

## Study on TCP Performance Improvement in Dynamic Spectrum Access Networks Based on Spectrum Switch Estimation

Peng Zhang<sup>1</sup>, Kai Shi<sup>1</sup>, Sheng Lin<sup>1</sup>, Jinsong Wang<sup>1</sup>, Guangping Xu<sup>1</sup>, Peng Zhang<sup>2</sup>

<sup>1</sup>*School of Computer and Communication Engineering, Tianjin University of Technology, Tianjin, 300384, China*

<sup>2</sup>*Department of Computer Science, Tianjin Medical University, Tianjin, 300070, China*

### Abstract

Dynamic Spectrum Access (DSA) is supposed to be the key technology to improve wireless communication. And at the other hand, Transport Control Protocol (TCP) will be used in DSA networks, considering it's the de-facto standard in wired communication and also has been widely deployed in wireless networks. However, TCP may suffer significant decrease in end-to-end throughput performance caused by spectrum switches in typical DSA networks. In this paper, TCP DSA is proposed to mitigate this issue. This is achieved by monitoring the RTT and throughput periodically, and using them to indicate the spectrum switches. We also calculate the optimal congestion windows of different channels, and use them to adjust TCP congestion window. Thus, we can fully utilize the channel capacity, and obtain higher throughput. TCP DSA is evaluated through simulation and the result shows significant improvement in throughput performance.

*Keywords: Dynamic Spectrum Access, TCP, throughput.*

### 1. Introduction

Recently, wireless devices have been largely deployed and come into use, such as mobile phone and tablet. However, the radio spectrum these devices could use are limited in the 900MHz and 2.4GHz industrial, scientific and medical (ISM) bands, which becomes more and more congested. While at the other hand, the spectrum bands allocated to licensed users are under-utilized, according to the FCC report. Considering the situation, FCC issues new regulation in the Broadcast Television Bands, which allow wireless user share the TV band, which is a case of what we could call Dynamic Spectrum Access (DSA).

Dynamic Spectrum Access is a term referring that wireless devices could switch their channels between different frequency bands. Primary user (PU) is the licensed owner of a frequency band, which is unaware of the existence of Secondary User (SU). SU, in the other hand, should have the ability to sense the

presence of PU. Thus, when a primary user doesn't use the band, a secondary user could sense the white space and use it for communication. But when the primary user returns, the secondary user is forced to vacate the channel immediately.

DSA is supposed to increase the capacity of wireless networks. And there are many works of DSA in MAC/PHY layer, which mainly focused to implement the sensing functionality, channel selection algorithm, etc, which are DSA network's fundamental functionality. However, there are very few work focused on the study and improvement of TCP in DSA networks.

In DSA networks, TCP may suffer decrease in end-to-end throughput performance. As a example, when an intermediate node performs spectrum sensing, it can't forward the packet, which could temperately disconnect the link, and may make RTO timeout, which could then cause TCP to decrease it's congestion window to 1, and significantly affect the throughput. Also, the sudden presence of PU and periodic spectrum sensing could have serious influence of TCP end-to-end throughput, especially when a PU presence is mis-detect. Channel switch may also affect TCP throughput, and TCP may be unable to dynamically adjust itself fast enough to fully utilize the available bandwidth. For example when switch performed between channels with different capacity, TCP may not catch up the bandwidth fast enough while it's performing congestion avoidance.

TCP is the de-facto standard in the wired communication. And it is also widely used in wireless communication in form of several variants. So, as far as we can see, it'll be deployed in the DSA networks. Considering the problem we mentioned above, TCP should be adjusted to fully utilize the capacity in DSA network. In this paper, we mainly focused on the channel switch problem mentioned above, that is, TCP may lose the ability to keep up with the available bandwidth when a switch performed between channels with different bandwidths, which causes under-utilize of bandwidth. Our approach will measure the RTT and throughput periodically, and use them to indicate the switch between channels with different bandwidth. We also calculate the optimal congestion windows of different channels, and apply them to adjust TCP congestion window when a switch is performed. Thus, we can cut off the major time TCP uses to catch up with the bandwidth, then obtain the available bandwidth fast enough to fully utilize the channel capacity.

