Open Box Protocol (OBP)

Paulo Loureiro[1], Saverio Mascolo[2], Edmundo Monteiro[3]

[1]Polytechnic Institute of Leiria, Leiria, Portugal, loureiro.pjg@gmail.pt
[2]Politecnico di Bari, Bari, Italy, saverio.mascolo@gmail.com
[3]University of Coimbra, Coimbra, Portugal, edmundo@dei.uc.pt

Abstract. In this paper we propose a new explicit congestion control approach, Open Box Protocol (OBP). This is an approach which provides important information to traffic sources, about the network path state. This solution enables an efficient use of the network capacity, since the sources make congestion control decisions based on information about the network path state, which carries traffic from the sources to the destinations end systems. The OBP gives sources the capacity to look inside the network and to make their congestion control decisions, based, not only on packets loss or packets delay, but also based on another kind of information. For example, the most restricted interface capacity, the available bandwidth, the RTT variations or the presence of heterogeneous transmission means.

In the paper we describe the OBP and discuss its evaluation results based on ns-2 simulations. The results show the OBP´s capacity to provide traffic sources with the required information, in static or dynamic network scenarios, and to make correct and quick congestion control decisions. Also, it is visible that the OBP avoids the full queue problem and tries to keep the queues near zero occupation. Moreover the results show that with the OBP it is possible to obtain better throughput than other explicit congestion control protocols, avoid congestion and simultaneously keep the queues near zero occupation.

Keywords: Network Protocols, Transport Protocols, TCP Congestion Control, Routing and Forwarding

1. **Introduction**

Congestion control algorithms have many objectives to address. They must maintain the performance independently of the flows size distribution; they must guarantee short flow completion times in presence of a mix of flows; they have to ensure all flows with a fair share of available bandwidth; they have to make an efficient use of high bandwidth-delay links; they must have the capacity to react to sudden changes in network paths (wireless links); algorithms must be stable, robust [1], [2] and deployable in the Internet. On the other hand, implicit congestion control algorithms [3], [4], [5], [6], the most used in the Internet [7], give traffic sources poor information about network state, typically information about packets loss [3] and / or round trip time (RTT) variations [8], [5], [6]. With these two pieces of information, congestion control algorithms have to control the congestion inside the network. The great number of proposals about congestion control algorithms shows that any algorithm that only uses these two pieces of information will have limitations in making a better use of network capacity.

Internet traffic is of dynamic nature and the characteristics of this traffic are changing. Beyond the traditional applications, Internet is used by a set of new applications, for example the multimedia applications VoIP and IPTV. This kind of applications has changed the characteristics of the internet traffic and therefore the congestion control actions must be adjusted. Besides that, the wireless networks are in expansion. In this case the congestion control decisions may not be the best when using the detection of packets loss as the only criterion to identify the congestion situations. In wireless networks the corruption in packets is more frequent than in wire networks. Moreover, the current network capacity is larger than in the past. This new Internet has new congestion control challenges and the transport protocols have to control the congestion situations and efficiently use the network resources. It is important to bear in mind that short capacity networks still exist and need to coexist with high capacity networks. Congestion control solutions based on router collaboration have good potential because routers are the place where the congestion normally occurs. Therefore, they can quickly detect the congestion situations. Using routers, the congestion symptoms are identified in a direct way because routers detect that the amount of packets that arrives at interfaces is bigger than the capacity of the interface to process all the packets. This solution contrasts with the traditional solutions, by which the congestion symptoms are detected by the identification of packet loss or time outs.

Router based congestion control mechanisms allow quick congestion detection and the actions taken to control the congestion can be more effective. Without router collaboration the congestion detection is done in an indirect way, this means that the end systems only consider that the network is congested after the reception of duplicated ACK packets, or time outs. The actions to control the congestion are taken with delay in reference to the instant when the congestion began. This factor is important because the congestion duration can be shortened, the packet loss reduced and the network resources better used.

There are currently some solutions for congestion control based on router collaboration. These solutions can be categorized by the role played by routers in controlling the congestion. There are solutions that use the routers to detect congestion situations and to send this information to end systems, which make decisions to control the congestion inside network. Explicit Congestion Notification (ECN) [9], [10] and Quick-Start [11] are two protocols that fall in this category. The transmission rate of each flow is decided at end systems and is based on information received from the network and the characteristics of each application.

There are other kinds of solutions that propose the increase of the role played by routers. Routers have the responsibility, not only, to identify the network resources, but also, to share these resources by each flow that crosses routers. The end systems only receive the recommendations from routers and adjust the flow transmission rate to fit the recommendations received. Explicit Congestion Notification (XCP) [12], [13] and Rate Control Protocol (RCP) [1] are two protocols that fall in this category. This approach allows routers to decide the transmission rate that each flow must use to reach the best resource utilization without causing congestion inside the network. However, the implementation of these solutions can be complex because there can be thousands of flows crossing the routers and the flows can have different expectations because they may have differentiated characteristics, for example long flows, short flows, traffic constant, traffic variable, etc. Obviously, a solution based on an entity that makes all the transmission rate decisions for thousands of flows can have problems achieving the objectives of each flow and controlling the congestion inside the network.

