# Research and Simulation of Transport Protocols Optimization on Wireless Multi-Hop Networks

Weifeng Sun, Haotian Wang, Mingchu Li, Mengci Zhou, Zhenquan Qin*
*School of Software, Dalian University of Technology, Dalian 116621*
wfsun@dlut.edu.cn, wanghaotian@mail.dlut.edu.cn, li_mingchu@yahoo.com, zhoumengci@gmail.com, qzq@dlut.edu.cn

***Abstract*—It will encounter problems when traditional transport protocols are applied in wireless multi-hop networks (WMNs). Stream Control Transmission Protocol (SCTP) has the potential to be an alternative transport protocol that may be able to better satisfy the requirements of WMNs than traditional protocols, TCP and UDP. In this paper, we analyze and simulate the impact of some factors on the performance of transport protocols on WMNs. (e.g., hops, CWmin, packet size and maximum window of TCP, chunk size of SCTP). Moreover, via extensive simulations of four transport protocols (TCP, SCTP, UDP and DCCP) in competition scenarios, we analyze the relationship between network performance (e.g., throughput) and network factors. And we propose optimization methods with proper values of factors. The results of simulations also can be used as the foundation of further studies on transport protocols on WMNs.**

***Keywords-wireless multi-hop networks; TCP; SCTP; NS2***

## I. INTRODUCTION

The Stream Control Transmission Protocol (SCTP) [1] provides connection-oriented, reliable end-to-end transmission. It is a message based new transport protocol and has similar flow control, congestion control and retransmission mechanisms to those of TCP which are designed for wired networks.

Wireless multi-hop networks (WMNs) have some differences from wired networks that encounter problems when TCP or SCTP is used on WMNs, such as delay, less bandwidth and higher packet loss rate. When a packet is lost, both TCP and SCTP assume that the loss is due to congestion instead of link break or collision in the radio access medium, then TCP and SCTP will reduce the rate of sending packets [2].

We study the performance of transport protocols with different parameters and find the factors that have impact on the performance of transport protocols on WMNs. We use NS-2 to study the performance of TCP and SCTP over two kinds of topologies. From the results of our simulations, we find that modifying the parameters of protocols can improve the throughput of transport protocols on WMNs and it is significant to find the optimization values of these parameters. Through the simulations of competition, we compare the behaviors of the four transport protocols (TCP, SCTP, UDP, DCCP) when these protocols are used at the same time on WMNs.

We can analyze the performance of transport layer protocols on WMNs via network simulation. The simulation can provide relatively accurate data and information for the development of network scheme and protocols. NS-2 is a discrete event and object-oriented simulator. It supports simulation of wireless multi-hop networks and has good expansion ability, and it has been applied widely by academia and education [3].

The rest of this paper is organized as follows: Section Ⅱ introduces an overview of related work about this paper. The optimization methods of this paper are described in Section Ⅲ. Section Ⅳ gives the configurations of the simulation, which mainly include scenario, settings of related parameters and evaluation metrics of the network performance. We analyze the results in Section Ⅴ and Section Ⅵ. Finally, Section Ⅶ summarizes the paper and our future work.

## II. RELATED WORK

The analysis and optimization of transport protocol performance on WMNs have received numerous researchers' attentions in previous research [4, 5, 6]. In [4], TCP and UDP are studied as different transport layer alternatives and their performance is evaluated with respect to QoS. Hongqiang Zhai et al. [5] present a new wireless congestion control protocol (WCCP) based on TCP's congestion control to efficiently and fairly support the transport service on WMNs. Nelson V. et al. [6] provides an exhaustive performance analysis by simulation of the SCTP transfer protocol in WiMAX and Wi-Fi networks. P. Navaratnam et al. [7] evaluate the performance of DCCP with TCP Friendly Rate Control (TFRC) in wireless mesh networks using NS2 simulations, in terms of fairness and throughput smoothness.

The optional RTS/CTS in IEEE 802.11 MAC layer [8] is a mechanism of avoiding collision, essentially and it can solve Hidden Terminals problem [9] through shaking hands mechanism. And in [10], Zhefu Shi et al. present a new analytical model for multi-hop CSMA networks, using matrix exponential methods that model node queuing and back-off processes.

SCTP inherits the congestion control principles of TCP and uses the SACK extension of TCP which is an option in TCP. The congestion control of SCTP also includes slow start, congestion avoidance and fast retransmission. The major differences between TCP congestion control algorithm and SCTP congestion control algorithm are:

(1) The congestion window (*cwnd*) is suggested to be usually 1 * *MTU* in TCP, which is at least 2 * *MTU* in SCTP.
(2) In TCP, the increment of *cwnd* is controlled in general by the number of new acknowledgement received; while

*Corresponding author: Zhenquan Qin

*cwnd* is controlled by the number of acknowledge bytes in SCTP.

(3) In TCP, fast transmission is triggered by three duplicate ACKs; while in SCTP, the fourth missing report of a chunk triggers fast retransmission.

(4) In TCP, it is optional to be either in slow-start phase or in congestion avoidance phase when the slow-start threshold, *ssthresh* is equal to *cwnd*; while SCTP is required to be in slow-start phase when *ssthresh* is equal to *cwnd*.

