GRIFFITH COLLEGE DUBLIN

Assignment Cover Sheet

Student name: 2968095

Student number: Aditya Ratna Shakya

Faculty: Computing Science

Course: BSCH **Stage/year:** 3

Subject: Numerical Optimization

Study Mode: Full time **Y** Part-time

Lecturer Name: Barry Denby

Assignment Title: F1 Tyre degradation simulation

No. of pages:

Disk included? Yes No

Additional Information: (ie. number of pieces submitted, size of assignment, A2, A3 etc)

Date due: 24/ 12/ 2020

Date submitted: 24/ 12/ 2020

Plagiarism disclaimer:

I understand that plagiarism is a serious offence and have read and understood the college policy on plagiarism. I also understand that I may receive a mark of zero if I have not identified and properly attributed sources which have been used, referred to, or have in any way influenced the preparation of this assignment, or if I have knowingly allowed others to plagiarise my work in this way.

I hereby certify that this assignment is my own work, based on my personal study and/or research, and that I have acknowledged all material and sources used in its preparation. I also certify that the assignment has not previously been submitted for assessment and that I have not copied in part or whole or otherwise plagiarised the work of anyone else, including other students.

Signed: Aditya **Date: 24/12/2020**

Please note: *Students **MUST** retain a hard / soft copy of **ALL** assignments as well as a receipt issued and signed by a member of Faculty as proof of submission.*

Folder Structure:

AdityaRatnaShakya_2968095_Assignment/
- ExperimentalFiles/
  - __init__.py
  - Car.py (Used to implement makeshift SA algorithm)
  - Main.py
  - Test.py
  - Tyre.py (Used to implement makeshift SA algorithm)
- Screenshots/
  - griplevels.png
  - laptimeswithpitstop.png
  - tyredegradation-60kg.png
  - tyredegradation-60kg-fueldepletion.png
  - tyredegradation-105kg-fueldepletion.png
  - tyredegradation-105kg-fueldepletion.png
- Assignment.py (run for the simulation)
- SimulatedAnnealing.py (run for the makeshift SA algorithm)
- GeneticAlgorithm.py

Simulation

The simulation works as intended. However, there are a few factors that seem a bit buggy. This will be later explained with the graph that corresponds to possible flaw in the simulation.