Traditional TCP's congestion control and avoidance algorithms [14] are powerful but not enough to provide good service in a lot of network conditions since they handle the network as if it were a black box [9]. The goal of our work is to create a new congestion control model, which we call Open Box Protocol (OBP), using router collaboration to identify the network resources along the path and to provide this information to end systems. Additionally, the congestion control decisions are made by the end systems by using the information received from routers. Moreover, the model must have the capacity to efficiently use the network resources, avoid congestion, reacting well to sudden changes in network paths, being easily deployed in the Internet and coexisting with other congestion control protocols.

The remainder of this paper is organized as follows. Section 2 provides the related work on congestion control mechanisms based on router collaboration. Section 3 describes the characteristics and the design of the proposed scheme. Results of the evaluation using simulations in ns-2 [15] are presented in Section 4. Section 5 presents the conclusions and some directions for future work.

2. Background and Related Work

Over the past few years, several solutions have been proposed to give TCP better and more network feedback, beyond packets loss information and RTT variations. In addition, the research community has been specifying alternative solutions to the TCP architecture. As the OBP, some of these models are classified in category of “modification of the network infrastructure”. They are briefly explained as follows.

The ECN and the Quick-Start are two solutions that use the router collaboration to address the congestion control problem. With the ECN, the router collaboration is done by detecting congestion situations and by informing the end systems about this situation. With the Quick-Start, the collaboration of routers is used to decide the value of the initial congestion window. Along the path routers accept or not an initial congestion window proposed by the end system. These two mechanisms are nevertheless used by the traditional congestion control mechanisms. This means that the algorithms slow start, congestion avoidance, fast retransmission and fast recovery [14] are still used and therefore the problems associated to theses mechanisms remain.

Explicit Control Protocol (XCP) is designed to work well in networks with large bandwidth-delay products. This Internet congestion control protocol outperforms TCP in conventional environments and remains efficient, fair, scalable, and stable in high bandwidth-delay product networks. The XCP generalizes the Explicit Congestion Notification proposal (ECN). In addition, the XCP introduces the new concept of decoupling utilization control from fairness control. Routers provide feedback, in terms of incremental window changes, to the sources in multiple round-trip times, which works well when all flows are long-lived. However, in a dynamic network' environment, the XCP can increase the duration of each flow beyond the ideal and can contribute to maintain more active flows in the network [1].

Rate Control Protocol (RCP) is designed to efficiently use high bandwidth-delay product networks, such as the long optical links; to be stable independently of link-capacity, round-trip times and number of active flows and to try to emulate processor sharing. Each router maintains a single fair-share rate per link. Each packet carries the rate of the bottleneck link. For each packet, the router compares the two values. If the router's fair-share rate is smaller, it overwrites the value in the packet. This way, the source learns the fair-share rate of bottleneck link of the path. The RCP gives the same transmission rate to all flows. This solution can not be the best because the flows can have differentiated characteristics, for example long flows, short flows, traffic constant, traffic variable, etc. Another relevant point, at routers, the RCP uses information presents in packets, and received from end systems, to decide the fair-share rate. This means that routers cannot simultaneously receive RCP packets and packets from other transport protocols. The implementation of this model in Internet is conditioned by this factor.

The differences between the XCP, the RCP and the OBP are the kind of feedback that end systems receive and the entities that have to make decisions about congestion control. In the OBP, the routers only have to provide sources with the network state information, whereas in the XCP and the RCP the routers have to make congestion control decisions. In the OBP, the end systems receive network state information and make decisions about congestion control, whereas with the XCP and the RCP the congestion control decisions are made by routers.

Opposite to the ECN and the Quick-Star, the OBP makes all congestion control decisions based on information received by routers. The transmission rate of each flow is only increased if the network has available resources and is immediately reduced if the network indicates the lack of resources to process all the traffic.

The OBP is computationally simpler than the XCP and the RCP, since routers do not have to make decisions about congestion control and only need to provide feedback information about the network state.

3. Open Box Protocol (OBP)

We will answer the following question: Is it possible for end systems to constantly see the network state between the source and the receiver? The answer is yes if we can represent the network path through a small set of variables and if we can continually put this information at sources. With this information, the sources can quickly make decisions about the efficient use of network capacity and quickly adapt to sudden network changes. The basic OBP algorithm operates as follows.

1) The end systems receive from routers information about the state of the network path, and continually have a current image of the network path;

2) The end systems make congestion control decisions using information about the state of the network;

3) The routers continually update their state and provide this information to the end systems.

4) The routers must compare the network state with the previous one and update this information in packets.

Any flow begins with an SYN packet. When this packet arrives at first router, the variables that represent the network state are updated. At second router the packet is evaluated and if any variable needs to be updated the router performs this exchange. Information about the network state will arrive at the end system inside the ACK packets. The end system, with this information, can make congestion control decisions for efficiently use the network resources and avoid the congestion.