The rest of the paper is organized as follows. Section 2 is the related work. System model is described in Section 3, follows the mechanism in Section 4 and the evaluation in Section 5. We conclude the paper in Section 6.

## **2. Related Work**

### **2.1 MAC/PHY Layer**

Towards the deployment of DSA, there have been sufficient research efforts in MAC/PHY layer. IEEE 802.22 group has been organized to standardize the DSA

protocol, specifically operation in TV band [1]. Several MAC/PHY protocols have been discussed in different papers [2,3,4]. [2] proposes a decentralized cognitive MAC protocol considering the problem of partial sensing, and develops an analytical framework for DSA. [4] takes one step further, takes hardware constraints into consideration, and proposes HC-MAC to identify the issue of optimal spectrum sensing decision.

## **2.2 TCP Layer**

As for TCP layer, most of the papers published have focused on the traditional wireless network. For example, Ad hoc TCP(ATCP) [5] distinguishes packet loss caused by mobility or channel error from network congestion, by adding a layer between IP and transport layer to collect information of network, such to properly adapt the congestion window. [6] is a new protocol which adapts its sending rate by gathering feedback from intermediate nodes. There have been other efforts on improving TCP performance. The main approaches of these papers are focused at different packet loss from congestion and caching.

## **2.3 Cross-layer**

TP-CRAHN proposed in [7] is a novel reliable transport protocol for DSA-enabled ad hoc network. It does improve TCP's end-to-end throughput by gathering information from intermediate node and destination node, and keeping updated network status. However, it has some shortcomings. First, it requires a completely new transport protocol, which causes a problem of deployment incompatibility. Secondly, it does not address the issues when a DSA network acts as an access network.

Another work, however, focused on the problem when a DSA network acts as an access network, while failed to apply to an ad hoc network. The main idea of DSASync proposed in [8] is buffering and utilizing a proxy. It adds a DSASync LL component (DSASync\_LL) to Link Layer, gathering all the information related to DSA parameters and provided to upper layer. In this way, the DSASync TCP component (DSASync\_TCP) in transport layer have the awareness of all the DSA operations undertaken, thus could act in advance to ease transmission and prevent packet loss that may occur. Through the information collection in Link Layer and the corresponding action in transport layer, DSASync could improve TCP's end-to-end performance. However, it does have some defect. For example, it's a cross layer implementation, which means it needs information from the under layer.

Along the above works related to TCP end-to-end performance in DSA network, we don't see a lot of discussion about the impact channel switching in DSA could bring on TCP. The TCP could be unable to keep track of the available bandwidth. And when the bandwidth fluctuates greatly, the bandwidth utilization may get even worse. That's the issue this paper's will focus on. Our approach is to periodically measure RTT and throughput, which will be used to indicate the channel switch. And we also calculate the optimal congestion windows of different channel, and apply them to TCP when a switch performed,

through which we can cut off the major time TCP uses to catch up with the bandwidth, and set the congestion window to the optimal value directly. Thus we can obtain the available bandwidth fast enough to fully utilize the channel capacity.

### 3. System Model

As mentioned above, we mainly focused on the issue that TCP may fail to keep track of the available bandwidth when a channel switch is performed between channels with different bandwidths, which would lead to bandwidth under-utilization. The system model is shown in Fig.1.

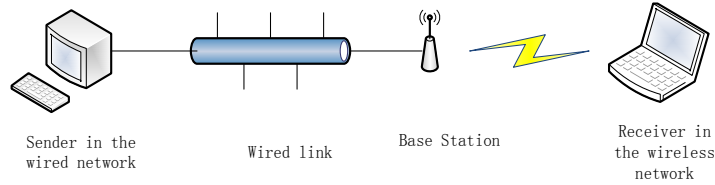


Fig.1: System model

In Fig.1, a base station (BS) connects the wired network and wireless network, which has DSA functionality enabled. We have a sender (SH) lies in the wired network, while receiver (RH) in the wireless network.

