Speculative Parallelism Techniques for the Flow Analysis and Congestion Avoidance using Hadoop Framework

Suma S
Research Scholar, Dept.of MCA,
DSCE, Bangalore, India

Rakshitha Kiran P
Research Scholar, Dept.of MCA,
DSCE, Bangalore, India

***Abstract*— Speculative parallelism techniques are becoming mainstream technologies in the high performance computations and in managing the high volume of data created in today's world through predicting the values ,if the prediction goes correct, the techniques are incorporated otherwise it squashes the prediction allows for the normal execution. The packets that flow from different producers to multiple consumers routed though a new proposed routing technique Speculative value prediction routing for the flow analysis and congestion avoidance along with the Hadoop framework. In this paper a novel technique is proposed for the performance analysis of the flow control of packets in the network. The experimental results shows the performance of this algorithm over the network of 4 producers and 4 consumers. The results are promising and efficient.**

Keywords— Speculative value prediction, flow analysis, congestion, Hadoop framework, producers, consumers.

IV. INTRODUCTION

The Speculative Parallelism reduces the Instruction level parallelism and thread level speculation to avoid dependencies in the high performance computations and big data analytics. The approach is to improve performance exploiting the instruction level parallelism through speculative value prediction. The data produced by the streams is structured or unstructured that needs storage system to fetch, gather, analyse and process the data. The online social networking sites like facebook, twitter generates huge amount of data each day. To perform distributed processing of the large volume of data Hadoop uses data sets and traces of data to provide efficient output.

The flow analysis of the network traffic cluster for pattern and sequences of the traffic by providing information to the network controllers to understand the type of network, its usage and behaviour of the network.

David Kaeli and pen-chung yew [8] describes that the data value speculation consists of mechanisms for the prediction, verification and recovery in step by step procedure. without value speculation, the dependent instruction executes in series and requires several cycles to execute. Based on the correct value prediction, it increases the instruction level parallelism and improves the performance.

Hadoop framework is used for the flow analysis and congestion control over the big data. The Hadoop cluster is used for determining the network traffic flow by using two functions Mao and Reduce. It accepts the text file as input and generates the output in another text file used for the flow control of the packets in the network [7].

II. LITERATURE SURVEY

M.Yu and et.al [1,5] proposes the state of enterprise network about the analysis done on the passive and active techniques for communication within the network. J.Shaffer and et.al [2,6] proposed a technique for analysis of traffic with TCP connection as the traffic classification on the fly. The other authors T.Benson and et.al [3,4] deals with the network traffic characteristics of data centres in the WILD analysis of the data sets from data centres for their flow level and packet level.

The memory systems are designed to stream the data when the pattern is linearly accessed and perfected [11].

Chen Tian and et.al [10] proposes techniques for the speculative parallelism to support the dynamic data structures.

Suma and Gopalan[9] describes about the interthread data dependences with the speculative parallelization techniques.

III. PROPOSED METHODOLOGY

We propose a new routing technique called Speculative Value Prediction Routing (SVPR) for the performance analysis that consists of large volume of packets flow through the network to predict the path of flow of packets without congestion and to capture the recovery time of the flow of the packets from the producer node to the consumer nodes. For the performance analysis which is optimal and efficient.

The design of the methodology is as follows. The nodes are represented as the producer nodes and consumer nodes, the node which sends the data and receives the data respectively.

Case 1: Single producer to single consumer without speculation

Fig.1 represents the flow analysis without speculation from the network, the packets flow, the bandwidth is captured and stored onto the input file which in turn is provided to Hadoop framework that create the output file which is given for the analysis for training the data to know congestion recovery time and control time.

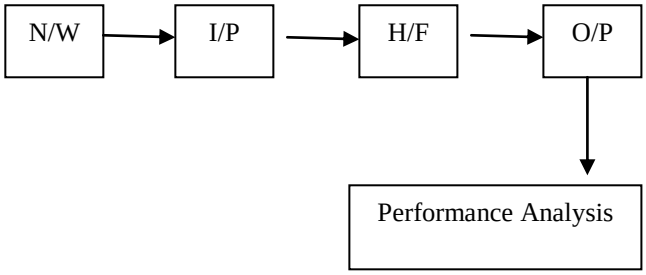

Fig. 1 Single producer to single consumer

Case 2: Multiple producers to multiple consumers with speculative value prediction.

