One-Way Delay Measurement: State of the Art

Luca De Vito, Sergio Rapuano, *Member, IEEE*, and Laura Tomaciello

***Abstract*—Nowadays, the evaluation of performance measurement in computer networks is an important issue. To ensure the quality of service of the network communication, one of the most important network performance parameters is the one-way delay (OWD). For accurate OWD estimation, it is essential to consider some parameters that can influence the measure, such as the operating system and, in particular, the threads, which are concurrent with the measurement application. Moreover, OWD estimation is not an easy task, because it can be affected by synchronization uncertainties. This paper aims to review the different solutions proposed in the scientific literature for OWD measurement. These solutions adopt different methods to guarantee a reasonable clock synchronization based on the Network Time Protocol, the Global Positioning System, and the IEEE 1588 Standard. These different approaches are critically reviewed, showing their advantages and disadvantages.**

***Index Terms*—Global Positioning System (GPS), IEEE 1588, network, Network Time Protocol (NTP), one-way delay (OWD), synchronization.**

I. Introduction

As COMPUTER networks become more complex and larger, measurement infrastructures and methodologies become essential in characterizing network performances [1]. The metrics of the greatest relevance for network performance can be divided into four main groups: 1) availability; 2) loss and error; 3) delay; and 4) bandwidth.

Availability metrics assess how robust the network is, i.e., the percentage of time the network is running without any problems that impact the availability of services. Loss and error metrics indicate network congestion conditions, transmission errors, and/or equipment malfunctioning. They usually measure the fraction of packets lost in a network due to buffer overflows or the fraction of errored bits or packets. Bandwidth metrics assess the amount of data that a user can transfer through the network in a time unit that is both dependent and independent of the existing network traffic. Finally, delay metrics also assess network congestion conditions or the effect of routing changes. They measure the delay [one-way delay (OWD) and round-trip delay (RTD)] and Internet Protocol delay variation (IPDV, or "jitter") of the packets transferred by a network [2]. In particular, the OWD is the time between the occurrence of the first bit of a packet on the first observation point, e.g., the transmitting monitor interface, and the occurrence of the last bit of a packet on the second observation point (RFC 2679 [3]). The RTD is considered to be the time interval between the time instant a request packet is sent by a source node and the time instant a response packet is received from the destination node (see RFC 2681 [4]). Finally, the IPDV is the difference in the OWD of a selected pair of packets in a test stream (see RFC 3393 [5]).

It is worth noting that each metric deals with time: time percentage, time delay, and time unit; for this reason, the main indicators in evaluating the network performance are the network delays. Network delays are composed of three components: 1) equipment delay; 2) transmission delay; and 3) propagation delay [6].

The first is the delay introduced by the equipment before it becomes emitting equipment. This delay consists of the processing time, packet switching, and queueing delays and depends on the network load and congestion.

The second is the time taken to transmit all the bits of the frame containing a packet. It depends on the data rate, media, and distance and can only be controlled in a limited way by the network planners [7].

The third is the time between the emission of the first bit (or the last bit) of a packet by the transmitting equipment and the reception of this bit by the receiving equipment [8].

Therefore, the equipment, transmission, and propagation delay indicators allow the determination of the time that the packet spends to travel from source to destination. This time is called OWD.

The OWD, as shown in Fig. 1, is constituted by three time contributions. Fig. 1(a) represents the equipment delay, which is the time interval between instant t_0, when the packet is scheduled for sending, and instant t_1, when the packet reaches the interface. Fig. 1(b), which represents the transmission delay, is the time interval between instants t_1 and t_2, when the packet is completely transmitted onto the medium. Finally, Fig. 1(c) indicates the propagation delay, which is the time interval from t_2 up to t_3, when the packet reaches the destination interface [9].

The OWD can be obtained by the sum of the transmission delay and the propagation delay, which could be defined as the time between the emission of the first bit of a packet by the source and the reception of the last bit of this packet by the receiver [3].

Some papers [10]–[24] deal with a reliable and accurate way to measure the OWD. To measure the OWD, a sequence of probe packets is to be sent from one end of the monitored network to the other end. Each probe packet is marked with

Manuscript received October 12, 2007; revised April 23, 2008. First published June 13, 2008; current version published November 12, 2008.

L. De Vito and L. Tomaciello are with the Laboratory of Signal Processing and Measurement Information, Department of Engineering, University of Sannio, 82100 Benevento, Italy, and also with the Benevento Research Laboratory, Telsey Telecommunications S.p.A, 82018 San Giorgio del Sannio, Italy (e-mail: devito@unisannio.it; laura.tomaciello@unisannio.it).