3.1 State of network path

The state of the network path is a key element of this model. The sources can only make the best congestion control decisions if they have correct information about the state of the network path.

To represent the network path we only need the information that is used or important to make decisions about congestion control. It is not relevant to represent the network path at other levels. Fig.1 shows a network path with four routers. Each box represents the router output interface and has two kinds of information: the output interface capacity and the available bandwidth. In this example the *narrow link* is 45 Mbps (the most restricted interface capacity) and the available bandwidth is obtained in *tight link* (the most restricted available bandwidth).

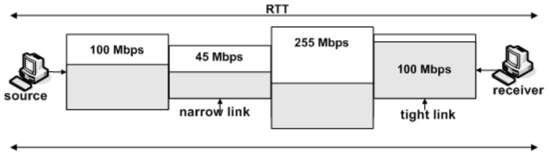

Fig. 1. Network state

To represent the network path from one end system to other we need the following small set of variables:

- *narrow link*: along the path, this is the link with the most limited capacity. In the previous image the *narrow link* is 45 Mbps;
- *tight link*: along the path, this is the link with the least available bandwidth. In the previous image the *tight link* is near zero Mbps;
- *round trip time (RTT)*: time needed to send a data packet and to receive the associated ACK packet;
- *heterogeneous path*: having or not heterogeneous means along path, for example wireless links.

With the exception of RTT, which is obtained by using information from TCP header, all the other variables are carried inside data packets in IP header. This information returns to the sources inside the TCP header of ACK packets.

3.2 Router processing

The OBP model assumes that congestion control decisions are made by the sources. The routers, along the path, only provide information to the sources. The routers do not make any kind of congestion control decisions and do not have to process any algorithm to do this. This approach is relevant if there are many flows crossing the routers. For example, in the XCP, the routers must estimate a window increment or decrement over the current window size for each packet; in the RCP, the routers must manage a fair-share rate for each flow. These decisions involve processing for each packet.

The OBP considers that, when a new packet is inside the first router of the network path, and before the packet leaves, the router updates three variables: *narrow link*, *tight link* and *heterogeneous path*. When this packet is at the second router these variables are changed or not. The value of the *narrow link* is changed if the output interface capacity is less than the previous one. The value of the *tight link* is changed if the available bandwidth is less than the previous one. The value of the *heterogeneous path* is changed if this router has a wireless link. The available bandwidth at the *tight link* is obtained through the following steps: at the output interface, and for short periods of time, all packets that get in are counted. Then, all of the packets that got in are divided by the period of time. This procedure gives us the used bandwidth.

3.3 Source processing

Unlike other explicit congestion control protocols, the OBP congestion control decisions are made in sources. However, the OBP makes decisions using explicit information received from the network. The sources have the most critical task because they have to make decisions concerning the following objectives: performance independently of the flows size distribution, short flow completion times, fair share of available bandwidth, efficient use of high bandwidth-delay product links, capacity to react to sudden changes in the network paths, avoiding congestion.

To address those objectives the OBP uses the following principles:

- New flows begin with a high transmission rate. The initial transmission rate depends on the available bandwidth at *tight link* and the interface capacity at *narrow link,* and is calculated after the sources have received the SYN-ACK packet. This method assures short completion times for short flows;
- Every time the sources receive an ACK packet the transmission rate is tuned. This is done using the feedback information received from the network. These transmission rate adjustments are done to come near to zero the available bandwidth and the RTTs near the minimum. The RTT near minimum means that the queue occupation is near zero;
- The OBP model tries to efficiently use the network path capacity and tries avoiding congestion. This means that the available bandwidth must always be near zero;

The following equations show how the transmission rate is adjusted. The initial transmission rate $W(_{t0})$ depends on the available bandwidth $AB(_{t0})$, the capacity $CN(_{t0})$ at *narrow link* and the constants α and β.

$$W(_{t0}) = \alpha * AB(_{t0}) + \beta * CB(_{t0})$$

Every time a new ACK packet is received, the feedback information inside the packet is used to make adjustments in transmission rate. These adjustments are done based on feedback information and based on an equilibrium point. The equilibrium point is updated in multiples of RTT and is calculated based on the mean of the transmission rate during the previous period (this period is equal to an RTT average). The transmission rate is updated whenever an ACK packet is received and is always around the equilibrium point. Fig. 2 shows, for a flow, an example of the behavior of these two variables: the transmission rate and the equilibrium point. In this example, at the beginning, we can see that the transmission rate is always higher than the equilibrium point. This means that there is available bandwidth to be used. After the first RTTs the source finds the equilibrium point that enables filling the entire network path between the source and the destination.

The network warnings can include negative available bandwidth - this case corresponds to a transmission rate below the equilibrium point; or positive available bandwidth – in this case corresponds to a transmission rate above the equilibrium point. Although this image is obtained from a test with 100 flows, it is visible, for this flow, that the OBP quickly finds the maximum equilibrium point and stabilizes in this value. This situation corresponds to the efficient use of network resources.

