IDENTIFYING NETWORK LOAD BALANCING REQUIREMENTS ON HISTORICAL TRAFFIC FLOW USING MACHINE LEARNING APPROACH FOR FLEXIBLE MULTIPATH NETWORKS

K BALAMURUGAN [1]

Assistant Professor – CSE, Adhiyamaan College of Engineering,
Hosur, Tamil Nadu, India

Dr.V.PALANISAMY [2]

Principal, Info Institute of Engineering & Technology,
Coimbatore Tamil Nadu, India.

Abstract

Multipath routing uses several paths to distribute traffic from a source to destination. This not only improves network performance but also achieves load balancing and fault tolerance. Though multipath routing is not deployed widely in the internet, current systems chooses the paths with the same lowest cost to the destination and the same administrative distance.

In this paper we propose a flexible multipath network wherein multipath routing algorithm is invoked when the quality of service is affected. Our proposal provides a method to gradually shift from a single path existing network to a reliable multipath network in the future.

We have used machine learning approach to identify the instance when network load balancing is required with good results.

Keywords **:** Network load balancing, Multipath routing, Quality of service, Neural Network, Naïve Bayes.

1.Introduction

1.1 Network Load balancing

Load balancing is an important factor in multipath communication networks to optimize bandwidth in this modern era of communication. Now-a-days, the convergence of the computer, communications, entertainments and consumer electronics industry is driving an explosive growth in multimedia applications [1]. Now ISPs are confronted to provide the quality of service (QoS) due to the huge development of internet based multimedia applications. To meet this capacity expansion one efficient solution is to install new links in parallel with existing ones. This requires an effective approach for routing and distributing huge volume of traffic through a set of parallel links. There are some unipath routing protocols [1][2] which can adapt very quickly to changing network conditions but they become unstable under heavy load and bursty traffic conditions and at any given time some subnets can be heavily congested whereas others remain under-utilized.

Over the last decade-and-a-half, Internet use has dramatically increased, placing an extraordinarily high level of demand on underlying hardware. In order to keep up with the increase in user requests and preclude saturation of hardware resources, the hardware itself has become much more powerful and capable. However, the key to successfully serving a customer base that continues to grow is recognition that the solution cannot be achieved merely by investing large sums of money in the latest and greatest hardware. Rather, the answer lies in an understanding of how the network can be used to your advantage and how you can distribute requests to many servers within a cluster that can then process them in an expeditious manner. This concept, aptly called load balancing, is neither complex nor novel, and appropriately used, it can help ensure that no server becomes so overburdened with requests that it ends up failing to properly function. Load balancing has been around for years. Some load balancing implementations have been hardware-based while others only required installation of special software.

Network Load Balancing (NLB) is protocol agnostic. It works with any TCP or UDP application-based protocol. This means that you can configure a variety of NLB clusters within your organization, and each cluster can have its own specific function. For example, one cluster may be dedicated to handling all Internet-originated HTTP traffic while another may be used to serve all intranet requests. If your employees have a need for transferring files, you can centralize file storage and closely monitor both uploads and downloads by creating a FTP cluster. Lastly, if there’s a requirement for secure remote access to the corporate network, NLB supports the PPTP protocol which can

be used by employees to establish a Virtual Private Network (VPN) connection. NLB support for PPTP is significantly less costly than many other alternative VPN solutions.

1.2 Approaches To Achieve Network Load Balancing

Round Robin Dns (Rrdns)

Prior to the advent of NLB, Round Robin DNS was used to manage server congestion. With Round Robin, a DNS server contains multiple "A" records for a single host, e.g., the Internet resource www.auerbach-publications.com might correspond to three Internet Protocol (IP) addresses: 208.254.79.10, .11, and .12. The machines with these IP addresses are all identically configured – each is running a web server that has a complete copy of the Auerbach Publications Website, so no matter which server a request is directed to, the same response is provided. This elementary "load balancing" mechanism works as soon as a DNS query is made. When a client attempts to access the Website, a local DNS lookup is performed to determine what the corresponding IP address is. The first time this query is made, the remote DNS server returns all the address records it has. The local DNS server then determines what address record to return to the client. If all records are returned, the client will take the first one that it is given. With each request, the Round Robin algorithm rotates the order in which the address records are returned, so each DNS query will result in a client using a different IP address. When the fourth query is made, the address records are returned in the same order as the first.

