Comparison of Nash equilibria and Optimum in a Road Network

A Project Report

submitted by

HIMAJA.K

in partial fulfilment of the requirements
for the award of the degree of

BACHELOR OF TECHNOLOGY

DEPARTMENT OF Electrical Engineering
INDIAN INSTITUTE OF TECHNOLOGY MADRAS.

JUNE 2013

THESIS CERTIFICATE

This is to certify that the thesis titled **Comparison of Nash equilibria and Optimum in a Road Network**, submitted by **Himaja.K**, to the Indian Institute of Technology, Madras, for the award of the degree of **Bachelor of Technology**, is a bona fide record of the research work done by him under our supervision. The contents of this thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Prof. Krishna Jagannathan
Research Guide
Assistant Professor
Dept. of Electrical Engineering
IIT-Madras, 600 036

Place: Chennai

Date: 7th June 2013

ACKNOWLEDGEMENTS

I would like to take this opportunity to thank all the people who have given me unwavering support and guidance at all times.

I am grateful to Dr. Krishna Jagannathan and Dr. Gaurav Raina for their continual support and invaluable advice throughout the duration of the project. I would like to thank Dr. Gitakrishnan Ramadurai and Dr. Karthik Srinivasan from Civil Engineering Department for their help with obtaining the traffic data. I would also like to thank all the professors who have taught me at IIT Madras for imparting their knowledge.

There are simply no words to express how grateful I am for the wonderful set of friends and classmates I have at IIT Madras. I am thankful to Maneesha, Meghana, Pranitha and Madhavi for being the best friends anyone could wish for, Sunaina, Sohini, Meghana Bande, Pranitha, Meghana Kolan, Chitra and Prakruthi for the incredibly fun mugging sessions and the even more fun non-elec sessions. There are many more people I would like to thank for making my stay at IIT Madras special.

Most importantly, I would like to thank my father for the never-ending support and encouragement at each and every point of my life, my mother for being my moral compass and the voice inside my head and last but not the least, Hemanth for being the jolliest brother and my most scathingly honest friend.

ABSTRACT

KEYWORDS: congestion games; non-cooperative games; Nash equilibrium; Wardrop equilibrium; social cost; marginal cost pricing; series-parallel graph; Price of Anarchy; Price of Collusion

The distribution of traffic on roads converges to an equilibrium known as Nash equilibrium. As a result of each user(player) acting in their own interest, the system reaches a state where no individual user can benefit from changing their strategy. However, the collective price paid (in terms of resources or time taken) by the users can be further minimised if there is additional information provided to the users. Price of anarchy is a measure of damage caused to social benefit by anarchy in the network. Levying additional tax at edges likely to cause maximum damage is one way to provide additional information to the users.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Introduction

1.1.1 Congestion games

The behaviour of traffic on roads can be modelled as a $congestion\ game$. A congestion game is a $non - cooperative\ potential\ game$. In general, network games differ in the kind of $interaction$ between the players and the type of traffic. Games are usually classified as $cooperative$ and $non - cooperative$, $discrete$, $atomic$ and $nonatomic$.

In Roughgarden *et al.* (2002) a network game is denoted by $\Gamma(E, c, S, n, a)$,

where$P$ is the set of n players of k player types,E is the set of edges e and s is a $strategy$ belonging to the set S.

The edge cost (or latency) functions are given by $c_e(\xi)$ where ξ is the $load$ on the edge e. We usually assume the latency functions to be positive and convex functions in ξ. $a_{S,e}$ is the rate of consumption on the edge e due to the strategy S which is usually taken to be 1, i.e., any two players are assumed to be interchangeable. An action distribution x is the $measure$ of the number of players based on their choice of strategy.

1.1.2 Classification of games

If there exists a partition on P such that the elements are either $atoms$ (i.e., they cannot be split further) or $null\ sets$ then the game is $atomic$. Otherwise, it is a $nonatomic$ game. $Discrete$ games are a subset of $atomic$ games whose probability measure is a $discrete$ probability distribution.

In a $noncooperative$ game, the players do not form coalitions and choose their strategies such that at equilibrium none of the players have an incentive to change their strategies. If the incentive of players to change their strategy is given by a global $potential$ function then the game is known as a $potential$ game.

In$congestion$ games the payoffs are determined by a global $potential\ function$ and they are $noncooperative$, which means that the payoff to each player depends only on the player's strategy and on the number of other players choosing an interfering strategy.

Games can be either of $pure$ or $mixed\ strategy$. $Pure\ strategy$ means the players have a decided path once they choose a strategy. In a $mixed\ strategy$, the player can randomly choose from the set of $pure\ strategies$ with a finite probability. The corresponding equilibria (if possible) are called $pure\ equilibrium$ and $mixed\ equilibrium$.

In a congestion game with predetermined cost function c, $x_e = \sum_{i=1}^{k} \sum_{S \epsilon S_i} a_{S,e} x_S$ gives the total measure of traffic on edge e due to all strategies $s \epsilon S$. $c_S(x) = \sum_{e \epsilon S} a_{S,e} c_e(x_e)$ gives the total price paid by the users choosing a particular strategy $s\ \epsilon\ S$. Thus, the $social\ cost\ function$ for a given distribution of traffic x is calculated as,

$$C(x) = \sum_{i=1}^{k} \sum_{S \epsilon S_i} c_S(x) x_S$$

Our goal is to minimise this $social\ cost\ function$ in order to maximise the collective benefits of all the players in P.

1.1.3 Nash equilibrium, Price of Anarchy and Marginal cost

$Nash\ equilibrium$ is defined as the state of equilibrium in a $noncooperative$ game where knowing the strategies of other players, no player can find an alternate strategy to increase his $gain$. $Wardrop\ equilibrium$ is defined as the state of equilibrium in which the latency due to all the routes is equal and is lesser than the latency associated with any unused routes in the network. $Price\ of\ Anarchy$ (PoA) is defined as the worst-case ratio between the latency at $Nash\ equilibrium$ to the latency when there is $optimal$ flow of traffic. $Marginal\ social\ cost\ function$ $c_e^*(\xi)$ is given by the derivative of the original social cost,

$$c_e^*(\xi) = \xi c_e^{'}(\xi) + c_e(\xi)$$

Thus, *marginal cost* can be understood as the cost incurred after adding an *additional load* to the system. Marginal cost gives us an idea of which links are more prone to create bottlenecks.

Roughgarden *et al.* (2002)quantifies the lower bound on the price of anarchy in noncooperative games with convex latency functions. The paper studies and identifies nonatomic congestion games whose equilibria are *approximately equal*. That is, the optimizing strategy for individuals *approximately* leads to a social optimum. Every *nonatomic congestion* game has *atleast* one equilibrium and all distinct equilibria have equal social cost.

A class of *differentiable* and *semi−convex*[1] cost functions as *standard*. Let x be the equilibrium action distribution and x^* be the desired optimal action distribution. For this class of *standard* cost functions, the price of anarchy $\rho(\Gamma)$ can be upper bounded by,

$$\alpha(C) = sup_{c \epsilon C} \, sup_{x:\, c(x)>0} \left[\lambda\mu + (1-\lambda)\right]^{-1}$$

where $\lambda = x^*/x$, $\mu = c(x^*)/c(x)$ and $c^*(x^*) = c(x)$.

If x is an equilibrium action distribution for (E, c, S, n, a) and $x^*$is the action distribution for $(E, c, S, (1+\delta)n, a)$ for $\delta > 0$, then

$$C(x) \leq \frac{1}{\delta} C(x^*) \tag{1.1}$$

Roughgarden and Tardos (2002) works with a *continuous flow* of traffic, which is ensured by the assumption that each player controls ε amount traffic as $\varepsilon \to 0$. It upper bounds the price of anarchy to $4/3$ when the latency function is linear. If the latency function is continuous and non-decreasing, then the total latency of the routes chosen by selfish players maybe arbitrarily larger than the minimum possible total latency. However, the latency at Nash equilibrium is no more than the optimal latency due to *twice* the traffic, which is a special case of the *continuous traffic* equivalent of the result obtained in (4).

Grosu and Chronopoulos (2002) classifies load-balancing approaches in networks into *global*, *cooperative* and *noncooperative* approaches. In the *global* approach there

[1]If $xf(x)$ is a *convex function* then $f(x)$ is said to be a *semi − convex function*.

is a single decision maker who tries to arrive at a $social\ optimum$. In the $cooperative$ approach, there are several decision makers each trying to optimise the $collective$ delay of its own set of players. This has a structure similar to a cooperative game. A network that adopts a $noncooperative$ approach behaves like a noncooperative game where each player tries to minimise its own load in the given situation. This paper gives an algorithm to balance load in a $cooperative\ game$ where each decision maker places a bid on the amount of load it can handle. The goal is to arrive at a payment mechanism where each agent has to make a $truthful\ bid$ to gain $maximum\ profit$. The algorithm succeeds in arriving at an optimal load distribution by defining an appropriate $payment\ function$. This way the players $profit$ the most by bidding only the true value of the load.

