

# Optimising the Performance of a Formula One Car using a Genetic Algorithm

Krzysztof Wloch and Peter J. Bentley<sup>1</sup>

<sup>1</sup> Department of Computer Science, University College London, Gower St., London, UK  
{k.wloch, p.bentley}@cs.ucl.ac.uk

**Abstract.** Formula One motor racing is a rich sport that spends millions on research and development of highly optimized cars. Here we describe the use of a genetic algorithm to optimize 66 setup parameters for a simulation of a Formula One car and demonstrate performance improvements (faster lap times) better than all other methods tested.

## 1 Introduction

Formula One is arguably the première motor racing sport of the world. In the quest to produce ever more competitive racing cars, the teams that design, build and race them spend millions each year on research and development. A modern Formula One car has almost as much in common with a jet fighter as it does with an ordinary road car<sup>1</sup>.

Even minor changes in wing height, suspension stiffness, or tyre rubber compound can make the difference between a car winning and coming last in a race. And with sponsors' money dependent on the television coverage a car receives, a losing car may result in its team losing funding and going out of business.

It is therefore very important, both for the sport and for business, to have effective methods of optimising Formula One car settings. Here we describe the use of a genetic algorithm (GA) to optimise such settings, and demonstrate that the resulting performance is better than all other methods tested of producing car setups.

## 2 Background

Formula One is notoriously secretive, so it is difficult to know if any teams have begun using genetic algorithms for similar purposes as those described here. A recent press release from the Jordan team (Jordan Grand Prix, 2004) hints that they may be considering such technologies. It describes how the team has "...signed up *Scientio Inc.*, an artificial intelligence and data mining specialist, to assist the team in various problem-solving projects," and, "The company's technology includes: data mining

---

<sup>1</sup> From [www.formula1.com](http://www.formula1.com), the official formula one website.

techniques, fuzzy logic, genetic algorithms and software which learns optimum solutions to problems.”

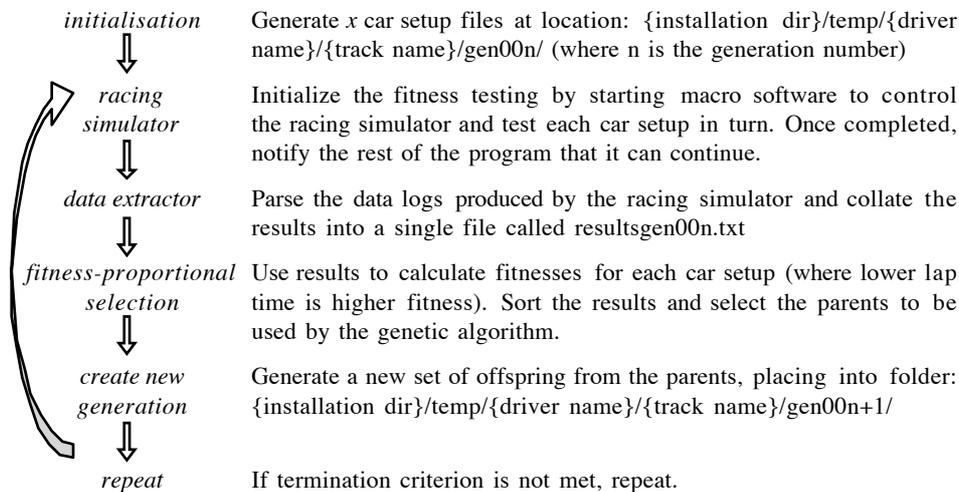
But it seems that published work in the area of car optimization using genetic algorithms is rare. Deb and Saxena (1997) describe work on the optimization of car suspension design for comfort, by minimizing factors such as bouncing, pitching and rolling. They claim to have evolved a suspension design that is “better than a design used by one automobile industry.”

