An Improved TCP for Reduced Packet Delay in IEEE 802.11s-Based Smart Grid AMI Networks

Nico Saputro(✉) and Kemal Akkaya

Department of Computer Science, Southern Illinois University,
Carbondale, IL 62901, USA
nico@siu.edu, kemal@cs.siu.edu

Abstract. Transmission Control Protocol (TCP) can handle packet losses by retransmitting them when the corresponding acknowledgement (ACK) packets are not received within a certain time interval. This time interval is referred to as *retransmission timeout* (RTO) and setting its value is critical to reduce the packet delay in Smart Grid Advanced Metering Infrastructure (AMI) networks. In this paper, we propose a novel mechanism to set the RTO of each smart meter (SM) to improve the performance of TCP in IEEE 802.11s-based AMI networks in terms of packet delay. The idea is based on using the location of the SMs in the network topology and assign an RTO based on its distance from the gateway. In addition, we propose eliminating the doubling of RTO value when ACK packets are not received. The simulation results under ns-3 simulator indicate that the delay performance can be improved at least 40% with the use of these mechanisms.

Keywords: IEEE 802.11s · Advanced Metering Infrastructure · Transmission Control Protocol · Retransmission Timeout

1 Introduction

Advanced Metering Infrastructure (AMI) is one of Smart Grid (SG) applications which is used specifically for the collection of periodic power consumption readings from the customers. Typically, these fine-grained power readings are sent to the utility companies at some pre-defined intervals. In addition to billing purposes, this huge amount of data can be used for leakage detection, demand response, state estimation [1] and various statistical analysis [2].

Recently, there has been several proposals for forming the communication infrastructure to be used in AMI applications [3,4]. Among these alternatives, wireless mesh networking has been one of the viable options exploited by several utility companies. In this architecture, the smart meters (SMs) form a wireless mesh network (WMN) among them and forward their readings to a single gateway for relaying to the utility company. This WMN is operated in a neighborhood area network (NAN) and can be implemented using proprietary or open standards. Our focus is on WMN that based on standard IEEE 802.11s [5] and we refer this WMN as SG AMI networks hereafter.

N. Mitton et al. (Eds.): ADHOCNETS 2014, LNICST 140, pp. 86–97, 2014.
DOI: 10.1007/978-3-319-13329-4_8

SG AMI networks are expected to carry different types of traffic due to availability of various other SG applications that need to communicate with SMs. For instance, in addition to sending regular power readings, the network will carry traffic for demand response applications where the utility will need to access the SM and the SM needs to collaborate with the neighbors for reducing the power usage in the neighborhood [6]. Similarly, plug-in hybrid electric vehicle (PHEV) power stations may need to communicate with each other through this NAN in order to share the load for PHEV charging [7]. For such cases, the end-to-end delay of the sent packets is crucial for real-time state estimation of the grid to prevent power outages as well as reducing the stress on the grid in a timely fashion. Thus, in any case the end-to-end delay of the packets sent/received on the NANs is a critical issue to be tackled.

To address this issue one of the options is to use user datagram protocol (UDP) so that the overhead of the protocols can be avoided and thus end-to-end delay can be reduced. However, this is not a strong option due to the characteristics of SG applications. Reliable delivery of the packets in SG AMI application is of utmost concern since the data are used for crucial issues such as billing. Therefore, in most cases TCP becomes the only option for guaranteeing reliable delivery of the readings to be received at the utility company. This is done through congestion control and retransmission mechanisms. With these mechanisms, TCP is showing high end-to-end (ETE) data delivery ratio. However, this performance is at the expense of high ETE delay.

We tackle the problem of increased delays because of the use of TCP and propose two mechanisms to reduce the delay so that TCP can be used in SG AMI networks. The first mechanism is based on the idea of proper setting of RTO values for the SMs. Rather than using a single RTO value for all the nodes as done in traditional networks, we propose to set the RTO value for each SM based on its location in the mesh topology given that SG AMI networks are large-scale multi-hop networks. In this way, closer SMs will have smaller RTO values compared to distant SMs. As a result, distant nodes from the gateway will have more time to respond and thus less retransmissions will occur. The assignment is done using a spanning tree of the topology so that the nodes at similar distances from the gateway will have same RTOs. The second mechanism we propose is based on the idea of limiting the number of doubling when the RTO timer expires. This is motivated from the fact that doubling the RTO increases the RTO value too much that negatively affects the packet delay. The simulation results indicate that both approaches contribute to reduction of ETE delay under a variety of network conditions without any impact on other metrics such as packet delivery ratio and throughput.