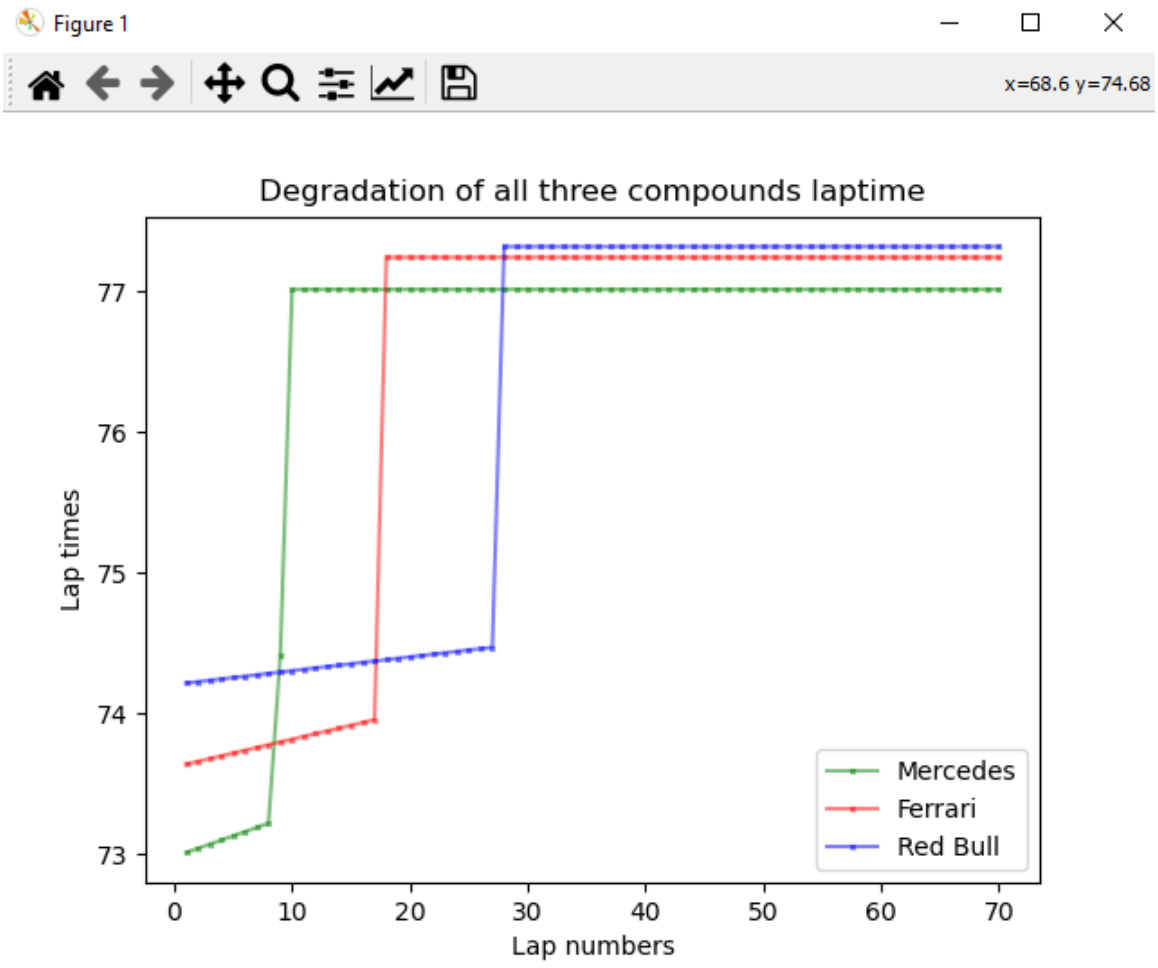
Bugs:
- Grip Level - The grip level for tyres sometimes falls below 0, this results in odd lap times that are not realistic at times. Unable to recreate as the configuration used was not recorded.
- Starting Fuel - If the starting fuel is below 105, which is assumed that all cars start with 105, it results in odd lap times as calculating whilst calculating grip loss due to fuel 105 is hardcoded.

Possible Fixes:
- Grip Level – Setting a check to make sure that if grip level falls below 0 it is set to 0.