Here $b = (b_1, b_2, ..., b_n)$ is a vector of $bids$ placed, $\Lambda(b) = (\lambda_1(b), \lambda_2(b), ..., \lambda_n(b))$ is the vector of the $output$ loads, $P(b) = (P_1(b), P_2(b), ..., P_n(b))$ is the vector of the $payments$ made to n agents and the $true$ values of load are taken to be t_i. The corresponding cost function is $cost_i(t_i, \lambda(b)) = t_i\lambda_i(b)$ and profit function is $profit_i(t_i, b) = P_i(b) - cost_i(t_i, \lambda(b))$ for agent i. Let b_{-i} represent the vector $(b_1, b_2, ..., b_{i-1}, b_{i+1}, ..., b_n)$.

If $P(b_{-i}, b_i) = b_i\lambda_i(b_{-i}, b_i) + \int_x^\infty \lambda(b_{-i}, x)dx$, then each agent makes a $truthful\ bid$, that is, $t_i = b_i$. Thus, any agent trying to bid a value lower than its true value t will get a $lower\ payment$ and incur a $higher\ cost$ leading to a $lower\ profit$. On the other hand an agent trying to bid a value higher than its true value is bound to get a $lower\ profit$. The $profit\ earned$ reaches its peak only when the user is truthful.

1.1.4 Edge Pricing

Since a $Nash\ equilibrium$ does not lead to the minimum possible latency, $edge\ pricing$ is used to achieve an optimal solution. $Marginal\ cost\ pricing$ is an older method used to eliminate the inefficiency of a $Nash\ equilibrium$. The method asserts that the players should be charged with taxes corresponding to the additional congestion due to their presence. However, this method makes an assumption of $homogeneity$ that all players will apply the same tradeoff between latency and taxes. Cole *et al.* (2003)considers a network with a $heterogeneous$ set of players and devises a pricing mechanism to achieve optimal routing of traffic. Each player is assigned a $tax\ sensitivity$ $\alpha(a)$ and minimum total latency in the network is computed by minimising the sum of $latency$ and $\alpha(a)$ times the tax that player a pays.

1.1.5 Effects of cooperation on Social Cost function

Hayrapetyan *et al.* (2006) quantifies the price of collusion of players in a game. Price of anarchy assumes that the agents act selfishly, and also independently. Collusion might lead to a positive or a negative impact on the social benefit factor. The $price\ of\ collusion$ (PoC) depends upon the disparity in power among the game participants. In $convex\ games$, this quantity could become arbitrarily large. For games with $convex$ latencies, in symmetric nonatomic games the PoC is 1 while in symmetric discrete congestion games it is upper bounded by 2. In games with $concave$ latencies, the PoC is upper bounded by 2 only if a $pure$ equilibrium exists. For $mixed$ equilibria, PoC lies between $8/7$ and 4.

1.1.6 Interesting properties of graphs with equilibria

Bhaskar *et al.* (2009) discusses the existence of multiple equilibria based on the type of graph. The type of game studied is an $atomic\ splittable\ flow\ game$ where each player controls a discrete amount of traffic. An $atomic\ splittable\ flow\ game$ has an $agreeing\ cycle$ if and only if atleast one of the circulations has the same direction as the net circulation along its edges. A game with only two players has a unique equilibrium iff the graph is a $generalized\ series-parallel\ graph$ and a game with multiple players of two types has a unique equilibrium iff the graph is a $s-t-series-parallel\ graph$. For games with multiple types of players a unique equilibrium exists iff the graph is a $generalized\ nearly\ parallel\ graph$. Equilibria are not unique for games in $2-terminal\ nearly\ parallel\ graphs$. A $s-t-series-parallel\ graph$ is a two-terminal graph where the nodes other than the terminals are connected to only two other nodes. A $generalized\ series-parallel\ graph$ does not contain K_4 as its minor. In a $generalized\ nearly\ parallel\ graph$ the components have two nodes which are connected by $vertex-disjoint$ paths. An alternate definition is that a $generalized\ nearly\ parallel\ graph$ contains atleast one $agreeing\ cycle$.

1.2 Problem Statement

We have a data set of link characteristics and input traffic for Chennai city roads. The data shows the existence of 4235 edges and 1569 nodes. Of these 1569 nodes, only 290 nodes/zones (from 929 to 1218) are potential $origins/destinations$. The $potential$ function for this network has been given as,

$Travel\ time$ $=$$Freeflow\ time$ $*(1 + \alpha(v/C)^{\beta})$ where,

$travel\ time$ is the actual time taken to traverse the link, $freeflow\ time$ is the time taken when there is no competition, v is the volume of traffic on the link $vehicles\ per\ hour$ and C is the capacity of the link in the units $vehicles\ per\ hour$.

The objective is to minimise travel time from any origin to any destination. However, both minimising $total\ travel\ time$ and determining $link{-}wise\ traffic\ distributions$ are impossible given the size of the network.

1.2.1 Decluttering the network

This is an image of the entire network as a graph.

Figure 1.1: Original network

In order to prune the network, we classify the links based on their properties.

α, β and C

The network has only 4 sets of (α, β) values: $(1.167, 1.189)$, $(1.549, 1.064)$, $(2.067, 1.33)$ and $(2.152, 1.013)$. Furthermore, there are only 9 values of $Capacity\ C$, $(1400, 1900, 2800, 3400, 3800,$
each falling distinctly into one of the (α, β) categories defined above.

$d(cost)/dv$

Though the above classifications help us divide the network into more manageable sections, the measures in themselves do not have much meaning.

$$\frac{d(cost)}{dv} = (Freeflowtime) * \alpha * \beta * (v/C)^{(\beta-1)}$$

gives us the $marginal\ cost\ function$ for the network for the defined $c(x)$. The $marginal\ cost$ computed at $v = 1$ showed a well-defined partition among the links. The values were either $greater\ than$ 100 or $lesser\ than$ 0.5. From the definition of marginal cost function, it is obvious that the links with the $higher$ marginal cost at $unit\ volume$ are more prone to become $congested$.

The network simplifies to:

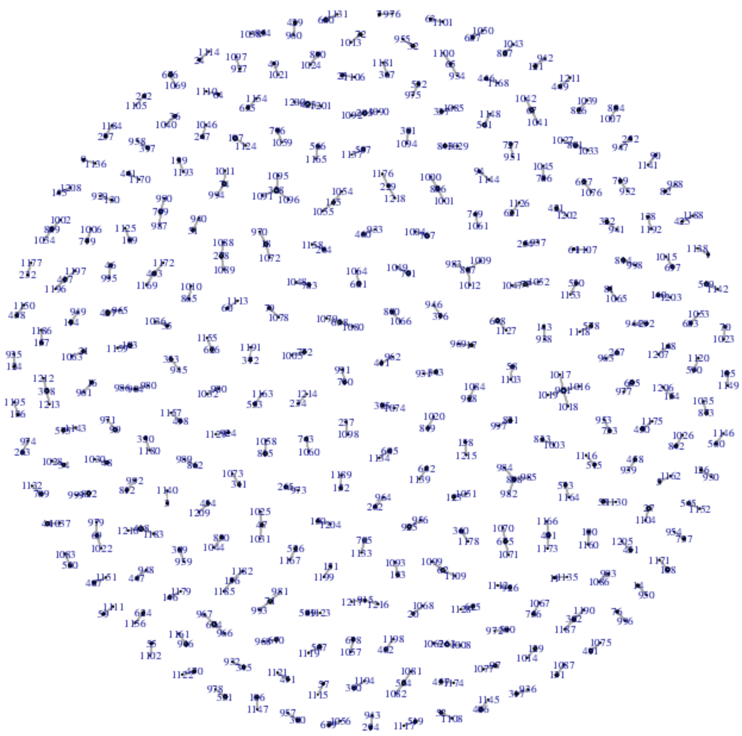

Figure 1.2: Links with greater marginal cost

Making use of this property, the aim is redefined to finding and observing only the $worst - case\ routes$ for each of these 290 origin nodes.

1.2.2 Assumptions

We will be calculating an $approximate$ distribution of traffic under $optimal$ and $nash\ equilibrium$ conditions. We will be making the following assumptions:

1. We are only concerned with reaching the next node on the route at any given point of time.

2. Among all the possible next nodes, we will consider the route with the highest

difference in $volume\ of\ traffic$ under $optimal$ and $nash\ equilibrium$ conditions to be our $worst - case\ route$. This is justified by the argument that with the addition or removal of each unit of traffic this $link$ will be the worst affected. The rest of the terms in the optimisation can be ignored.

3. While calculating the traffic distribution for $nash\ equilibrium$ conditions, we will equate the costs as seen by a single user on each of these competing links based on assumption 1.