This paper is organized as follows. The next section summarizes the related work. Section 3 provides preliminaries on SG AMI, TCP retransmission timeout and description of the problem. Section 4 includes the description of the proposed approaches. In Section 5, we assess the performance of the proposed approaches. Finally, the paper is concluded in Section 6.

2 Related Work

The use of TCP in wireless networks has been a major area of research in the past. The main issue was the triggering of slow start algorithm employed by TCP when congestion occurs in wired networks. However, wireless networks have four major characteristics that distinguish them from wired networks [8]: (1) channel contention; (2) signal fading; (3) mobility; and (4) limited power and energy. These characteristics induce non-congestion events which cause packet loss. Hence, the traditional TCP congestion control mechanisms may react inappropriately due to the misinterpretation of the caused of the packet loss in wireless networks. Therefore, a lot of approaches have been proposed to improve the performance of TCP in terms of various metrics. A good summary of these efforts can be found in [8].

Among these metrics, throughput has been widely studied. A number of approaches have been proposed for the reduction of channel contention to improve the TCP throughput such as multichannel assignment for forming WMNs without hidden node problem [9] and decreasing the number of transmitted ACKs. To this end, instead of sending an ACK for every segment (e.g., TCP layer packet), an ACK is delayed until after two segments are received or after a specific time limit (typically 0.2s) is reached (i.e., if two segments are not received within this limit) [10]. The delayed ACK can also be based on certain criteria, such as the number of segment [11], channel condition [12], and the path length [13]. Note that all of these approaches focus on TCP throughput improvement and thus may bring additional delay which is contradicting with our goal of improving the ETE delay.

The works which dealt with RTOs were proposed in [14,15]. In [14], the main motivation was to handle the spurious RTO timeouts due to the delayed ACKs used in various previous works. The authors proposed an adaptive minimum RTO to identify the data segments whose ACKs will possibly be delayed. A fixed extended Minimum RTO is assigned for those identified segments. In [15], the authors presented a heuristic mechanism for mobile ad hoc network, called Fixed RTO, that assumes a route failure occurs when there are two consecutive RTOs. The un-ACKed segment is retransmitted without doubling the RTO timer anymore (i.e. the RTO timer is kept fixed) until an ACK has been received for that segment. These approaches also adjust RTOs as ours but the goals are very different and the process is done on a need basis. In our case, we do it for every node in advance based on their location in the network topology.

With the deployment of SMs and several other SG devices, TCP is started to be revisited for tuning its performance for SG domain. The main focus of these efforts is on reliability since this is one of the most important metrics for SG applications. Very recently, two revisions had been proposed for TCP [16,17]. The work in [16] focuses on the reliability and throughput performance of TCP in a large-scale setting. The main goal is to aggregate the TCP traffic from SMs at certain regional aggregators. The domain of the work in [17] is not AMI. It focuses on the monitoring aspect of the SG. Phasor Measurement Units (PMUs) which are sent real-time are considered as the data traffic sources. The work addresses issues regarding reliability along with security.

3 Preliminaries

3.1 SG AMI Networks

In a SG AMI network, SMs will send their readings to the data collector using mesh path selection and forwarding mechanisms of IEEE 802.11s called Hybrid Wireless Routing Protocol (HWMP) [5]. Mesh path selection enables path discovery within the mesh network using layer-2 (MAC) addresses rather than IP addresses. The frequency of readings (or reports) from SMs may change from one utility to another and type of the consumers. For instance, residential homes' data can be collected in minutes while industrial building's data can be collected in seconds. However, it is not uncommon to collect meter data from 5 to 30 seconds [18,19].