S. Rapuano is with the Department of Engineering, University of Sannio, 82100 Benevento, Italy (e-mail: rapuano@unisannio.it).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIM.2008.926052

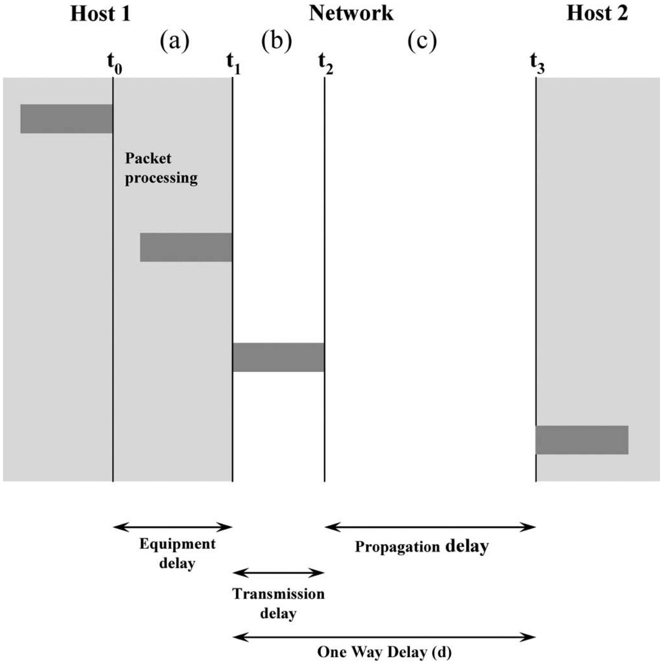


Fig. 1. (a) Equipment delay. (b) Transmission delay. (c) Propagation delay. (d) OWD.

a timestamp immediately before its departure from the source (instant t_1 in Fig. 1). After its arrival to the destination, the difference between the sender timestamp and the time measured by the receiver corresponds to the OWD if end-to-end hosts are perfectly synchronized [25]. The perfect synchronization between the sender and receiver clocks cannot be guaranteed. Different solutions have been proposed to achieve such a target; however, research in this field is still in progress.

Even if the clocks are perfectly synchronized, the sender timestamp in the packet and the receiver measurement instant depend on the operating systems (OSs, interruptions), packetization, and packet compression [7]. Finally, in case of an overloaded link, there will be an increasing OWD trend (as the queue builds up) due to network congestion. During a period of congestion, the packets suffer extra delay due to the time needed to transmit the packets that were previously queued to that interface.

Relevant literature often reports different OWD definitions and different measurement methods. Thus, it is difficult for a researcher to approach this field with a clear view of the problems related to this type of measurement. In this paper, a review of the state of the art of the research dealing with the OWD measurement is presented. Several papers have been reviewed and classified based on the way they face the problems related to the first uncertainty source in OWD measurement, which is the time synchronization between the two hosts doing the measurement. The following three main methods of synchronization have been taken into consideration:

1) synchronization by means of the Network Time Protocol (NTP);
2) synchronization by means of the Global Positioning System (GPS);
3) synchronization by means of the IEEE 1588 Standard.

This paper is organized as follows: In Section II, the importance of synchronization in OWD measurements is shown. Sections III–V describe some methods for the OWD accuracy estimation, depending on the adopted synchronization techniques. In Section VI, a comparison of some characteristics of the solutions proposed in the literature for measuring the OWD is reported. The accuracy of OWD measurements obtained with different synchronization methods is also discussed.

II. OWD Measurement

Estimating OWD between the sender and the receiver is an easy job if it is guaranteed that there is a global time synchronization between them. In that case, the forward delay can simply be calculated by taking the difference of the receiver's clock and the sender's timestamp. The receiver can write this value into the acknowledgement packet header, similarly obtaining the reverse delay [26].

Unfortunately, as clearly stated in [27], OWD measurements are not possible without synchronization, i.e., OWD cannot accurately be obtained because the end system clocks are not synchronized with each other. Any error in the time synchronization heavily affects the accuracy of the OWD measurement. The goal of clock synchronization is to match the values of all clocks in the system and to establish a network-wide time base [28]. Then, it is possible to estimate the OWD ΔT_{ij}^k using the following expression [29]:

$$\Delta T_{ij}^k = x_{ij}^k - \tau_i + \tau_j \tag{1}$$

i.e., the OWD of probe packet k while traveling from the ith sender to the jth receiver x_{ij}^k plus the difference between the two clock offsets τ from the Universal Time Coordinate (UTC). The two offsets can be estimated as the difference of the time at instant 0 [a reference time $Time_0(t_0)$] and the time actually sent in packet i $(Time_i(t_0))$, i.e.,

$$\tau_i = Time_0(t_0) - Time_i(t_0). \tag{2}$$

To evaluate the accuracy of the OWD estimation, all uncertainty sources should be taken into account. They can be grouped into two classes [3].

1) Uncertainties in the clocks of the source and destination hosts: This class contains all uncertainty sources due to the time measurement in both the source and destination hosts. It comprises the uncertainty in the synchronization between the hosts, which can be reduced but not eliminated by choosing an appropriate synchronization method. Further sources in this category are the resolutions of the OS clocks in both the source and destination hosts. If the OS of the source host has, for example, a tick period of 10 ms, then this adds 10 ms of uncertainty to any time value measured with it.
2) Uncertainties due to the difference between the "wire time" and the "host time:" This class contains the uncertainties due to the fact that the time measurement for OWD estimation is made in software. Thus, it is possible

