

A COMPARISON OF CONGESTION CONTROL VARIANTS OF TCP IN REACTIVE ROUTING PROTOCOLS MANET

Mahmoud Abu-Zant and Dr. Mohammad Hamarsheh

Department of computer science, faculty of engineering and information technology, Arab American University, Jenin, Palestine

ABSTRACT

A widely used TCP protocol is originally developed for wired networks. It has many variants to detect and control congestion in the network. However, Congestion control in all TCP variants does not show similar performance in MANET as in wired network because of the fault detection of congestion. In this paper, we do a performance comparison between TCP variants NEW RENO, SACK and Vegas in AODV and DSR reactive (On-Demand) routing protocols. Network traffic between nodes is provided by using File Transfer Protocol (FTP) application. Multiple scenarios are created and the average values of each performance parameter are used to evaluate the performance. The results show that TCP variants perform better in terms of throughput and Packet drop with DSR routing protocol compared with AODV routing protocol. TCP variants show a lower Jitter in AODV compared with DSR.

KEYWORDS

MANET, Congestion control, TCP Sack, TCP Vegas, TCP New Reno, AODV, DSR, Jitter, Throughput.

1. INTRODUCTION

The technology of Mobile Ad hoc Network (MANET) is widely used these days, and it's considered as a hot topic in network filed. Each node in MANET works as router and client. Nodes communicate with each other through wireless links like radio or microwave with no fixed infrastructure to control communication between them.

Nodes in MANET are moving randomly and send packets to each other instead of depending on the router in coordinating the flow of packets in the network. In single-hop, nodes are connected directly while in multi-hop the connection is sequential with the help of an intermediate node using ad-hoc routing protocols. Ad-hoc routing protocols are used to determine routes packets should travel through. Every node senses the broadcasting from its neighbors to establish the connection.

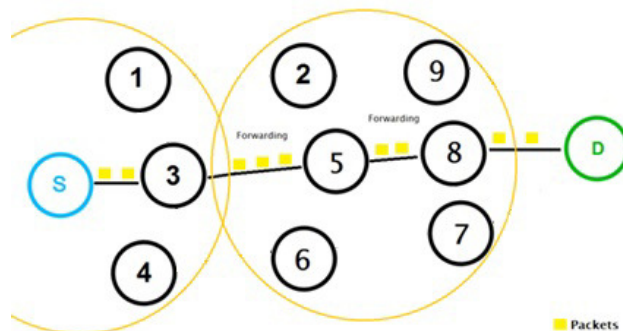


Figure 1. Example of MANET

The Throughput of these wireless links is less than that of the wired links because the capacity of these links is affected by noise, interference, and multipath in addition to congestion. Transmission control protocol (TCP) used in file transfer has many features like flow control, reliability and congestion control. Congestion control used in TCP is very important to avoid long delays and packet loss. Congestion occurs when a network link carries numerous data that causes queuing, dropping of packets and blocking new connections lead to the reduction of throughput and affect the network performance. TCP variants for congestion control have proved its performance in wired network unlike in MANET. Because there are many reasons for packet loss in MANET in addition to network congestion, like the effect of fading, interference from other devices, noise, multipath propagation and link failure in addition to network congestion[12]. This causes a fault detection of congestion at the nodes. These false determinations of packet loss drive TCP to call congestion control algorithm which decreased the Throughput of the network.

The performance parameters of MANET are also affected by the routing protocol used. Ad hoc On-Demand Distance Vector (AODV) and Dynamic Source Routing (DSR) are examples of On-Demand routing protocols. On-Demand protocol finds the route to the destination node by broadcasting route request (RREQ) packets in the entire network when the route is needed. Another type of protocols is Table-Driven protocols like Destination Sequence Distance Vector (DSDV) which maintain a route for each node in its table. Hybrid protocols combine both schemas of On-Demand and Table-Driven protocol [14]. TCP New Reno improves the performance of TCP Reno. TCP New Reno can detect multi-packets loss better than TCP Reno by modifying the fast recovery algorithm. The sender must receive an acknowledgment for every packet that is sent in order to exit fast recovery algorithm phase, which overcomes the TCP Reno problem of reducing the CWND multiple times. TCP New Reno suffers from the need for one Round Trip Time (RTT) to detect and recover a packet loss.

TCP SACK (Selective Acknowledgment) is a modification of TCP Reno, it avoids cumulative acknowledgments and uses selective acknowledgments. The sender should only acknowledge the lost or out of order packets. SACK sender maintains the information which packets is missed at receiver and only retransmits these packets.