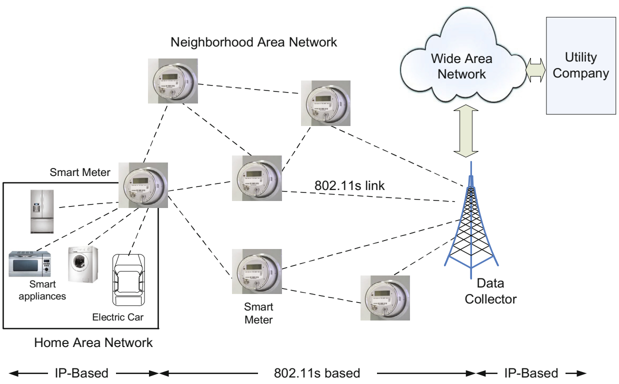

Fig. 1. A sample of SG AMI Network

While SG AMI network uses IEEE 802.11s, it still communicates with other networks that are part of the SG network. For instance, as depicted in Fig. 1, home appliances may communicate with the SMs, or electric stations for PHEVs can talk to each other via this SG AMI network. Similarly, sensors and utility company will be communicating with this SG AMI network. Therefore, IP based protocols are still used. At the transport layer, TCP or UDP can be used. However since reliability is an important metric in collecting power readings, TCP is the most widely used option at the transport layer for the utilities.

3.2 Overview TCP Retransmission Timeout

TCP uses retransmission timeout (RTO) timer to identify segment loss. Whenever a sender sends a data segment to a destination, an RTO timer is activated. TCP assumes segment loss when RTO timer expires before receiving the ACK. For this case, TCP retransmits the segment and a new RTO timer is assigned for it. The value of an RTO timer is defined in RFC-6298 [20]. At the beginning when there is no measured samples, 1sec is recommended as the initial RTO timer. The RTO timer for the next transmitted segment is the maximum between the minimum RTO and the value computed in Eq. 1. The recommended minimum RTO is 1sec. In case of retransmission, the new RTO timer is doubled from the previous RTO timer.

$$RTO = SRTT + 4 \times RTTVAR \tag{1}$$

where, SRTT represents smoothed Round Trip Transmission (RTT) delay of the measured samples, and RTTVAR represents the RTT variation. Note that the RTT samples must not be taken from packets that were retransmitted.

3.3 Problem Motivation

SG AMI networks introduce a lot of contention when used with TCP. This is because the contention is not only between the transmission of upstream data packets with their downstream acknowledgements (ACKs), but also from the management and control frames of IEEE 802.11s. These contentions eventually cause collision and packet loss which increase the delay. Another major issue is the retransmissions of the packets. The TCP data streams in SG AMI networks are performed in a multi-hop manner towards the gateway. This may create several issues. First, there will be traffic bottlenecks and this will increase the chance of packet collisions. Second, the probability of packet drops due to wireless environment characteristics will increase due to increased hop counts. Finally, the TCP streams may be received out-of order at the destinations. All these problems contribute to increasing number of retransmissions and thus ETE delay.

The problem of retransmissions can introduce additional delays in SG AMI Networks because of the increased frequency of data collections. Recall that the data collection frequency in some AMI applications can vary from 5secs to 30 secs. In such a case, it is important to collect all the required SM readings for each round so that this data can be used for real-time objectives such as state estimation. However, there is a risk for some of the segments not to be received by the destinations before the next round of data collection begins. This is mainly due to retransmissions as detailed below.

The recommended values in RFC 6298 may not be suitable for periodic data reporting in SG AMI networks. Typically, all SMs will send their reading at the same predefined time schedule. Initially, when there is no measured RTT samples, such as after the network restart, all SMs will have the same initial RTO timer (e.g 1sec). When all SMs send their readings, the network contention for media access are high and heavy congestion may occur. Some SMs may not receive ACKs for their readings either due to packet loss or the RTO timer expires. These SMs will retransmit their data at the same time again and will have the same new RTO timer (i.e. doubled from the previous one). As a consequence, when some of the SMs still do not receive any ACKs, the similar problem persists: these SMs will retransmit again at the same time and have the same new RTO timer and so on. Doubling the RTO timer may also pose additional delay, especially when its value exceeds the next power reading reporting time. To assess the seriousness of the problem, we conducted some experiments to measure the number of segments which cannot be transmitted until the next round. The results depicted in Fig. 2 indicate that there are indeed a good number of segments which could not be transmitted.

