the rail. In order to account for the growth trends of the crack shown in Figure 5.2 and to avoid penetration of the rail head's running surface for worn and unworn profiles, a new crack location relationship is developed that prevents the top edge and the gauge edge of the elliptical crack from intersecting the rail's running surface. This is achieved by applying a constant dimension between the top edge of the crack and the crown of the rail (node 10), as well as between the gauge edge of the crack and the gauge edge of the rail (node 13) as shown in Figure 5.1. The unworn rail profile illustrated in Figure 5.1 and the multiple side-wear profiles use identical crack location definitions.   This location relationship and the shape and size

parameters of the crack are entered into the sketcher parametric manager for the surface partition sketcher interface. The parametric representation of the crack for the unworn rail is shown in Figure 5.3.
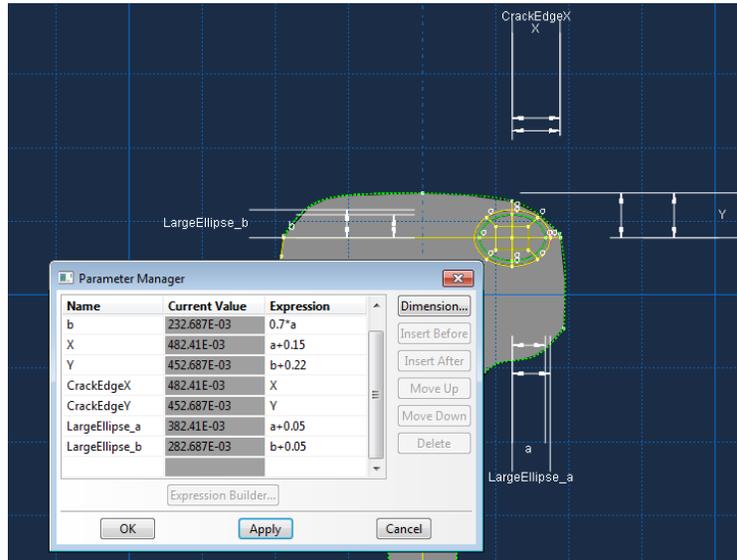
Figure 5.3 Parametric representation of the detail fracture in the Abaqus FE model.

The elliptical crack is characterized by the semi-major and semi-minor axes, *a* and *b*, and an aspect ratio (*a/b*) of 0.7 [15]. The top-wear profiles use a different crack location definition that moves the crack farther into the center of the rail. The dimension *X* as shown in Figure 5.3 is given by the expression *a* + 0.3 for the top-wear profiles. The dimension *Y* is given by the expression *b* + 0.18. The different crack location for the top-wear profiles coincides more closely with real cracks in similar worn profiles than the location definition shown in Figure 5.3. This conclusion is determined from examining multiple top-worn rail specimens with detail fractures of various sizes. This location also allows the crack to grow to larger sizes before penetration of the rail's worn outer surface. Stress results from FE models using the new crack location relationship and the location used by Jeong et al. for an unworn 136RE rail with a 10% HA flaw are compared to determine the magnitude of the new location's effect on the stress results. The location of the crack produced von Mises stresses which are within 5% of those produced using the crack location featured in the work of Jeong et al. Therefore, the new location is considered equivalent to that used in by Jeong et al.

The crack is formed by tying the degrees of freedom of two rail sections together at every node on the cross section except the nodes formed by the crack surface partition. This is shown in Figure 5.4 through the use of the Abaqus constraint manager.
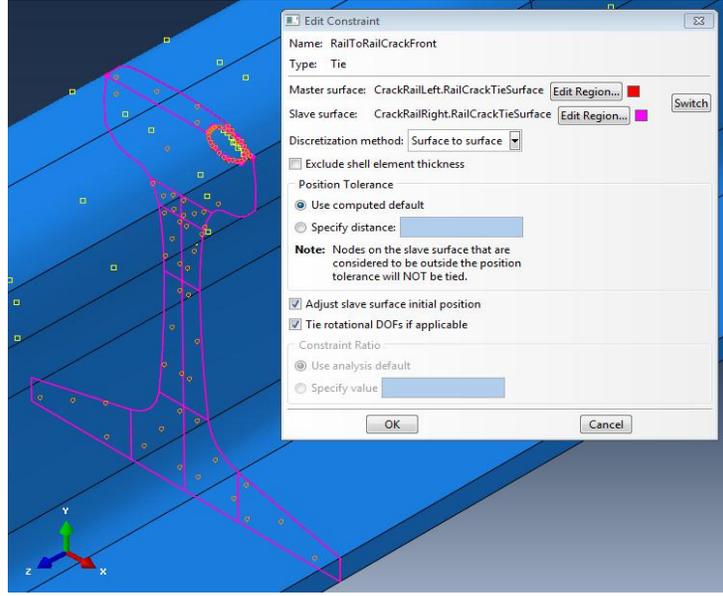
Figure 5.4 Surface tie between rail sections to form crack.

The surface formed on each rail's cross-section by the crack surface partition are free to move relative to each other. In addition, surface- to-surface contact between the crack surfaces is defined using the penalty normal and tangential interaction definitions in Abaqus. The coefficient of friction used in the tangential definition for the crack interaction is 0.65 representing dry steel on steel [20].

Five crack sizes are evaluated in each FE model to develop the crack-growth curve for the given worn geometry. The crack sizes that are generated in the FE model of an unworn 136RE rail are 5%, 10%, 20%, 35% and 50% HA flaws. The semi-major axis dimension, a, shown in Figure 5.3 can be calculated for each flaw size using the equation:

$$a = \sqrt{\frac{\left(\frac{A}{A_H}\right)\frac{A_H}{100}}{0.7\pi}} \tag{5.1}$$

Where *a* is the crack's semi-major axis dimension, *A* is the crack's surface area, A_H is the unworn 136RE rail head's cross-sectional area, and $\left(\frac{A}{A_H}\right)$ is the % HA flaw size. The rail-head cross-sectional area for an unworn 136RE rail is 4.86 inches^2 (0.00314 meters^2) as given in Appendix A [15]. The parameters for the five crack sizes evaluated in the unworn 136RE rail FE model are provided in Table 5.1.

Table 5.1 Crack parameters for various crack sizes.

Rail HA in^2 (m^2)	Crack % HA	Crack Area in^2 (m^2)	a in (m)	b in (m)
	5	0.243 (1.57e-4)	0.332 (8.440e-3)	0.233 (5.910e-3)
	10	0.486 (3.13e-4)	0.470 (1.194e-2)	0.329 (8.260e-3)
	20	0.972 (6.27e-4)	0.667 (1.689e-2)	0.465 (1.182e-2)
4.86	35	1.701 (1.097e-3)	0.879 (2.234e-2)	0.616 (1.564e-2)
(3.14e-3)	50	2.430 (1.570e-3)	1.051 (2.670e-2)	0.736 (1.869e-2)

The geometry of the crack is updated automatically by entering the semi-major axis dimension, a, for each flaw size into the parametric editor. The parametric modeling of the crack enables the FE model to automatically update the partitioning and meshing of the rail section. This feature enables the study of multiple crack sizes within each worn rail cross-section in a short period of time by updating one parameter.

Unfortunately, the majority of worn profiles used in the development of the response surfaces are unable to contain a 50% HA crack without intersection of the rail's cross-section. The majority of side-wear profiles are not able to house cracks larger than 20% HA without intersection of the rail's outer surface. The majority of top-wear profiles are unable to house cracks larger than 15% HA. Therefore the five crack sizes that are used to calculate the crack-growth rate in the side-wear profiles are 5%, 7.5%, 10%, 15%, and 20% HA flaws. The crack sizes used in the top-wear profiles are 5%, 7.5%, 10%, 12.5%, and 15% HA flaws. This method assures that the crack-growth curve used to calculate the crack-growth rate is developed with five data points representing the five crack sizes for each worn profile. The development of the crack-growth curve and the calculation of the crack-growth rate are explained in Chapter 7.

5.3 Model Assembly and Material Definition

The FE model represents a section of rail with two wheel load locations spanning multiple railroad ties. The full section of rail modeled is 97.5 inches (2.4765 meters) long, and is created by tying two shorter rail section parts together at the crack front as defined in Section 5.2. The two wheel locations modeled represent the standard bending and the reverse bending load cases. The standard and reverse bending vertical and lateral wheel loads are applied through contact between a deformable train wheel and the rail. The right rail section part is longer than the left rail section part in order to represent the reverse bending wheel location. Therefore, two deformable wheel parts and two rigid axle parts are also instanced in the

FE model. The assembly of the two rail section instances, the two wheel instances, and the two rigid axle instances that compose the FE assembly is shown in Figure 5.5.
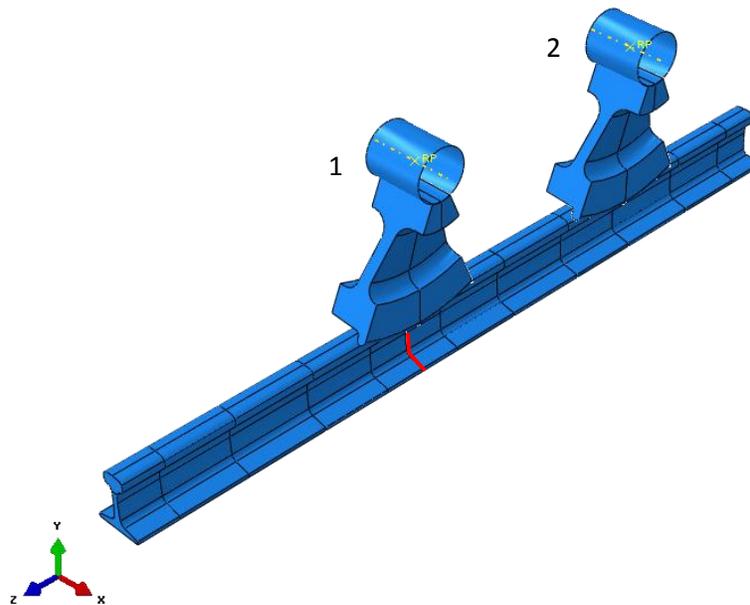
Figure 5.5 Finite element model assembly.

The standard bending load is applied at wheel location one, and the reverse bending load is applied at wheel location two, as labeled in Figure 5.5. The crack front is located directly under the standard bending load; therefore, the left and right rail sections are tied together at the partition shown in red in Figure 5.5.

The wheel loads are transmitted through the rigid axle and elastic wheel to the rail through contact. The geometry used to model the train wheel is that of an unworn 40 inch (1.016 meter) diameter locomotive wheel. The complexity of the wheel's cross section is reduced by removing small radii, fillets, and other features in order to make the cross-section easier to mesh. In addition, virtual topology is used to combine faces with smaller surface areas to continue to improve the mesh of the wheel. The cross-section of the train wheel used in the FE model is shown in yellow Figure 5.6 [4].
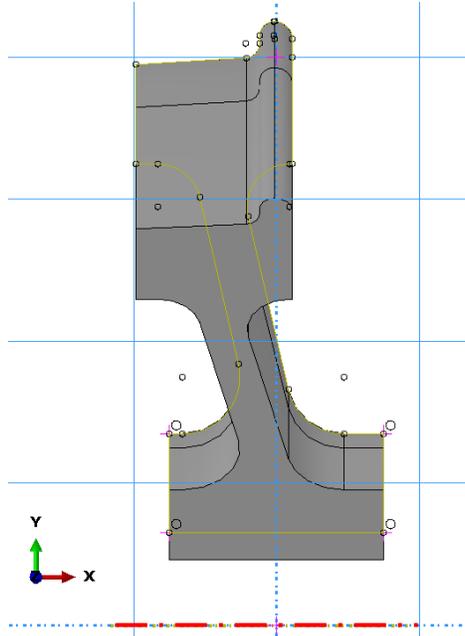
Figure 5.6 Wheel profile developed for use in FE model.

A 1/8 model of the train wheel is created by revolving the cross-section shown in Figure 5.6 by 45 degrees about the axis of rotation shown in red. The 1/8 model is used instead of a full 360 degree rotation of the wheel cross-section to reduce the number of elements necessary to model the train wheel. In order to determine that the 1/8 wheel model is sufficient to transfer the load into the rail, models with a full wheel and models with a 1/8 wheel are compared. Contact stress results between models featuring the full wheel and models featuring the 1/8 wheel differ by less than 1%. Therefore, the 1/8 wheel model adequately transmits the wheel load from the rigid axle to the rail.

The wheel location for the standard bending wheel load is directly above the crack front and corresponds to the point along the rail with the maximum bending moment magnitude from the wheel load. Therefore, the center plane of the standard bending 1/8 wheel is on the same plane as the surface tie between the left rail and the right rail or the cross-section crack plane. This standard bending wheel load represents the maximum vertical and lateral wheel loads for the fatigue calculations. The reverse bending load represents the minimum wheel loads for the fatigue calculations. The location for the reverse bending load is the wheel location that induces the maximum tensile stress in the longitudinal or z-direction at the intersection of crack plane and the top of the rail. For a rail treated as a beam on a constant vertical elastic foundation, the reverse bending wheel location is given as the longitudinal distance from the crack plane using the equation:

$$x_0 = \frac{\pi}{2\lambda_v} \tag{5.2}$$

where x_0 is the longitudinal distance and λ_v is defined in the following equation [15]. The variable λ_v is a function of the rail's material and section properties. It is calculated using the equation:

$$\lambda_v = \sqrt[4]{\frac{k_v}{4EI_{xx}}}$$

(5.3)

where k_v is the modulus of the elastic foundation, E is the rail material's elastic modulus, and I_{xx} is the horizontal area moment of inertia of the rail [15]. The horizontal area moment of inertia for an unworn 136RE rail can be found in Appendix A, and is 14.5 in^4 (603.536 cm^4) [15]. Using Equation 5.2, x_0 is calculated as 43.3 inches (1.10 m). In reality, the track is supported by railroad ties that have varying spacing, thickness, and stiffness. Therefore, the reverse bending wheel location used in the FE model is developed using the *S33* (σ_z) stress results from a single wheel model for the given railroad tie configuration. The distance x_0 is found by querying the location of the maximum tensile stress in the rail head as shown in Figure 5.7.
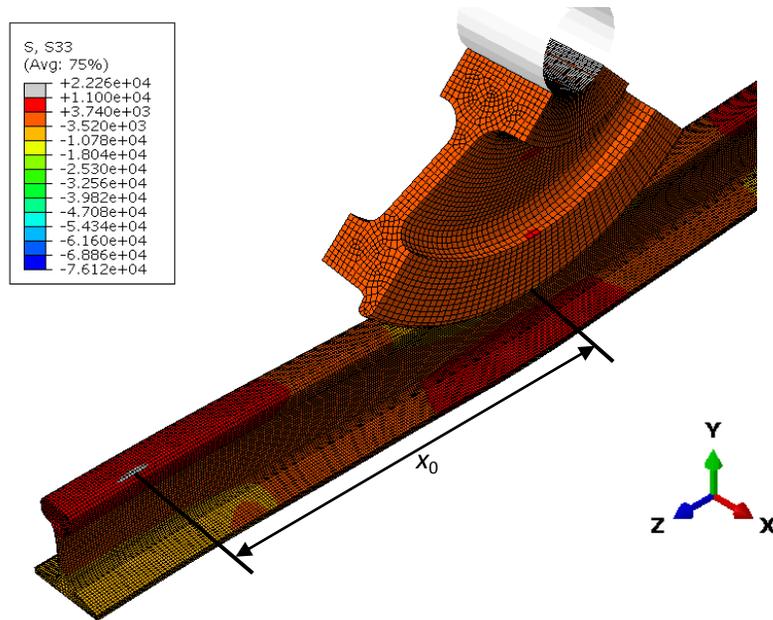
Figure 5.7 Location of the reverse bending wheel load.

The reverse bending distance x_0 is defined as 33.3 inches (0.846 meters). In order to generate the reverse bending stress state at the crack location, the reverse bending wheel load is applied at a distance x_0 inches from the crack along the length of the rail. The location x_0 is verified with the *S33* stress results from the reverse bending load step. This plot is shown in Appendix D.1.

The standard and reverse bending wheels are positioned on the rail so that the flange of the wheel is in contact with the gauge face of the rail. This wheel position represents the worst case scenario for the loading and propagation of the elliptical crack. The wheel position is illustrated in Figure 5.8.
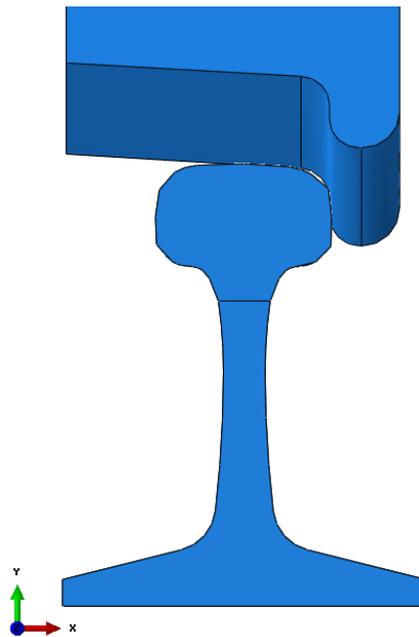
Figure 5.8 Wheel and rail contact position.

This wheel position is achieved by moving the wheel in the negative x-direction until the surface of the wheel flange is in contact with the gauge surface of the rail while maintaining vertical contact between the rail head and the tread surface of the wheel. It is assumed that the side-wear profiles evaluated in the response surface are developed due to higher lateral loads caused by track curvature. The cant of the wheel is defined as the rotation of the wheel about the longitudinal or z-axis. The high rail, low rail geometry of a curved section of track as well as the vehicle dynamics of a train in a curve cause wheel cants greater than zero. Therefore, the wheel is canted (rotated) at 2 degrees about the z-axis for the side-wear profiles. The cant of the wheel moves the vertical point of contact between the rail and wheel from the field side of the rail closer to the top center point of the rail, and moves the gauge face contact between the wheel and the rail to a lower point on the rail's gauge face. This contact position is shown in Figure 5.9.
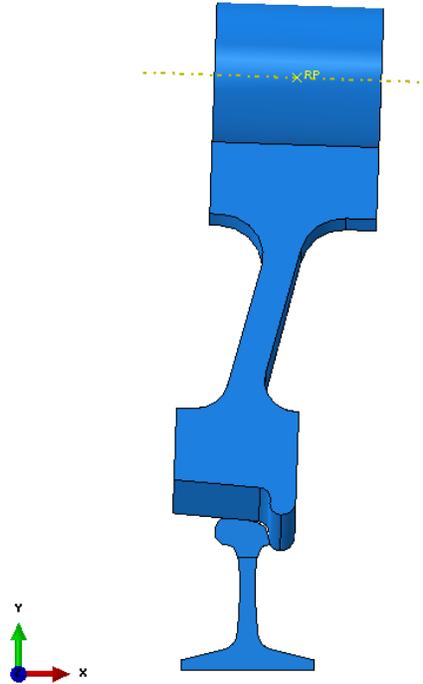
Figure 5.9 Canted wheel contact position for side-wear profiles.

A wheel cant of 2 degree is representative of a curved section of rail on a good elastic foundation where the foundation modulus is approximately 3,000 psi (206.843 KPa) [2]. The cant of the wheel is held at zero degrees for the unworn profile and the top-wear profiles.

In order to generate the elastic stress results necessary for the fatigue calculations, a static incremental non-linear elastic analysis is required. This incremental non-linear analysis is needed to resolve the contact between the rail and the two train wheels. For a standard elastic analysis, the only material properties needed are the elastic modulus (E) and Poisson's ratio (v) for the material. However, because a thermal load is applied to the rail, the coefficient of thermal expansion (α) also needs to be defined. These material properties are relatively constant for different types of steel. Therefore, the material properties used in the model do not change for the different rail grades evaluated in the response surface. The material properties used in the FE model for the two rail sections and the two train wheels are given in the Table 5.2.

Table 5.2 FE model material properties.

E ksi (MPa)	v	α in/in-°F (m/m-°C)
29700 (204774.34)	0.3	6.45×10^{-6} (11.7×10^{-6})

The wheel axles are analytical rigid parts and do not require material property definitions.

5.4 Load Definitions, Contact Definitions, and Boundary Condition Definitions

The vertical and lateral wheel loads and the thermal load are applied through the use of independent load steps in the FE model. In order to complete the fatigue calculations in the fatigue post processor, the independent stress state for the maximum and minimum magnitude of each load must be known. Each load step represents an independent stress state solution for the given loads applied during that step. The results of each step can either be propagated to the next step or terminated. The maximum and minimum magnitudes for the vertical and lateral wheel loads are represented by the standard bending and reverse bending load cases. The vertical and lateral wheel loads are coupled; therefore, the maximum vertical wheel load and the maximum lateral wheel load are applied simultaneously as the standard bending load step, and the minimum vertical and lateral wheel loads are applied simultaneously as the reverse bending load step. The results from each wheel load step are terminated from propagating into the subsequent step to maintain that the solutions for each load step are independent. In the standard bending and reverse bending load steps, the wheel loads are applied as concentrated forces to the reference point of the analytical rigid axle in the negative y-direction and the negative x-direction. The load application dialogue for the standard bending vertical and lateral wheel loads in the standard bending load step is shown in Figure 5.10.
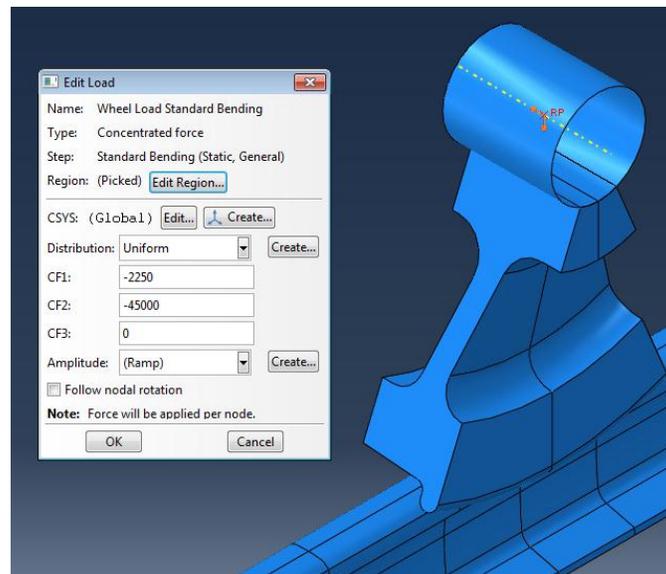
Figure 5.10 Standard bending wheel loads applied as concentrated forces

Assuming tangent track with a curvature of 0.0 degrees, the magnitude of the vertical and lateral components of the wheel load are 45,000 lbs. (200169.9 N), and 2,250 lbs. (10008.5 N) respectively. The maximum vertical load of 45,000 lbs. represents an approximate "average" value from the WILD data

available, and the lateral load is calculated by simply multiplying the lateral-to-vertical load ratio for tangent track (0.05) by the maximum vertical load.

A smaller "ghost" wheel load equal to the weight of the train wheel is applied to the reverse bending wheel in the standard bending load step. This load ensures that the reverse bending wheel maintains contact with the rail during the standard bending load application. The value of the ghost load is more than an order of magnitude smaller than the primary wheel load applied to the standard bending wheel, and does not influence the stress results at the crack front. This was verified by running analyses with just the ghost load active. The vertical component of the ghost load is 2000 lbs. (8896.44 N), and the lateral component is 100 lbs. (444.822 N). The vertical and lateral wheel loads applied to the reverse bending wheel in the reverse bending load step are applied using the technique shown in Figure 5.10, and the ghost load is applied to the standard bending wheel. The maximum magnitude for the temperature load corresponds to the rail installation (neutral) temperature and a zero state of stress. For that reason, this load does not need to be represented in the FE model. The minimum temperature is applied as a thermal load on the rail's volume in a third independent load step—the minimum temperature load step. Therefore, the stress states representing the maximum and minimum cases of the coupled vertical and lateral wheel loads and the temperature load can be developed using a total of three independent load steps.

As previously mentioned, the rail is supported by a vertical elastic foundation that represents the thickness, spacing, and stiffness of an average railroad tie. The tie width is modeled as 9 inches (0.2286 meters), and the tie spacing is modeled as 19.5 inches (0.4953) center to center [2]. The track modulus defined in the elastic foundation interaction manager is an average value of 3,000 psi (20684271.8 Pa) [2]. In addition, a lateral foundation modulus of 3,000 psi (20684271.8 Pa) stiffness is applied to the lateral edge of the rail foot on both the inside and outside edges to counter the lateral load applied by the train wheel. The surfaces to which the elastic foundation is applied are shown in Figure 5.11.
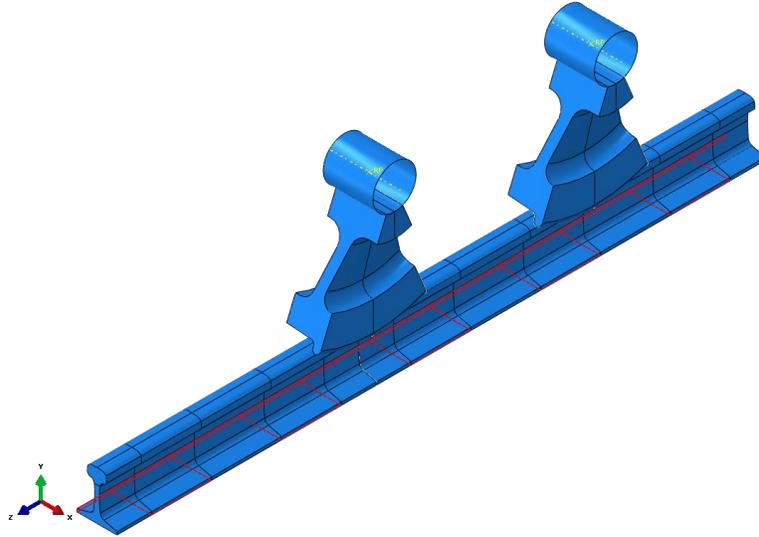
Figure 5.11 Elastic foundation application.

In order to transmit the load from the rigid axle, through the wheel, and into the rail, the interactions between the axle and the wheel and wheel and the rail must be defined. The degrees of freedom of the nodes that compose the outer surface of the rigid axle and the journal surface of the train wheel are tied together. In this way, the lateral and vertical loads applied to the reference point of the rigid axle are immediately transmitted into the deformable train wheel. The load is transmitted into the rail through contact between the rail and the wheel. The contact definition for the reverse bending wheel location is shown in Figure 5.12.
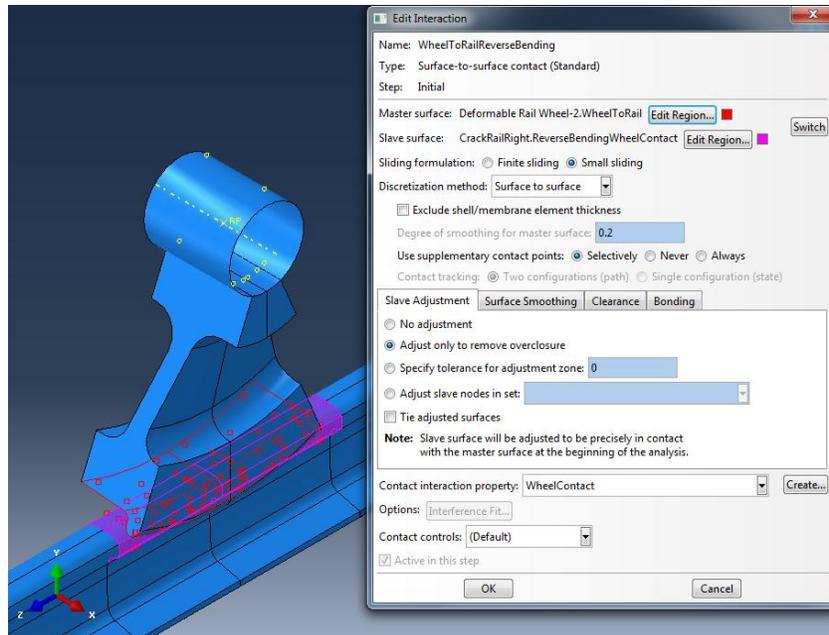
Figure 5.12 Reverse bending contact definition.

The wheel-to-rail interaction is defined as surface-to-surface contact with the wheel tread and flange as the master surface and over-closure adjustment and surface smoothing turned on. The penalty contact property algorithm is used for the normal and tangential contact behavior with a coefficient of friction of 0.4 applied to the entire contact surface. The low coefficient of friction is representative of a lubricated rail running surface [2]. The interaction definition for the standard bending wheel location is identical. The interaction definitions used to develop the rail's elastic foundation, wheel-to-rail contact, and axle-to-wheel surface ties are applied in the initial step and propagated into each load step.

The boundary conditions for the FE model are also applied in the initial step and propagated into the subsequent three load steps. In order to prevent the standard bending and reverse bending wheel instances from moving in the z-direction, the z-displacement of the nodes on the center plane of each wheel is fixed. In modern rail operations, sections of rail are welded together to form continuous ¼ mile plugs that are installed to form new track [2]. The length of the rail section in the FE model is 97.5 inches (2.4765 meters), and z-symmetry boundary conditions are applied to each end of the rail to represent the continuous nature of the rail. The displacement boundary conditions applied to the wheel instances and the symmetry boundary conditions applied to the rail ends are shown in Figure 5.13.

57
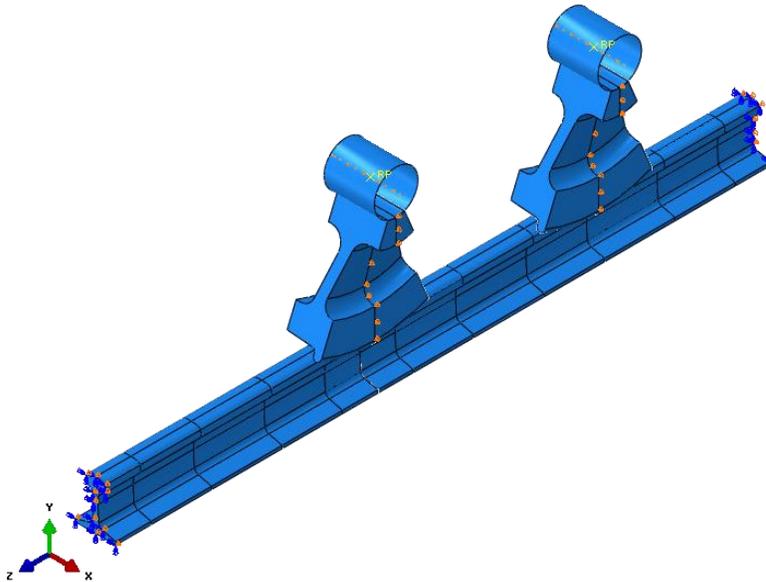
Figure 5.13 Boundary conditions applied to the FE model.

The z-symmetry boundary condition fixes the in-plane displacement and out-of-plane rotations of each node on the two rail-end surfaces. Because rotational degrees of freedom do not exist for three dimensional continuum elements, this reduces to fixed z-displacement for the nodes on the rail-end surface. These boundary conditions and the two elastic foundation definitions assure that the assembly is in equilibrium and the model can be solved for the given wheel and temperature loads, but do not interact with effects of the wheel/rail loading.

5.5 Mesh Development and FE Model Results

The purpose of the meshing scheme used in the FE model is to properly represent the quantity of interest by using a fine mesh at points of interest and a coarse mesh elsewhere. Convergence is reached when the stress and strain results remain unchanged at the points of interest as the mesh density increases. Therefore, multiple meshes are developed to complete a convergence study for the FE model. For the rail model, the mesh is generated on the part. If there are multiple instances of a given part in the model assembly, the mesh only needs to be generated once for the part and is used for each instance of the part in the model assembly. Partitioning of each part's volume enables the Abaqus user to have increased mesh control by dividing the part into smaller cells or volumes. The volumes of the left and right rail sections are partitioned to model the size and spacing of the rail's elastic foundation as shown in Figure 5.11. This partitioning technique divides the overall rail section length into smaller lengths that can meshed independently. The primary point of interest in the finite element model is the volume surrounding the crack front in the left and right rail sections. This location represents the contact location

for the standard bending wheel load and the point of expected failure—the crack front. Therefore, the mesh density along the length of the rail must be much higher in the left and right rail sections surrounding the crack than other sections of the rail to attain a converged solution. In addition, the mesh density along the rail at the point of contact between the rail and wheel for the reverse bending wheel load must also be high to achieve a converged solution.

The mesh in the train wheel is obtained using C3D8 hexahedral elements, a 0.4 global part seed, and the sweep mesh control with advancing front mapped mesh algorithm. The global part seed specifies an approximate element size for the mesh. The sweep mesh control sweeps the mesh of the wheel's cross-section about the axis of revolution used to build the part. The mesh of the train wheel is shown in Figure 5.14.
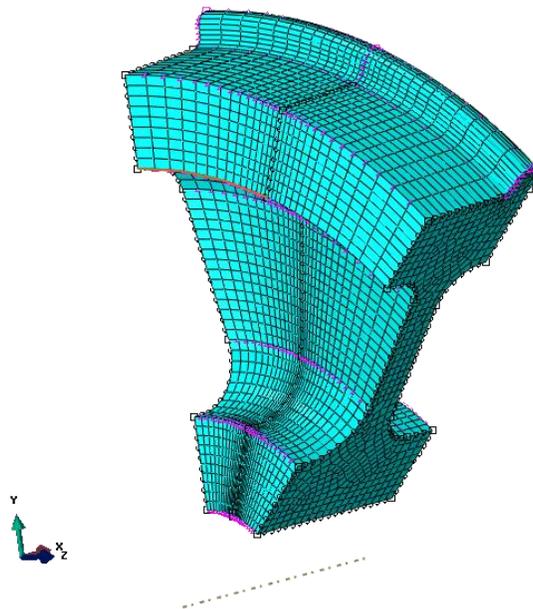
Figure 5.14 Train wheel seeding and mesh.

A biased edge seed is used to concentrate the mesh at the center plane of the wheel. This edge seed is shown as pink triangles in Figure 5.14. In addition, the mesh density in the wheel flange is increased by seeding the flange edge. The edge seeds used to mesh the train wheel use edge seed-by-size definitions. The edge seed-by-size definition specifies an elemental size for the selected edge. In this manner, the wheel mesh at the points of contact between the rail and the wheel is refined. The global part seed is shown as black circles. Because the stress results of the wheel are not of interest, the stress gradient in the wheel does not need to be converged. Therefore, the density of the wheel mesh is only required to represent the shape of the wheel and to resolve the contact between the wheel and the rail. In addition, the wheel serves as the master surface for the contact definition between the wheel and the rail. In the

contact definition, the master surface is typically reserved for the stiffer surface. The master surface's mesh is also coarser in relation to the slave surface to ease computation of the intersection of the slave node with the master surface. The slave node is not permitted to penetrate the master surface. For these reasons, the density of the wheel mesh is not increased as the model's mesh is refined during convergence.

The rail sections are meshed by sweeping the mesh of the rail cross-section along the length of the rail. The elastic foundation cell partitions along the length of the rail allow for the edge seed to be varied for different sections of the overall rail length. The sweep meshing control used to mesh the rail sections uses the medial axis algorithm with reducing mesh transition turned on. Additional surface partitions of the rail's cross section are developed to control the mesh around the crack and to create an organized mesh in the rail's web and base. These surface partitions create a polar mesh in the rail head that is centered around the elliptical crack and biased towards the crack edge. The partitioning of the rail cross-section and volume is shown in Figure 5.15.
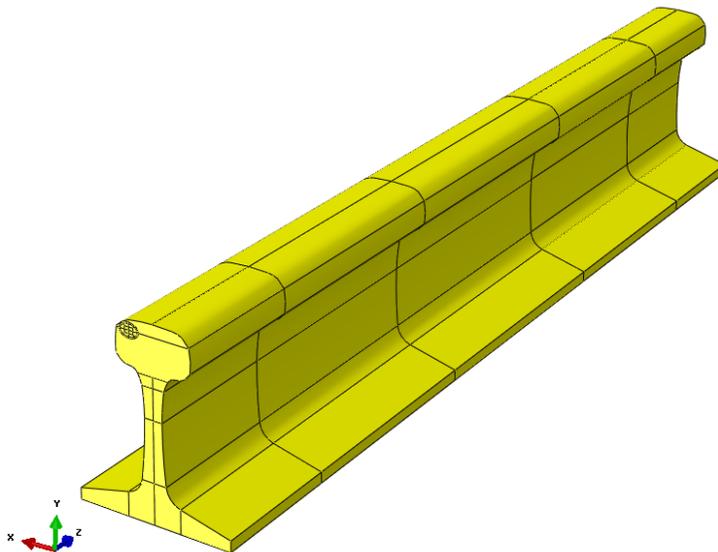
Figure 5.15 Rail partitioning scheme.

Because the rail cross-section partitions are surface partitions and not cell partitions, they do not extend into the volume of the rail. The swept mesh technique generates the mesh on the partitioned surface and then sweeps the mesh along the length of the rail. The surface partitioning of the rail cross-section to create the elliptical crack and maintain a polar mesh about the crack utilizes sketching constraints and the sketcher parametric editor which enable the face partitions to update automatically for new worn geometries and different crack sizes. The partitioning of the rail's web and base are tied to stationary global nodes on the rail's cross-section and do not change for the different rail geometries and crack sizes

60

explored. Therefore, the partitioning of the rail's web and base remains constant. The density of the mesh for the rail is assigned using a combination of global part and edge seeds. The part is initially seeded with a global part seed of 0.2. Multiple edge sets containing edges from the face partition of the cross-section as well as edges along the length of the rail are created to enable rapid refinement of the mesh and convergence exploration. These sets are given in Table 5.3 organized by seeding function.

Table 5.3 Edge sets used to seed the rail model.

Web and Foot	Along Rail Length	Rail Head
RailWebWidthSeed	*RailLengthBiasSeed*	*CrackBiasInLongSeed*
RailFootHeightSeed	*RailCoarseMeshSeed*	*CrackBiasInShortSeed*
	ReverseBendingBiasSeed	*CrackBiasOutSeed*
		CrackEdgeSeed
		RadialSeed

The mesh density of the web and base of the rail is assigned by seeding the *RailWebWidthSeed* and *RailFootHeightSeed* as a seed-by-number seed. An seed-by-number seed definition specifies the number of elements along the selected edge. Because the stress results in the rail head are of primary interest, the mesh of the rail web and base remains relatively coarse. The *RailFootHeightSeed* maintains a constant edge seed equal to 3 elements. The mesh density along the length of the rail is assigned by seeding the *RailLengthBiasSeed*, *RailCoarseMeshSeed*, and *ReverseBendingBiasSeed* as an edge by size seed. The *RailCoarseMeshSeed* set contains the edges along the length of the rail that are in between the two wheel load locations. These edges are seeded with larger elements, because the elements are primarily in bending and do not need a fine mesh to reach convergence. The *RailLengthBiasSeed* is biased towards the crack front to achieve a fine mesh at the crack front and a coarse mesh in the span of the rail away from the crack. The *ReverseBendingBiasSeed* also uses a biased seed to maintain a finer mesh at the point of contact between the reverse bending wheel and the rail. The *CrackBiasInLongSeed*, *CrackBiasInShortSeed*, *CrackBiasOutSeed*, *CrackEdgeSeed*, and *RadialSeed* are sets containing the face partitions that maintain the polar mesh around the crack in the rail head. These sets are seeded with a combination of biased and unbiased edge seed-by-number and edge seed-by-size seed definitions.

In order to explore the stress convergence in the rail head, four meshes with different edge seeds for the edge sets listed above are created. Table 5.4 illustrates the seeding definition for the four meshes used in the convergence study.

Table 5.4 Seeding definitions for convergence study.

	Mesh 1	Mesh 2	Mesh 3	Mesh 4
CrackBiasInLongSeed (By Size)	0.1 and 0.3	0.08 and 0.25	0.06 and 0.2	0.05 and 0.15
CrackBiasInShortSeed (By #)	3	4	5	6
CrackBiasOutSeed (By #)	2 Bias 5	3 Bias 5	5 Bias 5	7 Bias 5
CrackEdgeSeed (By #)	3	5	7	8
RadialSeed (By #)	1 Bias 3	2 Bias 3	3 Bias 4	4 Bias 4
RailCoarseMeshSeed (By Size)	0.75	0.5	0.5	0.5
RailLengthBiasSeed (By Size)	0.1 and 0.75	0.1 and 0.5	0.08 and 0.5	0.06 and 0.5
ReverseBendingBiasSeed (By Size)	0.1 and 0.75	0.1 and 0.5	0.1 and 0.5	0.1 and 0.5
RailWebWidthSeed (By #)	3	4	5	6

The density of the mesh surrounding the crack in the rail head increases substantially with each subsequent mesh in the convergence study. The cross-sectional mesh and seeding corresponding to Mesh 1 and Mesh 4 for an unworn rail with a 5% HA crack is shown in Figure 5.16.
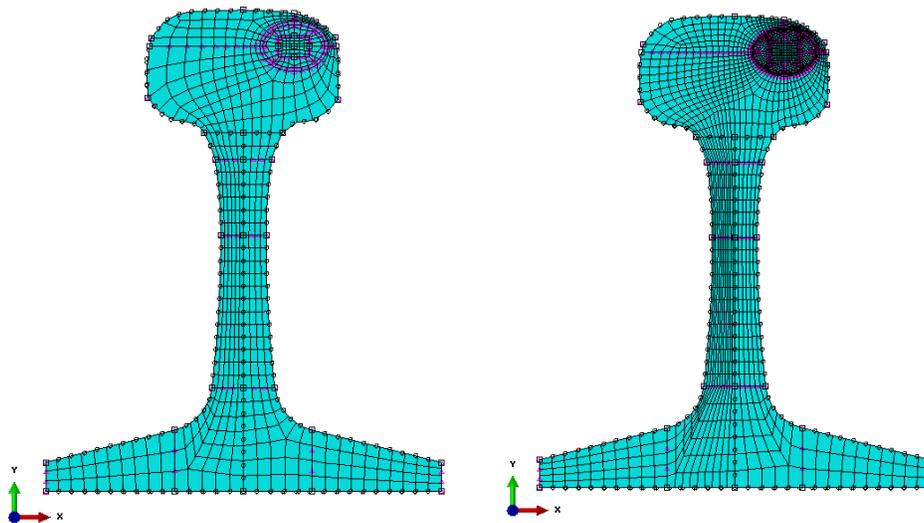
Figure 5.16 Mesh 1 and Mesh 4 mesh densities used in convergence study.