TCP Vegas is another modification of TCP Reno, it does not need multiple acknowledgments to detect a packet loss, and it does not rely on packet loss to detect congestion. TCP Vegas can detect network congestion before a packet loss received, because it is developed using proactive measure techniques to counter network congestion, and also it has a modified slow start algorithm which avoids congesting the network.

In this paper, the performance of MANET is studied under a combination of TCP variants and different on-demand routing protocols. Three variants of TCP have been studied: TCP New Reno, TCP Sack, and TCP Vegas, along with two on-demand routing protocols AODV and DSR. The rest of the paper is organized as the follows: Section 2 presents a related work, Section 3 presents the methodology, Section 4 shows the Results and discussion and finally a conclusion is shown in section 5.

2. RELATED WORK

Congestion control in MANET has been studied widely as an interesting research topic in the network field. Researchers provided many comparisons between TCP variants in MANET network, some of these comparisons are summarized in this section.

Poonam Tomar et. al. [4] performed a comparison between the performance parameters of MANET network under TCP variants (TCP Tahoe, TCP Lite, and TCP Reno) using DSR routing protocol. He used NS2 as the simulation tool to study some performance parameters like packet

loss, received bytes and Throughput when using FTP as network traffic with a different number of nodes. Some of TCP Protocols showed a high performance under some conditions. The conclusion of this comparison shows that none of the TCP variants can overcome the congestion of the network, each protocol can perform better under specific conditions. TCP Vegas and TCP SACK under AODV routing protocol are not within the scope of this paper.

Yuvaraju B N et. al. [5] performed a comparison between the performance of six TCP variants TCP TAHO, TCP RENO, TCP New Reno, TCP SACK, TCP FACK and TCP Vegas in MANET with AODV routing protocol under different scenarios and different environmental parameters using the NS2 simulator. The results show that TCP Vegas performed better than the other TCP variants in data transmission. However, no information provided about the performance of these variants with DSR routing protocol.

Neha Arora [6] compared the behavior of four TCP congestion variants TCP Tahoe, TCP Reno, TCP New Reno, and TCP Sack under three routing protocols AODV, TORA, and OSLR in MANET using OPNET simulator, the results showed that the behavior of TCP variants is better under AODV routing protocol. TCP Vegas and DSR protocol are not included in this study.

Suneel et. al. [7] performed a comparison between the performance of six TCP variants TCP TAHO, TCP RENO, TCP New Reno, TCP SACK, TCP Vegas and TCP FACK in MANET with AODV routing protocol the results are similar to the results of [4]. However, no information is provided about the performance of these variants with DSR routing protocol.

M.Jehan et. al. [8] compared the behavior of three TCP congestion variants TCP Binary Increase Congestion Control (TCP BIC), SCALABLE TCP and TCP Vegas under DSR and DSDV in MANET using NS2 simulator. The results showed that SCALABLE TCP showed the highest Throughput and TCP Vegas showed a better round-trip delay in DSDV. the performance under AODV protocol is not included.

Iffat Syad, et. al. [9] compared the performance of TCP Vegas and TCP New Reno in both DSDV and DSR routing protocols in MANET using NS2 simulator. The results showed that the performance of TCP variants in proactive DSDV routing protocol has a higher Throughput than reactive DSR routing protocol, but with a higher packet drop rate and delay. AODV routing protocol was not used in this study.

Hrituparna Paul et. al. [10] compared TCP variants (TCP RENO, TCP New RENO, and TCP TAHOE) in MANET when different routing protocols are used (AODV, TORA, and DSR) in two scenarios using OPNET simulator. The first scenario uses three nodes and the second one uses five nodes. The throughput of each TCP variant in these scenarios is analyzed. The results show that AODV performed a higher throughput than DSR and TORA while increasing the number of nodes. TCP Vegas and TCP SACK are not used in this study.

3. METHODOLOGY

The simulation was performed using NS2 simulator to compare the performance of TCP variants New Reno, Sack, and Vegas with reactive routing protocols (DSR and AODV) in MANET. The traffic is provided by using file transfer protocol (FTP) between 10 different nodes. The number of nodes is a design parameter and varies from 20 to 60 nodes. Each node is located randomly within 850x850 m. Speed varies from 1 to 20 sec with 5s pause time. Each scenario is executed three times and the average of each parameter is used to evaluate the performance. The multiple executions are used to take the average behavior of each protocol and to remove the effect of randomness that may affect the results. Table 1 below summarizes the environmental parameters of the simulation scenario. Throughput, Jitter and packet drop was monitored to measure the network performance with increasing number of nodes.