(5) TCP has explicit fast recovery algorithm that is unused in SCTP. In SCTP, the parameter *MaxBurst* is used to avoid flooding in the network after fast retransmission.

## III. OPTIMIZATION METHODS

Through changing the parameters of network according to the current situation of network dynamically to improve the throughput of WMNs. So we propose optimization methods below about improving the throughput of transport protocols on WMNs through the simulations of this paper:

(1) In the environment of WMNs of this paper, if TCP is adopted by the transmission between stations, the version of TCP should be DSACK, packet size takes 1024 bytes, maximum window takes 4 and CWmin takes 63 to achieve higher throughput.

(2) If SCTP is applied in WMNs, the chunk size should take 256 bytes and CWmin takes 127 to reach higher throughput.

(3) When we choose a transport protocol of reliable transmission to compete with UDP on WMNs, SCTP should be our good choice and it can make better use of the bandwidth of WMNs. Compared with TCP, the throughput increases 23.08% by SCTP.

(4) When DCCP is working on WMNs, if it is detected that TCP is also working in the same scenario, DCCP should adopt the congestion control algorithm of CCID3 to make the competition between protocols fairer and the throughput of TCP increases 80.24% compared with CCID2; while if it is detected that SCTP is working at the same time, DCCP should adopt CCID2 and the throughput of DCCP increases 137.84% compared with CCID3. The process of optimization method can be seen in Fig. 1.

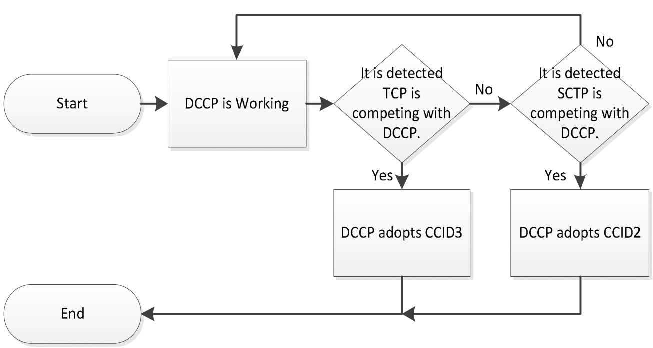



Figure 1. Process of optimization method about DCCP.

## IV. SIMULATION SETTINGS AND METRIC

### A. Simulation Scenario

Fig. 2 is a string topology that is used in some of the simulation studies. It consists of 8 hosts which are static in the 2000m*2000m playground. The maximum transmission range of hosts is 250m and the distance between any two neighboring hosts is 200m, which allows a host only can communicate to its neighboring host directly.


Figure 2. String topology.

Fig. 3 shows another topology used in our simulation. Host 0 and host 1can communicate to host 2 directly, while host 3 is out of the maximum transmission range of host 0 and host 1. This topology is used to study the competition of two different transport protocols (TCP/SCTP, TCP/UDP, SCTP/UDP, DCCP/TCP and DCCP/SCTP) used at the same time in a scenario.

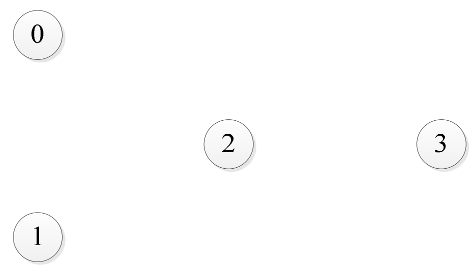



Figure 3. Competition of transport protocols topology.

### B. Parameters Configuration

Our simulator is NS-2.35 that supports several kinds of transport protocols, like TCP, UDP, SCTP and DCCP. The data traffic loads between stations are generated by data traffic generator which adopts FTP over TCP, SCTP and DCCP, and CBR (Constant Bit Rate) over UDP. The settings of parameters in our simulation are shown in Table Ⅰ.

In the simulations of this paper, we use IEEE 802.11b DCF with RTS/CTS mechanism to reduce the influence of imperfect MAC layer on the protocols of other layers.

TABLE I. PARAMETERS SETTING

| | |
|---|---|
| Channel type | Wireless channel |
| Radio propagation model | TwoRayGround |
| Network interface | WirelessPhy |
| MAC type | 802.11 |
| Interface queue type | PriQueue |
| Link layer type | LL |
| Antenna type | OmniAntenna |
| Max packets in ifq | 50 |
| Route protocol | DSR |
| RTS/CTS | On |
| Scenario size | 2000m*2000m |
| Bandwidth | 2Mbps |
| Transmission Range | 250m |
| CWmin | 31 (default) |

The representative factors which have impacts on the performance of transport protocols in our simulation are: hops, CWmin, packet size and maximum window of TCP, chunk size of SCTP.

### *C. Evaluated Metrics of Network Performance*

In order to evaluate the efficiency, real-time ability and reliability of network accurately, we choose throughput and packet delivery rate as the evaluated metrics of network performance.

The throughput's unit kbps corresponds to the amount of data in bits that is transmitted over the channel per unit time and it is the main metric for evaluating the performance of network,

$$Thoughput = \frac{Total\ number\ of\ bits\ successfully\ transmitted\ during\ T}{T}$$

where the T is a period of time which we want to calculate the throughput of all data flows in the application layer.

