

THE: traffic light simulator and optimization using Genetic Algorithm

Ashfaq H. Farooqi¹, Ali Munir¹ and A. Rauf Baig¹

¹ Department of Computer Sciences,
National University of Computer and Emerging Sciences,
Islamabad, Pakistan.

Abstract. Almost all the populated countries especially under developed countries are facing traffic problem in their major big cities like Karachi, Lahore, etc in Pakistan. There are number of possible solutions as widening the roads, make overheads, and make different routes to same destination etc but these all require healthy resources. There is a requirement to optimize the current road model by properly allocating the time to traffic signals. Researchers have proposed different simulation models and used various approaches to optimize the traffic light signals. In this paper we have provided information about THE traffic light simulator that is implemented using Visual Studio 2008 in C#. We have also illustrated the genetic algorithm (GA) approach that we have used to optimize the traffic light signals using THE simulator. Result shows that our simulator works better and GA provides better results as it matures.

Keywords: THE traffic light simulator, Genetic Algorithm (GA)

1. Introduction

Traffic is increasing at a rapid pace on the roads proportional to the increase in population. The traffic problem is faced by all big cities in all the countries. As the traffic increases it causes traffic congestion. Due to this drivers and passengers are facing number of serious problems like wastage of time, depression, pollution, risk of accidents, and other serious problems. These problems cause an extra burden to the society and business. According to [1], in 2002, 750 million Euros loss has been reported in Greater Copenhagen due to traffic problem. Different approaches can be used to minimize the traffic problem such as public transport can be improved, existing communication mechanisms can be expanded or using the present available roads, by proper handling of the signal times. The first two approaches require many resources as money, labor, area etc. They seem very difficult to achieve in many places due to lack of resources. But the third approach looks a better choice to adopt in these circumstances. Researcher has proposed different strategies to provide a mechanism to properly set the traffic signal parameters.

People are using cars or other vehicles that basically use roads as media; for moving to their destination; is increasing day by day. Current analysis of big cities shows that the amount of cars on the roads is increasing tremendously and the available resources are limited. Proper handling of traffic seems to be the market demand. According to [2], intelligent optimization algorithms are the necessity of an optimal control of traffic light signals. Traffic light problem is a well known optimization problem from a decade or so. Traffic congestions can be minimized by proper handling of traffic light [3]. They may be helpful in enhancing the capacity of the roads and traffic flow too. Several intelligent algorithms are proposed for achieving this purpose as fuzzy logic, swarm algorithms as particle swarm optimization (PSO) or ant colony optimization (ACO), reinforcement learning, genetic algorithm etc.

Simulations made for analyzing different problems are thought to be powerful tools for analyzing complex problem that change with situation as in traffic light problem; the amount of traffic changes randomly [4]. Initially, we were looking for an open source traffic light simulator. But we can't find the one

we are looking forward. That's why, we developed our own traffic light simulator and named it THE. Some information about THE traffic light simulator that we have implemented in C# using Visual Studio .NET 2008 is discussed briefly in Section 2.

Genetic algorithms are used in various applications for searching for the optimal result. Section 3 will cover the implementation detail of the proposed methodology based on GA. Section 4 will illustrate the experiments and analysis. Future dimensions are discussed in Section 5 and finally conclusion.

2. THE traffic light simulator

The basic purpose of developing a traffic light simulator is to make a tool that can be used to learn or test different optimization techniques. On the traffic problem firstly, we were looking for open source tools but we didn't find a suitable one. Actually, evolutionary algorithms as genetic algorithm, particle swarm optimization (PSO), ant colony optimization (ACO) etc require the chromosome encoding. This is accomplished by creating a string of complete solution. In our implementation, we have created a separate class that controls the values of signals. Name of our traffic light simulator is THE. The sense behind THE is to create a proper tool for testing algorithms that require chromosome encoding. It is implemented in C# using the Visual Studio .NET 2008. The road model objects are assembled using the Expression Blend trial version. The simulation is heavy one require Microsoft .NET framework 3.5 to run. THE traffic light simulator is briefly discussed in subsections.

2.1. Basic Information about THE

The basic components of THE traffic light simulator are linkRoad1, linkRoad2, Highway1, and Highway2 as shown in Fig. 1.

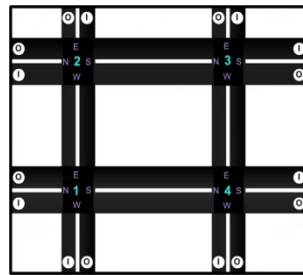


Fig. 1: THE traffic light simulator.

The vertical roads are the two link roads. All the four roads are two way. The left one is linkRoad1 and the right one is linkRoad2. The horizontal roads are highways. The top one is Highway1 and bottom one is Highway2. Traffic on the link road will be less than highways.