The most common research in this area seems to focus on car design optimization, with numerous examples, which include automobile valvetrain optimization (Kazancioglu et al, 2003), structural automobile design (Leiva et al, 2001), and aerodynamic conceptual design of a sports car (Bentley and Wakefield, 1997).

The apparent lack of work using GAs to optimize Formula One car settings is surprising, especially given the success demonstrated in this work.

### 3 System

Although the authors did not have access to real Formula One cars or data, the popularity of the sport means that accurate car and track simulators capable of data logging are widely available. The Formula One evolutionary optimization system combines a genetic algorithm with such a simulator (linked using additional macro and data parsing software), see figure 1.



**Figure 1.** The Formula One evolutionary optimization system.

### 3.1 Macro Software

From the outset of the development of this system, it was clear that a method for automation would be essential. Linking the GA optimiser to the racing simulator efficiently was critical, so it was determined that macro software would enable the automation of the testing of the cars. The macro would have to be able to launch the simulation software, and initialize any required options, such as loading the current evolved configurations of the car. It would need to be able to simulate the pressing of combinations of any keys which might be necessary, for example throttle control.

Several different macro programs were tried in order to achieve the required effect of total automation from initiation, by the main program, all the way through to exiting after completion in a predictable amount of time. The main difficulty encountered was that certain macro tools did not operate at a significantly low enough operating system level. This meant that once the vehicle simulation was loaded, the required key presses and mouse movements were not being carried out. After a lengthy period of testing different macro software the final solution was to use a combination of two different programs, each handling a different section of the automation. *Macro Scheduler<sup>TM</sup>* was used to load the car simulator; *Hot Keyboard Pro<sup>TM</sup>* was used to simulate the throttle of the car being pressed and to load the desired configuration into the vehicle simulator.

### 3.2 Racing Simulator

Several existing racing simulations were considered but the option selected was *Formula One Challenge '99-'02* by *Electronic Arts<sup>TM</sup>*. Released in 2003, it has an advanced real time physics engine, and models specific cars and tracks with remarkable accuracy, in addition to details such as track/tyre temperatures, and weather effects. Other important factors included the feature for F1 Challenge to output track results to disk<sup>2</sup>, have 68 input parameters for configuration<sup>3</sup>, and artificially driven cars. The latter ability was enabled by altering the configuration of the program to control the steering, braking, and throttle level, see below. The throttle was 'pressed' using a windows macro while the simulator controlled the pressure and the gears.

The software included a driver aids options to help the driving with tasks such as: steering assistance (aids the user with following the correct driving line), auto shifting (keeps the car in the correct gear for the optimum power output), and clutch assistance (ensures clutch is engaged to the correct level for maximum power with minimum wheel spin). These options were not intended to be used simultaneously with each other, but it was discovered that if each was enabled to their maximum setting, (as well as other features), then the desired effect of an artificial driver controlling the car

---

<sup>2</sup> Track results are logged to a location within the installation directory and stored within text files for easy extraction. Each file is named in reference to the time it was created.

This feature is enabled by changing the logging settings within the config file in the root of the simulator.

<sup>3</sup> See (Wloch 2004) for an exact list of all the parameters available.

could be achieved. Another feature was to have the weather set to a constant value, as rain and temperature affect the way the tyres grip the road.

Table 1 provides a description of the majority of some important variables that can be altered in configuration files. In total there are 68 variables. The ranges specified are whole integer numbers which translate to decimal values in the simulator.

**Table 1.** Main variables that affect the performance of the car. Other variables include fuel consumption and steering controls.