TABLE I
ACCURACY OF THE TIME-SYNCHRONIZATION METHODS

Protocol	Characteristics	Interface	Expected synchronization accuracy
GPS	Standard output from GPS receivers	Directly wired	1 μs
NTP	IPS (Internal Protocol Suite) standard for time sync.	Ethernet	WAN:10-20ms LAN:<1ms
IEEE1588	Standard for instrumentation devices	Ethernet	<1μs

to measure the time at the source just prior to sending the test packet and that at the destination just after having received the test packet. The times collected this way are called "host times." As OWD is defined, however, the time should be measured when the test packet leaves the network interface of the source host and when it (completely) arrives at the network interface of the destination host. These times are called "wire times." The differences between "wire times" and "host times" have an extent that can accurately be estimated and compensated and an extent that is uncertain and has to be counted as an uncertainty source. Then, the total uncertainty about the OWD measurement U can be described by the following expression:

$$U_{ij} = U_{\text{synch},ij} + R_{\text{source}} + R_{\text{dest}} + H_{\text{source}} + H_{\text{dest}} \quad (3)$$

where $U_{\text{synch},ij}$ denotes an upper bound on the uncertainty in synchronization; R_{source} and R_{dest} denote the resolution of the source and destination clocks, respectively; H_{source} and H_{dest} are the contributions due to the difference between the host time and wire time. This estimate of the total uncertainty should be included in the uncertainty analysis of any measurement implementation.

Different synchronization methods can be used [9] to reduce the uncertainty contribution due to the synchronization: NTP, GPS, and IEEE Standard 1588 [30]. Table I reports a comparison among the methods available and how they perform the comparison with the expected accuracy [31].

Most applications working on wide area networks (WANs) or local area networks (LANs) are sufficiently well synchronized by using means such as the NTP, ensuring a synchronization accuracy on the order of milliseconds, or the GPS, working on the order of microseconds. The recent interest in using Ethernet for industrial automation or even in sensor networks led to the adoption of the IEEE 1588 Standard in industry LANs, ensuring an expected synchronization on the order of microseconds. The different synchronization methods will be better described in the following.

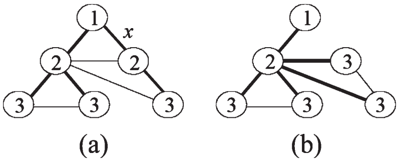


Fig. 2. NTP subnet synchronization.

III. NTP-BASED SYNCHRONIZATION

NTP [32] is a protocol used to synchronize the clock of a client to a reference time source, such as a radio or satellite receiver. It can provide accuracy typically within a millisecond on LANs and up to a few tens of milliseconds on WANs.

The NTP system consists of a hierarchy of primary and secondary time servers, clients, and interconnecting transmission paths [33]. Under normal circumstances, clock synchronization is determined by using only the most accurate and reliable servers and transmission paths so that the actual synchronization paths usually assume a hierarchical configuration with the primary reference source at the root and servers of decreasing accuracy at increasing layers toward the leaves. A primary time server is directly synchronized to a primary reference source, which is usually a timecode receiver or a calibrated atomic clock. Secondary time servers are sychronized with the primary servers. The difference between the primary and secondary time servers is the measurement accuracy. Obviously, the primary server has better accuracy than the other servers. Finally, clients synchronize to the secondary servers.

A typical hierarchy is shown in Fig. 2(a), in which the nodes represent subnet servers, with the layer numbers determined by the hop count to the root. The thick lines represent the active synchronization paths and the directions of the timing information flow. The thin lines represent backup synchronization paths where timing information is exchanged; however, these paths are not necessarily used to synchronize the local clocks. Fig. 2(b) shows the same subnet as in Fig. 2(a) but with the line marked x out of service. In these cases, the subnet automatically reconfigures itself to use backup paths, resulting in one of the servers dropping from layer 2 to layer 3. NTP may take several minutes or even hours to adjust a system's time to the ultimate degree of accuracy. There are several reasons for this. To reduce the effects of variable latency, NTP averages the results of several time exchanges. In addition, it often takes several adjustments for NTP to reach synchronization [34]. In cases requiring a higher accuracy for the synchronization than the estimates in Table I, there are several options, as follows [35]:

1) *Connecting directly to a reference clock:* If a server is directly attached to a reference clock, the accuracy is limited only by the reference clock accuracy and the hardware and software latencies involved in the connections to it.
2) *Pulse per second (PPS):* Clocks can use PPS radio receivers, which receive on-the-second radio pulses from a national standards organization. If the time is within a fraction of a second, the PPS pulses can be used to be

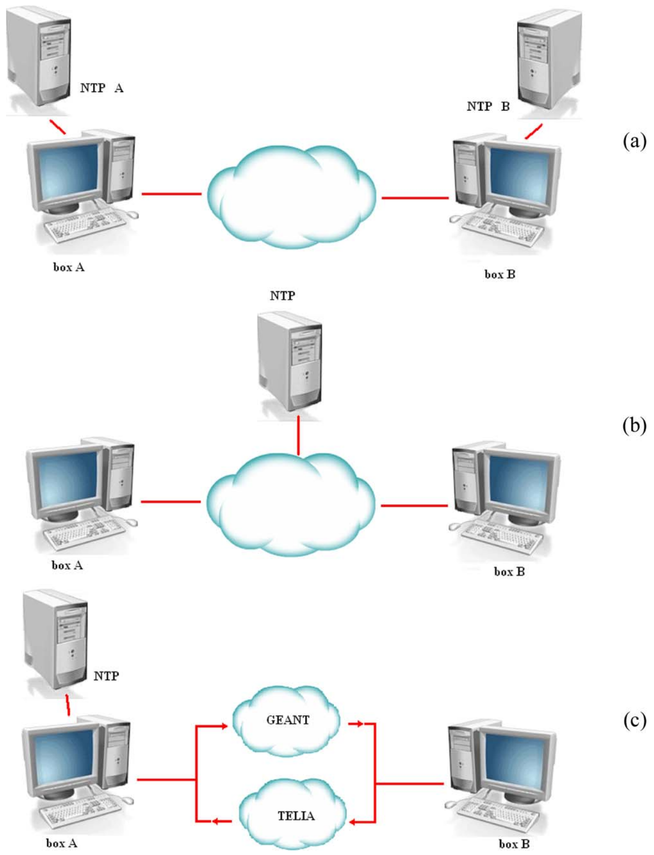


Fig. 3. Three OWD measurement setups using NTP. (a) NTP servers located in MPs. (b) NTP servers located outside MPs. (c) The NTP server tests the influence of asymmetric routing.

precisely synchronized with the tick of the second. This method achieves accuracies to within microseconds [36].