Packet delivery rate can reflect the reliability of network. Packet delivery rate is the percentage of successfully received packets' number by the number of total sent packets.

$$Packet\ Delivery\ Rate = \frac{Number\ of\ \ packets\ received}{Number\ of\ \ packets\ sent} \times 100\%$$

## V. SIMULATION RESULTS AND ANALYSIS

In this section, we analyze the results of our simulations and these results well validate the optimization methods which are proposed in Section Ⅲ. When a specific factor is examined, the other parameters and factors are configured as default values.

### *A. Influence of Maximum Window of TCP*

The maximum window (*win*) indicates the maximum sending rate of sender. Overlarge window will aggravate the confliction in MAC layer; while too small window will degrade the utilization of bandwidth. To find the optimum *win* of TCP, we design a simulation. The topology of this simulation is string topology and we take four kinds of TCP which are NewReno, SACK (Selective ACK), DSACK (SACK + DA) and Vegas. We vary *win* from 1 to 32 with increments 1, 2, 4, 8, 16 and 32. The packet size takes 1024 bytes. The total time of simulation lasts 120 seconds.

Fig. 4 shows the results for the simulation between host 0 and host 4. As shown in Fig. 3, the throughputs of NewReno, SACK and Vegas are almost the same when *win* is equal to 1 or 2; while the throughputs of NewReno and SACK decrease with the increment of *win* with the reason of severer conflict problem. Due to the special increment and decrement strategy of window in Vegas, Vegas always keeps the window to a break-even point according to current network situation. So the throughput of Vegas does not change with *win*. On the other hand, the throughput of DSACK is very low when *win* is equal to 1. The reason is that the sender will not transmit next packet until the corresponding ACK of last sent packet is received. While receiver that adopts DA (Delayed ACK) mechanism will not transmit ACK until it receives the next packet or a fixed timer (100ms) is timeout. So the sender needs waiting for 100ms to transmit the next packet and the performance of DSACK is poor when *win* is 1. While the throughput of DSACK is higher than that of the other three versions of TCP when *win* is equal to 2 or 4 and the throughput of DSACK reaches maximum value when *win* is 4.

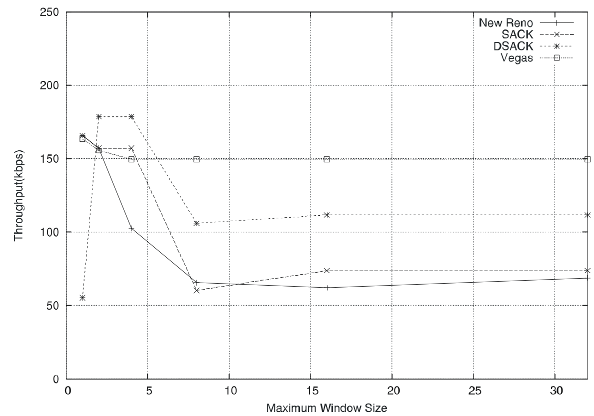
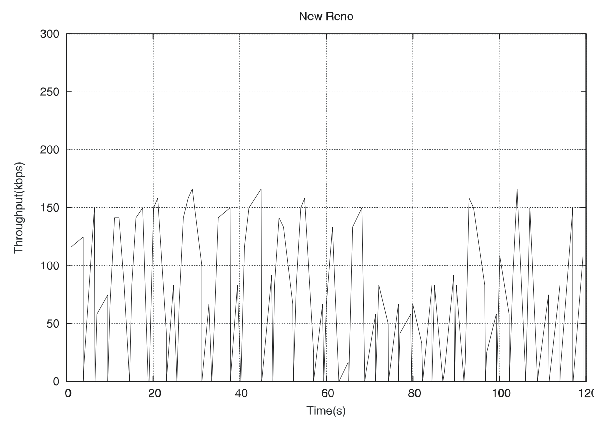
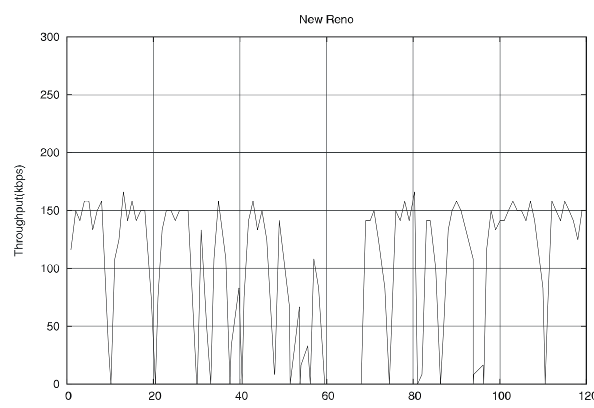



Figure 4. Throughput with different maximum windows of TCP.




(a) Throughput of NewReno *win*=32 *s*=1024B.




(b) Throughput of NewReno *win*=4 *s*=1024B.

Figure 5. Instantaneous throughput of NewReno with different *win*.

Fig. 5 shows the instantaneous throughput of NewReno with packet size (*s*) of 1024 bytes when *win* is equal to 32 and 4. As shown in Figures 5 (a) and 5 (b), we can see that overlarge *win* will lead to an overquick sending rate and increase the probability of conflict that will excite the instability of TCP; however proper *win* (4 in our simulation) will reduce the time of the throughput dropping to 0 and it will improve the stability of TCP.

