Author: David Stewart
Supervisor: Dr. Francis Franklin
MEC8024

# Optimising the Ride Comfort and Stability of a Railway Bogie Through Numerical Analysis

Vehicle Dynamics

MEC2024

School of Engineering (Mechanical)

David Stewart – 180297339

Supervised by

Dr. Francis Franklin

13th May 2022

**Words – 3793**

**Pages - 6**

(Without Title Page, Contents and References)

Author: David Stewart
Supervisor: Dr. Francis Franklin
MEC8024

## Table of Contents



# Abstract

The following analysis describes the iterative method used to optimise the ride comfort and stability of a railway vehicle. A MATLAB script was used to simulate the journey of the vehicle travelling down a length of track, returning the spatial coordinates of the two wheelsets and bogie relative to the origin. Two equations were derived and used to represent the ride comfort and the stability of the vehicle associated with a set of input parameters as factors with a value between zero and one. The comfort of the ride was related to the magnitude of the rate of change of the resultant acceleration and the magnitude of the resultant acceleration. The stability of the vehicle was related to the maximum transverse and angular displacement of the wheelsets relative to each other. Trends between the input parameters were visually presented and their relationship with the comfort and stability factors were discussed.

## 1. Introduction

The main components of a single railway vehicle include a chassis and a set of wheelsets attached to a bogie. In this analysis we will only be considering a single bogie with a front and back wheelset.

Railway tracks are not consistently straight nor perfectly parallel to each other. This causes the components of a railway vehicle to displace vertically (bounce) and transversely (lurch), as well as rotate about its x-axis (roll), y-axis (pitch) and z-axis (yaw). The front wheelset, back wheelset and bogie have their own axis of freedom which they displace and rotate about, and how these axes interact relative to one another depends on the stiffnesses of the mechanism coupling them.

The purpose of this paper is to analyse the relative linear and rotational motion between the wheelsets and bogie to produce an optimum set of parameters that maximises the vehicles ride comfort and stability. This was done using an intuitive MATLAB script that simulates the journey of a wheelset and bogie travelling down a straight but poor-quality length of track. The script produces data describing the linear and rotational coordinates of the bogie, and both front and back wheelsets relative to their origin, based on a set of variable input parameters. The input parameters varied in this analysis are outlined below in table 1 and describe the properties of the vehicle under analysis. It should be noted that the moments of inertia of the wheelset and bogie about their axis vary in this analysis due to the changing wheel diameter and wheel-base dimension, changing the geometry and subsequently. the mass of each. This process was automated by adding the MATLAB script found in appendix A to the simulation code.

| Parameter | Description | Units |
| --- | --- | --- |
| $k_{wheel}$ | Bogie to Wheel Longitudinal Stiffness | [N/m] |
| $k_{shaft}$ | Bogie to Wheelset Transverse Stiffness | [N/m] |
| $k_{brace}$ | Cross Bracing Stiffness | [N/m] |
| $I_{w,y}, I_{b,z}$ | Moments of inertia of the wheelset and bogie about the y and x axis respectively | [kg/m$^2$] |
| $D_0$ | Neutral diameter of the wheel | [m] |
| $c$ | Conicity of the wheel as a ratio | [dimensionless] |
| $L_{wb}$ | Wheelbase separation | [m] |

*Table 1. Input parameters varied throughout the analysis*

# 2. Methodology

The purpose of this analysis is to maximise the passenger's ride comfort and the stability of the rail vehicle. Linear and rotational displacements can result in sharp changes of accelerations (defined as jerk (3)) which is felt by the passenger traveling on the vehicle. It is extremely critical that these changes of accelerations are minimised. If they get too large, the passenger's muscles physically can't respond fast enough to counteract the force exerted by the new acceleration which greatly affects the ride comfort. Track irregularities cause the wheelset to laterally displace and yaw. If these motions are excessive, the train will become unstable and derail. The analysis approach taken was to derive two equations, one representing the passenger's ride comfort and the other, the stability of the vehicle. Each equation contained the parameters that affected the corresponding factor.

## 2.1. Comfort Factor

The comfort factor gives a value that represents how comfortable an experience the passenger had on the simulated journey. It is a function of the magnitude of the resultant rate of change of acceleration (or jerk) and the magnitude of the resultant acceleration. The MATLAB script in appendix B was created to calculate the second and third time derivatives of the bogies and both wheelsets' displacements to find the acceleration and jerk of each component. From the cell of results, the maximum value was extracted and normalised by dividing each with a reference value. The reference value for the acceleration was the standard comfort rate of $\alpha_{ref} = 0.1\ ms^2$ (3) and that for jerk was chosen to be $J_{ref} = 0.03\ ms^3$. It should be noted that most trains are designed for a jerk rate of $0.35ms^3$ (4), however this was unreasonably high for the present work.

The jerk and acceleration normalised ratios were combined on a ten to one weighing, giving the jerk factor ten times more influence on the overall comfort factor relative to the acceleration factor as jerk has a more significant effect on the passenger's ride experience (2). The combined factor is divided by 11 to give it a range between zero and one, with zero representing no forces or yank experienced by the passenger's and one representing unacceptable forces and yank, with yank being defined as the product of jerk and mass (units [N/s]) (5).

$$Comfort\ Factor = \frac{\left(\frac{10 \times \sqrt{(\dddot{x})^2 + (\dddot{y})^2}}{J_{ref}}\right) + \frac{\sqrt{(\ddot{x})^2 + (\ddot{y})^2}}{\alpha_{ref}}}{11} \qquad [1]$$