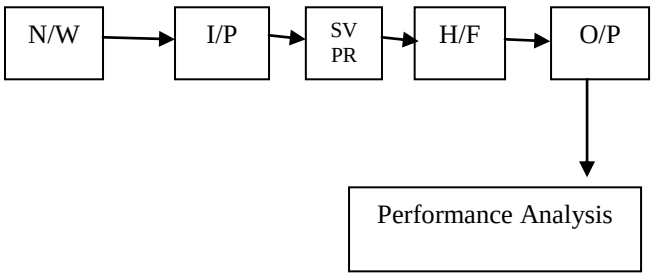

Fig.2 Multiple Producers to Multiple Consumers

Fig.2 represents the flow analysis diagram with speculative value prediction routing algorithm before the path is optimized by the Hadoop Framework. The speculative value prediction algorithm predicts the successful path of flow of packets considering the promising bandwidth from the producer nodes to the consumer nodes.

The cosmetics of the speculative parallel technique is that speculative value prediction predicts the flow path from multiple producers to multiple consumers parallel, once the Hadoop cluster decides on the flow path of the packets ,if the prediction is correct, Hadoop need not to execute that part of the executable instruction as already that path is predicted by the SVPR algorithm results in saving of the execution time. Suppose the prediction goes wrong, the Hadoop squashes the prediction made such that the path decided by Hadoop is used for the packet flow. The SVPR algorithm is basically used to increase the instruction level parallelism and increase the accuracy of the prediction algorithm as it results in saving of execution time and number of cycles needed for the execution.

The SVPR algorithm does not disturbs the architecture of the system if the prediction goes wrong making system in stable state. The SVPR algorithm is as follows. The approach is towards the dynamically changing the flow routing policy with speculative prediction.

The SVPR algorithm purpose is to minimize the total delivery time of the packets that flow from multiple producers to multiple consumers. The algorithm design uses greedy approach with the speculative value prediction as selecting next nodes that shifts from one phase to another by change in either topology or traffic dynamically explored and the best predicted value of the path for the flow of packets is dynamically adopted in each phase of searching and exploration.

In the phase of searching, consider the fig. 16 ,the producer p1,p2 .. need to route the flow of packets to the consumers c1,c2…the first phase is searching for the congestion i.e., high load across the flow path. The highly loaded paths are predicted based on the trained data with profiling and not selected for the routing of packets over a period of time such that the highly loaded paths gets recovered due to congestion control and the trained data holds the traces of the information on the network traffic, recovery time and congestion control is determined and after each recovery time, the predictive algorithms are called to predict the path of flow if the prediction goes correct based on trained data the route is selected otherwise route is s quashed and Hadoop cluster selects its path of flow of packets on that the performance is carried out to check for the predicting routes in the future.
The SVPR algorithm is as follows: The tables needed for the algorithms

Algorithm SVPR(no. of nodes)
PDp(C,N) predicted delivery time from the producer to consumer via nearest node.
BDp(C,N) Best predicted delivery time from the producer to consumer via nearest node.
RDp(C,N) Recoverable rate of the flow path
SS =(transmission delay+Queue time+ min(PDp(C,Z))-PDp(C,N)

PDp(C,N) ←PDp(C,Z)+SS
BDp(C,Z)←min(BDp(C,Z),PDp(C,N))
For all the producers,
If (SS < 0)
SR←SS
BDp(C,N)
Prediction is successful
Send to the Hadoop framework
Else if (SS > 0)
RDp(C,N)
Prediction is failed
Hadoop recovers its path for the flow of packets.
Endif.
SU(C,N) is current time

The nearest node is calculated as follows.