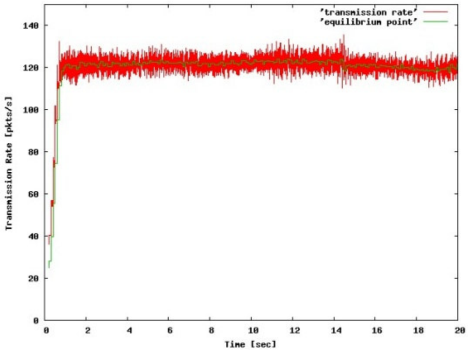

Fig. 2. Transmission rate vs Equilibrium point of a flow. This test had 100 flows, a bottleneck link equal to 1 Gbps and the RTT equal to 0.1 *ms.*

The transmission rate is updated every time an ACK packet is received. The transmission rate $W_{(t)}$ depends on the current equilibrium point $EP_{(k)}$, the available bandwidth $AB_{(t)}$, the capacity $CN_{(t)}$ at *narrow link* and a constant δ. Also, it is affected by the RTT if this value is different from the minimum RTT, affected by a constant μ.

$$W_{(t)} = EP_{(k)} + EP_{(k)}*[(\delta * AB_{(t)})/(AB_{(t)}+CN_{(t)})] + EP_{(k)}*\mu*[RTT_{min} - RTT]$$

This formula allows obtaining transmission rates around the equilibrium point. If the AB(t) is near zero the W(t) obtained is the EP(k). However, if the AB(t) received is negative or if the RTT is large, the value obtained of W(t) is less than the EP(k). In an extreme case, the W(t) obtained may be near zero, for example if the AB(t) has a high negative value or if the RTT is very high. This solution protects the network against congestion collapse, because it can instantly reduce the transmission rate to few packets [2]. This formula also enables a quick adaptation to sudden or transient events [2] as it admits changes in the transmission rate whenever ACK packet is received.

The equilibrium point is updated one time per RTT. When this occurs, the equilibrium point is updated with the mean of all transmission rates calculated whenever an ACK packet is received during the previous period. In Fig. 2 this corresponds to green line.

$$EP(k) = \text{mean}(\textstyle\sum W_{(t)});\ \text{during last RTT}$$

The formulas used by OBP assure that the increase in transmission rate is always decided by the feedback received from the routers. Also, the transmission rate can be updated every time an ACK packet is received. Opposite to this behavior the traditional congestion control algorithms allow the sources to increase the transmission rate without knowing if the network is near congestion.

3.4 Destination processing

When one data packet reaches the destination end system, this system creates a new ACK packet. At transport level, this ACK has three new variables, *narrow link*, *tight link* and *heterogeneous path*. These variables are filled with information received from data packet. Information present in ACK packet arrives at the source and is used to make congestion control decisions.

3.5 Understanding the OBP algorithm

The sources put packets inside network at a certain transmission rate. Every packet includes three new IP level variables *narrow link, tight link and heterogeneous path*. These variables are updated at the routers belonging to the path. The receiver inserts this information in the ACK packet and it will arrive at the source. With this information the sources update the transmission rate using the formulas presented above. Fig. 3 shows the algorithm.

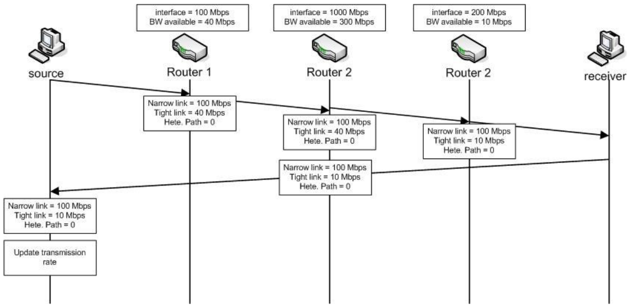

Fig. 3. The OBP algorithm

3.6 Deployable in internet

To be deployable in internet any new congestion control approach should consider the overhead in terms of packet header size, the added complexity at end systems or routers and the interaction with other transport protocols [16]. In terms of packet size overhead this model has three new variables: *narrow link, tight link* and *heterogeneous path*. The *narrow link* and the *tight link* are represented in units of KBps and use 3 bytes each. The *heterogeneous path* is a boolean (1 bit).

In terms of complexity, the OBP implementation puts the load processing in the sources side. This means that the routers just have to provide information. The congestion control decisions are made at the sources. The additional processing in the routers is only to maintain up-to-date the current available bandwidth. This is maintained by counting the number of packets that arrive at output interface for short periods of time. At end systems the processing corresponds to updating the transmission rate and the equilibrium point and implementing the transmission rate, using the formulas above.

Moreover, each flow makes its congestion control decisions. We can have the network being used by flows that are controlled by different congestion control algorithms, among which the OBP. In this case the used bandwidth calculated at the routers contains the traffic received, which has packets from different congestion control protocols.