3) *Precision-time kernel:* The precision-time kernel support, which is incorporated into the OS kernels such as Solaris and Linux, improves the resolution of the kernel clock to the microsecond (not necessarily microsecond accuracy). Precision-time kernels are defined in RFC 1589 [37].

In the literature, there are many systems using NTP as the synchronization protocol for measuring OWD. In [10], the OWD measurement is performed using three setups with different locations of NTP servers. In Setup I [see Fig. 3(a)], two NTP servers located in the measurement points (MPs) with RTDs of less then 1 ms have been used, i.e., the NTP servers are located in the same network as the sender and receiver. In Setup II [see Fig. 3(b)], one common NTP server located in a third network outside both MPs, i.e., in the GEANT network [38], is used. In Setup III [see Fig. 3(c)], one common NTP server located at the Czech Educational and Scientific Network (CESNET) [39] is used. In the first architecture, the RTD between each MP and its local NTP server was less than 1 ms, and the uncertainty of the OWD measurement had values below ±500 μs. In Setup II, a single common NTP server located at a third network was used. In this case, the conditions for good time synchronization were much worse compared with those in Setup I. Consequently, the reported results are much worse than those for Setup I, but the uncertainty on OWD measurement is still far below ±1 ms. In Setup III, the influence of asymmetric routing has been tested. The traffic in one direction is routed through the Pan-European GEANT network (OWD of about 20 ms), and traffic in the opposite direction is routed via the Telia network, which supplies the commodity Internet traffic to CESNET (OWD of about 37 ms). To demonstrate the influence of asymmetric routing, one common NTP server located in Ireland was used. The average error of OWD measurement was 8 ms, which is about one half of the OWD difference in both directions. This way, it is possible to demonstrate that the OWD measurements are influenced by network asymmetry.

The synchronization by means of NTP can be improved by a hardware solution [11]. In this case, some specific hardware cards are used, which are capable of inserting a timestamp in the packets just before sending them. The clock, from which the timestamp is taken, needs to be logically synchronized to one card clock, which, in turn, is synchronized with an NTP server. With this method, a high accuracy in the insertion of the timestamps can thus be obtained (less than 100 ns), as compared with the timestamp insertion made in software, which gives an uncertainty of about 10 μs.

IV. GPS-Based Synchronization

Several architectures relying on the GPS system for measuring OWD can be found in the literature. GPS is used to determine the precise location of a GPS receiver.

The GPS system is based on 24 satellites rotating on six different orbital planes with a 12-h orbital period. It ensures that at least five, but usually six or more, satellites are visible any time, anywhere on the earth. The GPS receiver calculates more or less precise position (latitude, longitude, and altitude) and time based on signals received from four satellites (three are used to determine the position and the fourth is used to correct the time), which are broadcasting their current positions and UTC time [40].

With the installation of a GPS system, each internal computer software clock can be synchronized to the GPS clock.

The Communication Measurement (CM) Toolset proposed in [12] offers measurement features for the evaluation of data communication protocols and networks. The CM Toolset consists of a communication platform, graphical user interface, and database (DB) that manages the test parameters and the measurement results for each test and user. Each test has its start time, and so, it is possible to make a time schedule to repeat tests daily, weekly, and so on. The CMCaller is the management module of the CM Toolset. It is responsible for the test scheduling and the handling of the measurement data. It also connects the DB and the remote MPs. The CMDaemon controls the start of the measurement. The generator generates different kinds of traffic between two or more MPs. At each remote MP, the system clock is synchronized with this GPS time. The GPS clock integration in the CM Toolset allows OWD measurements with an accuracy to within about 10 ms.

Another architecture that relies on GPS, as described in [13], is composed of several MPs, a measurement system, and a data collector (DC). An MP samples packets and builds a report containing the packet's timestamp and an identifier (packet ID). These reports are then sent to the DC, where the OWD is evaluated. The DC parses the response messages received from the MPs, matches packet records with the same task ID and packet ID values, and calculates the OWD. The operation of the DC once the packet OWD is calculated stores the delay values in a local DB. Reference [14] is focused on achieving the better tradeoff between the cost and accuracy of the OWD measurement system, i.e., to obtain the needed accuracy (not the highest accuracy) with respect to the application needs while keeping both costs and uncertainty as low as possible. Furthermore, in [14], it is suggested that an accuracy on the order of tens of microseconds is reliable enough, with the typical delays being in the millisecond range.

