

Control of Road Traffic Using Learning Classifier System

Rezaee A*

Department of Mechatronics and system Engineering, Faculty of New science and technologies, University of Tehran, Tehran, Iran

*Corresponding author: Rezaee A, Department of Mechatronics and system Engineering, Faculty of New science and technologies, University of Tehran, Tehran, Iran, Tel: 982161118530; E-mail: arrezee@ut.ac.ir

Received date: June 29, 2016; Accepted date: July 20, 2016; Published date: July 26, 2016

Copyright: © 2016 Rezaee A. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Abstract

Road traffic control through proper control of junction signals is one of the complicated control issues. A conventionally used system is the rule-based system which is often employed in designing systems with deterministic states. In this paper, we have tried to study the control issue with the idea of distributed control through using Learning Classifier Systems (LCS). It means, controlling signal of any junction is done separately from other junctions through an independent Learning Classifier System and with the purpose of decreasing the lines of automobiles queue in the conducted streets to the junction. Furthermore, learning classifier systems have been used in order to control junction traffic within distributed control system.

Keywords: Traffic control; Learning; Classifier; System; Junction

Introduction

The approach of controlling junction traffic is one of the complex controlling approaches because of span and complexity of urban transportation network and having the essence of random, time-flexibility and existence of various and unpredictable rummages. Yet, many control strategies have been applied in order to encounter this issue. Strategies for signal control of road traffic have developed from fixed-time plans in which the controlled signals have a recalculated and fixed time with the possibility of selection between various recalculated plans, to flexible and responsive systems, in which the signal timings that are implemented vary according to traffic flows at the time of implementation.

Considering that the urban transportation network is much extended and since the control signals are many, applying distributed strategies have satisfactory results. Substantial benefits have been achieved by progressing on each of the two fronts of enhancing the responsiveness of traffic control systems and extending advanced optimization approaches from isolated road junctions to road networks with a high density of signal-controlled junctions. The approaches that have been developed successfully for responsive control include heuristics such as rule-based approaches, optimization approaches such as MOVA and more flexible approaches that respect the uncertainties inherent in the data and models.

Early responsive systems demonstrated the difficulty in improving on the performance that could be achieved by good fixed-time control systems such as TRANSYT (Robertson, 1969; Vincent et al., 1980). But the development of various successful responsive systems led to the creation of more successful systems. These include the SCOOT system [1] which uses a feed-forward approach to plan for the arrival at a junction of traffic that is detected as it leaves an upstream junction, or SCATS [2] which uses feed-back from stop-line detectors to inform on queue exhaustion.

A conventionally used system is the rule-based system which is often employed in designing systems with deterministic states.

However, since knowledge is naturally imprecise and involves uncertainty, expert systems do not work well in many cases. Other methods have emerged to help humans in solving the problems they need to solve. One such method is creating expert systems using Bayesian networks based on Bayes' theory of probability. Rules of probability are utilized in these networks to handle uncertainty and to represent uncertain knowledge [3,4].

In this paper, learning classifier systems have been used in order to control junction traffic within distributed control system. For example, control system of a simulated robot that each leg is represented a separate system [5] or the commercial systems in which the learning classifier systems have been used to represent a trader in an artificial market [5]

Learning Classifier Systems (LCS) can be used for optimization in a way that offers substantial promise for application in traffic-responsive signal control systems where the way in which the control responds to variations in traffic flows can be adapted according to measured conditions. This is important in order to achieve traffic control that is sufficiently flexible to respond rapidly when traffic conditions change in a fundamental way, as occurs at the start of a peak period. Yet, it is not unduly sensitive to short-term variations in flow that may happen because of an accident.

The importance of this approach for traffic control is that it offers a means by which signal control strategies can be developed directly according to their performance. This turnover stays opposite of the strategies which have been designed according to mathematic formulas. This closed-loop approach to development of control strategies offers several advantages over the use of traditional explicit optimization formulations. These include flexibility in respect of objectives so that multiple and varying needs can be accommodated, ability to use various different kinds of detector data according to their availability and freedom from dependence on a single explicit evaluation formula that is intended to embody the whole of a traffic mode[6,7].