4 Evaluation

To evaluate the OBP model, we have created simulations on the ns-2 simulator [15] (version 2.30) with the OBP implementation. Fig. 4 shows the network topology used in the simulations (this topology is known as dumbbell network). The bottleneck link is set to 1 Gbps or 2.4 Gbps. The RTT is set to 100 ms.

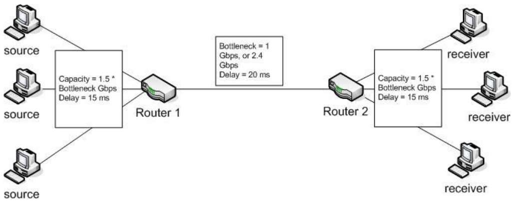

Fig. 4. Network topology

To evaluate the performance of OBP we did tests with long flows. We also did with flows created by Poison distribution and the flows size was defined by Pareto distribution. We equally tested the OBP and compared it with the TCP Reno, the XCP, the RCP and the TCP Reno with Quick-Star.

4.1 Behavior with long flows

In these two tests we generated 100 flows at instant 0 seconds. The bottleneck link was equal to 1 Gbps or 2.4 Gbps. The RTT was 100 ms, and the size was 1000 bytes per packet. The transmission rate used the following configurations: $\alpha = 0.0$, $\beta = 0.002$, $\mu = 1.0$, $\delta = 1.0$. The objective of these tests was to verify if the available bandwidth, at bottleneck link, was near zero; which meant that sources generated enough traffic to fill the path, and if the RTT had no oscillations, which meant that the routers' queues were near zero occupation.

Convergence and stability. Fig. 5 shows the available bandwidth received at sources for tests with the bottleneck link equal to 1 Gbps and 2.4 Gbps. We can see that, in few RTTs, the sources generated traffic to fill the path and the feedback received about the available bandwidth was near zero. The available bandwidth can be less than zero. This means that, at those instants the amount of traffic generated is greater than the capacity of output interface. In this case, packets were momentarily stored in the routers' queues. It is thus visible the capacity of the OBP model to generate packets, which quickly fill the path. Another observation is the stability of the OBP, after the firsts RTTs the OBP finds the correct equilibrium point. Also the OBP discovers the equilibrium point in same time, for both bottleneck configurations.

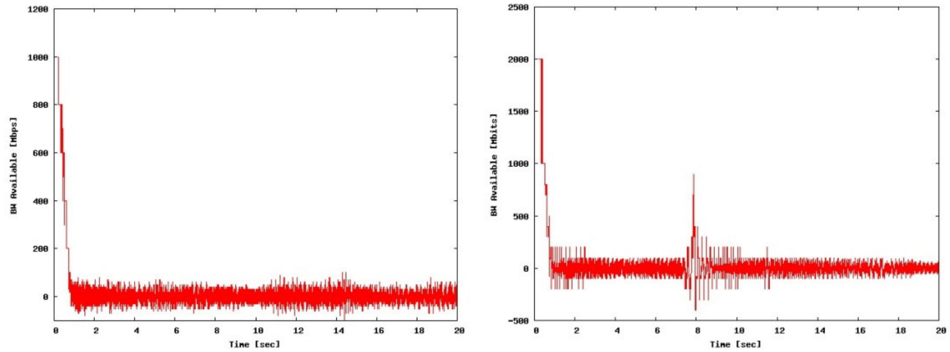

Fig. 5. Available bandwidth sent to sources by the bottleneck router, with the bottleneck interface equal to 1Gbps and 2.4Gbps.

Fairness between flows. The OBP model uses explicit information to make congestion control decisions. The increasing or decreasing of the transmission rate is based on the network state. Although the congestion control decisions are made for each flow, Fig. 6 shows that the formula used tends to guarantee fairness to flows with equal characteristics. In these figures we can see the average transmission rate obtained for each flow. All the 100 flows received similar throughput. We obtained the same results when we used the bottleneck configuration equal to 1 Gbps and 2.4 Gbps.

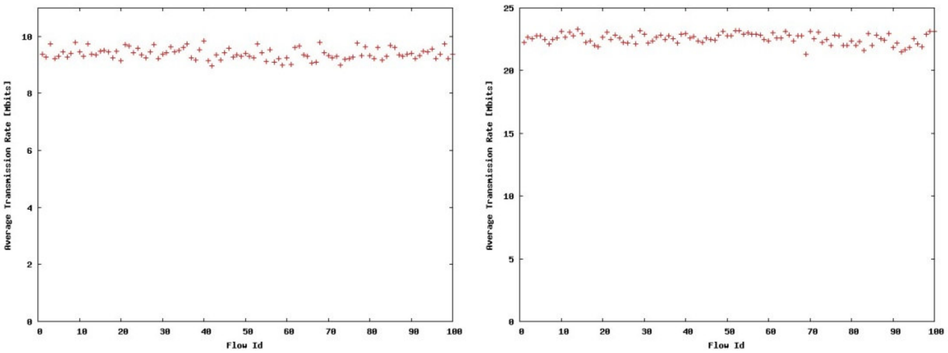

Fig. 6. Fairness between 100 flows. The bottleneck link is 1 Gbps and 2.4 Gbps.