The architecture shown in Fig. 4 [15] uses an emitter called the Real-time User datagram protocol Data Emitter (RUDE) and a receiver called the Collector for RUDE (CRUDE), like in [41]. RUDE is a small and flexible program that generates traffic to the network, which can be received and logged on the other side of the network with CRUDE. Currently, these programs generate and measure only User Datagram Protocol traffic [41]. A tool called *Qosplot* reads packet snapshots produced by CRUDE, computes all quality-of-service (QoS) characteristics, and produces diagrams depicting characteristics such as OWD and jitter. An example diagram of OWD obtained from *Qosplot* is shown in Fig. 5.

In [16], a measurement system architecture (active measurement tool) is proposed. This measurement system relies on the GPS system for synchronization and measures the OWD as follows:

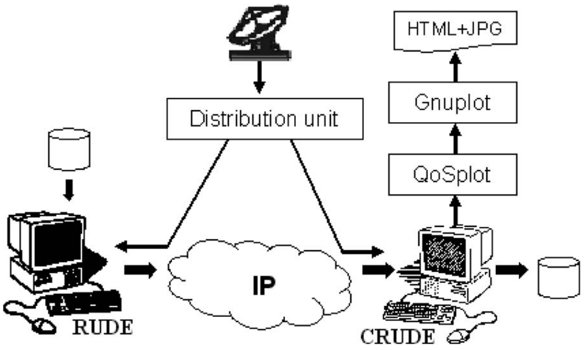


Fig. 4. RUDE/CRUDE system architecture.

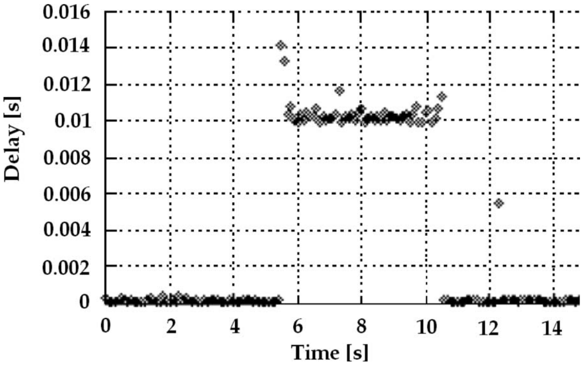


Fig. 5. OWD displayed with the *QoSplot*.

1) initialization of a registry that receives all control messages called asynchronous transfer mode daemon (ATMD);
2) fork of measurement processes;
3) establishment of control channel used to communicate with ATMD;
4) confirmation of readiness from ATMD with an acknowledgment message;
5) transmission of measurement start message;
6) start of actual measurement;
7) injection of measurement packets;
8) storage and analysis of measurement records.

This architecture allows the efficient and steady execution of one-way measurement, i.e., on the order of microseconds.

V. IEEE 1588 Standard-Based Synchronization

The IEEE 1588 "Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems" [31] was established to provide a common sense of time in distributed systems found in many measurement and control applications [17], as well as in telecommunication networks [18]. IEEE 1588 defines a protocol enabling precise clock synchronization in measurement and control systems implemented with technologies such as network communication, local computing, and distributed objects [19].

The protocol supports system-wide synchronization accuracy in the submicrosecond range with minimal network and local clock computing resources [42]. In particular, maximum accuracy in the range of 100 ns is only possible with hardware support; on the other hand, a pure software implementation on standard components provides precision in the range of 10–100 μs.

The synchronization process is divided into two phases [30]. First, the time difference between master and slave is measured and corrected. During this offset correction, the master cyclically transmits a unique synchronization (SYNC) message to the related slave clocks at defined intervals (by default, every 2 s). This SYNC message contains an estimated value for the exact time the message was transmitted. The second phase of the synchronization process, i.e., the delay measurement, determines the delay or latency between slave and master. For this purpose, the slave clock sends a so-called "delay request" packet to the master and determines the exact transmission time of the message.

The design and implementation of two IEEE 1588 prototypes for wireless LAN (WLAN) are presented in [20]. The former is implemented using a Linux PC platform and a standard IEEE 802.11 WLAN with modifications to the network device driver. The latter prototype is implemented using an embedded WLAN development board that provides the synchronization function by means of programmable logic device (PLD) circuits. The results reported in [20] show that synchronization accuracy obtained by PLD is almost three decades better than that with Linux PC implementation. The extremely high accuracy of this prototype mainly results from reducing the outbound and inbound latencies using hardware for timestamping.

Reference [43] reports that IEEE 1588 is capable of synchronizing the internal clocks of networked equipment to within 10 μs, because it requires hardware in every network node that can adjust its internal clock. In [21], a software-based master-selection mechanism that will tolerate faulty master clocks and maintain synchronization in any IEEE 1588 subnet is proposed. The proposed algorithm was originally developed for a controller area network, which is also a multicast network, and deterministically guarantees an upper bound for the clock skew and the message overhead. This approach can achieve the synchronization precision up to 1 μs.

Some papers, such as [22], deal with the sources of timing fluctuations in an IEEE 1588 system to improve synchronization techniques. A major source of timing errors is the network latency fluctuations introduced by network elements such as repeaters, switches, and routers [22].

Additional sources of fluctuation within a node are the oscillators. It is necessary to pay attention to the oscillators, because they are subject to frequency errors. These errors result from the sensitivity of oscillator frequency to temperature, pressure, supply voltage, etc. Temperature is the most difficult parameter to control in the target environments. There are only three ways to improve this situation: 1) Decrease the sampling interval (this was chosen as a compromise between responsiveness and network loading); 2) adopt oscillators with better temperature coefficients; and 3) carefully design from the thermal point of view, which can minimize the thermal drift rates [44]. Currently, several research groups are working on real-time solutions for Ethernet. In [23], IEEE 1588, when applied over an Ethernet in a network of measuring devices equipped with displacement sensors, provides a synchronization accuracy on the order of 20 μs.