Algorithm nearnode(Z)
For each nearest node of C of N
ST=current time-SU(C,N)
PDp(C,N)=max{ PDp(C,N)+BDp(C,N),RDp(C,N)}
Z=min(PDp(C,N)

The algorithm SVPR accepts the number of nodes, it predicts the delivery time, best predicted time. The SS represents the delay time in sending the packets from the producer to consumer. Considering all the producers, the best predicted delivery time from producer to consumer and rate of recoverable flow path is calculated. if the prediction is successful, the Hadoop framework sends the packets else

recovery is called if the prediction goes wrong. The Hadoop recovers the path and sends packets in the regular path calculating the near node.

Test is conducted on only 4 producers and 4 consumers and it works promising if the correct trained data is provided for the prediction such that accuracy of the prediction is higher.
The algorithm is comparatively efficient compared to the PQ algorithm used in a dynamically changing network load majorly and the algorithm does not disturb the architectural state of the system even though the prediction goes wrong. The Hadoop cluster takes care of the normal execution of the system for routing the flow of packets.

The experiment is conducted on the I3 HP laptop with Ubuntu as operating system Hadoop is installed on it. To create the topology, we used Mininet tool. Once the flow is generated, it is captured by making use of WireShark Tool. The captured flows are then given to SVPR algorithm, if the prediction goes correct, Hadoop captures the delivery time and recovery time from the predicted path else if the prediction goes wrong, Hadoop applies map reduce technique to find the path of flow of the packets and provides a output file. This output file is sent for the performance tool for the analysis.

IV. Results and Interpretation

The algorithm works efficiently with 4 producer nodes as it is depicted in the performance graph in fig.21. The algorithm has to be tested on multiple producer nodes to measure the performance analysis as the future enhancements. The predictive algorithm works according if the no. Of nodes is 4.and it is to be checked on increasing the nodes.