4. Once the user reaches the next node on the $worst - case\ route$, we can opt to either,

i) Continue without reestimating the traffic volumes using optimal case to improve the Nash equilibrium routing. We will call this the $Without\ Look\ Back$ method. Or,

ii) Reestimate the traffic volumes as seen in the optimal case before routing in the next node for both optimal and Nash equilibrium cases. We will call this the $Look\ Back\ method$.

1.3 Solution

Based on the structure of the network and the assumptions mentioned above, we can do the following operations at each node.

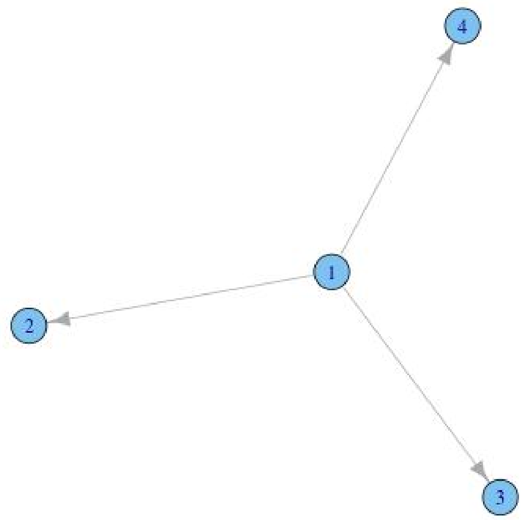

Figure 1.3: Example

Let V be the volume of traffic coming into node 1. Let C_{12},C_{13},C_{14} be the capacities, $v = (v_{12}, v_{13}, v_{14})$ be the traffic volumes after optimal distribution and $v^{'} = (v^{'}_{12}, v^{'}_{13}, v^{'}_{14})$ be the traffic volumes after achieving nash equilibrium, t_{12},t_{13},t_{14} be the freeflow times and $t^{'}_{12}$,$t^{'}_{13}$,$t^{'}_{14}$ be the real travel times for the links $1-2$,$1-3$ and $1-4$ and $v_{12}+v_{13}+v_{14} = V$, $v^{'}_{12}+v^{'}_{13}+v^{'}_{14} = V$. The $minimisedobjective\ function$ is of the form,

$$t = t^{'}_{12} + t^{'}_{13} + t^{'}_{14}$$

$$t = t_{12}(1+\alpha_{12}(v_{12}/C_{12})^{\beta_{12}} + t_{13}(1+\alpha_{13}(v_{13}/C_{13})^{\beta_{13}} + t_{14}(1+\alpha_{14}(v_{14}/C_{14})^{\beta_{14}}$$

Similarly, we will consider only the nearest neighbours for calculating nash equilibrium. Therefore,

$$t_{12}(1+\alpha_{12}(v^{'}_{12}/C_{12})^{\beta_{12}} = t_{13}(1+\alpha_{13}(v^{'}_{13}/C_{13})^{\beta_{13}} = t_{14}(1+\alpha_{14}(v^{'}_{14}/C_{14})^{\beta_{14}}$$

Now the worst-case route will be the one which will show a major variation between the distributions for optimal and nash equilibrium conditions. Thus, we choose the link for which the difference between the vectors v and $v^{'}$ is maximum and proceed further in a similar manner.

For computing traffic distributions $Without\ Looking\ Back$,

$v = findopt(links, linkchar, V_{Opt})$

$v^{'} = findnash(links, linkchar, V_{Nash})$

For computing traffic distributions $While\ Looking\ Back$,

$v = findopt(links, linkchar, V_{Opt})$

$v^{'} = findnash(links, linkchar, V_{Opt})$

Here, the Nash equilibrium traffic distribution is kept updated in the new configuration using $v^{'}$ but the input traffic flow at the next step is always taken from the Optimum traffic distribution, which is updated using v.

The $termination\ condition$ for these routes is that they either $reach$ one of the $zones$ or any $node$ is $repeated$ (leading to $cycles$). Since some links are far more sensitive to changes in traffic volumes, when we solve the above equations we will find that these links are chosen more often as the worst links.

1.4 Results

1.4.1 Graphs

The graphs show the behaviour of the worst-case route starting from one of the 290 $zones$. The X-axis gives the $sequence\ number$ of the intermediate nodes $along\ the\ route$. The Y-axis gives the $steady\ state\ volume$ of $traffic\ inflow$ at the intermediate nodes. Each node also contains a label with the $amount\ of\ traffic\ inflow\ at\ the\ instant\ of\ routing$, $fraction\ of\ traffic\ being\ sent\ through\ the\ worst\ link$ and the $current\ node\ number$.

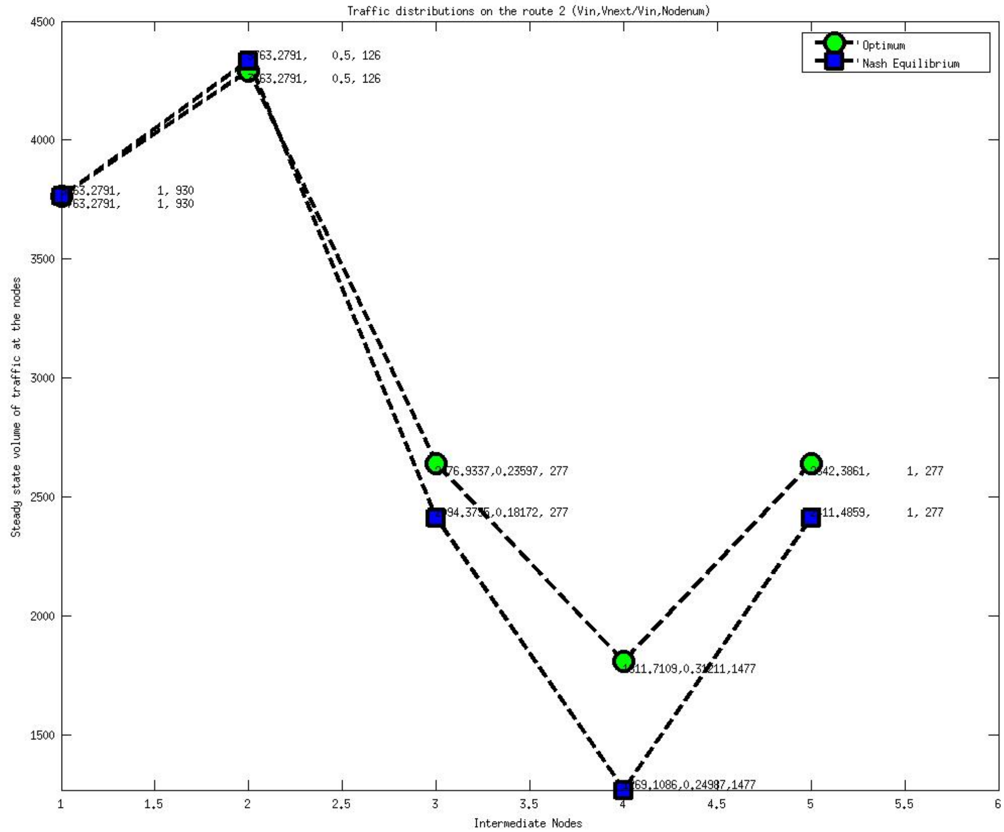

Figure 1.4: Worst-case route for route 269(Without Look Back)

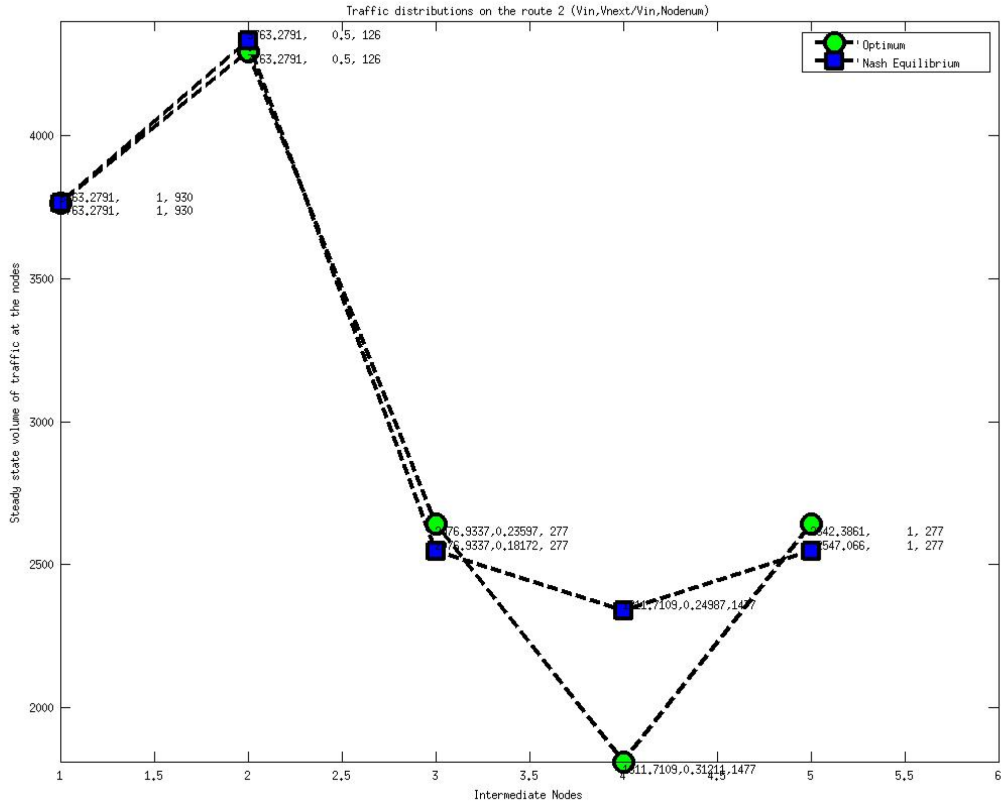

Figure 1.5: Worst-case route for route 269(With Look Back)