## 2.2. Stability Factor

The stability factor is a value that represents how stable the designed railway vehicle is. It is a function of the magnitude of the maximum transverse displacement of the front wheelset relative to the back wheelset at any time instance, and the maximum relative angular displacement of the two wheelsets. The MATLAB script in appendix C calculated both values and normalised them using corresponding referencse values. The reference value for the angular and transverse displacement was $\Delta\theta_{ref} = 4°$ and $\Delta y_{ref} = 20\ \mathrm{mm}$ respectively, with both taken from the limits within the simulation script. If the wheelsets angular or transverse displacement exceeded these values, an if statement stopped the analysis and returned a 'derailed' message.

$$Stability\ Factor = \left(\frac{\left(1 + \frac{\Delta y}{\Delta y_{ref}}\right) \times \left(1 + \frac{\Delta\theta}{\Delta\theta_{ref}}\right) - 1}{3}\right) \qquad [2]$$

The stability factor above returned a normalised value between zero and one representing how stable the bogie design is, with one being unacceptably unstable resulting in derailment and zero representing the most stable state with no oscillations.

### 2.3. Experimental Approach

Two vehicles were analysed in parallel to each other throughout the investigation. The only difference between these models was the bogie width, bogie height and shaft length which was kept constant throughout. The first model was based on a light, smaller theoretical bogie obtained from a research paper (1) and the second model was based off the heavier, wider WINK train produced by the Swedish manufacturer Stadler Rail (2). Modelling trains of two different masses showed how the mass altered the relationship the parameters had on the comfort and stability factors.

The experiment took an iterative approach, consisting of 3 iterations with the solution more resolved each time by reducing the range and increasing the number of data points. For the first iteration, the input variables in table 1 were given initial values based off trains in service. The analysis method involved changing a single parameter to isolate their effect on the stability and comfort factors. A range of values for this parameter was processed through the simulation to give a corresponding range of comfort and stability factor values. The range of values was plotted against the two factors to visually represent the data. Once analysis of one parameter was completed, the varied parameter was reverted to its initial value and the process was repeated for the remaining variables.

The optimum values resulting from the first iteration was chosen as base parameters for the next iteration. These values were taken from the plots that gave the lowest comfort and stability factors. The process above was repeated until ultimately finishing with a set of optimum parameters giving the lowest comfort and stability factors.

## 3. Results and Discussion

The plotted results discussed in this section follow a pattern. The parameter under investigation was plotted against its corresponding comfort and stability factors calculated from the MATLAB script in appendix B and C using equations [1] and [2]. The plot on the left displays the results from the first iteration using estimated base parameters. The plot on the right uses more optimised parameters obtained from the previous iteration. The blue and red lines represent the comfort and stability factors respectively. The dashed lines represent the lighter, smaller theoretical bogie and the solid line represents the larger, heavier bogie. The final set of optimum parameters that resulted from this analysis were found by choosing the corresponding value to the parameter from the second iteration plot that gave the lowest stability and comfort factors. The first iteration's purpose was to identify areas of interest whilst the second iteration generated a more resolved solution within interesting areas by plotting more points in a smaller range. Table 2 displays the initial base parameters for each iteration along with the final optimised values.

| Parameter | 1st Iteration Initial Value | 2nd Iteration Initial Value | Final Optimised Values | Units |
|---|---|---|---|---|
| $k_{wheel}$ | 1000 | 400 | **250** | [kN/m] |
| $k_{shaft}$ | 3000 | 10000 | **100** | [kN/m] |
| $k_{brace}$ | 100 | 5000 | **3160** | [kN/m] |
| $D_0$ | 1.00 | 1.70 | **1.40 or 1.7** | [m] |
| $c$ | 0.10 | 0.06 | **1.20** | [no units] |
| $L_{wb}$ | 2.70 | 2.25 | **2.40** | [m] |

*Table 2. Input parameters varied throughout the analysis*

## 3.1. Stiffnesses

### 3.1.1. Cross Bracing Stiffness

The cross bracing stiffness represents the stiffness of the springs between opposite cornered wheels. Figures 1 shows a plot of the stability and comfort factors calculated across a range of stiffness values using the first (left) and second (right) iteration base parameters found in table 2. It is evident that the smaller, lighter theoretical bogie is generally more unstable and uncomfortable than the heavier, larger bogie. In both scenarios, the stability suddenly increases and the stability deceases at the same value, but at a faster rate for the lighter bogie. This suggests that the larger the mass of the bogie, the lower the effect of the brace's stiffness on comfort and stability.

