Simulation Data Analysis

Abhinav Bapanapalli

2/2/2022

Summary of Analysis

After running the nonlinear simulations using the given CAD model and material data, nodal data was extracted at the end of each simulation. Two files were extracted from each of these simulations, one of which represent the data from the outer shell of the tire, and the other which represents the sidewall rubber insert of the tire. This R Program aims to clean this data, and draw conclusions from it using various statistical tests.

The primary statistical test to be used in this program is the **Kruskal-Wallis Test**, which can test to see if the distribution of various samples of data are similar. This can help us understand if the distribution in stress, displacement, or strain differs between various concentrations of graphene, which can then help us answer questions about the efficacy of graphene in run-flat tires.

Summary of Data

The data being examined is comprised of 30 different CSV Files, each of which contain either the **strain**, **stress**, or **displacement** data of the **tire** or the **sidewall insert** of five different tires. The five tires being tested were of the same shape (as described by the testing procedure), but differed in material composition, specifically graphene concentration. The concentration of graphene in these tires were **0.0 PHR**, **0.1 PHR**, **0.5 PHR**, **1.0 PHR**, and **2.0 PHR** respectively.

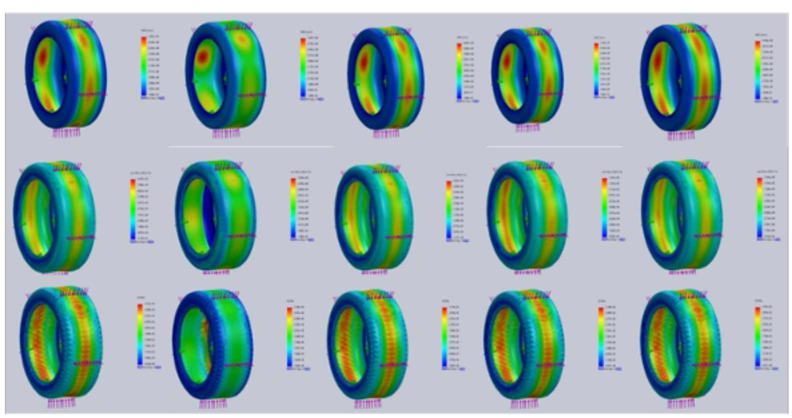


Figure 1: Simulated Tire Diagrams

Hypotheses

Displacement

$$H_0 : \tilde{x}_{0.0} = \tilde{x}_{0.1} = \tilde{x}_{0.5} = \tilde{x}_{1.0} = \tilde{x}_{2.0}$$

The displacement data between the five tires has an identical distribution (medians are equal)

$$H_1 ; \tilde{x}_{0.0} \neq \tilde{x}_{0.1} \neq \tilde{x}_{0.5} \neq \tilde{x}_{1.0} \neq \tilde{x}_{2.0}$$

The displacement data between the five tires varies (medians are not equal)

Stress

$$H_0 : \tilde{x}_{0.0} = \tilde{x}_{0.1} = \tilde{x}_{0.5} = \tilde{x}_{1.0} = \tilde{x}_{2.0}$$

The stress data between the five tires has an identical distribution (medians are equal)

$$H_1 ; \tilde{x}_{0.0} \neq \tilde{x}_{0.1} \neq \tilde{x}_{0.5} \neq \tilde{x}_{1.0} \neq \tilde{x}_{2.0}$$

The stress data between the five tires varies (medians are not equal)

Strain

$$H_0 : \tilde{x}_{0.0} = \tilde{x}_{0.1} = \tilde{x}_{0.5} = \tilde{x}_{1.0} = \tilde{x}_{2.0}$$

The strain data between the five tires has an identical distribution (medians are equal)

$$H_1 ; \tilde{x}_{0.0} \neq \tilde{x}_{0.1} \neq \tilde{x}_{0.5} \neq \tilde{x}_{1.0} \neq \tilde{x}_{2.0}$$

The strain data between the five tires varies (medians are not equal)

Cleaning the Data

The 30 CSV files were consolidated into 6 data frames, each of which contained stress, strain, or displacement data for the five tires or sidewall inserts respectively. These new data frames were comprised of two values; one being the value of stress, strain, ord displacement gathered from the simulation, and the other being the tire data it was extracted from.