In this article, we briefly introduce learning classifier systems in the second part. In the third part, traffic issue considered at this article has

been introduced with easing theories and graphical simulators. The result of applying the introduced methods which has been extracted from simulator software has been introduced in the fourth part and in the fifth part there is deduction.

Learning Classifier Systems

Machine learning is one of the modern research issues and yet many efforts have been done in order to use the developed techniques in this issues in the real world. The complex and/or ill-understood nature of many problem domains, such as data mining or process control, has led to the need for technologies which can adapt to the task they face. Learning Classifier Systems, in fact, are a machine learning technique which combines reinforcement learning and evolutionary computing and other heuristics to produce adaptive systems. The central idea in all evolutionary computing techniques is searching a problem space by evolving an initially random population of solutions such that better or fitter solutions are generated over time. Thus, the population of candidate solutions is seen to adapt to the problem. In Learning Classifier Systems, these evolutionary computing techniques have been used along with the reinforcement learning techniques.

Reinforcement learning is, in fact, learning through effort and error via the reception of a reward. Numerical reward is dedicated to the act-creator rule. The learner attempts to adapt the action input and system, with the aim of being able to maximize future reward. The approach is loosely analogous to what are known as secondary reinforcers in animal learning theory.

Learning Classifier Systems are rule-based systems, where the rules are usually in the traditional production system form of "IF state THEN action". Evolutionary computing techniques and heuristics are used to search the space of possible rules, whilst reinforcement learning techniques are used to assign utility to existing rules, thereby guiding the search for better rules.

The idea of LCS was firstly introduced by Holland in the year 1976 and it was completed as a codified technique in the year 1986 by him. Holland's Learning Classifier System receives a binary input from its environment. This binary input is, in fact, indicates the current situation of environment which can be any under-control system. This input binary has placed in an internal working memory space and it is saved in a message list (Figure 1). Then, the system determines an appropriate response based on the received input and after looking among the rules, performs the indicated action, usually altering the state of the environment. Afterwards, an appropriate reward is accrued to the occurred behavior in the environment which caused its reinforcement or weakening. This process is usually going on and the system is interacting with the environment.

The rule collection consists of a population of N condition-action rules or in other words N "classifiers". The rule condition and action are strings of characters from the ternary alphabet {0,1,#} in which # expresses the state of apathetic. So 1#1 input consists the both inputs of 101 and 111. Both condition and action components are initialized randomly in the primitive population. For any existing rule in the population, a fitness scalar associated to indicate the "usefulness" of a rule in receiving reward.

On receipt of an input to the system, the rule-base is scanned and any rule whose condition matches the input message is selected and becomes a member of new rule-base naming "Match Set"[M]. A rule is selected from those rules comprising, through a bidding mechanism,

to become the system's external action. The amount of rule authority during bidding is counted through the following formula:

$$\text{Rule Authority} = \beta(\text{specificity}) (\text{fitness})$$

In which the parameter of specificity is the proportion of non-# bits in the rule. The parameter β is a constant less than one. The received reward from the environment is dedicated to the winning rule and it increases the fitness.

Moreover, the LCS employs a mechanism of genetic algorithm which is applied to all the rules population in order to make a rule-base cooperate each other to solve a problem. The possibility of execution of this algorithm in the cycle of this system depends on a "P" supposition that it has been considered $p=0.1$ in this article (Figure 1).