Setting name	Range	Its function
<b>Suspension</b>		
Anti-Sway	0-25	Has an effect on the under/oversteer of the car, and the contact that the tyres have with the ground. The value relates to the stiffness of the anti-sway bar.
Toe In settings	0-40	Relates to the angle of the wheels in relation to each other. The variable alters how much the wheels point forwards. This has an effect on directional stability.
Camber settings	0-80	Camber is the angle of the wheel relative to the vertical. The variable alters this angle, and affects the tyres' performance while cornering.
Spring rates.	0-40	The spring rates determine how stiff the springs are and how the vehicle responds in cornering and bumpier surfaces. The can affect understeer/oversteer also. Measured in N/mm.
Packer settings.	0-40	Useful in high-speed situations the packers are related to the spring and ride height.
Ride height.	0-40	This can be varied in millimetres and affects the down force of the car on the track.
Bump damping.	Multiple	There are several variables associated with these settings, with ranges from 0-40. They affect how quickly the suspension responds to the road surface.
<b>Engine</b>		
Rev Limit	0-10	Variations to how many revolutions per minute the engine can reach. Affects acceleration in certain rev ranges.
Gear Ratios	Multiple	There are 15 variables associated with changing the gear ratios. They effect the acceleration of the vehicle. They vary in range up to 0-75.
<b>Aerodynamics</b>		
Brake duct size	0-6	Relates to the size of the ducts, and affects cooling.
Radiator size	0-4	Also affects cooling, and the aerodynamics of the car.
Wings	0-49	Varies the height and position of the wings, changing the down force of the vehicle and its grip on the road.
<b>Other</b>		
Tyre pressure	0-105	Can be set individually for each tyre.
Brake pressure and bias	Multiple	Varies how hard the brakes are applied, and the distribution between front and rear break pressure. Several variables associated with this ranging from 0-45.

### 3.3 Genetic Algorithm

A simple canonical genetic algorithm (Michalewicz 1996) was used to optimise the variables described above. Ordered, fixed-length binary coding was used, where each parameter was encoded into a binary string with maximum length for range of that value, e.g. for the range 0-31, the value 14 is encoded as 01110. Parents were randomly picked from the best 40 percent of the population each generation. Single point crossover was used 100 percent of the time, mutation occurred with a probability of 0.01. Elitism was used to preserve the best individual in the population each generation.

## 4 Experiments

The racing simulator permitted many different racing tracks and cars to be simulated. For this work two tracks and one car was chosen. Several factors were considered in deciding to select two racing circuits which are considerably different to each other:

- **Silverstone, Great Britain**

The Silverstone circuit, located in Northamptonshire, is a generally fast circuit with several slow corners, and a selection of fast sweeping turns. This means that the car should be tuned for higher speeds, with less down force for cornering and instead aerodynamics designed to provide greater straight line speed.

- **Nürburgring, Germany**

The Nürburgring circuit was chosen for exactly the opposite reason to Silverstone. It is a very twisty and tough circuit. The track is quite uneven and the car would need to be configured for high down-force to handle tight corners at speed.

The racing car chosen was the *Williams BMW, 2002*, for purely personal reasons of the first author. The car uses a 10 cylinder engine provided by BMW, while Williams provides the chassis and aerodynamic design. The FW25 chassis in this model was a new generation design with notable differences from the previous models. The 900bhp engine was the most powerful of the Formula One cars at the time, capable of 19,050rpm.

**Table 2.** The five experiments performed using the Formula One evolutionary optimization system.

Track Name	Vehicle	Pop. Size	Generations Completed	Test
Great Britain, Silverstone	2002 Williams	10	23	1
Great Britain, Silverstone	2002 Williams	30	20	2
Great Britain, Silverstone	2002 Williams	40	43*	3
Great Britain, Silverstone	2002 Williams	40	40*	4
Germany, Nürburgring	2002 Williams	40	40	5

## 4.1 Testing Procedure

In total, five experiments were performed, detailed in Table 2. Population sizes were varied to determine the effect on optimisation. The first four experiments tested the 2002 Williams car on the Silverstone track. The final experiment tested the system on Nürburgring. Because the racing simulator models Formula One cars in real time, fitness evaluation was lengthy and each experiment took several days to complete. Evolution was halted manually when convergence of the GA had occurred.