Statistical summary of data

After having consolidated the data frames, we can find the summary statistics of the data to briefly understand what it is going to show.

```
summary_tdis
```

```
## # A tibble: 5 x 6
##   Components  mean    q1 median    q3   max
##   <chr>      <dbl> <dbl>  <dbl> <dbl> <dbl>
## 1 GE_0        2.76 0.876   1.27  4.73 10.2
## 2 GE_0.1      2.08 0.908   1.41  3.27  7.36
## 3 GE_0.5      2.67 0.850   1.22  4.60  9.75
## 4 GE_1        2.59 0.826   1.18  4.56  9.03
## 5 GE_2        2.81 0.885   1.28  4.82 10.2
```

```
summary_sdis
```

```
## # A tibble: 5 x 6
##   Components  mean    q1 median    q3   max
##   <chr>      <dbl> <dbl>  <dbl> <dbl> <dbl>
## 1 GE_0        1.41 0.898   1.33  1.63  3.12
## 2 GE_0.1      1.44 0.769   1.36  1.92  3.84
## 3 GE_0.5      1.37 0.866   1.29  1.58  2.99
## 4 GE_1        1.32 0.837   1.25  1.53  2.81
## 5 GE_2        1.43 0.906   1.34  1.65  3.15
```

```
summary_tstrain
```

```
## # A tibble: 5 x 6
##   Components    mean      q1  median     q3    max
##   <chr>        <dbl>   <dbl>   <dbl>  <dbl>  <dbl>
## 1 GE_0       0.0123  0.00676 0.00964 0.0154 0.0326
## 2 GE_0.1     0.00748 0.00301 0.00523 0.0119 0.0198
## 3 GE_0.5     0.0118  0.00652 0.00930 0.0148 0.0313
## 4 GE_1       0.0115  0.00633 0.00904 0.0144 0.0297
## 5 GE_2       0.0125  0.00685 0.00977 0.0157 0.0329
```

```
summary_sstrain
```

```
## # A tibble: 5 x 6
##   Components   mean      q1 median     q3    max
##   <chr>       <dbl>   <dbl>  <dbl>  <dbl>  <dbl>
## 1 GE_0       0.0115 0.00674 0.0123 0.0156 0.0224
## 2 GE_0.1     0.0157 0.00913 0.0143 0.0237 0.0336
## 3 GE_0.5     0.0111 0.00647 0.0120 0.0151 0.0214
## 4 GE_1       0.0108 0.00627 0.0116 0.0148 0.0203
## 5 GE_2       0.0117 0.00683 0.0125 0.0159 0.0228
```

```
summary_tstress
```

```
## # A tibble: 5 x 6
##   Components   mean      q1 median     q3     max
##   <chr>       <dbl>   <dbl>  <dbl>  <dbl>   <dbl>
## 1 GE_0        2753.  1614.   2357   3215.   8267
## 2 GE_0.1     22569. 13878.  18100  31292.  70500
## 3 GE_0.5       143.    84.1   123.   167.    426.
## 4 GE_1       49347. 28780   42160  57592. 139100
## 5 GE_2       57553. 33532.  49060  67360  170300
```