The black circles shown in Figure 5.16 represent the global part seed of 0.2, and the pink triangles represent the edge seeds-by-size and seed-by-number for each mesh. The radial mesh around the crack and small element sizes at the crack edge are developed to capture the high stress and strain gradient at the crack front. Without the radial partitioning scheme, the high gradient at the crack edge would not exist, and a converged solution at the crack edge would not be possible. The meshing along the length of the rail for the 3rd mesh of the unworn rail with a 5% HA crack is shown in Figure 5.17.
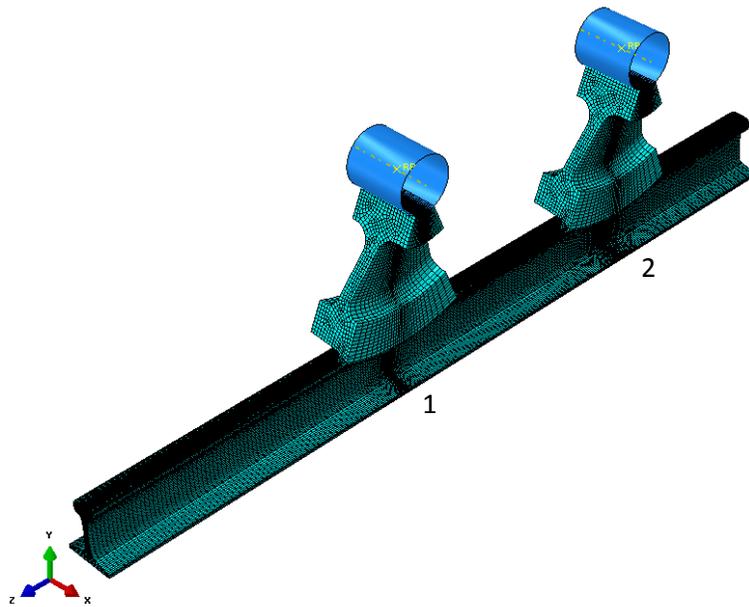
Figure 5.17 Mesh three for the unworn rail assembly.

Figure 5.17 illustrates the biasing of the mesh along the length of the rail in the areas surrounding the crack and at the points of contact between the standard bending and reverse bending wheels and the rail. The area of high mesh density surrounding the crack is labeled as 1 in Figure 5.17. The crack front is the primary point of interest in the model, so the mesh surrounding the crack is refined more than any other point in the model to reach a converged solution at the crack edge. The mesh density at the location of the reverse bending wheel contact is also refined and is labeled as 2 in Figure 5.17. The refined mesh at the reverse bending location maintains that the reverse bending load is transferred into the rail correctly through contact, and that the mechanics of the problem are appropriately represented. The mesh along the length of the rail in sections that are dominated by bending and do not include contact with the wheels maintain a coarser mesh to reduce the number of elements in the model and therefore the run time. As shown in Table 5.4, the density of the mesh in these sections is only refined once during the mesh convergence study. In this way the model captures the full mechanics of the problem without using unnecessary elements

The stress convergence of the model is evaluated at four points around the crack edge in the rail head. These points represent the crack front, and a high stress gradient location. The convergence points are shown in Figure 5.18 for an unworn rail with a 5% HA crack.

63
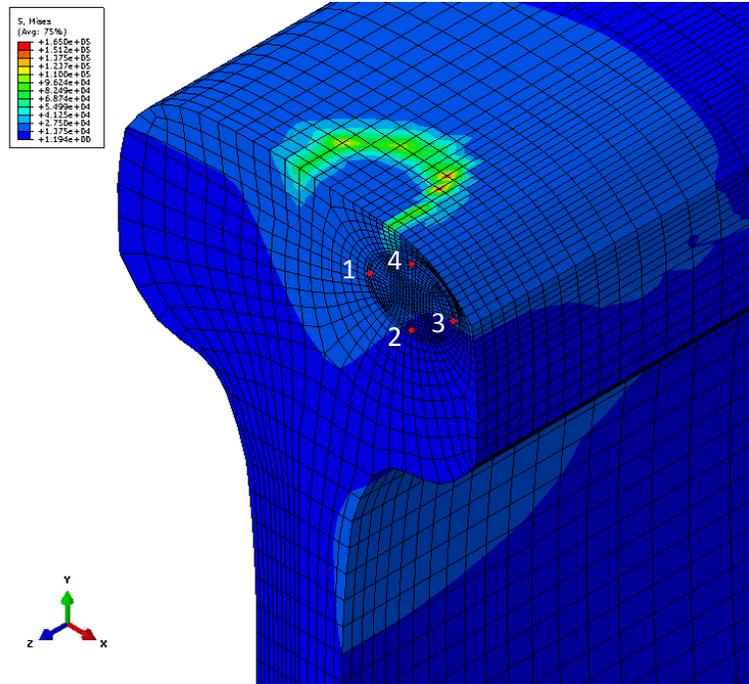
Figure 5.18 Convergence points for FE model.

Convergence points one through four are nodes formed by the intersection of the cross-section surface partitions that define the crack, making them convenient nodes to query stresses and strains. The locations of these nodes remain constant for a model of a given profile despite the changing mesh density in the rail for the four meshes evaluated in the convergence study. Each convergence point represents a point on the crack edge that is easy to query. Convergence point one represents the left crack edge, convergence point two represents the bottom crack edge, convergence point three represents the right crack edge, and convergence point four represents the top crack edge. While the location of the convergence points do not change as the mesh is refined, their location will change as the crack size grows. Therefore, convergence for a model is evaluated for a constant profile geometry and crack size using mesh one through mesh four.

The convergence of many models featuring different worn profiles and different crack sizes is explored to determine the mesh that represents a converged stress solution in the majority of models. The convergence study features four FE models including the unworn rail profile with a 5% HA crack, the unworn rail profile with a 50% crack, the 8/16th side-wear profile with a 5% crack, and the 8/16th side-wear profile with a 20% crack. Each model used in the convergence study features identical mesh definitions for mesh one through mesh four. Therefore, convergence plots are generated for each model using the total number of elements in the model's mesh and the von Mises stress at each of the

convergence points. The von Mises stress is an equivalent representation of the overall state of stress at a given point in the model and is calculated using the equation:

$$\bar{\sigma} = \frac{1}{\sqrt{2}}\sqrt{(\sigma_1 - \sigma_2)^2 + (\sigma_2 - \sigma_3)^2 + (\sigma_3 - \sigma_1)^2}$$ (5.4)

where $\bar{\sigma}$ is the von Mises equivalent stress, σ_1 is the maximum principal stress, σ_2 is the middle principal stress, and σ_3 is the minimum principal stress. Example convergence plots for the 5% HA and 50% HA crack unworn rail models are shown in Figure 5.19
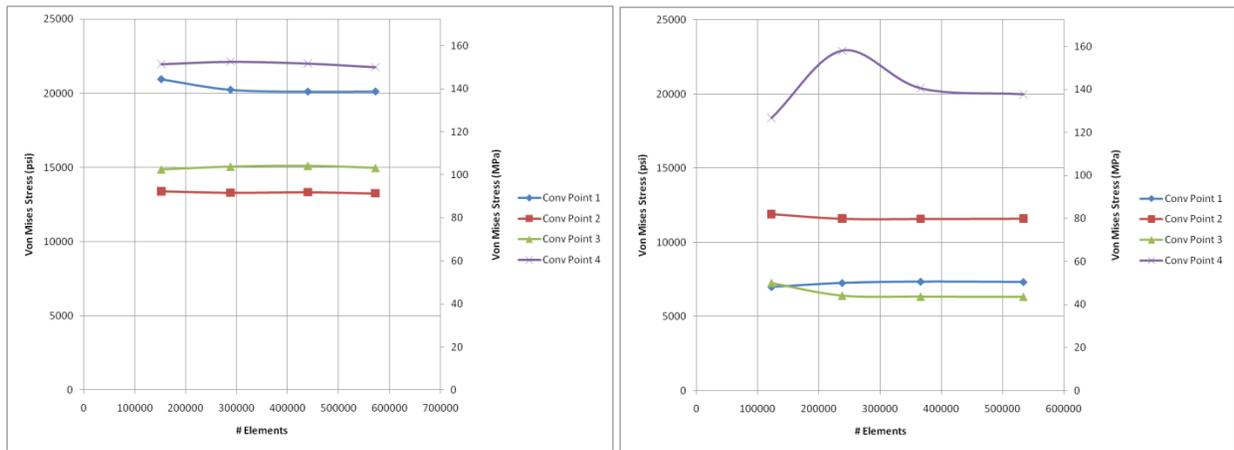
Figure 5.19 Convergence plots for 5% HA and 50% HA flaws in unworn rails.

The convergence plots evaluated as part of the FE model convergence study show oscillatory trends for each convergence point about a converged stress value. This trend is especially evident in the right plot of Figure 5.19, and is a result of the incremental analysis necessary to resolve the contact present in the model. The stress result at a convergence point is considered to be converged when the percent difference in the von Mises stress between the current mesh and the previous mesh is less than 3%. The percent difference is calculated using the equation:

$$\%Diff = \frac{100(\bar{\sigma_2} - \bar{\sigma_1})}{\bar{\sigma_1}}$$ (5.5)

where $\bar{\sigma_1}$ is the von Mises stress for the initial mesh, and $\bar{\sigma_2}$ is the von Mises stress for the new mesh. Table 5.5 and Table 5.6 show the mesh information and percent difference calculations for the convergence of the 5% HA and 50% HA unworn rail models shown in Figure 5.19 at convergence points 1 through 4.

Table 5.5 Convergence table for unworn rail model with 5% HA flaw.

Mesh #	# Elements	Stress @ 1 (psi)	% Diff	Stress @ 2 (psi)	% Diff	Stress @ 3 (psi)	% Diff	Stress @ 4 (psi)	% Diff
1	152532	20957.6	N/A	13398.4	N/A	14868.7	N/A	21975.6	N/A
2	288436	20233.7	-3.454	13307.7	-0.677	15064.1	1.314	22137	0.734
3	440091	20108.1	-0.621	13339.1	0.236	15119.9	0.370	22016.1	-0.546
4	572606	20116.3	0.041	13261.6	-0.581	14977.9	-0.939	21767.9	-1.127

Mesh #	# Elements	Stress @ 1 (MPa)	% Diff	Stress @ 2 (MPa)	% Diff	Stress @ 3 (MPa)	% Diff	Stress @ 4 (MPa)	% Diff
1	152532	144.4975654	N/A	92.37871607	N/A	102.516078	N/A	151.5164283	N/A
2	288436	139.5064506	-3.454	91.75336159	-0.677	103.863313	1.314	152.6292421	0.734
3	440091	138.6404691	-0.621	91.96985697	0.236	104.248041	0.370	151.795666	-0.546
4	572606	138.6970061	0.041	91.43551328	-0.581	103.268985	-0.939	150.0843872	-1.127

Table 5.6 Convergence table for unworn rail model with 50% HA flaw.

Mesh #	# Elements	Stress @ 1 (psi)	% Diff	Stress @ 2 (psi)	% Diff	Stress @ 3 (psi)	% Diff	Stress @ 4 (psi)	% Diff
1	122314	6979.73	N/A	11894.8	N/A	7244.06	N/A	18401.2	N/A
2	237721	7254.57	3.938	11594.8	-2.522	6396.44	-11.701	22918.4	24.548
3	366017	7343.4	1.224	11580.5	-0.123	6327.62	-1.0760	20388	-11.041
4	533900	7323.47	-0.271	11599.1	0.161	6315.38	-0.193	19984.3	-1.980

Mesh #	# Elements	Stress @ 1 (MPa)	% Diff	Stress @ 2 (MPa)	% Diff	Stress @ 3 (MPa)	% Diff	Stress @ 4 (MPa)	% Diff
1	122314	48.1235443	N/A	82.01175901	N/A	49.9460355	N/A	126.8718078	N/A
2	237721	50.01849939	3.938	79.94333183	-2.522	44.1019013	-11.701	158.0168055	24.548
3	366017	50.63096068	1.224	79.8447368	-0.123	43.6274041	-1.0760	140.5703116	-11.041
4	533900	50.49354817	-0.271	79.97297928	0.161	43.5430123	-0.193	137.7868981	-1.980

Each model evaluated in the convergence study reaches convergence relatively quickly—in the second or third mesh—at the four convergence points. It is important to understand that convergence at the point of contact between the rail and the wheel is harder to achieve due to load singularities that cause fictitious high stresses as a result of the discretization of the surface during meshing and the linear element type. In order to converge the contact stresses, extensive mesh refinement would be necessary around the contact point to represent the smooth surfaces that are actually in contact. Because the stress results at this point are not of interest and the primary objective of the contact is to transfer the wheel load into the rail, continued mesh refinement at the point of contact is believed to be unnecessary. Therefore, mesh three is

considered the converged stress solution for every model evaluated for the development of the response surfaces. Additional models of various worn profiles and crack sizes are used to "spot check" the decision to use mesh three as the converged solution. The additional convergence results and plots used in the convergence study are found in Appendix D.2. The results for an Abaqus analysis are saved as a .ODB file. Each .ODB file contains the elastic stress results for each load step applied in the model. The fatigue post-processor uses the stress results in the Abaqus .ODB file to complete the fatigue calculations for the rail.

5.6 Conclusion

The FE model provides the elastic stress results needed for the fatigue life calculations in the fatigue post-processor. In order for the fatigue life to be calculated, the stress state for the maximum and minimum magnitudes for each load must be represented in the output .ODB file. These stress states are represented through the use of three independent load steps: the standard bending load step, the reverse bending load step, and the minimum temperature load step. The vertical and lateral wheel loads are considered coupled loads, and are applied as concentrated forces to the reference point of the rigid axle in the standard and reverse bending load steps. The minimum temperature load is applied as a temperature load to the volume of the rail section in the minimum temperature load step. The maximum temperature represents a zero state of stress, and therefore is not applied to the rail volume.

The geometry of the rail is created using a parametric representation of the rail head. The nodal coordinates that are calculated in Chapter 3 are used to update the geometry of the rail head and upper web for each worn profile that is fitted using the MATLAB program developed in Chapter 3. An elliptical detail fracture is created in the rail head to represent the worst case scenario for the rail's previous load history. A 10% HA flaw is the minimum detectable crack size, and the fatigue life of the rail is evaluated with this crack size present. This fatigue life is the life output variable for the response surfaces. In addition, the crack-growth rate at different crack sizes is of interest. Therefore, four additional crack sizes are created in the rail head. The fatigue life results for these crack sizes and the 10% HA crack size can be used to develop an expression for the crack-growth rate as a function of load cycles and crack size.

Considerable effort went into achieving a converged solution for the FE model. Four mesh definitions with increasing mesh densities in the rail are defined and used for each geometry evaluated. The results from multiple models representing different worn cross-sections and different crack sizes are compounded to form convergence plots. The convergence plots evaluated demonstrate an oscillatory convergence trend for the stress results that reach convergence by the second or third mesh. Therefore,

the third mesh is considered the converged solution for each model, and it used in the development of the response surfaces. The .ODB file containing the results from this mesh are read into the fatigue post-processor and used to develop the current load history for the rail and to calculate its fatigue life.

Chapter 6. Fatigue Calculations with FE-Safe

The output variables of the response surfaces are the remaining cycles to failure or fatigue life for the given rail and the crack-growth rate. The fatigue life of the rail is calculated directly in the fatigue post-processor using the Abaqus elastic stress results for the given rail geometry with a 10% HA crack. The crack-growth rate is calculated using the fatigue lives of the rail at multiple crack sizes calculated in the fatigue post-processor. The fatigue life calculations are explained in Chapter 7. The fatigue post-processor used to calculate the fatigue life of the rail using the Abaqus stress results is FE-Safe [22]. FE-Safe is a fatigue program capable of reading and modifying Abaqus output .ODB files. The elastic nodal stress results saved in the Abaqus .ODB file are used to generate load histories that are used in the fatigue-life calculations. Multiple stress and strain algorithms are available for the fatigue calculations, but Strain-Life Fatigue Theory is used for the analysis of the rail and the development of the response surfaces. Strain-Lased Fatigue Theory is selected because of its ability to represent the localized plastic strain state of the material at the crack front and other geometric stress raisers. The local stress and strain at each node in the model is necessary in order to complete the strain-life fatigue calculations for the rail. Neuber's rule and Ramberg-Osgood's stress-strain law are used to approximate the total stress and strain at each point in the model using the Abaqus elastic stresses. The load history for each node in the model is built using the approximated stresses and strains for each load step.   The development of the material properties needed for the fatigue calculations, the approximation of the plastic stresses and strains using Neuber's rule, the development of the fatigue load history, and the life calculations for the rail are explained in the following chapter.

6.1 Importing Abaqus .ODB File

FE-Safe requires the elastic stresses at each point in the model in order to complete the fatigue life calculations for a FE model. Once the FE interface settings are configured for Abaqus version 6.10-1, the stress results are accessed by loading the Abaqus .ODB file for a given rail model into the FE-Safe Open FE Model database. In addition to the stress results for each load step time increment, FE-Safe imports the surfaces and sets created in the Abaqus model as elemental groups. The FE-Safe default is to analyze every element in the FE model, but the groups can be used to define which elements in the model are analyzed. In order to decrease the number of elements for which the fatigue life is calculated, a fatigue group is created as an element set in Abaqus. This fatigue group contains the elements that make up the rail head in at the crack front. Once the FE model's elastic stress results are imported into FE-Safe, the analysis group is limited to only the elements in the fatigue group. The fatigue group that becomes the FE-Safe analysis group is shown in Figure 6.1 for the unworn rail profile.
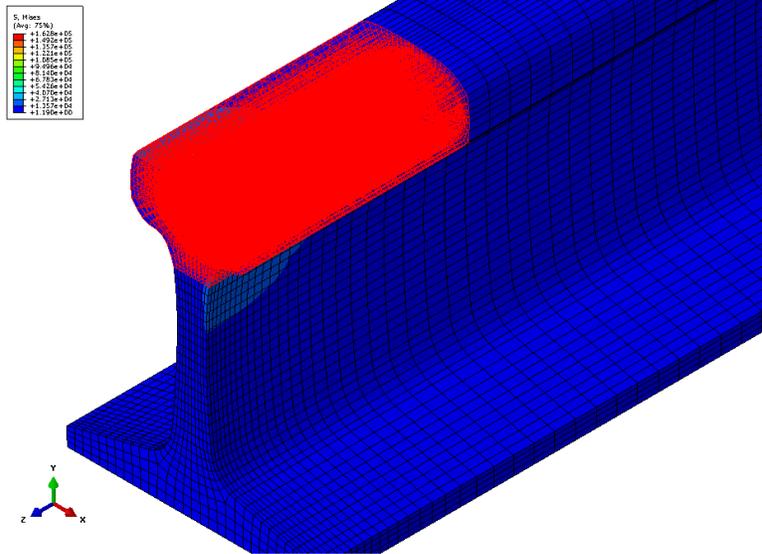
Figure 6.1 FE-Safe elemental analysis groups.

The reduction of the number of elements analyzed in FE-Safe greatly reduces the computational time necessary to calculate the rail's fatigue life. The nodal stresses of the elements that make up the fatigue analysis group are used to generate the load history for each node in the group that is used in the fatigue calculations.

6.2 Strain Development and Fatigue Theory

The total stress and strain at every point in the FE model is needed to calculate the fatigue life of the rail using Strain-Life Fatigue Theory. The total strain is defined as the sum of the plastic and elastic strains, and is given in the equation:

$$\varepsilon_t = \varepsilon_e + \varepsilon_p \tag{6.1}$$

where ε_t is the total strain, ε_e is the elastic portion of the strain, and ε_p is the plastic portion of the strain. The total stress and strain can be generated using an elastic-plastic analysis in Abaqus, but the computational time required to generate the number of sample results necessary to build the response surfaces could be excessive. It is an accepted practice to approximate the total local stress and strain for the model using the elastic stress results from Abaqus and Neuber's rule to approximate the elastic-plastic response of the material.

During elastic deformation, both the local stress concentration factor and the local strain concentration factor are equal to the theoretical stress concentration factor. This is shown in the equation:

$$k_t = k_\sigma = k_\varepsilon \tag{6.2}$$

where k_t is the theoretical stress concentration factor, k_σ is the local stress concentration factor equal to the local stress divided by the nominal stress, and k_ε is the local strain concentration factor equal to the local strain divided by the nominal strain. After localized yielding occurs, the local stress concentration factor begins to decrease, and the local strain concentration factor begins to increase. Neuber's rule states that after localized yielding has occurred, the theoretical stress concentration factor can be approximated by the geometric mean of the local stress and strain concentration factors as given in the equation:

$$k_t^2 = k_\sigma k_\varepsilon \tag{6.3}$$

This equation can be rewritten in terms of nominal and local stresses and strains in the form:

$$k_t^2 = \left(\frac{\sigma}{S}\right)\left(\frac{\varepsilon}{e}\right) \tag{6.4}$$

where ε is the total local strain, σ is the total local stress, S is the nominal stress, and e is the nominal strain [21]. The total stress and strain in the model can be approximated from the elastic stress results by combining Neuber's rule with an elastic-plastic stress-strain law. FE-Safe uses Ramberg-Osgood's stress-strain law, given by the equation:

$$\varepsilon = \frac{\sigma}{E} + \left(\frac{\sigma}{H'}\right)^{\frac{1}{n'}} \tag{6.5}$$

where E is the elastic modulus of the material, H' is the cyclic strength coefficient for the material, and n' is the cyclic strain hardening exponent of the material [6,22]. The elastic stress-strain equation, $e = S/E$, Equation 6.4, and Equation 6.5 can be combined to form an equation for localized yielding. After simplifying, this equation becomes [6]:

$$k_t S = \sqrt{\sigma^2 + \sigma E \left(\frac{\sigma}{H'}\right)^{\frac{1}{n'}}} \tag{6.6}$$

The elastic stress in the FE model is equivalent to the left hand side of Equation 6.6—the theoretical stress concentration factor multiplied by the nominal stress. The total local stress is calculated by solving Equation 6.6 for σ using an iterative solution. The total local strain is calculated by plugging this σ value into Equation 6.5. This method is used by FE-Safe to generate the total stresses and strains at each node in the elemental analysis group.

In order to validate the approach of approximating the total local stress and strain in the model using Neuber's rule and the elastic Abaqus stress results, an elastic-plastic version of the rail model is created.

This model utilizes the Abaqus deformation plasticity material model characterized by the Ramberg-Osgood equation in the form:

$$\varepsilon = \frac{\sigma}{E} + \alpha \frac{\sigma}{E} \left(\frac{|\sigma|}{\sigma_o}\right)^{n-1} \qquad (6.7)$$

where α is the yield offset coefficient, σ_o is the material's yield strength, and n is the cyclic hardening exponent [23]. In order to account for potential large wheel loads and increased plasticity in the rail head, the maximum vertical load applied to the standard and reverse bending wheel instances is increased to 80,000 lbf (355,857.6 N). The constants H' and n' developed in Equation 6.5 can be related to the constants of Equation 6.7 using the equations:

$$H' = \frac{E}{\left[\alpha\left(\frac{\sigma_o}{E}\right)^{1-n}\right]^{\frac{1}{n}}} \qquad (6.8)$$

$$n' = \frac{1}{n} \qquad (6.9)$$

Assuming a 0.2% strain yield offset, the value of α can be calculated using the equation:

$$\alpha = 0.002 \left(\frac{E}{\sigma_o}\right) \qquad (6.10)$$

The values of the constants used in the elastic-plastic analysis characterize the material properties of standard rail and are σ_o = 70 ksi (482.63 MPa), E = 29700 ksi (204774.34 MPa), α = 0.84857, and n = 6.667. The stress-strain curves for the elastic material model and the elastic-plastic material model given by Equation 6.7 are shown in Figure 6.2.
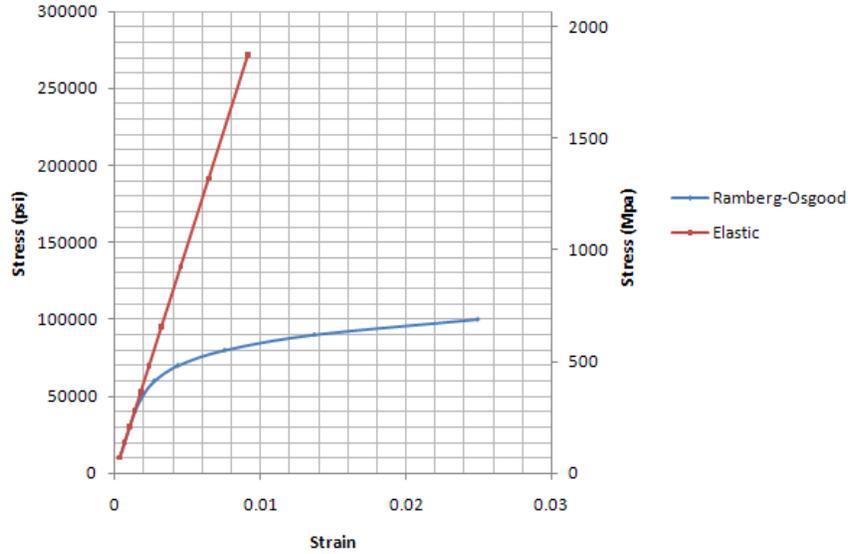
Figure 6.2 Elastic and elastic-plastic stress-strain curves for standard rail.

As suggested in Neuber's rule, and shown in Figure 6.2, the elastic stress results are increasingly higher than the elastic-plastic stress results after yielding occurs, and the elastic strain results are increasingly lower than the elastic-plastic strain results.

The total von Mises equivalent stress and the plastic strain are queried at each convergence point around the crack as defined in Chapter 5. The elastic von Mises stress developed in the elastic Abaqus model is also queried at these points. Using Equation 6.6 and the elastic results, the total local stress at each convergence point is approximated. The second term of Equation 6.5 is used to approximate the local plastic strain at each convergence point. The percent difference between the total stress developed in the elastic-plastic analysis and the total stress approximated with Equation 6.6 is calculated. The percent differences between the elastic-plastic plastic strain and the approximated plastic strain are also calculated. These results are shown in Table 6.1.

Table 6.1 Elastic-plastic and Neuber's rule approximation results.

| Conv. Points | Elastic-plastic | | Elastic | Neuber's Rule | | Percent Difference | |
	σ (psi)	ε_p	$\sigma = k_t S$ (psi)	σ (psi)	ε_p	σ (%)	ε_p (%)
1	30967	7.6e-6	30131.6	30026.55	7.1e-6	-3.04	-6.76
2	20097.6	0.6e-6	20405	20397.01	0.5e-6	1.49	-7.11
3	23112.1	1.2e-6	23026.1	23008.27	1.2e-6	-0.45	1.09
4	33981.1	15.9e-6	33821	33752.37	15.5e-6	-0.67	-2.58

73

	Elastic-plastic		Elastic $\sigma = k_t S$	Neuber's Rule		Percent Difference	
Conv. Points	σ (MPa)	ε_p	(MPa)	σ (MPa)	ε_p	σ (%)	ε_p (%)
1	213.51	7.6e-6	207.75	207.03	7.1e-6	-3.04	-6.76
2	138.57	0.6e-6	140.69	140.63	0.5e-6	1.49	-7.11
3	159.35	1.2e-6	158.76	158.67	1.2e-6	-0.45	1.09
4	234.29	15.9e-6	233.19	232.71	15.5e-6	-0.67	-2.58

The similarities between the elastic-plastic and elastic stress results, and the low magnitude of plastic strain present in this elastic-plastic model at the four convergence points suggest that there is little non-linear material behavior present at the crack front. The results in Table 6.1 are consistent with low-strain theory and suggests that Neuber's rule is a good approximation for the total local stresses and strains in this rail model. In addition, the stresses and plastic strains developed in the elastic-plastic analysis approximated from the elastic results using Neuber's rule also show good agreement with calculated percent differences less than 5% for the stress approximation and 8% for the strain approximation. It is concluded that Neuber's rule provides a good approximation of the total local stresses and strains present in this model, and will be used in the development of the response surfaces.

The fatigue life of the rail can be calculated using multiple strain life equations. FE-Safe offers several uniaxial and biaxial formulations. The Coffin-Manson relationship describes the uniaxial strain-life of an engineering material assuming mean stress is equal to zero, and is given by the equation:

$$\varepsilon_a = \frac{\sigma_f'}{E}\left(2N_f\right)^b + \varepsilon_f'\left(2N_f\right)^c \tag{6.11}$$

where ε_a is the strain amplitude for the given load cycle, N_f is the fatigue life, and σ_f', ε_f', b, and c are strain-life constants for the given material [6]. Morrow's strain-life equation corrects for the effect of a non-zero mean stress by substituting the quantity N^* for N_f in Equation 6.11. N^* is given by the equation:

$$N^* = N_f\left(1 - \frac{\sigma_m}{\sigma_f'}\right)^{\frac{1}{b}} \tag{6.12}$$

where σ_m is the mean stress [6]. Equation 6.11 becomes Morrow's strain-life equation given as:

$$\varepsilon_a = \frac{\sigma_f'}{E}(2N^*)^b + \varepsilon_f'(2N^*)^c \tag{6.13}$$

The Smith-Watson-Topper relationship offers another correction for non-zero mean stress, and is given by the equation:

$$\sigma_{max}\varepsilon_a = \frac{(\sigma_f')^2}{E}(2N_f)^{2b} + \sigma_f'\varepsilon_f'(2N_f)^{b+c} \tag{6.14}$$

where σ_{max} is the maximum local stress [6]. The strain-life fatigue formulation selected to calculate the fatigue life of the rail is the biaxial Brown-Miller-Morrow strain-life equation. The cyclic Brown-Miller strain-life equation is given by Equation 6.15.

$$\frac{\Delta\gamma_{MAX}}{2} + \frac{\Delta\varepsilon_n}{2} = 1.65\frac{\sigma_f'}{E}(2N_f)^b + 1.75\varepsilon_f'(2N_f)^c \tag{6.15}$$

where γ_{MAX} is the maximum shear strain, and ε_n is the maximum strain normal to the shear strain [26]. This equation is corrected for zero mean stress by substituting the quantity N^* defined in Equation 6.12 for N_f in Equation 6.15. The strain-life equation given by Equation 6.15 is the FE-Safe default algorithm for ductile steels. Both Morrow and Smith-Watson-Topper are used to verify the fatigue life results calculated in FE-Safe through the use of hand calculations and simplified example problems. A specific classical problem that is used to verify the techniques used in FE-Safe to approximate fatigue life is the problem of a thin rectangular plate with a circular through hole. Approximations using the uniaxial Morrow mean-stress equation given in Equation 6.13 and the biaxial Brown-Miller-Morrow algorithm used in FE-Safe have percent difference less than 20%.

6.3 Material Property Definitions

As stated in Section 5.3, the material properties necessary for the elastic analysis of the rail (E, v, and α) are considered constant for the different rail steels used in the railroad industry. However, the Ramberg-Osgood material constants (H' and n') and the strain-life fatigue constants (σ_f', ε_f', b, and c) cannot be treated as constant for the different rail steels. These material properties are not readily available or known for the rail materials currently produced and in use. In order to complete the fatigue calculations for the rail, these constants must be approximated for each rail material evaluated in the response surfaces. FE-Safe offers a material approximation tool that uses Seeger's method to estimate the desired material constants. Seeger's method is a published empirical relationship for plain-carbon, low, and medium-alloy steels that estimates the Ramberg-Osgood and strain-life constants using the material's ultimate tensile strength and elastic modulus [25]. The relationships for the six material constants are given in Table 6.2.

Table 6.2 Seeger approximation equations for fatigue material properties.

σ_f'	ε_f'	b	c	n'	H'
$1.5\sigma_{ut}$	$0.59a$	-0.087	-0.58	0.15	$1.65\sigma_{ut}$

The variable σ_{ut} in Table 6.2 is the material's ultimate tensile strength, and a is defined in the equation [25]:

$$a = \begin{cases} 1.00 & for \ \frac{\sigma_{ut}}{E} < 0.003 \\ 1.375 - \frac{125\sigma_{ut}}{E} & for \ \frac{\sigma_{ut}}{E} \geq 0.003 \end{cases} \qquad (6.16)$$

This method is used to generate the material properties required in the FE-Safe fatigue calculations for each rail material of interest using only the material's ultimate tensile strength and elastic modulus. The constant elastic modulus value defined in Section 5.3 for the FE model is used again in the approximation of the six fatigue material constants. Therefore, the six material constants needed to represent the fatigue properties of the rail can be represented by a single variable—the ultimate tensile strength of the material. The ultimate tensile strength is well defined for the rail materials currently in use in the rail industry, and is known for each section or track. Therefore, the ultimate tensile strength of the rail material is the only independent material variable, and is the eighth and final independent variable for the response surfaces. This variable is developed from historical and manufacturer material data for the rails currently in service. Once the Ramberg-Osgood and strain-life fatigue constants are approximated for the given rail material using the material's elastic modulus and ultimate tensile strength, the hysteresis stress-strain loops for initial and cyclic loading and the strain-life curve can be plotted. Figure 6.3 illustrates example curves for a given steel.
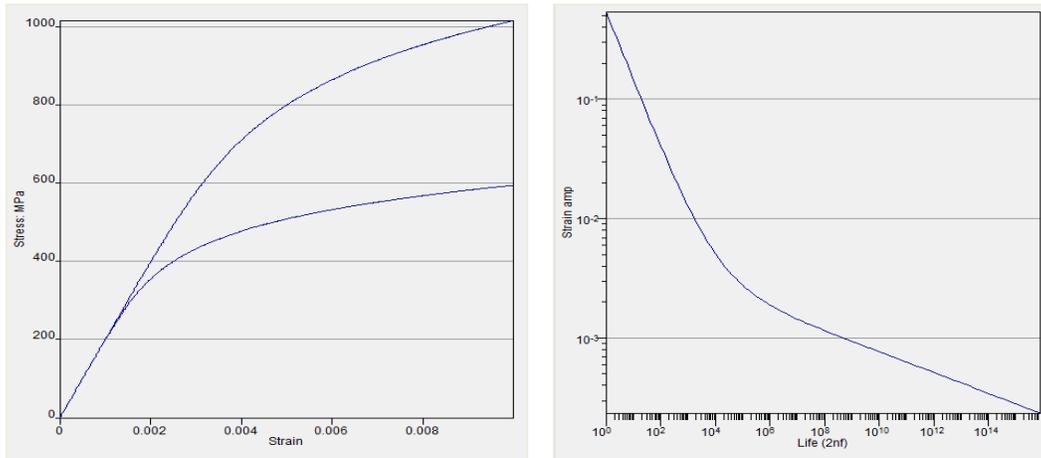
Figure 6.3 Initial and cyclic stress-strain curve and strain-life curve for an example steel.

These plots represent the cyclic and fatigue constants derived using Seeger's rule. If experimental material curves are available for the given rail material, these plots can be used to verify the approximated constants from Seeger's method.

6.4 Load History Development

The load history for the fatigue calculation is developed from the stress states corresponding to the three load steps in the FE model. Each stress state represents the maximum or minimum magnitude for either the wheel loads or the temperature load, and the overall load history can be built by combining and scaling these stress states. The three stress states imported into FE-Safe are 1) the standard bending stress state, 2) the reverse bending stress state, and 3) the minimum temperature stress state. The history developed in FE-Safe represents the loads applied during a 24-hour period of operation. The temperature cycles between the minimum temperature and the maximum temperature once in the 24 hour time period. This is represented as a sine curve and is shown in Figure 6.4.
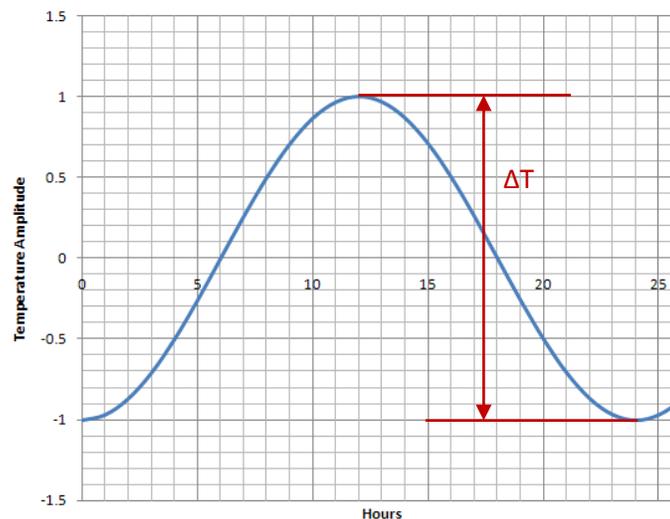
Figure 6.4 Temperature load history.

The maximum value of the temperature load history has a temperature amplitude value of 1 and corresponds to the maximum temperature that the rail experiences. As stated in Chapters 4 and 5, this temperature is typically the rail neutral temperature and is the stress-free reference temperature. The minimum value of the temperature load history has a temperature amplitude of -1 and corresponds to a temperature value equal to $-\Delta T$. The stress state generated by this temperature load is developed in the FE model during the minimum temperature load step.

Using operation records from a major American railroad company, the average number of trains at night and the average number of trains during the day can be estimated. The average number of axles per train can also be estimated. The frequency estimates for the loads used in the load history derivation are developed from track data. Geometry and failure data from a section of track will be used in the

validation stage of the project to verify the results of the response surfaces. The frequency data for the loads is shown in Table 6.3

Table 6.3 Frequency data for the loads in the fatigue load history [2].

Avg. # Day Trains	Avg. # Night Trains	Avg. Locomotive Axle Count	Avg. Car Axle Count
17	18	16	295

A train-load block, representing the number of wheel-load applications applied by a single train, is formed by summing 311 (equal to the locomotive axle count plus the car axle count for a train) alternating applications of the standard-bending stress state and the reverse-bending stress state developed in the FE model. An additional overload cycle is applied at the end of the train load history to represent an increased dynamic load that occurs once per train. This increased dynamic load can be caused by wheel deformations such as a flat, and accounts for the long tail of the second mode of the peak load distribution examined in Section 4.1. Data from WILD sites characterize 80,000 lb. wheel loads (355.86 kN) as occurring once per train [2]. Using the conservative maximum vertical load of 57,000 lbs. (253.55 kN) from the example WILD distribution in Section 4.1, the overload factor can be estimated as 1.4. This factor is adopted as the constant overload factor in the development of the train load history for the estimation of the fatigue life of the rail. An example of the train load history for a reduced number of axles is shown in Figure 6.5.
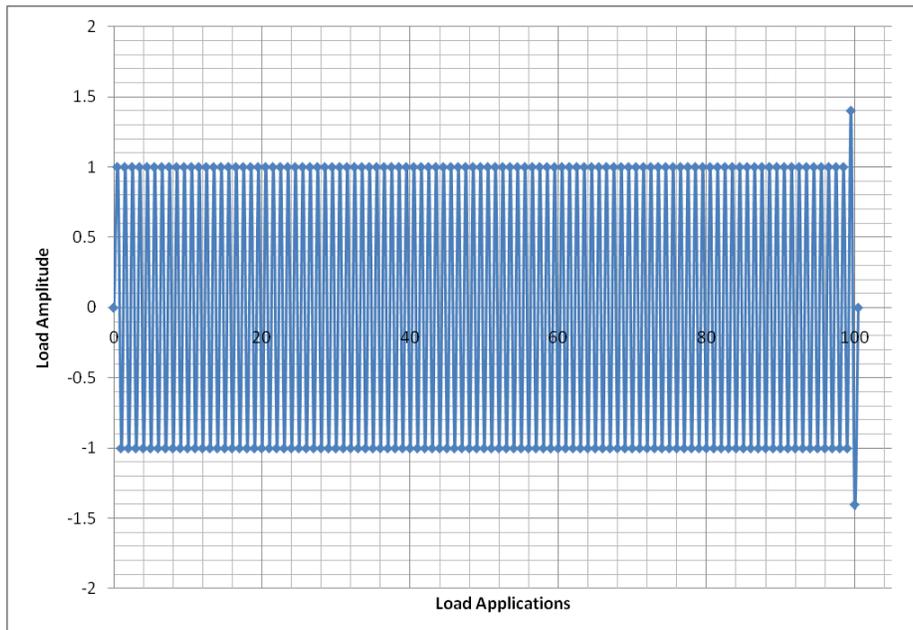
Figure 6.5 Train load history.

In Figure 6.5, there are 100 load cycles that oscillate between a load amplitude equal to 1 and a load amplitude equal to -1. The load amplitude equal to 1 corresponds to the stress state in the rail produced by the application of the standard- bending wheel load in the standard-bending load step. The load amplitude equal to -1 corresponds to the stress state generated by the application of the reverse-bending wheel load in the reverse-bending load step. There is one overload cycle in Figure 6.5 that has a maximum load amplitude equal to 1.4 and a minimum load amplitude equal to -1.4. The maximum load amplitude of the overload cycle corresponds to the state of stress developed in the standard-bending load step scaled by a factor of 1.4. The minimum load amplitude of the overload cycle corresponds to the reverse-bending stress state scaled by a factor of 1.4. The ability to scale the stress states generated in the load steps of the FE model is a direct result of the elastic analysis completed using a linear material law. Therefore, the elastic stress state is scaled before Neuber's rule is used to generate the total local stresses and strains present in the model. If elastic-plastic results are imported from the FE model, the ability to scale the stress results would disappear.

The overall load history for a 24-hour time period is generated by superimposing the temperature-load-history block and the train-load-history block on top of each other. Using the train frequency and axle count estimates from Table 6.3, the number of axles applied during the day and the night can be calculated. Four independent stress states are imported into FE-Safe from the Abaqus .ODB file. Stress state one (σ_1) is the stress state at the beginning (first increment) of the first load step prior to the application of loads. This stress state represents the zero stress state generated by the maximum temperature load. Stress state two (σ_2) is the stress state at the end (last increment) of the standard-bending load step, and therefore represents the stress state generated by the application of the standard-bending wheel load. Stress state three (σ_3) is the stress state at the end of the reverse-bending load step, and represents the stress generated by the application of the reverse-bending wheel load. Stress state four (σ_4) is the final stress state, and represents the stress generated in the minimum-temperature load step by the application of the minimum-temperature load. Figure 6.6 illustrates the combination of the four stress states to generate the 24-hour load history.
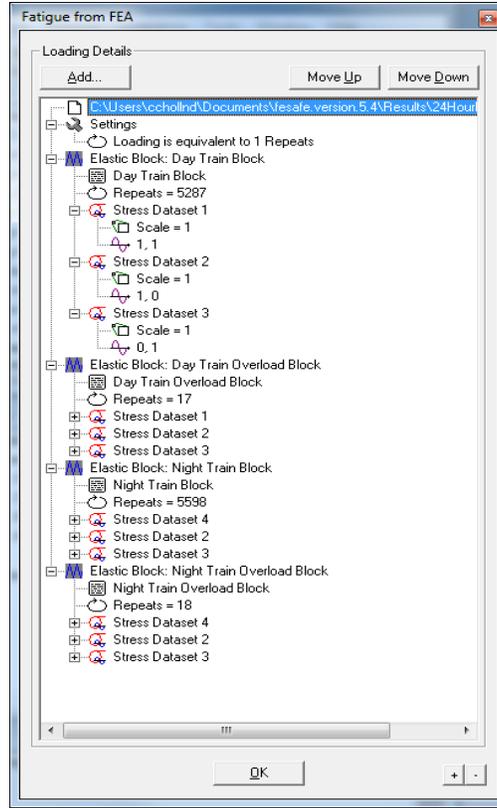
Figure 6.6 24-hour load history generate in FE-Safe using Abaqus stress results.