The state of routers' queues. The OBP also uses the time that a packet spends inside the routers' queues to make congestion control decisions. This situation allows maintaining the queues' occupation near zero, or with tendency to near zero. This is important because problems associated to full queues and tail drop are well known. Fig. 7 shows that the RTT is always near 0.1 ms. This means that the identified RTT corresponds to the propagation delay and, therefore, packets haven't spent extra time inside the routers' queues. By these results we can conclude that the OBP has capacity to generate traffic that tends to use the path capacity, without filling the routers' queues.

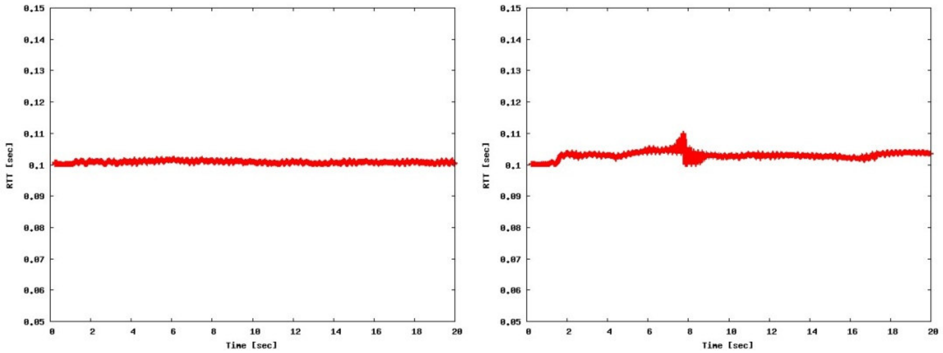

Fig. 7. Packets' round trip time information received at sources. The bottleneck link is 1 Gbps and 2.4 Gbps.

4.2 OBP behavior with variation of traffic characteristics

In this section our goal is to find out the OBP performance in presence of traffic with dynamic characteristics. The simulations used the Pareto distribution to define the flows' size, with the mean of 125 packets (bottleneck link 1 Gbps) and 300 packets (bottleneck link 2.4 Gbps), and the shape was 1.8. The arrival of flows was defined by the Poisson distribution with the lambda equal to 950 packets. The duration of the tests was 20 seconds and the packets had 1000 bytes.

The results of these tests are analyzed through the available bandwidth sent by the routers, the average completion time per flow and the RTT. The average completion time represents the difference between two instants, SYN packet departure and arrival of the last data packet.

Convergence and stability. Fig. 8 shows the available bandwidth sent to the sources by the bottleneck router. Although these tests generated 950 new flows per second and the flows' size varied between 56 and 16340 packets (1 Gbps) or between 134 and 39216 packets (2.4 Gbps), the results show the capacity of OBP model to manage the transmission rate and to fill the network path without congesting the bottleneck link. The Poisson distribution is used to define the instant of flows creation, so there are some variations in the available bandwidth along the time. The stability of the OBP is equally confirmed through these results and the available bandwidth tends to zero.

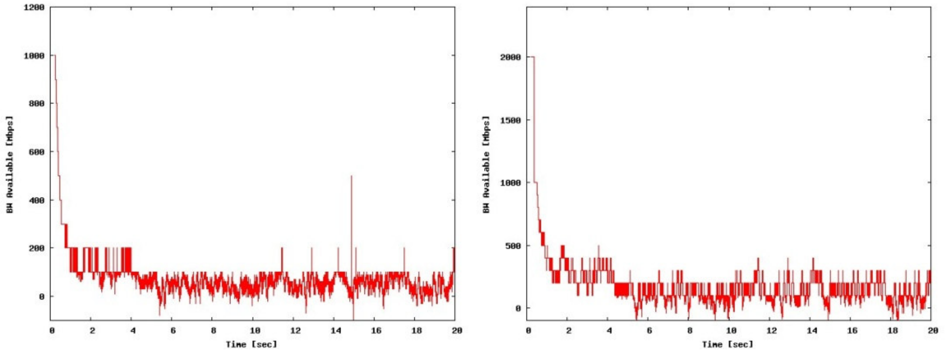

Fig. 8. Available bandwidth sent to sources by the bottleneck router. The bottleneck is 1 Gbps and 2.4 Gbps and the traffic is flows Pareto distribution with mean of 125 packets and shape 1.8, and Poisson flow arrival with lambda 950.

The state of routers' queues. The RTT calculated at the sources is always near the minimum RTT. The OBP tries to use the available bandwidth, without filling the routers' queues. This is visible in Fig. 9, which shows the RTT that includes the propagation delay (0.1 ms) and the extra time spent inside routers' queues. This extra time is near zero.