1.4.2 Comparison between Without Look Back and Look Back methods

The difference between routes computed without looking back and looking back are as follows:

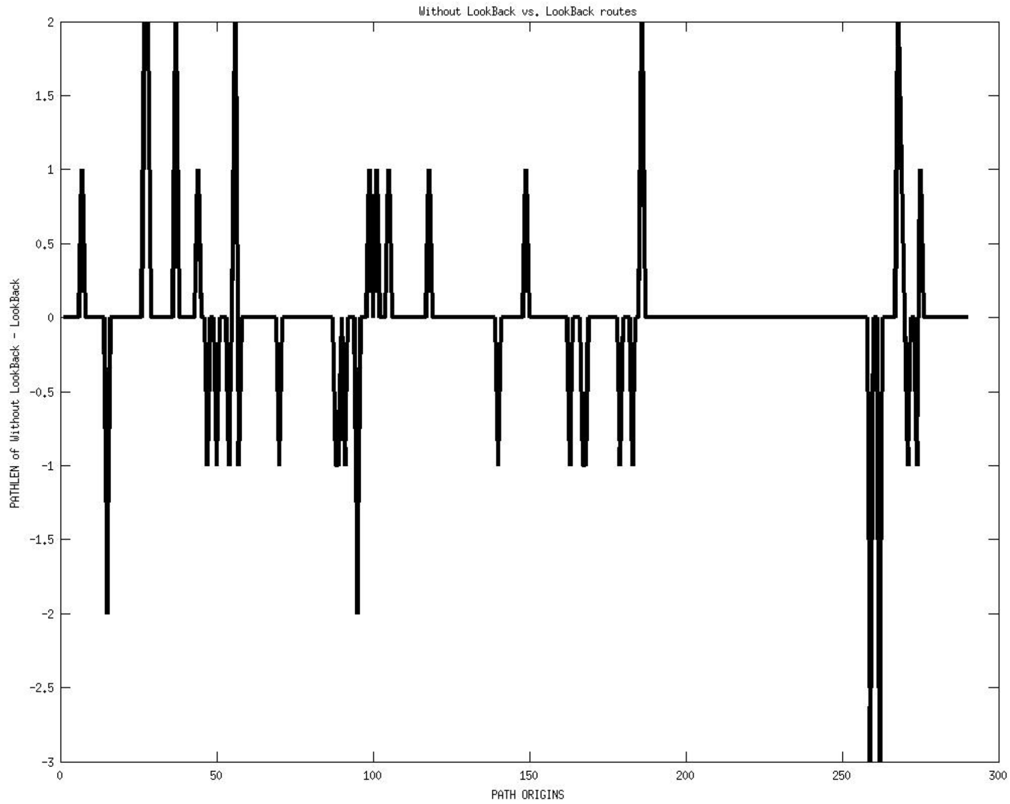

Figure 1.6: Difference in path length

Measuring differences in path lengths is not an accurate method of comparing the two methods as the measure only keeps track of the $number\ of\ links$ traversed. So instead we will look at how the $sum\ of\ minimized\ net\ travel\ time$ taken along each of the 290 routes varies for these two methods.

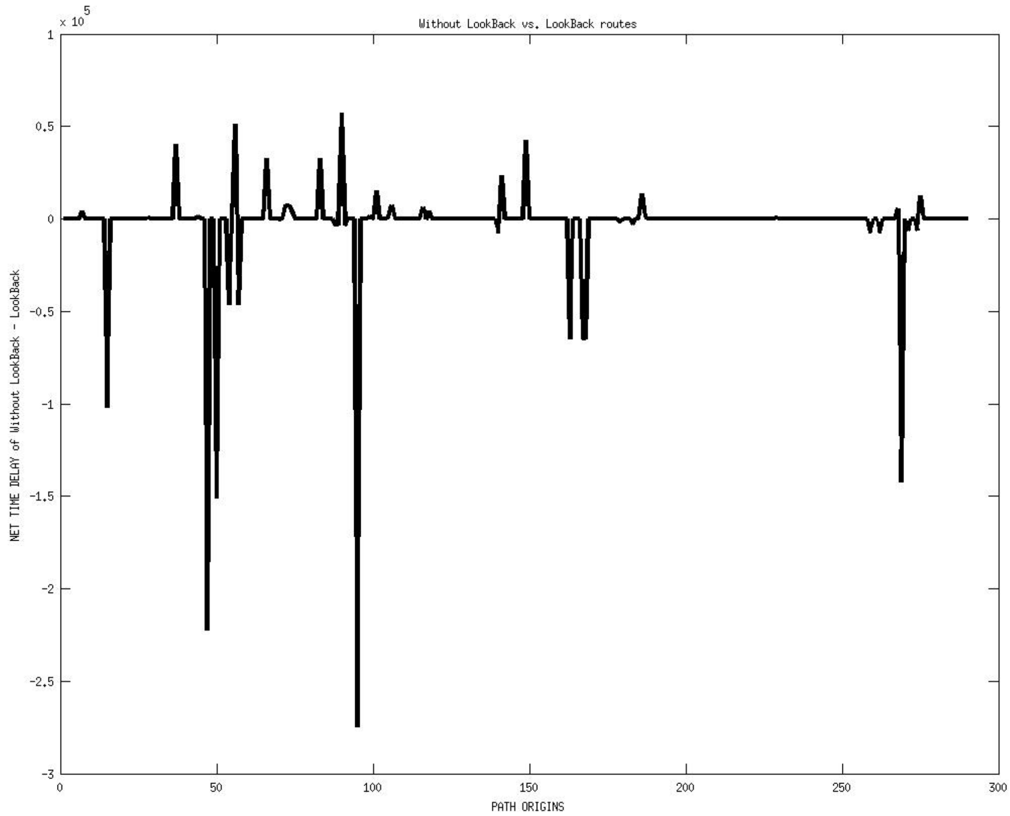

Figure 1.7: Difference between sum of minimised objective functions

It is not intuitive that the LookBack method returns such high values of $cumulative$ $objective\ function$ for certain routes. So to observe the difference we will look at the routes for which the $difference\ in\ cumulative\ objective\ function$ is less than -10000. These routes start at 15,47,50,95,163,167,168 and 269 numbered zones.

1.4.3 Anamolous Routes

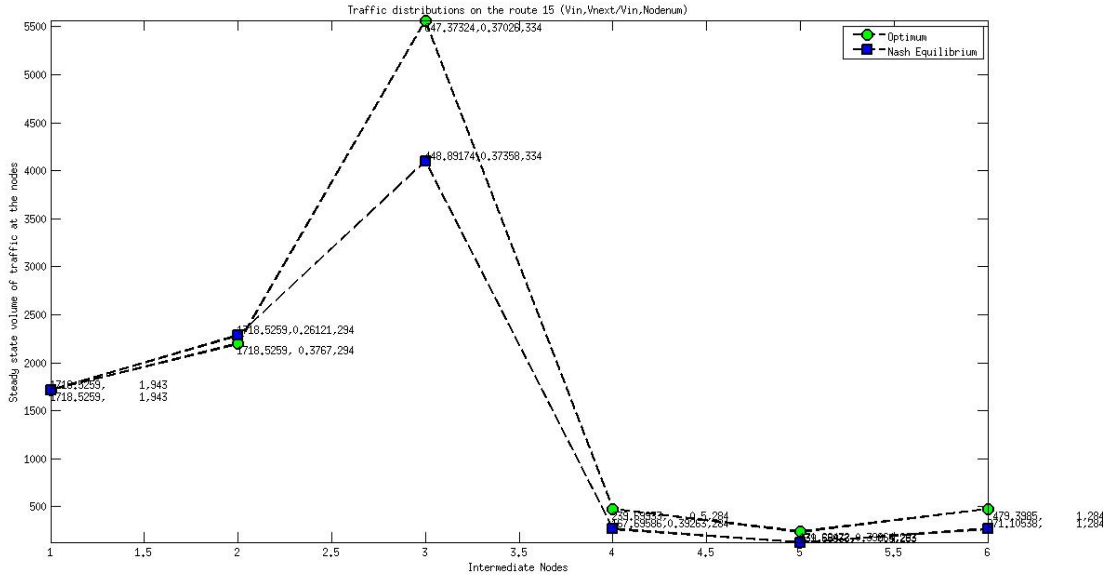

Figure 1.8: Worst-case route for route 15(Without Look Back)