## 5 Results

Figure 2 shows graphs of fitness against time for the five experiments.

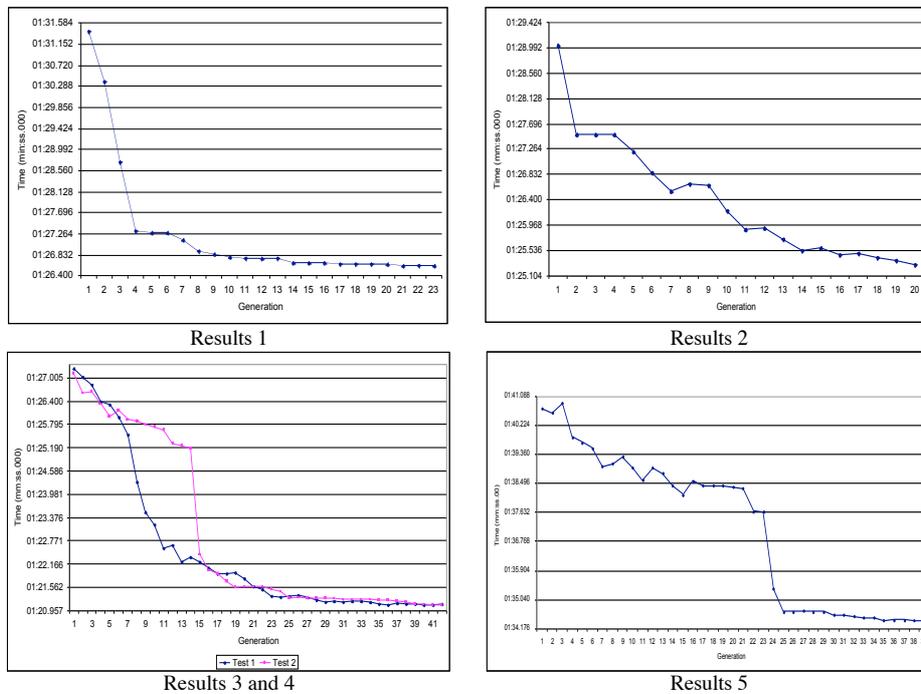


Figure 2. Results of experiments 1 to 5.

### 5.1 Experiment 1

The first test that was performed was with a population size of 10. The initial population, as with all tests, was randomly generated. The aim of this was to initially test the system, see how quickly it converged, and fine-tune GA parameters. The algorithm was halted after 23 generations.

The graph in figure 2 shows the lap time of the fastest car setup for that generation. It was observed that the population appeared to converge after the 14<sup>th</sup> generation, but when watching the vehicles being fitness tested it became apparent that not all the variables were being optimized.

The fastest time achieved was 1 minute 26.608 seconds. Although elitism is being employed, because of noise, the lap time may appear to increase minutely at the 22<sup>nd</sup> generation.

It was decided to run until the 23<sup>rd</sup> generation to see what affect it would have on the vehicle on track. Although the lap time did not appear to decrease any further, noticeably, when doing so, it became apparent that certain variables did improve over time, and this was taken into account when planning further runs of the algorithm. The decrease in lap time from the initial randomly generated variables to the final optimized 23<sup>rd</sup> generation car setups was very significant. The lap times improved by up to 5 seconds from some of the better initial configurations. It seems likely that the apparent inability of the algorithm to optimize some of the parameters was because the population size was so limited. For example, only the first 3 gear ratios were properly optimized with the top gears not even being used when testing the car.

## 5.2 Experiment 2

This test was with a population size of 30. The aim of this was to see what effect slightly increasing the population size would have on the algorithm's ability to optimize the variables. It was run for 20 generations.

The fastest lap time achieved was 1 minute 25.397 seconds after 20 generations, see figure 2, Results 2. The car setups appeared to converge after the 18<sup>th</sup> generation (note the difference in scale compared to the previous graph). The increase in population size had a marked effect on the ability of all the variables to converge to their optimum.