### B. *Influence of Packet Size of TCP*

We vary packet size of TCP from 64 to 1024 bytes with increments 64, 128, 256, 512 and 1024. And the version of TCP is DSACK. The other settings are the same as the previous simulation.

From the result of Fig. 6 we can find that the throughput increases with the increment of packet size at the same hop. When the hops are smaller (1~3 hops), increasing packet size will increase the throughput effectively; while the impact of increment of packet size on the throughput is reduced when the hops are larger. The reason for this phenomenon is that the conflict of packet and Hidden Terminals problem are not very severe when the hops are smaller. At this time, increasing packet size will not increase the probability of packet conflict but improve the proportion of valid data, and then improve the throughput; however, overlarge packet size will increase the time of transmission and then increase the probability of packet conflict which leads to packet loss. At this time, the station will reduce the sending rate and throughput of TCP degrades. So the improvement of throughput by increasing the packet size will decrease greatly.

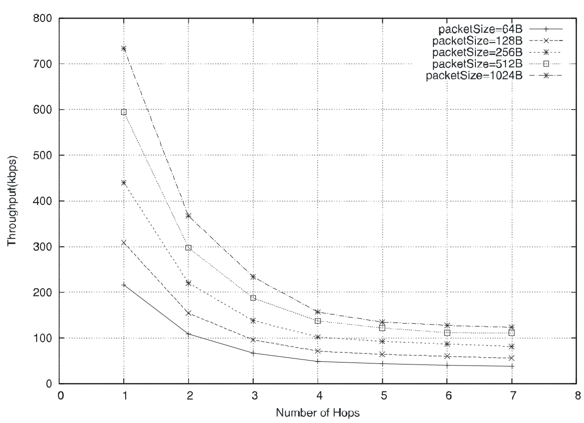



Figure 6. Throughput with different packet sizes of TCP.

Fig. 7 shows the instantaneous throughput of NewReno with packet size (*s*) of 512 bytes and *win* of 4. After comparing the result of Fig. 7 with Fig.5 (a), we can see that decreasing packet size will improve the stability of TCP, however the average throughput deceases. So we could choose larger packet size of TCP (1024B in our simulation) on WMN to maximize the througput of TCP. But the stability of TCP is sacrificed for increasing throughput.

### C. *Influence of Chunk Size of SCTP*

We vary chunk size (*Cs*) of SCTP from 64 to 1024 bytes with increments 64, 128, 256, 512 and 1024. Fig. 8 and Table Ⅱ show the throughput as a function of hops for different chunk sizes.

From the result we can find that the throughput decreases with the increment of chunk size when the hops are smaller (1 or 2 hops); while the throughput fluctuates with the increment of chunk size and reaches the maximum value with a certain chunk size. The reason for this phenomenon is that the smaller chunk size is, the more chunks can be bundled in one packet. So one packet could take more valid data and then the transmission efficiency will be improved when the conflict in network is not severe, so the throughput increases; while with the increment of hops, the conflict probability of larger packet is higher than that of smaller packet. So the advantage of taking more valid data may be partly offset by the conflict. From the result in Table Ⅱ,we can find that the throughput is relatively high when chunk size is equal to 256 bytes no matter how many the hops are.

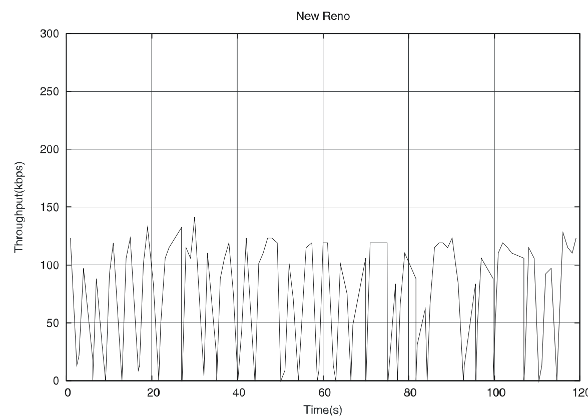



Figure 7. Throughput of NewReno *win*=32 *s*=512B.

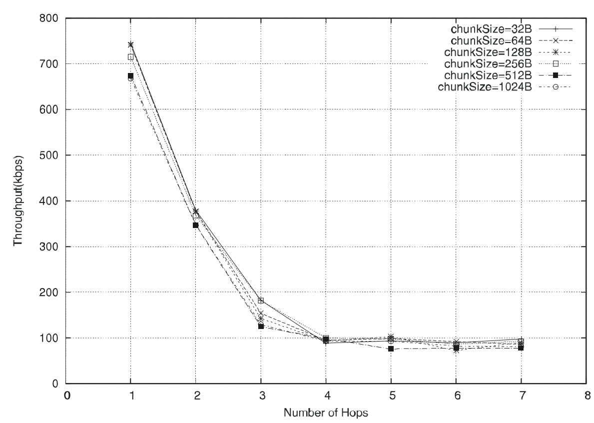



Figure 8. Throughput with different chunk sizes of SCTP.