In the DSA protocol deployed, we use  $N$  to denote the number of channels with different bandwidths, which is set to two currently. Channel switch is performed after fixed intervals, which is devoted by *beacon\_peroid*. As we mainly focus on the impact caused by channel switch and also for the reason of simplicity, we do not consider PU interpretation at this moment. We summarize the parameters of interest as follows:

- i)  $N$  is number of channels wireless nodes can access.
- ii) *beacon\_peroid* is fixed interval between channel switch performed.
- iii)  $delay_{wd}$  is the delay of wired link in Fig.1.
- iv) *bw\_flag* indicates the current bandwidth status.
- v)  $BW_i$  is the bandwidth of channel  $i$ .

We studied the impact on TCP performance using different parameter combinations, which lie in Section Mechanism. Our approach of improving TCP performance is straightforward. We estimate the time when a channel switch is performed; while at the same time, keeping track of the available bandwidth. So as soon as we find out a channel switch is performed, wet set the congestion window the optimal window we measured, thus cutting off the time TCP would have to take to adjust its window, and improving bandwidth utilization.

The functionality could be divided into two parts, making the judgment when a channel switch is performed and measuring the available bandwidth. Details will be given in Section Mechanism.

## 4. Mechanism

### 4.1 Spectrum switch estimate

We use  $rtt\_avg$  to denote the average RTT since the start of communication between two end users, and  $rtt\_var$  to denote the variance of RTT. Each time RTT is updated, we update them as in Eq.1 and Eq.2:

$$rtt\_avg = sum / rtt\_count \quad (1)$$

$$rtt\_var = rtt\_now^2 / rtt\_count - rtt\_avg^2 \quad (2)$$

$$\begin{cases} rtt\_now > rtt\_avg + 3rtt\_var \\ rtt\_now < rtt\_avg - 3rtt\_var \end{cases} \quad (3)$$

If any of the criterions in Eq.3 returns true, we think that a channel switch is under-taken. Then we set  $bw\_flag$  to the state of current bandwidth.

### 4.2 Measure the bandwidth

Since we have been indicated that the channel switch is performed, we could set the optimal congestion window as soon as we switch to a new channel. The question is how to get the available bandwidth. In our approach, we use the formula in Eq.4 to measure the available bandwidth:

$$opt\_cwnd = pkt\_size \times rtt\_now / ack\_int \quad (4)$$

where  $ack\_int$  is the interval between two new ACKs. And we use  $opt\_cwnd$  to denote the available bandwidth, which is used to set the optimal congestion window.

## 5. Evaluation

We use NS2, with a modified extension to emulate the dynamic spectrum access environment, and implement our algorithms above as a new TCP protocol in NS2, which we call TCP DSA. For simplicity, we set the parameters to make channel switch performed periodically after fixed time intervals devoted by  $beacon\_period$  between two channels with different bandwidths. So after  $beacon\_period$  time interval, nodes which are communicating would switch to another channel with a different bandwidth. The topology is shown in Fig.1. We compare our TCP DSA with TCP-Reno in several scenarios. Simulations below consist of different scenarios and corresponding results which we'll go through in details.

### 5.1 Cwnd and RTT variation

Fig.2 shows the cwnd variation through one communication lasting for 100 seconds, when applied our algorithm. In this scenario the channel switch is performed every 5 seconds between two channels, which have a bandwidth of 25Mb and 9Mb respectively. The initial ssthresh is 1000, and  $delay_{wd}$  is set to 100ms. As we can see in the plot, our algorithm can continually keep track of the available bandwidth such to maintain a high throughput. Fig.3 is a corresponding plot of RTT variation in the very same scenario, from which we can see the periodical variation of RTT every 5 seconds.