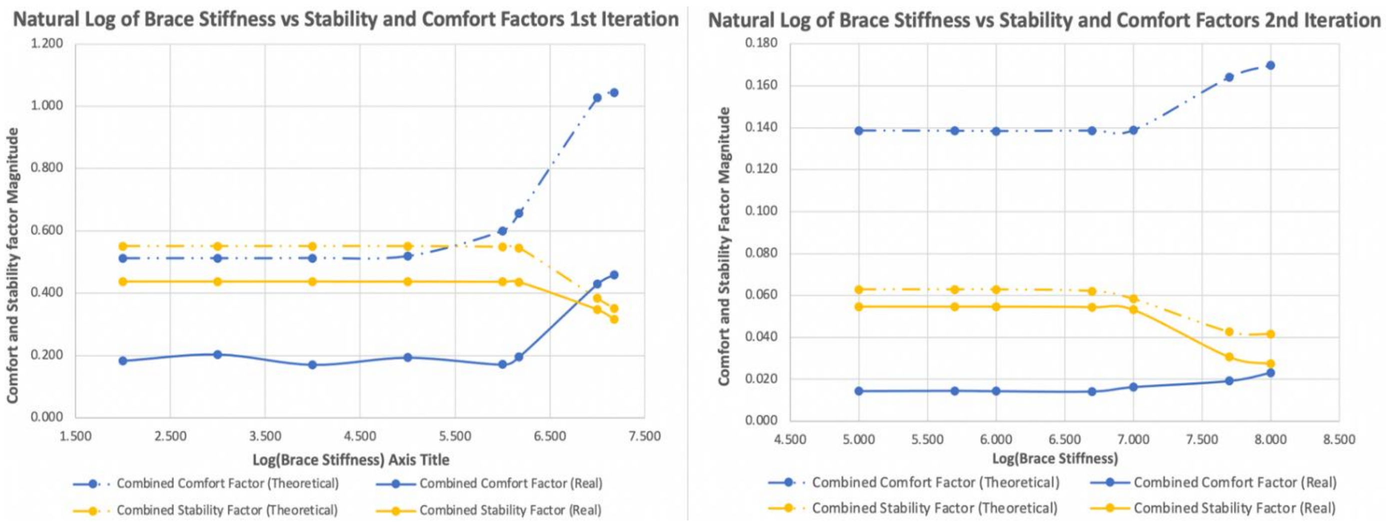


*Figure 1. A plot of factors against brace stiffnesses using the first (left) and second (right) iteration base parameters.*

### 3.1.2. Shaft Stiffness

The comfort and stability factors calculated from a range of shaft stiffness values using the first iteration base parameters on the left of figure 2 shows there is very little effect of the wheel stiffness on the comfort and stability factors outside of the range of 316 N/m to 31.6 kN/m The right plot in figure 2 is a more resolved solution using the second iteration base parameters that suggests keeping the stiffness below 10MN/m for a comfortable and stable ride. The shaft stiffness had no effect on the stability factor.

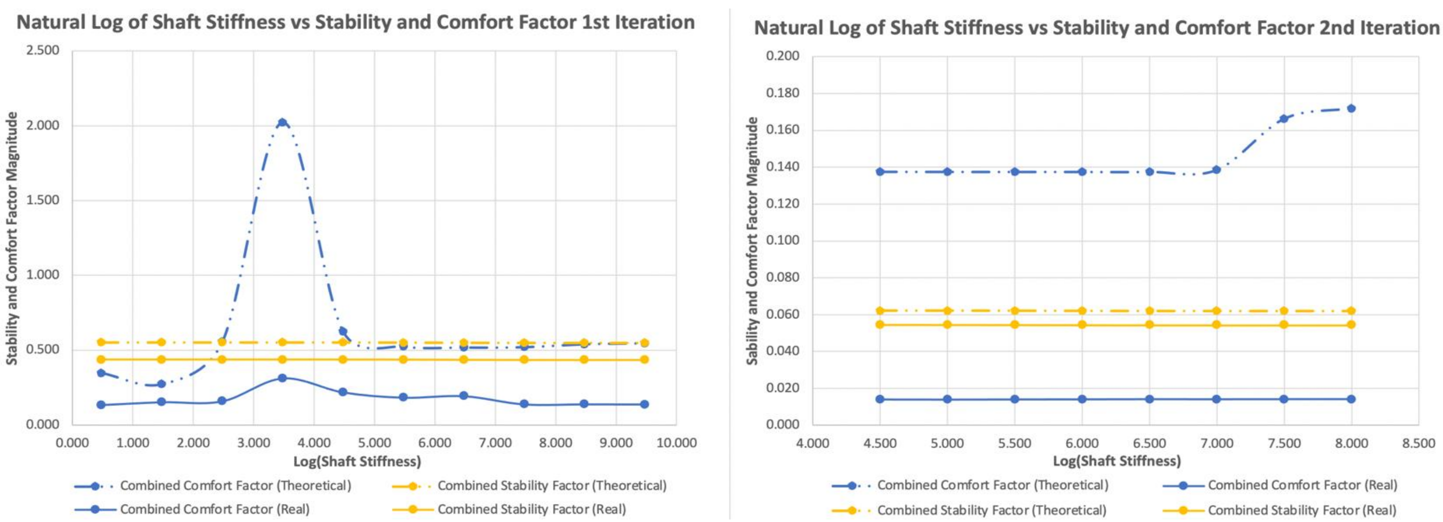


*Figure 2. A plot of factors against shaft stiffnesses using the first (left) and second (right) iteration base parameters.*

#### 3.1.3. Wheel Stiffness

For both sets of base parameters, figure 3 suggests the optimum wheel stiffness is around 500kN/m. The figure on the left was used to identify areas of interest while the right figure gives a more resolved solution, identifying 250 kN/m to be the optimum wheel stiffness. Similar to the shaft's stiffness, the stability factor is seen to be unaffected by the wheel stiffness.