Four elastic blocks are created in the load history definition file (.LDF) to represent the overall load history for the 24-hour period. The "Day Train" elastic block combines the constant maximum temperature stress dataset (σ_1) with the alternating standard bending and reverse bending stress datasets (σ_2 and σ_3) by summing the stress dataset amplitude curves. The amplitude curve for σ_1 is defined as (1,1) so that the stress stays constant during the load block. The amplitude curves for σ_2 and σ_3 are (1,0) and (0,1) respectively to create alternating sine curves. When combined, the σ_2 and σ_3 stress dataset amplitude curves create one sine curve that alternates between σ_2 and σ_3, and the σ_1 stress dataset constant amplitude curve elevates the overall level of stress. There are 5287 repetitions of the "Day Train" elastic block to represent the total number of wheel loads applied during the day. The total number of wheel loads is developed from the data in Table 6.3. In addition, each stress dataset has a scale factor that can be used to vary the magnitude of the load for the generation of the response surfaces. The "Day Train" σ_2 and σ_3 scale factors for a given rail profile are calculated using the Abaqus vertical wheel load of 45,000 lbs. (200.17 kN) and the maximum vertical wheel load calculated from the WILD distribution as shown in Section 4.1. Both scale factors are identical and are given by the equation:

$$Scale\ Factor = \frac{P_{MAX}}{45,000 lbs} = \frac{P_{MAX}}{200.17\ kN} \tag{6.17}$$

For the 57,000 lb. (253.55 kN) maximum vertical load from the example WILD distribution fit in Section 4.1, the "Day Train" σ_2 and σ_3 scale factor is 1.27.

The definition for the "Day Train Overload" elastic block combines stress datasets σ_1, σ_2 and σ_3 using identical amplitude curves as the ones used in "Day Train" elastic block. The only differences between the two elastic load blocks are the number of block repetitions and the scale factors for the stress datasets. According to Table 6.3, there are 17 trains that occur during the day. Therefore, there are 17 repeats of the "Day Train Overload" elastic block representing a single application of the overload once per train. In addition, the scale factors for stress datasets σ_2 and σ_3 are set at a factor of 1.78. The "Day Train Overload" scale factors are calculated using an equation similar to Equation 6.17. The overload maximum vertical wheel load is calculated by multiplying the maximum vertical wheel load from the WILD distribution fit by the constant overload factor of 1.4. Then the "Day Train Overload" scale factor for the load history in FE-Safe is calculated by dividing the overload maximum vertical wheel load by the 45,000 lb. (200.17 kN) Abaqus load. This scale factor is multiplied by the two stress datasets (the standard bending wheel load and the reverse bending wheel load stress states) to create the overload wheel load stress state. The "Night Train" and "Night Train Overload" elastic load blocks are created by copying the "Day Train" and "Day Train Overload" elastic blocks and replacing the σ_1 stress dataset with σ_4. The σ_4 stress dataset is the minimum temperature stress state that corresponds to the lowest temperature reached during the night. Therefore, the alternating σ_2 and σ_3 stress datasets are now combined with the minimum temperature stress state. The number of repetitions for the "Night Train" elastic block is 5598 representing the total number of wheel loads applied during the night. The "Night Train Overload" elastic block is repeated 18 times for one overload per night train. The wheel load applications are summarized in Table 6.4

Table 6.4 Wheel load applications for daily load history.

Load	Day Load	Day Overload	Night Load	Night Overload
Repetitions	5287	17	5598	18
Scale Factor	1.27	1.78	1.27	1.78

6.5 Life Estimation Results

The results from the fatigue analysis are used to provide an estimation of the rail's remaining life and the crack-growth-rate equation for the rail given the current worn geometry of the rail, the load inputs, and rail material properties. The output of the fatigue analysis is a contour plot of the life of the rail with values equal to the base-10 logarithm of the life. This contour plot is added to the Abaqus .ODB file as

an additional step named fe-safe_01 at the end of the analysis with the output variable LOGLife-Repeats. The life at any point in the analysis group, as defined in Section 6.1, can be found by querying this variable at the node of interest. An example of a life contour plot is shown in Figure 6.7.
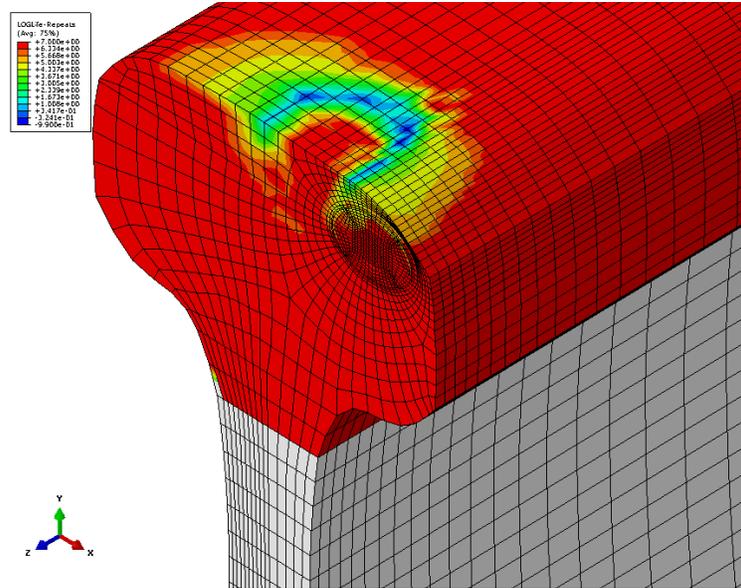
Figure 6.7 Life contour for the unworn rail with a 5% HA crack analysis group.

The color contour for the life of the rail shown in Figure 6.7 is reversed from the color conventions used in standard finite element contour plots. In the life contour plot, red colors indicate infinite life— approximately 10^7 cycles—and blue colors indicate the critical points in the model with the lowest life. Figure 6.7 demonstrates that the lowest life values are located at the contact points between the wheel and the rail. This is expected due to the load singularities and therefore artificially high contact stresses at these points due to the mesh selected. In addition, the deformation at the points of contact is compressive in nature. For these reasons, a crack is not likely to initiate and propagate at the contact locations. If a mesh was developed to converge the stress results at the points of contact between the rail and the wheel, the life results at the point of contact would be valid. The elliptical crack developed in the FE model represents a critical location in the rail head where damage has already initiated. In order for the crack to continue to propagate and eventually cause a rail failure, the critical location for the fatigue life of the rail must be located on the crack front. Therefore, the life is queried at every node along the elliptical crack edge. The node with the minimum-average nodal life value is determined as the propagation site for the crack, and the life at that node is specified as the remaining life of the rail. This life estimation is

82

completed for each worn geometry, crack size, load input, and material ultimate tensile strength evaluated in the response surfaces. The remaining life of the rail is estimated using the 10% HA crack-life results for each model, and the crack-growth rate is estimated using this life estimation and the life estimations for four additional crack sizes represented in the given model.

6.6 Conclusion

The analysis completed in the fatigue post-processor provides the life estimates necessary for the development of the response surface output variables. These response outputs are the remaining life of the rail (fatigue life) and the crack-growth rate. The fatigue life of the rail is estimated directly from the fatigue analysis output for the rail profile with a 10% HA crack present. The post-processing of the crack-growth rate for a given profile, load inputs, and material ultimate tensile strength is examined in the following chapter.

The fatigue analysis is completed using the total local stress and strain for each element in the model. The total stress and strain in the model is approximated from the purely elastic stress results developed in the finite element analysis. This approximation uses Neuber's rule and Ramberg-Osgood's stress-strain law to predict the total stress and strain from the elastic stress results. The material properties necessary for the fatigue analysis and the development of the total stress and strain in the model are approximated using Seeger's method, the elastic modulus of the material, and the material ultimate tensile strength. The fatigue analysis is limited to the rail-head volume surrounding the crack front using sets developed in the FE model and imported into FE-Safe. By decreasing the elements in the analysis group, the run-time is greatly reduced without sacrificing accuracy of results.

The load history of the rail is constructed using the stress results from the three load steps in the FE model and the load frequency data shown in Table 6.3. The independent stress states generated in the FE model are scaled and combined to form a load history block that represents the thermal and train loads applied during a 24-hour period. The remaining life of the rail that is developed in the fatigue analysis is presented as the number of cycles until failure occurs where a cycle is defined as a one repetition of the 24-hour load block. Therefore, the life estimate of the rail is in days. The life results for each model are added to the Abaqus .ODB file as an additional load step. This makes it easy to access the life results for the development of the response surfaces and the post-processing calculations of the crack-growth rate.

Chapter 7. Development of the Crack-Growth Rate

The two outputs of the response surfaces are the remaining fatigue life of the rail and the crack-growth rate. The fatigue life of the rail is estimated for a profile with a 10% HA crack present in the rail head. This crack size represents the "worst case" load history because a 10% HA crack is the minimum detectable crack size for the Ultrasonic Test car. The previous chapter explains how the fatigue life is calculated for a rail with the eight independent variables defined. The calculation of the rail's remaining life using the 10% HA crack and FE-Safe provides a single estimate for the rail's remaining fatigue life. It is also important to understand how quickly the rail's structural integrity is decreasing. This information is provided through the estimation of the crack's growth rate. Therefore, the fracture mechanics stage of the project focuses on the development of the second response output variable—the crack-growth-rate equation. This variable is determined for given rail profile using the fatigue life estimates developed in FE-Safe for multiple crack sizes modeled in the rail head.

As stated in previous chapters, five crack sizes are evaluated in each worn rail profile. The five crack sizes are used to generate a crack size vs. load cycles to failure plot using the fatigue life of each crack generated in FE-Safe and the crack semi-major axis dimension, a. This plot has five data points representing the five crack sizes. Using specific assumptions that will be justified in this chapter, this plot can be transformed to a crack size vs. applied load cycles plot for the given worn profile. The five data points can be fit with an exponential curve. The equation of this line represents the crack size as a function of applied load cycles. The derivative of this line (da/dN) is the change in crack size as a function of the change in load cycles and therefore the crack-growth-rate equation. This equation can be used to predict the crack-growth rate for any number of cycles or crack sizes. The development of the crack size vs. load cycles to failure plot, the crack size vs. applied load cycles plot, and the equation for the crack-growth rate is explained in the following chapter.

7.1 Crack-Growth Rate

The propagation of a crack can be illustrated using a plot of the crack dimension a versus applied load cycles. This plot is typically characterized by an exponentially increasing crack-growth curve, and the instantaneous slope of the curve is defined as the crack-growth rate. An example of a crack-growth curve is shown in Figure 7.1
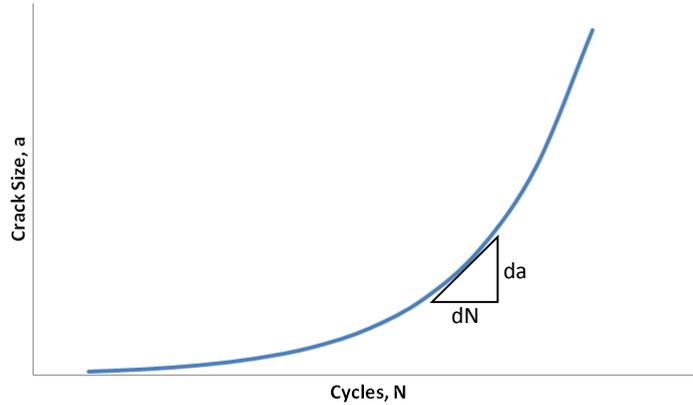
Figure 7.1 Exponential crack-growth curve.

The crack-growth rate is typically used to describe the propagation behavior of the crack. The crack-growth rate can be related to the stress intensity range using the Paris equation:

$$\frac{da}{dN} = C(\Delta K)^m \tag{7.1}$$

where $\frac{da}{dN}$ is the crack-growth rate and C and m are constants developed from fitting the linear portion of experimental crack-growth rate vs. stress intensity range data for a given material on a log-log plot [6]. The constants developed in Equation 7.1 are not independent of the stress ratio R equal to the minimum nominal stress divided by the maximum nominal stress. Therefore, when the crack-growth rate and stress intensity range data are plotted for various stress ratios, it forms multiple curves on a log-log plot. Multiple equations have been developed that attempt to characterize the relationship between the crack-growth rate and the stress ratio. The most common is given by Walker in the equation:

$$\frac{da}{dN} = C_0(\overline{\Delta K})^m \tag{7.2}$$

where C_0 is the value of C corresponding to $R=0$, and $\overline{\Delta K}$ is the equivalent $R=0$ stress intensity that causes the same growth rate as the original stress intensity range [6]. The value of $\overline{\Delta K}$ is given by the equation:

$$\overline{\Delta K} = \frac{\Delta K}{(1-R)^{1-\gamma}} \tag{7.3}$$

where γ is a material constant [6]. When plotted, crack-growth rate vs. $\overline{\Delta K}$ data for various stress ratios fall on a single curve.

85

After the critical crack size and the initial crack size are estimated for a given component, life estimations can be completed using the previous equations. The number of cycles to grow a crack from the initial size to the critical size is given by the equation:

$$N_{if} = \int_{a_i}^{a_f} \left(\frac{dN}{da}\right) da \qquad (7.4)$$

where N_{if} is the number of cycles to grow the crack from the initial crack size to the final crack size and a_i and a_f are the initial and final crack sizes [6]. The critical crack size is determined by plugging the material fracture toughness in for K in Equation 2.1 and solving for a. Numerical integration and incremental solutions using small increases in crack size are typically necessary for solving Equation 7.4 for N_{if}.

7.2 Plotting Load Cycles to Failure versus Crack Size

The first step in the development of the crack-growth-rate equation is to complete the fatigue analysis for the five crack sizes modeled in each worn profile. For each crack size evaluated, the semi-major axis dimension, a, is known, and the fatigue life is calculated. Once the fatigue analyses has been completed for the five crack sizes modeled in a given worn profile, the crack dimension, a, can be plotted as a function of the rail's remaining fatigue life in days. Each plot generated includes five data points corresponding to the five crack sizes. A crack size vs. load cycles to failure plot is shown in Figure 7.2 for an example side-wear profile with load cycles to failure represented in terms of days.
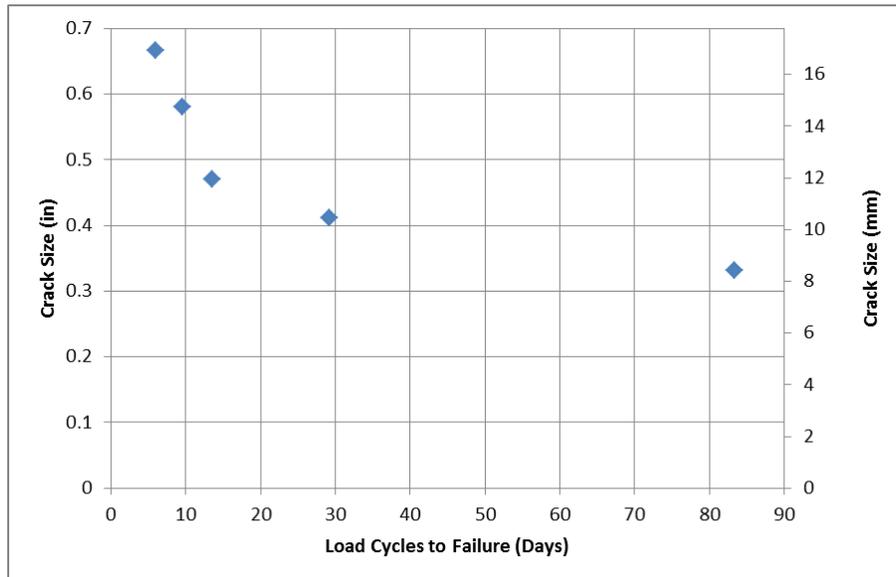
Figure 7.2 Crack size as a function of load cycles to failure for an example side-wear profile.

Figure 7.2 illustrates an exponentially decreasing trend that predicts that as the crack size decreases, the fatigue life of the rail increases. This trend matches our expectations for the rail. Assuming a similar form to Equation 7.1, a minimum of two data points are required to fit the exponential curve. By using five crack sizes, the method for calculating the crack-growth rate is over determined. Therefore, an exponential fit of the data is developed using five data points instead of two, and each data point is a smaller fraction of the total data set, providing a more accurate solution. This method captures the general exponential trend of the date and accounts for some of the scatter seen in the fatigue life results for the various crack sizes. This scatter can be attributed to mesh sensitivity or modeling techniques, and will be discussed in Chapter 9.

7.3 Plotting Load Cycles Vs. Crack Size

The crack-growth rate is a function of the number of load cycles that have been applied to the rail, not the number of load cycles remaining till failure. Therefore, a new plot of crack size vs. applied load cycles needs to be generated using a conversion between fatigue life and applied load cycles. Similar curves are typically represented in terms of thousands of applied load cycles. In order to correctly represent the true high number of applied load cycles and expected shape of the curve, the applied load cycles in this plot are represented in terms of wheel load reversals instead of days. Using Table 6.3 and Figure 6.6, the total number of wheel load applications in 24-hour period can be calculated as 10,920 applied wheel load cycles. The fatigue life for each crack size can be converted to wheel load reversals by multiplying the fatigue life in days by 10,920 applied wheel load cycles. After the conversion to wheel load reversals is calculated, the fatigue life of the rail needs to be converted to the number of applied load cycles. In order to do this, a generalization is defined that specifies that the fatigue life for the smallest crack size corresponds to a given number of applied load cycles. This number is assumed as 100,000 applied load cycles. This assumption is made because the rail installation date is not readily available for many rails currently in service. The number of applied load cycles for the nth crack size is calculated using the equation:

$$N_n = \left| N_{f_n} - N_{f_1} \right| + 100,000 \tag{7.5}$$

where N_{f_n} is the fatigue life for rail profile with the nth crack size, and N_{f_1} is the fatigue life for the rail profile with the smallest crack size. It is important to note that only the shape of the load cycles vs. crack size curve is important in the derivation of the crack-growth-rate equation because the crack-growth rate is the derivative of the curve. Therefore, the generalization specifying that the life of the rail profile with the smallest crack corresponds to 100,000 applied load cycles does not affect the calculation of the crack-

growth rate.   After the conversion of fatigue lives to applied load cycles, Figure 7.2 can be expressed in terms of crack size vs. applied load cycles. This plot is shown in Figure 7.3.
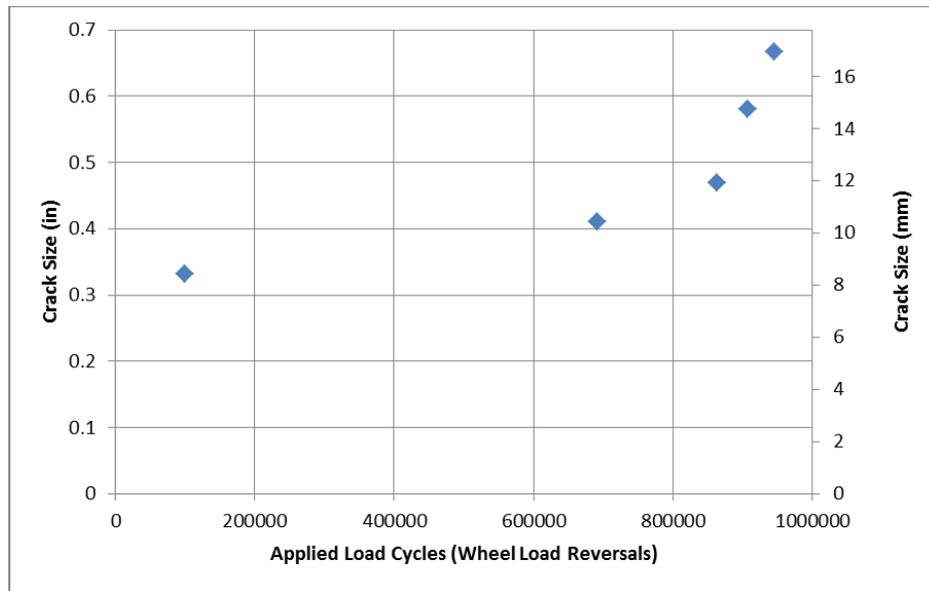
Figure 7.3 Crack size as a function of applied load cycles for the example side-wear profile.

Figure 7.3 displays an exponentially increasing trend that suggests that the crack size increases as the number of applied load cycles increases. In order to determine the crack-growth rate for the given profile, the five data points describing the crack-growth behavior are fit with an exponential equation.

7.4 Calculating the Crack-Growth Rate

The data displayed in Figure 7.3 is fit using the exponential form given by the equation:

$$y = ae^{bx} \tag{7.6}$$

where y is the variable representing the crack size, x is the variable representing the applied load cycles, a is the exponential constant, and b is the exponential exponent. The curve fit of the five data points is completed using the MATLAB function "nlinfit" and the form of Equation 7.6. The MATLAB function "nlinfit" uses the least squares method to estimate the constants a and b. The exponential fit of Figure 7.6 is shown in Figure 7.4.
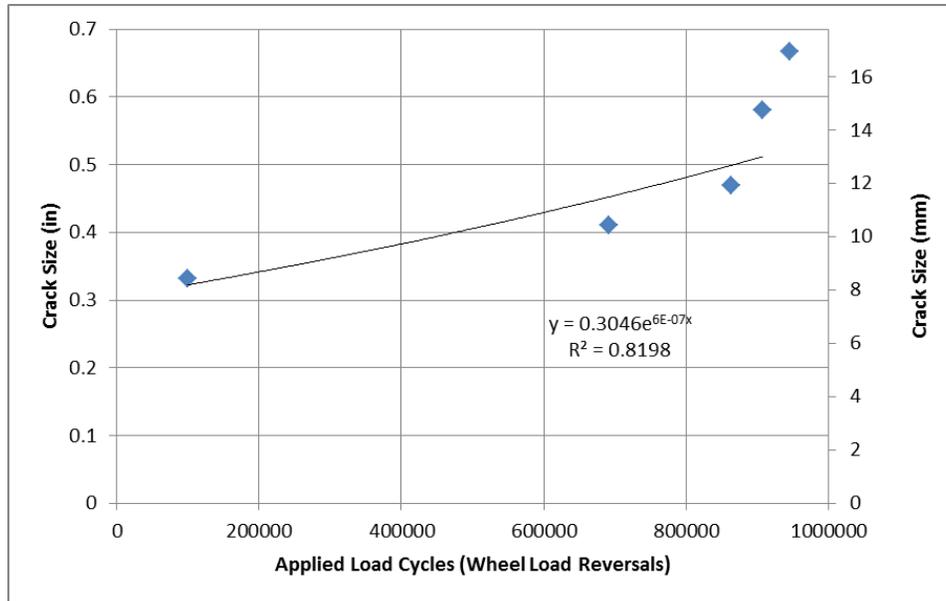
Figure 7.4 Least squares fit of crack size vs. load cycles plot.

The instantaneous crack-growth rate is equal to the slope of the fitted curve at a given number of cycles. The instantaneous crack-growth rate for the number of load cycles corresponding to the 10% HA crack provides an estimation of the worst case crack-growth rate for the current geometry and loading. It is desired to obtain the crack-growth-rate equation to estimate the crack-growth rate at any number of load cycles or crack sizes. This information can be used to provide a "what if" scenario where an undetected crack has grown above the 10% HA detectable size. The crack-growth-rate equation is developed by taking the derivative of the equation of the fitted line in the form of Equation 7.6. The crack-growth rate is given by the equation:

$$\frac{da}{dN} = Ce^{bN} \tag{7.7}$$

where *da/dN* is the crack-growth rate, *C* is the exponential constant equal to the product of original constant *a* and *b*, and *N* is the applied load cycles in days. Using this equation, the crack-growth rate response surface output variable can be represented for any rail geometry, loading, and material using the two constants *C* and *b*.

7.5 Conclusion

The constants for the exponential crack-growth-rate equation are the last two output variables for the response surfaces. They provide additional information for the rail regarding how quickly the detail fracture is growing in the rail head. The crack-growth-rate equation is determined from multiple life estimations for the rail using different crack sizes. Using the crack semi-major axis dimension and the

calculated fatigue life for each crack size, a plot can be developed of crack size vs. load cycles to failure for the rail. Five data points representing five crack sizes are used for each plot. This plot is used to generate a second plot of applied crack size vs. applied load cycles. The number of applied load cycles is determined for each crack by specifying that the smallest crack size corresponds to zero applied load cycles. By fitting the crack size vs. applied load cycles data with an exponential curve, an equation is developed that represents the cracks growth. The derivative of this equation is the crack-growth-rate equation.

The crack-growth-rate equation is used to calculate the instantaneous crack-growth rate for any number of applied load cycles. This information shows how quickly the crack is growing at any point in the rail's life. The number of applied load cycles is related to the crack size using Equation 7.6. Therefore, the crack-growth rate can be indirectly related to crack size. By inverting the crack-growth-rate equation given in Equation 7.7 and integrating with respect to a as is done in Equation 7.4, the number of applied load cycles in wheel load reversals that it takes for a crack to grow to a certain size can be calculated. After converting back to days, this can be used to determine a new inspection interval for the rail. This capability also allows for the evaluation of "what if" scenarios where crack sizes larger than the detectable 10% HA size are present in the rail head.

Chapter 8. Response Surface Development

The purpose of this project is to predict when a rail will fail due to fracture. The response surfaces that are developed provide this information by estimating the remaining life of the rail in days and the equation for the rail's crack-growth rate. These quantities become the output or dependent variables of the response surfaces. The information necessary to complete the life estimations for the rail are provided through eight input or independent variables. The derivation and use of these variables are explained in Chapters 3 through 6 and include the four geometric independent variables (h, A, I_{xx}, I_{yy}), three independent load variables (P_{MAX}, $°C$, ΔT), and one independent material variable (σ_{UT}). The independent variables are used to describe the geometry of the worn rail, the loading applied to the rail, and the material properties needed for the fatigue calculations. Each life estimation used in the creation of the response surfaces are developed from fatigue post-processing of the finite element stress results as explained in Chapters 5 through 7. The finite element models utilized for the life estimations are built using the geometry and loads developed from the eight independent variables.

Because there are multiple response output variables, multiple response surfaces are created using the same input variables. Three response surfaces are developed to provide the output variables of interest. The output of the first response surface is the estimate of the rail's remaining life in days. The second and third response surfaces are used to estimate the two constants necessary in the representation of the crack-growth-rate equation explained in Section 7.4. Therefore, there are three output variables that are used to describe the life and crack-growth rate for the rail. The response surfaces are created by calculating nodal values for the output variable corresponding to the current response surface and using interpolation functions to represent a continuous solution between the nodal values. The calculation process for the nodal output values is similar to the process used in Section 3.2 for the fitting of the worn rail profile. The eight independent variables are varied linearly in the response surfaces between their maximum and minimum values. Therefore, the nodal values of the output correspond to maximum or minimum values for each of the independent variables. Because each independent variable is linear in nature, they are called two level factors. The minimum number of analyses needed to gather the samples necessary for the completion and implementation of the response surface is given by the equation:

$$n = l^f \tag{8.1}$$

where n is the number of runs necessary, f is the number of factors, and l is the number of levels for each factor [24]. This represents the total number of combinations for f number of variables with l levels per variable. For eight linear variables characterized by two levels, the number of samples necessary to build the response surface is 256. Each sample consists of the values for the eight independent variables and

the three output variables. The formulation of the response surfaces, the development of the samples needed, the calculation of the response nodal values, and the implementation of the response surfaces are explained in the following chapter.

8.1 Transforming and Coding of Independent Variables

Each of the eight independent variables used to describe the geometry, loading, and material properties of the rail have different magnitudes and relationships with the response output. The independent variables are treated linearly and allowed to vary between a maximum and minimum value. In order to represent non-linear relationships between an independent variable and the response output, the independent variables can be transformed. For example if a variable is expected to have an exponential relationship with the response output, this relationship can be represented linearly by taking the natural logarithm of the independent variable. Therefore, the transformed independent variable is given by the equation:

$$x_T = ln(x) \tag{8.2}$$

where x is the variable of interest and x_T denotes the transformed variable. No variable transformations are utilized in the first generation of the response surfaces. After some experience with the independent variables has been developed, this assumption will be revisited for future generations of the response surfaces.

Because the independent variables have significantly different orders of magnitude, the variables with the highest magnitudes can notably outweigh the effects of the other variables. In order to avoid this problem, the independent variables are coded to convert the variable range to -1 to +1. The variable coding is completed using the equation:

$$\xi_x = \frac{2(x - x_{mean})}{x_{max} - x_{min}} \tag{8.3}$$

where x_{max} is the maximum value for the variable x, x_{min} is the minimum value for the variable x, x_{mean} is the variable x's mean, and ξ_x is the coded variable [24]. If variable transformation is used, the transformation is completed before the variable is coded. Coding is completed for all eight independent variables, and the independent variable vector is given by:

$$\bar{\xi} = [\xi_h, \xi_A, \xi_{I_{xx}}, \xi_{I_{yy}}, \xi_{P_{max}}, \xi_{\circ C}, \xi_{\Delta T}, \xi_{S_{UT}}] \tag{8.4}$$

where ξ denotes the coded independent variable. Once coding is completed for each independent variable, the development of the response surfaces can continue.

8.2 Response Surface Formulation

As previously stated, three response surfaces are generated to estimate the overall life and degradation of the rail. The first response surface has an output variable of remaining days till failure for the rail. This output variable is estimated for the given worn rail profile with a 10% HA elliptical flaw present in the rail head. This flaw size is the minimum detectable size for the Ultrasonic Test car and represents the "worst case" past load history for the rail. The life results necessary for the development of this response surface are generated using the fatigue post-processor FE-Safe and the Abaqus elastic stress results for the worn profile. The life estimation process using the Abaqus elastic stress results is explained in Chapter 6. The output variable of the second response surface is the constant C given in Equation 7.3 describing the crack-growth rate for the given worn profile and loading. The output variable of the third response surface is the exponent b from Equation 7.3. These output variables are determined using the FE-Safe results for five different crack sizes in each rail head. The estimated remaining days till failure for each crack size are used to create a applied load cycles vs. crack size plot. The crack-growth rate is determined from this plot and is explained in Chapter 7.

Each response surface has the same input variables, and is generated using an identical formulation. Because each independent variable used in the response surfaces is treated linearly, the variation of each variable is represented using a maximum and minimum value. The maximum and minimum values for each independent variable correspond to a nodal value for the output variable and represent the range of that independent variable in the response. Each output nodal value represents a combination of the eight independent variables at either their maximum or minimum value. The number of nodal output values needed for the definition of the response surface is calculated using Equation 8.1 as previously explained. Therefore, 256 nodal output values are used in the definition of each response surface. The value of the response output is given by the equation:

$$n(\bar{\xi}) = N_1(\bar{\xi})n_1 + N_2(\bar{\xi})n_2 + N_3(\bar{\xi})n_3 + \dots + N_{255}(\bar{\xi})n_{255} + N_{256}(\bar{\xi})n_{256} \qquad (8.5)$$

where $n(\bar{\xi})$ is the output variable for the given response surface, $\bar{\xi}$ is the vector of input variables after coding, $N_n(\bar{\xi})$ is the nth interpolation function, and n_n is the nth nodal value for the output variable [17]. After coding, the minimum value for each independent variable is equal to -1 and the maximum value for each independent variable is equal to +1. The interpolation functions used in the response definition are 8-D tensor products of linear 1-D interpolation functions commonly used in finite element formulations. The linear 1-D interpolation functions corresponding to a single variable are given in terms of the coded variable by the equations:

$$N_1 = \frac{1}{2}(1 - \xi) \tag{8.6}$$

$$N_2 = \frac{1}{2}(1 + \xi) \tag{8.7}$$

The linear 1-D interpolations functions are defined so that N_1 has a value of 1 at a ζ value equal to -1 and a value of zero at a ζ value equal to +1. N_2 is defined so that it has a value of zero at ζ value equal to -1 and a value of 1 at a ζ value equal to +1 [17]. This is demonstrated in the plotted 1-D interpolation functions in Figure 8.1
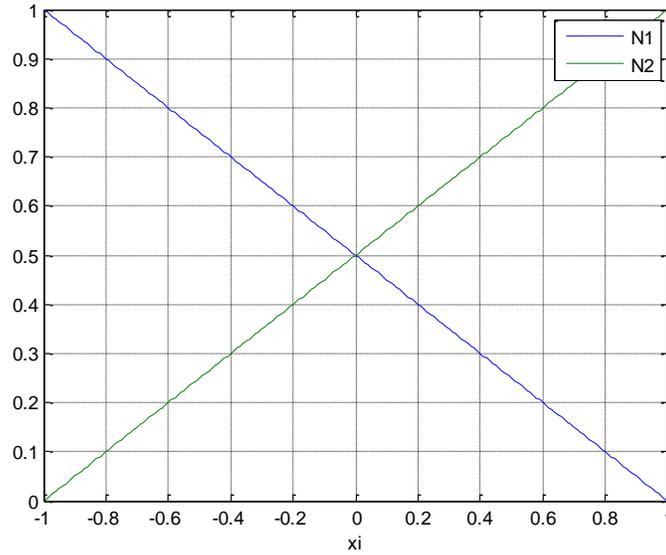
Figure 8.1 1-D interpolation functions.

The calculation of the first 8-D interpolation function used in Equation 8.5 is given in the equation:

$$N_1(\bar{\xi}) = N_1(\xi_h)N_1(\xi_A)N_1(\xi_{I_{xx}})N_1(\xi_{I_{yy}})N_1(\xi_{P_{max}})N_1(\xi_{\circ C})N_1(\xi_{\Delta T})N_1(\xi_{S_{UT}}) \tag{8.8} [17]$$

The remaining 255 interpolation functions are calculated using different combinations of the two 1-D interpolation functions. Therefore, the last 8-D interpolation function is given by the equation:

$$N_{256}(\bar{\xi}) = N_2(\xi_h)N_2(\xi_A)N_2(\xi_{I_{xx}})N_2(\xi_{I_{yy}})N_2(\xi_{P_{max}})N_2(\xi_{\circ C})N_2(\xi_{\Delta T})N_2(\xi_{S_{UT}}) \tag{8.9}$$

Using the calculated 256 calculated nodal output values and interpolation functions, the response output for the given response surface can be estimated for any value of the independent variables between their minimum and maximum values. The nodal output value n_1 corresponds to interpolation function $N_1(\bar{\xi})$ and the output of the response surface when all the independent variables are at their minimum (-1) value. The 8-D interpolation functions are used to interpolate the solution when the independent variables are

not at their minimum or maximum value. This formulation creates a continuous solution for the response for all independent variables.

8.3 Gathering Variable Samples

In order to calculate the nodal values necessary for the completion of the response surfaces, multiple samples of independent and output variables must be gathered. According to Equation 8.1, the minimum number of samples necessary for the completion of the response surfaces is 256. Therefore, 256 samples of the three response surface output variables and the eight independent variables must be generated. Ideally, the response surfaces would be generated using an over determined system with more than 256 samples. Unfortunately, each sample generated requires at least one lengthy Abaqus analysis run and post-processing with FE-Safe. In order to keep the overall analysis time at a minimum, the first generation of the response surfaces are generated with the minimum number of samples necessary. If additional samples are generated at a later date, the nodal output values can be re-calculated for the over determined system.

The samples are generated by varying the eight independent variables and determining the response output variables. The geometric independent variables (h, A, I_{xx}, I_{yy}) are varied by fitting multiple worn profiles from the Track Geometry car using the MATLAB program developed in Chapter 3, and developing Abaqus models around those geometries. These variables are related to each other and cannot be varied independently. Because there are four geometric independent variables and the variables are defined using linear interpolation functions, sixteen profiles are required for the response surface generation according to Equation 8.1. One unworn profile, seven side-wear profiles, and eight top-wear profiles are provided by a major American railroad for the development of the response surfaces. The side-wear profiles have wear levels ranging from 0/16^{ths} to 9/16^{ths}, and the top-wear profiles have wear ranging from 2/16^{ths} to 11/16^{ths}. The sixteen fitted profiles are found in Appendix E.1. The remaining five independent variables are independent of each other and can be varied linearly between two values. The two values are called the sampling points for the variables. The suggested sampling point locations for Gauss Quadrature in order to minimize integration error are ξ equal to $\pm \frac{1}{\sqrt{3}}$ [17]. The sampling points for each coded independent variable can be converted back to the original variable by solving Equation 8.3 for x, the original independent variable. This conversion equation is given as:

$$x = \frac{\xi_x (x_{max} - x_{min})}{2} + x_{mean} \tag{8.10}$$

This equation is used to calculate the sampling points for the variables P_{MAX}, $°C$, ΔT, and σ_{UT}. The sampling points for the three independent load variables and the material-property-independent variable

are shown in Table 8.1 with the independent variable maximum and minimum values (operational range) defined.

Table 8.1 Independent variable sampling points.

Independent Variable	P_{MAX} (kips)	$°C$ (deg)	ΔT (deg F)	σ_{UT} (ksi)
Min Value	40	0	25	133
Max Value	60	13.5	105	171
Sampling Value 1	44	2.85	42	141
Sampling Value 2	56	10.65	88	163

Independent Variable	P_{MAX} (kN)	$°C$ (deg)	ΔT (deg C)	σ_{UT} (MPa)
Min Value	177.93	0	-3.89	917.00
Max Value	266.89	13.5	40.56	1179.00
Sampling Value 1	195.72	2.85	5.56	972.16
Sampling Value 2	249.10	10.65	31.11	1123.85

The operational range for the maximum vertical wheel independent load variable is determined from wheel load data for the WILD sites implemented on rails in America. The operational range for the track curvature independent variable is adopted directly from Table 4.2 [7]. The operational range for the track temperature range independent variable is determined from historical weather and rail history data, and the ultimate tensile strength range is adopted from Orringer et al. and the AREMA handbook [4,7]. The load and material property independent variables are varied between the sampling values provided in Table 8.1for the generation of the response surfaces

As previously stated, the geometric independent variables are varied by developing sixteen Abaqus models representing the sixteen worn profiles provided. The first response surface estimates the remaining life of the rail with a 10% HA flaw present in the rail head. Therefore, a constant 10% HA crack is modeled in each of the sixteen worn profiles modeled. The second and third response surfaces estimate the crack-growth rate for the given worn profile and require multiple crack sizes. To gather the information necessary to generate the crack-growth rate, four additional cracks are modeled in each worn profile model. The total number of Abaqus models necessary for the second and third response surfaces is equal to 80—the total number of profiles multiplied by the number of crack sizes.

In order to continue the development of the response surfaces, the remaining four independent variables must also be varied. Because an elastic analysis is completed in Abaqus, the loads can be varied in FE-

Safe by scaling the stress datasets that make up the load history in the .LDF file. Therefore the maximum vertical load and the temperature range are varied between the variable sample values from Table 8.1 in FE-Safe. Additionally, the material properties used in the Abaqus elastic analysis are considered constant. Therefore, the material only needs to be varied in FE-Safe. Two materials are generated using Seeger's method as explained in Section 6.3, based on the constant elastic modulus, and the two sample values for the material's ultimate tensile strength shown in Table 8.1. The material ultimate tensile strength is varied in FE-Safe by changing the material definition for the rail between the two materials created.

The curvature of the rail determines the lateral-to-vertical load ratio and the magnitude of the lateral wheel load. Because the application of the vertical and lateral wheel loads are coupled in the Abaqus model, the magnitude of the lateral load cannot be varied independently of the vertical load by scaling the stress datasets that make up the load history in FE-Safe. Therefore, the curvature of the rail is varied in Abaqus by changing the magnitude of the lateral wheel load in the load manager. The two sample magnitudes of the lateral load that are used in the FE models are determined using the sample curvatures given in Table 8.1, the lateral-to-vertical load ratio calculated from Equation 4.3, the maximum vertical wheel load, and Equation 4.7.

In brief, the geometric independent variables (h, A, I_{xx}, I_{yy}) and the curvature of the rail are varied in the Abaqus FE model. The generation of the response surface that estimates the remaining life of the rail with the 10% HA flaw requires 32 Abaqus analysis runs because there are two lateral loads for each of the sixteen worn profiles. The response surfaces that estimate the crack-growth rate of the rail require 160 Abaqus analysis runs because there are five crack sizes for each worn profile and two lateral loads for each crack size. In order to run the Abaqus jobs as quickly as possible, a Python script is developed that submits jobs to the computing cluster queue. The Python script submits jobs, one at a time, to a computing cluster, and is provided in Appendix E.2.

The maximum vertical load, the temperature range for the rail, and the ultimate tensile strength of the rail material are varied in the fatigue analysis in FE-Safe. In order to generate the remaining life response surface, 256 fatigue analysis runs are required. In order to generate the crack-growth rate response surfaces, 1280 fatigue analysis runs are necessary because there are five times as many initial Abaqus models. These fatigue analyses are submitted using a batch file generated by an additional Python script. This batch file is created using different keyword (.KWD) files that specify the different load variables for the fatigue analysis. This script used to generate the batch file is documented in Appendix E.3. An Excel sheet is used to organize the input variables and the results from the FE-Safe analyses. The fatigue results

are extracted from the FE-Safe ODB files and imported into the Excel sheet using a third Python script given in Appendix E.4. The Excel spread sheet summarizing the development of the response surface variable samples is documented in Appendix E.5. The fracture mechanics post-processing is completed using the FE-Safe life results in the Excel spread sheet and a MATLAB code given in Appendix E.6. The results from the fracture mechanics post-processing are also recorded in the spreadsheet in Appendix E.5 to be used in the response surface generation. Once the Abaqus analysis runs, the fatigue analysis runs, and the fracture post-processing is complete, the independent and output variables can be compiled and the response surfaces nodal output values can be calculated.