Weighted Fair Routing Approach (WFR)

The two most common internet transport protocols are TCP and UDP. Each TCP connection requires its packets to arrive at the destination in order. If a TCP connection, routes packets on multipath simultaneously, those packets sent on different paths may arrive at the destination out of order. Packet based load sharing approaches may not work well for TCP flows and other connection oriented flows that requires packets at the destination in order. Yet, a call-based multiple path routing approach can be applied for load sharing. A UDP connection or any other connectionless traffic allows packets to arrive at the destination out of order, without affecting protocol performance.

Because of the above requirement, a load sharing approach, called Weighted Fair Routing (WFR) has been proposed [3][4][5]. The packet by packet WFR (PWFR) is a packet level WFR in which a set of packets is split on a set of outgoing channels or links and sent the packet as a whole whereas the call by call WFR
(CWFR) is a call-connection level WFR in which a set of connections is split on a set of outgoing channels and all packets belonging to the same connection are routed on the same path.

Packet Based WFR (PWFR)

Suppose there is a sequence of packet, namely, packet1, packet2 …, to be split on a set of N paths or channels. Denote the size of packet k by S(k) bytes. The routing weight for path i is given as pi, where

$$\sum_{i=1}^{N} Pi = 1$$

Define the routing weight vector as P = (p1 p2 … pN) and assume *wpi* and *wpi* ˆ be the expected and actual workload in byes to be sent on path i.

A metric is introduced to measure the traffic under load on a path. The residual workload of path i, where i = 1, 2… N, just before the routing decision for packet k is made, Ri p (K) is defined as the amount of work that should be fed on path i in order to achieve the expected workload Wi P(K). According to [3][5]

1.3 Data Mining

We are in an age often referred to as the information age. In this information age, because we believe that information leads to power and success, and thanks to sophisticated technologies such as computers, satellites, etc., we have been collecting tremendous amounts of information. Initially, with the advent of computers and means for mass digital storage, we started collecting and storing all sorts of data, counting on the power of computers to help sort through this amalgam of information. Unfortunately, these massive collections of data stored on disparate structures very rapidly became overwhelming. This initial chaos has led to the creation of structured databases and database management systems (DBMS)[6].

The efficient database management systems have been very important assets for management of a large corpus of data and especially for effective and efficient retrieval of particular information from a large collection whenever needed. The proliferation of database management systems has also contributed to recent massive gathering of all sorts of information. Today, we have far more information than we can handle: from business transactions and scientific data, to satellite pictures, text reports and military intelligence. Information retrieval is simply not enough anymore for decision-making. Confronted with huge collections of data, we have now created new needs to help us make better managerial choices. These needs are automatic summarization of data, extraction of the "essence" of information stored, and the discovery of patterns in raw data[6].

1.4 Naïve Bayes classifier

An NB classifier is a simple probabilistic classifier based on applying Bayes theorem with strong (Naïve Bayes) independence assumptions. A more descriptive term for the underlying probability model would be independent feature model. In mathematics, a classifier is a mapping from a discrete or continuous feature space X to a discrete set of labels [7]. Classifiers have practical applications in many branches of science and society. Depending on the precise nature of the probability model, NB classifiers can be trained very efficiently in a supervised learning setting. In many practical applications [8], parameter estimation for NB models uses the method of maximum likelihood; in other words, one can work with the Naïve Bayes NB model without believing in Bayesian probability or using any Bayesian methods.

Supervised learning is a machine learning technique for creating a function from training data. Maximum Likelihood Estimation (MLE) is a popular statistical method used to make inferences about parameters of the underlying probability distribution from a given dataset. Bayesian probability is an interpretation of probability suggested by Bayesian theory, which holds that the concept of probability can be defined as the degree to which a person believes a proposition. In simple terms, an NB classifier assumes that the presence or absence of a particular feature of a class is unrelated to the presence or absence of any other feature, [8].