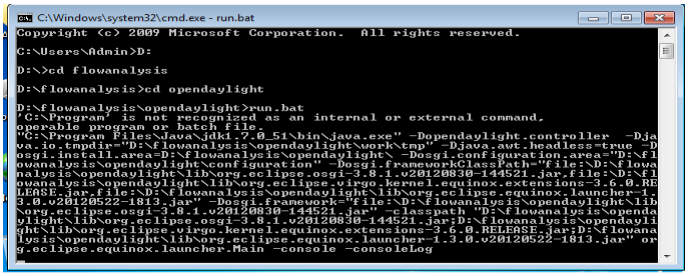

Fig.3 Running of Open Daylight

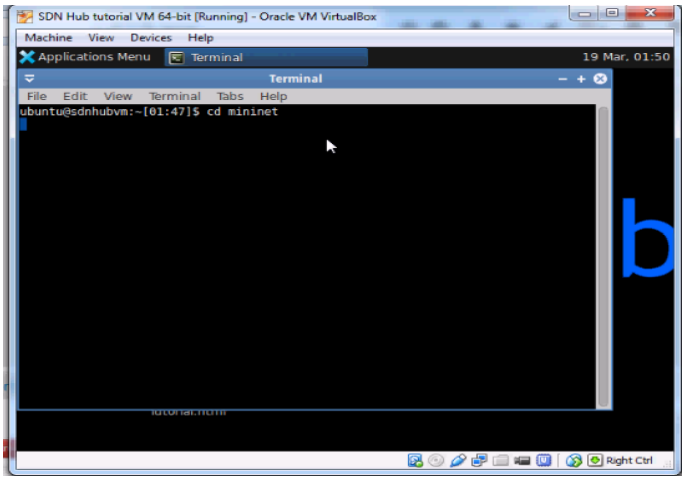

Fig.4 Running of Mininet

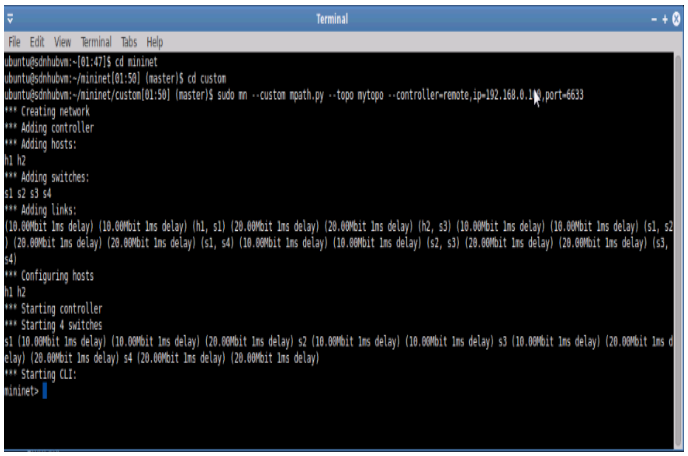

Fig.5 creating topology

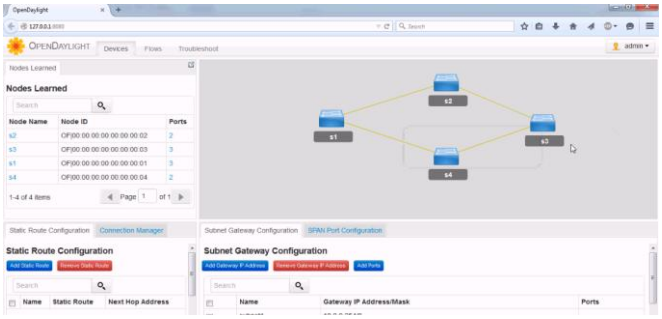

Fig.6 Topology created by user

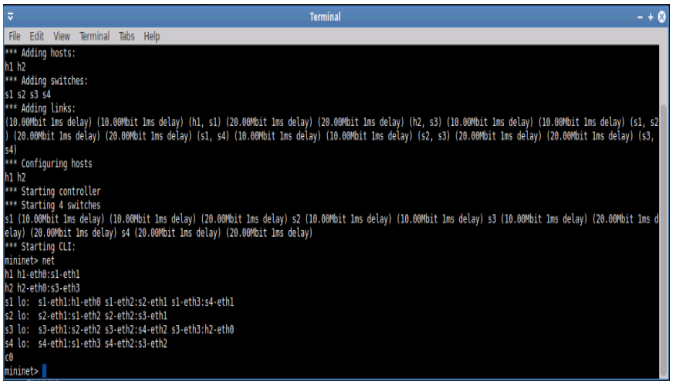