8.4 Calculating Nodal Values

Once the samples of the independent and output variables are generated, the output variable nodal values can be calculated for the three response surfaces. The individual output variable of each response surface is represented using Equation 8.5. Using the multiple variable samples explained in the previous section, Equation 8.5 can be converted to a matrix equation given by:

$$\{n(\bar{\xi})\} = [N(\bar{\xi})]\{n\} \qquad (8.11)$$

where $\{n(\bar{\xi})\}$ is the 1 x 256 vector of the sample response surface output variable values, $[N(\bar{\xi})]$ is the 256 x 256 matrix of interpolation functions, and $\{n\}$ is the vector of the output variable nodal values. Each row of the interpolation matrix $[N(\bar{\xi})]$ is formed by evaluating the 256 interpolation functions at the set of coded independent variables for the given sample. The nodal vector $\{n\}$ in Equation 8.11 can be calculated using the least squares method given by the equation:

$$\{n\} = [B]^{-1}[N(\bar{\xi})]^{T}\{n(\bar{\xi})\} \qquad (8.12)$$

where $[B]$ is given by the equation:

$$[B] = [N(\bar{\xi})]^{T}[N(\bar{\xi})] \qquad (8.13) \ [17].$$

Equation 8.12 is used to calculate the nodal output values for each response surface. This calculation is completed using the Excel results spread sheet given in Appendix E.5 and the MATLAB code given in Appendix E.7. Once the nodal output values for each response surface are calculated, the response surface calculations are complete, and the response surfaces are ready for implementation.

8.5 Response Surface Implementation

After the nodal output values are calculated for the three response surfaces, the definition for each response surface is completed by plugging the calculated nodal values into Equation 8.5. This forms

three independent response surface equations; the first equation is the equation that represents the response surface that estimates the remaining life of the rail, the second equation represents the response surface that estimates the crack-growth-rate coefficient, and the third equation represents the crack-growth-rate exponent for the for the rail. Each response surface equation is a function of the coded variable vector, and is therefore ready to be used in the evaluation of rail sections.

In order to evaluate a rail section against the three response surfaces, the eight independent variables must be defined first. The four independent geometry variables are developed using the Track Geometry car to provide the X-Y data describing the worn profile. The Track Geometry car data is fit using the MATLAB code explained in Section 3.1 through 3.4 and provided in Appendix B.1. Once the fitted profile of the worn rail is created, the section properties are calculated approximately using the MATLAB code explained in Section 3.5 and provided in Appendix B.2. The section properties are the four geometric independent variables. The maximum vertical independent load variable is developed from WILD data for the section of track of interest. The second mode of the maximum vertical load distribution is fit using the MATLAB code explained in Section 4.1 and provided in Appendix C. The value of the maximum vertical load is calculated using Equation 4.3. The curvature of the rail in degrees is the 6th independent variable and is determined from track data for the given section of track. The temperature range that the rail experiences is developed from historical maximum and minimum temperatures from weather and track data. The temperature range independent variable is calculated using Equation 4.6. The final independent variable is the ultimate tensile strength of the rail. This value of this variable is determined from rail material data for the current section of track.

Once all eight of the independent variables are defined for the given rail section, the independent variables are coded using Equation 8.3. After the variables are coded, the vector $\bar{\bar{\xi}}$ is defined, and the 256 8-D interpolation functions are calculated for the given 1x8 independent variable vector. The calculated interpolation functions are plugged into the three response equations and the three output variables are calculated. The three response equations are given by the equations:

$$n(\bar{\bar{\xi}})_{life} = N_1(\bar{\bar{\xi}})n_{1life} + N_2(\bar{\bar{\xi}})n_{2life} + N_3(\bar{\bar{\xi}})n_{3life} + \dots + N_{255}(\bar{\bar{\xi}})n_{255life} + N_{256}(\bar{\bar{\xi}})n_{256life} \quad (8.14)$$

$$n(\bar{\bar{\xi}})_C = N_1(\bar{\bar{\xi}})n_{1C} + N_2(\bar{\bar{\xi}})n_{2C} + N_3(\bar{\bar{\xi}})n_{3C} + \dots + N_{255}(\bar{\bar{\xi}})n_{255C} + N_{256}(\bar{\bar{\xi}})n_{256C} \quad (8.15)$$

$$n(\bar{\bar{\xi}})_b = N_1(\bar{\bar{\xi}})n_{1b} + N_2(\bar{\bar{\xi}})n_{2b} + N_3(\bar{\bar{\xi}})n_{3b} + \dots + N_{255}(\bar{\bar{\xi}})n_{255b} + N_{256}(\bar{\bar{\xi}})n_{256b} \quad (8.16)$$

where $n(\bar{\bar{\xi}})_{life}$ is the estimate for the remaining days until failure for the given rail, $n(\bar{\bar{\xi}})_C$ is the estimated crack-growth-rate equation coefficient for the rail, and $n(\bar{\bar{\xi}})_b$ is the estimated crack-growth-rate

exponent for the rail. The three estimated outputs provide the information necessary to determine when the rail should be removed from service. The MATLAB code that is used to evaluate a given set of independent variables using the three response surfaces is provided in Appendix E.8.

8.6 Conclusion

The development of the response surfaces is the final stage of the project. The purpose of the response surfaces is to determine when a rail should be taken out of service using the rail's geometry, loading, and material properties as inputs. This estimation is completed using the remaining life in days for a rail with a 10% HA flaw and the crack-growth rate as response surface output variables. Therefore, three response surfaces are created. The geometry, loading, and material properties of the rail are described using eight linear independent variables that serve as the inputs for the three response surfaces. The eight independent variables are the four geometric independent variables developed in Chapter 3 (h, A, I_{xx}, I_{yy}), the three independent load variables developed in Chapter 4 (P_{MAX}, °C, ΔT), and the one independent material variable (σ_{UT}) developed in Chapter 6. The life estimations for the rail with a 10% HA crack are completed using the fatigue post-processing of the finite element elastic stress results. This estimation is explained in Chapters 5 and 6. The crack-growth rate is developed by calculating the life for the rail with five different crack sizes. The approximation of the crack-growth rate from the life data for the five crack sizes is explained in Chapter 7.

The outputs of the three response surfaces are described by three equations using nodal values for the response output variables and interpolation functions. The interpolation functions are functions of the eight independent variables after transformation and coding. The interpolation functions interpolate the solution between the calculated nodal values. The response surfaces are created by calculating the value of the output nodes used in the response equations. This is completed by generating multiple samples of the independent and output variables for each response surface. The minimum number of samples necessary for the development of the response surfaces with eight linear independent variables is 256. The least squares method is used to calculate the nodal output values for each response surface equation. The calculated nodal values for each response surface are plugged back into the appropriate response equation to complete the definitions of the three response surfaces. After the definitions of the response surfaces are completed, the response equations can be used to evaluate rail sections of interest and determine when they should be removed from service.

Chapter 9.   Results and Discussion

The next step in the progression of the project is an evaluation of the effectiveness and accuracy of the response surfaces generated to predict the remaining life and crack-growth rate of a worn rail. After the three response surfaces are generated as discussed in Chapter 8, the first step in the evaluation of the current model is a detailed examination of the trends seen in the life results and the effect of the various independent variables. By examining the life values used in the generation of the response surfaces and summarized in Appendix E.5, an initial understanding of the model's strengths as well as its short-comings is developed.

At first glance, it is clear that the many of the life results do not agree with certain expected trends. The main issue that is immediately apparent is that for many of the worn profiles, the life estimates do not exhibit a decreasing trend as the crack size increases. The estimation of the crack-growth rate in the rail is dependent on the assumed decreasing trend; therefore, the predicted crack-growth rates for rails exhibiting this unexpected trend are likely incorrect. This trend and others are evaluated in the following chapter. In addition, explanations and potential solutions are provided for the problems identified.

9.1 Modeling Discrepancies

After the initial evaluation of the response surfaces' results, a better understanding of the effects of the various independent variables is developed. When inspected, the trends associated with the independent load variables meet expectations. For example, if the maximum vertical load increases, the life estimate decreases. In addition, if the temperature gradient increases, the life estimate decreases. Finally, if the curvature of the rail increases, the life estimate decreases. The independent material variable, the ultimate tensile strength, is shown to have little effect on the life results. However, the life results for the material characterized by the higher ultimate tensile strength sample value are slightly lower than the life results for the lower ultimate tensile strength sample. This is likely due to a higher brittleness for materials with high ultimate tensile strengths.

While the geometric independent variables exhibit a general trend of decreasing life with increasing wear, there is significant scatter or "noise" in the results. Because of the scatter, there are a significant number of cases where the life increases as the amount of rail wear increases. This is similar to the problem associated with the rail profiles that do not exhibit the expected trend of decreasing life with increasing crack size. In both situations (the effect of the crack size and the effect of the independent geometry variables) there are some results that match the expected trends and some results that do not. It is believed that these problems are related and are caused by the same mechanics in the finite element model.

After evaluating the stress results for the finite element models in question, explanations for the unexpected trends are formulated. By examining the stress contours in the rail head, specifically near the crack edge, the load transfer and state of stress at the crack edge are better understood. It is believed that three factors contribute to the unexpected trends and inaccuracies shown in the life results. The first factor deals with inconsistent mesh density at the interaction between the rail and the wheel and the penetration of the slave and master surface in the wheel/rail contact definition. This inconsistency refers to significantly different element sizes and surface penetration amounts at the contact location for the 5 crack size models for a given worn profile. This mesh inconsistency is a consequence of the partitioning and meshing strategy utilized to mesh the rail head and the method of positioning the wheel on the rail. Without a consistent mesh at the contact point and consistent surface penetration, the effect of the load transfer from the wheel to the rail is also inconsistent and may affect or "bleed into" the area surrounding the crack edge. The second factor is the representation of the crack in the rail head. It is believed that the current method of tying two partitioned rail sections together to form a crack and the element choice used to mesh the rail head are unable to represent the singularity in the stress and strain at the crack characterized by a sharp crack tip. Without representing this singularity, the stress and strain state at the crack edge are likely erroneous. The third factor is the material assumptions used to generate the material models implemented in the fatigue calculations and explained in Section 6.3. With the number of assumptions made in characterizing the fatigue material constants, the life results are likely unreliable. This factor is not expected to affect the trends seen in the results, but instead the accuracy of the results.

9.2 Contact Issues

As previously stated, a significant portion of the life results fail to meet the expected trend of decreasing life with increasing crack size for the different crack sizes modeled in a given worn profile. Because the trend is defined for a given worn profile using the life results for the five different crack sizes, nothing in the model is changed except the geometry of the crack and the subsequent mesh. The failure to meet the expected trend suggests that there is an inconsistency in the finite element model that surfaces when the geometry of the crack is updated. The first factor believed to be partially responsible for the unexpected trends present in the life results is the inconsistency of the mesh density at the contact location between the rail and the wheel and the amount of interference between the master and slave surfaces due to the positioning of the wheel on the rail. The mesh density inconsistency refers to a change in element size from model to model for the five crack sizes modeled in each worn profile, and is due to the partitioning and meshing scheme used in the rail head. After evaluating the results for some of the profiles in question, this difference in mesh density is clearly identifiable. The inconsistency of the master/slave surface penetration is due to the method of positioning the wheel on the rail. This procedure is completed

manually by moving the wheel in small vertical and lateral increments until it makes contact with the rail. The over-closure or penetration of the contact surfaces is removed by adjusting the nodes on the slave surface to remove the over-closure as is explained in Chapter 5. This method is susceptible to inconsistent penetration amounts prior to the slave-node adjustment from model to model which slightly changes the contact patch or area. When there is more penetration of the contact surfaces, the points of high localized stress move farther away from the cross-section of the rail where the crack is located. While the positioning of the wheel is typically completed once for each worn profile, there are certain circumstances during the running of the analyses where the wheel position is changed for a given crack size to aid the convergence of the analysis. The combined effect of these two features is significant differences in the transfer of the load from the wheel into the rail from model to model.

The partitioning scheme used to mesh the rail focuses the mesh around the crack radially. This technique allows for biasing of the mesh towards the crack to maintain a controlled refined mesh at the crack edge, which is the primary area of interest. However, the mesh at the outer surface of the rail head is not well controlled. Many factors affect the mesh density at the interaction between the rail and the wheel. These factors are the type and amount of wear that the rail has undergone, the location of contact between the wheel and the rail which is also dependent on the rail's wear, and the vicinity of the crack to the outer surface of the rail and its geometry. Because the mesh is centered around the crack geometry, the mesh can be significantly different for each of the five crack sizes. Results that meet the expected trend of decreasing life with increasing crack size have relatively consistent element sizes at the contact location and penetration amounts between the two contact surfaces. Figure 9.1 provides an example of a worn profile that meets the expected trend.
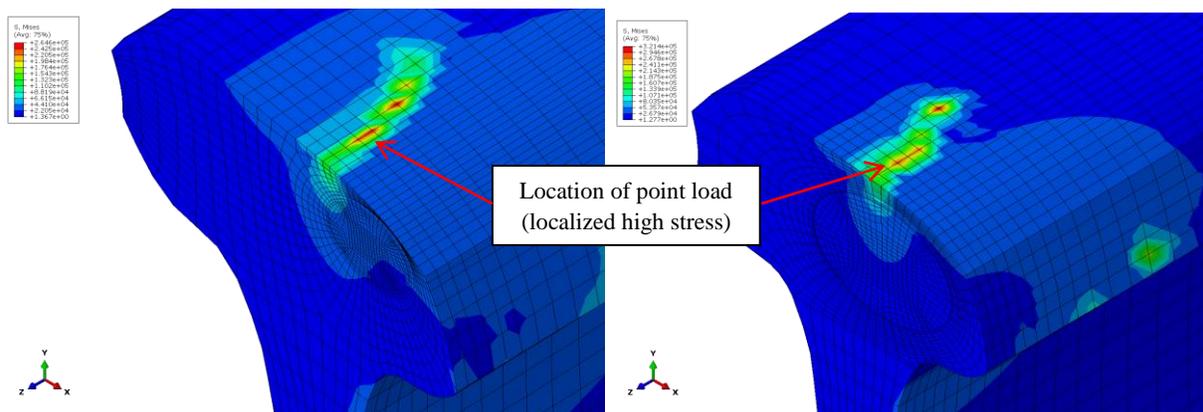
Figure 9.1 von Mises averaged stress contours for the 3/16[ths] side-wear profile for the 5% HA (left) and 20% HA (right) flaws.

Figure 9.1 displays the mesh and stress contours for the smallest and largest crack modeled in the 3/16[ths] side-wear profile. In Figure 9.1, the wheel instance has been removed from the view in order to visualize the stresses at the contact location between the wheel and the rail. Additionally, the rail is cut along the crack plane to show the stress contours at the crack edge. The crack plane is the x-y plane that the crack is located on. The models of the 3/16[ths] side-wear profile exhibit relatively constant mesh densities and therefore element size at the contact location for the five crack sizes. In addition, the points of localized high stresses are approximately the same distance along the z-axis from the crack plane. This shows that the penetration of the slave and master surface in the rail/wheel contact definition is constant. The combination of these two features suggest that the wheel load is transferred into the rail consistently, and that the life results should meet the expected trends. While the mesh density, and therefore the distribution of energy from the wheel into the rail is relatively constant for the 3/16[ths] side-wear models, the contact stresses at the point of contact between the wheel and rail is not considered converged. This enables the stress and life results to meet the expected trend of decreasing life with increasing crack size, but suggests that continued refinement of the mesh strategy is needed to verify the accuracy of the stress and strain state at the crack edge in these models. The von Mises averaged stress values and estimated fatigue lives for the 5 crack sizes modeled in the 3/16[ths] side-wear profile are given in Table 9.1.

Table 9.1 Estimated von Mises stress and life for 3/16[ths] side-wear models.

a (in)	Stress (psi)	Life (days)	a (mm)	Stress (MPa)	Life (days)
0.332	53440.5	83.392	8.4328	368.459373	83.392
0.411	60571.9	29.220	10.4394	417.6286579	29.220
0.47	67348.2	13.490	11.938	464.3496139	13.490
0.581	70534.7	9.530	14.7574	486.3197638	9.530
0.667	76915.1	5.984	16.9418	530.3110846	5.984

The stress and life values given in Table 9.1 represent the minimum stress and life queried at the crack edge. As expected, the stress increases at the crack edge as the crack size increases which produces decreasing fatigue life estimations. The crack-growth curve generated from the fatigue life values provided in Table 9.1 is given in Figure 9.2
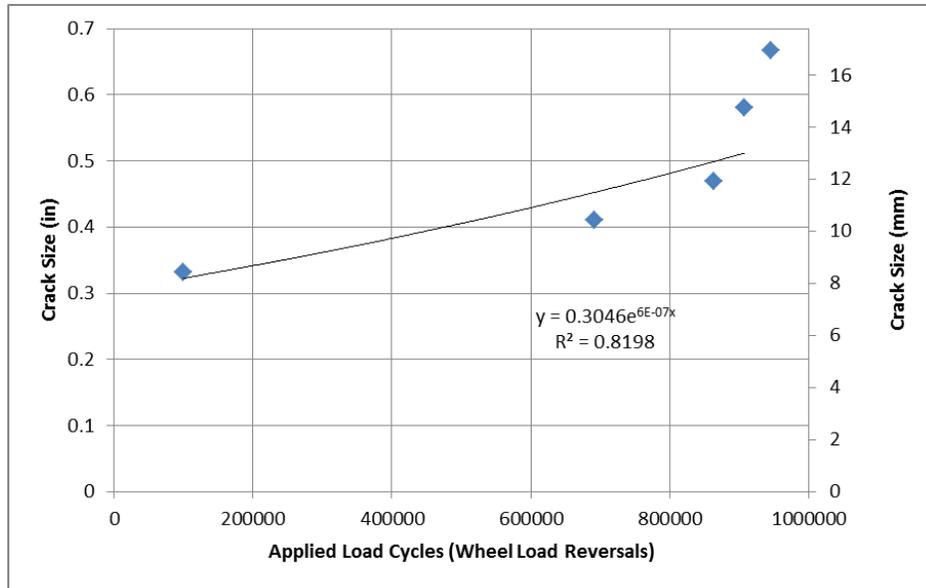
Figure 9.2 Crack size vs. applied load cycles for the 3/16^{ths} side-wear profile.

The 3/16^{ths} side-wear profile results illustrated in Figure 9.1 and 9.2 and recorded in Table 9.1 represent results that match the expected life trends.

Many profiles do not exhibit the positive features present in the 3/16^{ths} side-wear profile models and their respective meshes. Figure 9.3 provides an example of one of these profiles.
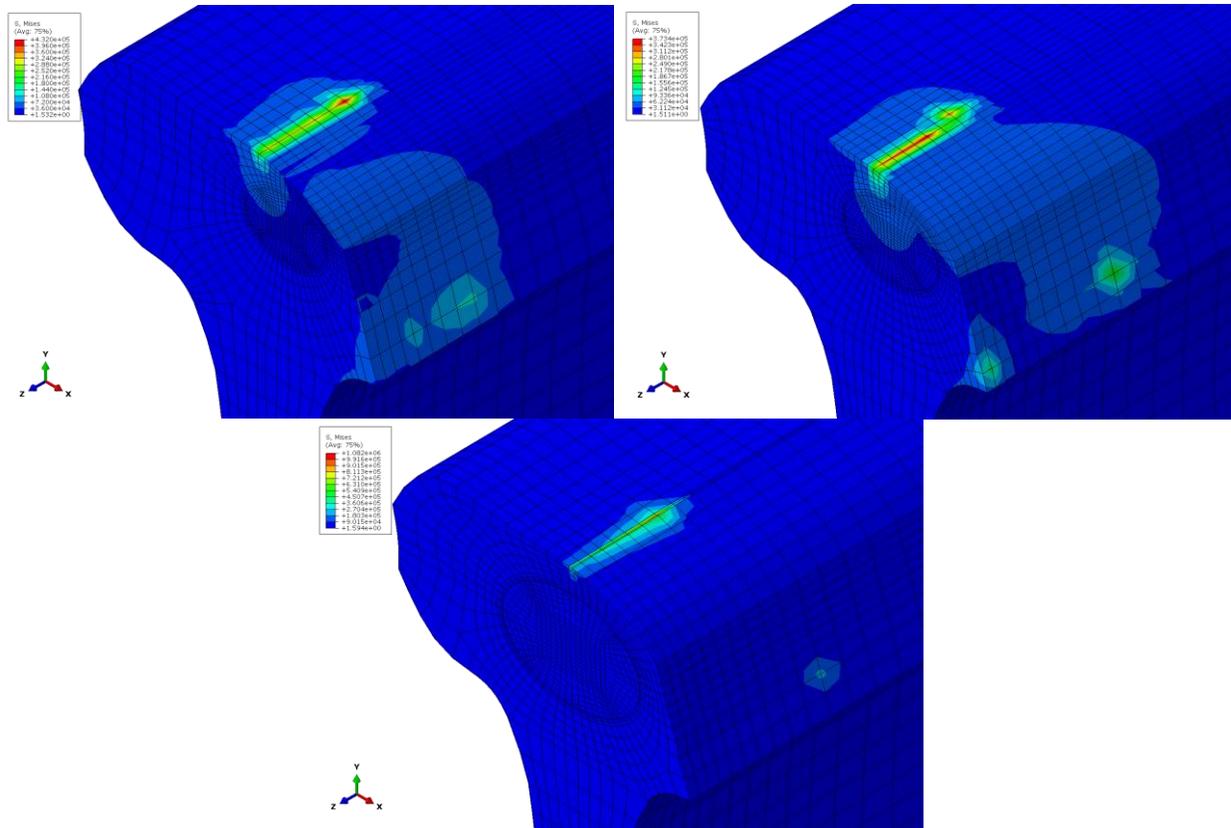
Figure 9.3 von Mises stress averaged contours for the 9/16[ths] side-wear profile 5% HA (left), 7.5% HA (right), and 15% HA (bottom) crack models.

The life results for the 9/16[ths] side-wear profile are considerably outside the expected decreasing life trend. Figure 9.3 provides the von Mises stress contours for the smallest (5% HA) and largest (15% HA) cracks as well at the 7.5% HA crack to illustrate the features of the models that contribute to their failure to meet the expected trend. First, the 5% HA and 7.5% HA illustrate the effects of inconsistent penetration between the wheel and rail contact surfaces. The 5% HA crack model features an interaction definition with more surface penetration than the 7.5% HA crack model. The 7.5% HA crack model with less surface penetration features points of high concentrated stress much closer to the crack plane than the 5% HA model. This difference in location for the areas of high concentrated stress results in inconsistent effects at the crack edge; the 7.5% HA crack model with closer concentrations of high contact stresses produces higher stresses at the crack edge, directly affecting the life results.

An example of inconsistent mesh density is illustrated in the 5% HA and 15% HA crack models provided in Figure 9.3. The mesh in the 15% HA crack model features a much smaller element size at the points of contact than the mesh for the 5% HA crack model. This smaller element size significantly raises the magnitude of the stress at the contact points, but decreases the distribution of the stress from the contact point to surrounding areas. The magnitude of the stress at the point of contact is approximately 432 ksi (2979 MPa) for the 5% HA crack model and 1082 ksi (7460 MPa) for the 15% HA crack model. These stress values are unrealistically high because the load is essentially being carried by a small number of nodes as point loads. By continued refinement of the mesh at the contact location, a converged contact stress could be determined. However, only the stress at the crack edge is of interest in the fatigue life analysis of the rail. While the stress is higher at the point of contact for the 15% HA crack model, it has a smaller effect on the magnitude of stress at the crack edge because the energy from the wheel load is more efficiently transferred into the rail and distributed by the smaller elements. In the 5% HA crack model, the contact stresses "bleed" into the crack edge area. This significantly increases the life estimates for the 15% HA crack model. The von Mises stresses and estimated fatigue lives for the 9/16^{ths} side-wear profiles are given in Table 9.2.

Table 9.2 Estimated von Mises stress and life for 9/16^{ths} side-wear models.

a (in)	Stress (psi)	Life (days)	a (mm)	Stress (MPa)	Life (days)
0.332	49900.9	58.90916	8.4328	344.0546837	58.90916
0.411	59777.4	17.64209	10.4394	412.1507718	17.64209
0.47	62271.8	13.16512	11.938	429.3490589	13.16512
0.581	49310.2	64.41322	14.7574	339.9819495	64.41322
0.667	51844.8	45.36021	16.9418	357.4574059	45.36021

The stress and life values given in Table 9.2 follow no understandable trend when only examining the life values, and are results of the mesh characteristics of each model. This is illustrated in the plot of the crack size vs. applied load cycles developed from the estimated life results. This plot is shown in Figure 9.4.
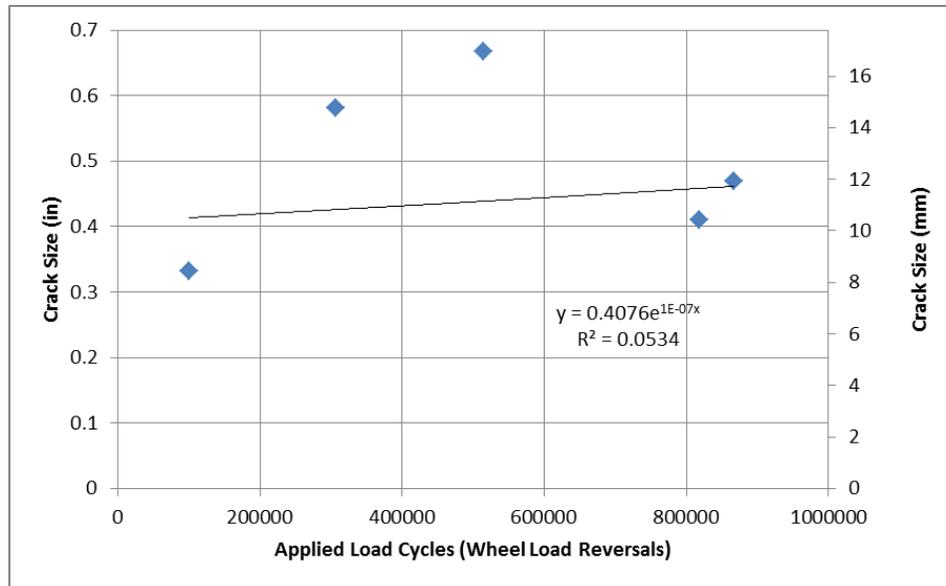
Figure 9.4 Crack size vs. applied load cycles for the 9/16^{ths} side-wear profile.

During the development of the finite element model, a convergence study was completed to determine the converged stress state at the crack edge. This process is explained in Section 5.5. Because the mesh was refined around the crack edge during the convergence study and not at the outside edge of the rail head, the stress and strain results appear to have converged at the points of interest: the crack edge. However, the local effects of the wheel load at the crack edge have not converged and greatly influence the stress results. In order to address the mesh and contact issues explained in this section, a new mesh and partitioning strategy must be adopted that allows for a controlled and refined mesh at the wheel. In addition, the penetration between the slave and master surface for the wheel/rail contact definition should be set to a constant value.

9.3 Crack Representation

The second factor believed to contribute to the inaccuracies present in the estimated life results is the representation of the crack in the rail model. The current method models the crack by tying two face partitioned rail sections together. The face partitions represent the geometry of the elliptical crack on the cross-section of both rail sections, and the tie constraints exclude the nodes contained by these partitions. This procedure is detailed in Section 5.2. The element choice to mesh the rail sections is three-dimensional linear bricks (C3D8). In addition, the mesh is significantly refined around the crack using a radially biased mesh to achieve convergence at the crack edge. According to linear elastic fracture mechanics, the theoretical stress at a sharp crack edge approaches infinity. While the detail fracture in the rail isn't expected to be infinitely sharp, the stress and strain state at the crack edge is expected to be characterized by a strong singularity. The current modeling of the crack geometry, element choice to

108

represent the stress and strain at the crack edge, and mesh density at the crack edge are unable to capture the true singularity of the actual crack. The linear elements have trouble representing the high stress gradient at the crack edge, and predict a seemingly converged but unrealistically low stress and strain at the crack front. The stress values at the crack edge for the 3/16ths and 9/16ths side-wear profiles are given in Tables 9.1 and 9.2 respectively. Assuming a material yield strength of 70,000 psi (482.63 MPa), the stresses in Tables 9.1 and 9.2 are mostly within the elastic region of the material. For a sharp crack, the stress and strain at the crack edge are expected to be highly plastic. This is consistent with Small Scale Yielding if the plastic effects at the crack edge are contained in a small region when compared to the size of the region characterized by the crack singularity. This strongly suggests that the stresses predicted in the finite element models do not represent converged solutions at the crack edge. Two of the profiles used in the convergence study explained in Section 5.5 are the 8/16ths side-wear profile and the unworn profile. While the stresses developed in models appear to be converged for the meshes selected, the life trends do not meet expected trends of decreasing life with increasing crack size. Examples of these life values are provided in Table 9.3.

Table 9.3 Life values for convergence study profiles.

8/16ths Side-Wear Profile			Unworn Profile		
a (in)	a (mm)	Life (days)	a (in)	a (mm)	Life (days)
0.332	8.4328	2.808	0.332	8.4328	810.261
0.411	10.4394	2.382	0.47	11.938	1040.579
0.47	11.938	1.672	0.667	16.9418	700.003
0.581	14.7574	2.254	0.879	22.3266	1460.915
0.667	16.9418	3.258	1.051	26.6954	2425.074

As shown in Table 9.3, the life values do not continually decrease as the crack size increases. This suggests that stress values developed in the finite element models have not reached convergence for any of the models. While providing a biased mesh at the crack edge, it is believed that the mesh technique, element type, and element size cannot capture the high stress gradient at the crack edge due to the singularity. Extremely small element sizes or higher order elements would be required to approach a converged solution.

There are many fracture-mechanics-based features in Abaqus that can be used to represent sharp cracks in bulk material. These features are developed using linear elastic fracture mechanics assuming SSY. The first method of representing a crack using the fracture mechanics tools available in Abaqus is the contour integral. The contour integral is used to characterize the stress intensity or strain energy release rate for a

given stationary crack. When using the contour integral, the crack is represented using a seam geometry that creates secondary uncoupled nodes along the crack face. This allows the crack to open as the part is loaded. The crack edge is represented as the edge of the faces selected during the seam definition. In addition, a small tube of concentrated wedge elements is created around the crack tip. The wedge elements are degenerated quadratic brick elements collapsed on one side with a singularity activated and the mid-side node moved from the 0.5 location to the 0.25 location. Using a considerably refined mesh and the degenerated element tube at the crack edge, the stress, strain, and stress intensity or strain energy release rate at the crack edge can be converged. The stress intensity or strain energy release rate typically converges with a coarser mesh than that required to converge the stress and strain near the crack tip. The stress intensity or the strain energy release rate is activated by requesting an additional field output that is documented in the Abaqus .DAT file from the analysis.

The second method of representing a crack in bulk material using the fracture mechanics tools in Abaqus is the extended finite element method (XFEM). XFEM is a mesh independent method that utilizes enriched elements featuring phantom that nodes that allow the enriched element to split in two. The XFEM method can be used to model initiation of a crack or the propagation of an existing crack. In addition, it can be used to characterize the stress intensity or strain energy release rate at the crack edge by requesting the appropriate field output. The crack is created by intersecting the meshed solid with a shell geometry characterizing the size and shape of the crack. The shell crack geometry should intersect the elements in the meshed solid instead of lying of the elements' edge. An enriched area in which the crack can propagate is defined in order to specify the enriched elements capable or separating. In order to model propagation of a crack or initiation, the fracture toughness of the material and initiation criterion must be defined.

9.4 Material Assumptions

The third factor believed to contribute to the inaccuracies present in the current model is the method of defining the fatigue material constants and the use of those constants in the fatigue analysis. The material property definition is explained in Section 6.3 and is empirically based. It is not believed that the material definition method contributes to inconsistency of the life results or their deviation from the expected trends. However, it is believed to be a weakness of the current analysis method that adds to the inaccuracies assumed to be present in the life results. This assumption cannot be quantified without comparison of the life results to historical data or testing. The current approach uses a strain-life algorithm to estimate the remaining life of a given worn rail with a crack of a certain size present. Typically strain-life algorithms are used to estimate the number of load cycles for a crack to initiate and the location of initiation. Fracture mechanics approaches are typically used to characterize the effects of a

stationary crack or to determine the number of cycles required for of a crack to grow from an initial size to critical, assuming an initial and final crack geometry. Because the fatigue material constants are defined using the empirically based Seeger's method, it is unclear of the definition of failure used. Failure could be defined as the initiation of a crack or it could be defined as the ultimate separation of the part into two pieces. Therefore, the life estimates could be representing the number of cycles for the crack to continue to propagate or to cause an ultimate failure of the rail. This uncertainty cannot be addressed without determining the material constants from tests on the actual materials of interest. In addition to the remaining life output of the response surfaces, the other output of the response surfaces is the crack-growth rate for the rail. This variable is most accurately represented using a fracture mechanics approach, and is shown to have specific relationship with the stress intensity range as given in Paris law provided in Equation 7.1. In order to more accurately represent the crack-growth rate and the remaining life of the cracked rail, it would be appropriate to utilize a full fracture mechanics approach. This would require the propagation of the crack to be characterized using the stress intensity at the crack edge instead of the stress and strain. In addition, the material constants used in Paris law would need to be defined for the material and stress ratio of interest.

9.5 Conclusion

The inconsistency in life results and their inability to meet the expected trends is mainly attributed to the mesh and surface penetration inconsistency at the contact location between the rail and the wheel. This is a main factor that changes substantially from model to model. However, it is believed that the representation of the crack in the current model, and the material constants used in the fatigue analysis add to the inaccuracies of the model. In order to create a model that more accurately estimates the fatigue failure of the rail, these three problems must be addressed. The solution includes adopting a fracture mechanics method to represent the crack in Abaqus and changing the mesh and partitioning technique in the rail head to guarantee and consistent refined mesh that correctly distributes the wheel load into the rail. The fracture mechanics method suggested is the Extend Finite Element Method, and it allows for the stress intensity to be characterized at the crack edge. This eliminates the need to reach a converged stress and strain solution at the crack edge, and allows for a more accurate representation of the crack-growth rate and remaining life of the rail. In order to adopt this method, additional material constants characterizing the Paris law must be determined from tests or other resources for the materials of interest.

While the effectiveness of the model is evaluated in this chapter, the accuracy of the model is not specifically addressed. It is important to understand that the accuracy of the response surfaces' results can only be validated by correlation to actual tests on actual rails whether in service or in the laboratory. After significant refinement of the modeling technique used in this project to address the problems

explained in this chapter, a validation stage will be necessary. In the validation stage, the results from the refined response surface models will be compared to historical and current rail data collected by American railroad companies. The ability of the response surfaces to correctly estimate the remaining life and crack-growth rate of the rail can only be determined after the validation stage is complete.

Chapter 10. Conclusions

Current rail maintenance procedures utilize wear limits and ultrasonic testing to determine when a rail should be removed from service. The rail failure model presented in this thesis attempts to couple rail wear with the propagation of sub-surface cracks titled detail fractures to develop a scientific approach for the evaluation of the structural integrity of worn rails. If the remaining life of a rail can be predicted accurately, the need for wear limits could be eliminated. The rail failure model developed here utilizes finite element models to characterize the state of stress and strain at an assumed flaw geometry. Using the predicted stress and strain results, life estimations are completed using strain-life fatigue and fracture mechanics methods. A functional relationship between rail wear and the life and crack-growth rate of a rail is developed through the generation of multiple response surfaces that reference the results of finite element and fatigue analyses. The functional relationship is developed using finite element models of multiple samples of worn rail profiles and their representative loads. The geometry, loading, and material properties of a given rail are represented with a limited number of independent variables in the response surfaces to keep the scope of the project within reason. The independent variables and other assumptions utilized in the development of the rail failure model are based on existing resources relating to rail failure due to fatigue and fracture as well as discussions with representatives from American railroad companies.

10.1 Summary

The rail failure model presented in this thesis consists of three response surfaces. The output variables of the three response surface are the remaining life, the crack-growth-rate constant, and the crack-growth-rate exponent for a given rail. The crack-growth-rate constant and exponent are paired with an exponential equation to provide a crack-growth-rate equation as a function of applied load cycles. The rail's remaining life and the crack-growth-rate equation are used to determine when a rail should be removed from service due to rail wear. Each response surface is a function of the same list of independent variables. The variables are divided into geometric section properties, independent load variables, and an independent material variable. The geometric section properties include the overall height of the rail, the cross-sectional area of the rail, the area moment of inertia about the horizontal axis going through the web origin, and the area moment of inertia through the vertical axis through the web origin (h, A, I_{xx}, I_{yy}). The independent load variables include the maximum vertical wheel load, the curvature of the rail, and the 24-hour temperature range that the rail endures (P_{MAX}, $°C$, ΔT). The independent material variable is the ultimate tensile strength of the rail material (σ_{UT}).

The independent variables must be defined for a given rail in order to evaluate its structural integrity. The geometry variables are calculated by fitting the profile data provided by a Track Geometry car in order to

create an approximation of the rail cross-section. The fitting procedure is completed using the least squares method. Once the rail cross-section has been determined, the section properties can be calculated approximately. In addition, the output from the profile fitting procedure can be imported into the finite element model to create a model of the worn profile. The range in wear severity and type can be represented by collecting multiple samples of worn profiles. These profiles are divided into side-wear and top-wear profiles.

The independent load variables are developed from collected track and weather data. The magnitude of maximum vertical wheel load is determined from load distribution data collected by WILD sites. An equation using the statistical factors of the distribution, as shown in Equation 4.3, is used to calculate the maximum vertical load to maintain a consistent method. The curvature of the rail is taken directly from track data for the given section of track, and can be used to calculate the magnitude of the lateral wheel load as a function of the maximum vertical wheel load. The temperature delta is defined using a rail neutral temperature between 85 and 110 °F, the average minimum temperature that rail experiences, and Equation 4.6. The load cases defined in the finite element models are developed from these three load variables.

In order to create a new worn profile finite element model, the nodal coordinates that are calculated as part of the profile fitting process are imported into the model. The geometry of the rail in the model is set up parametrically to automatically update and mesh with the new worn geometry. There are three loads applied in the finite element model. The loads are the vertical wheel load, the lateral wheel load, and the thermal load. Because the loads are cyclic in nature, the maximum and minimum magnitudes of each load must be represented. The vertical and lateral wheel loads are coupled, so the maximum magnitude for both the vertical and lateral wheel loads are applied simultaneously. This is also true for the minimum magnitude for both loads. The maximum magnitude for the rail thermal load corresponds to a zero state of stress. Therefore, only the minimum thermal load needs to be represented in the model. Three independent load steps are created to apply these loads. The standard-bending load step corresponds to the maximum magnitude of the vertical and lateral wheel loads. The reverse-bending load step corresponds to the minimum magnitude of the vertical and lateral wheel load, and the thermal load is applied in the minimum-temperature load step. The loads steps are independent static-elastic analyses so that they can be superimposed and scaled to form load histories at different magnitudes of the individual loads.

The fatigue life of the rail is calculated using a strain-life algorithm in a fatigue post-processor for the Abaqus elastic stress results. The fatigue life of the rail corresponds to a rail with a detail fracture that is

10% HA. The geometry of the crack is developed by Jeong et al. as introduced in Section 2.4. This flaw size is the minimum detectable size for the ultrasonic testing equipment, and is used to represent the "worst case" scenario because the previous load history of the rail is unknown. Four additional cracks of different sizes are modeled in each rail, and the life results for all five cracks are used to generate a crack-growth curve for each worn profile, load variables, and material properties. The crack-growth curve is used to generate the crack-growth-rate equation.

In order to generate the three response surfaces, samples of the independent variables and the value of their respective output variables must be collected. According to Equation 8.1, with 8 independent variables that are considered to have linear relationships with the output variables, 256 samples of the variables must be collected in order to develop the response surfaces. These samples are generated using multiple worn profiles and varying the load and independent material variables in the finite element models and the fatigue post-processor between two sample values. Once all the samples have been collected, the three response surfaces are generated using a least squares method. Each response surface is described by an eight-dimensional equation for the respective output variable.

10.2 Conclusions

The method for generating the rail failure response surfaces presented in this thesis depends on the collection of multiple analysis samples to populate the response equations. The analysis samples are generated using finite element models and strain-life fatigue models that are built around the independent variables that define the rail's geometry, loading, and material properties. The rail's ability to withstand cyclic loading is evaluated using the remaining fatigue life of the rail assuming a flaw equal to the minimum-detectable crack size and the crack-growth-rate equation.

The independent geometry variables used to represent the worn geometry in the response surfaces describe the section properties of the worn rail, and are calculated approximately from the fitted rail profile. The profile fitting program developed in MATLAB has been shown to be able to handle a wide range of side-wear and top-wear profiles. The first generation of the response surface is limited to these profile types because they are the profiles most commonly seen in rail operations, and feature smooth geometries that can be represented with the curve element order selected for the curve fitting procedure. In addition, the approximate methods for calculating the independent geometry variables has been shown to produce realistic values for the profiles examined during the generation of the response surfaces. The 16 fitted profiles used in the development of the response surfaces are provided in Appendix E.1.

The independent load variables used to represent the loads that are applied to a rail include all the components necessary to develop deterministic loads that can be applied in the finite element models to

represent the exterior loads that a rail endures. However, residual stresses due the plastic deformation of the rail during manufacture and use are not represented in the current generation of the response surface. While there has been research that suggests that residual stresses in the rail head play a large part in the development and growth of sub-surface flaws [7], the information necessary to quantify the stresses and represent them in the finite element models is not readily available. The current representation of the independent load variables allow for the loads to be superimposed and scaled to characterize different load magnitudes for the three loads applied in the finite element models: the vertical wheel load, the lateral wheel load, and the rail thermal load. This allows for easier varying of the load variables in the fatigue post-processer for the generation of the samples needed for the response surfaces, which greatly reduces the computational time required to generate the response surfaces. Each load variable is quantified systematically from historical data for either the track or the weather. At this point, the list of independent load variables sufficiently represents all of the loads included in the rail study. The definition of the independent material variable (σ_{UT}) is well known for the rail types currently in service. The fatigue constants necessary for the strain-line fatigue calculations can be approximated from the material ultimate tensile strength and the constant elastic modulus utilized in the finite element models. Because material data is not readily available for the rail steels of interest, this approximation is convenient for the fatigue analysis. However, if more accurate material properties can be attained, that would be preferred.