A CMU tool uses a random waypoint model to generate the random movement of nodes. ‘*setdest*’ is the command used to generate the positions of nodes, their moving speeds and moving directions.

Coordination of the terrain has been set to 850 x 850 in order to simulate a big space for mobile nodes to move freely and to test the routing protocols used with the low coverage of mobile nodes.

In the experiment, we compared the performance of TCP variants in DSR and AODV routing algorithms. Both AODV and DSR are on-demand routing protocol. This means that the route between source and destination is obtained when it is needed. When a source node needs to communicate with destination node in DSR protocol, the source node broadcast a Route Request Packet (RREQ). Each node receives RREQ rebroadcasts the RREQ unless it’s the destination node or it has a route to the destination node. When an intermediate node which has a route to the destination or it is the destination itself receives an RREQ, it sends the RREP to the source node which contains a full path from source to destination in the header of the packet.

AODV routing protocol combines the advantages of both DSDV and DSR protocols. It uses the same mechanism of flooding to obtain a route to the destination node as used in DSR, and it uses also a sequence number and hop-by-hop routing similar to DSDV. The use of sequence number helps to avoid infinite loops and to determine the age of the route. AODV uses a routing table to determine a path from the source to the destination. DSR caches the route in the packet header but AODV stores the route in the nodes tables.

Type	J R G D U	Reserved	Hop Count	Type	R A	Reserved	Prefix Sz	Hop Count
RREQ ID				RREQ ID				
Destination IP Address				Destination IP Address				
Destination Sequence Number				Destination Sequence Number				
Originator IP Address				Originator IP Address				
Originator Sequence Number				Lifetime				

Figure 2. RREQ and RREP Packet Format in AODV

Option Type	Opt Data Len	Identification	Option Type	Opt Data Len	L	Identification
Target Address			Address[1]			
Address[1]			Address[2]			
Address[2]					
.....			Address[n]			
Address[n]						

Figure 3. RREQ and RREP Packet Format in DSR

Figures 2 and 3 shows the packet format for both RREQ and RREP in AODV and DSR protocols respectively. The fields of the frame in both RREQ and RREP are very similar in both cases. Three types of TCP variants were analyzed TCP-New Reno, TCP-Sack, and TCP-Vegas, TCP-Vegas keeps track of the time needed for acknowledgment to be received by the sender to calculate an approximated RTT. Whenever a sender receives two duplicates, TCP-Vegas compares the current time of the segment transmission time with approximated RTT. If the current time is higher, the sender starts to retransmit immediately without waiting for a third acknowledgment to be received.

Table1. Environment parameters.

Mobility model Random Way Point	
Speed Minimum 1 mps Maximum 20 mps	
Pause	Time
Pause:5s	
Time Simulation Time 300s	
Terrain Coordination 850 * 850 m	
Connection FTP (File transfer protocol): between 10 different nodes Item size 512(byte)	
Radio/physical layer parameters: Radio type: 802.11b Radio Data rate: 2Mbps	
MAC Protocol: 802.11	
Routing Protocol: AODV & DSR	
Transport Protocol: TCP New Reno, TCP Sack & TCP Vegas	
Node: 20,40,60	
Node Placement: Random	

4. RESULTS AND DISCUSSION

The simulation is executed many times and the number of nodes varies as 20, 40, and 60. AODV and DSR routing protocols are configured for each node with a maximum speed of 20 m/s and with 5s pause time. The mobility of the nodes is very low in all scenarios. For each design parameters, the simulation is executed many times and the average value of each performance parameter is used to evaluate the performance.

4.1 PACKET DROP

Packets drop usually when the sender fails to deliver some or all packets to the receiver or when the receiver buffer is full. In MANET there are more reasons for packets to drop, like fading, interference from other devices and noise, so it's impossible to determine the reason for packets dropped. A number of dropped packets can be measured as:

$$NDP = PS - PD$$

Where NDP is the Number of Dropped Packets, PS is the Packets Sent and PD is the Packets Delivered.