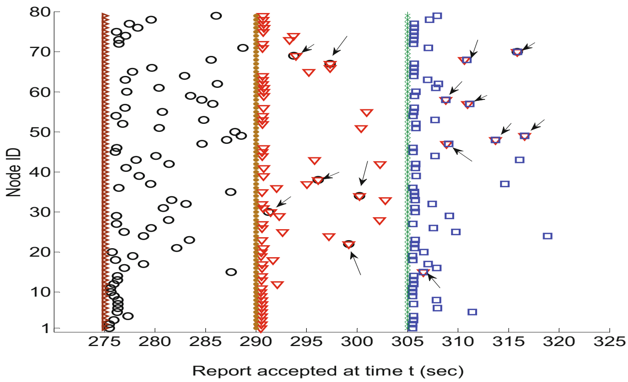

Fig. 2. Arrival time for some of the readings exceed the next reading schedule. These are shown by arrows in the figure. The reporting is done every 15 secs (i.e., 275, 290 and 305s).

In such a case, the next data reporting action at the SMs must wait for the previous reading to be acknowledged first. In case of waiting too much, the RTO timer may expire before the SM can send its reading to the gateway. In case the timer expires, this next reading can be sent together with the previous reading, either partially or in a whole, depending on the window size. However, since TCP sends data in byte streams, there is no definite boundary between the previous and the next data readings. Hence, the receiver must know exactly the data size in order to recover both readings individually. The receiver also needs a temporary storage when the next reading is partially sent due to the window size limitation. This partial reading must be stored until the next segment arrives and the whole reading can be recovered. These results suggest that retransmissions not only will increase the delay but also complicate things in order to handle failed readings. Therefore, it is important to address this problem for SG AMI networks.

3.4 Problem Definition

Given the issues regarding the retransmissions in SG AMI networks, our problem focuses on reduction of ETE packet delay. The problem can be defined as follows: "Given a SG AMI network with a certain number of SMs, their locations, topology and data collection frequency, our goal is to propose a mechanism to reduce the number of retransmissions and thus the ETE delay when TCP is employed."

4 Improved TCP for SG AMI Networks

We propose two mechanisms for SG AMI networks in order to reduce the delay when TCP is used: (1) spanning tree-based minimum RTO setting, and (2) freezing RTO mechanism. The first mechanism takes into account the location of SM and the IEEE 802.11s proactive tree-based path selection. Typically, the location of a SM is attached to a certain household location. Hence, its location is known a priori. IEEE 802.11s proactive tree-based path selection on the other hand, builds a tree topology rooted at the gateway. Hence, each SM has an associate node's position in the network tree topology and different hop count to the gateway. It takes longer time to send to and receive data from the gateway as the SM's hop count increases.

To identify the unique location of each SM in the network, we propose building a spanning tree (ST) for the SG AMI network. An ST of a connected graph G can be defined as a maximal set of edges of G that contains no cycle, or as a minimal set of edges that connect all vertices. Determining the minimum ST (MST) of a network has been widely studied in the literature (e.g., Prim's MST Algorithm [21]) and this can be done by the gateway node after collecting the MAC addresses of the SMs in the SG AMI network.

Our goal is to assign different minimum RTO for each SM based on its position in the ST. Specifically, the SMs at the same ST depth level will have a similar minimum RTO while the nodes that are further away from the gateway (i.e., has a higher network tree depth level) will be assigned a higher minimum RTO. We use the following formula for setting the RTO for a particular node i at level d_i:

$$minRTO_d^i = 0.1 \times d_i + r_i \tag{2}$$

where r_i represents a random value introduced for each dept level so that the nodes at the same depth level will have different RTO values.

The second mechanism is proposed due to the fact that RFC 6298 does not specifically define the upper bound of the doubling mechanism when the RTO timer expires. We also follow an idea for stopping the doubling of RTO timer when the ACK is not received. Basically, if an ACK is not received after first doubling of RTO timer, then the timer is not re-doubled but frozen until the ACK is arrived. In this way, whenever RTO timer expires again after that, the retransmission intervals for the corresponding packet is constant.