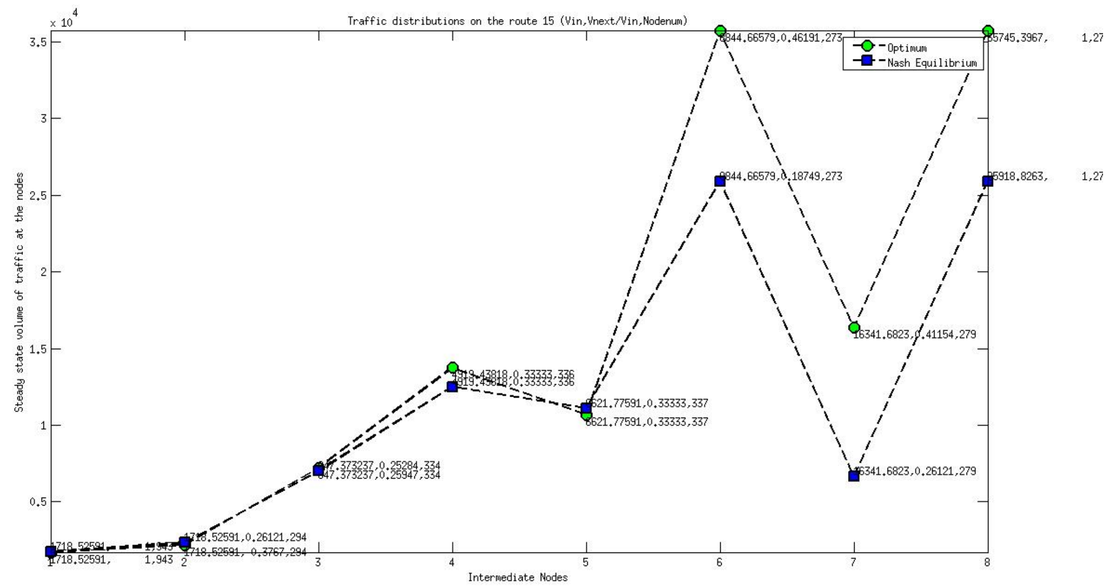

Figure 1.9: Worst-case route for route 15(With Look Back)

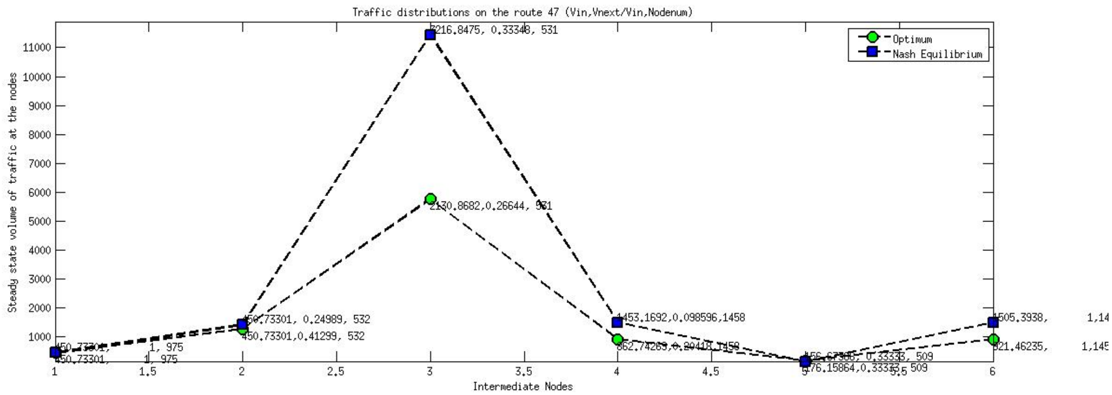

Figure 1.10: Worst-case route for route 47(Without Look Back)

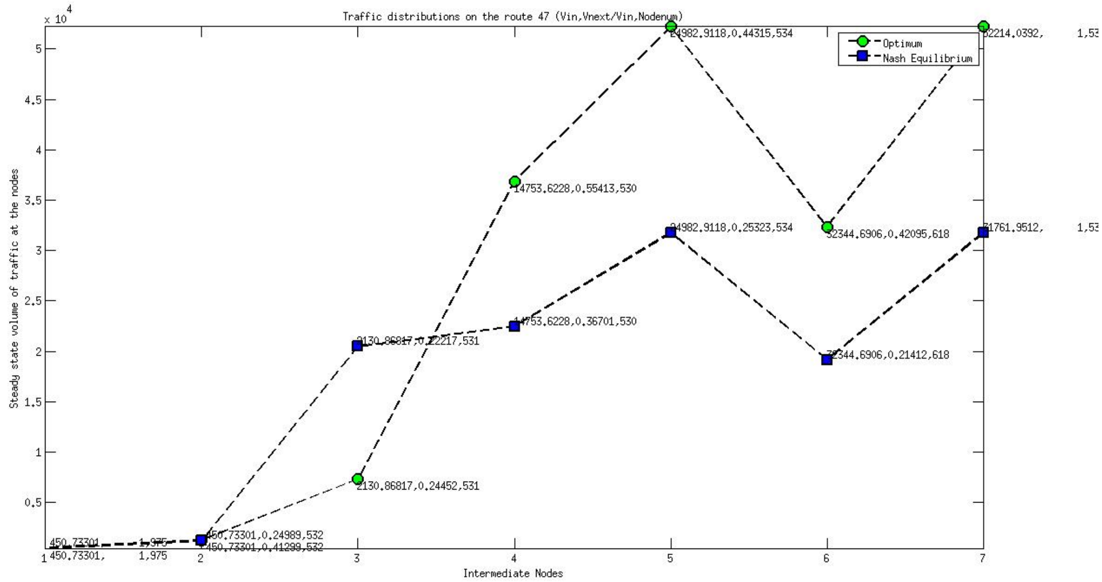

Figure 1.11: Worst-case route for route 47(With Look Back)

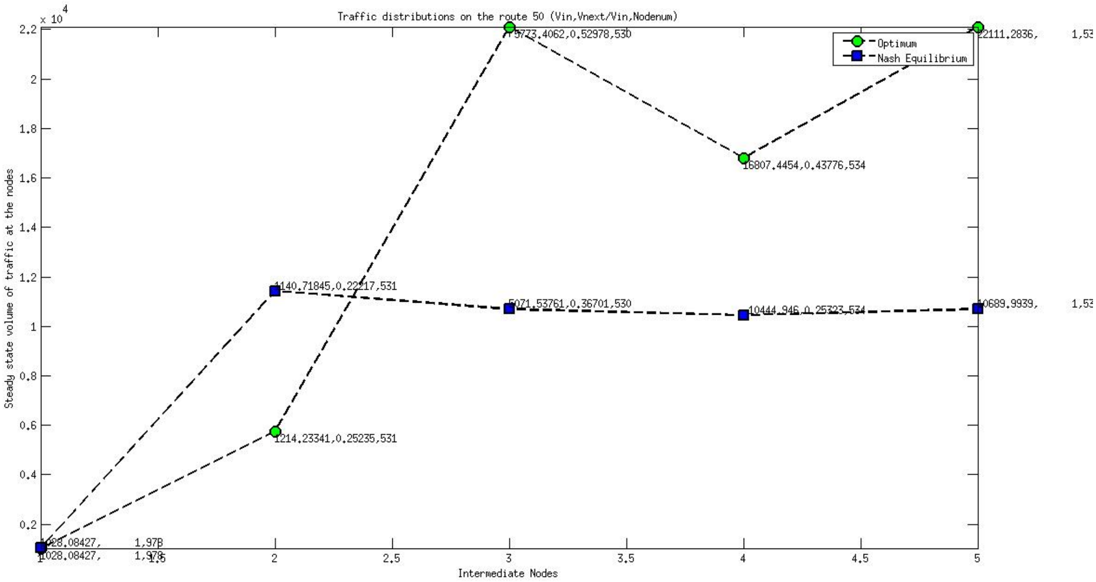

Figure 1.12: Worst-case route for route 50(Without Look Back)