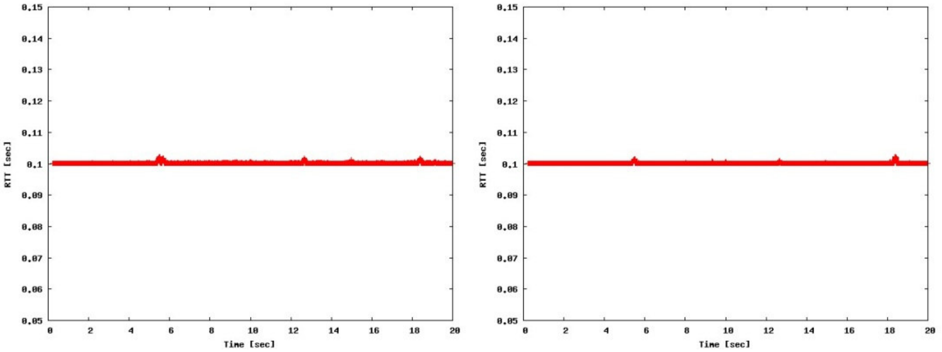

Fig. 9. Packets' round trip time information received at sources. The bottleneck link is 1 Gbps and 2.4 Gbps.

Average completion time. The average completion time is a good metric for elastic flows and for short flows because it gives the necessary time to transfer all data of the flow. Fig. 10 shows that there are not great variations in completion time of flows with identical size. The exceptions are related to the instants of flows' creation and, also, to the network load at those instants. The behavior of the OBP is good with the bottleneck link of 1 Gbps likewise 2.4 Gbps. These results are again analyzed in the next section where they will be compared with the results of other transport protocols.

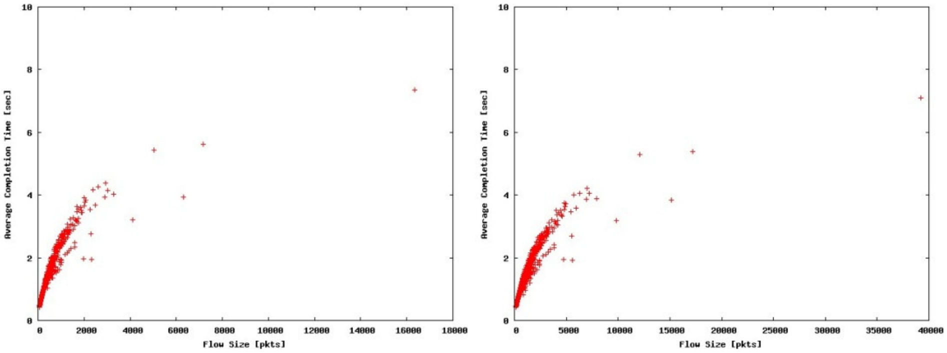

Fig. 10. Average completion time per flow size. The bottleneck link is 1 Gbps and 2.4 Gbps.

4.3 Comparative results among the OBP, the TCP Reno, the TCP Reno with Quick-Start, the XCP and the RCP

In this section our goal is to compare the OBP performance with other transport protocols. In these tests we compare the OBP with the TCP Reno, the XCP, the RCP and the TCP Reno with Quick-Start with the request rate equal to 100 KBps. We chose the XCP and the RCP because these two models use explicit congestion information to define their transmission rates. These tests were done using the Pareto distribution to define the flows' size, with mean of 1250 packets and shape 1.8. The bottleneck link was 1 Gbps and the queue' size was 2000 packets. The arrival of flows was by Poisson distribution with lambda 95. The duration of tests was 20 seconds and packets had 1000 bytes. The results of these tests are analyzed through the average completion time per flow, the average throughput and the RTT of each packet.

Completion time and Throughput. From Fig. 11, the average completion time for the OBP is more stable than for the TCP Reno, the TCP Reno with Quick-Start and the XCP. This is visible by the smallest variations between flows with near sizes. For example, in the TCP Reno, these variations are greater. The longer flows are also concluded more quickly by the OBP than by the TCP Reno, the TCP Reno with Quick-Start and the XCP.

The results of the RCP have to be analyzed together with the packets' RTT results. The RCP is very aggressive and defines high transmission rates. This is visible in Fig. 11 where the packets' RTT is always higher than the propagation delay value. The consequence of this aggressiveness is the packets loss, showed by Fig. 12, where the RCP' throughput is less than the OBP or the XCP. Fig. 11 and Fig. 12 show the capacity of the OBP to get the best throughput and, at the same time, to keep the RTT near the minimum (0.1 ms). This means that the routers' queues are near zero. On the other hand, the TCP Reno, the TCP Reno with Quick-Start and the RCP results show that these protocols fill the network and packets have to be delayed inside router' queues.