2.2. Traffic Generation

There are two different rates by which cars arrive on various roads:

- Link Road: There are two link roads in our simulator. There is one car per two seconds arrival rate on the link roads. The total number of car(s) on the link road is equal to the time of simulation divided by two. If simulation time is 300 seconds or 5 minutes then there will be 150 cars on each link road.
- Highway: There are two highways too in our simulator. There is one car per second arrival ratio on the highways. The total number of car(s) on the highway is equal to the time of simulation in seconds. If simulation time is 300 seconds or 5 minutes then there will be about 300 cars on each highway.

The total number of cars that will try to reach their destination in each simulation will be cars on link roads plus cars on highways = $4 * \text{simulation time} / 2 + 4 * \text{simulation time}$. If time is 300 seconds then the total number of cars is 1800.

2.3. Signals

Fig. 1 shows that there are four intersections namely 1, 2, 3, and 4. Each intersection has four signals; North N, South S, East E, and West W. Hence there are 16 signals in total. When car(s) will reach close to any signal it will check the status of that signal. The status of signal is red or green. If it is green cars will continue their motion to reach its destination point else it will stop at red signal.

2.4. Car movement

In THE traffic light simulator, the highways are the roads with longer length as well as heavy load roads. The attribute of the cars on highway differ here. As the cars on highways cover a distance after each tick or second is equal to $HIGHWAY_LENGTH / 35$ while cars on the link road covers distance after a single tick or second is equal to $LINKROAD_LENGTH / 20$. Hence, the speed of the cars is higher on highways then link roads.

2.5. Simulation timing

The time for every simulator is hard coded. A simulation timer is set to some value. It uses the class DispatcherTimer. The timer that is integrated into the System.Windows.Threading.Dispatcher queue is processed at a specified interval of time and at a specified priority. The value is decremented after as second passes. The particular simulation ends when timer reaches to zero. The simulation time was initially set to 120 seconds but latter changed to 300 seconds.

2.6. Future enhancement

We are looking forward to add some more functionalities in THE. Some of the limitations that will be removed in future work are mentioned below:

- The road scenario depicted in THE traffic light simulator so far is a trivial one which require enhancement.
- Cars are moving in straight direction. Thus it is not showing real world problem.
- Cars climb on one another to have more cars or traffic. But there should be any other way of doing so. One solution might be increasing the width of roads.

3. Proposed optimization approach

In [5], authors have provided information about the design and the tests they have performed on their evolutionary optimization architecture. J. Sanchez et al. applied genetic algorithm as the core process to optimize traffic light problem for Las Ramblas. Genetic algorithm was first presented and utilized for solving a problem by John Holland in 1975 [6]. Our approach is also based on the genetic algorithm but it differs from the one used by J. Sanchez et al. Our approach differs from that in various aspects such as simulator, encoding of chromosome, cross-over technique, mutation rate and selection criteria.

3.1. Chromosome Encoding

Encoding of the chromosome is the major step in using any evolutionary optimization approach. There are 4 intersections and 4 traffic lights each in the presented model. The chromosome should have the knowledge about all the four intersections and for each signal in each intersection. The size of chromosome will be $4 * 4 = 16$ as shown in Fig. 2. The numbers ranging from 1 to 4 are just showing the sequence in which the signal will green. As initially the North signal opens for time t assigned by the chromosome.

①				②				③				④			
N	S	W	E	N	S	W	E	N	S	W	E	N	S	W	E
1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4

Fig. 2: Chromosome Encoding.

The values assigned to each signal will be in a range from 5-20. Initially the values are assigned randomly between that ranges. This value is set to maximum of 20. As the traffic is coming at a nice pace so it is our assumption that the signals should not be red for more than 60 seconds. Each signal's turn is after the other three signals. If it is close for more time then the wait time might increase tremendously. To keep the follow smoothly we kept the time between 5 and 20. Then this time is further optimize using genetic algorithm.

3.2. Fitness Function

Fitness of each chromosome is the total wait time of ALL THE CARS. This total wait provides the information about the traffic that how much cars waited at the signals to reach the destination. Lower the

total wait time greater the throughput. So, the criteria is set as lesser the total wait time better the chromosome will be and it has the better chance of survival in the next generation.

3.3. Population size and Selection criteria

Population size is the number of individuals or chromosomes in one generation. In our implementation, the population size is ten to analyze the diverse search space.

3.3.1. Initial population

Initial population is generated by randomly assignment of values with some constraints. In our implementation, 10 chromosomes are generated by randomly assigning values ranging from 5 - 20. Each simulation then runs assigning values to the signals based on the particular chromosome values. When the evaluation of initially generated 10 chromosomes completes, selection criteria is applied for the next generation population.

3.3.2. Selection criteria

Selection criteria are used to get representation of chromosomes that covers the search space nicely. Diversity in the search space is tried to achieve here too. Here, we have used two elite from the previous generation having best fitness. Two offspring are achieved from the two selected parent after single fixed point crossover operator. Other six are randomly generated chromosomes as generated in initial population. These ten are then evaluated and again the same selection criteria are used for the next generation.