Fig.7 checking the internal connection

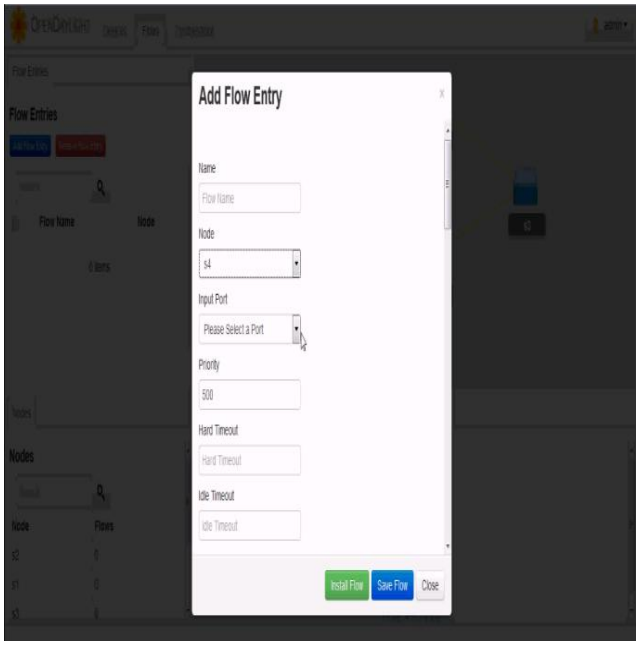

Fig.8 Adding flows to the topology

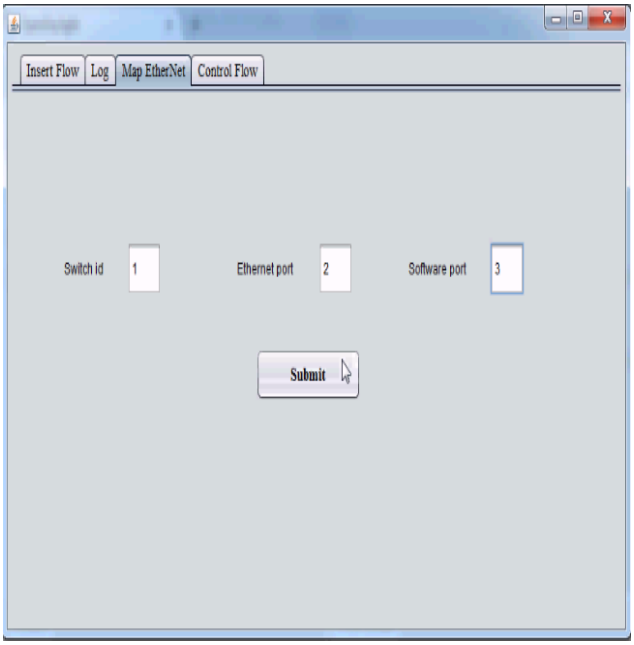

Fig.11 Interface for mapping node 1, port 2

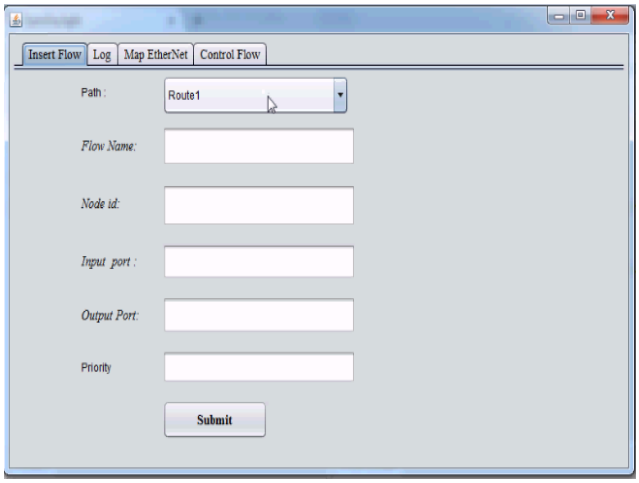

Fig.9 interface for Inserting flows

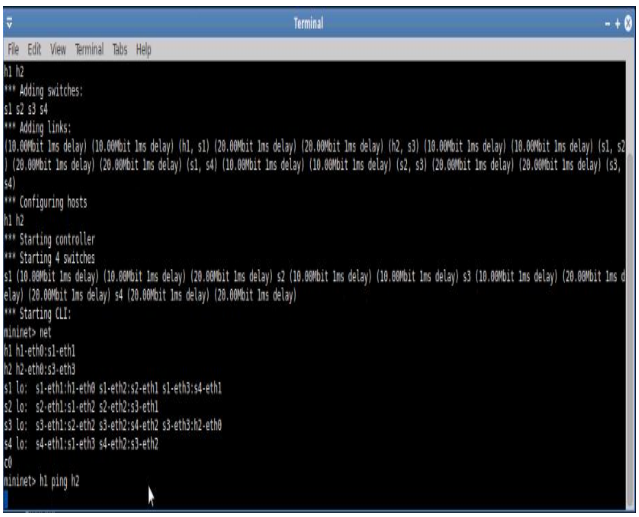

Fig.12 sending packets from host1 to host 2