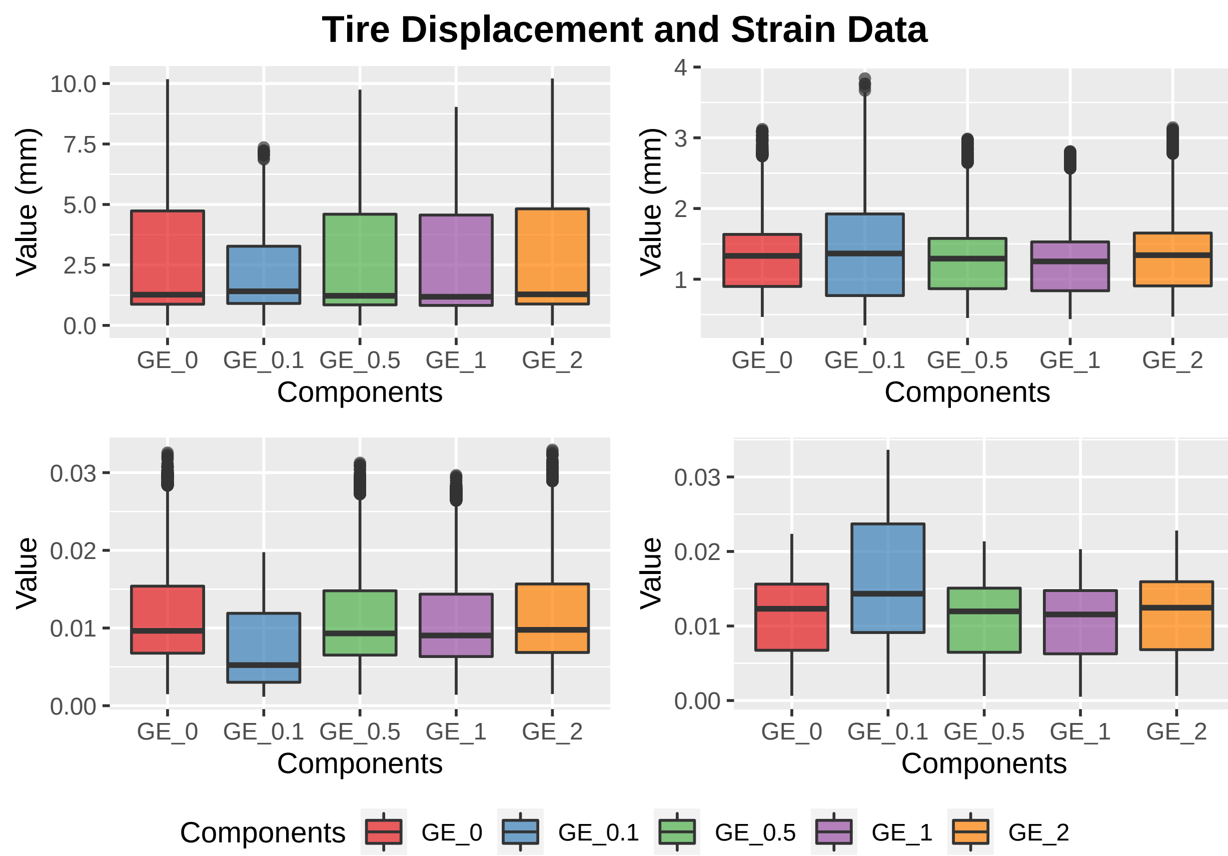
```
summary_sstress
```

```
## # A tibble: 5 x 6
##   Components   mean      q1 median     q3     max
##   <chr>       <dbl>   <dbl>  <dbl>  <dbl>   <dbl>
## 1 GE_0        2764.  1150    3094.  3801.   5733
## 2 GE_0.1       192.    74.7   179.   284.    520.
## 3 GE_0.5       144.    59.3   162.   201.    295.
## 4 GE_1       49765. 20075   55725  70070   98560
## 5 GE_2       57718. 23802.  64355  79632. 120000
```