In spite of their naïve design and apparently over-simplified assumptions, NB classifiers often work much better in many complex real-world situations than one might expect. Recently, careful analysis of the Bayesian classification problem has shown that there are some theoretical reasons for the apparently unreasonable efficacy of NB classifiers. An advantage of the NB classifier is that it requires a small amount of training data to estimate the parameters means and variances of the variables necessary for classification. The advantage of Bayesian models is that various important but non-measurable factors such as software complexity, architecture, quality of V and V activities, and test coverage are easily incorporated in the model. The data then can be classified by the attempts made [9].

1.5 Neural network

Neural networks consist of multiple layers of computational units, usually interconnected in a feed-forward way. Each neuron in one layer has directed connections to the neurons of the subsequent layer. In many applications the units of these networks apply a sigmoid function as an activation function.

The feed-forward neural network was the first and arguably simplest type of artificial neural network devised. As the majority of faults are found of its modules, there is a need to investigate the modules that are affected severely as compared to other modules and proper maintenance has to be done on time, especially for the critical applications (Ardil et al.2009). This chapter investigates classification method based on neural network methods with data aggregation using normal density methods to test for goodness of fit and randomness.

2. Goal

In this paper we propose a flexible multipath routing system which can be implemented over the single path existing routing mechanism when the quality of service decreases. The challenge in deploying the above system is the capability of the network to analyse itself and make a decision as when Network Load Balancing is required. To achieve this we have used machine learning approach.

We propose to create an anonymized dataset of packets sent from a server in India to an server in US. The route, number of hops, delay for both incoming and outgoing traffic are captured approximately every thirty minutes for a period of fifteen days. Total of over 42000 packets were sent and received. Packet drops and duration when the quality of services is poor are identified and trained to the machine learning algorithm. The machine learning algorithm was subjected to a test set to identify its accuracy in classification.

3. Experimental Investigation and Analysis

A total of 42840 instances were used for investigation. Seven attributes including delay, number of hops were considered. The analysis of our results is given below.
Instances: 42840

Naïve Bayesian Classification

Table 1 lists the normal distribution of the attributes and Table 2 lists the classification accuracy of Naïve Bayes classifier.

Naïve bayes classifier	**Class bal**	**Class nbal**
Delay outgoing: Normal Distribution.		
Mean	148.5487	163.0176
StandardDev	0.9471	32.6334
WeightSum	37863	4977
Precision	0.401086957	0.401087
Nhops outgoing: Normal Distribution.		
Mean	12.007	13.0904
StandardDev	0.5	2.3087
WeightSum	37863	4977
Precision	3	3
Delay incoming: Normal Distribution.		
Mean	139.7846	144.9375
StandardDev	2.4437	21.5619
WeightSum	37863	4977
Precision	0.245473496	0.245473
Nhops incoming: Normal Distribution.		
Mean	17	16.5696
StandardDev	1.4167	2.6704
WeightSum	37863	4977
Precision	8.5	8.5
Route incoming: Normal Distribution.		
Mean	34486973.53	33798779
StandardDev	279217.7521	5459748
WeightSum	37863	4977
Precision	763679.8478	763679.8
Route outgoing: Normal Distribution.		
Mean	33897047.87	34138428
StandardDev	204929.7071	495048.7
WeightSum	37863	4977
Precision	59554.05882	59554.06

Table 1 : Normal distribution values of the attributes

Correctly Classified Instances	97.7619%
Incorrectly Classified Instances	2.2381%
Kappa statistic	0.8798
Mean absolute error	0.0241
Root mean squared error	0.1437
Relative absolute error	11.78%
Root relative squared error	45.24%

Table 2: The classification results by Naïve Bayes Classifier.

=== Neural Network Based Classification ===
Table 3 lists the classification accuracy by Neural network. For the dataset we obtained Bayes algorithm shows a relative improvement in classification over Neural network methods.