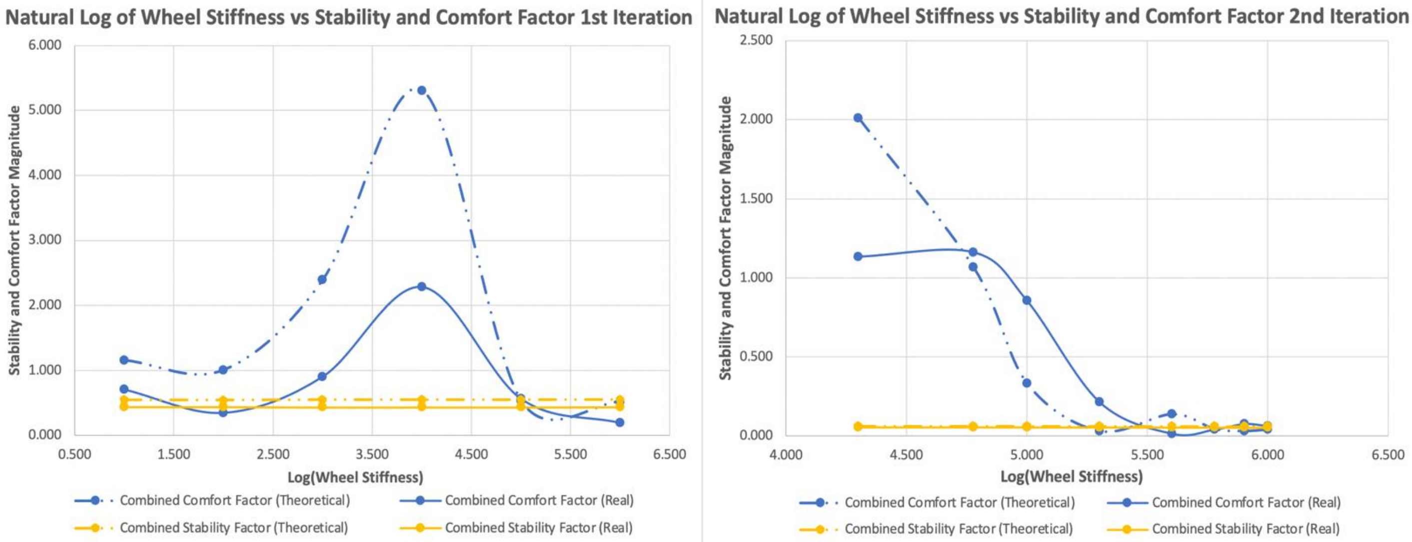


*Figure 3. A plot of factors against wheel stiffnesses using the first (left) and second (right) iteration base parameters.*

### 3.2. Wheel Diameter

The wheel diameter changed the mass of the wheelset and subsequently its moment of inertia. Figure 4 suggests that as the wheels diameter increases, so does the comfort and stability factor. The ideal scenario would be to have an infinitely large wheel, but this isn't practical. For this reason, a diameter of 1.7 m was chosen based off the SR Marchant Navy Class's 1.88m driven wheel (6).

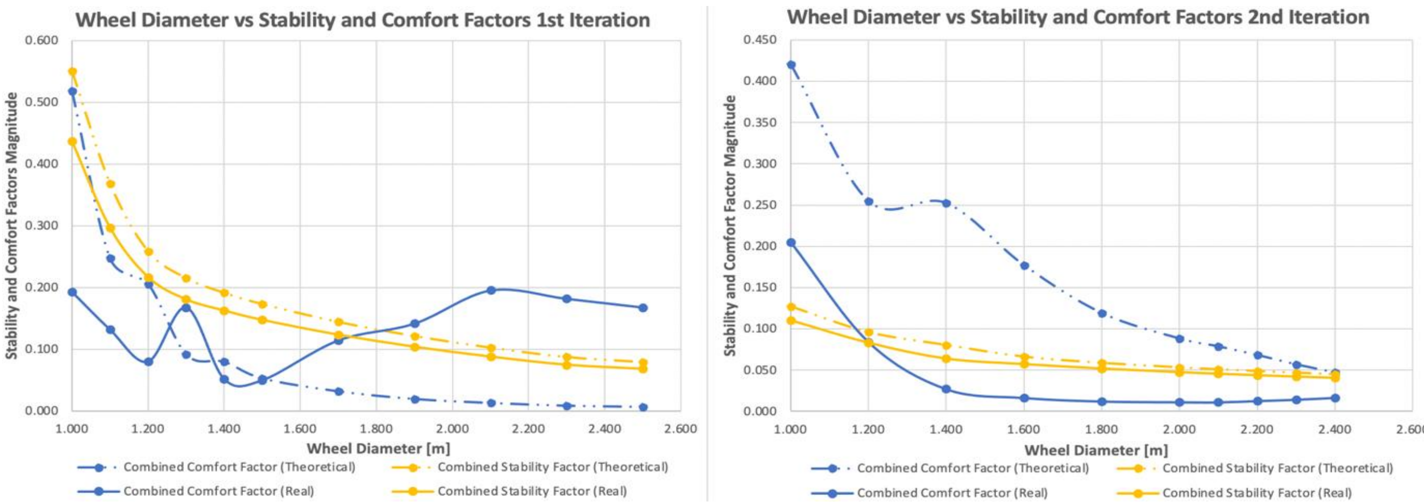


*Figure 4. A plot of factors against wheel diameter using the first (left) and second (right) iteration base parameters.*

### 3.3. Wheelbase

The wheelbase had to be larger than the wheel diameter to prevent the wheels from colliding. For this reason, dimensions below 1.6 m were not plotted for the second iteration as the wheel diameter was set at 1.7 m. It is clear from both plots in figure 5 that as the wheelbase increased, the ride experience got more comfortable but less stable. After the optimum wheel diameter was set, the lower boundary for the wheelbase was defined, and an optimum value of 1.6m was chosen.