The finite element modeling ties directly into the profile fitting procedure through the use of parameterized geometry and partitioning. This provides the ability to generate new models of worn profiles quickly and efficiently if the profiles are available from the Track Geometry car. The application of the loading also ties directly into the definition of the independent load variables. As stated in Chapter 9, the meshing scheme and crack representation do not capture the true stress and strain gradients at the crack edge. By addressing the concerns listed in Chapter 9 and the suggestions in Section 9.5, the accuracy of the models results could be increased, and the need for the fatigue post-processor would be eliminated. The model, as is, provides a strong foundation for the continued development and exploration of the propagation of cracks in worn rails.

The post-processing of the elastic stress results in the fatigue post-processor allow for the superposition and scaling of the stress results in order to generate a 24-hour load history. This feature allows for the loads applied during a day to be represented using rainflow counting. The fatigue life that is calculated through the fatigue post-processor provides an estimate of the rail's remaining life that can be used to determine when the rail should be removed from service or inspected again. By post-processing the

fatigue life results to generate the crack-growth-rate equation, more information is available relating to how quickly the rail's structural capability is decreasing.

The development of the response surfaces from the input and output variable samples generated with the finite element models, fatigue post-processing of the stress results, and the development of the crack-growth rate from the fatigue life results is completed with the lowest order representation for the relationship between the output variables and the eight independent variables. Once additional experience has been developed with each of the output variables, the relationships can be reevaluated to better represent the output. The response surface formulation allows for the use of higher-order relationships by defining higher order interpolation functions. The MATLAB codes for the generation and use of the response surfaces are staged for the next stage in the refinement of this model.

10.3 Contribution of This Work

While there has been significant amounts of work in the area of rail failure due to fatigue and fracture, the work of Jeong et al., introduced in Section 2.4, is one of the few works that address rail wear and its contribution to the deterioration of the rail's structural integrity. Jeong et al. use a simplified geometric block representation of the rail, and develops uncoupled side-wear and top-wear wear limits based on mechanics of materials calculations for a beam on a continuous elastic foundation. This work expands the capability of a rail failure model that is coupled with rail wear. Instead of using block geometry to represent the rail, the rail's worn profile is more accurately represented and tied into the wear inspection equipment currently in use through a profile fitting program. This eliminates the need for uncoupled side-wear and top-wear limits, and represents the true cross-section of the rail. Instead of using mechanics equations for a beam on an elastic foundation and uncoupled lateral and vertical load applications, the state of stress and strain at the assumed flaw front is resolved using a three-dimensional finite element model and coupled lateral and vertical wheel loads applied through contact with a deformable wheel similar to the model developed by Rice and Rungta [8]. A thermal load is also applied to represent the temperature fluctuation that a rail undergoes in a 24-hour period. The result is a powerful finite element model that represents all of the loading necessary for a fatigue or fracture-based life estimate calculation. The fatigue life of the rail is estimated using a strain-life formulation and the results from the finite element analysis. Multiple crack sizes are modeled in a given rail profile to populate a crack-growth curve. This crack-growth curve is used to develop an equation for the crack-growth rate for the detail fracture in the rail head.

The results of multiple analyses are formulated into a series of three response surfaces that can be run in real-time with the Track Geometry car. These response surfaces are functions of independent geometry

variables that are calculated on the fly for a given worn profile provided by the Track Geometry car, independent load variables that are known for the track, and an independent material variable that is also known for the section of track. After the analyses are completed for the generation of the response surfaces, the functional relationship developed by the response equations eliminate the need to run analyses for each worn profile and loading of interest. The response surfaces form a continuous solution for each of the independent variables that interpolated between known maximum and minimum values. The independent variables only need to be developed for the given worn geometry, loading, and rail material, and the remaining fatigue life and crack-growth rate can be estimated. After a verification and validation process, this tool could replace the current wear limits implemented by the railroad companies in America in the evaluation of worn rails.

10.4 Recommendations for Future Work

The next generation of the response surfaces should address the problems discussed in Chapter 9. In order to correct the limitations associated with the representation of the stress/strain field and singularity at the crack edge, it is suggested that the fracture mechanics tools available in Abaqus are used. By solving the problem using fracture mechanics techniques, the need for a fatigue post-processor would be eliminated, and the assumptions associated with the development of the fatigue material constants would be removed. However, this approach would require additional material properties that are currently unavailable. The suggested technique for developing the next generation of the rail failure response surfaces is explained below.

A similar approach to the one outlined in this thesis will be adopted in the development of the next generation of the rail failure response surfaces. Multiple Abaqus models will be developed for the given worn rail profiles, and multiple cracks will be modeled in each worn profile. In order to correctly represent the singularity at the crack edge, the crack modeling technique developed during this project will be changed substantially. Instead of representing the crack by tying two partitioned worn rail sections together, the crack will be modeled by intersecting one long rail section with an elliptical 2-D part representing the shape of the crack. The 2-D ellipse will be positioned in the rail head using the dimensions shown in Figure 5.3. Using XFEM in Abaqus, this 2-D ellipse will be selected as the crack, and the rail head will be selected as the enriched area. The mesh of the rail head is independent of the crack geometry, and will therefore not change for the different crack sizes. This will allow for a uniform mesh size to be developed in the rail head that correctly distributes the wheel load into the rail. The element size in the rail head will be determined using convergence studies and will be adopted in each worn rail model. The rail head is meshed with quadratic bricks (C3D20). The adoption of the new

meshing technique will eliminate the mesh inconsistency at the point of contact between the rail and the wheel, and will therefore reduce the variation in the results.

The stress intensity field output will be requested for the enriched area that includes the elements intersected by the crack edge. Instead of using a strain-based fatigue post-processor to calculate the fatigue life of the rail using the stress and strain results from the model, the remaining life of the rail will be evaluated using the stress intensity at the crack edge and Paris law given in Equation 7.1. Therefore, the stress and strain at the crack edge will no longer be directly used in the analysis. This should reduce the number of elements necessary to achieve a converged solution because the stress intensity converges more quickly than the stress and strain at the crack edge. In order to verify the effectiveness of the XFEM method, additional convergence studies should be completed for the stress intensity at the crack edge. In order to use Paris law, the crack-growth constants C and m must be characterized for the given material and stress ratio R. It is suggested that these constants be developed for each material of interest from testing of multiple compact specimens machined from actual rails and fitting the ΔK versus da/dN curve. If desired, these constants can also be borrowed from multiple technical works mentioned in Chapter 2. The maximum and minimum stress intensity factors for a given load cycle and crack size, assuming a constant stress ratio, can be evaluated in Abaqus to provide the effective ΔK. In order to use linear elastic fracture mechanics, only linear elastic material will be defined in Abaqus. Therefore Small Scale Yielding is assumed where the non-linear effects at the crack tip are contained within a small plastic region. Using the crack-growth constants and the stress intensity range, the crack-growth rate can be calculated using Paris law. The number of cycles to grow the crack to the given crack size can also be calculated by inverting the equation and integrating from zero to the given crack size as shown in Equation 2.6. By completing this procedure for the five crack sizes for each worn profile, the crack-growth curve can be populated for each profile and load history. This method eliminates the need for a fatigue post-processor and the inaccuracy present in the current material model, but also adds complications to the analysis procedure.

In order to complete the analysis procedure, every load variable must be varied in the Abaqus model in addition to the geometry variables. This increases the minimum number of models by a factor of 2^3 or 8. In addition, the load history is no longer created in FE-Safe. The user must use rain-flow cycle counting to create the equivalent load history and run the necessary load cases in Abaqus to quantify the stress intensity ranges of interest. However, the resulting model will characterize the propagation of the crack using a true fracture mechanics approach based on the stress intensity at the crack edge and will eliminate the mesh inconsistencies and material assumptions burdening the current model. The end result should be a model that provides a more accurate and stable solution.

In addition, the method for developing the stochastic independent load variables used in the first generation of the response surfaces should be evaluated before the second generation is developed. The distribution fitting method of determining the maximum vertical load is conservative in nature, and likely to cause low life values that would end up requiring an increase in rail replacement costs. However, once the representation of the detail fracture and the meshing of the rail have been updated, a change in the loading definition would be a minor change for the finite element model.

References

[1] Norfolk Southern. *MW&S Standard Procedure: Rail Wear Limits.* Norfolk, VA: File# 123-168, 2005.

[2] Woody, Sean S. "Discussions Regarding Norfolk Southern Corporation." Personal Interviews by Chase Holland.

[3] SperryRail.com, "Rail Testing," http://www.sperryrail.com/railtesting.html.

[4] American Railway Engineering and Maintenance-of-Way Association. "AREMA Manual for Railway Engineering." AREMA 1, no. 4 (2005): 4.i-4.4.24

[5] ZETA-TECH. *Application of ZETA-TECH's Risk-Based Ultrasonic Rail Test Scheduling Software (RailTest^{TM}) on the Norfolk Southern Rail Network: Report.* Cherry Hill, NJ: Harsco Rail Business Unit, 2010.

[6] Dolwing, Norman E. *Mechanical Behavior of Materials: Engineering Methods for Deformation, Fracture and Fatigue.* 3rd ed. Upper Saddle River, NJ: Prentice Hall, 2007.

[7] Orringer, O., Y.H. Tang, J.E. Gordon, D.Y. Jeong, J.M. Morris, and A.B. Permlamn, "Crack Propagation Life of Detail Fractures in Rails," Final Report, DOT/FRA/ORD-99/13, 1988.

[8] Rice, R. C. and R. Rungta. "Modeling of Embedded Fatigue Cracks in Railroad Rails," *Fatigue Fract. Engng Mater. Struct.* 11 no. 3 (1988): 189-203.

[9] Rail Defect Identification Guide. Research and Tests Laboratory. Southern Railway System. 1972.

[10] Andreikiv, A. E., E. A. Shchur, and A. I. Darchuk. "Prediction of the Life of Railroad Rails Under Service Conditions," *Fiziko-Khimicheskaya Mekhanika Materialov* 24, no. 2 (April 1988): 88-91.

[11] Jablonksi, D., Y.H. Tang, and R.M. Pelloux. "Simulation of Railroad Crack Growth Life Using Laboratory Specimens," *Theoretical and Applied Fracture Mechanics*, no. 14 (1990): 27-36.

[12] Tehrani, Hosseini P. and M. Saket. "Fatigue crack Initiation Life Prediction of Railroad," Paper presented at 7th International Conference on Modern Practices in Stress and Vibration Analysis, 2009.

[13] Plu, J., S. Bondeux, D. Boulanger, R. Heyder. "Application of Fracture Mechanics Methods to Rail Design and Maintenance," *Engineering Fracture Mechanics*, no. 76 (2009): 2602-2611.

[14] Ravaee, R. and A. Hassani. "Fracture Mechanics Determinations of Allowable Crack Size in Railroad Rails," *J Fail. Anal. And Preven.*, no. 7 (2007): 305-310 .

[15] Jeong, D.Y., Y.H. Tang, and O. Orringer. "Estimation of Rail Wear Limits Based on Rail Strength Investigations," Volpe Center Final Report, DOT/FRA/ORD-98/07, 1998.

[16] Jeong, D.Y. "Analytical Modeling of Rail Defects and Its Applications to Rail Defect Management," Volpe Center Technical Report for the UIC/WEC Joint Research Project of Rail Defect Management, 2003.

[17] Cook, Robert D., David S. Malkus, Michael E. Plesha, and Robert J. Witt. *Concepts and Applications of Finite Element Analysis.* 4th ed. Hoboken, NJ: John Wiley & Sons, Inc., 2002.

[18] "MATLAB User Manual" ver. R2011a, ©MATLAB. 2011.

[19] Shigley, Joseph E., Charles R. Mischke, Richard G. Budynas. *Mechanical Engineering Design.* 7th ed. New York: McGraw Hill, 2004.

[20] EngineersHandbook.com, "Coefficient of Friction Reference Table," http://www.engineershandbook.com/Tables/frictioncoefficients.htm.

[21] Hakergard, G. and T. Mann. "Neuber Prediction of Elastic-plastic Strain Concentration in Notched Tensile Specimens Under Large-Scale Yielding," *Strain Analysis* 38, no. 1 (2003): 79-94.

[22] "FE-Safe Fatigue Theory Reference Manual" ver. 5.4-06, ©safetechnology. 2010.

[23] "Abaqus Theory Manual" ver. 6.10, ©Abaqus, Inc. 2010.

[24] Box, George E. P., Stuart J. Hunter, William G, Hunter. *Statistics for Experimenters: Design, Innovation, and Discovery.* 2nd ed. Hoboken, NJ: Wiley-Interscience, 2005.

[25] Boller, C. and T. Seeger. "Materials Data for Cyclic Loading," *Elsevier Science Publishers*, ISBN 0-444-42875-5, 1987.

[26] Kandil, F., M. Brown, and K Miller. "Biaxial Low Cycle Fatigue Fracture of 316 Stainless Steel at Elevated Temperatures," *The Metals Society*, Book 280, 1982.

Appendix A. Rail Section Properties from Jeong et al. [15]

*Uses different xyz axis system
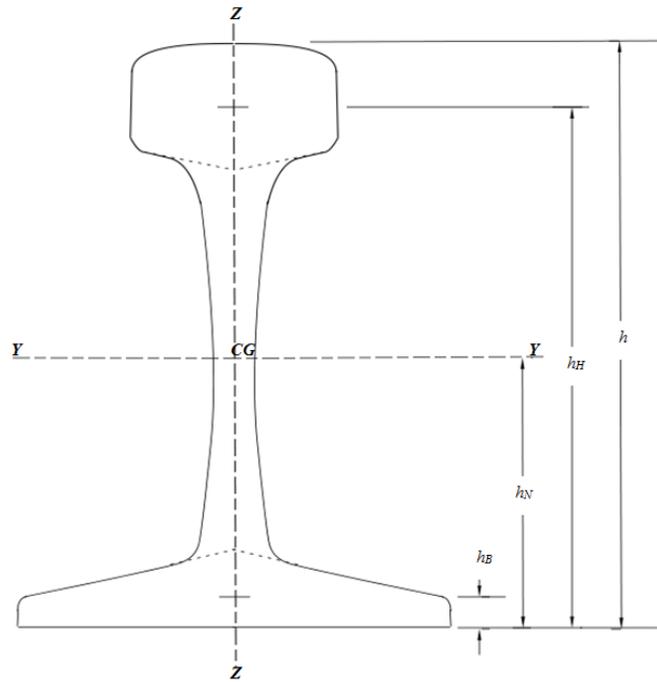
Figure A.1 Dimensions for generic rail section.

NOMENCLATURE:

Rail Dimensions

h Total rail height
h_H Distance from bottom of the rail to the centroid of the rail head
h_N Distance from the bottom of the rail to the centroid of the entire rail
h_B Distance from the bottom of the rail to the centroid of the rail base
w_H Rail head width

Cross-Sectional Areas

A_R Cross-sectional area of the entire rail
A_H Cross-sectional area for the rail head only
A_W Cross-sectional area for the rail web only
A_B Cross-sectional area for the rail base only

Second Area Moments of Inertia

I_{yy} Vertical bending inertia for the entire rail
I_{zz} Lateral bending inertia for the entire rail
I_{yyH} Vertical bending inertia for the rail head only
I_{zzH} Lateral bending inertia for the rail head only

Table A.1 Section properties for new or unworn rails.

	70 ASCE	85 ASCE	100 RE	115 RE	132 RE	136 RE	140 RE
h (in)	4.625	5.188	6.00	6.625	7.125	7.313	7.313
h_H (in)	4.01	4.46	5.23	5.80	6.3	6.39	6.35
h_N (in)	2.22	2.47	2.75	2.98	3.2	3.35	3.37
h_B (in)	0.299	0.321	0.394	0.411	0.436	0.435	0.436
w_H (in)	2.4375	2.5625	2.6875	2.7188	3.00	2.9375	3.00
A_R (in^2)	6.81	8.33	9.95	11.26	12.95	13.35	13.8
A_H (in^2)	2.81	3.49	3.80	3.91	4.42	4.86	5.00
A_W (in^2)	1.41	2.25	2.23	3.16	3.67	3.62	3.94
A_B (in^2)	2.59	2.59	3.92	4.19	4.86	4.87	4.86
I_{yy} (in^4)	19.7	30.07	49.00	65.6	88.1	94.9	96.8
I_{zz} (in^4)	4.86	6.95	9.35	10.4	14.2	14.5	14.7
I_{yyH} (in^4)	0.329	0.558	0.714	0.729	0.837	1.17	1.38
I_{zzH} (in^4)	1.24	1.75	2.12	2.13	2.84	3.03	3.14

	70 ASCE	85 ASCE	100 RE	115 RE	132 RE	136 RE	140 RE
h (cm)	11.75	13.18	15.24	16.83	18.10	18.58	18.58
h_H (cm)	10.19	11.33	13.28	14.73	16.00	16.23	16.13
h_N (cm)	5.64	6.27	6.99	7.57	8.13	8.51	8.56
h_B (cm)	0.76	0.82	1.00	1.04	1.11	1.10	1.11
w_H (cm)	6.19	6.51	6.83	6.91	7.62	7.46	7.62
A_R (cm^2)	43.94	53.74	64.19	72.65	83.55	86.13	89.03
A_H (cm^2)	18.13	22.52	24.52	25.23	28.52	31.35	32.26
A_W (cm^2)	9.10	14.52	14.39	20.39	23.68	23.35	25.42
A_B (cm^2)	16.71	16.71	25.29	27.03	31.35	31.42	31.35
I_{yy} (cm^4)	819.98	1251.61	2039.53	2730.48	3667.00	3950.04	4029.12
I_{zz} (cm^4)	202.29	289.28	389.18	432.88	591.05	603.54	611.86
I_{yyH} (cm^4)	13.69	23.23	29.72	30.34	34.84	48.70	57.44
I_{zzH} (cm^4)	51.61	72.84	88.24	88.66	118.21	126.12	130.70

Appendix B.1 MATLAB Profile Mapping Code

Main Program:

```
clear all
clc
%% Import Data and set up program:
xdata =(xlsread('Full136RERail.xlsx', 1, 'B11:B232'));
%xlsread('Name.xlsx',sheet#, 'range')
xdata = xdata';
ydata =(xlsread('Full136RERail.xlsx', 1, 'D11:D232'));
ydata = ydata';
xdatametric = 2.54.*xdata;
ydatametric = 2.54.*ydata;
% xdata =(xlsread('SideWear136RERail.xlsx', 10, 'F11:F203'));
%xlsread('Name.xlsx',sheet#, 'range')
% xdata = xdata';
% ydata =(xlsread('SideWear136RERail.xlsx', 10, 'H11:H203'));
% ydata = ydata';
plot(xdata,ydata, 'o');
grid on
xlabel('X');
ylabel('Y');
title('Track Geometry Car data')
NSamples = length(xdata);
NElements = 4;
NNodes = 13;
Ng = zeros(NSamples,NNodes);
%% Devide Data into Four Elements:
[E1xdata, E1ydata, E2xdata, E2ydata, E3xdata, E3ydata, E4xdata, E4ydata] =
DefineElementalVectors(xdata, ydata, NSamples);
%Plot element breaks:
E1xdatametric = 2.54.*E1xdata;
E2xdatametric = 2.54.*E2xdata;
E3xdatametric = 2.54.*E3xdata;
E4xdatametric = 2.54.*E4xdata;
E1ydatametric = 2.54.*E1ydata;
E2ydatametric = 2.54.*E2ydata;
E3ydatametric = 2.54.*E3ydata;
E4ydatametric = 2.54.*E4ydata;
figure
plot(E1xdata, E1ydata, 'o',E2xdata, E2ydata, 'o',E3xdata, E3ydata,
'o',E4xdata, E4ydata, 'o');
grid on
xlabel('X');
ylabel('Y');
legend('Element 1', 'Element 2', 'Element 3', 'Element 4')
hold off
[assyvrE1,assyvrE2,assyvrE3,assyvrE4] = DefineAssemblyVectors(xdata, ydata,
NSamples);
for k = 1:NElements
    if k == 1
        order = 4;
        nSamples = length(E1xdata);
        assyvr = assyvrE1;
        assyvc = [1;2;3;4];
```

```
        elseif k == 2
            order = 4;
            nSamples = length(E2xdata);
            assyvr = assyvrE2;
            assyvc = [4;5;6;7];
        elseif k == 3
            order = 4;
            nSamples = length(E3xdata);
            assyvr = assyvrE3;
            assyvc = [7;8;9;10];
        elseif k == 4
            order = 4;
            nSamples = length(E4xdata);
            assyvr = assyvrE4;
            assyvc = [10;11;12;13];
        end
%% Compute N Matrix:
Ne = zeros(nSamples,order);
if order == 2
    for l = 1:nSamples
        zeta = EstimateZeta(l,nSamples);
        [N1,N2] = LinearInterpolationFunctions(zeta);
        Ne(l,1) = N1;
        Ne(l,2) = N2;
    end
elseif order == 4
    for l = 1:nSamples
        zeta = EstimateZeta(l,nSamples);
        [N1,N2,N3,N4] = CubicInterpolationFunctions(zeta);
        Ne(l,1) = N1;
        Ne(l,2) = N2;
        Ne(l,3) = N3;
        Ne(l,4) = N4;
    end
end
%% Assemble Elemental Interpolation Matrices into Global
[Ng] = AssembleGlobalInterpolationMatrix(Ng, Ne, assyvr, assyvc);
end
%% Partition Global Matrix (X7 = 0 specified) (Y7 not known)
Ngpartition(1:NSamples, 1:6)= Ng(1:NSamples, 1:6);
Ngpartition(1:NSamples, 7:12) = Ng(1:NSamples, 8:13);

%% Solve for X1, X2, X3, X4, X5, X6, X8, X9, X10, X11, X12, X13: (X7=0)
B = Ngpartition'*Ngpartition;
Xpartition = B\Ngpartition'*xdata';
X(1:6, 1) = Xpartition(1:6);
X(7, 1) = 0;
X(8:13, 1) = Xpartition(7:12);
Xmetric = 2.54.*X;
%X = inv(B)*N'*xdata';
%% Solve for Y1, Y2, Y3, Y4, Y5, Y6, Y7:
B = Ng'*Ng;
Y = B\Ng'*ydata';
Ymetric = 2.54.*Y;
%% Plot Data and Fit
syms z;
zrange = -1:0.01:1;
```

```
figure
plot1 = plot(X,Y,'d'); hold on
set(plot1,'Color','red','LineWidth',2);
for k = 1:NElements
    if k == 1
        [XFit1] = XCubicFitEquation(X(1,1),X(2,1),X(3,1),X(4,1));
        [YFit1] = YCubicFitEquation(Y(1,1),Y(2,1),Y(3,1), Y(4,1));
        p = plot(subs(XFit1,z,zrange), subs(YFit1,z,zrange));
    elseif k == 2
        [XFit2] = XCubicFitEquation(X(4,1),X(5,1),X(6,1),X(7,1));
        [YFit2] = YCubicFitEquation(Y(4,1),Y(5,1),Y(6,1),Y(7,1));
        p1 = plot(subs(XFit2,z,zrange), subs(YFit2,z,zrange));
    elseif k == 3
        [XFit3] = XCubicFitEquation(X(7,1),X(8,1),X(9,1),X(10,1));
        [YFit3] = YCubicFitEquation(Y(7,1),Y(8,1),Y(9,1),Y(10,1));
        p2 = plot(subs(XFit3,z,zrange), subs(YFit3,z,zrange));
    elseif k == 4
        [XFit4] = XCubicFitEquation(X(10,1),X(11,1),X(12,1),X(13,1));
        [YFit4] = YCubicFitEquation(Y(10,1),Y(11,1),Y(12,1),Y(13,1));
        p3 = plot(subs(XFit4,z,zrange), subs(YFit4,z,zrange));
    end
end
set(p,'Color','red','LineWidth',2);
set(p1,'Color','red','LineWidth',2);
set(p2,'Color','red','LineWidth',2);
set(p3,'Color','red','LineWidth',2);
plot(xdata, ydata, 'o');
grid on
title('Data vs. Fit Line');
legend('Nodes','fit');
xlabel('X');
ylabel('Y');
hold off
for k = 1:NElements
    if k == 1
        figure
        plot(zrange, subs(XFit1,z,zrange));
        hold on
        plot(zrange, subs(YFit1,z,zrange),'r');
        grid on
        title('X and Y Fit Equations');
        legend('XFit','YFit');
        xlabel('zeta');
        ylabel('fit values');
        hold off
    elseif k == 2
        figure
        plot(zrange, subs(XFit2,z,zrange));
        hold on
        plot(zrange, subs(YFit2,z,zrange),'r');
        grid on
        title('X and Y Fit Equations');
        legend('XFit','YFit');
        xlabel('zeta');
        ylabel('fit values');
        hold off
    elseif k == 3
```

```matlab
            figure
            plot(zrange, subs(XFit3,z,zrange));
            hold on
            plot(zrange, subs(YFit3,z,zrange),'r');
            grid on
            title('X and Y Fit Equations');
            legend('XFit','YFit');
            xlabel('zeta');
            ylabel('fit values');
            hold off
        elseif k == 4
            figure
            plot(zrange, subs(XFit4,z,zrange));
            hold on
            plot(zrange, subs(YFit4,z,zrange),'r');
            grid on
            title('X and Y Fit Equations');
            legend('XFit','YFit');
            xlabel('zeta');
            ylabel('fit values');
            hold off
        end
end
%% Calculate xj error:
Error_x = zeros(NSamples,1);
ylimit = max(ydata);
xcenter = 0;
for k = 1:NSamples
     if (xdata(1,k) <= xcenter)
        if (ydata(1,k) <= (0.93*ylimit))
            X1 = X(1,1);
            X2 = X(2,1);
            X3 = X(3,1);
            X4 = X(4,1);
            zeta = EstimateZeta(k,length(E1xdata));
            [N1,N2,N3,N4] = CubicInterpolationFunctions(zeta);
            %Calculate x as a function of zetaj
            Xofzeta = N1*X1 + N2*X2 + N3*X3 + N4*X4;
            error_xi = Xofzeta - xdata(1,k);   %X Residual Error
            Error_x(k,1) = error_xi;   %Vector of the x residuals
        end
        if (ydata(1,k) >= (0.93*ylimit))
            X4 = X(4,1);
            X5 = X(5,1);
            X6 = X(6,1);
            X7 = X(7,1);
            zeta = EstimateZeta(k-length(E1xdata),length(E2xdata));
            [N1,N2,N3,N4] = CubicInterpolationFunctions(zeta);
            %Calculate x as a function of zetaj
            Xofzeta = N1*X4 + N2*X5 + N3*X6 + N4*X7;
            error_xi = Xofzeta - xdata(1,k);   %X Residual Error
            Error_x(k,1) = error_xi;   %Vector of the x residuals
        end
    elseif (xdata(1,k) >= xcenter)
        if (ydata(1,k) >= (0.93*ylimit))
            X7 = X(7,1);
            X8 = X(8,1);
```

```
            X9 = X(9,1);
            X10 = X(10,1);
            zeta = EstimateZeta(k-length(E2xdata)-
length(E1xdata),length(E3xdata));
            [N1,N2,N3,N4] = CubicInterpolationFunctions(zeta);
            %Calculate x as a function of zetaj
            Xofzeta = N1*X7 + N2*X8 + N3*X9 + N4*X10;
            error_xi = Xofzeta - xdata(1,k);  %X Residual Error
            Error_x(k,1) = error_xi; %Vector of the x residuals
        end
        if (ydata(1,k) <= (0.93*ylimit))
            X10 = X(10,1);
            X11 = X(11,1);
            X12 = X(12,1);
            X13 = X(13,1);
            zeta = EstimateZeta(k-length(E3xdata)-length(E2xdata)-
length(E1xdata),length(E4xdata));
            [N1,N2,N3,N4] = CubicInterpolationFunctions(zeta);
            %Calculate x as a function of zetaj
            Xofzeta = N1*X10 + N2*X11 + N3*X12 + N4*X13;
            error_xi = Xofzeta - xdata(1,k);  %X Residual Error
            Error_x(k,1) = error_xi;  %Vector of the x residuals
        end
      end
end
%% Calculate yj error:
Error_y = zeros(NSamples,1);
for k = 1:NSamples
    if (xdata(1,k) <= xcenter)
        if (ydata(1,k) <= (0.93*ylimit))
            Y1 = Y(1,1);
            Y2 = Y(2,1);
            Y3 = Y(3,1);
            Y4 = Y(4,1);
            zeta = EstimateZeta(k,length(E1ydata));
            [N1,N2,N3,N4] = CubicInterpolationFunctions(zeta);
            %Calculate x as a function of zetaj
            Yofzeta = N1*Y1 + N2*Y2 + N3*Y3 + N4*Y4;
            error_yi = Yofzeta - ydata(1,k);  %Y Residual Error
            Error_y(k,1) = error_yi;  %Vector of the y residuals
        end
        if (ydata(1,k) >= (0.93*ylimit))
            Y4 = Y(4,1);
            Y5 = Y(5,1);
            Y6 = Y(6,1);
            Y7 = Y(7,1);
            zeta = EstimateZeta(k-length(E1ydata),length(E2ydata));
            [N1,N2,N3,N4] = CubicInterpolationFunctions(zeta);
            %Calculate y as a function of zetaj
            Yofzeta = N1*Y4 + N2*Y5 + N3*Y6 + N4*Y7;
            error_yi = Yofzeta - ydata(1,k);  %Y Residual Error
            Error_y(k,1) = error_yi;  %Vector of the y residuals
        end
    elseif (xdata(1,k) >= xcenter)
        if (ydata(1,k) >= (0.93*ylimit))
            Y7 = Y(7,1);
            Y8 = Y(8,1);
```

```matlab
            Y9 = Y(9,1);
            Y10 = Y(10,1);
            zeta = EstimateZeta(k-length(E2ydata)-
length(E1ydata),length(E3ydata));
            [N1,N2,N3,N4] = CubicInterpolationFunctions(zeta);
            %Calculate y as a function of zetaj
            Yofzeta = N1*Y7 + N2*Y8 + N3*Y9 + N4*Y10;
            error_yi = Yofzeta - ydata(1,k);  %Y Residual Error
            Error_y(k,1) = error_yi;  %Vector of the y residuals
         end
         if (ydata(1,k) <= (0.93*ylimit))
            Y10 = Y(10,1);
            Y11 = Y(11,1);
            Y12 = Y(12,1);
            Y13 = Y(13,1);
            zeta = EstimateZeta(k-length(E3ydata)-length(E2ydata)-
length(E1ydata),length(E4ydata));
            [N1,N2,N3,N4] = CubicInterpolationFunctions(zeta);
            %Calculate y as a function of zetaj
            Yofzeta = N1*Y10 + N2*Y11 + N3*Y12 + N4*Y13;
            error_yi = Yofzeta - ydata(1,k);  %Y Residual Error
            Error_y(k,1) = error_yi;  %Vector of the y residuals
         end
      end
end
%% Plot Residuals:
figure;
plot(Error_x, '.'); %Plot of x Residuals
hold on
p=plot(length(E1xdata),0,'d');
p2=plot(length(E2xdata)+length(E1xdata),0,'d');
p3=plot(length(E2xdata)+length(E1xdata)+length(E3xdata),0,'d');
set(p,'Color','red','LineWidth',2);
set(p2,'Color','red','LineWidth',2);
set(p3,'Color','red','LineWidth',2);
hold off
xlabel('Sample #');
ylabel('Error');
title('X Error for Fit');
grid on
figure
plot(Error_y, '*'); %Plot of y Residuals
hold on
p=plot(length(E1ydata),0,'d');
p2=plot(length(E2ydata)+length(E1ydata),0,'d');
p3=plot(length(E2ydata)+length(E1ydata)+length(E3ydata),0,'d');
set(p,'Color','red','LineWidth',2);
set(p2,'Color','red','LineWidth',2);
set(p3,'Color','red','LineWidth',2);
hold off
xlabel('Sample #');
ylabel('Error');
title('Y Error for Fit');
grid on
%% Calculate statistical properties for X and Y using Matlab functions:
xME = mean(Error_x);
yME = mean(Error_y);
```

```matlab
EEV_X = var(Error_x);
EEV_Y = var(Error_y);
ESD_X = std(Error_x);
ESD_Y = std(Error_y);
SNR_X = max(abs(xdata))/ESD_X;
SNR_Y = max(abs(ydata))/ESD_Y;
%% Curve Fit Rail Web
RailWebCurveFit
%% Find Intersection between Rail Head Fit and Web Fit
figure
plot(X,Y,'*'); hold on
for k = 1:NElements
    if k == 1
        [XFit1] = XCubicFitEquation(X(1,1),X(2,1),X(3,1),X(4,1));
        [YFit1] = YCubicFitEquation(Y(1,1),Y(2,1),Y(3,1), Y(4,1));
        plot(subs(XFit1,z,zrange), subs(YFit1,z,zrange));
    elseif k == 2
        [XFit2] = XCubicFitEquation(X(4,1),X(5,1),X(6,1),X(7,1));
        [YFit2] = YCubicFitEquation(Y(4,1),Y(5,1),Y(6,1),Y(7,1));
        plot(subs(XFit2,z,zrange), subs(YFit2,z,zrange));
    elseif k == 3
        [XFit3] = XCubicFitEquation(X(7,1),X(8,1),X(9,1),X(10,1));
        [YFit3] = YCubicFitEquation(Y(7,1),Y(8,1),Y(9,1),Y(10,1));
        plot(subs(XFit3,z,zrange), subs(YFit3,z,zrange));
    elseif k == 4
        [XFit4] = XCubicFitEquation(X(10,1),X(11,1),X(12,1),X(13,1));
        [YFit4] = YCubicFitEquation(Y(10,1),Y(11,1),Y(12,1),Y(13,1));
        plot(subs(XFit4,z,zrange), subs(YFit4,z,zrange));
    end
end
plot(Xweb,Yweb,'*');
plot(-Xweb,Yweb,'*');
plot(subs(WebXFitLeft,z,zrange), subs(WebYFitLeft,z,zrange));
plot(subs(WebXFitRight,z,zrange), subs(WebYFitRight,z,zrange));
grid on
title('Rail Fit');
legend('Nodes','fit');
xlabel('X');
ylabel('Y');
hold off
syms n;
Equation1 = subs(WebXFitLeft, z, n) - XFit1;
Equation2 = subs(WebYFitLeft, z, n) - YFit1;
LeftIntersect = solve(Equation1, Equation2, 'z', 'n');
Equation1 = subs(WebXFitRight, z, n) - XFit4;
Equation2 = subs(WebYFitRight, z, n) - YFit4;
RightIntersect = solve(Equation1, Equation2, 'z', 'n');
zleft = eval(LeftIntersect.z(1,1)); % Depends on where Element is intersected
nleft = eval(LeftIntersect.n(1,1)); % Depends on where Element is intersected
zright = eval(RightIntersect.z(1,1)); %Depends on where Element is
intersected
nright = eval(RightIntersect.n(1,1)); % Depends on where Element is
intersected
%Plot completed cross-section
figure
zrange1 = -1:0.01:nleft;
zrange2 = zleft:0.01:1;
```

```
zrange3 = -1:0.01:zright;
zrange4 = nright:0.01:1;
for k = 1:NElements
    if k == 1
        [XFit1] = XCubicFitEquation(X(1,1),X(2,1),X(3,1),X(4,1));
        [YFit1] = YCubicFitEquation(Y(1,1),Y(2,1),Y(3,1), Y(4,1));
        plot(subs(XFit1,z,zrange2), subs(YFit1,z,zrange2));
        hold on
    elseif k == 2
        [XFit2] = XCubicFitEquation(X(4,1),X(5,1),X(6,1),X(7,1));
        [YFit2] = YCubicFitEquation(Y(4,1),Y(5,1),Y(6,1),Y(7,1));
        plot(subs(XFit2,z,zrange), subs(YFit2,z,zrange));
    elseif k == 3
        [XFit3] = XCubicFitEquation(X(7,1),X(8,1),X(9,1),X(10,1));
        [YFit3] = YCubicFitEquation(Y(7,1),Y(8,1),Y(9,1),Y(10,1));
        plot(subs(XFit3,z,zrange), subs(YFit3,z,zrange));
    elseif k == 4
        [XFit4] = XCubicFitEquation(X(10,1),X(11,1),X(12,1),X(13,1));
        [YFit4] = YCubicFitEquation(Y(10,1),Y(11,1),Y(12,1),Y(13,1));
        plot(subs(XFit4,z,zrange3), subs(YFit4,z,zrange3));
    end
end
plot(subs(WebXFitLeft,z,zrange1), subs(WebYFitLeft,z,zrange1));
plot(subs(WebXFitRight,z,zrange4), subs(WebYFitRight,z,zrange4));
hold off
grid on
title('Rail Cross Section');
legend('fit');
xlabel('X');
ylabel('Y');
%% Calculate section properties:
[A, Ixx, Iyy] = SectionProperties(webydata, ydata, zleft, nleft, zright,
nright, XFit1, YFit1, XFit2, YFit2, XFit3, YFit3, XFit4, YFit4, WebXFitLeft,
WebYFitLeft, WebXFitRight, WebYFitRight);
h = max(ydata)+ 1.1875;
%% Reassemble overall vector of Nodal Coordinates:
%For X
XTotal(1:3,1)= Xweb(1:3,1);
XTotal(4,1)= subs(XFit1, z, zleft);
XTotal(5:15,1) = X(2:12,1);
XTotal(16,1) = subs(XFit4, z, zright);
XTotal(17,1) = -Xweb(3,1);
XTotal(18,1) = -Xweb(2,1);
XTotal(19,1) = -Xweb(1,1);
%For Y
YTotal(1:3,1)= Yweb(1:3,1);
YTotal(4,1)= subs(YFit1, z, zleft);
YTotal(5:15,1) = Y(2:12,1);
YTotal(16,1) = subs(YFit4, z, zright);
YTotal(17,1) = Yweb(3,1);
YTotal(18,1) = Yweb(2,1);
YTotal(19,1) = Yweb(1,1);
figure
plot(XTotal, YTotal, '*');
hold on
for k = 1:NElements
    if k == 1
```

```
            [XFit1] = XCubicFitEquation(X(1,1),X(2,1),X(3,1),X(4,1));
            [YFit1] = YCubicFitEquation(Y(1,1),Y(2,1),Y(3,1), Y(4,1));
            plot(subs(XFit1,z,zrange2), subs(YFit1,z,zrange2));
        elseif k == 2
            [XFit2] = XCubicFitEquation(X(4,1),X(5,1),X(6,1),X(7,1));
            [YFit2] = YCubicFitEquation(Y(4,1),Y(5,1),Y(6,1),Y(7,1));
            plot(subs(XFit2,z,zrange), subs(YFit2,z,zrange));
        elseif k == 3
            [XFit3] = XCubicFitEquation(X(7,1),X(8,1),X(9,1),X(10,1));
            [YFit3] = YCubicFitEquation(Y(7,1),Y(8,1),Y(9,1),Y(10,1));
            plot(subs(XFit3,z,zrange), subs(YFit3,z,zrange));
        elseif k == 4
            [XFit4] = XCubicFitEquation(X(10,1),X(11,1),X(12,1),X(13,1));
            [YFit4] = YCubicFitEquation(Y(10,1),Y(11,1),Y(12,1),Y(13,1));
            plot(subs(XFit4,z,zrange3), subs(YFit4,z,zrange3));
        end
end
plot(subs(WebXFitLeft,z,zrange1), subs(WebYFitLeft,z,zrange1));
plot(subs(WebXFitRight,z,zrange4), subs(WebYFitRight,z,zrange4));
hold off
grid on
title('Rail Cross Section');
legend('nodes', 'fit');
xlabel('X');
ylabel('Y');
```

Sub-Functions and Scripts (in order of being called in Main Program):

1.

```
function [E1xdata, E1ydata, E2xdata, E2ydata, E3xdata, E3ydata, E4xdata,
E4ydata] = DefineElementalVectors(xdata, ydata, NSamples)
% Devide data into 4 Elemental data vectors
j = 1; %length of E1 vector
k = 1; %length of E2 vector
l = 1; %length of E3 vector
m = 1; %length of E4 vector
ylimit = max(ydata);
xcenter = 0;
for i=1:NSamples
    if (xdata(1,i) <= xcenter)
        if (ydata(1,i) <= (0.93*ylimit))
            E1xdata(1,j) = xdata(1,i);
            E1ydata(1,j) = ydata(1,i);
            j = j + 1;
        end
        if (ydata(1,i) >= (0.93*ylimit))
            E2xdata(1,k) = xdata(1,i);
            E2ydata(1,k) = ydata(1,i);
            k = k + 1;
        end
    elseif (xdata(1,i) >= xcenter)
        if (ydata(1,i) >= (0.93*ylimit))
            E3xdata(1,l) = xdata(1,i);
            E3ydata(1,l) = ydata(1,i);
```

```
                l = l + 1;
        end
        if (ydata(1,i) <= (0.93*ylimit))
            E4xdata(1,m) = xdata(1,i);
            E4ydata(1,m) = ydata(1,i);
            m = m + 1;
        end
    end
end
```

2.

```
function [assyvrE1,assyvrE2,assyvrE3,assyvrE4] = DefineAssemblyVectors(xdata,
ydata, NSamples)
% Devide data into 4 Elemental data vectors
assyvr = zeros(NSamples,1);
j = 1; %length of E1 vector
k = 1; %length of E2 vector
l = 1; %length of E3 vector
m = 1; %length of E4 vector
for i = 1:NSamples
    assyvr(i,1) = i;
end
ylimit = max(ydata);
xcenter = 0;
for i=1:NSamples
    if (xdata(1,i) <= xcenter)
        if (ydata(1,i) <= (0.93*ylimit))
            assyvrE1(1,j) = assyvr(i,1);
            j = j + 1;
        end
        if (ydata(1,i) >= (0.93*ylimit))
            assyvrE2(1,k) = assyvr(i,1);
            k = k + 1;
        end
    elseif (xdata(1,i) >= xcenter)
        if (ydata(1,i) >= (0.93*ylimit))
            assyvrE3(1,l) = assyvr(i,1);
            l = l + 1;
        end
        if (ydata(1,i) <= (0.93*ylimit))
            assyvrE4(1,m) = assyvr(i,1);
            m = m + 1;
        end
    end
end
```

3.

```
function [zeta] = EstimateZeta(i,NSamples)
%Estimates zeta based on sample number and overall number of
%samples for element
zeta = -1 + (i-1)*(2/(NSamples-1));
end
```

4.

```matlab
function [N1,N2] = LinearInterpolationFunctions(zeta)
% Provides Interpolation functions, N1, N2
N1 = 1/2*(1-zeta);
N2 = 1/2*(1+zeta);
end
```