5 Performance Evaluation

5.1 Experiment Setup

We evaluated the performance of our proposed approaches in ns-3 [22]. We created SG AMI networks that consist of **N** by **N** mesh nodes. One mesh node acts as a data collector while the rest act as SMs. The transmission range of each node is assumed to be 120m. The underlying MAC is assume to be 802.11g. HWMP proactive modes is used to determine the paths among the SMs and gateway. The power readings are put in a packet size of 512 bytes and these packets are transmitted every 15secs which is consistent with some of the real SMs on the market [19]. The packets are assumed to be sent at the same times by all the SMs since this data will be used by the utility to do real-time state estimation. We measure at the application layer, the average End-to-end (ETE) delay of all the packets sent from SMs to the gateway. We also looked at the packet delivery ratio (PDR) which indicates the ratio of the total number of packets sent by SMs and successfully received by the gateway. A final metric is throughput which is the total number of bits received at the gateway in a second. We conducted experiments to assess these three metrics and compared them to the baseline. The baseline is the basic operations of HWMP and TCP/IP protocol stacks in ns3.18, and uses the recommended initial RTO and minimum RTO values in RFC 6298 (i.e 1sec).

5.2 ETE Delay Performance

We first conducted experiments to assess the ETE delay of the tree-based minimum RTO allocation. The results are depicted in Fig. 3. On average, tree-based approach improves the ETE delay around 18% compared to the baseline. However, there are fluctuations in the results and the improvement is not very significant. Therefore, we decided to do more investigation through additional experiments.

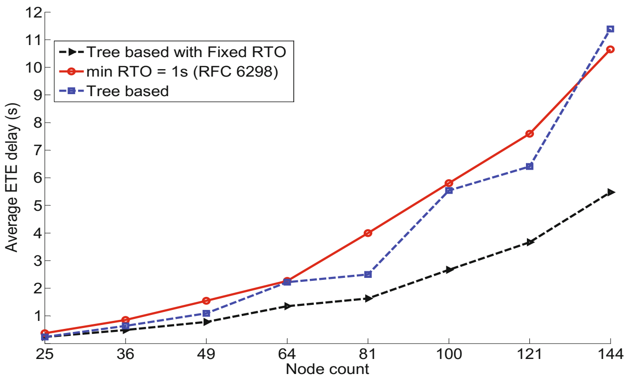

Fig. 3. The ETE delay performance results

Firstly, the ETE delay is less than that of the baseline approach except for 144 nodes. When we looked at the situation in more details, we figured out that there is one node that is not able to establish a TCP connection with the gateway due to Address Resolution Protocol (ARP) issues. Basically, ARP requests could be lost and some of the nodes may not establish a TCP connection as shown in previous study [23]. This may result in less traffic and thus less contention in the network, reducing the ETE delay for the baseline. In the tree-based approach however, all the nodes were able to send their readings.

Secondly, we picked the topology with 81 nodes and investigated the behavior of RTO setting and timeouts. The maximum depth level for this topology is 16 and hence the highest minimum RTO based on Eq. 2 would be around 2s. Fig. 4 shows a snapshot of arriving packets and RTO expiration times. The results indicate that the variation of RTO timer is very high. In fact, some of the RTO values are higher than 2s. This indicates that the network was previously having high contention and collision that make the RTTs of previous sending reports are high. It is also estimated by the RTT estimator that the current report will having the same issue.

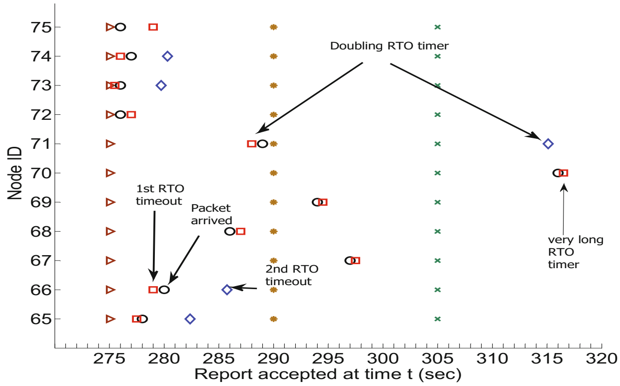

Fig. 4. Doubling the RTO timer exceeds the next reading schedule, we only show for packets sent at 275s