Plotting the Summary Data

After the summary statistcs have been pulled, box and whisker plots can be used to show the distribution of each of the samples in the data frames.

```
p1 <- GE_tire_displacement %>%
  ggplot(aes(x = Components, y = `Value (mm)`, fill = Components)) +
  geom_boxplot(alpha = 0.7) +
  theme(legend.position="none") +
  scale_fill_brewer(palette="Set1")
p2 <- GE_sidewall_displacement %>%
  ggplot(aes(x = Components, y = `Value (mm)`, fill = Components)) +
  geom_boxplot(alpha = 0.7) +
  theme(legend.position="none") +
  scale_fill_brewer(palette="Set1")
p3 <- GE_tire_strain %>%
  ggplot(aes(x = Components, y = `Value`, fill = Components)) +
  geom_boxplot(alpha = 0.7) +
  theme(legend.position="none") +
  scale_fill_brewer(palette="Set1")
p4 <- GE_sidewall_strain %>%
  ggplot(aes(x = Components, y = `Value`, fill = Components)) +
  geom_boxplot(alpha = 0.7) +
  theme(legend.position="none") +
  scale_fill_brewer(palette="Set1")
p5 <- GE_tire_stress%>%
  ggplot(aes(x = Components, y = `Value (N/m^2)`, fill = Components)) +
  geom_boxplot(alpha = 0.7) +
  theme(legend.position="none") +
  scale_fill_brewer(palette="Set1")
p6 <- GE_sidewall_stress %>%
  ggplot(aes(x = Components, y = `Value (N/m^2)`, fill = Components)) +
  geom_boxplot(alpha = 0.7) +
  theme(legend.position="none") +
  scale_fill_brewer(palette="Set1")
p <- ggpubr::ggarrange(p1, p2, p3, p4, common.legend = T, legend = "bottom", ncol = 2, nrow = 2)
annotate_figure(p,
               top = text_grob("Tire Displacement and Strain Data",
                               color = "black", face = "bold", size = 14))
```



From these box and whisker plots, we can see that the lowest and most compact distribution for the displacement and strain data of the tire comes from the **0.1 GE** tire, which is also the source of the highest and most spread out distribution of displacement and strain data for the sidewall inserts.

Kruskal Wallis Test

The Kruskal Wallis Test is a statistical test that is used for 2 or more groups of continuous independent variables, modeled by non-parametric functions. It determines whether the medians of the groups of data are the same or not, which thereby demonstrates if the distribution of the data is similar or varying between samples. For a Kruskal Wallis Test, the data does not have to follow normality, and sample sizes can be different between samples.

The formula for the Kruskal Wallis Test is as follows:

$$H = [\frac{12}{n(n+1)} \sum_{j=1}^{c} \frac{T_j^2}{n_j}] - 3(n+1)$$

Where:
n = total number of data points
c = number of samples/groups
j = sample/group number
T_j = sum of ranks for the j^{th} sample
n_j = number of data points in/sample size of j^{th} sample

For this data, we will be applying the Kruskal Wallis Test to the six different data frames to see if there is a variation in distribution between the five types of tires. If the distribution is different, we can then prove that by adding graphene to the tires, there is a significant change in displacement, stress, or strain data. This can then help prove the claim that graphene can help improve the durability and flexural strength of run-flat tires.

```r
kruskal.test(`Value (mm)` ~ Components, data = GE_tire_displacement)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Value (mm) by Components
## Kruskal-Wallis chi-squared = 43.062, df = 4, p-value = 1.004e-08
```

```r
kruskal.test(`Value (mm)` ~ Components, data = GE_sidewall_displacement)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Value (mm) by Components
## Kruskal-Wallis chi-squared = 67.45, df = 4, p-value = 7.837e-14
```

```r
kruskal.test(`Value (N/m^2)` ~ Components, data = GE_tire_stress)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Value (N/m^2) by Components
## Kruskal-Wallis chi-squared = 14768, df = 4, p-value < 2.2e-16
```

```r
kruskal.test(`Value (N/m^2)` ~ Components, data = GE_sidewall_stress)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Value (N/m^2) by Components
## Kruskal-Wallis chi-squared = 6789.9, df = 4, p-value < 2.2e-16
```

```r
kruskal.test(`Value` ~ Components, data = GE_tire_strain)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Value by Components
## Kruskal-Wallis chi-squared = 539.64, df = 4, p-value < 2.2e-16
```

```r
kruskal.test(`Value` ~ Components, data = GE_sidewall_strain)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Value by Components
## Kruskal-Wallis chi-squared = 142.77, df = 4, p-value < 2.2e-16
```

Conclusions from the Kruskal Wallis Test

As shown by the Kruskal Wallis Tests, all six data frames have significant data, which means that the distribution of data between each of the samples does indeed vary. Since there are more than two samples were are calculating variance from, it is necessary to run a post-hoc test to see where those variations lie.