5.

```matlab
function [N1,N2,N3,N4] = CubicInterpolationFunctions(zeta)
% Provides Interpolation functions, N1, N2, N3, N4
N1 = - (9*zeta^3)/16 + (9*zeta^2)/16 + zeta/16 - 1/16;
N2 = (27*zeta^3)/16 - (9*zeta^2)/16 - (27*zeta)/16 + 9/16;
N3 = - (27*zeta^3)/16 - (9*zeta^2)/16 + (27*zeta)/16 + 9/16;
N4 = (9*zeta^3)/16 + (9*zeta^2)/16 - zeta/16 - 1/16;
end
```

6. Originally written by Dr. Robert West, modified by Chase Holland

```matlab
function [Ng] = AssembleGlobalInterpolationMatrix(Ng, Ne, assyvr, assyvc)
%   Assemble Global interpolation Matrix for X by Direct Stiffness Method

%   Input:   Ng    - exisitng assembled global matrix
%            Ne    - elemental matrix
%            assyv - global assembly vector
%
%   Output:  Ng - Interpolation function matrix
%
NSamples = length(assyvr);
order = length(assyvc);
% do for each row dof -> row coefficient in element matrix
for row_id = 1: NSamples
   rdof_id = assyvr(row_id);            % get current assy row dof id
% do for each column dof -> column coefficient in element matrix
   for col_id = 1:order
      cdof_id = assyvc(col_id);         % get current assy column id
      % now assemble element interpolation func with global interpolation
      % func
      Ng(rdof_id, cdof_id) = Ng(rdof_id, cdof_id) + Ne(row_id, col_id);
   end;                                 % end do for each column dof
end;                                    % end do for each row dof
```

7.

```matlab
function [XFit] = XCubicFitEquation(X1,X2,X3,X4)
%Define fit function for x
syms z;
N1func = - (9*z^3)/16 + (9*z^2)/16 + z/16 - 1/16;
N2func = (27*z^3)/16 - (9*z^2)/16 - (27*z)/16 + 9/16;
```

```
N3func = - (27*z^3)/16 - (9*z^2)/16 + (27*z)/16 + 9/16;
N4func = (9*z^3)/16 + (9*z^2)/16 - z/16 - 1/16;
XFit = N1func*X1+N2func*X2+N3func*X3+N4func*X4;
End
```

8.

```
function [YFit] = YCubicFitEquation(Y1,Y2,Y3,Y4)
%Define fit function for x
syms z;
N1func = - (9*z^3)/16 + (9*z^2)/16 + z/16 - 1/16;
N2func = (27*z^3)/16 - (9*z^2)/16 - (27*z)/16 + 9/16;
N3func = - (27*z^3)/16 - (9*z^2)/16 + (27*z)/16 + 9/16;
N4func = (9*z^3)/16 + (9*z^2)/16 - z/16 - 1/16;
YFit = N1func*Y1+N2func*Y2+N3func*Y3+N4func*Y4;
end
```

9.

```
%% RailWebCurveFit script
%% Import Data and set up program:
webxdata =[-0.4278, -0.7450, -0.8462, -1.2168, -1.4534]';
webydata =[3.8438, 4.3534, 4.3991, 4.4917, 4.8027]';
%plot(webxdata,webydata, 'o');
NSamples = length(webxdata);
NNodes = 4;
%% Define Assembly Vectors:
order = 4;
assyvc = [1;2;3;4];
assyvr = zeros(NSamples,1);
for k = 1:NSamples
    assyvr(k,1) = k;
end
%% Compute N Matrix:
Ng = zeros(NSamples,NNodes);
    for l = 1:NSamples
        zeta = EstimateZeta(l,NSamples);
        [N1,N2,N3,N4] = CubicInterpolationFunctions(zeta);
        Ng(l,1) = N1;
        Ng(l,2) = N2;
        Ng(l,3) = N3;
        Ng(l,4) = N4;
    end
%% Partition Global matrix for known X1 and Y1
% (X1 = webxdata(1,1) = -0.4278, Y1 = webydata(1,1) = 3.8438)
Ngwebpartition(1:NSamples, 1:3) = Ng(1:NSamples, 2:4);
% subtract vector from x and y data
webxdatapartition = webxdata - webxdata(1,1)*Ng(1:NSamples,1);
webydatapartition = webydata - webydata(1,1)*Ng(1:NSamples,1);
%% Solve for X1, X2, X3, X4:
B = Ngwebpartition'*Ngwebpartition;
Xwebpartition = B\Ngwebpartition'*webxdatapartition;
Xweb(1,1) = webxdata(1,1);
Xweb(2:4, 1) = Xwebpartition(1:3, 1);
%X = inv(B)*N'*xdata';
```

```matlab
%% Solve for Y1, Y2, Y3, Y4:
Ywebpartition = B\Ngwebpartition'*webydatapartition;
Yweb(1,1) = webydata(1,1);
Yweb(2:4, 1) = Ywebpartition(1:3, 1);
%% Plot Data and Fit
syms z;
zrange = -1:0.01:1;
figure
plot(Xweb,Yweb,'*'); hold on
plot(-Xweb,Yweb,'*');
[WebXFitLeft] = XCubicFitEquation(Xweb(1,1),Xweb(2,1),Xweb(3,1),Xweb(4,1));
[WebYFitLeft] = YCubicFitEquation(Yweb(1,1),Yweb(2,1),Yweb(3,1), Yweb(4,1));
[WebXFitRight] = XCubicFitEquation(-Xweb(4,1),-Xweb(3,1),-Xweb(2,1),-
Xweb(1,1));
[WebYFitRight] = YCubicFitEquation(Yweb(4,1),Yweb(3,1),Yweb(2,1), Yweb(1,1));
plot(subs(WebXFitLeft,z,zrange), subs(WebYFitLeft,z,zrange));
plot(subs(WebXFitRight,z,zrange), subs(WebYFitRight,z,zrange));
plot(webxdata, webydata, '.');
plot(-webxdata, webydata, '.');
grid on
title('Data vs. Fit Line');
legend('Nodes');
xlabel('X');
ylabel('Y');
hold off
%Plot Data Fits x and y
figure
plot(zrange, subs(WebXFitLeft,z,zrange));
hold on
plot(zrange, subs(WebYFitLeft,z,zrange),'r');
grid on
title('X and Y Fit Equations');
legend('XFit','YFit');
xlabel('zeta');
ylabel('fit values');
hold off
%% Calculate xj error:
WebError_x = zeros(NSamples,1);
for k = 1:NSamples
    X1 = Xweb(1,1);
    X2 = Xweb(2,1);
    X3 = Xweb(3,1);
    X4 = Xweb(4,1);
    zeta = EstimateZeta(k,NSamples);
    [N1,N2,N3,N4] = CubicInterpolationFunctions(zeta);
    %Calculate x as a function of zetaj
    Xofzeta = N1*X1 + N2*X2 + N3*X3 + N4*X4;
    Weberror_xi = Xofzeta - webxdata(k,1);  %X Residual Error
    WebError_x(k,1) = Weberror_xi;  %Vector of the x residuals
end
%% Calculate yj error:
WebError_y = zeros(NSamples,1);
for k = 1:NSamples
    Y1 = Yweb(1,1);
    Y2 = Yweb(2,1);
    Y3 = Yweb(3,1);
    Y4 = Yweb(4,1);
```

137

```matlab
        zeta = EstimateZeta(k,NSamples);
        [N1,N2,N3,N4] = CubicInterpolationFunctions(zeta);
        %Calculate x as a function of zetaj
        Yofzeta = N1*Y1 + N2*Y2 + N3*Y3 + N4*Y4;
        Weberror_yi = Yofzeta - webydata(k,1);  %Y Residual Error
        WebError_y(k,1) = Weberror_yi;  %Vector of the y residuals
end
%% Plot Residuals:
figure;
plot(WebError_x, '.'); %Plot of x Residuals
xlabel('Sample #');
ylabel('Error');
title('X Error for Fit');
grid on
figure
plot(WebError_y, '*'); %Plot of y Residuals
xlabel('Sample #');
ylabel('Error');
title('Y Error for Fit');
grid on
%% Calculate statistical properties for x and y using Matlab functions:
WebxME = mean(WebError_x);
WebyME = mean(WebError_y);
WebEEV_X = var(WebError_x);
WebEEV_Y = var(WebError_y);
WebESD_X = std(WebError_x);
WebESD_Y = std(WebError_y);
WebSNR_X = max(abs(webxdata))/ESD_X;
WebSNR_Y = max(abs(webydata))/ESD_Y;
```

Appendix B.2 MATLAB Section Property Calculation Code

Section Property code called in Main Program provided in Appendix B.1

```
function [A, Ixx, Iyy] = SectionProperties(webydata, ydata, zleft, nleft,
zright, nright, XFit1, YFit1, XFit2, YFit2, XFit3, YFit3, XFit4, YFit4,
WebXFitLeft, WebYFitLeft, WebXFitRight, WebYFitRight)
%Calculate Rail Section properties:
syms x; syms y;
ymax = max(ydata);
ymin = min(webydata);
A = 0;
Ixx = 0;
Iyy = 0;
ElementSize = 0.1; %0.1inch by 0.1inch square elements
%% Try Four: Evaluate Rail Head Bounds (Intersect elements with line y = k)
syms z
for k = ymin:ElementSize:ymax
   % Left Web element
   Equationleft = WebYFitLeft - k;
   Solleft = solve(Equationleft, 'z');
   if (abs(eval(imag(Solleft(1,1)))) <=  0.00001) && (eval(Solleft(1,1)) >=
-1) && (eval(Solleft(1,1)) <= nleft)
       p = real(eval(Solleft(1,1)))
       Yintercept1 = k
       Xintercept1 = subs(WebXFitLeft, z, p)
   elseif (abs(eval(imag(Solleft(2,1)))) <= 0.00001) && (eval(Solleft(2,1))
>= -1) && (eval(Solleft(2,1)) <= nleft)
       p = real(eval(Solleft(2,1)))
       Yintercept1 = k
       Xintercept1 = subs(WebXFitLeft, z, p)
   elseif (abs(eval(imag(Solleft(3,1)))) <= 0.00001) && (eval(Solleft(3,1))
>= -1) && (eval(Solleft(3,1)) <= nleft)
       p = real(eval(Solleft(3,1)))
       Yintercept1 = k
       Xintercept1 = subs(WebXFitLeft, z, p)
   end
   % First Head Element
   Equation1 = YFit1 - k;
   Sol1 = solve(Equation1, 'z');
   if (abs(eval(imag(Sol1(1,1)))) <=  0.00001) && (eval(Sol1(1,1)) >= zleft)
&& (eval(Sol1(1,1)) <= 1)
       p = real(eval(Sol1(1,1)))
       Yintercept1 = k
       Xintercept1 = subs(XFit1, z, p)
   elseif (abs(eval(imag(Sol1(2,1)))) <= 0.00001) && (eval(Sol1(2,1)) >=
zleft) && (eval(Sol1(2,1)) <= 1)
       p = real(eval(Sol1(2,1)))
       Yintercept1 = k
       Xintercept1 = subs(XFit1, z, p)
   elseif (abs(eval(imag(Sol1(3,1)))) <= 0.00001) && (eval(Sol1(3,1)) >=
zleft) && (eval(Sol1(3,1)) <= 1)
       p = real(eval(Sol1(3,1)))
       Yintercept1 = k
       Xintercept1 = subs(XFit1, z, p)
   end
   % Second Element
```

```
    Equation2 = YFit2 - k;
    Sol2 = solve(Equation2, 'z');
    if (abs(eval(imag(Sol2(1,1)))) <= 0.00001) && (eval(Sol2(1,1)) >= -1) &&
(eval(Sol2(1,1)) <= 1)
        p = real(eval(Sol2(1,1)))
        Yintercept1 = k
        Xintercept1 = subs(XFit2, z, p)
    elseif (abs(eval(imag(Sol2(2,1)))) <= 0.0001) && (eval(Sol2(2,1)) >= -1)
&& (eval(Sol2(2,1)) <= 1)
        p = real(eval(Sol2(2,1)))
        Yintercept1 = k
        Xintercept1 = subs(XFit2, z, p)
    elseif (abs(eval(imag(Sol2(3,1)))) <= 0.00001) && (eval(Sol2(3,1)) >= -1)
&& (eval(Sol2(3,1)) <= 1)
        p = real(eval(Sol2(3,1)))
        Yintercept1 = k
        Xintercept1 = subs(XFit2, z, p)
    end
    %Second Element Continued (Two intersections within Element2)
    if (abs(eval(imag(Sol2(1,1)))) <= 0.00001) && (eval(Sol2(1,1)) >= -1) &&
(eval(Sol2(1,1)) <= 1) && (abs(eval(imag(Sol2(2,1)))) <= 0.0001) &&
(eval(Sol2(2,1)) >= -1) && (eval(Sol2(2,1)) <= 1)
        if eval(Sol2(2,1)) > eval(Sol2(1,1))
            p1 = real(eval(Sol2(1,1)))
            p2 = real(eval(Sol2(2,1)))
            Xintercept21 = subs(XFit2, z, p1)
            Xintercept22 = subs(XFit2, z, p2)
            Yintercept21 = k
        else
            p2 = real(eval(Sol2(1,1)))
            p1 = real(eval(Sol2(2,1)))
            Xintercept21 = subs(XFit2, z, p1)
            Xintercept22 = subs(XFit2, z, p2)
            Yintercept21 = k
        end
    elseif (abs(eval(imag(Sol2(1,1)))) <= 0.00001) && (eval(Sol2(1,1)) >= -1)
&& (eval(Sol2(1,1)) <= 1) && (abs(eval(imag(Sol2(3,1)))) <= 0.0001) &&
(eval(Sol2(3,1)) >= -1) && (eval(Sol2(3,1)) <= 1)
        if eval(Sol2(3,1)) > eval(Sol2(1,1))
            p1 = real(eval(Sol2(1,1)))
            p2 = real(eval(Sol2(3,1)))
            Xintercept21 = subs(XFit2, z, p1)
            Xintercept22 = subs(XFit2, z, p2)
            Yintercept21 = k
        else
            p2 = real(eval(Sol2(1,1)))
            p1 = real(eval(Sol2(3,1)))
            Xintercept21 = subs(XFit2, z, p1)
            Xintercept22 = subs(XFit2, z, p2)
            Yintercept21 = k
        end
    elseif (abs(eval(imag(Sol2(2,1)))) <= 0.00001) && (eval(Sol2(2,1)) >= -1)
&& (eval(Sol2(2,1)) <= 1) && (abs(eval(imag(Sol2(3,1)))) <= 0.0001) &&
(eval(Sol2(3,1)) >= -1) && (eval(Sol2(3,1)) <= 1)
        if eval(Sol2(3,1)) > eval(Sol2(2,1))
            p1 = real(eval(Sol2(2,1)))
            p2 = real(eval(Sol2(3,1)))
```

```
            Xintercept21 = subs(XFit2, z, p1)
            Xintercept22 = subs(XFit2, z, p2)
            Yintercept21 = k
        else
            p2 = real(eval(Sol2(2,1)))
            p1 = real(eval(Sol2(3,1)))
            Xintercept21 = subs(XFit2, z, p1)
            Xintercept22 = subs(XFit2, z, p2)
            Yintercept21 = k
        end
    end
    % Third Element
    Equation3 = YFit3 - k;
    Sol3 = solve(Equation3, 'z');
    if (abs(eval(imag(Sol3(1,1)))) <= 0.00001) && (eval(Sol3(1,1)) >= -1) &&
(eval(Sol3(1,1)) <= 1)
        p = real(eval(Sol3(1,1)))
        Yintercept2 = k
        Xintercept2 = subs(XFit3, z, p)
    elseif (abs(eval(imag(Sol3(2,1)))) <= 0.00001) && (eval(Sol3(2,1)) >= -1)
&& (eval(Sol3(2,1)) <= 1)
        p = real(eval(Sol3(2,1)))
        Yintercept2 = k
        Xintercept2 = subs(XFit3, z, p)
    elseif (abs(eval(imag(Sol3(3,1)))) <= 0.00001) && (eval(Sol3(3,1)) >= -1)
&& (eval(Sol3(3,1)) <= 1)
        p = real(eval(Sol3(3,1)))
        Yintercept2 = k
        Xintercept2 = subs(XFit3, z, p)
    end
    % Third Element Continued (Two intersections within Element3)
    if (abs(eval(imag(Sol3(1,1)))) <= 0.00001) && (eval(Sol3(1,1)) >= -1) &&
(eval(Sol3(1,1)) <= 1) && (abs(eval(imag(Sol3(2,1)))) <= 0.0001) &&
(eval(Sol3(2,1)) >= -1) && (eval(Sol3(2,1)) <= 1)
        if eval(Sol3(2,1)) > eval(Sol3(1,1))
            p1 = real(eval(Sol3(1,1)))
            p2 = real(eval(Sol3(2,1)))
            Xintercept31 = subs(XFit3, z, p1)
            Xintercept32 = subs(XFit3, z, p2)
            Yintercept31 = k
        else
            p2 = real(eval(Sol3(1,1)))
            p1 = real(eval(Sol3(2,1)))
            Xintercept31 = subs(XFit3, z, p1)
            Xintercept32 = subs(XFit3, z, p2)
            Yintercept31 = k
        end
    elseif (abs(eval(imag(Sol3(1,1)))) <= 0.00001) && (eval(Sol3(1,1)) >= -1)
&& (eval(Sol3(1,1)) <= 1) && (abs(eval(imag(Sol3(3,1)))) <= 0.0001) &&
(eval(Sol3(3,1)) >= -1) && (eval(Sol3(3,1)) <= 1)
        if eval(Sol3(3,1)) > eval(Sol3(1,1))
            p1 = real(eval(Sol3(1,1)))
            p2 = real(eval(Sol3(3,1)))
            Xintercept31 = subs(XFit3, z, p1)
            Xintercept32 = subs(XFit3, z, p2)
            Yintercept31 = k
        else
```

```
            p2 = real(eval(Sol3(1,1)))
            p1 = real(eval(Sol3(3,1)))
            Xintercept31 = subs(XFit3, z, p1)
            Xintercept32 = subs(XFit3, z, p2)
            Yintercept31 = k
        end
    elseif (abs(eval(imag(Sol3(2,1)))) <= 0.00001) && (eval(Sol3(2,1)) >= -1)
&& (eval(Sol3(2,1)) <= 1) && (abs(eval(imag(Sol3(3,1)))) <= 0.0001) &&
(eval(Sol3(3,1)) >= -1) && (eval(Sol3(3,1)) <= 1)
        if eval(Sol3(3,1)) > eval(Sol3(2,1))
            p1 = real(eval(Sol3(2,1)))
            p2 = real(eval(Sol3(3,1)))
            Xintercept31 = subs(XFit3, z, p1)
            Xintercept32 = subs(XFit3, z, p2)
            Yintercept31 = k
        else
            p2 = real(eval(Sol3(2,1)))
            p1 = real(eval(Sol3(3,1)))
            Xintercept31 = subs(XFit3, z, p1)
            Xintercept32 = subs(XFit3, z, p2)
            Yintercept31 = k
        end
    end
    % Fourth Element
    Equation4 = YFit4 - k;
    Sol4 = solve(Equation4, 'z');
    if (abs(eval(imag(Sol4(1,1)))) <= 0.00001) && (eval(Sol4(1,1)) >= -1) &&
(eval(Sol4(1,1)) <= zright)
        p = real(eval(Sol4(1,1)))
        Yintercept2 = k
        Xintercept2 = subs(XFit4, z, p)
    elseif (abs(eval(imag(Sol4(2,1)))) <= 0.00001) && (eval(Sol4(2,1)) >= -1)
&& (eval(Sol4(2,1)) <= zright)
        p = real(eval(Sol4(2,1)))
        Yintercept2 = k
        Xintercept2 = subs(XFit4, z, p)
    elseif (abs(eval(imag(Sol4(3,1)))) <= 0.00001) && (eval(Sol4(3,1)) >= -1)
&& (eval(Sol4(3,1)) <= zright)
        p = real(eval(Sol4(3,1)))
        Yintercept2 = k
        Xintercept2 = subs(XFit4, z, p)
    end
    % Right Web element
    Equationright = WebYFitRight - k;
    Solright = solve(Equationright, 'z');
    if (abs(eval(imag(Solright(1,1)))) <=  0.00001) && (eval(Solright(1,1))
>= nright) && (eval(Solright(1,1)) <= 1)
        p = real(eval(Solright(1,1)))
        Yintercept2 = k
        Xintercept2 = subs(WebXFitRight, z, p)
    elseif (abs(eval(imag(Solright(2,1)))) <= 0.00001) &&
(eval(Solright(2,1)) >= nright) && (eval(Solright(2,1)) <= 1)
        p = real(eval(Solright(2,1)))
        Yintercept2 = k
        Xintercept2 = subs(WebXFitRight, z, p)
    elseif (abs(eval(imag(Solright(3,1)))) <= 0.00001) &&
(eval(Solright(3,1)) >= nright) && (eval(Solright(3,1)) <= 1)
```

```matlab
        p = real(eval(Solright(3,1)))
        Yintercept2 = k
        Xintercept2 = subs(WebXFitRight, z, p)
    end
    %% Calculate Section Properties
    if (exist('Xintercept1') == 1) && (exist('Xintercept2') == 1)
        % Calculate Head Area
        Ak = (Xintercept2 - Xintercept1)*ElementSize;
        A = A + Ak;
        % Calculate Moment of inertia about x and y axis (x=0, y=0: origin is
the intersection of the lines tangent to the slope of each rail foot)
        yk = Yintercept1 - 0;
        Ixx = Ixx + (yk^2 * Ak);
        for l = Xintercept1:ElementSize:Xintercept2
            elementxCG = l;
            xkl = elementxCG - 0;
            Akl = ElementSize^2;
            Iyy = Iyy + (xkl^2 * Akl);
        end
    end
    % For two intersections within element2
    if (exist('Xintercept21') == 1) && (exist('Xintercept22') == 1)
        % Calculate Head Area
        Ak = (Xintercept22 - Xintercept21)*ElementSize;
        A = A + Ak;
        % Calculate Moment of inertia about x and y axis (x=0, y=0)
        yk = Yintercept21 - 0;
        Ixx = Ixx + (yk^2 * Ak);
        for l = Xintercept21:ElementSize:Xintercept22
            elementxCG = l;
            xkl = elementxCG - 0;
            Akl = ElementSize^2;
            Iyy = Iyy + (xkl^2 * Akl);
        end
    end
    % For two intersections within element3
    if (exist('Xintercept31') == 1) && (exist('Xintercept32') == 1)
        % Calculate Head Area
        Ak = (Xintercept32 - Xintercept31)*ElementSize;
        A = A + Ak;
        % Calculate Moment of inertia about x and y axis (x=0, y=0)
        yk = Yintercept31 - 0;
        Ixx = Ixx + (yk^2 * Ak);
        for l = Xintercept31:ElementSize:Xintercept32
            elementxCG = l;
            xkl = elementxCG - 0;
            Akl = ElementSize^2;
            Iyy = Iyy + (xkl^2 * Akl);
        end
    end
end
```

Appendix C. MATLAB Distribution Fitting Code

```
clear all
clc
%% Import Data and Plot Historgrams
PeakLeftLoad = (xlsread('FLATROCK_VERTICAL_FORCE_DATA_30_DAYS.xlsx', 1,
'F2:F348996'));
PeakRightLoad = (xlsread('FLATROCK_VERTICAL_FORCE_DATA_30_DAYS.xlsx', 1,
'I2:I348996'));
l = 0:1:max(PeakLeftLoad);
L = hist(PeakLeftLoad, l);
hist(PeakLeftLoad, l);
title('Left Track Peak Load')
xlabel('Load (kips)')
ylabel('Frequency')
grid on
figure
r = 0:1:max(PeakRightLoad);
R = hist(PeakRightLoad, r);
hist(PeakRightLoad, r);
title('Right Track Peak Load')
xlabel('Load (kips)')
ylabel('Frequency')
grid on
%% Generate Mode 2 for Left Track by itself:
j = 0;
median = 38.75;
for i = 1:length(PeakLeftLoad)
    if PeakLeftLoad(i) >= median
        j = j + 1;
        Mode2PeakLeftLoadRightSide(j) = PeakLeftLoad(i);
    end
end
Mode2PeakLeftLoadRightSideTF = Mode2PeakLeftLoadRightSide - median;
Mode2PeakLeftLoadLeftSideTF = - Mode2PeakLeftLoadRightSideTF;
Mode2PeakLeftLoadTF = [Mode2PeakLeftLoadLeftSideTF
Mode2PeakLeftLoadRightSideTF];
Mode2PeakLeftLoad = Mode2PeakLeftLoadTF + median;
figure
hist(Mode2PeakLeftLoad, r)
title('Mode 2 Peak Load for Left Track')
xlabel('Load (kips)')
ylabel('Frequency')
grid on
%% Calculate 2 sigma Load for Left Track
%From dfittool: mu = 38, sigma = 6.273, nu = 2.33568
leftmu = 38.75;
leftsigma = 6.273;
leftnu = 2.33568;
LeftMean = leftmu;
LeftStdDev = sqrt(1/leftsigma*(leftnu/(leftnu-2)));
MaxLeftLoad = LeftMean + 3*LeftStdDev;
%% Generate Mode 2 for Right Track by itself:
j = 0;
median = 38.75;
for i = 1:length(PeakRightLoad)
    if PeakRightLoad(i) >= median
```

```
        j = j + 1;
        Mode2PeakRightLoadRightSide(j) = PeakRightLoad(i);
    end
end
Mode2PeakRightLoadRightSideTF = Mode2PeakRightLoadRightSide - median;
Mode2PeakRightLoadLeftSideTF = - Mode2PeakRightLoadRightSideTF;
Mode2PeakRightLoadTF = [Mode2PeakRightLoadLeftSideTF
Mode2PeakRightLoadRightSideTF];
Mode2PeakRightLoad = Mode2PeakRightLoadTF + median;
figure
hist(Mode2PeakRightLoad, r)
title('Mode 2 Peak Load for Right Track')
xlabel('Load (kips)')
ylabel('Frequency')
grid on
%% Calculate 2 sigma Load for Right Track
%From dfittool: mu = 38, sigma = 6.273, nu = 2.33568
rightmu = 38;
rightsigma = 6.273;
rightnu = 2.33568;
RightMean = rightmu;
RightStdDev = sqrt(1/rightsigma*(rightnu/(rightnu-2)));
MaxRightLoad = RightMean + 3*RightStdDev;
```

Appendix D.1 Reverse Bending Location
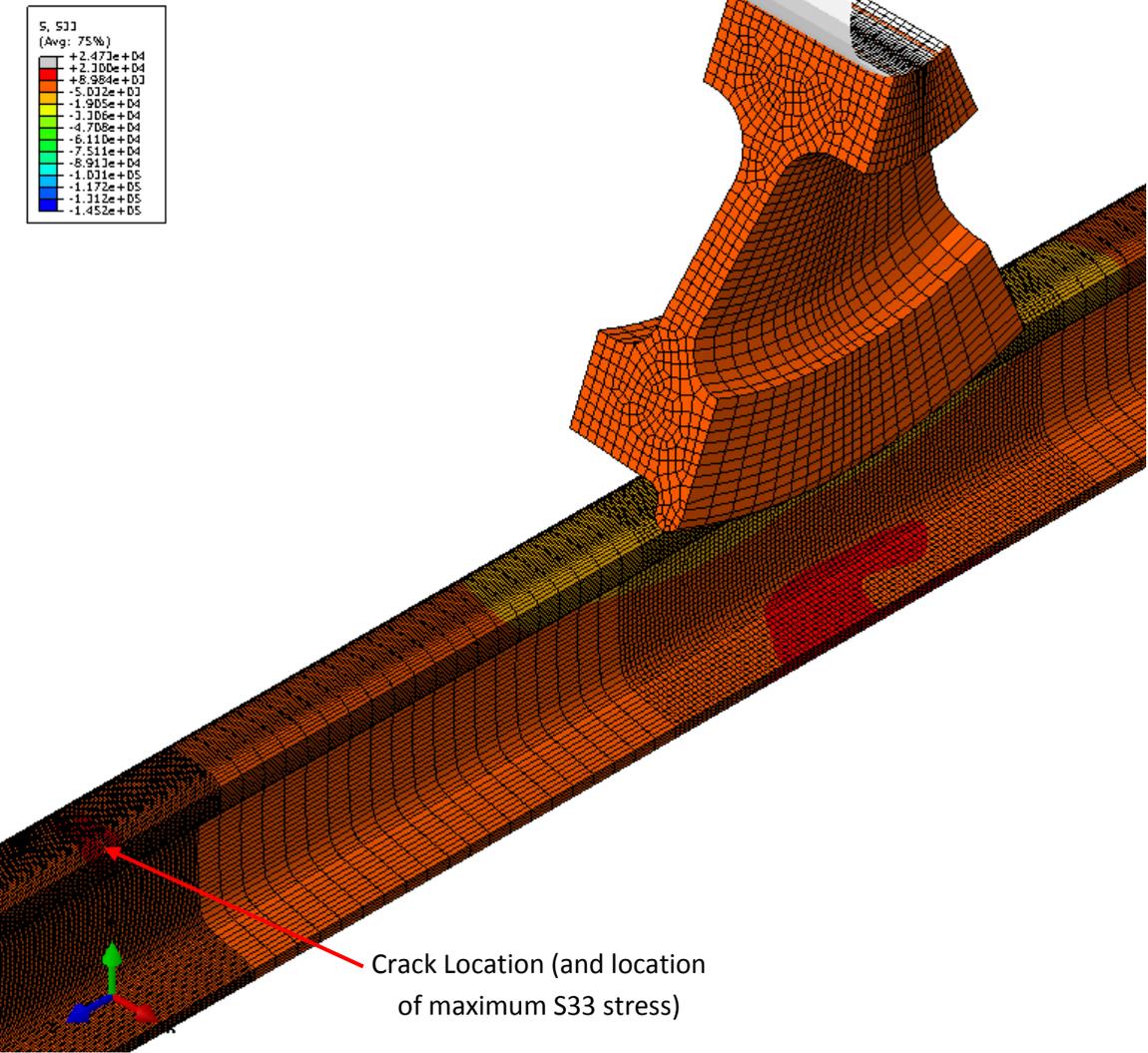
Figure D.1 Verification of reverse bending wheel location.

Appendix D.2 Convergence Study Additional Tables and Plots

Table D.1 Convergence table for 8/16^{ths} side-wear profile with 5% HA flaw.

Mesh #	# Elements	Stress @ 1 (psi)	% Diff	Stress @ 2 (psi)	% Diff	Stress @ 3 (psi)	% Diff	Stress @ 4 (psi)	% Diff
1	125910	19469	N/A	16168.7	N/A	32721.3	N/A	31546.1	N/A
2	264516	20523	5.41373	18487	14.33820	32280.3	-1.34775	32027.1	1.52475
3	413675	19764.7	-3.69488	17924.6	-3.04214	32453.6	0.53686	32136.8	0.34252
4	554983	19719.7	-0.22768	17781.5	-0.79834	33020.3	1.74618532	32240.5	0.32268

Mesh #	# Elements	Stress @ 1 (MPa)	% Diff	Stress @ 2 (MPa)	% Diff	Stress @ 3 (MPa)	% Diff	Stress @ 4 (MPa)	% Diff
1	125910	134.2340297	N/A	111.4792622	N/A	225.6054217	N/A	217.5027029	N/A
2	264516	141.5011039	5.41373	127.463378	14.33820	222.5648337	-1.34775	220.8190812	1.52475
3	413675	136.2728094	-3.69488	123.5857665	-3.04214	223.7596952	0.53686	221.5754361	0.34252
4	554983	135.9625453	-0.22768	122.5991268	-0.79834	227.6669541	1.74619	222.2904224	0.32268
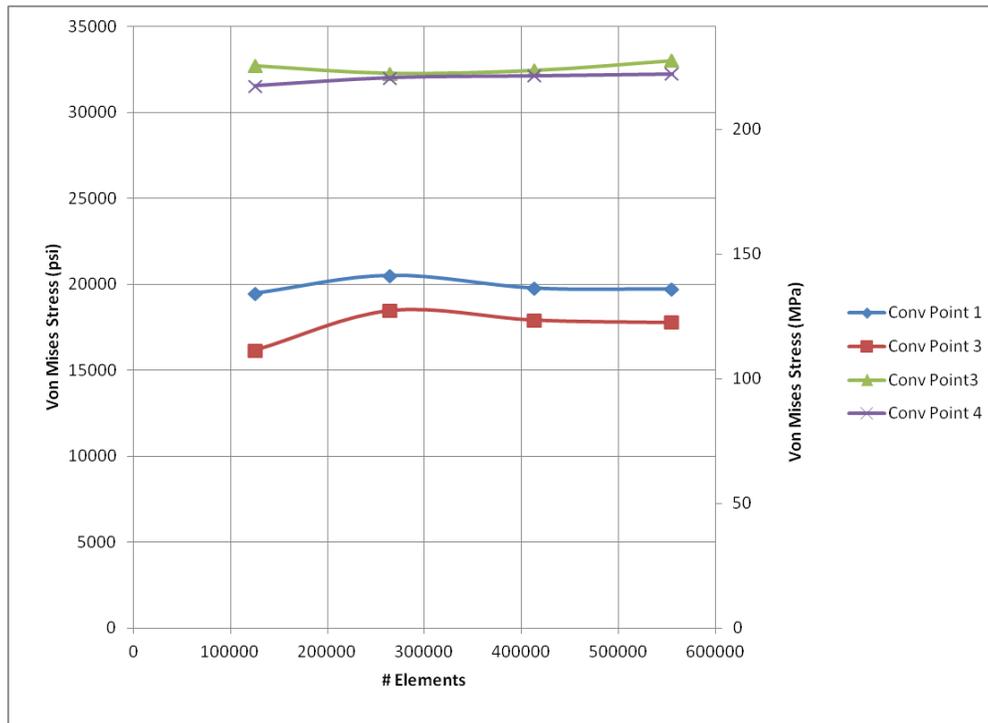
Figure D.1 Convergence plot for 8/16^{ths} side-wear profile with 5% HA flaw.

Table D.2 Convergence table for 8/16^{ths} side-wear profile with 20% HA flaw.

Mesh #	# Elements	Stress @ 1 (psi)	% Diff	Stress @ 2 (psi)	% Diff	Stress @ 3 (psi)	% Diff	Stress @ 4 (psi)	% Diff
1	122314	32277.3	N/A	5522.1	N/A	23017.4	N/A	25326.3	N/A
2	237721	33298.5	3.16383	4515.8	-18.22314	24332.1	5.71177	25111.2	-0.84931
3	366017	32725.4	-1.72110	6248.59	38.37172	24956.7	2.56698	25873	3.03371
4	533900	32522.8	-0.61909	6379.22	2.09055	25100.5	0.57620	25711.5	-0.62420

Mesh #	# Elements	Stress @ 1 (MPa)	% Diff	Stress @ 2 (MPa)	% Diff	Stress @ 3 (MPa)	% Diff	Stress @ 4 (MPa)	% Diff
1	122314	222.5441495	N/A	38.07353923	N/A	158.6993864	N/A	174.6186916	N/A
2	237721	229.5850756	3.16383	31.13534497	-18.22314	167.7639239	5.71177	173.1356293	-0.84931
3	366017	225.6336902	-1.72110	43.08251145	38.37172	172.0703893	2.56698	178.3880554	3.03371
4	533900	224.2368124	-0.61909	43.9831736	2.09055	173.0618554	0.57620	177.2745521	-0.62420
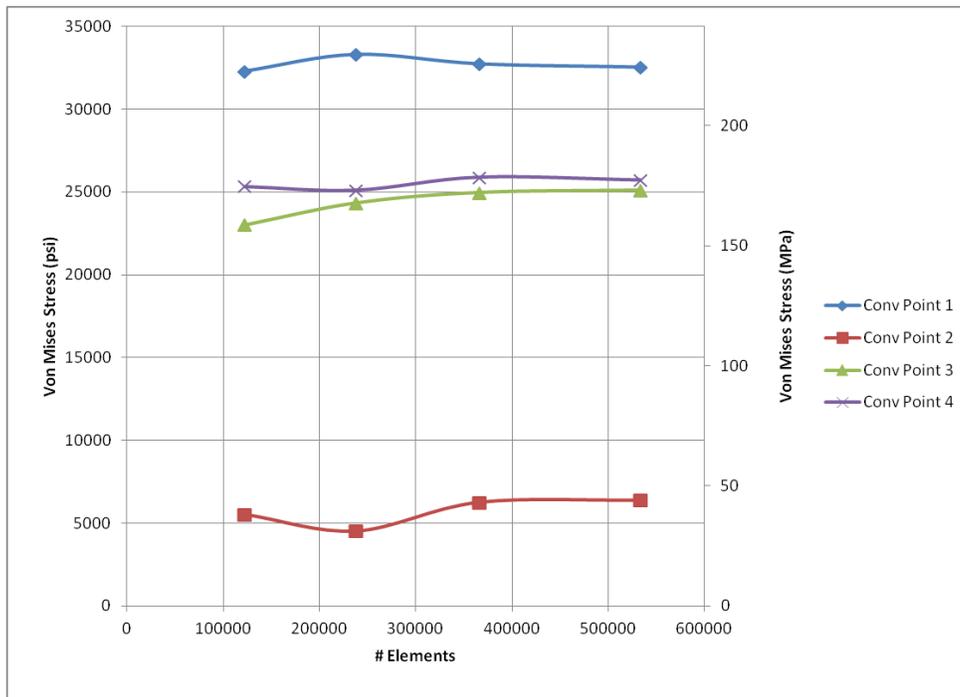
Figure D.2 Convergence plot for 8/16^{ths} side-wear profile with 20% HA flaw.

Appendix E.1 Fitted Rail Profiles
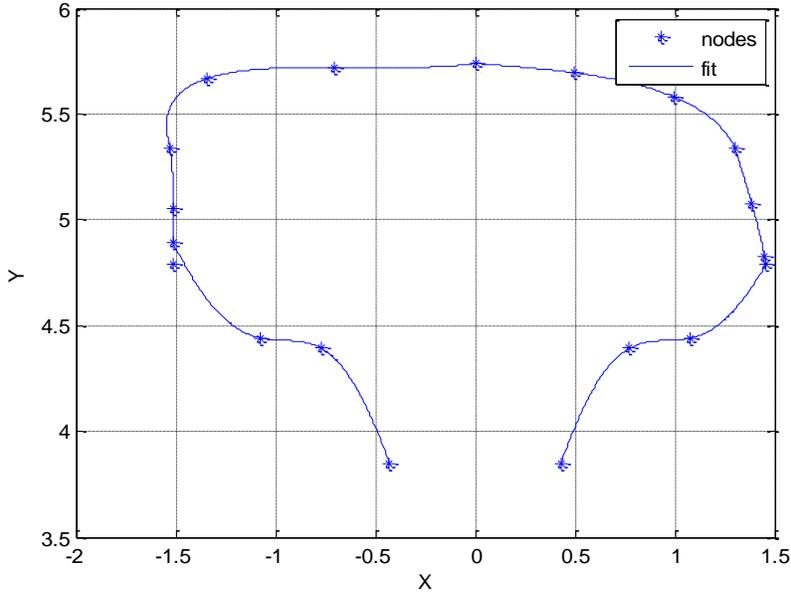
Figure E.1 0/16^{ths} side-wear fitted profile.
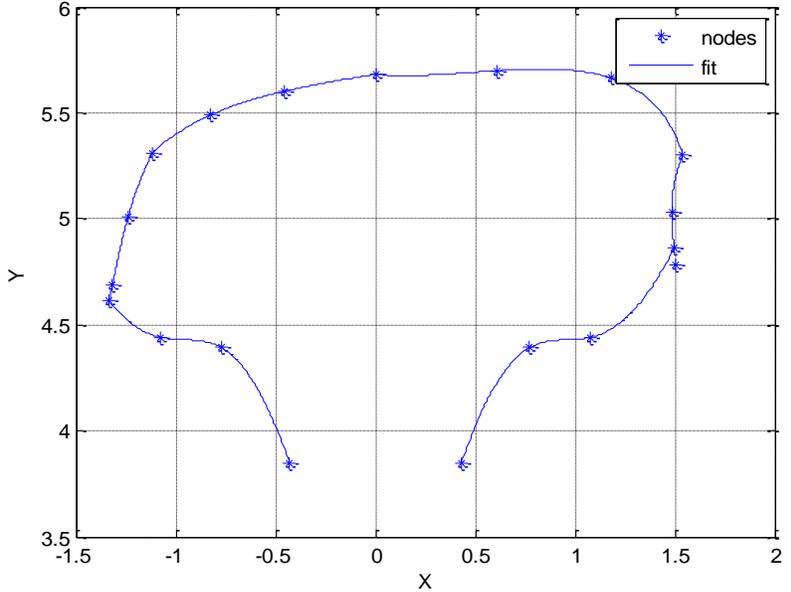
Figure E.2 3/16^{ths} side-wear fitted profile.
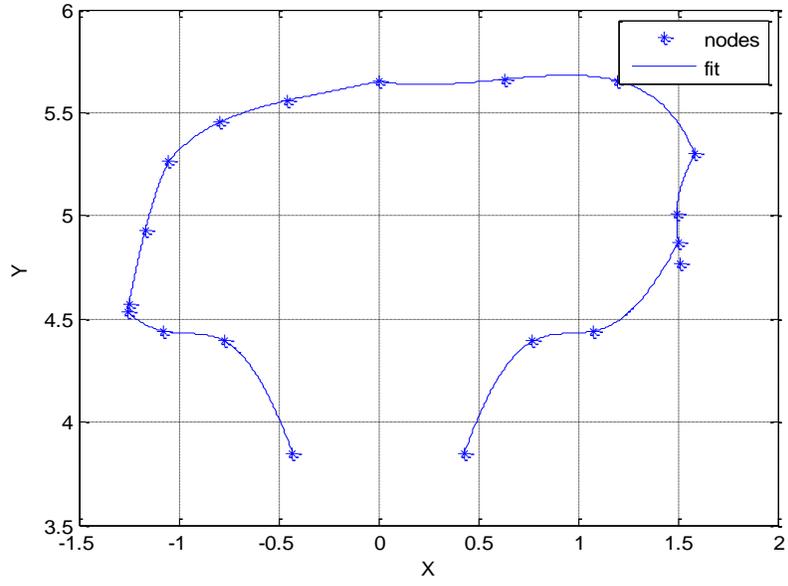
Figure E.3 4/16^{ths} side-wear fitted profile.
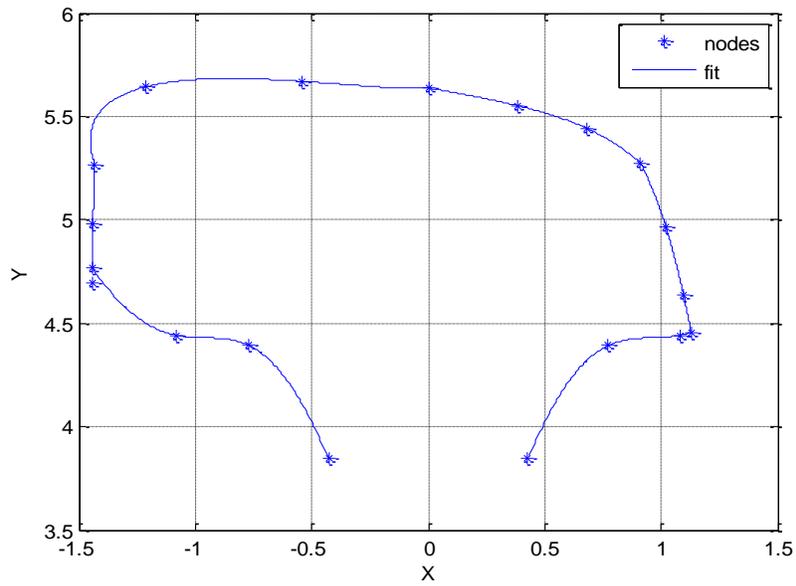
Figure E.4 6/16^{ths} side-wear fitted profile.
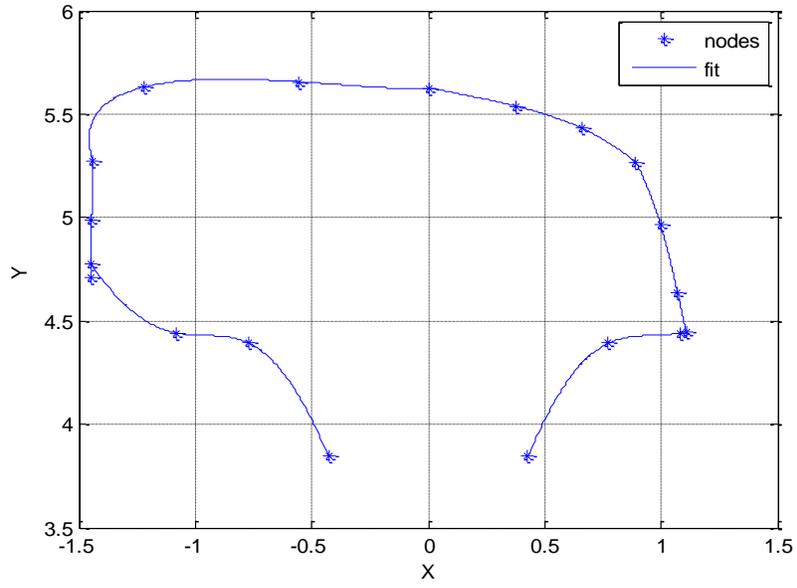
Figure E.5 7/16^{ths} side-wear fitted profile.