Improvements for synchronization are expected with the new version of the IEEE 1588 Standard. It features improved support for large redundant networks and high-performance telecommunications applications. Moreover, IEEE 1588 version 2 increases the maximum number of timing packets from 1/s to a value ranging from 30 to 40/s. Increasing the number of packets per second reduces the impact when a packet is discarded or has a long delay. This has the effect of improving accuracy and making it possible to reduce the cost of slaves by using less-expensive oscillators. IEEE 1588 version 2 also features a unicast transmission mode, in addition to the multicast transmission mode used in IEEE 1588 version 1. The addition of unicast transmission makes it possible to independently tune each client, which improves timing accuracy [45], [46]. In [24], Cosart presents results and analysis of measurements taken in laboratory networks and production networks in various locations throughout the world, increasing synchronization rates available in IEEE 1588 version 2. Networks of various types, ranging from LANs to WANs, are studied, and packet delays ranging from 335 μs to 5.08 ms are shown.

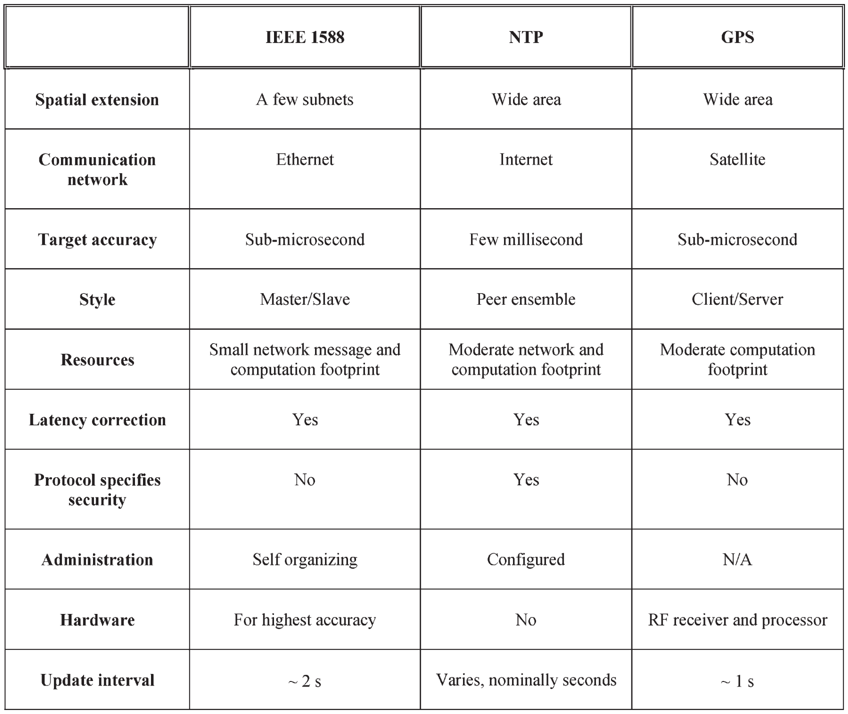
VI. Comparison and Comments on OWD Measurement Methods

A comparison of some characteristics of the solutions proposed in the literature to measure the OWD is reported in Table II. To obtain a specified accuracy in OWD measurement, it is necessary to consider six elements.

1) accuracy demanded: submicroseconds in IEEE 1588 and GPS, and a few milliseconds in NTP;
2) satellite coverage, which is required in the GPS-based synchronization;
3) availability of low-cost timing sources;
4) required additional hardware (boards/antennas) (only NTP does not need dedicated hardware);
5) network typology (LAN/WAN): IEEE 1588 applicable only to small networks;
6) application environment.

The most common methods for synchronizing clocks distributed over a network are based on the NTP. These methods are quite common in LANs or on the Internet and allow accuracies in the millisecond range [20]. NTP works by making occasional adjustments that aim at a long-term accuracy and interferes with the measurement process itself. Time synchronization protocols, such as NTP, do not provide enough accuracy for industrial automation applications [47]. Another possibility is the use of radio signals of the GPS satellites by adjusting each computer clock using the PPS signal from a GPS receiver. However, this requires relatively expensive GPS receivers, the appropriate antennas on the roof, and the necessary cabling [20]. In fact, GPS receivers require direct "intervisibility" with the satellites. This can become a severe problem since measurement devices are frequently positioned close to the main nodes of the infrastructure, which are frequently located in server rooms that may lack a window or may be located at the basement. Although this solution provides high accuracy,

TABLE II
COMPARISON AMONG VARIOUS METHODS FOR CLOCK SYNCHRONIZATION



	IEEE 1588	NTP	GPS
Spatial extension	A few subnets	Wide area	Wide area
Communication network	Ethernet	Internet	Satellite
Target accuracy	Sub-microsecond	Few millisecond	Sub-microsecond
Style	Master/Slave	Peer ensemble	Client/Server
Resources	Small network message and computation footprint	Moderate network and computation footprint	Moderate computation footprint
Latency correction	Yes	Yes	Yes
Protocol specifies security	No	Yes	No
Administration	Self organizing	Configured	N/A
Hardware	For highest accuracy	No	RF receiver and processor
Update interval	~ 2 s	Varies, nominally seconds	~ 1 s

it is often impractical for reasons of cost and effort [48]. Furthermore, GPS is still more expensive compared to other solutions [9].