Figure 4 shows the number of dropped packets for all TCP variants used with both AODV and DSR routing protocols. Packet drop is much lower when DSR routing protocol is used compared to AODV. The behavior of the TCP variants was the same with both AODV and DSR, the lowest is TCP-Vegas, and then TCP-SACK, the highest drop is with TCP-New Reno. TCP-New Reno shows the highest drop packet due to its mechanism that the sender node must receive an acknowledgment for every packet sent in the fast recovery phase. TCP-New Reno also needs one TTL in order to detect a dropped packet. TCP-SACK performed better than TCP-New Reno

because of the mechanism that TCP-SACK use depends on receiving acknowledgment only for the non-contiguous dropped packet which avoids the accumulative acknowledgment unlike TCP-New Reno, TCP-Vegas shows the lowest packet drop when DSR is used. This is because TCP-Vegas did not wait for the packets to drop to be able to detect congestion.

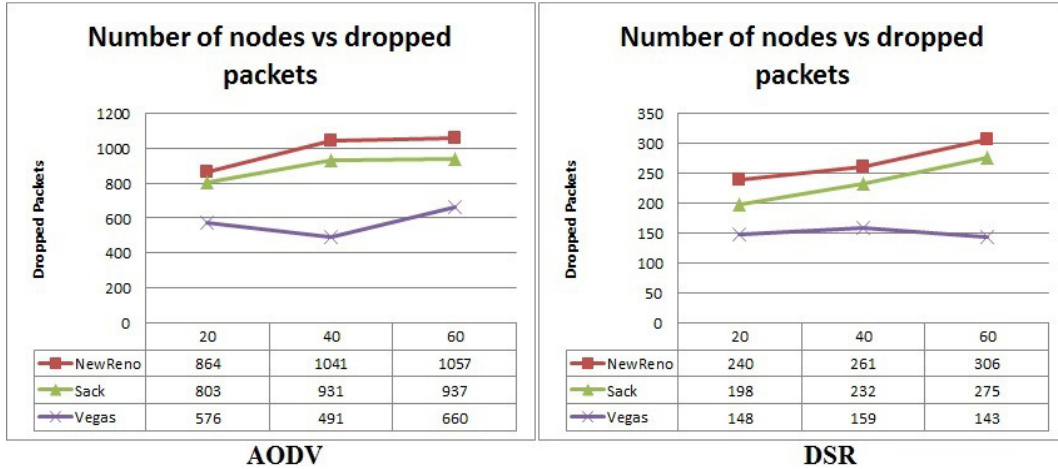


Figure 4. Dropped packets of TCP variants in AODV and DSR

4.2 THROUGHPUT

Throughput is the number of successfully transmitted data from source to destination per second. It depends on the number of packets sent and a number of Dropped packets through the network. If T is the Throughput, NKB is the Number of KiloBytes received, ST is the Start Time and ET is the End Time. It can be measured as:

$$T = \frac{NKB}{(ST - ET)}$$

Figure 5 shows the Throughput of TCP variants over AODV and DSR routing protocols. TCP-Vegas has the highest Throughput because of its mechanism for detecting a congestion. More packets can go through the network which shows better performance of the network. Comparing Throughput of TCP variants in both DSR and AODV, TCP variants have highest Throughput values in DSR than in AODV. This is expected as the DSR outperform AODV at low mobility used in this scenario.

TCP-New Reno Throughput decreases with increasing the number of nodes in DSR routing protocol, unlike TCP-SACK which almost maintained its Throughput while increasing the number of nodes. But TCP-SACK showed a different behavior in AODV than in DSR while increasing the number of nodes the Throughput decrease.