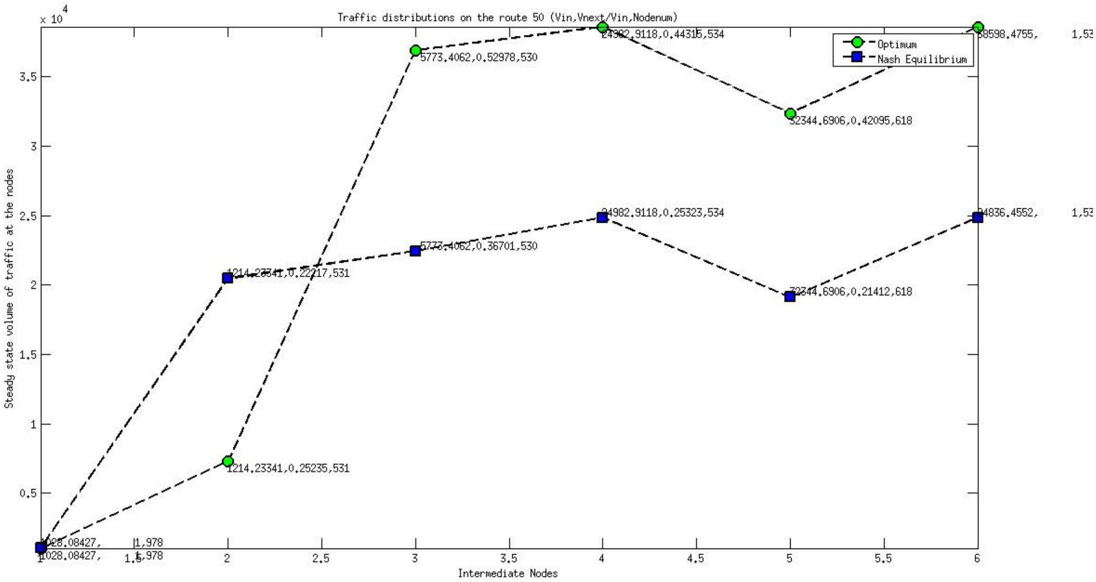

Figure 1.13: Worst-case route for route 50(With Look Back)

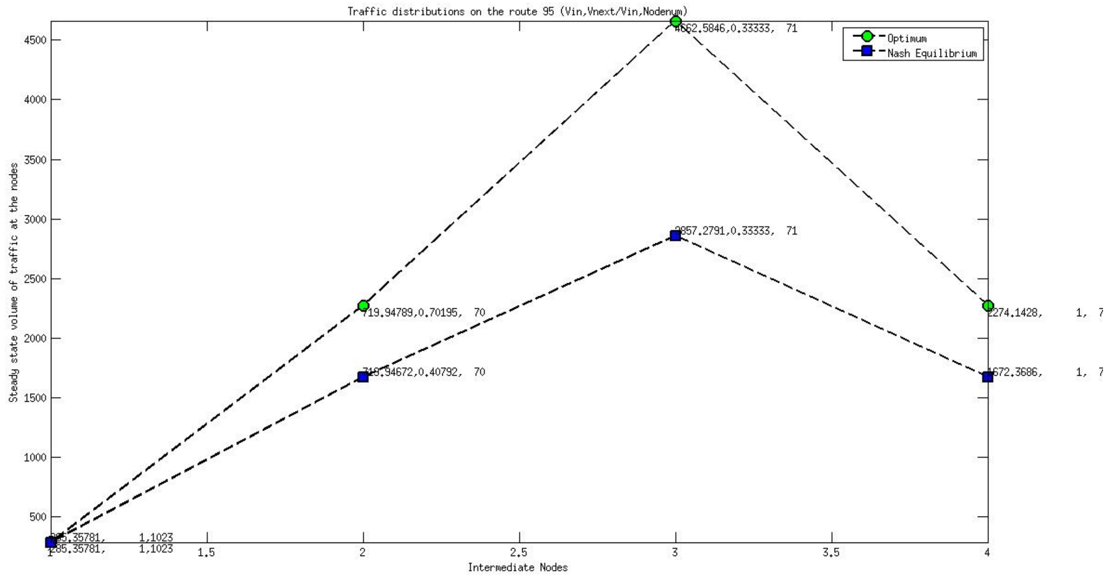

Figure 1.14: Worst-case route for route 95(Without Look Back)

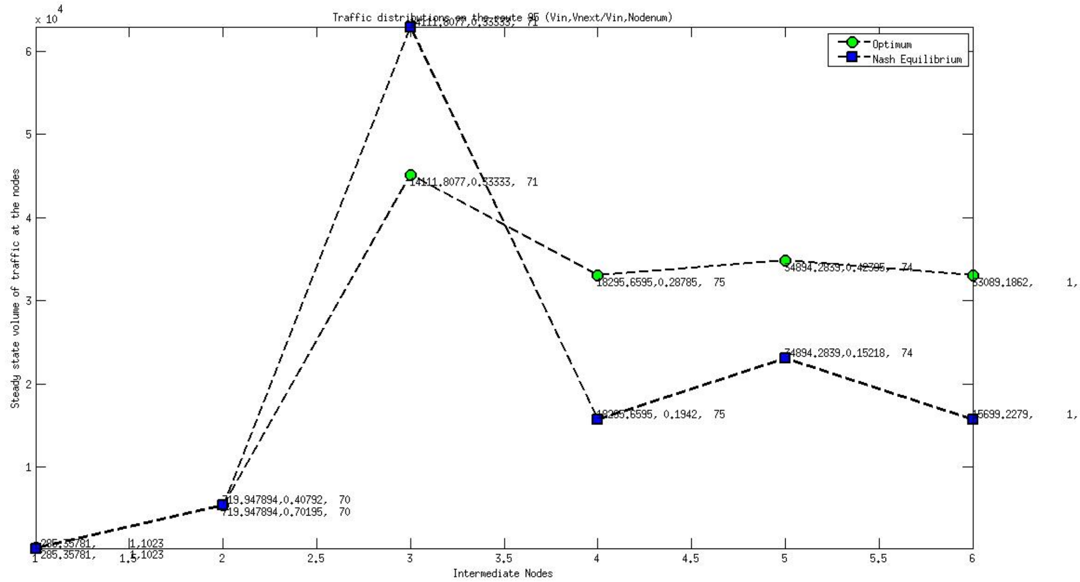

Figure 1.15: Worst-case route for route 95(With Look Back)

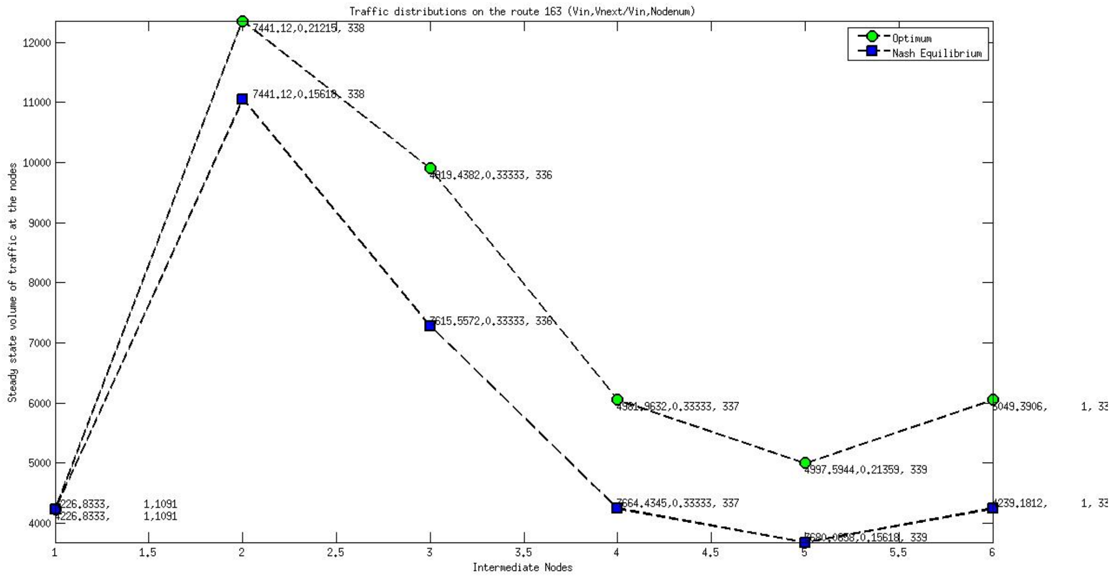

Figure 1.16: Worst-case route for route 163(Without Look Back)