Finally, the method reported in the IEEE 1588 Standard offers high accuracy over a data network but requires additional hardware, such as a network interface card, and its applicability is restricted to LANs (or metropolitan area networks) [49]. Alternatively, the IEEE 1588 Standard can be useful [47] in the industrial automation field.

Reference [48] assigns to the IEEE 1588 Standard an accuracy that is better than 1 μs. It is applicable to common and inexpensive networks, including but not limited to Ethernet and to heterogeneous systems where clocks of different capabilities can synchronize each other in a well-ordered manner with minimal computing and hardware resource requirements.

Currently, research is oriented to allow the synchronization of computers with high accuracy, for example, in a distributed network, without dedicated hardware.

VII. CONCLUSION AND FUTURE RESEARCH TRENDS

OWD measurements over a computer network are necessary to guarantee or improve the performance of network communication involved with real-time applications, such as voice over Internet Protocol and video conferencing. These applications require low OWD. Therefore, it is important to measure the OWD with minimum uncertainty.

This paper presented a wide review of this research on the measurement of OWD. Attention has mainly been devoted to the first uncertainty source, affecting the OWD measurement, i.e., the synchronization of the computers conducting the measurement. The reviewed works have been structured within the following three main classes, based on the solution that each paper that was taken into consideration proposes this problem, with each one using a different synchronization method:

1) synchronization by means of the NTP;
2) synchronization by means of the GPS;
3) synchronization by means of the IEEE 1588 Standard.

This paper provides a reference to researchers and software designers faced with OWD measurement or the measurement of other network parameters. Most of the network metrics, in fact, are related to time and often require synchronization between two hosts.

The analysis of the different approaches to the synchronization put into evidence some weak points. The NTP is the method that does not provide enough accuracy for industrial automation applications (only a few milliseconds). The GPS, on the other hand, provides high accuracy (submicroseconds), but it is often impractical for reasons of cost and effort. Finally, IEEE 1588 offers as high accuracy (submicroseconds) as the GPS method but requires additional hardware, and it is applicable just to LANs. Therefore, to accurately estimate OWD, it is necessary to analyze the application requirements (industrial, telecommunication, and military), degree of accuracy, and offered budget.

The trend of future research in OWD measurement goes, on the one hand, in the direction of reducing the uncertainties introduced by the software layers and by the OS. An important issue is still the synchronization, which remains as the first uncertainty source. In this case, the challenge is to reduce such uncertainty without adding a specific hardware, using network synchronization and conventional network cards.

REFERENCES

[1] J. Corral, G. Texier, and L. Toutain, “End-to-end active measurement architecture in IP networks (SATURNE),” in *Proc. PAM Workshop*, 2003, pp. 241–247.

[2] A. Hanemann, A. Liakopoulos, M. Molina, and D. M. Swany, “A study on network performance metrics and their composition,” *Campus-Wide Inform. Syst.*, vol. 23, no. 4, pp. 268–282, 2006.

[3] G. Almes, S. Kalidindi, and M. Zekauskas, *A One-Way Delay Metric for IPPM*, RFC 2679, Sep. 1999. IETF.

[4] G. Almes, S. Kalidindi, and M. Zekauskas, *A Round-Trip Delay Metric for IPPM*, RFC 2681, Sep. 1999.

[5] C. Demichelis and P. Chimento, *IP Packet Delay Variation Metric for IP Performance Metrics (IPPM)*, RFC 3393, Nov. 2002.

[6] A. Hernandez and E. Magaña, “One-way delay measurement and characterization,” in *Proc. 3rd ICNS*, Jun. 2007, pp. 114–120.

[7] N. Simar, *A First Step in the World of IP QoS*, pp. 1–11, Sep. 2001. DANTE in print, no. 43.

[8] W. U. Zaman, S. A. Ahmad, A. Abbas, and A. Qadeer, “Modeling of processor delay and overall reduction in network latencies for real time,” in *Proc. ISCON Emerging Technol.*, Aug. 2002, vol. 1, pp. 76–79.

[9] O. Gurewitz, I. Cidon, and M. Sidi, “Network time synchronization using clock offset optimization,” in *Proc. 11th IEEE ICNP*, Atlanta, GA, Nov. 2003, pp. 212–221.

[10] V. Smotlacha, “One-way delay measurement using NTP synchronization,” in *Proc. TERENA Networking Conf.*, Zagreb, Croatia, May 2003.

[11] D. Constantinescu, P. Carlsson, A. Popescu, and A. A. Nilsson, “Measurement of one-way Internet packet delay,” in *Proc. 17th NTS*, Oslo, Norway, Aug. 2004.

[12] U. Hofmann, T. Pfeiffenberger, and B. Hechenleitner, “One-way-delay measurements with CM toolset,” in *Proc. Int. Performance Comput. Commun. Conf.*, Phoenix, AZ, 2000, pp. 41–47.

[13] S. Niccolini, M. Molina, F. Raspall, and S. Tartarelli, “Design and implementation of a one way delay passive measurement system,” in *Proc. Netw. Oper. Manage. Symp.*, Seoul, Korea, Apr. 2004, vol. 1, pp. 469–482.