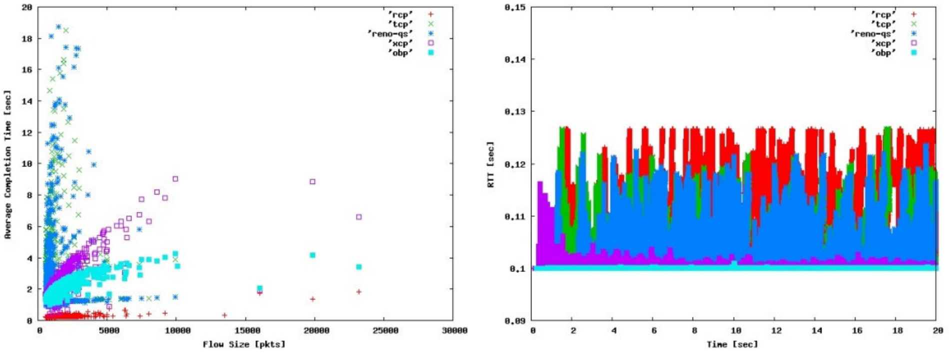

Fig. 11. Average completion time per flow reached in 20 seconds. Packets' round trip time information received at sources during 20 seconds.

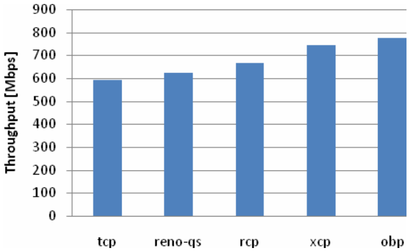

Fig. 12. Average throughput reached in 20 second.

Packets loss. The OBP model tries to efficiently use the network capacity as well as tries to maintain the router' queues near zero occupation. The consequence of this approach is the small variations between the RTTs and, simultaneously, the reduction of packets loss. All the tests showed the absence of losses.

5 Conclusions and Future Work

In this paper we present Open Box Protocol (OBP). This solution enables the efficient use of the network capacity because the sources make congestion control decisions based on information about the network state. The OBP gives sources capacities to look inside the network and to make their congestion control decisions, based on the most restricted interface capacity, the available bandwidth, the RTT variations and the heterogeneous transmission means.

We have shown through analysis and experimental evaluation that the OBP has capacity to put information about the network state in the sources. Also, it is visible that the OBP can efficiently use the network bandwidth, keeping the routers' queues near zero occupation. We have equally shown that OBP can have better performance than others congestion control solutions. Moreover, the OBP implementation puts the load processing on the sources side, in opposition to other congestion control approaches, which make congestion control decisions for all flows by the same routers, as the case of the XCP and the RCP.

The OBP can be used in networks where there are flows that use other congestion control protocols, because the OBP only needs to receive from routers congestion control information.

As part of our future work, we plan to test the OBP in wireless networks as in networks shared by flows that use different congestion control approaches.

References

1. Processor Sharing Flows in the Internet. Nandita Dukkipati, Masayoshi Kobayashi, Rui Zhang-Shen and Nick McKeown. [ed.] sa. Passau, Germany : Thirteenth International Workshop on Quality of Service (IWQoS), 2005.
2. Metrics for the Evaluation of Congestion Control Mechanisms. Floyd, Sally. s.l. : Internet draft - IETF, Expires: August 2007.
3. Congestion Avoidance and Control. Jacobson, Van. s.l. : ACM SIGCOMM 1988, 1988.
4. New Reno Modification to TCP's Fast Recovery. Floyd, S., Henderson, T. s.l. : RFC 2582, 1999.
5. FAST TCP: Motivation, Architecture, Algorithms, Performance. Cheng Jin, David X. Wei, Steven H. Low. Hong Kong : In Proceedings of IEEE Infocom 2004, 2004.
6. HighSpeed TCP for Large Congestion Windows. Floyd, Sally. s.l. : RFC 3649, December 2003.
7. Measuring the Evolution of Transport Protocols in the Internet. Alberto Medina, Mark Allman, Sally Floyd. s.l. : Computer Communications Review, April 2005.
8. TCP Westwood: End-to-End Bandwidth Estimation for Efficient Transport over Wired and Wireless Networks. Mascolo, S., Casetti, C., Gerla, M., Sanadidi, M., Wang, R. Rome, Italy : in Proceedings of ACM Mobicom 2001, 2001.
9. Performance Evaluation of Explicit Congestion Notification (ECN) in IP Networks. J. Hadi Salim, U. Ahmed. s.l. : RFC 2884, 2000.
10. The Addition of Explicit Congestion Notification (ECN) to IP. K. Ramakrishnan, S. Floyd, D. Black. s.l. : RFC 3168, 2001.
11. Quick-Start for TCP and IP. S. Floyd, M. Allman, A. Jain, P. Sarolahti. s.l. : RFC 4782, 2007.
12. Specification for the Explicit Control Protocol (XCP). A. Falk, Y. Pryadkin, D. Katabi. s.l. : Internet-Draft, Expires: May 9, 2007.
13. Internet Congestion Control for High Bandwidth-Delay Product Networks. Dina Katabi, Mark Handley, and Charles Rohrs. Pittsburgh : In Proceedings of ACM Sigcomm 2002, August, 2002.
14. TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms. Stevens, W. s.l. : RFC 2001.
15. The Network Simulator. http://www.isi.edu/nsnam/ns/.
16. Specifying New Congestion Control Algorithms. S. Floyd, M. Allman. s.l. : Internet-Draft, December 2006.