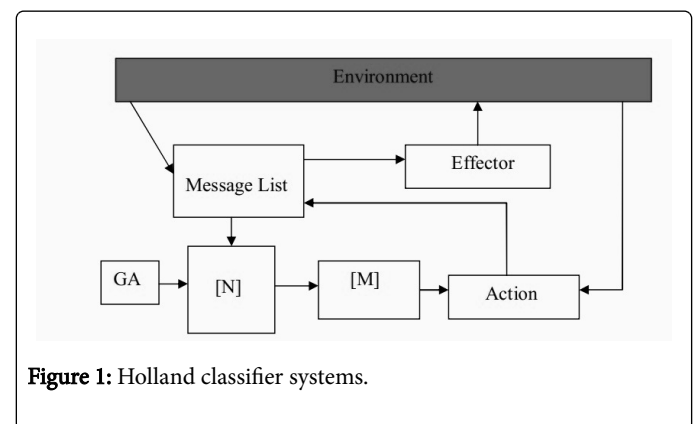


Figure 1: Holland classifier systems.

Traffic Control Issue

In this article, one issue of traffic control has been simulated to one simple transportation network with considering some easing theories. Controlling any junction is done separately and by a separate learning classifier system. In order to ease the process of modeling and graphic simulation, suppose the following supposition:

- The streets are north-south and east-west.
- The traffic flow randomly and with a fixed average rate enters the network from northern and western entries and evacuates from southern and eastern exits.
- All cars drive straightly and do not turn in junctions.
- All cars speed is stable.

In the programmed software, there is the possibility of creating north-south and east-west streets in any numbers.

Any junction's controlling signals which are made separately from other junctions include the condition of traffic light and the time of remaining lights in this condition.

The supposition is that there are the queue-length distinguisher sensors in the traffic lights of two streets conducted the junction.

Figure 2 shows a schematic from the designed simulator. The length of car queue created in any street conducted to any junction is showed beside the junction. The total time of simulation at seconds is shown at the right bottom corner of the map. In simulating programs the following parameters have been spotted in the map and each of them is adjustable.

- The length of any street between two junction=250 m
- Street Width=20 m

- Number of Street Riding Lines=4
- The Maximum distinguishable queue length by sensors=31 stable cars
- Cars speed=36 Km/h

The Maximum distinguishable queue length is 31 stable cars which equals 150 m. 2 queue length is registered for any junction and every one need 5 bits for registration. So the sum of two numbers is sent to learning classifier system of junction as an input message in the form of a 10 bits string. Four possible situations for time length of any situation of junction are 10, 20, 35 and 60 seconds which need 2 bits for selection. Totally, 4 bits is associated to determine the time length of two possible situations of any traffic light. According what was mentioned, any rule has been composed of 14 bits.

In order to minimize the maximum queue length created behind any traffic light, the reward dedicated to any winning control rule is calculated in the following formula:

$$\text{Reward}=4(16-L)2$$

$$L=\min(16,\max(L1, L2))$$

This reward needs 10 bits for saving.

Figure 3 shows the designed menu for adjusting learning classifier system parameters. The adjusted parameters have been supposed the same for all the systems of junctions.

The Results of Simulation

We do the simulation process considering the following condition:

- Number of North-south streets=5,
- Number of East-west streets=2,
- The rate of entering traffic to northern streets from left to right has been supposed 4000, 10000, 6000, 400, 12000 cars per hour.

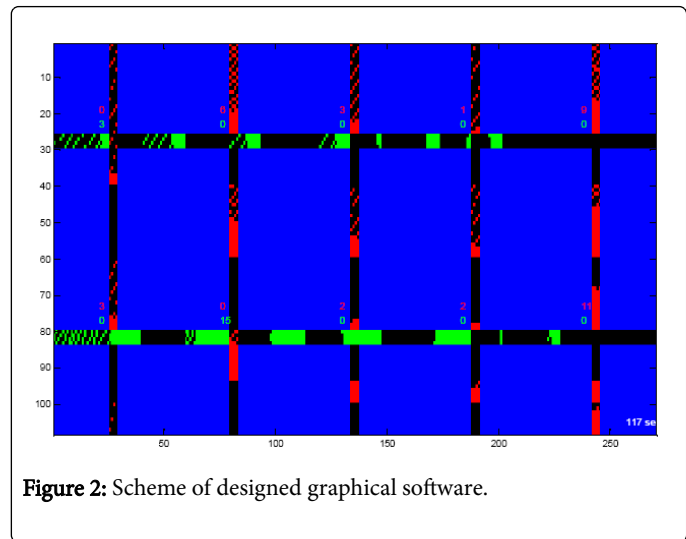


Figure 2: Scheme of designed graphical software.