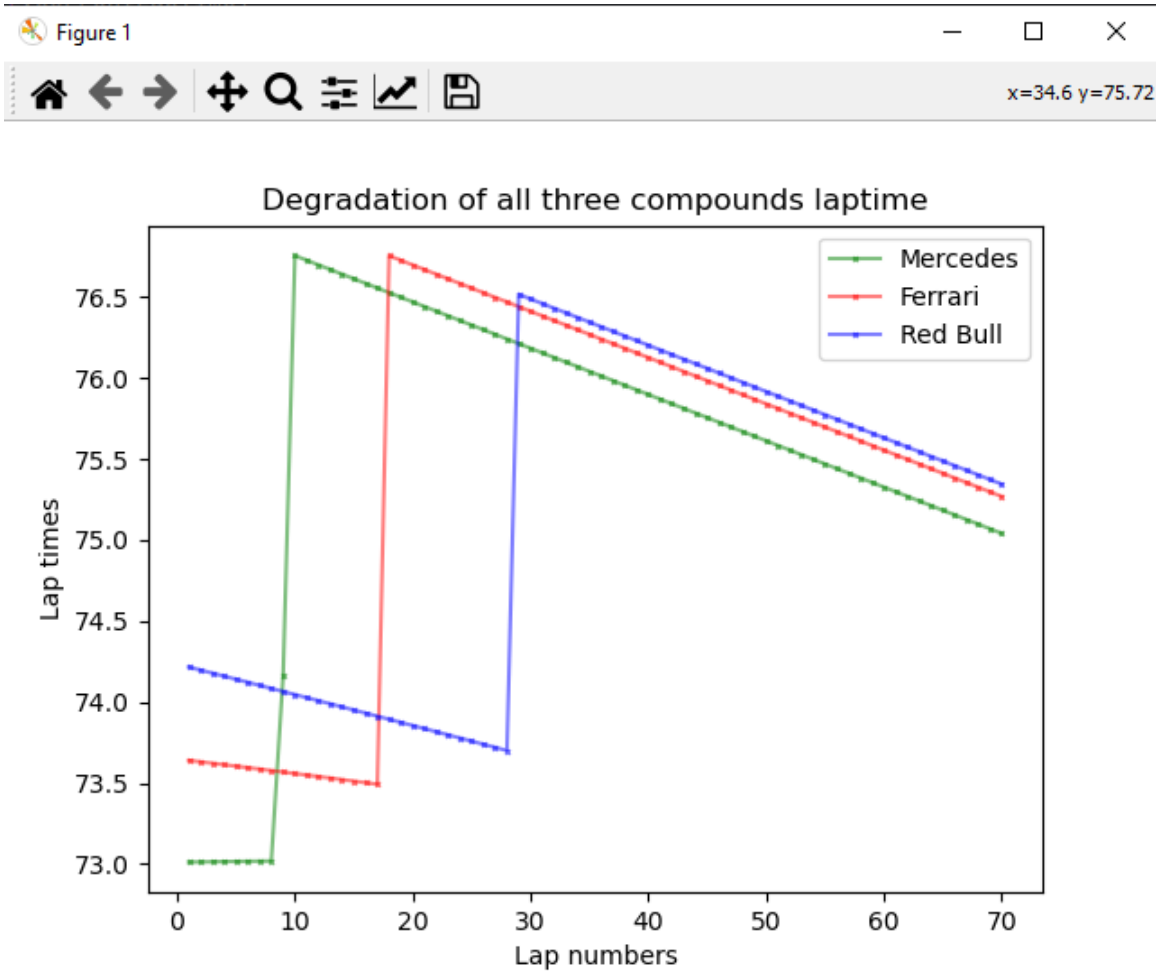
- Starting Fuel – This could be fixed by having starting fuel to be passed as an argument when creating the car object and setting the starting fuel for the car in the constructor.

Graphs:

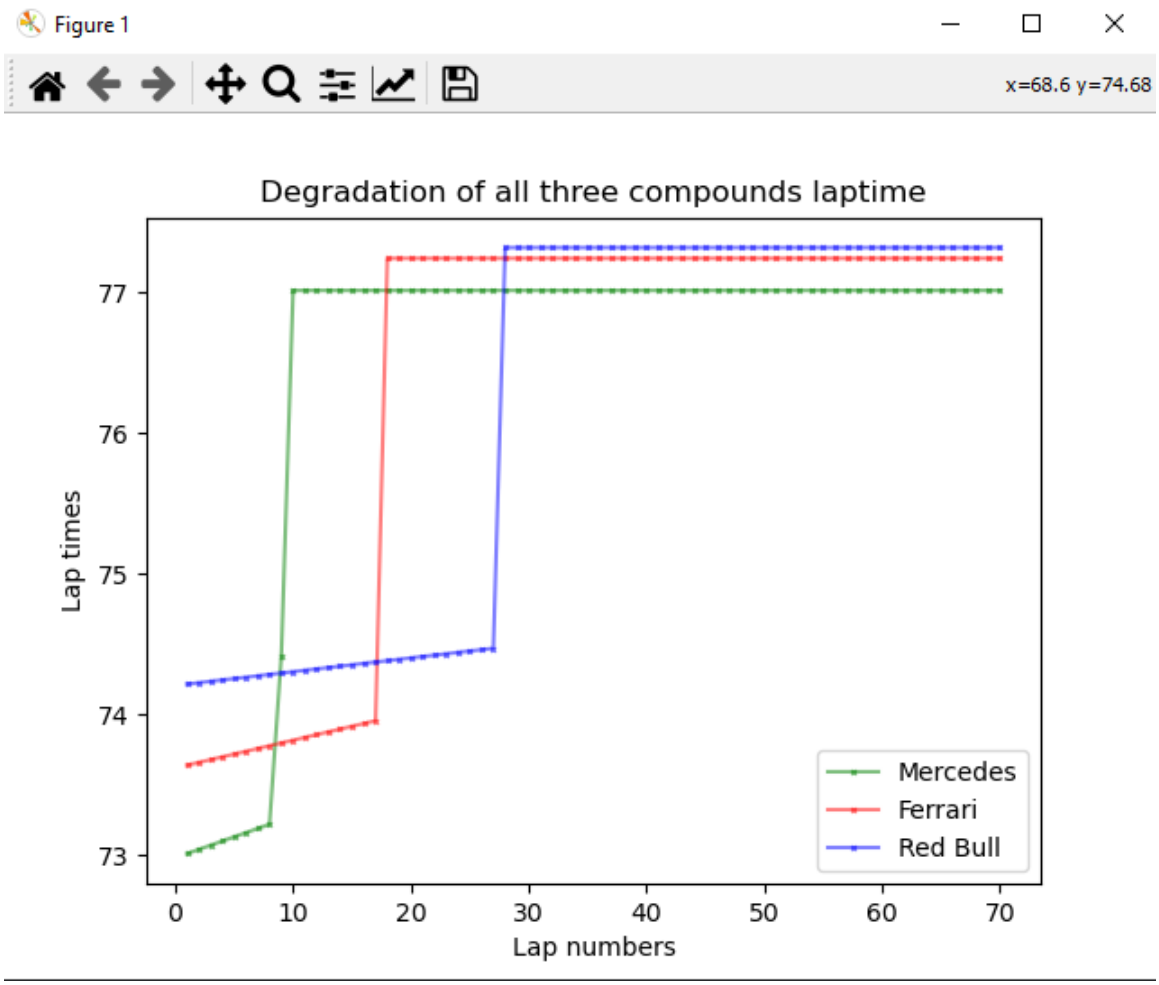
The graphs are not exact to what was given for comparison. The main reasoning for this being the graphs were all generated in the end after coding the entire simulation with grip loss factor and lap time factor set to their respective values, Also, all the graphs have fuel effect and grip loss taken into consideration.