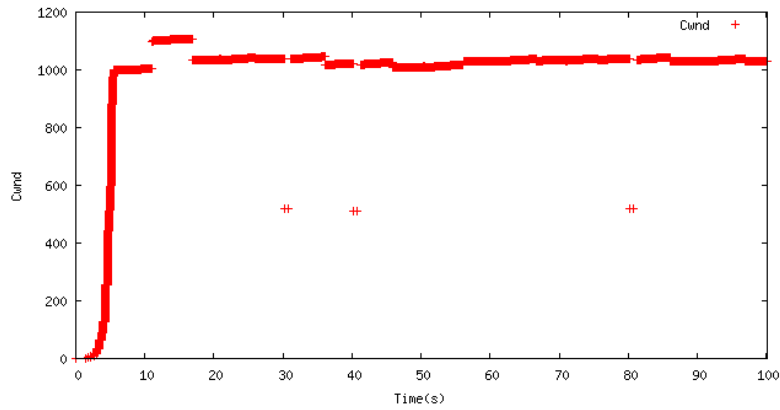


Fig.2: Cwnd variation using TCP DSA

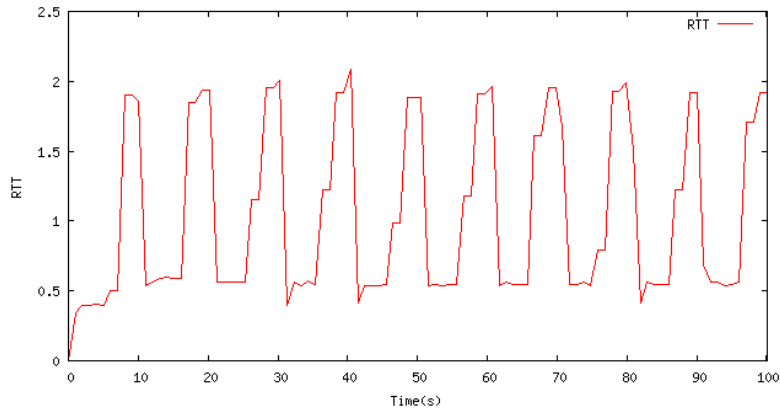


Fig.3: RTT variation using TCP DSA

### 5.2 Throughput under different ssthresh

Fig.4 shows comparison of throughput under different ssthresh, which denotes the slow start threshold. In this scenario, channel switch performs every 5

seconds, with a total simulation time of 100 seconds. As we can see from the figure, TCP DSA achieves better performance under most circumstances. That's because TCP DSA is more sensitive to the bandwidth fluctuation and can grasp the bandwidth more quickly.

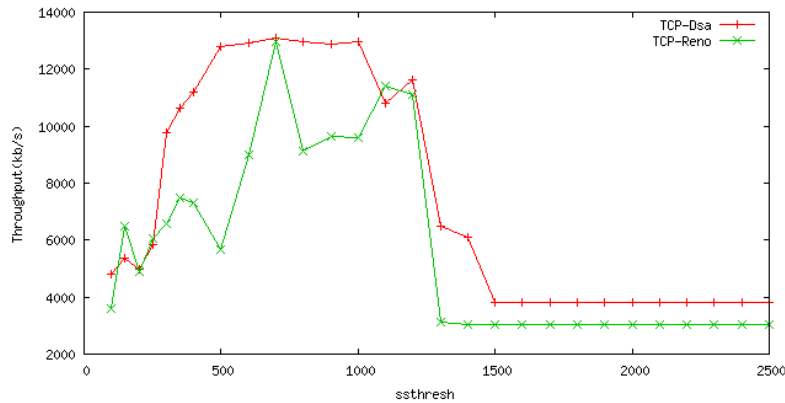


Fig.4: Throughput under different ssthresh

### 5.3 Throughput under different error rate

Fig.5 shows comparison of throughput under different error rate of wireless network. The parameters are the same in last scenario except we employ different error rates. Through the figure, TCP DSA achieves better performance, and at the same time, behaves more steady than TCP-Reno. As we can see in the figure, TCP DSA remains high throughput, under most circumstances.