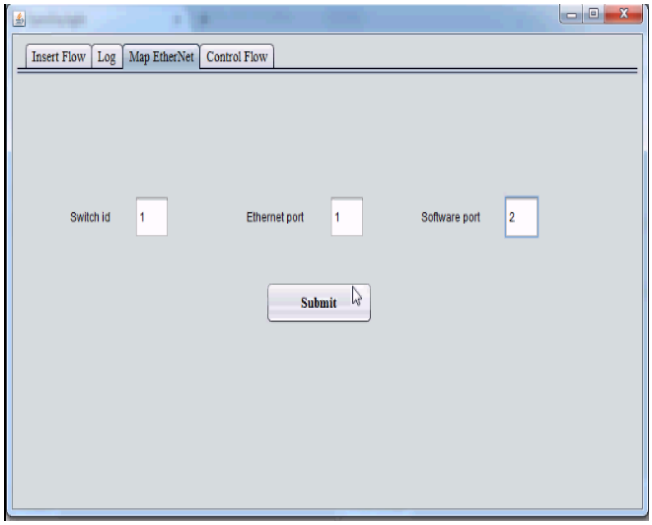

Fig.10 Interface for mapping node 1, port1

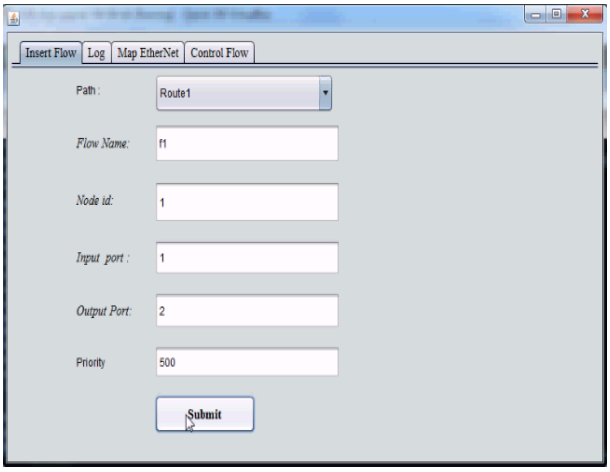

Fig.13 Interface for inserting flows-1

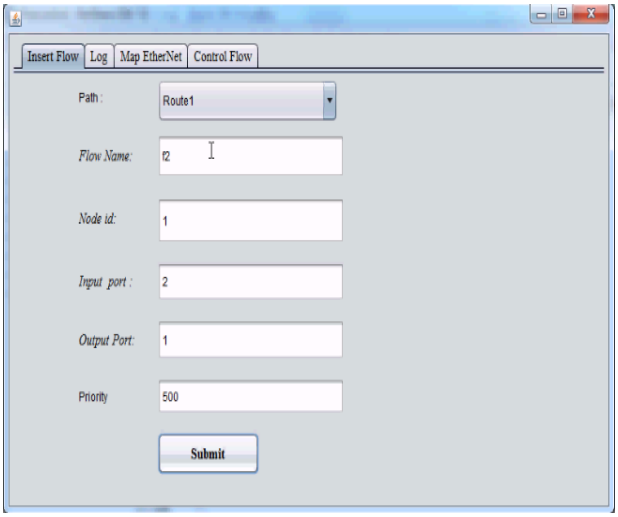

Fig.14 Interface for inserting flows-2

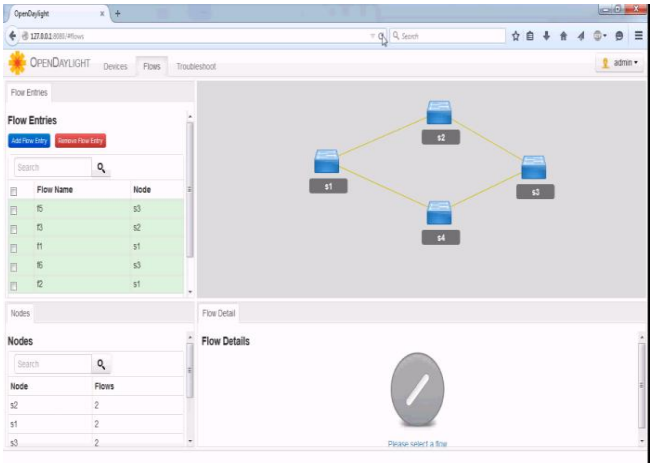

Fig.15 Single producer-consumer topology.