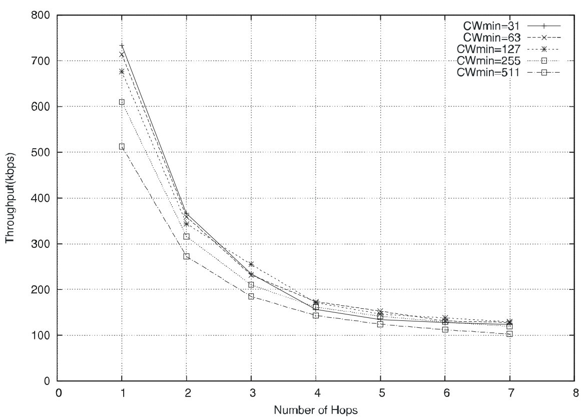
TABLE II. THROUGHPUT WITH DIFFERENT CHUNK SIZES OF SCTP

| hops | *Cs*= 32B | *Cs*= 64B | *Cs*= 128B | *Cs*= 256B | *Cs*= 512B | *Cs*= 1024B |
|---|---|---|---|---|---|---|
| 1 | 745.08 | 741.30 | 741.32 | 714.81 | 673.94 | 668.03 |
| 2 | 379.51 | 377.32 | 377.32 | 367.85 | 347.49 | 346.79 |
| 3 | 182.69 | 154.08 | 142.56 | 182.11 | 125.03 | 129.75 |
| 4 | 88.92 | 95.04 | 93.70 | 100.24 | 98.21 | 93.14 |
| 5 | 93.55 | 99.19 | 102.84 | 98.42 | 76.17 | 92.89 |
| 6 | 89.36 | 92.18 | 72.90 | 86.53 | 78.37 | 72.90 |
| 7 | 97.86 | 85.97 | 90.40 | 93.00 | 77.98 | 80.79 |

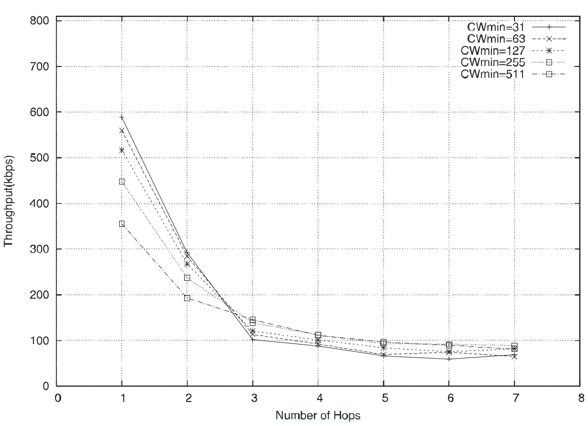
## D. Influence of CWmin

CWmin is the initial value of back-off window in IEEE 802.11 MAC layer and its default value is 31 in IEEE 802.11b. Fig. 8 shows the throughput as a function of hop for different CWmin values that include 31, 63, 127, 255 and 511.

From the result of Fig. 9 (a) and Table Ⅲ we can find that the throughput of TCP decreases with the increment of CWmin when the hops are smaller (1~3 hops) and the reason is that it is a waste of time to increase the back-off window when the network condition is not very severe, and then the throughput degrades. When the hops are larger, increasing CWmin will improve the throughput comparing with CWmin of default value, and it gives relatively stable increment of the throughput when CWmin is equal to 63. That is because reasonable back-off window will reduce the probability of waste of RTS because of conflict and reduce the times of failure channel access for stations greatly. So the stability of TCP can be improved and the throughput increases.




(a) Throughput with different CWmin values of TCP.




(b) Throughput with different CWmin values of SCTP.

Figure 9. Throughput with different CWmin values of TCP and SCTP.

As shown in Fig .9(b) and Table Ⅳ, the behavior of SCTP throughput is similar to that of TCP which can be seen in Fig. 9 (a) and Table Ⅲ, and the reasonable CWmin for SCTP is 127. Compared with the average throughput of TCP, we can find that the average throughput of SCTP is lower. The reason is that DSACK adopts DA mechanism and almost half the selective ACKs. This mechanism is extremely advantageous for the WMNs which have severe conflict in the network; while SCTP still adopts SACK mechanism that has higher probability of packet conflict.

TABLE III. THROUGHPUT WITH DIFFERENT CWMIN VALUES OF TCP

| hops | $CW_{min}$ =31 | $CW_{min}$ =63 | $CW_{min}$ =127 | $CW_{min}$ =255 | $CW_{min}$ =511 |
|---|---|---|---|---|---|
| 1 | 733.82 | 713.79 | 676.35 | 609.93 | 512.44 |
| 2 | 367.19 | 359.15 | 343.69 | 316.02 | 272.41 |
| 3 | 234.21 | 231.85 | 225.19 | 210.43 | 184.84 |
| 4 | 156.76 | 173.61 | 171.25 | 161.62 | 143.52 |
| 5 | 134.94 | 153.09 | 146.50 | 141.72 | 124.38 |
| 6 | 127.87 | 131.25 | 138.25 | 129.10 | 112.39 |
| 7 | 123.91 | 128.21 | 129.51 | 119.95 | 102.96 |