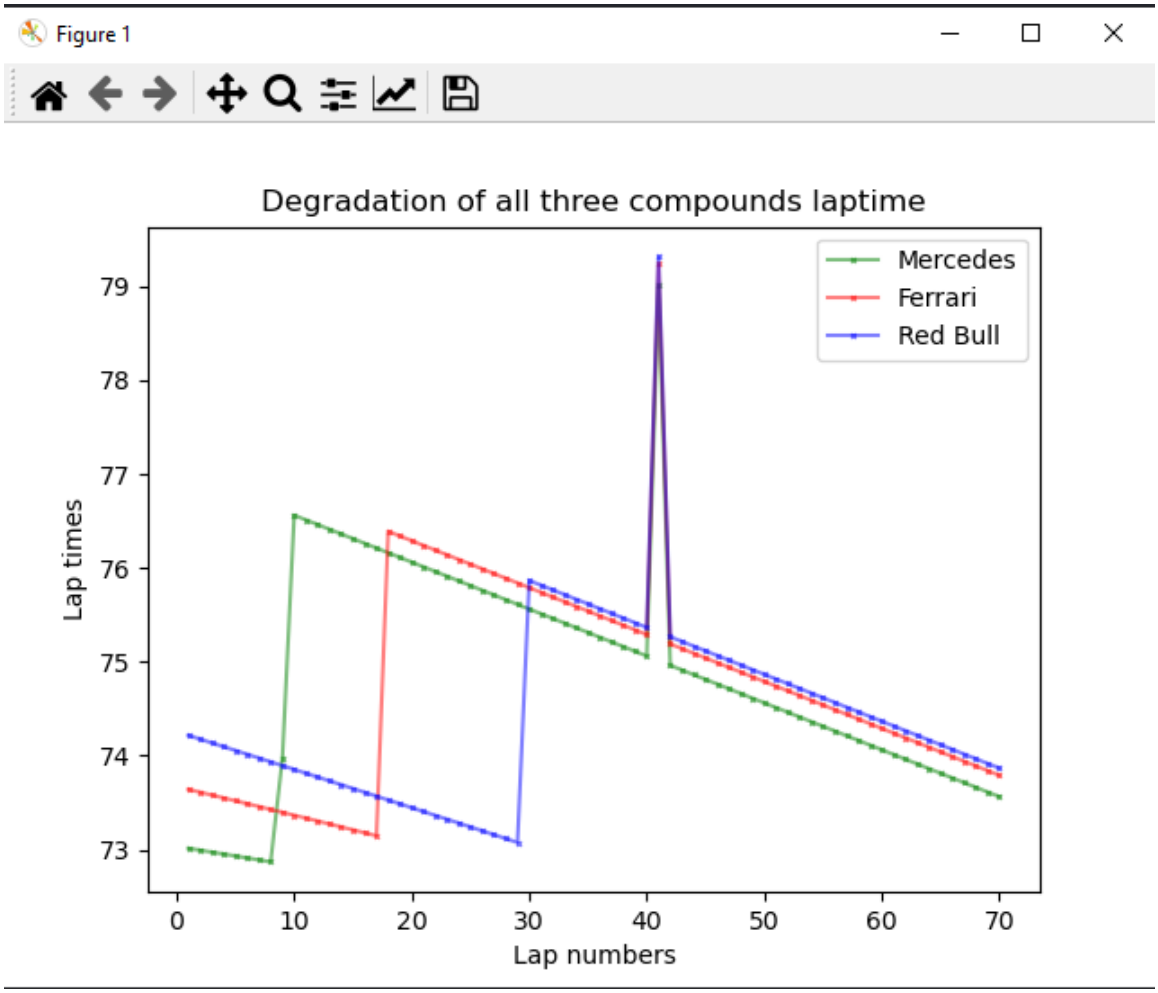
Tyre Degradation- laptimes- 105Kg fuel – no fuel loss



