International Journal of Research in Advent Technology (IJRAT) Special Issue
E-ISSN: 2321-9637
Available online at www.ijrat.org
National Conference on "Role of Information Technology in Social Innovations"
26th & 27th February 2019

Traffic Signal Controller Optimization Using Neural Networks

Mr. K.D.Bamane, Ketan Kulkarni, Sanchit Khedkar, Nitin Choure, Rajvardhan Gangurde

Abstract- The optimization of traffic light control systems is at the heart of work in traffic management. Traditional traffic light control systems use pre-timed controllers for managing traffic at intersections. But this method is extremely static and is very inefficient for a dynamically changing system like traffic flow. The traditional pre-timed controllers cannot adapt to the ever changing traffic conditions and hence result in large amounts of waiting time for vehicles. We propose a model for traffic controllers using Machine Learning as the core to address the issue of dynamically changing traffic so that our controller can determine which lane should be given the green signal and for how long to minimize the waiting time. By using machine learning algorithms based on the attribution of rewards according to the results of the actions selected by the controller, we can obtain a control policy that tries to optimize the urban traffic flow.

1. INTRODUCTION

Traffic Signal Controller (TSC) are at the heart of managing traffic everywhere. The existing systems use pre-timed controllers for these TSC but they are proving to be inadequate for the dynamically changing traffic conditions. Some lanes have green signals even when there is no traffic while other lanes are waiting even when there is heavy traffic. To address this dynamically changing traffic issue we have come up with a solution of Machine Learning controlled TSC.
These TSC's will issue how much time a signal should be green depending on the traffic in all lanes at that intersection. This will help minimize the average waiting time of vehicles at an intersection and thus improving average travel time as well. Another benefit could be reduction in pollution due to vehicular emissions as less time is wasted waiting.

2. PROPOSED SYSTEM

The proposed model in the project can be summarized as follows:
At each time step, control action is generated by the NN controller with the input of traffic condition. Then action of last time step is reinforced according to the observed reward form the traffic condition. Learning and training process is online and continuous
The proposed system contains the following modules:
NEURAL NETWORK MODULE:
This module has the following functions:
Create the neural network with 3 hidden layers.
Take the input and try to determine the best possible action to take from the action set to minimize the waiting time.
Train the neural network to minimize the waiting time.
Calculate reward from the action taken by deciding whether the action taken helped the final goal of minimizing the average waiting time or not.
Depending on the reward change the next action to be taken.
Calculating the cost function, gradient for learning of the neural network.

User Interface Module:
Render a representation of a road intersection on the screen with traffic signals for all lanes.
Show the number of vehicles in all lanes at the current instant of time.
Show the signal colors for each lane that is red, yellow, and green.
Display the average waiting time.

Pre Timed Controller Module:
For comparison with the neural network system.
Calculate the average waiting time when each lane has fixed time set for green signal.

2.1 System Specifications:

Equipment Requirements:

1. CPU Quad Core and above
2. Ram 4 GB and above
3. Hard Disk 400 GB and above

Programming Requirements:

1. Windows 8 and above
2. Linux operating system
3. Python
4. OpenGL

3. THEOREM:

The algorithm used is Neural Networks using backward propagation for error calculation and forward propagation for activation calculation. The input layer consists of 8 input nodes where 4 node give the number of vehicles in the 4 lanes of the intersection while the other 4 nodes consist of average waiting time of vehicles in the individual lanes.The next layer is the hidden layer which does some calculation to provide the activations for the output layer. The output layer consists of 2 nodes, one for determining which signal should be set green and the other for determining how long the signal should be set green. Using feedforward propagation we can calculate activations for the next layer based on the input. Using Backward Propagation we calculate the error in all the layers.

Backward-Propagation:

Given Training Set $\{(x^{(1)}, y^{(1)}) \ldots\ldots (x^{(m)}, y^{(m)})\}$

Set $\Delta_{i,j}^{(1)} := 0$ for all (l,i,j), (hence you end up having a matrix full of zeros)

For training example t =1 to m:

Set $a^{(1)} := x^{(t)}$

Perform forward-propagation to compute $a^{(l)}$ for l = 2,3,….,L

Using $y^{(t)}$, compute $\delta^{(L)} = a^{(L)} - y^{(t)}$

Where L is our total number of layers and $a^{(L)}$ is the vector of outputs of the activation units for the last layer. So our "error values" for the last layer are simply the differences of our actual results in the last layer and the correct outputs in y. To get the delta values of the layers before the last layer, we can use an equation that steps us back from right to left:

Compute $\delta^{(L-1)}, \delta^{(L-2)}, \ldots, \delta^{(2)}$ using $\delta^{(l)} = ((\Theta^{(l)})^T \delta^{(l+1)}) .* a^{(l)} .* (1-a^{(l)})$

The delta values of layer l are calculated by multiplying the delta values in the next layer with the theta matrix of layer l. We then element-wise multiply that with a function called g', or g-prime, which is the derivative of the activation function g evaluated with the input values given by $z^{(l)}$

The g-prime derivative terms can also be written out as:

$g'(z^{(l)}) = a^{(l)} .* (1-a^{(l)})$

$\Delta_{i,j}^{(1)} := \Delta_{i,j}^{(1)} + a_j^{(l)} \delta_i^{(l+1)}$

We update our new Δ matrix.

$D_{i,j}^{(l)} := (1 / m) (\Delta_{i,j}^{(1)} + \lambda\Theta_{i,j}^{(l)})$, if $j \neq 0$
$D_{i,j}^{(l)} := (1 / m) (\Delta_{i,j}^{(1)})$, if $j = 0$
The capital-delta matrix D is used as an "accumulator" to add up our values as we go along and eventually compute our partial derivative.

3.1 System Architecture

We proposed a system to deal with this problem using machine learning. The proposed system consists of a 3 Layered Feed-Forward Neural Network with 1 hidden layer.

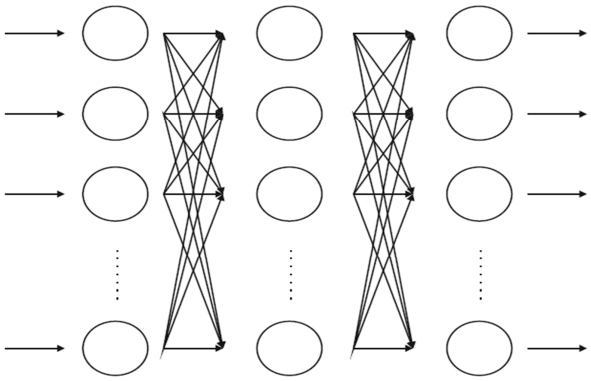
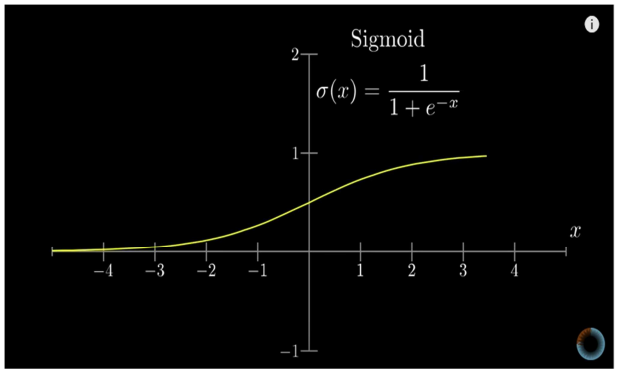


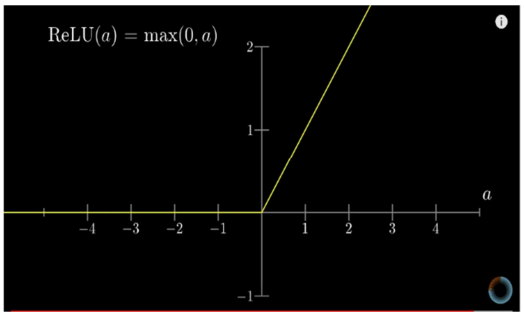
Figure: System Architecture of Proposed System

At each time step, control action is generated by the NN controller with the input of traffic condition. Then action of last time step is reinforced according to the observed reward form the traffic condition. Instead of using the sigmoid function for the process of normalization or rather squishing it between the range of 0 and 1 we will be using the ReLU function or the Rectified Linear Unit Function. The Sigmoid function is used for the process of squishing the data between a certain range (generally values between 0 and 1 are preferred). The graph of sigmoid function can be seen below.



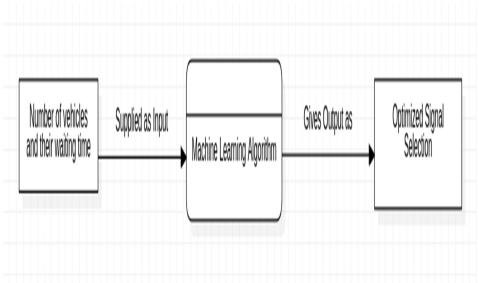
As seen from the above figure, the sigmoid function converts all negative values and brings them closer to zero while all positive values are converged towards one. While the values between negative two and positive two hang somewhere close to 0.5. But very few modern neural networks use Sigmoid Function anymore and it has become old-school. The main reason behind it is the emergence of ReLU or Rectified Linear Unit Function. ReLU has proven to be a lot easier to train and also lot faster as compared to Sigmoid. The ReLU function can be seen below.