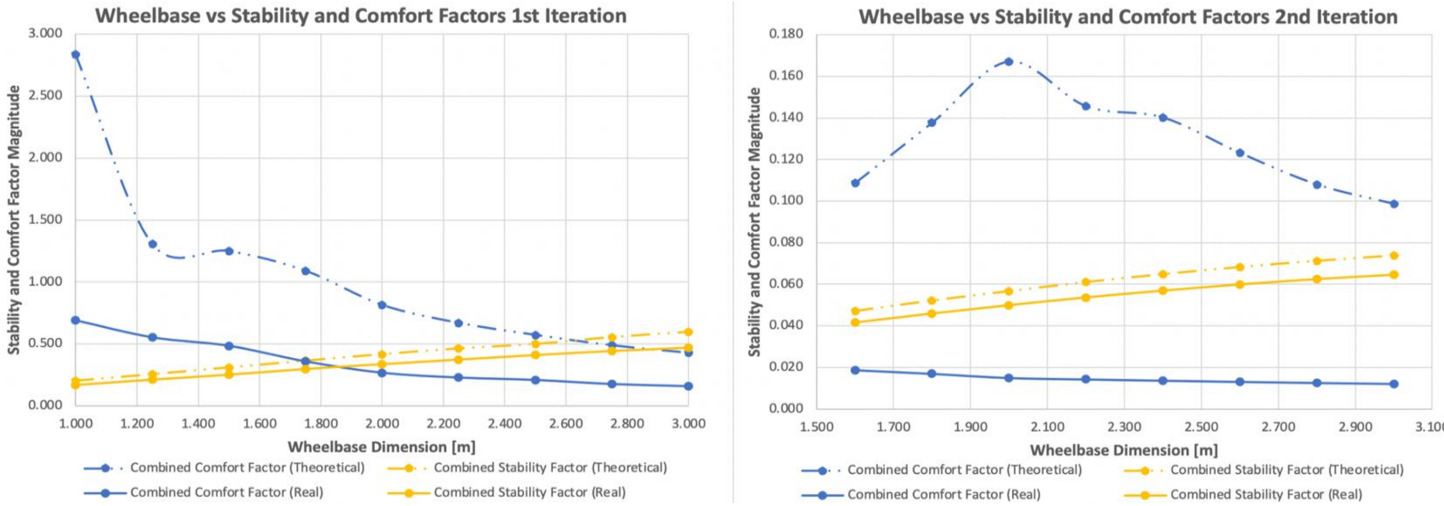


*Figure 5. A plot of factors against wheelbase dimensions using the first (left) and second (right) iteration base parameters.*

### 3.4. Conicity

Figure X represents the huge effect conicity has on the comfort and stability of the railway vehicle. The ideal value of conicity would be zero (a perfectly round wheel) but this isn't practical as the train would derail at the slightest corner. The conicity significantly affects the comfort factor after a taper of 0.15 whereas the stability factor is affected from the onset. A conical taper value of 1.2 was chosen to be the optimum, allowing tight cornering without sacrificing stability and comfort.

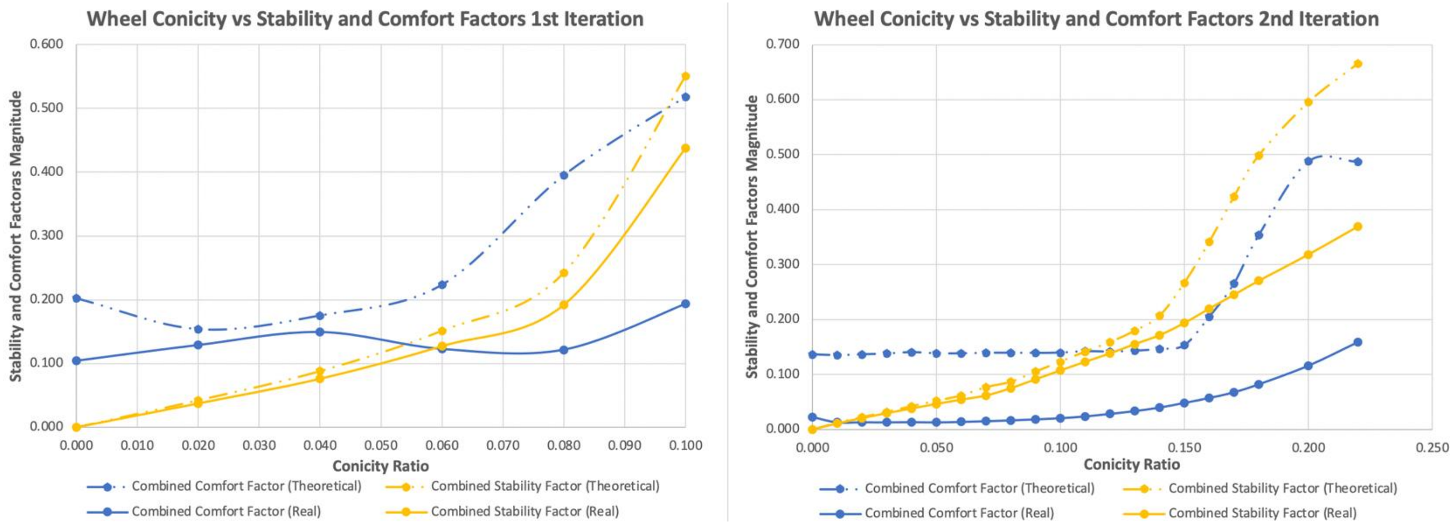


*Figure 6. A plot of factors against conicity using the first (left) and second (right) iteration base parameters.*

# 4. Conclusion

A MATLAB script was used to simulate the journey of a railway vehicle travelling down a length of track. The spatial coordinates of the bogie and wheelsets relative to the origin were processed to describe the comfort and stability of the journey using two normalised factors between zero and one. For the lighter and smaller theoretical bogie, the comfort of the ride was increased by 3942% and its stability increased by 2678%. The heavier and larger real bogie experienced a ride comfort increase of 278% and a stability increase of 876% (appendix E). The optimised set of parameters that result in the most comfortable and stable ride for the given masses can be found in table 2. If the comfort and stability of the vehicle was to be increased further, additional iterations following this method would be suggested.

# 5. References

1. Stadler Wink Zero Emission Rail Vehicle [Internet]. Stadlerrail.com. 2022 [cited 5 May 2022]. Available from: https://www.stadlerrail.com/media/pdf/warr0517e_print.pdf

2. Sharma R. Ride Analysis of an Indian Railway Coach using Lagrangian Dynamics. International Journal of Vehicle Structures and Systems. 2011;3.

3. Jiang Y, Chen B, Thompson C. A comparison study of ride comfort indices between Sperling's method and EN 12299. International Journal of Rail Transportation. 2019;7(4):279-296.