TABLE IV. THROUGHPUT WITH DIFFERENT CWMIN VALUES OF SCTP

| hops | $CW_{min}$ =31 | $CW_{min}$ =63 | $CW_{min}$ =127 | $CW_{min}$ =255 | $CW_{min}$ =511 |
|---|---|---|---|---|---|
| 1 | 588.72 | 559.52 | 516.58 | 447.75 | 355.59 |
| 2 | 292.96 | 284.33 | 267.14 | 236.93 | 193.27 |
| 3 | 101.84 | 113.51 | 120.95 | 138.61 | 145.39 |
| 4 | 87.66 | 92.33 | 101.08 | 112.25 | 111.45 |
| 5 | 66.30 | 69.27 | 83.97 | 93.01 | 96.44 |
| 6 | 59.43 | 74.39 | 75.67 | 91.76 | 89.44 |
| 7 | 68.67 | 64.70 | 81.66 | 88.69 | 81.32 |

# VI. SIMULATION OF COMPETITION BETWEEN TRANSPORT PRTOCOLS

## A. Competition of TCP and SCTP

To study the performance of two different transport protocols working at the same time on WMNs, we design the simulations of competition between different transport protocols. The topology of simulations in the rest of this paper can be been in Fig. 2. And the settings of transport protocols adopt the optimized values that are set from our previous simulations in this paper. The version of TCP is DSACK, packet size is 1024 bytes and maximum window is 4; the chunk size of SCTP is 256 bytes. The data traffic over TCP and SCTP are both FTP. CBR flow adopts UDP, the packet size of CBR is 1024 bytes and the send rate is 400kB/s.

The total time of simulation lasts 120 seconds, and a SCTP association is established between host 0 and 3 at 0s; a TCP connection is established between host 1 and host 3 at 0s. Fig. 10 shows the instantaneous throughput of TCP and SCTP.

From Fig. 10 we can find that the throughput of TCP is higher than that of SCTP in the initial stage (0~4s); while the throughput of SCTP is higher afterwards. The average throughput of SCTP reaches 325.38kbps while TCP only reaches 42.85kbps. That is because the value of back-off window will be doubled when a station fails to access the channel in IEEE 802.11 MAC layer, so the back-off is always beneficial to the station which transmits packets later. In the competition of TCP and SCTP, due to “four times handshake” of SCTP and “three times handshake” of TCP, TCP tries to access the channel and transmit packets first, and it leads to the higher throughput of TCP than that of SCTP; while the lagging transmission of SCTP is favorable for SCTP to access the channel and it makes the throughput of SCTP higher than that of TCP in subsequent time due to the unfairness of TCP.

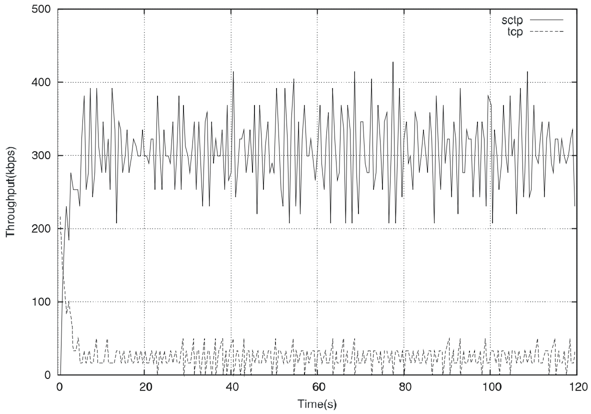



Figure 10. Instantaneous throughput of TCP and SCTP.

### B. Competition of TCP and UDP

A CBR flow is established between host 0 and 3 at 0s; a TCP connection is established between host 1 and host 3 at 0s Fig. 10 shows the instantaneous throughput of TCP and UDP.

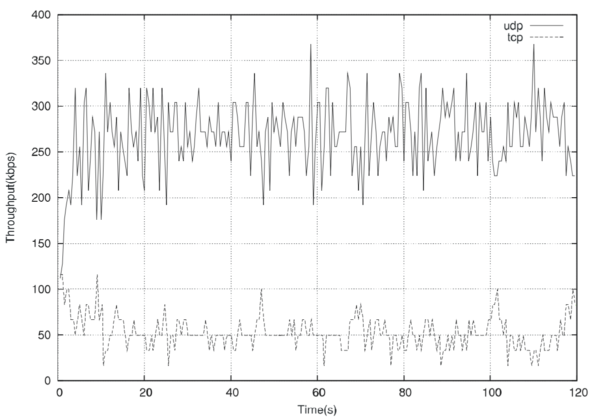



Figure 11. Instantaneous throughput of TCP and UDP.

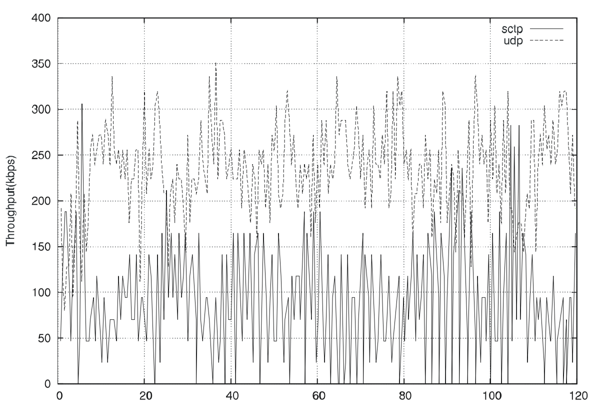



Figure 12. Instantaneous throughput of SCTP and UDP.