Tyre Degradation- laptimes- 105Kg fuel – Fuel loss

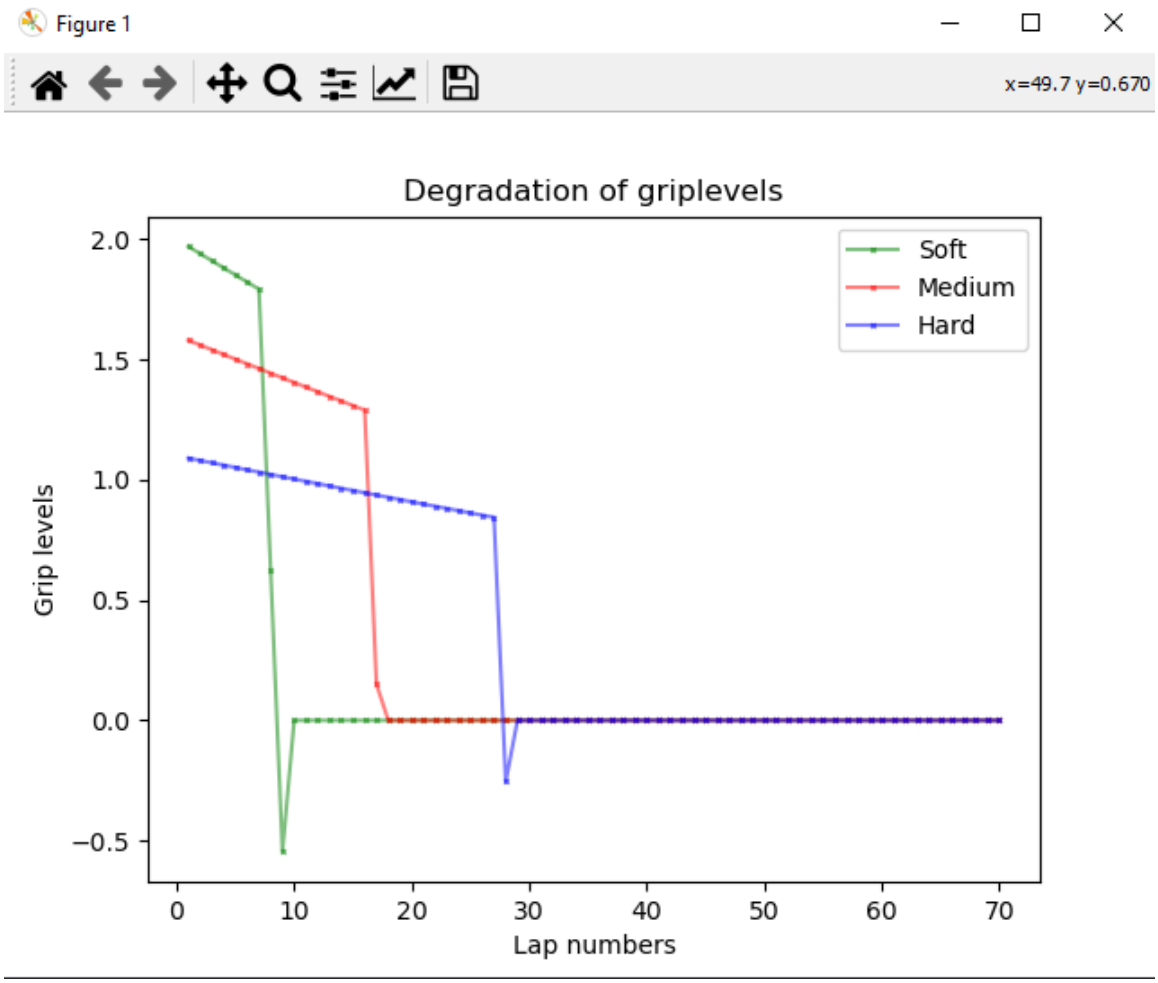


Tyre Degradation- laptimes- 60Kg fuel – no fuel loss



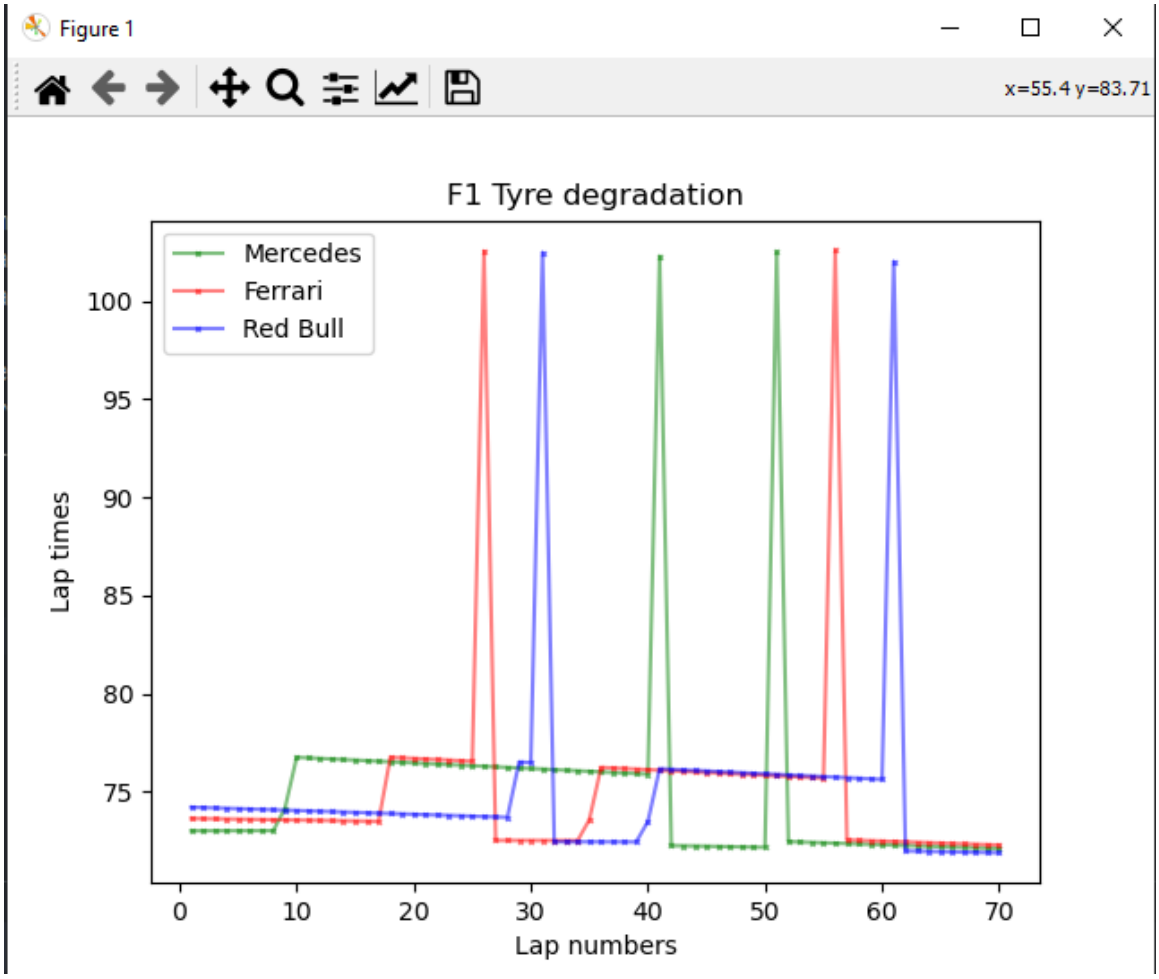
Tyre Degradation- laptimes- 60Kg fuel – Fuel loss