The effect of the larger population size was visible when comparing the way the gear ratios had been configured in Test 2, with Test 1. The ratios that evolved used all the gears in Test 2 and the ratios were well suited to the track.

## 5.3 Experiments 3 and 4

Test 3 and 4 were carried out in the same manner, and are included in this section together as the comparison of their results and the setups generated is revealing.

The algorithm was run on test 3 until the lap time no longer appeared to decrease significantly. Test 4 was then run for the same number of generations. The two were then compared, see figure 2, Results 2 and 3. The same optimum is reached after 40 generations on each run of the algorithm. The fastest time produced was 1 minute 21.050 seconds.

Although the same lap time was reached after 40 generations there were significant differences in the way the algorithm produced the results. For example, a common fault with all the car configurations developed after 40 generations on test 3. When

turning into one corner on Silverstone, known as 'Club,' there was a tendency for the front left wheel to lock up under braking, see fig. 3.



**Figure 3.** Image produced by the racing simulator, *Electronic Arts™ Formula One Challenge '99-'02*. The speed of the Formula One car had to drop from over 170 mph, on entry to this corner, to 50 mph. With settings evolved in experiment 3, the front left wheel locks. This problem did not occur in the setups produced in experiment 4.

Other differences between the evolved settings existed in the suspension settings, although there were similarities. For example, the brake bias between front and rear wheels, the weight distribution on the vehicle, the engine rev limit and the aerodynamic parameters all evolved to almost exactly the same settings in both experiments. The exact parameters can be found in (Wloch 2004).

#### 5.4 Experiment 5

The test at Nürburgring was carried out to see how effective the genetic algorithm would be at optimizing the car setup on a very different type of track. The test was carried out in an identical method to tests 3 and 4, running for 40 generations.

The graph shows that the program was capable of effectively optimizing the car setup on a track style which was totally different to Silverstone. The parameters appeared to converge at around the same generation number as in Test 3 and 4: generation 25. Nevertheless, the variables continued to evolve to better settings the longer the simulation was run.

The graph gives an impression that the track times increase and decrease quite considerably as the tests continue, but this is due to how difficult it is to achieve a consistent lap time on a more demanding track. The gear ratios which evolved were perfectly suited to the track, in that for each corner the engine revolutions stayed consistently high allowing for fast acceleration out of the corners, and a high enough final gear ratio so that in the straight part of the track the car could reach a top speed of 193mph. No visible problems developed with the car setup, as with test 3.

## 6 Discussion

The default settings of the car (provided by the simulation software itself) produce lap times which on all occasions were beaten by everything generated by the tests shown in the results. The initial results from Test 1 did not provide very significant gains over the default setups, but they were certainly visible, and subsequent tests showed huge increases in performance. The setups generated towards the end of Tests 3 and 4 produced gains, consistently, of over 5 seconds in lap times. Running the tests for longer and with larger populations could only have increased those gains.

Tests 3 and 4 also showed that the system is capable of producing various solutions to optimizing the car around the track. This was visible in the way that in test 3 the front left wheel of the car locked up under braking while in test 4 a different solution was created without similar problems. Although test 3 produced a setup which could cope with the lock up, in reality it would not be good for tyre and brake wear.

In comparison with tunings produced by a human, the system was easily able to compete and beat settings that the author was able to produce, and in every case that was tested it had faster lap times on car setups that were obtained from the internet, from sources that had greater experience with F1 Challenge than the author.

The best car setup, produced from Tests 3 and 4 (which was provided by test 4), was compared against 3 car setups. The first was recommended settings provided by the simulator, the second was a configuration produced by the first author, and the third was from an 'internet expert.' For the sake of consistency, all the vehicles were tested in exactly the same manner. The automatic driver drove the vehicles around the track and controlled all the usual functions, such as throttle level, braking and steering. The amount of fuel that the cars were carrying was set to the lowest level to allow them to complete the laps, each car was given the same tyres (soft compound), and the track/outside temperature was fixed on each test. Silverstone was the testing track. Table 3 summarises the results.