Figure E.6 8/16^{ths} side-wear fitted profile.

151
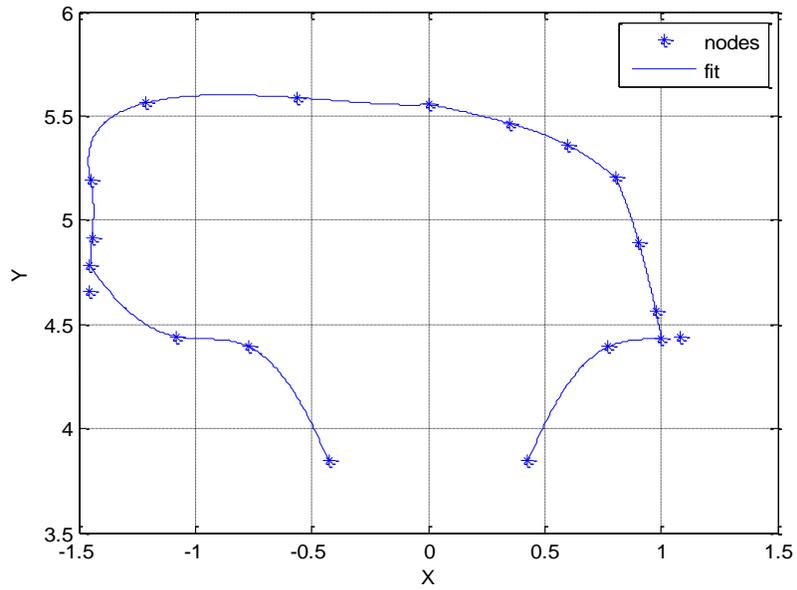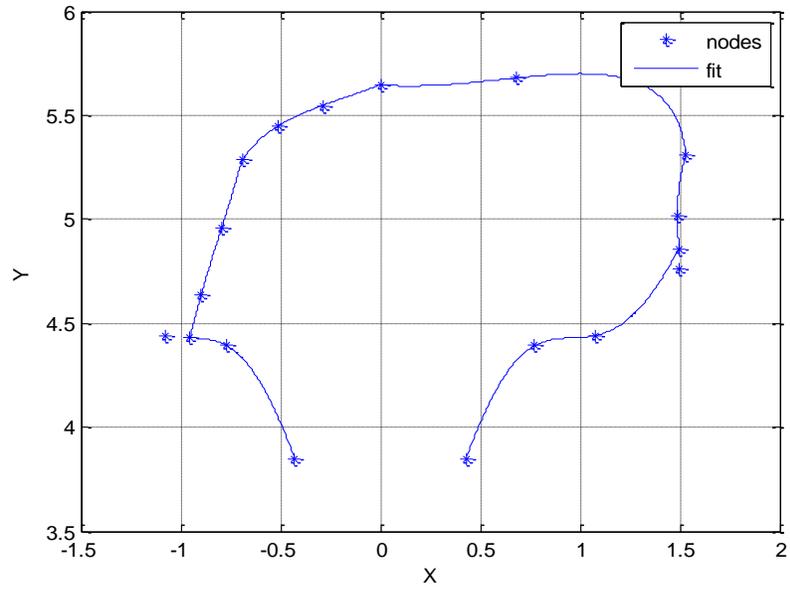
Figure E.7 9/16^{ths} side-wear fitted profile.
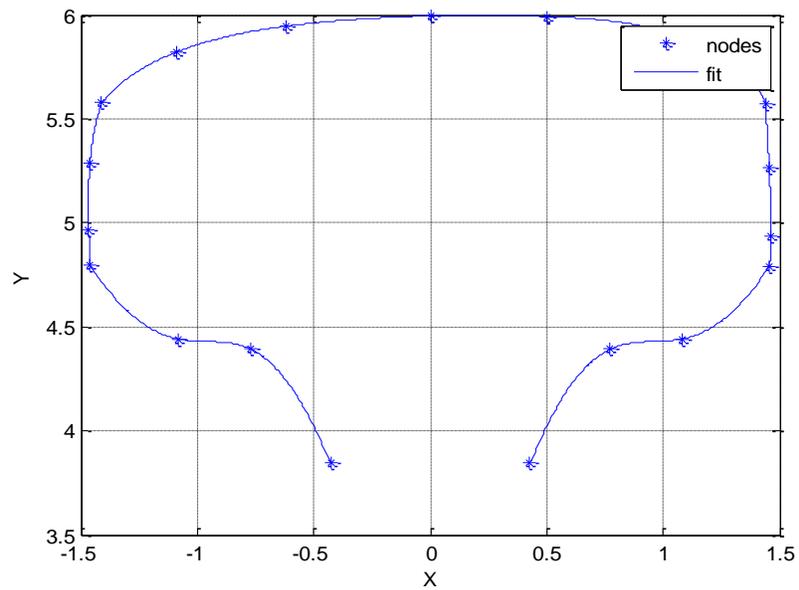
Figure E.8 2/16^{ths} top-wear fitted profile.
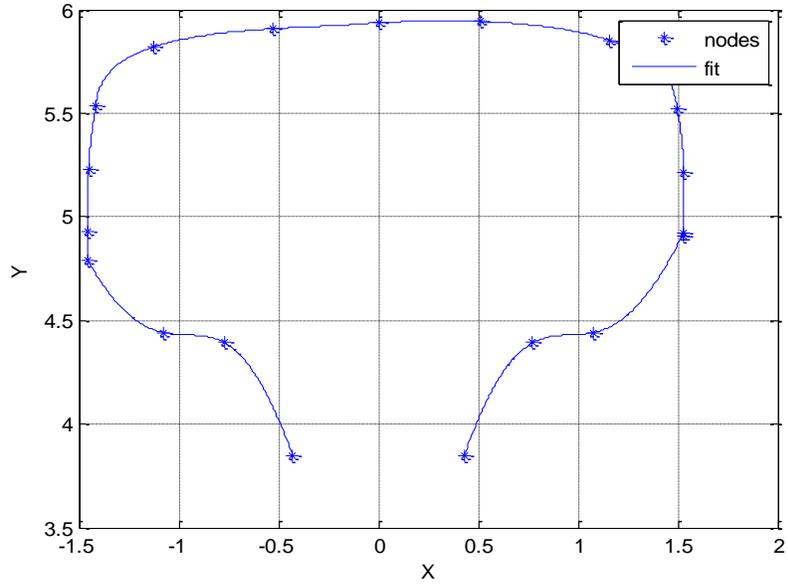
Figure E.9 3/16^{ths} top-wear fitted profile.
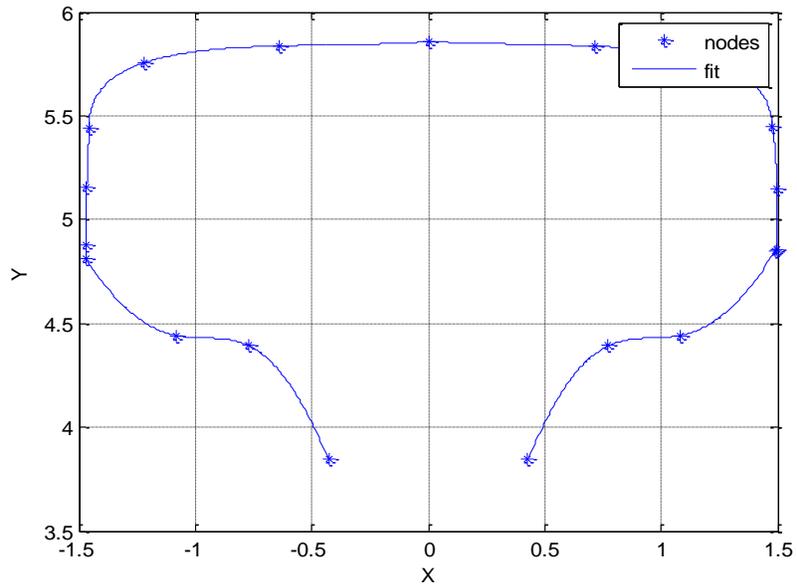
Figure E.10 4/16^{ths} top-wear fitted profile.
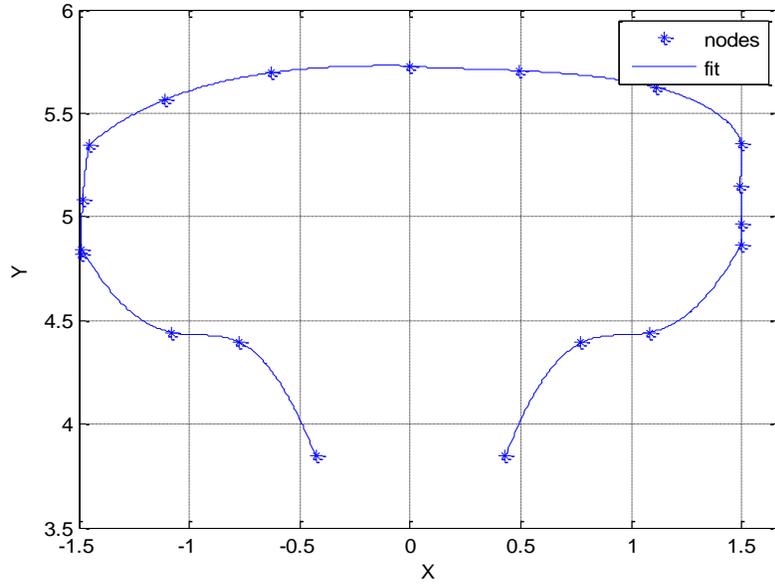
Figure E.11 6/16^{ths} top-wear fitted profile.
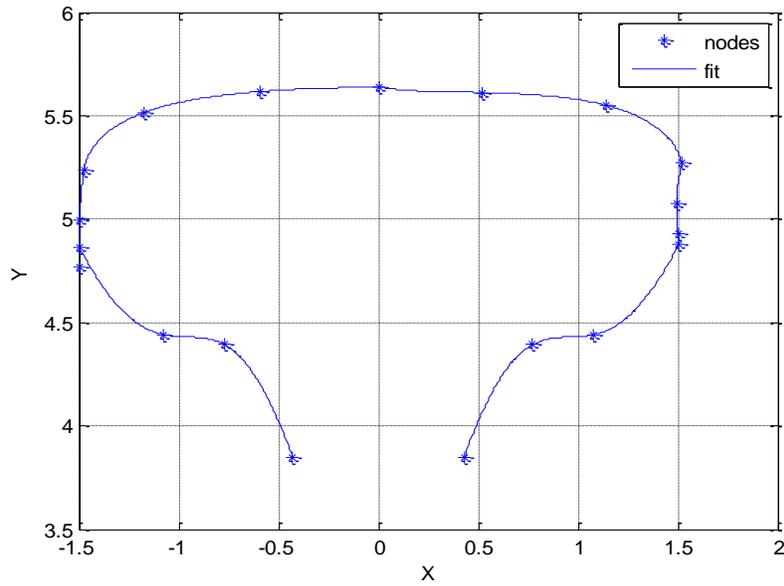
Figure E.12 8/16^{ths} top-wear fitted profile.
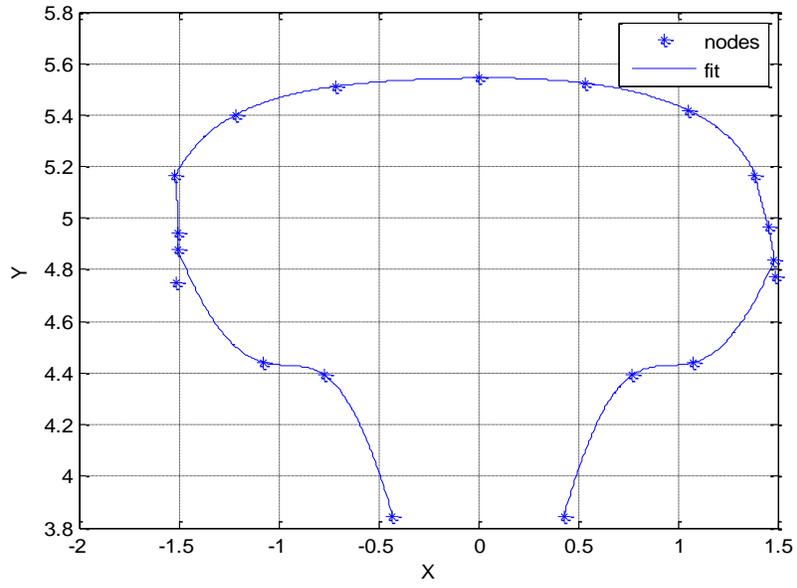
Figure E.13 9/16^{ths} top-wear fitted profile.
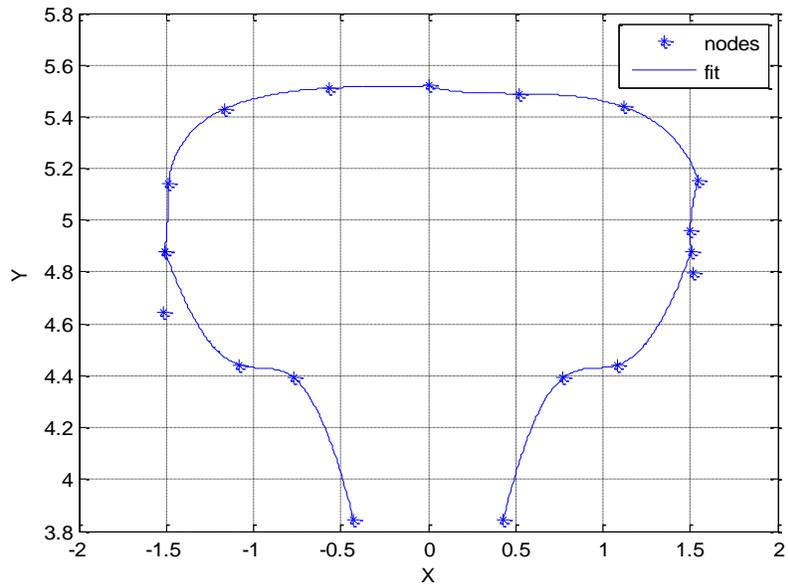
Figure E.14 10/16^{ths} top-wear fitted profile.
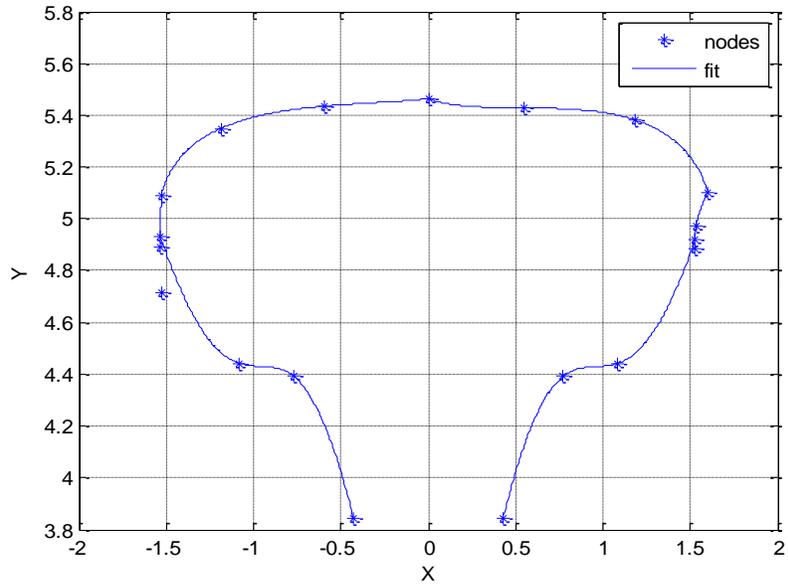
Figure E.15 11/16^{ths} top-wear fitted profile.
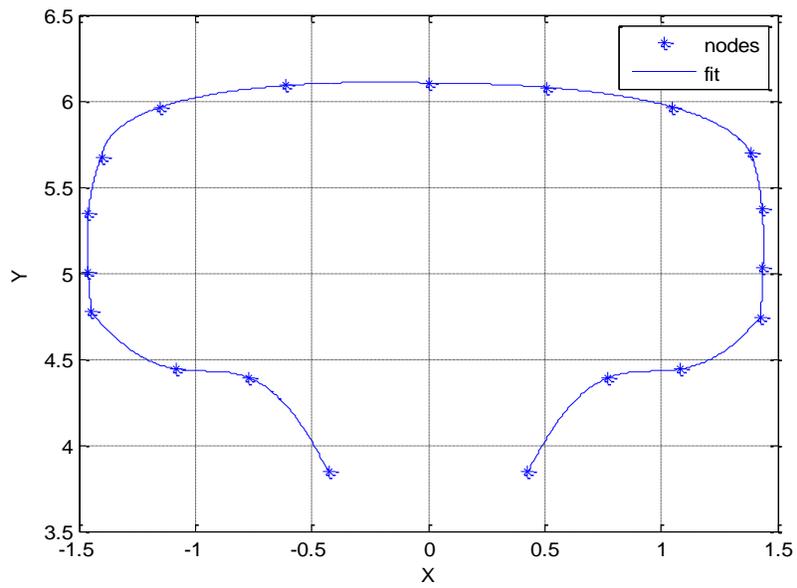
Figure E.16 Unworn fitted profile.

Appendix E.2 Python Queuing Script for Abaqus Runs
*Written by Brice Collamer

```python
import os
import sys
import time

fileList = []
directory = os.getcwd()
for filename in os.listdir(directory):
        basename, extension = filename.split('.')
        if extension == 'inp':
          fileList.append(basename)
print '\n'.join(fileList)
f = open('JobList.txt','w')
f.write('\n'.join(fileList))
f.close()

for i,j in enumerate(fileList):
        changeLine = '/apps/packages/physics/abaqus/Commands/abq6102 job=%s cpus=8 interactive'
%(fileList[i])
        a=[]
        g = open('qsub_abaqus.sh','r')
        for line in g.read().split('\n'):
                a.append(line)
    outputFile = '%s.sh' %(fileList[i])
        a[18] = changeLine
        h = open(outputFile,'w')
        h.write('\n'.join(a))
        h.close()

        cmd = 'qsub ./%s' %(outputFile)
        os.system(cmd)
        time.sleep(5*60)
    fname = directory+'/'+fileList[i]+'.lck'
    while os.path.exists(fname):
         time.sleep(500)
```

Appendix E.3 Python Batch Generation Script for FE-Safe Analyses
*Written by Brice Collamer

```python
import os
import sys

# Store all .odb file names to a list
fileList = []
directory = os.getcwd()
for filename in os.listdir(directory):
    basename, extension = filename.split('.')
    if extension == 'odb':
        fileList.append(basename)


print '\n'.join(fileList)


nJobs= len(fileList)
CommandList = []



for i in range(0, nJobs):
    odb_name= fileList[i]
    command1 = '"C:/Program Files (x86)/fesafe/version.5.4/exe/fe-safe_cl.exe" j=%s.odb, b=One.kwd, o=%s1KWD.odb'  %(odb_name, odb_name)
    CommandList.append(command1)
    command2 = '"C:/Program Files (x86)/fesafe/version.5.4/exe/fe-safe_cl.exe" j=%s.odb, b=Two.kwd, o=%s2KWD.odb' %(odb_name, odb_name)
    CommandList.append(command2)
    command3 = '"C:/Program Files (x86)/fesafe/version.5.4/exe/fe-safe_cl.exe" j=%s.odb, b=Three.kwd, o=%s3KWD.odb' %(odb_name, odb_name)
    CommandList.append(command3)
    command4 = '"C:/Program Files (x86)/fesafe/version.5.4/exe/fe-safe_cl.exe" j=%s.odb, b=Four.kwd, o=%s4KWD.odb' %(odb_name, odb_name)
    CommandList.append(command4)
    command5 = '"C:/Program Files (x86)/fesafe/version.5.4/exe/fe-safe_cl.exe" j=%s.odb, b=Five.kwd, o=%s5KWD.odb' %(odb_name, odb_name)
    CommandList.append(command5)
    command6 = '"C:/Program Files (x86)/fesafe/version.5.4/exe/fe-safe_cl.exe" j=%s.odb, b=Six.kwd, o=%s6KWD.odb'  %(odb_name, odb_name)
    CommandList.append(command6)
    command7 = '"C:/Program Files (x86)/fesafe/version.5.4/exe/fe-safe_cl.exe" j=%s.odb, b=Seven.kwd, o=%s7KWD.odb' %(odb_name, odb_name)
    CommandList.append(command7)
    command8 = '"C:/Program Files (x86)/fesafe/version.5.4/exe/fe-safe_cl.exe" j=%s.odb, b=Eight.kwd, o=%s8KWD.odb' %(odb_name, odb_name)
    CommandList.append(command8)

g = open('BatchJobs.txt', 'w')
g.write('\n'.join(CommandList))
g.close()  jdshfjkhdsf
```

Appendix E.4 Python Results Extraction Post-Processing Script
*Written by Brice Collamer

This example script is only for top-wear profile .odb files. Similar scripts were written for side-wear profile .odb files and unworn rail .odb files

```
import os
import sys
import re
from odbAccess import *
from abaqusConstants import *
from odbAccess import *
from tempfile import TemporaryFile
from xlwt import Workbook

fileList = []
directory = os.getcwd()
for filename in os.listdir(directory):
    basename, extension = filename.split('.')
    if extension == 'odb':
        fileList.append(basename)

convert = lambda text: int(text) if text.isdigit() else text
alphanum_key = lambda key: [convert(c) for c in re.split('([0-9]+)', key)]
fileList = sorted(fileList, key = alphanum_key)

# Input the appropriate profiles, wear type, and crack size
libProfile = ('Profile2', 'Profile3', 'Profile4', 'Profile6', 'Profile8', 'Profile9', 'Profile10', 'Profile11')
libWear = ('TopWear')
libCrack= ('5PercentCrack', '7_5PercentCrack', '10PercentCrack', '12_5PercentCrack',
'15PercentCrack')
libCurvature= ('HighC', 'LowC')
libKWD = ('1KWD', '2KWD', '3KWD', '4KWD', '5KWD', '6KWD', '7KWD', '8KWD')

L = []
for a in libProfile:
    for c in libCrack:
        for d in libCurvature:
            for e in libKWD:
                w = a,libWear,c,d,e
                results= ''.join(w)
                L.append(results)

# Display order of execution to the user
for file in fileList:
    print file

# Extract life by averaging nodal values
nJobs = len(L)
lifeList = []
```

```
j=0
for i in range(0, nJobs):
      if j >= len(fileList):
            lifeList.append('Empty')
            j = len(fileList)-1
      if fileList[j] == L[i]:
         jobName = fileList[j]

         # Job information
         odb = openOdb(path='%s.odb' % (jobName))

         # Part information
         partName='PART-1-1'
         steps = odb.steps.keys()
         if steps[3] == 'fe-safe_01':
            stepName = 'fe-safe_01'
         else:
            stepName = '01'
         setName='CRACKRAILRIGHT_OUTPUTCRACKEDGE'

         step = odb.steps[stepName]
         lastFrame = step.frames[-1]


         # Specify the crack location
         regionSet = odb.rootAssembly.instances[partName].nodeSets[setName]

         elementLife = lastFrame.fieldOutputs['LOGLife-Repeats']
         nodeLife = elementLife.getSubset(region=regionSet).values

         # Sort life values based on the node label
         sortLife= sorted(nodeLife, key=lambda nodeLife: nodeLife.nodeLabel)

         # Initialize variables
         nodeLabels = []
         lifeResults = []

         for i in sortLife:
            nodeLabels.append(i.nodeLabel)
            lifeResults.append(i.data)

         # Average the nodal values
         end = len(lifeResults)
         averageLife=[]
         total = []
         for i in range(0,end,4):
            total= lifeResults[i]+lifeResults[i+1]+lifeResults[i+2]+lifeResults[i+3]
            average= total/4
            averageLife.append(average)

         x = min(averageLife)
```

```python
            Life= 10**x
            lifeList.append(Life)
            print '\nLife  is  ' '%0.2f' ' days for %s ' %(Life, jobName)

            j = j+1

        else:
            lifeList.append('Empty')


# Write the life values to an excel file

book = Workbook()
sheet1 = book.add_sheet('Sheet 1')

r = 1
r0=r
number_in_column = 16
rMax = r + number_in_column
columnPattern = [10, 11, 12, 13, 14] # enter the columns to be written
j = 0
c = columnPattern[j]

for i in range(len(lifeList)):
    sheet1.write(r,c, lifeList[i])
    r = r+1

    if r == rMax:
        j = j + 1
        if j == 5:
            c = columnPattern[0]
            r = rMax
            r0 = r
            rMax = r + number_in_column
            j = 0
        c = columnPattern[j]
        r = r0


book.save('LifeResults.xls')
book.save(TemporaryFile())
```

Appendix E.5 Excel Results Spreadsheet

Table E.1 Independent and output variable samples for the generation of the response surfaces.

Wear	h (in)	A (in²)	Ixx (in⁴)	Iyy (in⁴)	C(deg)	P (kipps)	deltaT (deg F)	SUT (ksi)	5% Life	7.5% Life	10% Life	12.5% Life	15% Life	20% Life	35% Life	50% Life	da/dN Const 1	da/dN Const 2
Profile 0	6.919	4.1758	101.1898	2.4923	10.65	44	42	141	0.641394	1.514346	1.460411		0.891405	1.413839			0.4366	0.2085
Profile 0	6.919	4.1758	101.1898	2.4923	10.65	56	42	141	0.189627	0.448823	0.3865		0.226334	0.388329			0.4597	0.456
Profile 0	6.919	4.1758	101.1898	2.4923	10.65	44	88	141	0.606178	1.197223	0.954443		0.65906	1.027839			0.4675	0.1642
Profile 0	6.919	4.1758	101.1898	2.4923	10.65	44	42	163	0.490371	1.215136	1.169297		0.693426	1.130772			0.4373	0.248
Profile 0	6.919	4.1758	101.1898	2.4923	10.65	44	88	163	0.463367	0.948309	0.747739		0.505446	0.805378			0.4686	0.1919
Profile 0	6.919	4.1758	101.1898	2.4923	10.65	56	88	141	0.19522	0.334503	0.262437		0.208521	0.306214			0.4583	0.9932
Profile 0	6.919	4.1758	101.1898	2.4923	10.65	56	42	163	0.138182	0.336977	0.288187		0.165187	0.290085			0.4603	0.5836
Profile 0	6.919	4.1758	101.1898	2.4923	10.65	56	88	163	0.142225	0.247956	0.19242		0.151705	0.226191			0.459	1.2847
Profile 0	6.919	4.1758	101.1898	2.4923	2.85	44	42	141	332.4489	29.21965	13.48963		1426.839	3762.487			0.4139	0.0001
Profile 0	6.919	4.1758	101.1898	2.4923	2.85	56	42	141	24.11015	4.347604	2.316861		58.30422	95.31802			0.3725	0.0086
Profile 0	6.919	4.1758	101.1898	2.4923	2.85	44	88	141	281.2548	24.05747	11.28041		173.5002	292.2806			0.5187	-0.0005
Profile 0	6.919	4.1758	101.1898	2.4923	2.85	44	42	163	389.4036	27.64872	12.12272		1464.451	3917.869			0.4134	0.0001
Profile 0	6.919	4.1758	101.1898	2.4923	2.85	44	88	163	309.2431	22.93639	10.17009		190.5132	328.8894			0.5174	-0.0004
Profile 0	6.919	4.1758	101.1898	2.4923	2.85	56	42	141	28.74584	4.063264	1.916571		17.7419	33.77536			0.5067	-0.0025
Profile 0	6.919	4.1758	101.1898	2.4923	2.85	56	42	163	22.59956	3.667752	1.900312		57.2104	96.95606			0.3776	0.0081
Profile 0	6.919	4.1758	101.1898	2.4923	2.85	56	88	163	27.96083	3.434393	1.546322		16.72438	33.21047			0.5053	-0.0023
Profile 3	6.885	3.8823	92.79933	2.025	10.65	44	42	141	3.373456	1.514346	1.460411		0.738202	0.459965			0.2879	0.2662
Profile 3	6.885	3.8823	92.79933	2.025	10.65	56	42	141	0.817194	0.448823	0.3865		0.243178	0.14971			0.2931	1.1725
Profile 3	6.885	3.8823	92.79933	2.025	10.65	44	88	141	2.413932	1.197223	0.954443		0.585228	0.388173			0.2854	0.3858
Profile 3	6.885	3.8823	92.79933	2.025	10.65	44	42	163	2.847574	1.215136	1.169297		0.570953	0.346557			0.2863	0.31
Profile 3	6.885	3.8823	92.79933	2.025	10.65	44	88	163	1.99354	0.948309	0.747739		0.446915	0.288852			0.2842	0.4582
Profile 3	6.885	3.8823	92.79933	2.025	10.65	56	88	141	0.602282	0.334503	0.262437		0.222895	0.130994			0.2894	1.684
Profile 3	6.885	3.8823	92.79933	2.025	10.65	56	42	163	0.631393	0.336977	0.288187		0.178187	0.107969			0.2914	1.4986
Profile 3	6.885	3.8823	92.79933	2.025	10.65	56	88	163	0.457536	0.247956	0.19242		0.162977	0.093983			0.2883	2.1834
Profile 3	6.885	3.8823	92.79933	2.025	2.85	44	42	141	83.39211	29.21965	13.48963		9.529607	5.983771			0.2893	0.009
Profile 3	6.885	3.8823	92.79933	2.025	2.85	56	42	141	10.35142	4.347604	2.316861		2.027099	1.423066			0.2899	0.0788
Profile 3	6.885	3.8823	92.79933	2.025	2.85	44	88	141	60.789	24.05747	11.28041		7.174225	3.201106			0.2869	0.0127
Profile 3	6.885	3.8823	92.79933	2.025	2.85	44	42	163	82.86573	27.64872	12.12272		8.517752	5.18591			0.2902	0.0089
Profile 3	6.885	3.8823	92.79933	2.025	2.85	44	88	163	61.18923	22.93639	10.17009		6.394403	2.661951			0.2872	0.0124
Profile 3	6.885	3.8823	92.79933	2.025	2.85	56	88	141	7.999263	4.063264	1.916571		1.522124	0.859162			0.2938	0.1012
Profile 3	6.885	3.8823	92.79933	2.025	2.85	56	42	163	9.216691	3.667752	1.900312		1.648542	1.137693			0.2901	0.0864
Profile 3	6.885	3.8823	92.79933	2.025	2.85	56	88	163	7.103952	3.434393	1.546322		1.225604	0.661797			0.293	0.1117
Profile 4	6.853	3.8949	92.6935	2.0249	10.65	44	42	141	13.40988	19.0283	3.668597		3.724346	5.748771			0.3501	0.0492
Profile 4	6.853	3.8949	92.6935	2.0249	10.65	56	42	141	2.723485	3.029528	0.809003		0.755701	1.135795			0.3725	0.2235
Profile 4	6.853	3.8949	92.6935	2.0249	10.65	44	88	141	6.820638	8.477153	2.959205		2.965002	3.68744			0.3532	0.1247
Profile 4	6.853	3.8949	92.6935	2.0249	10.65	44	42	163	12.37799	17.74802	3.080172		3.207562	5.039936			0.3494	0.0519
Profile 4	6.853	3.8949	92.6935	2.0249	10.65	44	88	163	5.79996	7.173812	2.476994		2.519127	3.132384			0.3555	0.1436
Profile 4	6.853	3.8949	92.6935	2.0249	10.65	56	88	141	1.578883	2.002742	0.690796		0.676317	0.844015			0.3477	0.5542
Profile 4	6.853	3.8949	92.6935	2.0249	10.65	56	42	163	2.303828	2.563008	0.626542		0.586847	0.897325			0.3718	0.2568
Profile 4	6.853	3.8949	92.6935	2.0249	10.65	56	88	163	1.254079	1.593401	0.531037		0.521255	0.653808			0.3477	0.6822
Profile 4	6.853	3.8949	92.6935	2.0249	2.85	44	42	141	2807.372	68.47783	24.14349		15.79156	25.97168			0.3293	0.0002
Profile 4	6.853	3.8949	92.6935	2.0249	2.85	56	42	141	81.68644	9.037015	3.730098		2.583598	3.883739			0.3198	0.0066
Profile 4	6.853	3.8949	92.6935	2.0249	2.85	44	88	141	377.0292	59.10851	17.904		11.73951	19.22981			0.3135	0.0015
Profile 4	6.853	3.8949	92.6935	2.0249	2.85	44	42	163	2826.019	67.35192	22.5385		14.52112	24.68738			0.3293	0.0002
Profile 4	6.853	3.8949	92.6935	2.0249	2.85	44	88	163	380.8904	59.1766	16.93558		10.79009	18.32631			0.3134	0.0015
Profile 4	6.853	3.8949	92.6935	2.0249	2.85	56	88	141	43.59131	6.268664	3.112791		1.880291	2.616375			0.3145	0.0132
Profile 4	6.853	3.8949	92.6935	2.0249	2.85	56	42	163	80.0433	7.971223	3.133286		2.125201	3.274915			0.3207	0.0067
Profile 4	6.853	3.8949	92.6935	2.0249	2.85	56	88	163	40.3297	5.478353	2.60001		1.524491	2.164836			0.315	0.0141
Profile 6	6.862	3.7059	88.8964	1.7762	10.65	44	42	141	36.69231	4.890182	3.375592		1.800218	3.547317			0.3228	0.015
Profile 6	6.862	3.7059	88.8964	1.7762	10.65	56	42	141	4.777218	0.95307	0.722479		0.45525	0.892946			0.3291	0.1185
Profile 6	6.862	3.7059	88.8964	1.7762	10.65	44	88	141	12.77247	3.60807	2.411987		1.38508	2.834654			0.3219	0.0491
Profile 6	6.862	3.7059	88.8964	1.7762	10.65	44	42	163	35.66358	4.269726	2.868138		1.455794	3.025868			0.3233	0.0152
Profile 6	6.862	3.7059	88.8964	1.7762	10.65	44	88	163	11.15771	3.072734	2.001705		1.102745	2.374789			0.3217	0.0557
Profile 6	6.862	3.7059	88.8964	1.7762	10.65	56	88	141	2.84217	0.791954	0.593267		0.387793	0.611751			0.3171	0.231
Profile 6	6.862	3.7059	88.8964	1.7762	10.65	56	42	163	4.138805	0.749204	0.558663		0.34261	0.697028			0.329	0.1343
Profile 6	6.862	3.7059	88.8964	1.7762	10.65	56	88	163	2.331176	0.61355	0.452298		0.289734	0.467735			0.3173	0.2762
Profile 6	6.862	3.7059	88.8964	1.7762	2.85	44	42	141	205.4116	30.17345	21.02447		11.324	29.18939			0.331	0.0026
Profile 6	6.862	3.7059	88.8964	1.7762	2.85	56	42	141	23.7233	4.43532	3.271523		2.223054	3.657843			0.324	0.0243
Profile 6	6.862	3.7059	88.8964	1.7762	2.85	44	88	141	203.669	21.94826	19.0557		8.747824	17.87722			0.325	0.0026
Profile 6	6.862	3.7059	88.8964	1.7762	2.85	44	42	163	210.7415	28.69954	19.62456		10.15722	27.96083			0.3313	0.0025
Profile 6	6.862	3.7059	88.8964	1.7762	2.85	44	88	163	210.6081	20.80056	17.93391		7.790674	17.00787			0.3256	0.0025
Profile 6	6.862	3.7059	88.8964	1.7762	2.85	56	88	141	14.52613	3.314553	2.501641		1.597074	2.729135			0.3221	0.0418
Profile 6	6.862	3.7059	88.8964	1.7762	2.85	56	42	163	22.20624	3.767255	2.722701		1.813009	3.081059			0.3245	0.0254
Profile 6	6.862	3.7059	88.8964	1.7762	2.85	56	88	163	13.39754	2.770607	2.063954		1.2827	2.255148			0.3224	0.0442
Profile 7	6.847	3.6457	87.1383	1.7472	10.65	44	42	141	17.54284	5.652297	3.160458		2.649415	2.331578			0.2964	0.0421
Profile 7	6.847	3.6457	87.1383	1.7472	10.65	56	42	141	2.844952	1.105096	0.673636		0.593198	0.671583			0.3056	0.2684
Profile 7	6.847	3.6457	87.1383	1.7472	10.65	44	88	141	7.810881	3.839945	2.675777		1.965509	1.781763			0.2863	0.1199
Profile 7	6.847	3.6457	87.1383	1.7472	10.65	44	42	163	16.49681	4.856518	2.687355		2.218069	1.945696			0.2979	0.0433
Profile 7	6.847	3.6457	87.1383	1.7472	10.65	44	88	163	6.723958	3.231283	2.255537		1.61343	1.455878			0.2858	0.1376
Profile 7	6.847	3.6457	87.1383	1.7472	10.65	56	88	141	1.791637	0.871064	0.623089		0.500928	0.443685			0.2858	0.5346
Profile 7	6.847	3.6457	87.1383	1.7472	10.65	56	42	163	2.417408	0.877102	0.519727		0.454229	0.516357			0.3059	0.306
Profile 7	6.847	3.6457	87.1383	1.7472	10.65	56	88	163	1.439959	0.678539	0.478961		0.379839	0.333926			0.2858	0.65
Profile 7	6.847	3.6457	87.1383	1.7472	2.85	44	42	141	173.9001	27.35741	14.21592		20.89176	8.307589			0.3136	0.0034
Profile 7	6.847	3.6457	87.1383	1.7472	2.85	56	42	141	21.016	4.027402	2.416991		2.97235	1.460663			0.3092	0.0299
Profile 7	6.847	3.6457	87.1383	1.7472	2.85	44	88	141	141.1968	20.34581	10.88993		11.60981	5.39697			0.3111	0.0042
Profile 7	6.847	3.6457	87.1383	1.7472	2.85	44	42	163	177.5722	25.76618	12.96358		19.3019	7.310128			0.3145	0.0033
Profile 7	6.847	3.6457	87.1383	1.7472	2.85	44	88	163	144.6355	19.22096	9.904327		10.60413	4.667937			0.3123	0.004
Profile 7	6.847	3.6457	87.1383	1.7472	2.85	56	88	141	15.93034	3.133827	1.807902		1.981983	1.175303			0.308	0.0397
Profile 7	6.847	3.6457	87.1383	1.7472	2.85	56	42	163	19.49171	3.383374	1.976628		2.44076	1.161917			0.3107	0.0314
Profile 7	6.847	3.6457	87.1383	1.7472	2.85	56	88	163	14.58814	2.606304	1.461083		1.609626	0.923368			0.3096	0.0422