SS

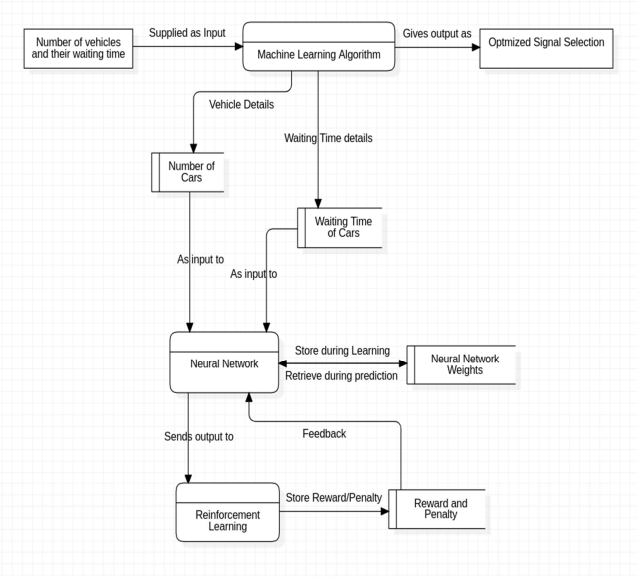


ReLU function is inspired from biological analogy that a neuron will be activated only if it passes a certain threshold and will remain inactive in other cases. As seen from the figure that ReLU remains inactive for all weighted sums leading upto 0 (which is the threshold value) and then it returns the same value for all positive values.

4. FIGURES, TABLES AND PHOTOGRAPHS



DFD - 0



DFD - 1

Acknowledgments

We have taken endeavors in this task; in any case, it would not have been conceivable without the benevolent help and help of numerous people and associations. We might want to stretch out our genuine gratitude to every one of them. We are very obligated to Mr. K.D.Bamane for his direction and steady supervision just as for giving essential data with respect to the undertaking and likewise for his help in finishing the task. We might want to offer our thanks towards our folks and our Head of I.T. Division Dr. Preeti Patil for their benevolent co-task and consolation which helped us in consummation of this venture. Moreover, I might likewise want to recognize with much thankfulness the vital job of the staff of DYPCOE, Akurdi, who gave the authorization to utilize all required hardware and the important materials to finish my undertaking stage I. We are likewise profoundly appreciative to the Principal of DYPCOE, Dr. Vijay Wadhai and my folks for their money related and calculated help and for giving vital direction concerning task's execution.

REFERENCES

[1] R. E. Skill, "Pay as you park," Planning, vol. 71, no. 5, pp. 4– 8,May 2005.
[2] National Travel Survey England, Department for Transport, London,U.K., Sep. 2, 2015. [Online]. Accessible: https://www.gov.uk/government/transfers/framework/transfers/attachment_data/record/457752/nts2014-01.pdf.

[3] D. C. Shoup, "Cruising for stopping," Transp. Strategy, vol. 13, no. 6,pp. 479– 486, Nov. 2006.
[4] A. le Fauconnier and E. Gantelet, "The time searching for a stopping space:Strategies, related aggravations and stakes of stopping the executives infrance," in Proc. And so on, Sep. 2006, pp. 1– 7.
[5] IBM Global Parking Survey: Drivers Share Worldwide Parking Woes,IBM, Armonk, NY, USA, Sep. 28, 2011. [Online]. Accessible: https://www-03.ibm.com/squeeze/us/en/pressrelease/35515.wss.
[6] D. C. Shoup, "The mind-boggling expense of free stopping," J. Plann. Educ. Res., vol. 17,no. 1, pp. 3– 20, Fall 1997.
[7] K. Mouskos, J. Tvantzis, D. Bernstein, and A. Sansil, "Mathematicalformulation of a deterministic Parking Reservation System (PRS) withfixed costs," in Proc. tenth MELECON, 2000, vol. 2, pp. 648– 651.
[8] Y. Geng and C. Cassandras, "New brilliant stopping framework dependent on resourceallocation and reservations," IEEE Trans. Intell. Transp. Syst., vol. 14,no. 3, pp. 1129– 1139, Sep. 2013
[9] SFpark, 2015. Gotten to on: Feb. 30, 2015.[Online]. Accessible: http://sfpark.org/.