As shown in Fig. 11 the throughput of UDP is always higher than that of TCP in the whole process of simulation and the average throughput of UDP reaches 270.53kbps while TCP only reaches 89.86kbps. The reason is that UDP will try to transmit datagrams which are submitted by applications and flows of UDP will take up the resource of bandwidth largely. So the throughput of TCP decreases. However, the packet delivery rate is only 69.25%, and that is because UDP does not adopt congestion control mechanism and will not retransmit the lost packets.

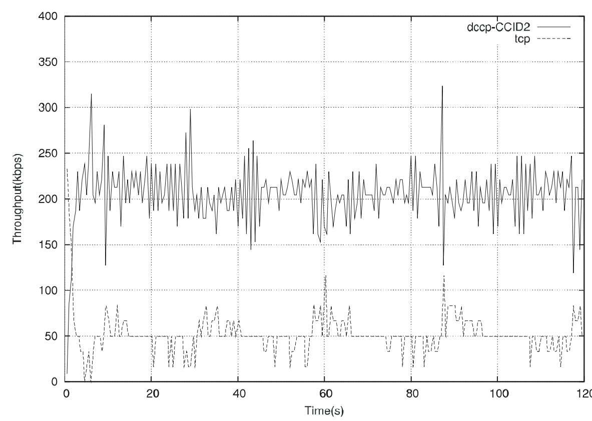
### C. Competition of SCTP and UDP

A CBR flow is established between host 0 and 3 at 0s; a SCTP association is established between host 1 and host 3 at 0s. Fig. 12 shows the instantaneous throughput of SCTP and UDP.

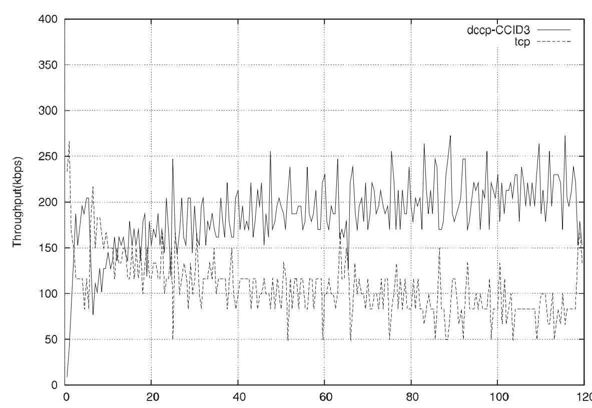
In Fig. 12, the throughput of UDP is higher than that of SCTP at the most of time. The average throughput of UDP reaches 250.93kbps while SCTP reaches 110.60kbps. Compared with the result of Fig. 10, we can find that SCTP is more competitive than TCP when SCTP tries to compete for bandwidth with UDP.

### D. Competition of DCCP and TCP

A FTP flow that adopts DCCP is established between host 0 and host 3 at 0s and the window of DCCP is 1000; a TCP connection is established between host 1 and host 3 at 0s. Fig. 13 shows the instantaneous throughput of TCP and DCCP with different congestion control algorithms.




(a) DCCP with CCID2.



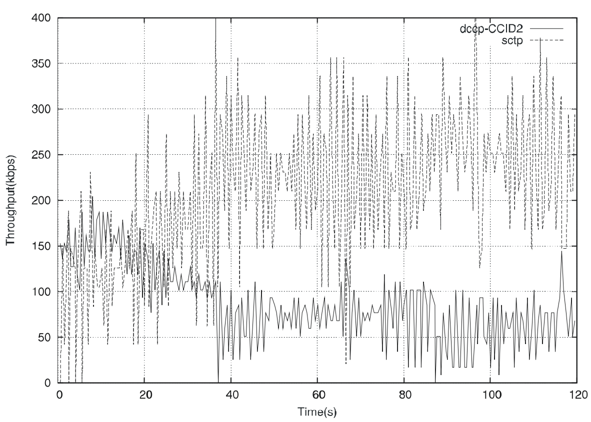

(b) DCCP with CCID3.

Figure 13. Instantaneous throughput of DCCP and TCP.

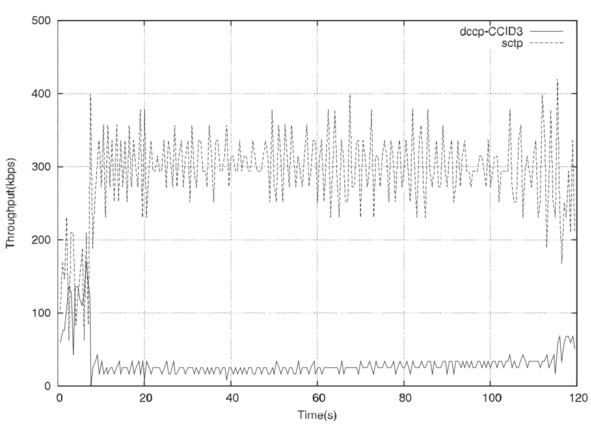
As shown in Fig. 13 (a), the throughput of TCP is higher than that of DCCP in the initial stage (0~2s); while the throughput of DCCP is higher afterwards. The average throughput of DCCP reaches 213.73kbps while TCP only reaches 68.43kbps. This is explained by the fact that DCCP with the congestion control algorithm of CCID2 will encounter the problem of unfairness which is familiar with TCP. However, as shown in Fig. 13 (b), the average throughput of DCCP is just a litter higher than that of TCP and the throughput of them are 195.53kbps and 123.34kbps respectively. That is because DCCP with CCID3 is TCP-friendly and provide more stable data flow. So the competition between DCCP and TCP is fairer when DCCP adopts CCID3.