The rate of entering traffic to eastern-western roads has been supposed 6000 and 12000 cars.

Number of rules of primitive population for any junction equals 400 rules.

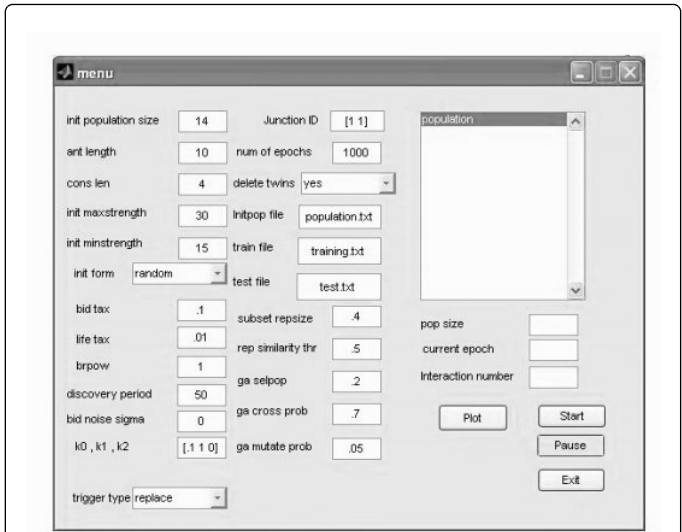


Figure 3: Menu that designed for jam traffic control.

If a car encounters with no traffic light, it needs 50 seconds time to go out of network in north-south streets. This time is 125 seconds in east-west streets. The outcomes of simulation considering the adjustments and the average of crossing time from the network have been shown in the Tables 1 and 2 for each street.

Street Number	1	2	3	4	5
Average of entering traffic rate (car/hour)	4000	10000	60000	4000	12000
Average Time of Navigating the Network	82	85/6	82/8	81/7	86/7

Table 1: The average time of navigating the network (to seconds) for north-south streets.

Street Number	1	2
Average Entering traffic rate (care/hour)	6000	12000
Average time of Navigating Network (Seconds)	220/1	226/4

Table 2: The average time of navigating the network (to seconds) for east-west streets.

Conclusion

In this paper, the usage of learning classifier system in solving the traffic control problem was studied and the system turnover was evaluated with considering a simple traffic model. The main profit of using learning classifier systems is the existence the high flexibility of these systems in front of input codes and optimization purposes. The designed system is in a distributed form, meaning that controlling any

junctions is done separately from any other junction. We can suppose the control purpose of any junction distinct from other junctions. So the possibility of applying stronger controls on the junctions which have significant role in the traffic of the whole network became easier.

References

1. Lanzi PL, Stolzmann W, Wilson SW (2000) Learning classifier systems: from foundations to applications. Springer, p: 13.
2. Bull L (2015) A brief history of learning classifier systems: From CS-1 to XCS and its variants. *Evolutionary Intelligence* 8: 55-70.
3. Bull L, Sha'Aban J, Tomlinson A, Addison JD, Heydecker BG (2005) Applications of learning classifier systems. Springer pp: 276-299.
4. Russell SJ, Norvig P (2004) Artificial intelligence: A modern approach. Prentice Hall.
5. Prasad R, Pal J, Pant AK (1995) Multivariable system approximation using polynomial derivatives. *J Inst Eng, India* 76: 186-188.
6. Wilson DA (1970) Optimal solution of model reduction problem. *Proc Inst Electr Eng* 117: 1161-1165.
7. Tomlinson A, Bull L (2001) An investigation into LCS road traffic junction controllers. UWE Learning Classifier Systems Group: Technical Report pp: 1-9.