4. Sharma S, Chaturvedi S. Jerk analysis in rail vehicle dynamics. Perspectives in Science. 2016;8:648-650.

5. Yank (physics) [Internet]. Academic Dictionaries and Encyclopedias. 2022. Available from: https://en-academic.com/dic.nsf/enwiki/1620873

6. SR Merchant Navy class - Wikipedia [Internet]. En.wikipedia.org. 2022. Available from: https://en.wikipedia.org/wiki/SR_Merchant_Navy_class

# 6. Appendix

## 6.1. Appendix A

MATLAB script to calculate the **wheelsets mass and moment of inertia** about the y axis as a function of wheel neutral diameter and the bogies mass and moment of inertia about the z axis as a function of wheelbase distance.

```
% ======= Variables to change ==============

wheel_base          = 2.25;
wheel_diameter      = 1.7;
Brace_stiffness     = 5e6;
Wheel_stiffness     = 4e5;
Shaft_stiffness     = 1e7;
conicity            = 0.6;

% ==========================================

% ----- Moments of Inertia ------

% ---- Moment of inetria bogie (about z axis) -----

density = 7000;
bogie_length = wheel_base;
bogie_mass = density * bogie_width * bogie_length * bogie_thickness;
bogie_I_z = (1/12) * (bogie_mass) * (bogie_length ^2) * (bogie_width^2);

% ---- Moment of inertia Wheelset (about y axis) -----

density = 8000;
wheel_thickness = 0.2; % [m]
shaft_diameter = 0.2; % [m]
shaft_length = 1.435; % [m]
wheel_volume = wheel_thickness * pi * ((wheel_diameter)^2)/4;
wheel_mass = 8000 * wheel_volume;
shaft_volume = shaft_length * pi * ((shaft_diameter)^2)/4;
shaft_mass = 8000 * shaft_volume;
wheelset_mass = 2*wheel_mass + shaft_mass;

% ===== Wheel moment of inertia ======

d_o = wheel_diameter ;
d_i = shaft_diameter;
wheel_I_y = ((density * wheel_thickness * pi )/32) * ((d_o^2) - (d_i^2)) *
((d_o^2)+(d_i^2));

% ===== Shaft moment of inertia  ======

shaft_I_y = ((density * pi * shaft_length * (shaft_diameter)^4 ) / 32 );

% ===== Wheelset moment of inertia  ======

wheelset_I_y = shaft_I_y + 2 * wheel_I_y;
```

### 6.2. Appendix B

MATLAB script to calculate the **comfort factor**, the forces exerted on a passenger due to the accelerations and the rate of change of force exerted on a passenger due to yank.

```
% Calculating the instantaneous Velocities, Accelerations
% and the rate of change of acceleration of the bogie

% ------- COMFORT FACTOR --------

m_passanger = 80; % average mass of a passanger [kg]
Time = To;

% --------- Longitudinal X direction -----------

Time = To;
x = S(:,9);  % x = the bogies displacement from the origin after each timestep
v_x = diff(x)./diff(Time);  % v = instantaneous velocities at each timestep
Time = Time(1:120);
a_x = diff(v_x)./diff(Time);  % a = instantaneous accelerations at each timestep
Time = Time(1:119);
jerk_x = diff(a_x)./diff(Time);  % instantaneous jerking accelerations

% --------- Transverse Y Direction -----------

Time = To;
y = S(:,10);  % y = the bogies displacement from the origin after each timestep
v_y = diff(y)./diff(Time);  % v = instantaneous velocities at each timestep
Time = Time(1:120);
a_y = diff(v_y)./diff(Time);  % a = instantaneous accelerations at each timestep
Time = Time(1:119);
jerk_y = diff(a_y)./diff(Time);  % a_dot instantaneous jerking accelerations

% --------- Overall Velocity at each displacement (relative to initial velocity) -
----------

omega_wheelsets = (S(:,4) + S(:,8)) / 2; % average angular velocity of wheelsets
v_x_overall = omega_wheelsets * T.D0/2 ; % overall bogie velocities at each time
step
v_x_final = v_x_overall(121,1); % Final Bogie Velocity

% ------- Maximum accelerations ----------

a_x_max = max(abs(a_x)); % maximum longitudinal acceleration
a_y_max = max(abs(a_y)); % maximum transverse acceleration
a_max_resultant = sqrt((a_x_max)^2 + (a_y_max)^2) ; % maximum resultant
acceleration

% ------- Maximum change in accelerations (jerk) ----------

a_max_jerk_x = max(abs(jerk_x)); % maximum change in longitudinal acceleration
a_max_jerk_y = max(abs(jerk_y)); % maximum change in transverse acceleration
a_max_jerk_resultant = sqrt((a_max_jerk_x)^2 + (a_max_jerk_y)^2); % maximum
resultant jerk

% ----- Forces exerted on the passanger due to accelerations --------
```

```
Force_x = m_passanger * a_x_max; % max longitudinal force on passanger due to
acceleration
Force_y = m_passanger * a_y_max; % max transverse force on passanger due to
acceleration
Force_acc_resultant = m_passanger * a_max_resultant; % max resultant force on
passanger due to acceleration

% ------ Yank forces exerted on the passanger due to jerking -------

Force_jerk_x = m_passanger * a_max_jerk_x; % change in force due to change in
longitudinal acceleration
Force_jerk_y = m_passanger * a_max_jerk_y; % change in force due to change in
transverse acceleration
Force_jerk_resultant = m_passanger * a_max_jerk_resultant; % maximum OVERALL force
due to jerking!!

% ----------  Comfort Factor -------

% ---- Jerking factor ----- dividing by reference after sqrt MAYBE USE THIS ONE

jerk_ref = 0.02; % maximum jerk allowable = 0.35 m/s/s/s taken from design TTS
standards
%a_max_jerk_resultant = 0.038; %TESTING
%a_max_resultant = 0.0; % TESTING

comfort_factor_jerking = a_max_jerk_resultant / jerk_ref;
%comfort_factor_jerking = ( comfort_factor_jerking / sqrt(2) ) * 1;


% ----- acceleration factor -----

acc_ref = 0.1; % Standard ride comfort acceleration
comfort_factor_acc = a_max_resultant / acc_ref; % normalised comfort factor
(acceleration contribution)

% ------ Combining comfort factors -------

combined_CF = ( ( 10 * comfort_factor_jerking) + (comfort_factor_acc) ) / 11 ;

% ------ Instantaneous forces ----- F = ma

Force_y = m_passanger * a_y;
Force_x = m_passanger * a_x;
Force_resultant = sqrt((Force_x).^2+(Force_y).^2);
max_force = max(Force_resultant); % maximum force a body will exprience due to
accelerations
max_force_theoretical = m_passanger * a_max_resultant; % maximum possible force
experienced using max accelerations THEORETICAL MAX VALUE

% ------- Instantaneous Yank forces [F/s] -------

Yank_x = m_passanger * jerk_x ;
Yank_y = m_passanger * jerk_y ;
Yank_resultant = sqrt((Yank_x).^2 + (Yank_y).^2);
max_yank = max(Yank_resultant); % maximum instantaneous yanking force/s TRUE MAX
VALUE
max_yank_theoretical = m_passanger * a_max_jerk_resultant;
```

```
% ---- Printing Values -------

fprintf ('Comfort Factor Jerking 2 (out of 1) = %.3f \n', comfort_factor_jerking);
fprintf ('Comfort Factor Acceleration 2 (out of 1) = %.3f \n',
comfort_factor_acc);
fprintf ('Combined Comfort Factor (out of 1) = %.3f \n', combined_CF);
fprintf ('Maximum Acceleration Force Experiences = %.3f [N] \n', max_force);  %
passanger experiences a maximum force of this
fprintf ('Maximum Yank Experienced = %.3f [N/s] \n', max_yank); % passanger
experiences a rate of force of this (maximum change in force wrt time)

% ==========================================
```