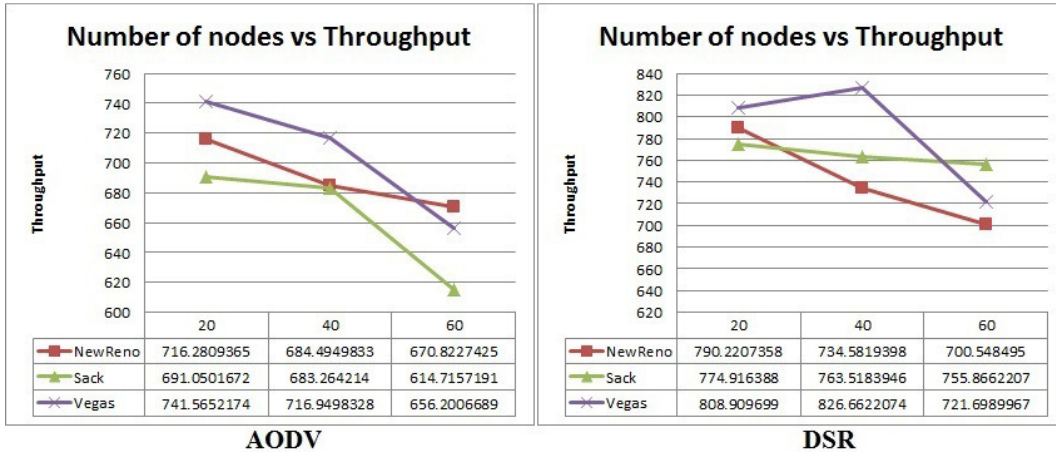


Figure 5. Throughput of TCP variants in AODV and DSR

4.3 JITTER

Jitter is the variation in the delay of received packets. In order to get a high performance network for real-time traffic, the Jitter value should be minimized as possible. Jitter can be measured as:

$$J = D(m + 1) - D(m)$$

Where J is the Jitter and D is the delay and m is a positive integer. Figure 6 shows Jitter for all TCP variants used with both AODV and DSR routing protocols, Comparing Jitter in DSR and AODV, TCP variants have a lower Jitter in AODV than DSR also TCP Vegas proved that it's has a lower jitter than other two variants in both DSR and AODV.

In AODV routing protocol the Jitter performance of both TCP-SACK and TCP-New RENO increases while increasing the number of nodes unlike in TCP Vegas which almost maintained its behavior.

DSR routing protocol TCP-SACK showed a decreasing in Jitter while increasing the number of nodes, unlike TCP-New Reno which showed that Jitter increase while the number of nodes increases. TCP-Vegas which almost maintained its behavior as in AODV routing protocol.

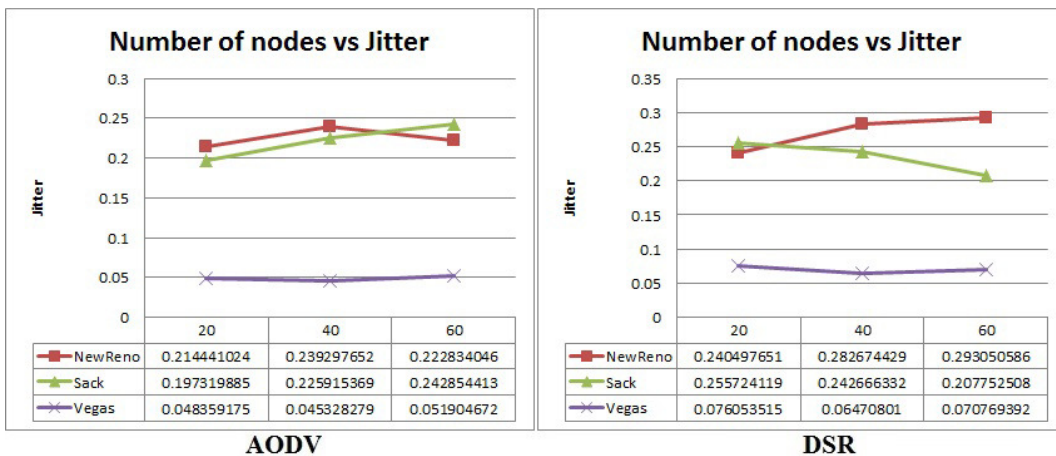


Figure 6. Jitter of TCP variants in AODV and DSR

5. CONCLUSION

This paper presented a performance comparison between TCP variants with a different reactive routing protocol. The comparison is held by running a simulation scenario many times using NS2. Three performance metrics were used to compare the performance of TCP variants Throughput, Jitter, and Packet drop.

TCP variants perform in a similar way in both DSR and AODV routing protocol. There is no much effect on the routing protocol on the TCP variants. TCP-Vegas outperform the other two variants in all parameters, packet drop, throughput, and jitter. The results show that DSR has better performance compared with AODV, because DSR routing protocol mechanism outperforms AODV at low traffic, nodes and mobility. It generates less routing load and depends more on caching.