The reduction in fuel weight carried by the cars is the explanation for the difference between the results quoted earlier and in table 3, but as is clearly visible, the reduction in lap time by the evolved settings is considerable, although the stresses on the car are probably far greater and it is visible that the car is driving on the limit.

**Table 3.** Comparison of laptimes produced by cars with evolved settings with three other sets of parameter settings.

Car setup	Lap time (mm:ss.000)	Difference (ss.000)
Evolved Settings	1:20.349	
Expert Setup	1:21.221	+0.879
Krzysztof's Setup (author)	1:21.439	+1.090
Default System Setup	1:22.769	+2.420

## 7 Conclusions

This work drew together several different software technologies to produce a system that fulfilled its goals of optimising the settings of a Formula One car using evolutionary computing, and achieving real improvements in lap times compared with all other methods tested. The choice to use the EA developed simulator proved highly successful when it came to the implementation. The automatic control of the cars around the track provided a highly consistent method of fitness testing, and the data output by the program was used very effectively by the genetic algorithm.

### 7.1 Future Work

Initial future work which could be conducted with this system would be to proceed with more tests. These would involve testing the same car on many more different tracks and testing older model of the car, or different cars.

If the program was to be developed further, then the telemetry feature on F1 Challenge should be employed. Although the data is encoded in an unrecognisable format, the belief is that a far more effective algorithm could be created if the exact details of the results from a lap could be incorporated into an evolutionary process. This would involve looking at the effectiveness of individual components of the car and certain parts of the track, and evolving the parameters, in a similar method to before, to take into account this extra information.

A faster testing method, which would reduce the time for fitness testing, and also the overall optimization process, would be very useful. Currently the time for testing makes using large population sizes unattractive.

Finally, it is anticipated that the same methods could be used to optimise real Formula One car setups with equal success, so useful collaborations with willing teams are a real possibility.

## References

- (1) Bentley, P. J. & Wakefield, J. P. (1997). Generic Evolutionary Design. Chawdhry, P.K., Roy, R., & Pant, R.K. (eds) Soft Computing in Engineering Design and Manufacturing. Springer Verlag London Limited, Part 6, 289-298.
- (2) Deb, K. and Saxena, V. (1997). Car suspension design for comfort using genetic algorithms. In Thomas Back (Ed.) Proceedings of the Seventh International Conference on Genetic Algorithms (pp. 553--560). [download size: 52K]
- (3) Jordan Grand Prix (2004). Jordan finds a new technical partner. Jordon Press Release, published on F1-Live.com at 15:07, 09/02/04.
- (4) Emre Kazancioglu, Guangquan Wu, Jeonghan Ko, Stanislav Bohac, Zoran Filipi, S. Jack Hu, Dennis Assanis and Kazuhiro Saitou. (2003) Robust Optimization Of An Automobile Valvetrain Using A Multiobjective Genetic Algorithm. Proceedings of DETC'03 ASME 2003 Design Engineering Technical Conferences Chicago, Illinois, September 2-6, 2003

- (5) Leiva, J.P., Wang, L., Recek S. and Watson, B. (2001), Automobile Design Using the GENESIS Structural Optimization Program , Nafems Seminar: Advances in Optimization Technologies for Product Design, Chicago, Illinois, USA, October 22-23, 2001.
- (6) Michalewicz, Z. (1996). Genetic Algorithms + Data Structures = Evolution Programs. 3rd edn. Springer-Verlag, Berlin Heidelberg New York .
- (7) Wloch, K. (2004) Optimising Formula One Settings using a Gentic Algorithm. B.Sc. final project report, Department of Computer Science, University College London.