Correctly Classified Instances	93.3544 %
Incorrectly Classified Instances	6.6456 %
Kappa statistic	0.575
Mean absolute error	0.0665
Root mean squared error	0.2578
Relative absolute error	32.50%
Root relative squared error	81.16%

Table 3: Classification by Neural Network

Figure 1 plots the kappa statistics for various parameters including TP rate, FP rate, Precision, Recall and F measure for each class label and the classification method used.

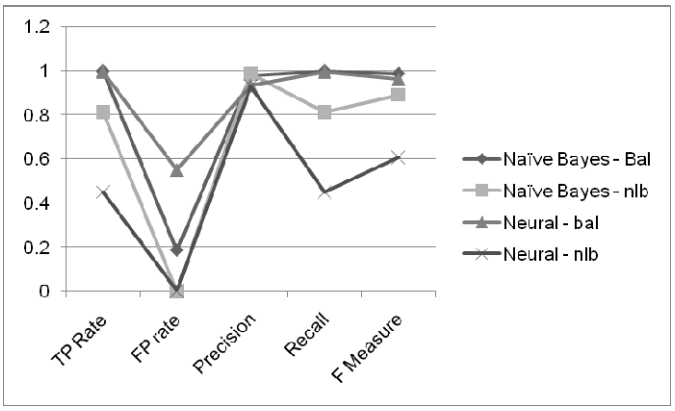

Figure 1 : The Kappa Statistics

4.Conclusion

In this paper we propose a flexible multipath network wherein multipath routing algorithm is invoked when the quality of service is affected. It provides a method to gradually shift from a single path existing network to a reliable multipath network in the future.

Network Load Balancing has also been addressed through mathematical models and realized the improvement in overall performance.

References

[1] V. O. K. Li and W. Lio. "Distributed Multimedia Systems", Proc. of the IEEE, Vol. 85, No. 17, pp. 1063-1108, July 1997.
[2] S. Vutukury and J. J. Garcia-Luna-Aceves. "A Simple Approximation to Minimum-Delay Routing", Computer Cornmunication Review, Vol. 29, No. 4, October 1999.
[3] Han J. and M. Kamber, Data Mining: Concepts and Techniques, 2nd edition. Morgan Kaufmann,
[4] K.C. Leung and V.O.K Li, "Generalized Load Sharing for Packet Switching Network", Proc.IEEE Int'l conf. Network Protocols, pp. 305-314, Nov 2000.
[5] K.C. Leung and V.O.K Li, "Generalized Load Sharing for Packet Switching Network II: Flow Based Algorithms", IEEE transaction Parallel and distributed systems, vol 17, no 7.
[6] K.C. Leung and V.O.K Li, "Generalized Load Sharing for Packet Switching Network I: Theory and Packet Based Algorithm", IEEE transaction Parallel and distributed systems, vol 17, no 7, pp. 694-702, July 2006.
[7] Chotirat, Ratanamahatana, Dimitrios and Gunopulos (2003), 'Scaling up the Naive Bayesian Classifier', Computer Science Department, University of California Riverside, CA 92521 1-909-787-5190.
[8] Yoshimasa Tsuruoka and Junichi Tsujii (1999), 'Training a Naive Bayes Classifier via the EM Algorithm with a Class Distribution Constraint', Department of Computer Science, University of Tokyo, Japan, Vol. 4, N0. 7, pp.456 – 478.
[9] Osmar R. Zaïane, 1999 Principles of Knowledge Discovery in Databases.
[10] Yoshimasa Tsuruoka and Junichi Tsujii (1999), 'Training a Naive Bayes Classifier via the EM Algorithm with a Class Distribution Constraint', Department of Computer Science, University of Tokyo, Japan, Vol. 4, N0. 7, pp.456 – 478.
[11] K.C. Leung and V.O.K Li, "Generalized Load Sharing for Packet Switching Network I: Theory and Packet Based Algorithm", IEEE transaction Parallel and distributed system