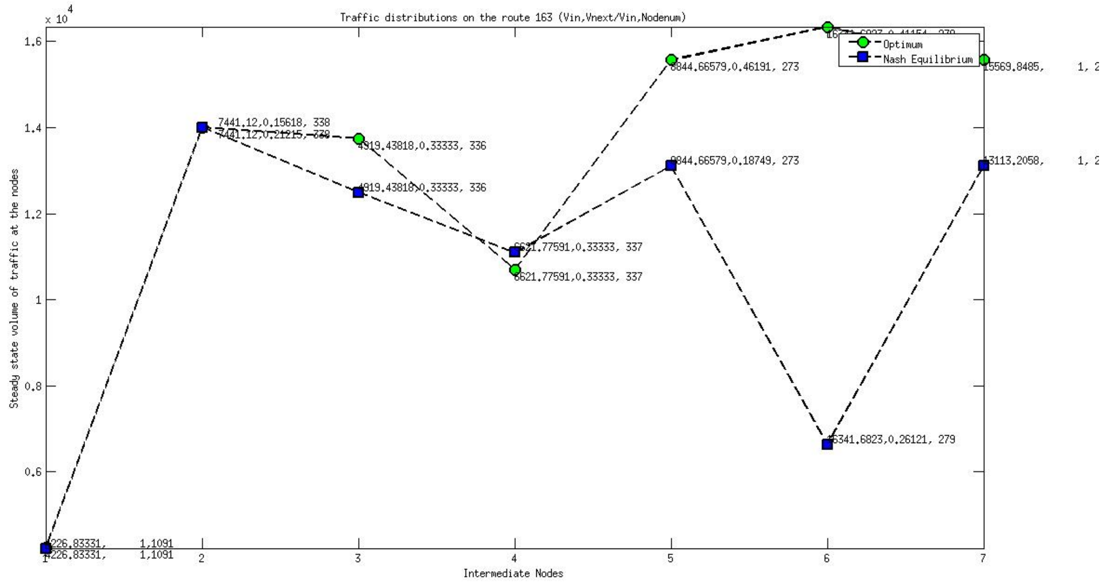

Figure 1.17: Worst-case route for route 163(With Look Back)

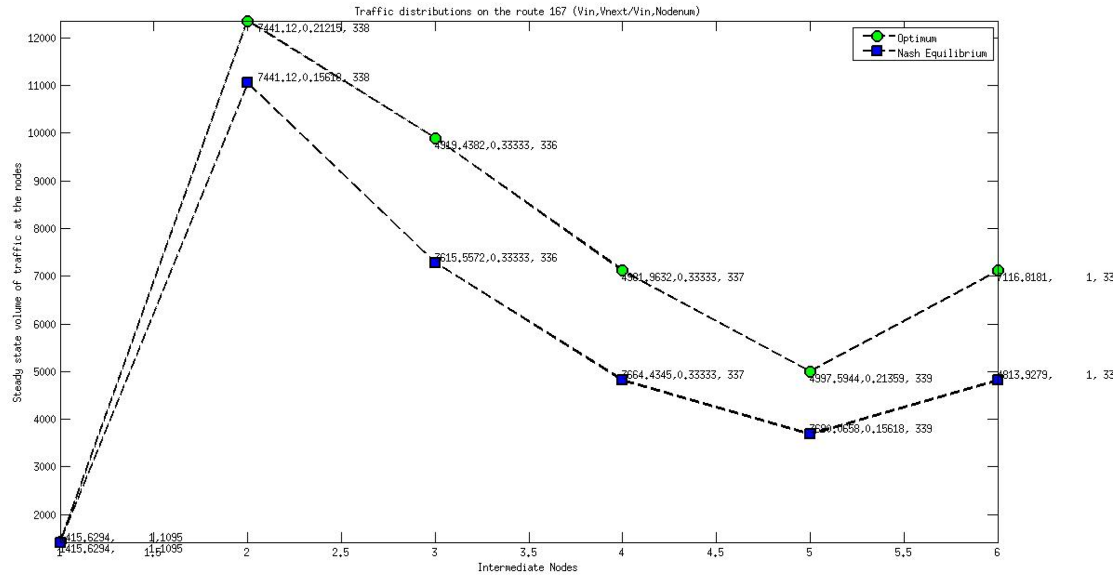

Figure 1.18: Worst-case route for route 167(Without Look Back)

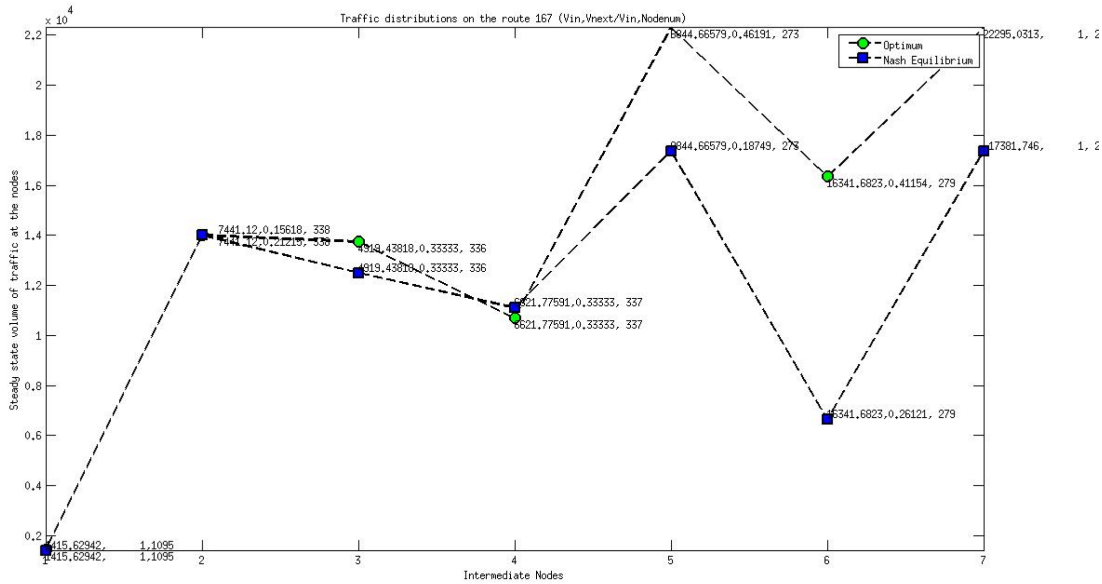

Figure 1.19: Worst-case route for route 167(With Look Back)

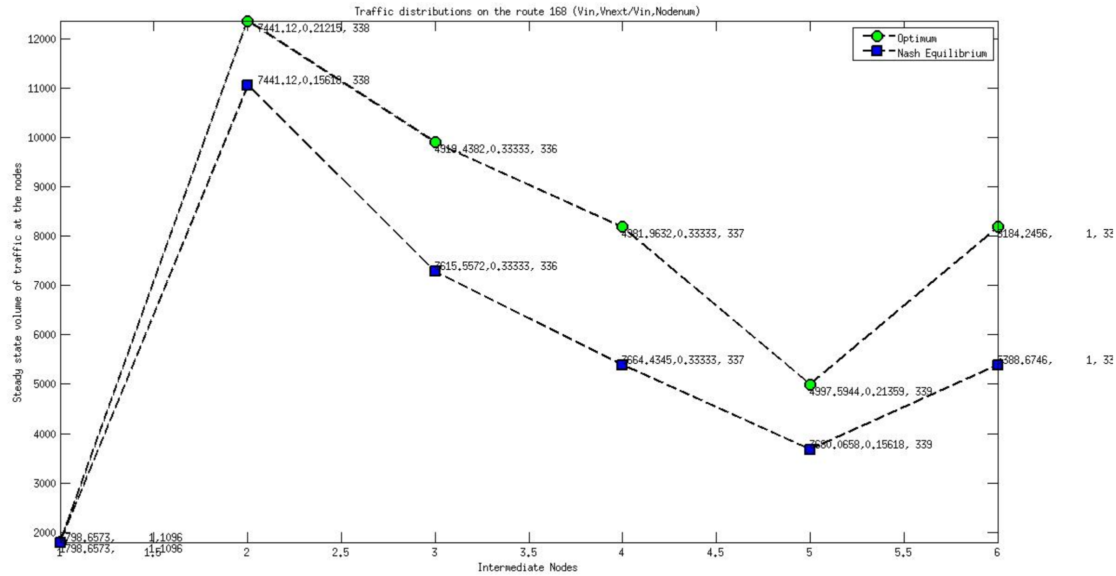

Figure 1.20: Worst-case route for route 168(Without Look Back)

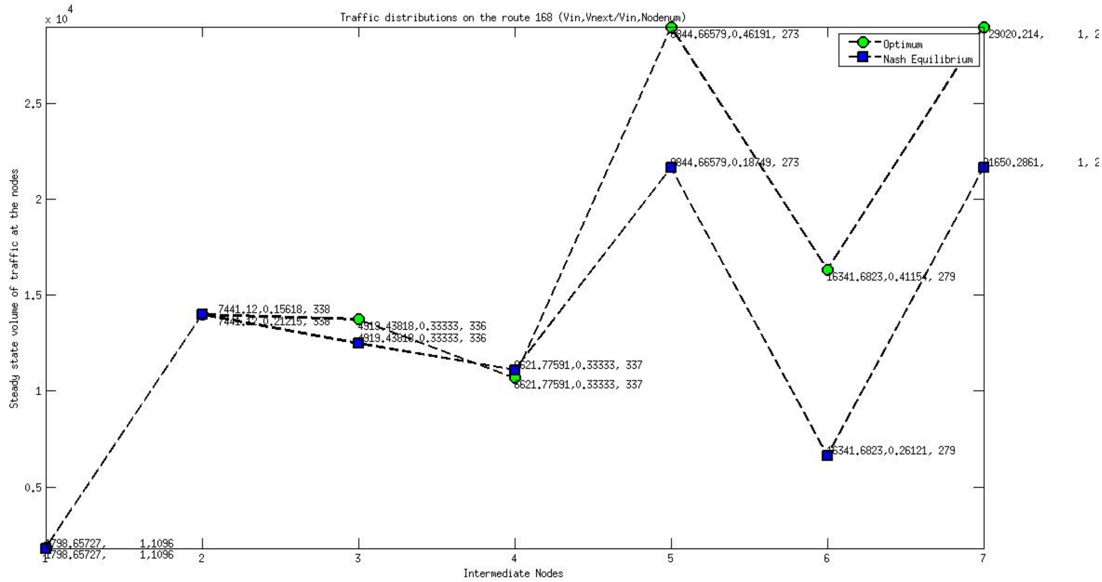

Figure 1.21: Worst-case route for route 168(With Look Back)