Post-Hoc Test: Dunn Test

In order to discover where the differences in variation lie, a post-hoc test such as the **Dunn Test** can be used. This test compares the distributions of each of the samples against each other, and generates separate p values for each. The insignificant p values generated can tell us where similar data lies. Although this test is generally considered conservative, it can help develop a deeper understanding of where the discrepancies in the data lie. For a more accurate post-hoc test, a Mann-U Whitney test can be conducted.

After we gather results from this test, we will only examine the **GE_0** column since that is our control group. From there, we can see in which of the tests the tire iterations are similar to the control groups.

```
dunn.test(GE_tire_displacement$`Value (mm)`, GE_tire_displacement$Components)
```

```
##   Kruskal-Wallis rank sum test
##
## data: x and group
## Kruskal-Wallis chi-squared = 43.0625, df = 4, p-value = 0
##
##
##                            Comparison of x by group
##                                 (No adjustment)
## Col Mean-|
## Row Mean |       GE_0     GE_0.1     GE_0.5       GE_1
## ---------+--------------------------------------------
##   GE_0.1 |   3.753692
##          |    0.0001*
##          |
##   GE_0.5 |   2.353951  -1.399741
##          |    0.0093*     0.0808
##          |
##     GE_1 |   4.512182   0.758489   2.158230
##          |    0.0000*     0.2241    0.0155*
##          |
##     GE_2 |  -0.833475  -4.587168  -3.187427  -5.345658
##          |     0.2023    0.0000*    0.0007*    0.0000*
##
## alpha = 0.05
## Reject Ho if p <= alpha/2
```

```
dunn.test(GE_sidewall_displacement$`Value (mm)`, GE_sidewall_displacement$Components)
```

```
##   Kruskal-Wallis rank sum test
##
## data: x and group
## Kruskal-Wallis chi-squared = 67.4496, df = 4, p-value = 0
##
```

```
##
##                            Comparison of x by group
##                                 (No adjustment)
## Col Mean-|
## Row Mean |       GE_0     GE_0.1     GE_0.5       GE_1
## ---------+--------------------------------------------
##   GE_0.1 |   0.369933
##          |     0.3557
##          |
##   GE_0.5 |   3.180822   2.810888
##          |    0.0007*    0.0025*
##          |
##     GE_1 |   6.167859   5.797925   2.987037
##          |    0.0000*    0.0000*    0.0014*
##          |
##     GE_2 |  -0.932026  -1.301960  -4.112849  -7.099886
##          |     0.1757     0.0965    0.0000*    0.0000*
##
## alpha = 0.05
## Reject Ho if p <= alpha/2
```

```
dunn.test(GE_tire_strain$`Value`, GE_tire_strain$Components)
```

```
##   Kruskal-Wallis rank sum test
##
## data: x and group
## Kruskal-Wallis chi-squared = 539.6421, df = 4, p-value = 0
##
##
##                            Comparison of x by group
##                                 (No adjustment)
## Col Mean-|
## Row Mean |       GE_0     GE_0.1     GE_0.5       GE_1
## ---------+--------------------------------------------
##   GE_0.1 |   19.18080
##          |    0.0000*
##          |
##   GE_0.5 |   2.058287  -17.12251
##          |    0.0198*    0.0000*
##          |
##     GE_1 |   3.816809  -15.36399   1.758522
##          |    0.0001*    0.0000*     0.0393
##          |
##     GE_2 |  -0.812640  -19.99344  -2.870928  -4.629450
##          |     0.2082    0.0000*    0.0020*    0.0000*
##
## alpha = 0.05
## Reject Ho if p <= alpha/2
```

```
dunn.test(GE_sidewall_strain$`Value`, GE_sidewall_strain$Components)
```

```
##   Kruskal-Wallis rank sum test
##
```

```