We argue that estimating the RTO value is critically important since it is expected to be the reflection of the network conditions. For instance, when the actual packet is not lost, a small RTO value may trigger a packet retransmission due to network congestion which eventually increase the ETE delay. However, when the packet is actually lost, a small RTO value gives the benefit of a fast retransmission and hence reduces the ETE delay. On the other hand, a longer RTO may prevent a packet retransmission since it will be sufficient to receive the ACK. The negative side of a longer RTO timer is when the packet is actually lost and the sender need to wait long enough before it can retransmit the packet. This situation increases the ETE delay. These situations can be observed in Fig. 4. Five packet retransmissions have occurred (e.g., node ID 65, 66, 71, 73, and 74) and the gateway receives those packets not long after the RTO expires. The node ID 73 and 74 have a smaller RTO timer compared to the others and thus they have lower ETE delay. Recall that doubling the RTO value when the RTO expires poses a problem when the value encompasses the next data reading scheduled. For instance, the node with the ID 71 has the second highest RTO value that exceeds the next two data reading scheduled. Fortunately, the data reading has been received by the gateway and its acknowledgement also has been received by the sender before the next scheduled so that the node can send its next scheduled report as planned. On the other hand, node with ID 70 has a very long RTO value that exceeds the next two scheduled. Hence, the next two packets are held until the previous packet has been received and acknowledged. These increase the ETE delay significantly. As a result, we decided that freezing RTO value at the SMs may reduce ETE delay as will be discussed next.

We repeated the same experiments by applying the idea of RTO freezing. Basically, for each SM, after the RTO is doubled, it stops there and no more doubling is performed. The results which are shown in Fig. 3 indicate that freezing the RTO significantly help reducing the ETE delay. Overall, this approach reduces the ETE delay around 48% (on average of all node count) and 41% compared to baseline and tree-based approach respectively. This is attributed to the fact that retransmissions are forced and thus a sender does not have to wait too long for the congested packets which really need for retransmission. Eventually, the ETE delay is reduced. We confirmed the effectiveness of this improvement by checking the arrival times of readings compared to the snaphot in Fig. 2. In our proposed approaches, the gateway was able to receive all the packets before the next reading scheduled starts.

5.3 PDR and Throughput Performance

We also looked at the two other performance metrics to verify that the proposed approaches do not impose any adverse effect on them. However, due to limited space, we can not show the results in a table or graphs. The results indicate that the PDRs for the proposed approaches are as better as the baseline. The PDR stays 100% for smaller network sizes. For larger network sizes, the PDR is still close to 100% and our tree-based approach along with freezing performs even slightly better which is promising. For the throughput, we observe a similar situation. The proposed approaches do not impact the throughput. On the contrary, in large-scales the throughput is even slightly higher.

6 Conclusion

In this paper, we proposed an improved TCP that can be used for SG AMI networks that can support a varity of applications. The idea of the approach was to set the RTOs of each SM in order to reduce the end-to-end packet delay. The setting was done by considering the distance of each SM from the gateway in the network. Specifically, distant SMs are provided with longer RTOs so that their RTO will not expire quickly and thus retranmission of data segments will be prevented. We also proposed a mechanism for freezing the RTO in order to further reduce the ETE delay. The proposed approaches are implemented and compared with the existing TCP in terms of packet delay, throughput and PDR.

The results indicate that significants reductions in the delay can be achieved regardless of the network size. In addition, the proposed approaches do not negtively impact the PDR and throughput performances. In the future, we plan to investigate whether the QoS-based EDCA (i.e., MAC protocol for QoS support in IEEE 802.11s) can be utilized to further accelerate the transmission of ACK packets and thus reduce the delay further.

Acknowledgments. This work is supported by US National Science Foundation under the grant number 1318872.