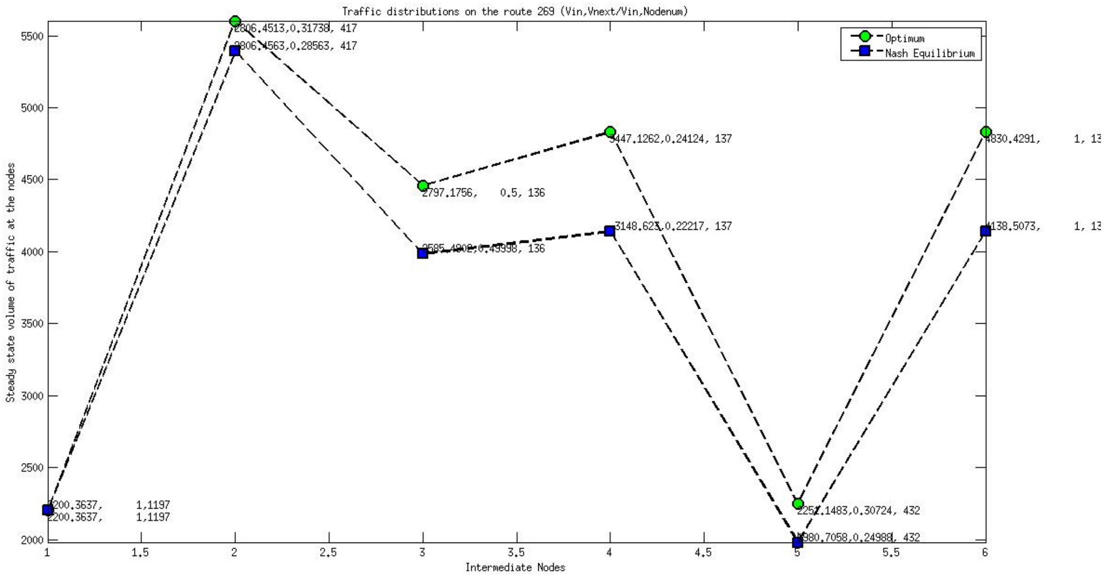

Figure 1.22: Worst-case route for route 269(Without Look Back)

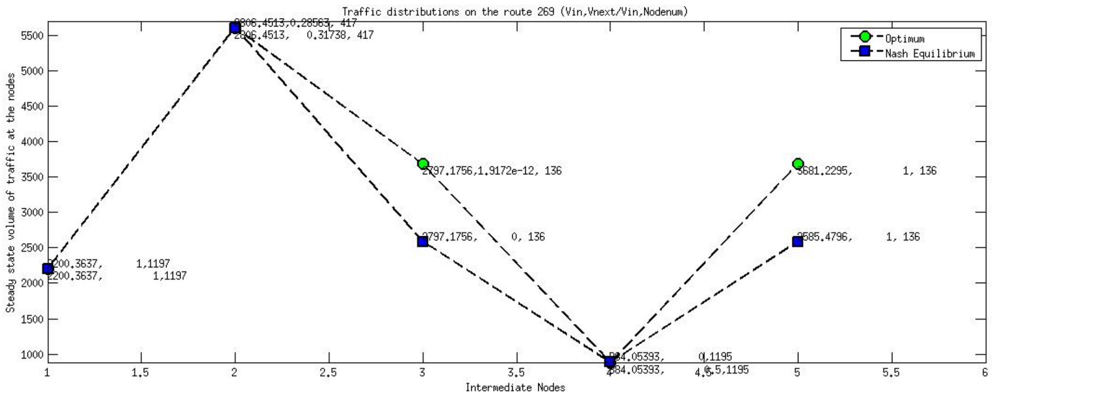

Figure 1.23: Worst-case route for route 269(With Look Back)

1.4.4 Reason for Anomalies

We can clearly see that in each of the above $anamolous\ cases$ the volume of traffic inflow becomes very high starting at some node on the route. There are nodes which accumulate high volumes of traffic on these worst case routes. In these cases, the volumes are higher than the combined capacities of links with lower values of $freeflow\ time$. The free flow travel times of links with $C = 99999$ are close to 167 and when compared with the lower values of $freeflow\ time$ which are close to 1. Due to this while finding $optimal$ and $nash$ solutions most of the traffic is routed through alternate links as the $cost\ function$ for links with $lower$ free flow times remain smaller than those with $higher$ free flow times, even when v becomes much greater than C. And without a constraint on the volume of traffic v allotted to each link, the $cumulative\ sum\ of\ travel\ time$ tends to become very, very high. The failure of the constraint $v/C <= 1$ immediately leads to $greater\ congestion$ along these routes. The reason why certain nodes accumulate more traffic than others seems to be that they are situated directly on the worst case routes for more than one zone and are also $adjacent$ to and thus, indirectly affected by more zones. The table given below demonstrates the same fact for the deviating nodes on the anamolous routes. We can also observe that the $difference$ in minimized objective function for each route varies in $inverse$ relation to the $number\ of\ routes$ sharing these high volume nodes and varies $direct\ relation$ to the $number\ of\ zones$ contributing to the traffic.

end node	start node	length	speed	traffic at origin	difference in sum of objective function	affected routes
71	981	3.12023	0.0187214	651.88513	-274670	95
71	993	1.44513	0.00867078	0	-274670	95
74	994	0.635915	0.00381549	95.73896	-274670	95
74	1011	0.928296	0.00556977	2328.38148	-274670	95
417	1196	0.520528	0.00312317	164.06063	-142390	269
417	1197	0.342551	0.00205531	2200.36371	-142390	269
338	1091	1.97845	0.0118707	4226.83331	-101800 -65060 -65060 -65060	15 163 167 168
338	1095	1.19495	0.00716973	1415.62942	-101800 -65060 -65060 -65060	15 163 167 168
338	1096	0.797588	0.00478553	1798.65727	-101800 -65060 -65060 -65060	15 163 167 168
534	1081	0.905861	0.00543517	3010.62288	-222400 -150720	47 50
534	1082	0.280404	0.00168242	10.35431	-222400 -150720	47 50

Table 1.1: Link characteristics for selected routes

NOTE: The initial node IDs given in the table are zones (in the 929 to 1218 range)

REFERENCES

1. **Bhaskar, U.**, **L. Fleischer**, **D. Hoy**, and **C.-C. Huang**, Equilibria of atomic flow games are not unique. *In Proceedings of the twentieth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '09. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2009. URL `http://dl.acm.org/citation.cfm?id=1496770.1496852`.

2. **Cole, R.**, **Y. Dodis**, and **T. Roughgarden**, Pricing network edges for heterogeneous selfish users. *In Proceedings of the thirty-fifth annual ACM symposium on Theory of computing*, STOC '03. ACM, New York, NY, USA, 2003. ISBN 1-58113-674-9. URL `http://doi.acm.org/10.1145/780542.780618`.

3. **Grosu, D.** and **A. T. Chronopoulos**, Algorithmic mechanism design for load balancing in distributed systems. *In Proceedings of the IEEE International Conference on Cluster Computing*, CLUSTER '02. IEEE Computer Society, Washington, DC, USA, 2002. ISBN 0-7695-1745-5. URL `http://dl.acm.org/citation.cfm?id=792762.793282`.

4. **Hayrapetyan, A.**, **E. Tardos**, and **T. Wexler**, The effect of collusion in congestion games. *In Proceedings of the thirty-eighth annual ACM symposium on Theory of computing*, STOC '06. ACM, New York, NY, USA, 2006. ISBN 1-59593-134-1. URL `http://doi.acm.org/10.1145/1132516.1132529`.

5. **Roughgarden, T.** and **E. Tardos** (2002). How bad is selfish routing? *J. ACM*, **49**(2), 236–259. ISSN 0004-5411. URL `http://doi.acm.org/10.1145/506147.506153`.

6. **Roughgarden, T.**, **E. Tardos**, and **ÃL'va Tardos** (2002). Bounding the inefficiency of equilibria in nonatomic congestion games. *Games and Economic Behavior*, **47**, 389–403.