### *E. Competition of DCCP and SCTP*

A FTP flow that adopts DCCP is established between host 0 and 3 at 0s and the window of DCCP is 1000; a SCTP association is established between host 1 and host 3 at 0s. Fig. 14 shows the instantaneous throughput of SCTP and DCCP with different congestion control algorithms.




(a) DCCP with CCID2.




(b) DCCP with CCID3.

Figure 14. Instantaneous throughput of DCCP and SCTP.

As shown in Fig. 14 (a), the throughput of DCCP is higher than that of SCTP during the period of 0~20s; while the throughput of SCTP is higher afterwards. The average throughput of DCCP only reaches 96.66kbps and SCTP reaches 228.55kbps. The reason for this phenomenon is that DCCP with CCID2 is similar to TCP, so it will encounter the same problem that also occurs in the competition of TCP and SCTP when DCCP with CCID2 tries to compete with SCTP. However, as shown in Fig. 14 (b), the average throughput of DCCP is just 40.64kbps and is far below the average throughput of TCP 312.52kbps. So the performance is very poor when DCCP with CCID3 tries to compete with SCTP.

The results of our simulation in Section Ⅴand Ⅵ present the impact of different factors on the performance of transport protocols on WMNs. The values of these factors can be chosen optimally to increase the throughput and DCCP can switch congestion control algorithm by detecting the kind of data flows' transport protocols on WMNs to increase throughput. However, these optimal values of the factors are inconsistent with each other sometimes, so some tradeoffs must be made when considering the chosen of the factors' optimal values.

## VII. CONCLUSION

In this paper, we analyze the relationship between network performance and network factors in terms of the evaluation metrics of throughput. And we put forward optimization methods about throughput of transport protocol on wireless multi-hop networks. According to the results of simulation, our optimization methods can improve the throughput of network better. To compare with previous researches, we take account into more factors of network and scenarios of competition between several transport protocols. The results of our simulation basically correspond with the conclusions of pervious researches. The analysis of results shows that changing the values of network factors will improve the performance of transport protocols on WMNs.

In future work, we will take account into more factors which also affect the performance of transport protocols on WMNs, e.g. taking account into multi-home and multi-stream of SCTP to improve the performance of SCTP. Improving our optimization methods also will be our future work.

## ACKNOWLEDGMENT

This work is supported in part by Natural Science Foundation of China under grant No. 61103233, NSFC-JST under grant No.51021140004, the Fundamental Research Funds for Central Universities (DUT12JR08).

## REFERENCES


[1] Steward, R., Xie, Q., "Stream Control Transmission Protocol", RFC2960, 2000

[2] Nelson, V., Tania, J., and Yezekael, H., "Performance of SCTP in Wi-Fi and WiMAX networks with multi-homed mobiles," Proceedings of the 3rd International Conference on Performance Evaluation Methodologies and Tools, 2008

[3] Issariyakul, T. and Hossain, E., "Introduction to network simulator NS2," Springer Verlag, 2011

[4] Giannoulis, S., Antonopoulos, C., Topalis, E., Athanasopoulos, A., Prayati, A. and Koubias, S., "TCP vs. UDP Performance Evaluation for CBR Traffic On Wireless Multihop Networks," Simulation, Vol. 14, pp. 43, 2009

[5] Zhai H., Chen X. and Fang Y., "Improving transport layer performance in wireless multihop ad hoc networks by exploiting MAC layer

information," IEEE Transaction on Wireless Communications, Vol. 6, No.5, pp. 1692-1701, 2007

[6] Nelson V., Tania J. and Yezekael H., "Performance of SCTP in Wi-Fi and WiMAX networks with multi-homed mobiles," ICST, 2008

[7] Navaratnam P., Akhtar N. and Tafazolli R., "On the performance of DCCP in wireless mesh networks," Proceeding of the 4th ACM International Workshop on Mobility Managemnent and Wireless Access, 2008

[8] Hiertz, G., Denteneer, D., Stibor, L., Zang, Y., Costa, X.P. and Walke, B., "The IEEE 802.11 universe," IEEE Communications Magazine, Vol. 48, No. 1, pp. 62-70, 2010

[9] Youngsoo Kim, Jeonggyun Yu and Sunghyun Choi, "A novel hidden station detection mechanism in IEEE 802.11 WLAN," IEEE Communications Letters, Vol. 10, No. 8, pp. 608-610, 2006

[10] Shi, Z., Beard, C. and Mitchell, K., "Tunable traffic control for multihop CSMA networks," 2008 IEEE Military Communications Conference (MILCOM '08), San Diego, November 2008