References

1. Baran, M., McDermott, T.: Distribution system state estimation using ami data. In: Power Systems Conference and Exposition PSCE 2009, pp. 1–3. IEEE/PES (March 2009)
2. Fang, X., Misra, S., Xue, G., Yang, D.: Smart grid - the new and improved power grid: A survey. IEEE Communications Surveys Tutorials (2011)
3. Saputro, N., Akkaya, K., Uludag, S.: A survey of routing protocols for smart grid communications. Comput. Netw. **56**(11), 2742–2771 (2012)
4. Gao, J., Xiao, Y., Liu, J., Liang, W., Chen, C.P.: A survey of communication/networking in smart grids. Future Gener. Comput. Syst. **28**, 391–404 (2012). http://dx.doi.org/10.1016/j.future.2011.04.014
5. Ieee standard for information technology-telecommunications and information exchange between systems-local and metropolitan area networks-specific requirements part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications amendment 10: Mesh networking. IEEE Std 802.11s-2011, 1–372 (October 2011)
6. Maharjan, S., Zhu, Q., Zhang, Y., Gjessing, S., Basar, T.: Dependable demand response management in the smart grid: A stackelberg game approach. IEEE Transactions on Smart Grid **4**(1), 120–132 (2013)
7. Erol-Kantarci, M., Sarker, J., Mouftah, H.: Communication-based plug-in hybrid electrical vehicle load management in the smart grid. In: 2011 IEEE Symposium on Computers and Communications (ISCC), pp. 404–409 (June 2011)
8. Leung, K.-C., Li, V.O.: Transmission control protocol (tcp) in wireless networks: issues, approaches, and challenges. IEEE Communications Surveys & Tutorials **8**(4), 64–79 (2006)
9. Tung, L.-P., Shih, W.-K., Cho, T.-C., Sun, Y., Chen, M.C.: Tcp throughput enhancement over wireless mesh networks. IEEE Communications Magazine **45**(11), 64–70 (2007)
10. Braden, R.: Rfc 1122 - requirements for internet hosts - communication layers (October 1989). http://tools.ietf.org/rfc/rfc1122.txt
11. Altman, E., Jiménez, T.: Novel Delayed ACK Techniques for Improving TCP Performance in Multihop Wireless Networks. In: Conti, M., Giordano, S., Gregori, E., Olariu, S. (eds.) PWC 2003. LNCS, vol. 2775, pp. 237–250. Springer, Heidelberg (2003)
12. de Oliveira, R., Braun, T.: A dynamic adaptive acknowledgment strategy for tcp over multihop wireless networks. In: Proceedings of the IEEE INFOCOM 2005 24th Annual Joint Conference of the IEEE Computer and Communications Societies, vol. 3, pp. 1863–1874 (March 2005)

13. Chen, J., Gerla, M., Lee, Y.Z., Sanadidi, M.Y.: Tcp with delayed ack for wireless networks. Ad Hoc Netw. **6**(7), 1098–1116 (2008). http://dx.doi.org/10.1016/j.adhoc.2007.10.004
14. Psaras, I., Tsaoussidis, V.: On the properties of an adaptive TCP minimum RTO. Computer Communications **32**(5), 888–895 (2009). http://www.sciencedirect.com/science/article/pii/S0140366408006610
15. Dyer T.D., Boppana, R.V.: A comparison of tcp performance over three routing protocols for mobile ad hoc networks. In: Proceedings of the 2nd ACM International Symposium on Mobile Ad Hoc Networking & Computing. ACM, pp. 56–66 (2001)
16. Khalifa, T., Abdrabou, A., Naik, K., Alsabaan, M., Nayak, A., Goel, N.: Split- and aggregated-transmission control protocol (sa-tcp) for smart power grid. IEEE Transactions on Smart Grid **5**(1), 381–391 (2014)
17. Kim, Y.-J., Thottan, M.: Sgtp: Smart grid transport protocol for secure reliable delivery of periodic real time data. Bell Labs Technical Journal **16**(3), 83–99 (2011). http://dx.doi.org/10.1002/bltj.20523
18. Pressley, L.: Austin smart meters (July 2013), http://www.austinsmartmeters.com
19. Korean electric power research institute (2013), http://www.kepri.re.kr/
20. Paxson, V., Allman, M., Chu, J., Sargent, M.: Rfc 2988 - computing tcp's retransmission timer (June 2011), http://www.rfc-editor.org/rfc/rfc6298.txt
21. Cormen, T.H., Stein, C., Rivest, R.L., Leiserson, C.E.: Introduction to Algorithms, 2nd edn. McGraw-Hill Higher Education (2001)
22. Andreev, K., Boyko, P.: Ieee 802.11 s mesh networking ns-3 model, IITP, WNS3 (March 2011)
23. Saputro, N., Akkaya, K.: An Efficient and Secure ARP for Large-scale IEEE 802.11s-based Smart Grid Networks. In: Mellouk, A., Sherif, M.H., Bellavista, P., Li, J. (eds.) ADHOCNETS 2013. LNICST, vol. 129, pp. 203–217. Springer, Heidelberg (2013)