3.4. Crossover operator

Crossover operator is used to combine the good feature of parents in the offspring. Single fixed point crossover is applied. Two parent chromosomes having best fitness value are selected. The crossover point is the midpoint. It can be any other point too. After crossover, newly created offspring will have half the values of one parent and half the values of second.

3.5. Mutation operator

Mutation is usually applied by randomly selecting any one of the 10 selected chromosomes for next generation, and then randomly selected the signal or a single value to be mutated. Once the mutant point is selected then it is randomly assigned a value from 5 - 20. In our selection criteria of population, mutation will only affect the first four chromosomes; two elite and two offspring. That's why mutation is applied to only first four chromosomes. The rate of mutation is about two values in two randomly selected chromosomes.

3.6. Termination criteria

Termination criterion is set to stop the working of algorithm. Usually, it is set to a fix number of generation. Maturity level can also be used as termination criteria. Here, simulation terminates if results don't improve for next few generation.

4. Experiments and analysis

THE traffic light simulator is still in its early age. Enhancement can be done in its model, traffic generator, type of intersection points etc. But, we have used it as a test-bed as alternatives are not available for our idea. It is simulated by applying the GA and simulation terminates after 10 generation because results are not improving. Each chromosome is given 300 seconds or five minutes and after that the fitness value for that chromosome is calculated. The results are stored in a file containing information about the start of the current session, population number or generation number, iteration number is the chromosome number, the values assigned to the chromosome and lastly the waited time.

THE traffic light simulator is left for about 10 hours in running condition. The results achieved after that time are shown in Fig. 3. It shows that there are some values for which the wait time is less and some are those which have more time. An interesting thing to note is that initial set of every generation has better values than the others. Reason is that, the first four are the chromosomes which are affected by genetic operators while others are random.

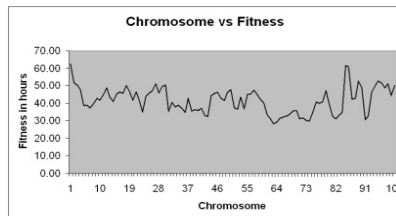


Fig. 3: Result after 10 generations.

Chromosome having best fitness in each generation is depicted in Table I. It contains the chromosome number with their fitness value. The results analysis clarify that after seventh generation fitness value starts increasing further. Initially the chromosome fitness or total wait was 62.58 hours. But later on as genetic algorithm progresses the wait time is reduced to 28.35 hours for 1800 cars. So, the wait time reduced is about 34 hours.

Table 1: Chromosome with best fitness value from each generation

Chromosome#	Time (s)	Time(Hrs)
7	133961	37.21
14	147537	40.98
23	126235	35.07
36	125081	34.74
43	117113	32.53
52	132048	36.68
63	102069	28.35
74	107605	29.89
82	112225	31.17
91	110112	30.59

5. Future Work

THE traffic light simulator is still in its early ages. We are looking to add more functionality in this simulator. After improving it we will apply genetic algorithm in different ways by changing the type of cross-over, mutation rate, or selection criteria to get better results.

6. Conclusion

Traffic problem is increasing as the population is tremendously enhancing in the major big cities especially in under-developed countries. There is a requirement of a proper traffic light simulator that could help the researcher to test their algorithms to optimize the traffic signal times. THE traffic light simulator will be a handy tool for the researcher to test their approaches. Work is in progress to make it a proper test-bed. We have further implemented a mechanism based on GA; to optimize the values of the traffic light signals. We have simulated for ten generations. Number of generations are small but are enough for the traffic model used in THE. We finally found a time sequence for the signals after evaluating 10 chromosomes for 10 generations that could reduce the total wait time of the cars efficiently from a random assignment of time.

7. References

- [1] A. Warberg, J. Larseny and R. M. Jorgensen. *Green Wave Traffic Optimization - A Survey*. Internet article. February 2008.
- [2] M. Wiering, J. V. Veenen, J. Vreeken and A. Koopman. Intelligent Traffic Light Control. *Technical report*, Institute of information and computing sciences, Utrecht University, 2004.
- [3] E. H. Nugrahani, L. Alamsyah and R. Ramdhan. Traffic Light Strategies in Optimizing the Traffic Flow. *Proc. of the 2nd IMT-GT Regional Conference on Mathematics, Statistics and Applications*. Malaysia. June 2006.
- [4] M. Hanna and J.Y. Ruwanpura. Simulation tool for manpower forecast loading and resource leveling. *Simulation Conference*. Washington. December 2007, pp. 2099-2103.
- [5] J. Sanchez, M. Galan and E. Rubio. Applying a Traffic Lights Evolutionary Optimization Technique to a Real Case: Las Ramblas Area in Santa Cruz de Tenerife. *Proc. of IEEE Transactions on Evolutionary Computation*. February 2008, **12** (1): 25-40.
- [6] A. P. Engelbrecht. *Computational Intelligence an Introduction*. John Wiley & Sons Ltd, England, 2002.