The anomaly in the graph above occurs due to the fuel going below 60 and the way fuel effect is implemented.



Grip levels for tyres after certain laps

The grip level falls below zero before resetting to 0, this is discussed in the bugs section above.



Full race simulated with varying pitstops for different cars

Simulated Annealing

Unable to implement correctly.
I had tried to implement it using the provided methods and failed to do it successfully.

I also tried something different that is not part of the simanneal library. This can be viewed by running/ opening the SimulatedAnnealing.py

For this I tried implementing it in the following way,
- Randomly generating first pit and second pit for the pitstops.
- Randomly get a tyre configuration
- Using the randomly generated pitstops to calculate total race time, compare the generated laptime with the previous laptime if first couple of runs, else compare it with the previous known best laptime. If better laptime, set it to

be best laptime, else ignore.
- After going through this, print out the best known pitstop configuration for the tyre configuration.
I know this is far from simulated annealing method as this is quite linear and removes randomization quite a bit. The best I can describe what I have coded is it calculates the ideal pitstops for the given random set of tyres for the car and the total race time for that configuration

Bugs:
- N/A

Possible Fixes:
- N/A

Genetic Algorithm

Unable to implement.

Bugs:
- N/A

Possible Fixes:
- N/A

Best Configuration

As I was unable to write algorithms to find the best strategy using Simulated Annealing and Genetic Algorithm. I have had to manually test for the best possible configuration for each car using a makeshift algorithm. I am aware that this is very inefficient and does not cover a lot of configurations. I am also aware that the configurations might not be the most optimum configuration.

Best configurations through manual testing / makeshift Simulated annealing algorithm implementation:

Mercedes:
Initial Tyre: soft
Pit Tyre Order: medium, hard
Pitstops: 21, 40

Ferrari:
Initial Tyre: hard
Pit Tyre Order: soft, medium
Pitstops: 35, 50

Redbull:
Initial Tyre: hard
Pit Tyre Order: medium, soft
Pitstops: 35, 59

From what I have observed,
-It is optimal for Mercedes to start with soft and then shift to either [medium, hard] or [hard, medium] with pitstops [21, 40] and [21, 50] respectively.
-It is optimal for Ferrari to start with hard or medium and then shift to soft for the first pit and then have the remaining tyre for the final pitstop where tyre configuration will be [medium, soft, hard] with pitstops [30, 40]
-It is optimal for Redbull to start with hard and then later transition into medium then soft. This order of tyres with pitstops at 35 and 59 seems to give the best race time.