## data: x and group
## Kruskal-Wallis chi-squared = 142.7651, df = 4, p-value = 0
##
##
##                           Comparison of x by group
##                                (No adjustment)
## Col Mean-|
## Row Mean |       GE_0     GE_0.1     GE_0.5       GE_1
## ---------+--------------------------------------------
##   GE_0.1 |  -7.727487
##          |    0.0000*
##          |
##   GE_0.5 |   1.800004   9.527491
##          |     0.0359    0.0000*
##          |
##     GE_1 |   3.205190   10.93267   1.405185
##          |    0.0007*    0.0000*     0.0800
##          |
##     GE_2 |  -0.780235   6.947251  -2.580240  -3.985425
##          |     0.2176    0.0000*    0.0049*    0.0000*
##
## alpha = 0.05
## Reject Ho if p <= alpha/2
```

```
dunn.test(GE_tire_stress$`Value (N/m^2)`, GE_tire_stress$Components)
```

```
##   Kruskal-Wallis rank sum test
##
## data: x and group
## Kruskal-Wallis chi-squared = 14768.1496, df = 4, p-value = 0
##
##
##                           Comparison of x by group
##                                (No adjustment)
## Col Mean-|
## Row Mean |       GE_0     GE_0.1     GE_0.5       GE_1
## ---------+--------------------------------------------
##   GE_0.1 |  -38.67772
##          |    0.0000*
##          |
##   GE_0.5 |   29.55067   68.22839
##          |    0.0000*    0.0000*
##          |
##     GE_1 |  -64.58218  -25.90446  -94.13285
##          |    0.0000*    0.0000*    0.0000*
##          |
##     GE_2 |  -70.89480  -32.21708  -100.4454  -6.312627
##          |    0.0000*    0.0000*    0.0000*    0.0000*
##
## alpha = 0.05
## Reject Ho if p <= alpha/2
```

```
dunn.test(GE_sidewall_stress$`Value (N/m^2)`, GE_sidewall_stress$Components)
```

```
##   Kruskal-Wallis rank sum test
##
## data: x and group
## Kruskal-Wallis chi-squared = 6789.9452, df = 4, p-value = 0
##
##
##                            Comparison of x by group
##                                 (No adjustment)
## Col Mean-|
## Row Mean |       GE_0     GE_0.1     GE_0.5       GE_1
## ---------+--------------------------------------------
##   GE_0.1 |   25.89830
##          |    0.0000*
##          |
##   GE_0.5 |   31.25393   5.355628
##          |    0.0000*    0.0000*
##          |
##     GE_1 |  -27.18380  -53.08210  -58.43773
##          |    0.0000*    0.0000*    0.0000*
##          |
##     GE_2 |  -31.76680  -57.66510  -63.02073  -4.583001
##          |    0.0000*    0.0000*    0.0000*    0.0000*
##
## alpha = 0.05
## Reject Ho if p <= alpha/2
```

As we can see with the test from the tire displacement data, the tires with 1.0 PHR, 0.1 PHR, and 0.5 PHR were the most significant, in that order. Similarly, from the strain data, the tire with 0.1 PHR, 1.0 PHR, and 0.5 PHR were most significant, in that order. After continued analysis of the results from the post-hoc test, we can conclude that the tire with 0.1 PHR graphene is statistically the most significant tire of the four.

Conclusions

After running the Post-Hoc tests, we can conclude that the tire with 0.1 PHR graphene is the most different from the control group tire (tire with 0.0 PHR graphene). Therefore, we can conclude that the tire with 0.1 PHR graphene is the most optimal design for a run-flat tire out of the five tested during experimentation. This is reinforced by visual analysis of the Nonlinear Simulation data, and results from other research conducted including "Optimize Design of Run-Flat Tires by Simulation and Experimental Research" by Xing et al. in 2014.

Further extensions of this testing include different tire shapes and sizes with the given material data, and using more accurate material data collected from physical testing to use in Nonlinear Simulations.