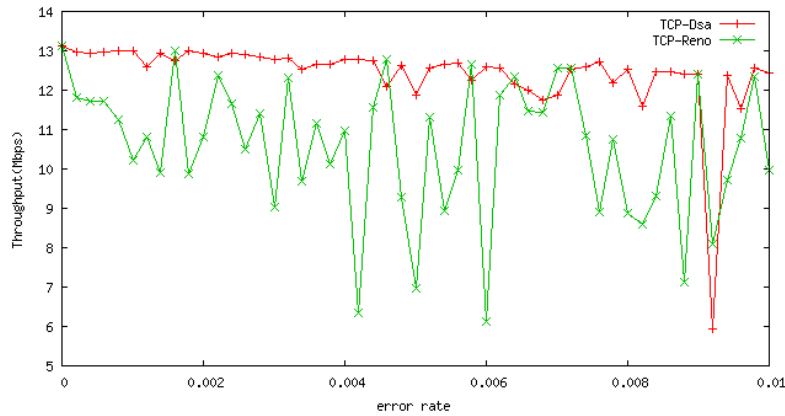


Fig.5: Throughput under different error rate

## 6. Conclusion

In this paper we have studied the impact of spectrum switches in a typical DSA network on TCP throughput performance. To address this issue, TCP DSA is proposed, based on monitoring RTT and throughput variations. We evaluate our protocol using NS2, and the result shows significant improvement. Further research would be focused on a more complex spectrum environment, for example, more available spectra, and more end users, where we would optimize our protocol to accommodate to it.

## Acknowledgements

The work is partly supported by an education commission project under Grant 20110801 and a Youth Program of Science Foundation of Tianjin Medical University, under GRANT. 2013KYQ12. The authors would like to thank the Tianjin Key Laboratory for Intelligence Computing and Novel Software Technology for their support to the work.

## References

- [1] Stevenson, C. R., Chouinard, G., Lei, Z., Hu, W., Shellhammer, S. J., & Caldwell, W. IEEE 802.22: The first cognitive radio wireless regional area network standard. *IEEE Communications Magazine*, 47(1), pp.130-138, 2009.
- [2] Zhao, Q., Tong, L., Swami, A., & Chen, Y. Decentralized cognitive MAC for opportunistic spectrum access in ad hoc networks: A POMDP framework. *Selected Areas in Communications, IEEE Journal on*, 25(3), pp.589-600, 2007.
- [3] Bahl, P., Chandra, R., Moscibroda, T., Murty, R., & Welsh, M. White space networking with wi-fi like connectivity. *ACM SIGCOMM Computer Communication Review*, 39(4), pp.27-38, 2009.
- [4] Jia, J., Zhang, Q., & Shen, X. HC-MAC: A hardware-constrained cognitive MAC for efficient spectrum management. *Selected Areas in Communications, IEEE Journal on*, 26(1), pp.106-117, 2008.
- [5] Liu, J., & Singh, S. ATCP: TCP for mobile ad hoc networks. *Selected Areas in Communications, IEEE Journal on*, 19(7), pp.1300-1315, 2001.
- [6] undaresan, K., Anantharaman, V., Hsieh, H.-Y., & Sivakumar, R. ATP: A reliable transport protocol for ad hoc networks. *Mobile Computing, IEEE Transactions on*, 4(6), pp.588-603, 2005.
- [7] Chowdhury, K. R., Di Felice, M., & Akyildiz, I. F, TP-CRAHN: A transport protocol for cognitive radio ad-hoc networks, in *Proc. 28th IEEE INFOCOM*, pp.2482 - 2490, 2009.
- [8] Kumar, A., & Shin, K. G. DSASync: managing end-to-end connections in dynamic spectrum access wireless LANs. *IEEE/ACM Transactions on Networking (TON)*, 20(4), pp.1068-1081, 2012.