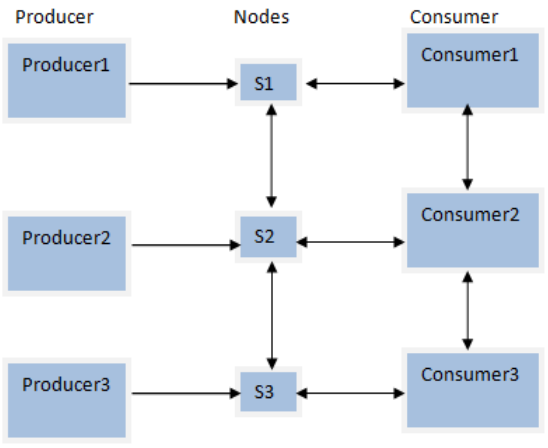

Fig.16 Multiple producer-consumer topology.

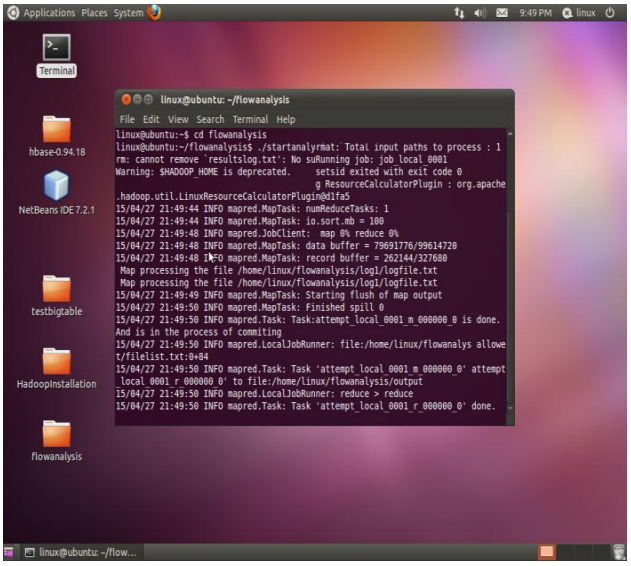

Fig.17 Hadoop Flow Analysis

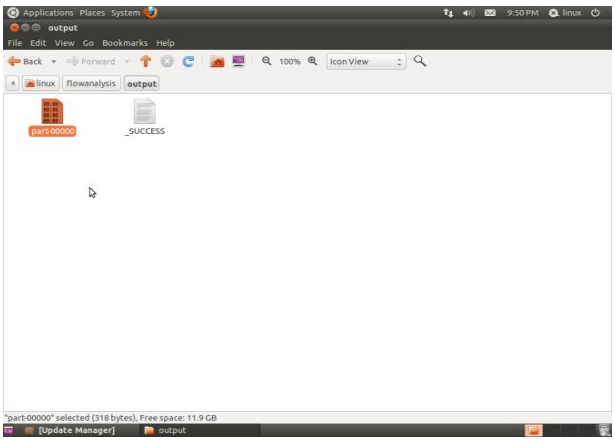

Fig .18 Output of Flow Analysis

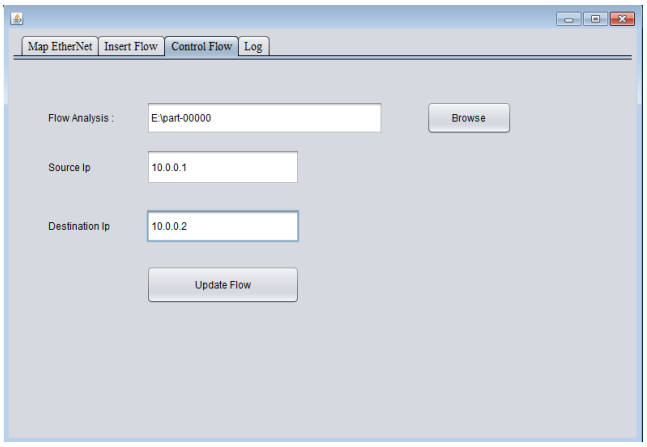

Fig.19 checking for congestion

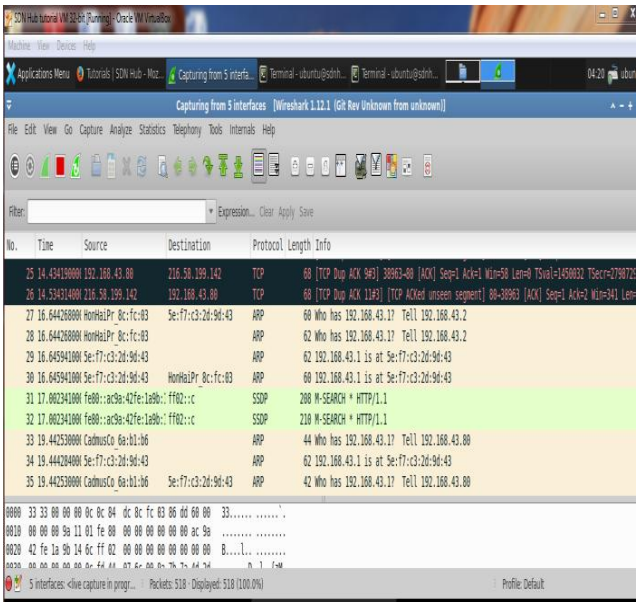

Fig.20 Flow capture.

Fig. 21 Performance Analysis graph.

V. CONCLUSION

The SVPR algorithm is efficient with the tested 4 producer-consumer nodes and the performance analysis of the algorithm is also promising. The algorithm has to be tested on the multiple producer-consumer networks. The performance analysis of the SVPR algorithm is provided that shows that the SVPR algorithm works more efficiently than the regular algorithm. The flow capturing of the packets provides necessary information regarding the path of flow of the packets which Hadoop sends.

REFERENCES

[1] M. Yu, L. Jose, and R. Miao," Software defined Traffic measurement with open sketch," in Proceedings 10th USENIX Symposium on Networked Systems Design and Implementation NSDI, vol, 13, 2013.

[2] Scscc J. Shafer, S. Rixner and Alan L. Cox, "The Hadoop Distribution File system:Balancing Portabilityand Profermance", in Proceedings of the 10th ACM SIGCOMM conference on Internet measurement ACM 2010.