REFERENCES

- [1] C. Perkins, E. Belding-Royer & S. Das, (2003) "Ad hoc On-Demand Distance Vector (AODV) Routing", RFC 3561.
- [2] D. Johnson & D. Maltz, (2007) "The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4", RFC 4728.
- [3] M. Allman, V. Paxson & W. Stevens, (1999) "TCP Congestion Control", RFC2581.
- [4] Tomar, Poonam, & Prashant Panse, (2011), "A Comprehensive Analysis and Comparison of TCP Tahoe, TCP Reno and TCP Lite." International Journal of Computer Science and Information Technologies (IJCSIT), Vol. 2, No 5, pp2467-2471.
- [5] Yuvaraju B. N & Niranjana N Chiplunkar, (2010) "Scenario Based Performance Analysis of Variants of TCP using NS2-Simulator" International Journal of Advancements in Technology, Vol. 1, No 2, pp223-233.
- [6] Neha Arora, (2013) "Comparative Analysis of Routing Protocols And TCP in MANETS", International Journal of Emerging Trends in Engineering & Technology (IJETET) Vol. 02, No. 1, pp19-28.
- [7] Suneel Kumar Duvvuri & Dr. S. Rama Krishna (2016) "Performance Evaluation of TCP alternatives in MANET using Reactive Routing Protocol", International Journal of Modern Computer Science (IJMCS), Vol. 4, No.4, pp35-39.
- [8] M. Jehan & Dr. G. Radhamani, (2011) "Scalable TCP: Better Throughput in TCP Congestion Control Algorithms on MANETs", International Journal of Advanced Computer Science and Applications (IJACSA), pp14-18.
- [9] Iffat Syad, Sehrish Abrejo & Asma Ansari, (2013) "analysis of proactive and reactive MANET routing protocols under selected TCP variants", International Journal of Ad hoc, Sensor & Ubiquitous Computing (IJASUC) Vol.4, No.4, pp17-26.
- [10] Hrituparna Paul, Anish Kumar Saha, Partha Pratim Deb & Partha Sarathi, (2012) "Comparative Analysis of Different TCP Variants in Mobile Ad-Hoc Network", International Journal of Computer Applications Vol. 52, No.13, pp19-22.
- [11] Md Nazmul Islam Khan, Rashed Ahmed & Md. Tariq Aziz, (2012) "a survey of TCP Reno, New Reno and Sack over mobile ad-hoc network", International Journal of Distributed and Parallel Systems (IJDPS) Vol.3, No.1, pp49-63.
- [12] Andreas Hafslund¹, Lars Landmark², Paal Engelstad³ and Frank Y. Liz (2004), "Testing and Analyzing TCP Performance in a Wireless-Wired Mobile Ad Hoc Test Bed", IEEE, International Workshop on Wireless Ad-Hoc Networks, pp115-119.

- [13] Alaa Seddik-GhalebYacine,Ghamri-Doudane & Sidi-Mohammed Senouci (2006), “Effect of Ad Hoc Routing Protocols on TCP Performance within MANETs”,IEEE,3rd Annual IEEE Communications Society on Sensor and Ad Hoc Communications and Networks, pp866-873.
- [14] Anuj K. Gupta, Harsh Sadawarti, & Anil K. Verma ,(2011)” Review of Various Routing Protocols for MANETs”,International Journal of Information and Electronics Engineering, Vol. 1, No. 3, pp251-259.
- [15] S.Floyd & T. Henderson , (2004)” The NewReno Modification to TCP's Fast Recovery Algorithm”, RFC 3782.
- [16] S Floyd, (1996) “ TCP Selective Acknowledgment Options”, RFC 2018.

AUTHORS

Mahmoud A M Abu-Zant received his BS in computer science from An-Najah National University, Nablus, Palestine, 2015. He is currently pursuing his Master of Computer Science from Arab American University, Jenin, Palestine. His researches interest includes computer networking, information security, wireless sensor network.

Mohammad M N Hamarsheh received his BS in electrical and computer engineering from An-Najah National University, Nablus, Palestine, in 1999, and his MS and Ph.D. degrees in computer and communication engineering and communications and network engineering, in 2002 and 2006, respectively from University Putra Malaysia, Serdang, Malaysia. He worked as the head of the FBG fabrication unit with Photronix, Cyberjaya, Malaysia. He was involved in optical CDMA network and fiber Bragg grating research and development. He worked as a lecturer with the faculty of information science and technology, Multimedia University, Malaysia, from 2005 to 2008. He is currently an associate professor at Arab American University, Jenin, Palestine. His research interests include computer networking and security, MANET, MPLS, optical code division multiple access systems, fiber Bragg gratings, dense wavelength division multiplexing, and polarization effect on optical fiber communication.