### 6.3. Appendix C

MATLAB script used to calculate and normalise the **stability factor** between one and zero.

```
% ==========================================

% -------  Stability Factor ----

% ------ difference in FW and BW transverse displacements ------

disp_y_FW = S(:,2); % transverse displacements of the FW
disp_y_BW = S(:,6); % transverse displacements of the BW
delta_y = abs(disp_y_FW - disp_y_BW); % Displacement of the FW relative to the BW
max_delta_y = max(delta_y); % Maximum displacement of the FW relative to the BW
(must be below 20 mm to avoid detailing)

% ------ difference in FW and BW angular displacements ------

theta_FW = (S(:,3)/(2*pi))*360; % FW angular displacement in degrees
theta_BW = (S(:,7)/(2*pi))*360; % BW angular Displacement in degrees
delta_theta = theta_FW - theta_BW;
max_delta_theta = max(abs(delta_theta));


disp_y_ave_FW = mean(disp_y_FW);
disp_y_ave_BW = mean(disp_y_BW);

% ------- Stability Factor ------

% ------- Normalising delta y and delta theta ------

delta_y_ref = 0.02; % reference delta y = 20 mm before derailing
SF_disp = max_delta_y / delta_y_ref;
delta_theta_ref = 4; % reference delta theta = 4 degrees before too unstable
SF_theta = max_delta_theta / delta_theta_ref;

% --- Combining ----

stability_factor = ( ( ( ( ( 1 + SF_disp) * (1 + SF_theta) ) ) -1 ) /3);

% ----- Printing Important Values ----

fprintf ('Maximum Transverse displacement of FW relative to BW = %.3f [mm] \n',
max_delta_y*1000);
fprintf ('Maximum angular displacement of FW relative to BW = %.3f [degrees] \n',
max_delta_theta);
fprintf ('Stabiltiy factor (between 0 and 1) = %.3f \n', stability_factor);
fprintf(1, '\n');
T_real_values = {comfort_factor_jerking,
comfort_factor_acc,max_yank,max_force,combined_CF, max_delta_y*1000
,max_delta_theta,stability_factor,SF_disp,SF_theta};
```

Author: David Stewart
Supervisor: Dr. Francis Franklin
MEC8024

## 6.4. Appendix D

Example spreadsheet used to analyse the dataset after processing in MATLAB

| Initial Values | |
|---|---|
| K_brace | 1.00E+05 |
| K_wheel | 1.00E+06 |
| K_shaft | 3.00E+06 |
| Wheelbase diameter | 2.70E+00 |
| wheel diameter | 1.00E+00 |
| conicity | 1.00E-01 |