	Profile 8	6.776	3.2309	74.9002	1.4918	10.65	44	42	141	0.365006	0.321588	0.306284			0.70587	1.063592	0.4036	0.7603
	Profile 8	6.776	3.2309	74.9002	1.4918	10.65	56	42	141	0.120462	0.109358	0.104046			0.233736	0.310617	0.3971	2.8363
	Profile 8	6.776	3.2309	74.9002	1.4918	10.65	44	88	141	0.33863	0.323519	0.276678			0.590677	0.910752	0.4074	0.9158
	Profile 8	6.776	3.2309	74.9002	1.4918	10.65	44	42	163	0.270131	0.237342	0.225463			0.538115	0.832051	0.405	0.9396
	Profile 8	6.776	3.2309	74.9002	1.4918	10.65	44	88	163	0.251044	0.239111	0.203294			0.446529	0.71015	0.4088	1.1338
	Profile 8	6.776	3.2309	74.9002	1.4918	10.65	56	88	141	0.114314	0.101251	0.097196			0.193319	0.27091	0.3992	3.4662
	Profile 8	6.776	3.2309	74.9002	1.4918	10.65	56	42	163	0.085852	0.077844	0.073744			0.170383	0.22856	0.3976	3.7731
	Profile 8	6.776	3.2309	74.9002	1.4918	10.65	56	88	163	0.08133	0.071755	0.068818			0.13987	0.19877	0.4	4.607
	Profile 8	6.776	3.2309	74.9002	1.4918	2.85	44	42	141	2.808342	2.381634	1.671764			2.254109	3.258179	0.4473	0.1745
	Profile 8	6.776	3.2309	74.9002	1.4918	2.85	56	42	141	0.700406	0.566304	0.428524			0.568166	0.762737	0.4753	0.2516
	Profile 8	6.776	3.2309	74.9002	1.4918	2.85	44	88	141	2.082452	1.580247	1.086551			1.883215	2.391801	0.484	0.0308
	Profile 8	6.776	3.2309	74.9002	1.4918	2.85	44	42	163	2.290868	1.952203	1.349041			1.820853	2.690915	0.442	0.2349
	Profile 8	6.776	3.2309	74.9002	1.4918	2.85	44	88	163	1.68374	1.265465	0.847813			1.518623	1.962117	0.4814	0.052
	Profile 8	6.776	3.2309	74.9002	1.4918	2.85	56	88	141	0.539759	0.450324	0.325555			0.506728	0.687464	0.4411	1.0541
	Profile 8	6.776	3.2309	74.9002	1.4918	2.85	56	42	163	0.53518	0.428672	0.320221			0.427391	0.581768	0.4762	0.2996
	Profile 8	6.776	3.2309	74.9002	1.4918	2.85	56	88	163	0.405695	0.336628	0.240049			0.380518	0.524536	0.4385	1.4236
	Profile 9	6.872	3.5198	84.3676	1.6432	10.65	44	42	141	32.91547	12.14018	6.351481			28.69954	35.08933	0.5175	-0.0052
	Profile 9	6.872	3.5198	84.3676	1.6432	10.65	56	42	141	4.359383	2.246079	1.167481			5.310067	1.167481	0.4833	0.0094
	Profile 9	6.872	3.5198	84.3676	1.6432	10.65	44	88	141	10.79692	7.359105	4.262604			12.13878	13.58939	0.4711	0.0138
	Profile 9	6.872	3.5198	84.3676	1.6432	10.65	44	42	163	31.7724	10.99512	5.665979			26.75777	33.55444	0.5181	-0.0053
	Profile 9	6.872	3.5198	84.3676	1.6432	10.65	44	88	163	9.3848	6.514784	3.642923			10.53659	11.90556	0.4687	0.0179
	Profile 9	6.872	3.5198	84.3676	1.6432	10.65	56	88	141	2.482276	1.531087	0.947545			2.419497	2.51507	0.5171	-0.1086
	Profile 9	6.872	3.5198	84.3676	1.6432	10.65	56	42	163	3.81022	1.827469	0.929929			4.594095	4.727157	0.4888	0.002
	Profile 9	6.872	3.5198	84.3676	1.6432	10.65	56	88	163	2.025816	1.224123	0.743575			1.919884	2.010134	0.5175	-0.1287
	Profile 9	6.872	3.5198	84.3676	1.6432	2.85	44	42	141	58.96916	17.64209	13.16512			64.41322	45.36021	0.5067	-0.0016
	Profile 9	6.872	3.5198	84.3676	1.6432	2.85	56	42	141	7.788431	2.905026	2.368645			8.267515	5.771318	0.5014	-0.009
	Profile 9	6.872	3.5198	84.3676	1.6432	2.85	44	88	141	45.78782	12.67579	10.06236			50.29212	39.2532	0.5153	-0.0032
	Profile 9	6.872	3.5198	84.3676	1.6432	2.85	44	42	163	57.54399	16.24052	11.78895			63.83002	44.4401	0.507	-0.0016
	Profile 9	6.872	3.5198	84.3676	1.6432	2.85	44	88	163	45.22723	11.6728	9.080296			50.76332	39.18096	0.5159	-0.0032
	Profile 9	6.872	3.5198	84.3676	1.6432	2.85	56	88	141	5.819692	2.299853	1.757013			6.701932	5.21705	0.5144	-0.0273
	Profile 9	6.872	3.5198	84.3676	1.6432	2.85	56	42	163	6.825351	2.387399	1.928413			7.303819	4.993729	0.5011	-0.0096
Side-Wear	Profile 9	6.872	3.5198	84.3676	1.6432	2.85	56	88	163	5.039355	1.87629	1.417751			5.883693	4.525588	0.5147	-0.0307
	Profile 2	7.185	4.9248	126.2557	2.93	10.65	44	42	141	11.9874	10.91001	10.9509	18.11027	5.127138			0.3845	0.0566
	Profile 2	7.185	4.9248	126.2557	2.93	10.65	56	42	141	2.013608	1.608607	1.903159	2.882704	0.995921			0.3769	0.3977
	Profile 2	7.185	4.9248	126.2557	2.93	10.65	44	88	141	8.379633	6.705405	8.118485	11.27068	4.187454			0.3831	0.0974
	Profile 2	7.185	4.9248	126.2557	2.93	10.65	44	42	163	10.91001	9.741481	9.86166	16.71379	4.456563			0.3824	0.0608
	Profile 2	7.185	4.9248	126.2557	2.93	10.65	44	88	163	7.428481	5.865094	7.191591	10.25888	3.605164			0.3817	0.1059
	Profile 2	7.185	4.9248	126.2557	2.93	10.65	56	88	141	1.645792	1.375942	1.58389	2.194952	0.877354			0.3821	0.5387
	Profile 2	7.185	4.9248	126.2557	2.93	10.65	56	42	163	1.642574	1.294047	1.539395	2.375883	0.780684			0.3756	0.4747
	Profile 2	7.185	4.9248	126.2557	2.93	10.65	56	88	163	1.317498	1.090373	1.275851	1.786282	0.681475			0.3828	0.6401
	Profile 2	7.185	4.9248	126.2557	2.93	2.85	44	42	141	995119.2	148072.7	186315.9	247001.7	6614.545			0	0.0001
	Profile 2	7.185	4.9248	126.2557	2.93	2.85	56	42	141	6906.769	3405.649	4170.373	4658.542	161.9571			0.378	0.0001
	Profile 2	7.185	4.9248	126.2557	2.93	2.85	44	88	141	85285.45	54218.82	60492.27	47629.39	3136.714			0.3754	0
	Profile 2	7.185	4.9248	126.2557	2.93	2.85	44	42	163	1016951	222177.5	196042.5	267963.2	6825.352			0	0.0001
	Profile 2	7.185	4.9248	126.2557	2.93	2.85	44	88	163	90938.97	57395.13	64576.58	50871.56	3278.688			0.3755	0
	Profile 2	7.185	4.9248	126.2557	2.93	2.85	56	88	141	637.1255	477.8593	415.6473	369.9346	83.47858			0.3704	0.0009
	Profile 2	7.185	4.9248	126.2557	2.93	2.85	56	42	163	7148.255	3466.172	4272.676	5772.98	165.8632			0.3924	0.0001
	Profile 2	7.185	4.9248	126.2557	2.93	2.85	56	88	163	703.5986	523.9623	457.2198	407.3802	86.70118			0.3705	0.0008
	Profile 3	7.13	4.8282	122.4528	2.9746	10.65	44	42	141	16.45603	18.50653	15.25194	22.1093	34.57204			0.4106	0.0202
	Profile 3	7.13	4.8282	122.4528	2.9746	10.65	56	42	141	2.933594	2.791579	2.551967	3.512975	4.719544			0.4031	0.22
	Profile 3	7.13	4.8282	122.4528	2.9746	10.65	44	88	141	11.85018	13.83089	10.75041	19.2287	27.24112			0.4008	0.0255
	Profile 3	7.13	4.8282	122.4528	2.9746	10.65	44	42	163	15.25106	17.32008	13.93879	20.62885	33.18372			0.4109	0.0204
	Profile 3	7.13	4.8282	122.4528	2.9746	10.65	44	88	163	10.72754	12.64008	9.780814	17.97836	26.17128			0.4021	0.0252
	Profile 3	7.13	4.8282	122.4528	2.9746	10.65	56	88	141	2.162096	2.38424	2.065737	2.598962	4.019759			0.4156	0.1874
	Profile 3	7.13	4.8282	122.4528	2.9746	10.65	56	42	163	2.450897	2.307411	2.086292	2.916923	4.029721			0.4036	0.2472
	Profile 3	7.13	4.8282	122.4528	2.9746	10.65	56	88	163	1.763498	1.958957	1.678128	2.134027	3.400752			0.4161	0.2115
	Profile 3	7.13	4.8282	122.4528	2.9746	2.85	44	42	141	123879.7	59319.83	64235.49	47634.86	48153.19			0.3158	0
	Profile 3	7.13	4.8282	122.4528	2.9746	2.85	56	42	141	2906.867	1482.518	913.5345	464.6223	496.1066			0.3144	0.0002
	Profile 3	7.13	4.8282	122.4528	2.9746	2.85	44	88	141	23284.93	39822.18	38563.37	30262.16	24867.1			0.4677	0
	Profile 3	7.13	4.8282	122.4528	2.9746	2.85	44	42	163	129055	60325.36	65629.64	48576.37	48955.33			0.316	0
	Profile 3	7.13	4.8282	122.4528	2.9746	2.85	44	88	163	42961.07	41376.15	40290.25	31668.32	25941.79			0.3894	0
	Profile 3	7.13	4.8282	122.4528	2.9746	2.85	56	88	141	632.776	564.6768	386.3892	271.597	186.6917			0.3561	0.0011
	Profile 3	7.13	4.8282	122.4528	2.9746	2.85	56	42	163	2945.1	1460.999	911.3286	745.4183	524.1131			0.3094	0.0002
	Profile 3	7.13	4.8282	122.4528	2.9746	2.85	56	88	163	700.0837	734.1335	424.0333	295.3589	201.674			0.3633	0.0009
	Profile 4	7.041	4.6517	116.5273	2.8393	10.65	44	42	141	36.11395	32.53681	44.2461	42.00732	26.83336			0.3446	0.052
	Profile 4	7.041	4.6517	116.5273	2.8393	10.65	56	42	141	4.79347	6.077151	7.63616	6.482985	4.649701			0.4463	0.0289
	Profile 4	7.041	4.6517	116.5273	2.8393	10.65	44	88	141	29.88822	26.9138	20.59326	26.85035	22.77063			0.3966	0.0329
	Profile 4	7.041	4.6517	116.5273	2.8393	10.65	44	42	163	34.35184	30.69552	42.60151	41.01332	25.32941			0.3372	0.0544
	Profile 4	7.041	4.6517	116.5273	2.8393	10.65	44	88	163	28.63848	25.37319	19.15689	26.50483	21.3944			0.4054	0.0286
	Profile 4	7.041	4.6517	116.5273	2.8393	10.65	56	88	141	3.796645	4.536282	5.571857	5.842853	3.937993			0.4301	0.0744
	Profile 4	7.041	4.6517	116.5273	2.8393	10.65	56	42	163	4.155516	4.342102	6.839117	5.660437	4.007744			0.4429	0.046
	Profile 4	7.041	4.6517	116.5273	2.8393	10.65	56	88	163	3.270958	3.831775	4.867153	5.116818	3.342912			0.4325	0.0791
	Profile 4	7.041	4.6517	116.5273	2.8393	2.85	44	42	141	9.364832	9.561477	12.37585	16.00755	14.37308			0.3812	0.0608
	Profile 4	7.041	4.6517	116.5273	2.8393	2.85	56	42	141	2.112881	1.812174	2.151295	2.690141	2.487998			0.396	0.5698
	Profile 4	7.041	4.6517	116.5273	2.8393	2.85	44	88	141	6.717768	6.670755	10.82742	18.63267	15.18711			0.3906	0.0321
	Profile 4	7.041	4.6517	116.5273	2.8393	2.85	44	42	163	8.216749	8.436746	11.06305	14.60915	13.02717			0.3813	0.0632
	Profile 4	7.041	4.6517	116.5273	2.8393	2.85	44	88	163	5.830086	5.817012	9.892362	17.84637	12.92707			0.398	0.0303
	Profile 4	7.041	4.6517	116.5273	2.8393	2.85	56	88	141	1.411318	1.423066	1.635875	2.329834	2.274705			0.3886	0.3975
	Profile 4	7.041	4.6517	116.5273	2.8393	2.85	56	42	163	1.724547	1.455794	1.746124	2.207623	2.028266			0.3998	0.6419
	Profile 4	7.041	4.6517	116.5273	2.8393	2.85	56	88	163	1.120986	1.131684	1.306622	1.914366	1.887883			0.3889	0.4544
	Profile 6	6.916	4.1854	101.2489	2.4377	10.65	44	42	141	0.382671	0.395777	0.458881	1.054083	1.072198			0.3954	0.4945
	Profile 6	6.916	4.1854	101.2489	2.4377	10.65	56	42	141	0.127262	0.12907	0.149839	0.308816	0.326701			0.395	1.7776
	Profile 6	6.916	4.1854	101.2489	2.4377	10.65	44	88	141	0.335583	0.376422	0.481282	0.702466	0.730592			0.3727	1.0564
	Profile 6	6.916	4.1854	101.2489	2.4377	10.65	44	42	163	0.283563	0.295206	0.344747	0.823616	0.846301			0.3953	0.6116
	Profile 6	6.916	4.1854	101.2489	2.4377	10.65	44	88	163	0.247642	0.28064	0.363747	0.540599	0.562698			0.3727	1.3242
	Profile 6	6.916	4.1854	101.2489	2.4377	10.65	56	88	141	0.116051	0.127923	0.142832	0.221017	0.245881			0.3838	3.151
	Profile 6	6.916	4.1854	101.2489	2.4377	10.65	56	42	163	0.090793	0.09231	0.10764	0.228139	0.242187			0.3951	2.3445
	Profile 6	6.916	4.1854	101.2489	2.4377	10.65	56	88	163	0.08268	0.09148	0.102394	0.161157	0.180146			0.3841	4.1917
	Profile 6	6.916	4.1854	101.2489	2.4377	2.85	44	42	141	67100.37	5650.346	12729.17	3728.851	8725.695			0.3365	0
	Profile 6	6.916	4.1854	101.2489	2.4377	2.85	56	42	141	643.6504	163.6345	204.0915	100.5078	113.8872			0.3198	0.0009
	Profile 6	6.916	4.1854	101.2489	2.4377	2.85	44	88	141	1458.478	444.0174	1409.208	600.9315	2192.426			0.417	0.0002
	Profile 6	6.916	4.1854	101.2489	2.4377	2.85	44	42	163	68053.42	5782.292	13251.04	3882.622	8867.478			0.3365	0
	Profile 6	6.916	4.1854	101.2489	2.4377	2.85	44	88	163	1609.904	490.7383	1467.827	845.2301	2398.004			0.419	0.0002
	Profile 6	6.916	4.1854	101.2489	2.4377	2.85	56	88	141	71.48666	67.24732	70.22229	51.35113	86.9561			0.3957	0.0177
	Profile 6	6.916	4.1854	101.2489	2.4377	2.85	56	42	163	702.3846	167.0225	212.2267	101.7478	115.8977			0.3203	0.0008
	Profile 6	6.916	4.1854	101.2489	2.4377	2.85	56	88	163	73.44293	68.12397	67.50718	52.35703	87.24188			0.384	0.0191

	Profile																		
	Profile 8	6.822	3.9667	94.3027	2.3422	10.65	44	42	141	0.606073	0.722936	1.220113	1.899984	2.072287				0.3703	0.2955
	Profile 8	6.822	3.9667	94.3027	2.3422	10.65	56	42	141	0.193341	0.224001	0.339352	0.462408	0.521405				0.3671	1.3867
	Profile 8	6.822	3.9667	94.3027	2.3422	10.65	44	88	141	0.509096	0.593403	0.996782	1.334827	1.555966				0.367	0.4393
	Profile 8	6.822	3.9667	94.3027	2.3422	10.65	44	42	163	0.457272	0.552777	0.961834	1.542677	1.682771				0.3709	0.3516
	Profile 8	6.822	3.9667	94.3027	2.3422	10.65	44	88	163	0.381922	0.449599	0.778395	1.061512	1.247312				0.3677	0.5302
	Profile 8	6.822	3.9667	94.3027	2.3422	10.65	56	88	141	0.170716	0.191007	0.292129	0.36815	0.428845				0.3672	1.795
	Profile 8	6.822	3.9667	94.3027	2.3422	10.65	56	42	163	0.139894	0.16323	0.250712	0.347176	0.392329				0.3676	1.7968
	Profile 8	6.822	3.9667	94.3027	2.3422	10.65	56	88	163	0.123346	0.138365	0.21487	0.273259	0.319853				0.3677	2.3534
	Profile 8	6.822	3.9667	94.3027	2.3422	2.85	44	42	141	316.3552	239.3867	307.7159	270.3958	622.9813				0.4185	0.001
	Profile 8	6.822	3.9667	94.3027	2.3422	2.85	56	42	141	38.83515	22.55148	32.81897	28.37756	41.24298				0.4589	0.001
	Profile 8	6.822	3.9667	94.3027	2.3422	2.85	44	88	141	151.2168	131.0388	198.8955	262.0897	211.8483				0.3871	0.0035
	Profile 8	6.822	3.9667	94.3027	2.3422	2.85	44	42	163	328.9463	247.5854	321.3291	282.1305	672.241				0.4199	0.0009
	Profile 8	6.822	3.9667	94.3027	2.3422	2.85	44	88	163	155.9104	137.7923	214.4865	266.4711	219.6595				0.3814	0.0036
	Profile 8	6.822	3.9667	94.3027	2.3422	2.85	56	88	141	22.02546	15.44632	19.37091	15.9919	18.51292				0.4121	0.03
	Profile 8	6.822	3.9667	94.3027	2.3422	2.85	56	42	163	38.24063	21.08264	31.45756	27.04737	39.88871				0.462	0.0001
	Profile 8	6.822	3.9667	94.3027	2.3422	2.85	56	88	163	20.96163	14.40456	18.2789	14.68504	17.30414				0.4085	0.0315
	Profile 9	6.732	3.6137	83.7598	2.0219	10.65	44	42	141	0.9737	0.751061	1.033654	1.832103	2.046209				0.3889	0.3612
	Profile 9	6.732	3.6137	83.7598	2.0219	10.65	56	42	141	0.29134	0.23037	0.283482	0.45672	0.481615				0.3855	1.9882
	Profile 9	6.732	3.6137	83.7598	2.0219	10.65	44	88	141	0.693825	0.627733	0.928218	1.316588	1.714747				0.3878	0.4187
	Profile 9	6.732	3.6137	83.7598	2.0219	10.65	44	42	163	0.763836	0.575639	0.805703	1.482177	1.683643				0.39	0.4209
	Profile 9	6.732	3.6137	83.7598	2.0219	10.65	44	88	163	0.534349	0.476431	0.722188	1.042137	1.39412				0.3891	0.4947
	Profile 9	6.732	3.6137	83.7598	2.0219	10.65	56	88	141	0.237151	0.199136	0.256906	0.361535	0.455565				0.3933	1.8698
	Profile 9	6.732	3.6137	83.7598	2.0219	10.65	56	42	163	0.215179	0.167987	0.208954	0.343242	0.363141				0.3855	2.563
	Profile 9	6.732	3.6137	83.7598	2.0219	10.65	56	88	163	0.17343	0.144486	0.188962	0.269262	0.342965				0.3936	2.4078
	Profile 9	6.732	3.6137	83.7598	2.0219	2.85	44	42	141	5229.678	12187.09	5361.359	1935.085	21331.68				0.4145	0
	Profile 9	6.732	3.6137	83.7598	2.0219	2.85	56	42	141	149.1163	174.5119	127.4676	133.3521	164.2953				0.4097	0.0076
	Profile 9	6.732	3.6137	83.7598	2.0219	2.85	44	88	141	2459.518	1196.327	863.376	528.4757	2738.104				0.4276	0.0001
	Profile 9	6.732	3.6137	83.7598	2.0219	2.85	44	42	163	9847.476	12429.4	5633.132	2050.808	21276.49				0.3648	0
	Profile 9	6.732	3.6137	83.7598	2.0219	2.85	44	88	163	3040.536	1260.086	944.1151	607.5403	2925.499				0.4408	0
	Profile 9	6.732	3.6137	83.7598	2.0219	2.85	56	88	141	80.59813	96.68297	54.50358	52.81412	81.19887				0.439	0.0036
	Profile 9	6.732	3.6137	83.7598	2.0219	2.85	56	42	163	151.5218	177.6029	132.4189	136.7571	168.6359				0.4062	0.0082
	Profile 9	6.732	3.6137	83.7598	2.0219	2.85	56	88	163	82.53725	100.5599	56.12741	53.4226	83.74329				0.4379	0.0035
	Profile 10	6.706	3.6716	85.3254	2.1397	10.65	44	42	141	0.9778	1.272404	7.118689	4.353865	2.593582				0.4286	0.032
	Profile 10	6.706	3.6716	85.3254	2.1397	10.65	56	42	141	0.287806	0.311476	1.400796	0.941348	0.63566				0.4241	0.194
	Profile 10	6.706	3.6716	85.3254	2.1397	10.65	44	88	141	0.817194	0.959953	4.841445	3.200553	1.86638				0.4276	0.0495
	Profile 10	6.706	3.6716	85.3254	2.1397	10.65	44	42	163	0.760983	1.013095	6.212981	3.662899	2.126302				0.4302	0.0349
	Profile 10	6.706	3.6716	85.3254	2.1397	10.65	44	88	163	0.627986	0.750628	4.112918	2.642409	1.510167				0.4286	0.0561
	Profile 10	6.706	3.6716	85.3254	2.1397	10.65	56	88	141	0.259612	0.281903	1.170577	0.804128	0.530365				0.4249	0.2324
	Profile 10	6.706	3.6716	85.3254	2.1397	10.65	56	42	163	0.211166	0.229906	1.109366	0.725772	0.481587				0.4255	0.2347
	Profile 10	6.706	3.6716	85.3254	2.1397	10.65	56	88	163	0.189692	0.207623	0.914166	0.616311	0.398818				0.4258	0.2873
	Profile 10	6.706	3.6716	85.3254	2.1397	2.85	44	42	141	6.578471	6.506913	108.5738	43.60889	7.154018				0.4513	0.0008
	Profile 10	6.706	3.6716	85.3254	2.1397	2.85	56	42	141	1.329077	1.314846	12.0573	5.841508	1.417262				0.4496	0.0088
	Profile 10	6.706	3.6716	85.3254	2.1397	2.85	44	88	141	4.293634	3.627229	91.68008	35.50585	4.037151				0.4516	0.0009
	Profile 10	6.706	3.6716	85.3254	2.1397	2.85	44	42	163	5.674793	5.640596	108.8366	42.06782	6.224794				0.4516	0.0008
	Profile 10	6.706	3.6716	85.3254	2.1397	2.85	44	88	163	3.622221	3.011445	94.18895	34.62382	3.496431				0.4521	0.0009
	Profile 10	6.706	3.6716	85.3254	2.1397	2.85	56	88	141	1.077457	0.893151	9.41727	4.118841	1.148088				0.4512	0.0101
	Profile 10	6.706	3.6716	85.3254	2.1397	2.85	56	42	163	1.049361	1.037947	10.87552	5.050682	1.136515				0.4499	0.0094
	Profile 10	6.706	3.6716	85.3254	2.1397	2.85	56	88	163	0.784874	0.68612	8.480569	3.500862	0.922678				0.451	0.0113
	Profile 11	6.642	3.5026	80.2554	2.0523	10.65	44	42	141	0.519009	0.763045	1.134358	1.652913	1.898891				0.3605	0.3449
	Profile 11	6.642	3.5026	80.2554	2.0523	10.65	56	42	141	0.163493	0.224104	0.331532	0.447224	0.513216				0.3585	1.3806
	Profile 11	6.642	3.5026	80.2554	2.0523	10.65	44	88	141	0.493174	0.608275	0.934437	1.354098	1.811549				0.3778	0.3434
	Profile 11	6.642	3.5026	80.2554	2.0523	10.65	44	42	163	0.389135	0.584689	0.889048	1.316892	1.522124				0.3612	0.4189
	Profile 11	6.642	3.5026	80.2554	2.0523	10.65	44	88	163	0.369828	0.46161	0.722811	1.066903	1.450858				0.3788	0.4171
	Profile 11	6.642	3.5026	80.2554	2.0523	10.65	56	88	141	0.152467	0.200678	0.281158	0.382406	0.410464				0.357	1.8207
	Profile 11	6.642	3.5026	80.2554	2.0523	10.65	56	42	163	0.117537	0.163174	0.24416	0.33264	0.384083				0.359	1.8093
	Profile 11	6.642	3.5026	80.2554	2.0523	10.65	56	88	163	0.10961	0.146007	0.206134	0.282569	0.304632				0.3572	2.4118
	Profile 11	6.642	3.5026	80.2554	2.0523	2.85	44	42	141	66.99232	22.16792	10.66351	7.694404	6.91393				0.3106	0.0086
	Profile 11	6.642	3.5026	80.2554	2.0523	2.85	56	42	141	7.887693	3.382984	1.949957	1.515741	1.390673				0.3099	0.0817
	Profile 11	6.642	3.5026	80.2554	2.0523	2.85	44	88	141	36.68597	15.18186	7.099455	5.217651	4.539155				0.311	0.0164
	Profile 11	6.642	3.5026	80.2554	2.0523	2.85	44	42	163	65.77336	20.75989	9.536741	6.755383	6.038443				0.3109	0.0086
	Profile 11	6.642	3.5026	80.2554	2.0523	2.85	44	88	163	36.04541	14.18241	6.274441	4.51414	3.871685				0.3109	0.0164
	Profile 11	6.642	3.5026	80.2554	2.0523	2.85	56	88	141	5.507126	2.580329	1.485252	1.170981	1.069547				0.3106	0.1203
	Profile 11	6.642	3.5026	80.2554	2.0523	2.85	56	42	163	7.027082	2.822929	1.574255	1.204967	1.104269				0.3099	0.089
Top-Wear	Profile 11	6.642	3.5026	80.2554	2.0523	2.85	56	88	163	4.808394	2.130345	1.1857	0.919073	0.835122				0.3104	0.1339
	N/A	7.296	5.205	136.4182	3.059	10.65	44	42	141	1.316513		0.505796		0.849914	1.510167	3.170662		0.519	0.3592
	N/A	7.296	5.205	136.4182	3.059	10.65	56	42	141	0.356123		0.16684		0.245132	0.404529	0.712279		0.5066	1.8702
	N/A	7.296	5.205	136.4182	3.059	10.65	44	88	141	0.865366		0.446735		0.585868	2.23036	2.776515		0.4442	0.4639
	N/A	7.296	5.205	136.4182	3.059	10.65	44	42	163	1.028667		0.38234		0.662026	1.228146	2.697274		0.5213	0.4061
	N/A	7.296	5.205	136.4182	3.059	10.65	44	88	163	0.672977		0.337074		0.44581	1.828626	2.343149		0.4508	0.5245
	N/A	7.296	5.205	136.4182	3.059	10.65	56	88	141	0.249689		0.149331		0.179546	0.286682	0.473015		0.499	3.2082
	N/A	7.296	5.205	136.4182	3.059	10.65	56	42	163	0.266164		0.120524		0.179773	0.303511	0.551093		0.508	2.3515
	N/A	7.296	5.205	136.4182	3.059	10.65	56	88	163	0.183242		0.107696		0.130137	0.211495	0.35639		0.5007	4.1366
	N/A	7.296	5.205	136.4182	3.059	2.85	44	42	141	810.2612		1040.579		700.0032	1460.915	2425.074		0.5111	0.0005
	N/A	7.296	5.205	136.4182	3.059	2.85	56	42	141	38.84633		34.25705		48.61553	96.90028	61.01688		0.5547	0.0098
	N/A	7.296	5.205	136.4182	3.059	2.85	44	88	141	249.9338		124.0295		163.7759	486.7713	655.0131		0.4411	0.0022
	N/A	7.296	5.205	136.4182	3.059	2.85	44	42	163	833.2013		1114.744		751.6229	2110.085	2557.997		0.4643	0.0005
	N/A	7.296	5.205	136.4182	3.059	2.85	44	88	163	280.5272		148.175		175.3174	540.2565	722.0213		0.4375	0.0021
	N/A	7.296	5.205	136.4182	3.059	2.85	56	88	141	25.4727		11.26614		9.437349	23.55863	44.32768		0.5282	0.0231
	N/A	7.296	5.205	136.4182	3.059	2.85	56	42	163	39.02339		34.08592		49.19546	97.25233	62.14411		0.5508	0.0099
Unworn	N/A	7.296	5.205	136.4182	3.059	2.85	56	88	163	25.01641		10.29497		8.607956	22.98	44.57589		0.5263	0.0226

Appendix E.6 MATLAB Fracture Mechanics Post-Processing Code

```
%% Fracture Mechanics da/dN Post-Processing:
clear all
clc
%% Define Crack Sizes:
Crack_5Percent = 0.33241;
Crack_7_5Percent = 0.40712;
Crack_10Percent = 0.4701;
Crack_12_5Percent = 0.52559;
Crack_15Percent = 0.57576;
Crack_20Percent = 0.66483;
Crack_35Percent = 0.87948;
Crack_50Percent = 1.05119;
a = zeros(256,1);
b = zeros(256,1);
%% Import Data For SideWear Profiles
Nf5Percent = xlsread('ResultsForResponseSurface.xlsx', 1, 'K2:K113');
Nf7_5Percent = xlsread('ResultsForResponseSurface.xlsx', 1, 'L2:L113');
Nf10Percent = xlsread('ResultsForResponseSurface.xlsx', 1, 'M2:M113');
Nf15Percent = xlsread('ResultsForResponseSurface.xlsx', 1, 'O2:O113');
Nf20Percent = xlsread('ResultsForResponseSurface.xlsx', 1, 'P2:P113');
CrackRadius = [Crack_5Percent; Crack_7_5Percent; Crack_10Percent;
Crack_15Percent; Crack_20Percent];
for i = 1:112
    Nf = [Nf5Percent(i); Nf7_5Percent(i); Nf10Percent(i); Nf15Percent(i);
Nf20Percent(i)];
    Cycles = [0; abs(Nf(2)-Nf(1)); abs(Nf(3)-Nf(1)); abs(Nf(4)-Nf(1));
abs(Nf(5)-Nf(1))];
    Exponential = @(p,x) p(1).*exp(p(2).*x);
    startingVals = [1, 0.0001];
    coefEsts = nlinfit(Cycles, CrackRadius, Exponential, startingVals);
    a(i) = coefEsts(1);
    b(i) = coefEsts(2);
end
%% For TopWear Profiles
Nf5Percent = xlsread('ResultsForResponseSurface.xlsx', 1, 'K114:K241');
Nf7_5Percent = xlsread('ResultsForResponseSurface.xlsx', 1, 'L114:L241');
Nf10Percent = xlsread('ResultsForResponseSurface.xlsx', 1, 'M114:M241');
Nf12_5Percent = xlsread('ResultsForResponseSurface.xlsx', 1, 'N114:N241');
Nf15Percent = xlsread('ResultsForResponseSurface.xlsx', 1, 'O114:O241');
CrackRadius = [Crack_5Percent; Crack_7_5Percent; Crack_10Percent;
Crack_12_5Percent; Crack_15Percent];
counter = 1;
for i = 113:240
    Nf = [Nf5Percent(counter); Nf7_5Percent(counter); Nf10Percent(counter);
Nf12_5Percent(counter); Nf15Percent(counter)];
    Cycles = [0; abs(Nf(2)-Nf(1)); abs(Nf(3)-Nf(1)); abs(Nf(4)-Nf(1));
abs(Nf(5)-Nf(1))];
    Exponential = @(p,x) p(1).*exp(p(2).*x);
    startingVals = [1, 0.0001];
    coefEsts = nlinfit(Cycles, CrackRadius, Exponential, startingVals);
    a(i) = coefEsts(1);
    b(i) = coefEsts(2);
    counter = counter + 1;
end
%% For Unworn Profiles
```

```matlab
Nf5Percent = xlsread('ResultsForResponseSurface.xlsx', 1, 'K242:K257');
Nf10Percent = xlsread('ResultsForResponseSurface.xlsx', 1, 'M242:M257');
Nf20Percent = xlsread('ResultsForResponseSurface.xlsx', 1, 'O242:O257');
Nf35Percent = xlsread('ResultsForResponseSurface.xlsx', 1, 'Q242:Q257');
Nf50Percent = xlsread('ResultsForResponseSurface.xlsx', 1, 'R242:R257');
CrackRadius = [Crack_5Percent; Crack_10Percent; Crack_20Percent;
Crack_35Percent; Crack_50Percent];
counter = 1;
for i = 241:256
    Nf = [Nf5Percent(counter); Nf10Percent(counter); Nf20Percent(counter);
Nf35Percent(counter); Nf50Percent(counter)];
    Cycles = [0; abs(Nf(2)-Nf(1)); abs(Nf(3)-Nf(1)); abs(Nf(4)-Nf(1));
abs(Nf(5)-Nf(1))];
    Exponential = @(p,x) p(1).*exp(p(2).*x);
    startingVals = [1, 0.0001];
    coefEsts = nlinfit(Cycles, CrackRadius, Exponential, startingVals);
    a(i) = coefEsts(1);
    b(i) = coefEsts(2);
    counter = counter + 1;
end
%% Calculate da/dN at different crack sizes
x = 0:0.01:max(Cycles);
plot(a(1)*exp(b(1)*x))
figure
plot(a(240)*exp(b(24)*x))
figure
plot(a(256)*exp(b(256)*x))
```

Appendix E.7 MATLAB Response Surface Generation Code

Main Code:

```
%% Rail Life and Crack Growth Rate Response Surface Generation Code
clear all
clc
%% Import Independent Variable Data
h = xlsread('ResultsForResponseSurface.xlsx', 1, 'C2:C257');
hMax = max(h);
hMin = min(h);
hBar = mean(h);
A = xlsread('ResultsForResponseSurface.xlsx', 1, 'D2:D257');
AMax = max(A);
AMin = min(A);
ABar = mean(A);
Ixx = xlsread('ResultsForResponseSurface.xlsx', 1, 'E2:E257');
IxxMax = max(Ixx);
IxxMin = min(Ixx);
IxxBar = mean(Ixx);
Iyy = xlsread('ResultsForResponseSurface.xlsx', 1, 'F2:F257');
IyyMax = max(Iyy);
IyyMin = min(Iyy);
IyyBar = mean(Iyy);
P = xlsread('ResultsForResponseSurface.xlsx', 1, 'G2:G257');
PMax = 60;
PMin = 40;
PBar = (PMax + PMin)/2;
C = xlsread('ResultsForResponseSurface.xlsx', 1, 'H2:H257');
CMax = 13.5;
CMin = 0;
CBar = (CMax + CMin)/2;
deltaT = xlsread('ResultsForResponseSurface.xlsx', 1, 'I2:I257');
deltaTMax = 105;
deltaTMin = 25;
deltaTBar = (deltaTMax + deltaTMin)/2;
SUT = xlsread('ResultsForResponseSurface.xlsx', 1, 'J2:J257');
SUTMax = 171;
SUTMin = 133;
SUTBar = (SUTMax + SUTMin)/2;
%% Import Data from FE-Safe for Rail Life with 10% HA Crack Response:
Nf = xlsread('ResultsForResponseSurface.xlsx', 1, 'M2:M257');
%% Import Data from Post-Processing for Rail da/dN Equation Response:
dadNConstant = xlsread('ResultsForResponseSurface.xlsx', 1, 'S2:S257');
dadNExponent = xlsread('ResultsForResponseSurface.xlsx', 1, 'T2:T257');
%% Define Coded Variables, and Develop Global Interpolation Function Matrix
for Life Estimation:
NSamples = 256;
NNodes = 256;
Ng = zeros(NSamples, NNodes);
for i = 1:NSamples
    zeta_h = (h(i,1) - hBar)/((hMax-hMin)/2);
    zeta_A = (A(i,1) - ABar)/((AMax-AMin)/2);
    zeta_Ixx = (Ixx(i,1) - IxxBar)/((IxxMax-IxxMin)/2);
    zeta_Iyy = (Iyy(i,1) - IyyBar)/((IyyMax-IyyMin)/2);
    zeta_P = (P(i,1) - PBar)/((PMax-PMin)/2);
    zeta_C = (C(i,1) - CBar)/((CMax-CMin)/2);
```

```
    zeta_deltaT = (deltaT(i,1) - deltaTBar)/((deltaTMax-deltaTMin)/2);
    zeta_SUT = (SUT(i,1) - SUTBar)/((SUTMax-SUTMin)/2);
    zeta = [zeta_h, zeta_A, zeta_Ixx, zeta_Iyy, zeta_P, zeta_C, zeta_deltaT,
zeta_SUT];
    [N] = InterpolationFunctions(NNodes, zeta);
    for j = 1:NNodes
        Ng(i,j) = N(1,j);
    end
end
%% Calculate Life Response Nodal Values (For 10% Crack)
B = Ng'*Ng;
n1 = B\Ng'*Nf;
%% Calculate da/dN Equation Constant and Exponent Response Nodal Values
(da/dN = ae^bt):
n2 = B\Ng'*dadNConstant;
n3 = B\Ng'*dadNExponent;
```

Sub-Functions:

```
function [N] = InterpolationFunctions(NNodes, zeta)
%Calcualte the interpoltion functions N1 through NNodes
syms z;
N1 = 1/2*(1 - z);
N2 = 1/2*(1 + z);
N = zeros(1,NNodes);
counter = 0;
for i = 1:2
    if i == 1
        Ni = N1;
    elseif i == 2
        Ni = N2;
    end
    for j = 1:2
        if j == 1
            Nj = N1;
        elseif j == 2
            Nj = N2;
        end
        for k = 1:2
            if k == 1
                Nk = N1;
            elseif k == 2
                Nk = N2;
            end
            for l = 1:2
                if l == 1
                    Nl = N1;
                elseif l == 2
                    Nl = N2;
                end
                for m = 1:2
                    if m == 1
                        Nm = N1;
                    elseif m == 2
                        Nm = N2;
```

```matlab
                        end
                        for n = 1:2
                            if n == 1
                                Nn = N1;
                            elseif n == 2
                                Nn = N2;
                            end
                            for o = 1:2
                                if o == 1
                                    No = N1;
                                elseif o == 2
                                    No = N2;
                                end
                                for p = 1:2
                                    if p == 1
                                        Np = N1;
                                    elseif p == 2
                                        Np = N2;
                                    end
                                        counter = counter + 1;
                                        N(1,counter) = subs(Ni, z,
zeta(1,1))*subs(Nj, z, zeta(1,2))*subs(Nk, z, zeta(1,3))*subs(Nl, z,
zeta(1,4))*subs(Nm, z, zeta(1,5))*subs(Nn, z, zeta(1,6))*subs(No, z,
zeta(1,7))*subs(Np, z, zeta(1,8));
                                end
                            end
                        end
                    end
                end
            end
        end
end

end
```

Appendix E.8 MATLAB Response Surface Code

```
%% Rail Life and Crack Growth Rate Response Surface Code
clear all
clc
NNodes = 256;
%% Load Response Surface Nodal Values
n1 = xlsread('ResultsForResponseSurface.xlsx', 1, 'AA3:AA258');
n2 = xlsread('ResultsForResponseSurface.xlsx', 1, 'AB3:AB258');
n3 = xlsread('ResultsForResponseSurface.xlsx', 1, 'AC3:AC258');
%% Define Independent Variables:
h = 6.919; %inches
A = 4.1758; %inches^2
Ixx = 101.1898; %inches ^4
Iyy = 2.4923; %inches ^4
P = 40000; % lbs.
C = 4.0; % degrees
deltaT = 50; %degrees F
SUT = 150; %ksi
%% Define Independent Variable Ranges:
hMax = max(xlsread('ResultsForResponseSurface.xlsx', 1, 'C2:C257'));
hMin = min(xlsread('ResultsForResponseSurface.xlsx', 1, 'C2:C257'));
hBar = mean(xlsread('ResultsForResponseSurface.xlsx', 1, 'C2:C257'));
AMax = max(xlsread('ResultsForResponseSurface.xlsx', 1, 'D2:D257'));
AMin = min(xlsread('ResultsForResponseSurface.xlsx', 1, 'D2:D257'));
ABar = mean(xlsread('ResultsForResponseSurface.xlsx', 1, 'D2:D257'));
IxxMax = max(xlsread('ResultsForResponseSurface.xlsx', 1, 'E2:E257'));
IxxMin = min(xlsread('ResultsForResponseSurface.xlsx', 1, 'E2:E257'));
IxxBar = mean(xlsread('ResultsForResponseSurface.xlsx', 1, 'E2:E257'));
IyyMax = max(xlsread('ResultsForResponseSurface.xlsx', 1, 'F2:F257'));
IyyMin = min(xlsread('ResultsForResponseSurface.xlsx', 1, 'F2:F257'));
IyyBar = mean(xlsread('ResultsForResponseSurface.xlsx', 1, 'F2:F257'));
PMax = 60000;
PMin = 40000;
PBar = (PMax + PMin)/2;
CMax = 13.5;
CMin = 0;
CBar = (CMax + CMin)/2;
deltaTMax = 105;
deltaTMin = 25;
deltaTBar = (deltaTMax + deltaTMin)/2;
SUTMax = 171;
SUTMin = 133;
SUTBar = (SUTMax + SUTMin)/2;
%% Code Independent Variables:
zeta_h = (h - hBar)/((hMax-hMin)/2);
zeta_A = (A - ABar)/((AMax-AMin)/2);
zeta_Ixx = (Ixx - IxxBar)/((IxxMax-IxxMin)/2);
zeta_Iyy = (Iyy - IyyBar)/((IyyMax-IyyMin)/2);
zeta_P = (P - PBar)/((PMax-PMin)/2);
zeta_C = (C - CBar)/((CMax-CMin)/2);
zeta_deltaT = (deltaT - deltaTBar)/((deltaTMax-deltaTMin)/2);
zeta_SUT = (SUT - SUTBar)/((SUTMax-SUTMin)/2);
zeta = [zeta_h, zeta_A, zeta_Ixx, zeta_Iyy, zeta_P, zeta_C, zeta_deltaT,
zeta_SUT];
[N] = InterpolationFunctions(NNodes, zeta);
%% Calculate Response Surface Estimates
```

```
Nf = N*n1;
dadN_a = N*n2;
dadN_b = N*n3;
```