[3] T. Benson, A. Akella, and D. A. Maltz, "Network traffic Characteristics of data centers in the wild," in Proceedings of the 10th ACM. SIGCOMM conference on Internet Measurement. ACM, 2010, pp. 267–280.

[4] A. W. Moore and K. Papagiannaki," Toward the accurate Identification of network applications," in Passive and Active network Measurement. Springer, 2005, pp. 41-54.

[5] L . Bernaille, R. Teixeira, I. Akodkenou, A.soule and K. Salamatian, "Traffic classification on the fly", ACM SIGCOMM Computer Communication Review, vol 36, no.2, pp 23-26,2006

[6] Yuanjun Cai, Min Luo, "Flow Identification and Characteristics Mining from Internet Traffic using Hadoop" in 978-1-4799-4383-8/14/ at IEEE 2014.

[7] Apache Hadoop Website, http://Hadoop. apache. org/

[8] David Kaeli., pen-chung yew., "Speculative Execution in high performance computer Architectures" Chapman & hall/CRC Chapter13,14.

[9] Suma S ,N.P. Gopalan," Coalesced Speculative Prefetching and Inter thread Data Dependences IEEE international Conference on Computer Communication and Informatics (ICCCI 2014) Sri Shakthi Engineering college, Coimbatore ,India Jan 3-5 2014. CFP1408R-CDR/ISBN978-1-4799-2352-6/14©2014IEEE.

[10] Chen Tian., Min Feng., Rajiv Gupta., "Supporting Speculative Parallelization in the Presence of Dynamic Data Structures" PLDI'10, June 5-10, 2010 Toronto, Ontorio, Canada.

[11] Uht A.K., Morano D., khalafi A., MDEAlba.,Kaeli Levo D., "A Scalable processor with High IPC" Journal of Instruction level Parallelism 5th August 2003.