VARYING K_BRACE Theoretical

| K_brace | Jerk Comfort Factor | Acceleration Comfo | Maximum Yank (T) | Maximum Acceleration | Combined Comfort ( | Maximum Transcerse | Maximum Displaced | Combined Stabili | Displacement Stabilit | Angular Stability Factor (T) | Log of Brace |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1.00E+02 | 0.509 | 0.534 | 0.787 | 4.225 | 0.511 | 19.675 | 0.674 | 0.551 | 0.984 | 0.337 | 2.000 |
| 1.00E+03 | 0.509 | 0.534 | 0.787 | 4.225 | 0.512 | 19.675 | 0.674 | 0.551 | 0.984 | 0.337 | 3.000 |
| 1.00E+04 | 0.510 | 0.534 | 0.787 | 4.225 | 0.512 | 19.675 | 0.674 | 0.551 | 0.984 | 0.337 | 4.000 |
| 1.00E+05 | 0.517 | 0.534 | 0.791 | 4.225 | 0.518 | 19.672 | 0.674 | 0.551 | 0.984 | 0.337 | 5.000 |
| 1.00E+06 | 0.606 | 0.534 | 0.834 | 4.225 | 0.600 | 19.592 | 0.672 | 0.548 | 0.980 | 0.336 | 6.000 |
| 1.50E+06 | 0.668 | 0.534 | 0.870 | 4.225 | 0.656 | 19.473 | 0.669 | 0.545 | 0.974 | 0.335 | 6.176 |
| 1.00E+07 | 1.076 | 0.535 | 1.641 | 4.225 | 1.027 | 14.119 | 0.521 | 0.383 | 0.706 | 0.261 | 7.000 |
| 1.50E+07 | 1.095 | 0.534 | 1.689 | 4.227 | 1.044 | 13.086 | 0.485 | 0.352 | 0.654 | 0.242 | 7.176 |

Ideal value for k brace
5.00E+06

Varying K_Brace

VARYING K_BRACE Real

| K_brace | Jerk Comfort Factor | Acceleration Comfo | Maximum Yank ® | Maximum Acceleration | Combined Comfort ( | Maximum Transcerse | Maximum Displaced | Combined Stabili | Displacement Stabilit | Angular Stability Factor (R) | Log of Brace |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1.00E+02 | 0.178 | 0.239 | 0.247 | 1.890 | 0.183 | 19.616 | 0.669 | 0.437 | 0.981 | 0.167 | 2.000 |
| 1.00E+03 | 0.200 | 0.239 | 0.293 | 1.890 | 0.204 | 19.616 | 0.669 | 0.437 | 0.981 | 0.167 | 3.000 |
| 1.00E+04 | 0.164 | 0.239 | 0.251 | 1.890 | 0.170 | 19.616 | 0.669 | 0.437 | 0.981 | 0.167 | 4.000 |
| 1.00E+05 | 0.189 | 0.239 | 0.271 | 1.890 | 0.193 | 19.614 | 0.669 | 0.437 | 0.981 | 0.167 | 5.000 |
| 1.00E+06 | 0.164 | 0.239 | 0.260 | 1.890 | 0.171 | 19.580 | 0.668 | 0.437 | 0.979 | 0.167 | 6.000 |
| 1.50E+06 | 0.192 | 0.239 | 0.304 | 1.890 | 0.196 | 19.539 | 0.667 | 0.436 | 0.977 | 0.167 | 6.176 |
| 1.00E+07 | 0.449 | 0.238 | 0.714 | 1.897 | 0.430 | 15.842 | 0.556 | 0.347 | 0.792 | 0.139 | 7.000 |
| 1.50E+07 | 0.482 | 0.238 | 0.756 | 1.890 | 0.460 | 14.459 | 0.524 | 0.316 | 0.723 | 0.131 | 7.176 |

| Initial Values | |
|---|---|
| K_brace | 1.00E+05 |
| K_wheel | 1.00E+06 |
| K_shaft | 3.00E+06 |
| Wheelbase diameter | 2.70E+00 |
| wheel diameter | 1.00E+00 |
| conicity | 1.00E-01 |

VARYING K_Wheel Theoretical

| K_Wheel | Jerk Comfort Factor | Acceleration Comfo | Maximum Yank (T) | Maximum Acceleration | Combined Comfort ( | Maximum Transcerse | Maximum Displaced | Combined Stabili | Displacement Stabilit | Angular Stability Factor (T) | Log of Brace |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1.00E+00 | 92.167 | 10.915 | 147.467 | 85.637 | 84.781 | 19.607 | 0.680 | 0.551 | 0.980 | 0.340 | 0.000 |
| 1.00E+01 | 1.188 | 0.842 | 1.869 | 6.624 | 1.156 | 19.599 | 0.666 | 0.546 | 0.980 | 0.333 | 1.000 |
| 1.00E+02 | 1.015 | 0.931 | 1.240 | 7.397 | 1.007 | 19.504 | 0.665 | 0.544 | 0.975 | 0.332 | 2.000 |
| 1.00E+03 | 2.536 | 1.001 | 3.985 | 7.977 | 2.396 | 19.621 | 0.671 | 0.549 | 0.981 | 0.336 | 3.000 |
| 1.00E+04 | 5.747 | 0.934 | 9.163 | 7.444 | 5.310 | 19.605 | 0.671 | 0.548 | 0.980 | 0.336 | 4.000 |
| 1.00E+05 | 0.534 | 0.538 | 0.771 | 4.257 | 0.534 | 19.604 | 0.671 | 0.548 | 0.980 | 0.336 | 5.000 |
| 1.00E+06 | 0.517 | 0.534 | 0.791 | 4.225 | 0.518 | 19.672 | 0.674 | 0.551 | 0.984 | 0.337 | 6.000 |
| 1.00E+07 | FAIL | | | | | | | | | | 7.000 |

Ideal value for k wheel
4.00E+05
or 100.000

Varying K_Wheel

## 6.5. Appendix E

Percentage increase of the optimised comfort and stability factors relative to the initial estimated for the theoretical and real bogies respectively.

| | Theoretical Bogie | Real Bogie |
|---|---|---|
| Initial **Comfort** Factor | 0.552 | 0.150 |
| Final **Comfort** Factor | 0.014 | 0.054 |
| (Initial/Final) * 100 | **3942%** | **273%** |

| | Theoretical Bogie | Real Bogie |
|---|---|---|
| Initial **Stability** Factor | 0.750 | 0.473 |
| Final **Stability** Factor | 0.028 | 0.054 |
| (Initial/Final) * 100 | **2678%** | **876%** |