[14] D. Adami, R. G. Garroppo, S. Giordano, and S. Lucci, “On synchronization technique: Performance and impact on time meter monitoring,” *Int. J. Commun. Syst.*, vol. 16, no. 4, pp. 273–290, May 2003.

[15] S. Ubik and V. Smotlacha, “Precise measurement of one-way delay and analysis of synchronization issues,” in *Proc. TERENA Netw. Conf.*, Zagreb, Croatia, May 2003.

[16] J. Jeong, S. Lee, Y. Kim, and Y. Choi, *Design and Implementation of One-Way IP Performance Measurement Tool*, vol. 2343. London, U.K.: Springer-Verlag, Jun. 2002, pp. 673–686. no. 2.

[17] J. C. Eidson and K. Lee, “Sharing a common sense of time,” *IEEE Instrum. Meas. Mag.*, vol. 6, no. 1, pp. 26–32, Mar. 2003.

[18] D. Tonks, “IEEE 1588 in telecommunications applications,” in *Proc. Conf. IEEE 1588*, Sep. 2004, pp. 184–189.

[19] J. Jasperneite, K. Shehab, and K. Weber, “Enhancements to the time synchronization standard IEEE-1588 for a system of cascaded bridges,” in *Proc. IEEE Int. Workshop Factory Commun. Syst.*, Vienna, Austria, Sep. 2004, pp. 239–244.

[20] J. Kannisto, T. Vanhatupa, M. Hännikäinen, and T. D. Hämäläinen, “Software and hardware prototypes of the IEEE 1588 precision time protocol on wireless LAN,” in *Proc. 14th IEEE Workshop Local Metropolitan Area Netw.*, Chania, Greece, Sep. 18–21, 2005.

[21] D. Lee and G. Allan, “A solution for fault-tolerant IEEE-1588,” in *Proc. Workshop IEEE-1588*, Sep. 2003, pp. 225–231.

[22] J. C. Eidson and B. Hamilton, “IEEE-1588 node synchronization improvement by high stability oscillators,” in *Proc. Workshop IEEE-1588*, Sep. 2003, pp. 102–112.

[23] P. Ferrari, A. Flammini, D. Marioli, and A. Taroni, “IEEE 1588-based synchronization system for a displacement sensor network,” *IEEE Trans. Instrum. Meas.*, vol. 57, no. 2, pp. 254–260, Feb. 2008.

[24] L. Cosart, “Precision packet delay measurements using IEEE 1588v2,” in *Proc. IEEE ISPCS Meas. Control Commun.*, Oct. 2007, pp. 85–91.

[25] J. Wang, J. Yang, G. Xie, Z. Li, and M. Zhou, “On-line estimating skew in one-way delay measurement,” in *Proc. 4th Int. Conf. PDCAT*, Aug. 2003, pp. 430–436.

[26] J. H. Choi and C. Yoo, “One-way delay estimation and its application,” *Comput. Commun.*, vol. 28, no. 7, pp. 819–828, May 2005.

[27] Spirent Communications, *Remote distributed testing*. [Online]. Available: http://www.spirentcom.com/documents/185.pdf

[28] G. Gaderer, R. T. Sauter, and H. Muhr, “Extending IEEE 1588 to fault tolerant clock synchronization,” in *Proc. IEEE Int. Workshop Factory Commun. Syst.*, Vienna, Austria, Sep. 2004, pp. 353–357.

[29] T. Zseby, S. Zander, and G. Carle, “Evaluation of building blocks for passive one-way-delay measurements,” in *Proc. PAM Workshop*, Amsterdam, The Netherlands, Apr. 2001.

[30] *IEEE 1588 Precision Clock Synchronization Protocol for Networked Measurement and Control Systems*, 1st ed., IEC 61588, 2004.

[31] *Draft Recommended Practice for Network Communication in Electric Power Substations*, IEEE P1615/D2, Jan. 2006.

[32] D. L. Mills, *Network Time Protocol, ver 3, Specification, Implementation and Analysis*, RFC 1305, Mar. 1992.

[33] A. S. Sethi, H. Gao, and D. L. Mills, “Management of the Network Time Protocol (NTP) With SNMP,” Univ. Delaware, Newark, DE, Tech. Rep. 98-09, Nov. 23, 1997.

[34] D. L. Mills, “Internet time synchronization: The network time protocol,” *IEEE Trans. Commun.*, vol. 39, no. 10, pp. 1482–1493, Oct. 1991.

[35] D. Deeths and G. Brunette, *Using NTP to Control and Synchronize System Clocks—Part I: Introduction to NTP2*. Sta. Clara, CA: Sun BluePrints OnLine, Jul. 2001. [Online]. Available: http://www.sun.com/blueprints/0701/NTP.pdf

[36] P. Holleczek, R. Karch, R. Kleineisel, S. Kraft, J. Reinwand, and V. Venus, “Statistical characteristics of active IP one way delay measurements,” in *Proc. ICNS*, 2006, pp. 1–5.

[37] D. Mills, *A Kernel Model for Precision Timekeeping*, RFC 1589, Mar. 1994.

[38] *GÉANT Project*. [Online]. Available: http://www.geant.net/

[39] [Online]. Available: http://www.ces.net/

[40] *MikroTik RouterOS V2.8*. [Online]. Available: http://www.mikrotik.com/docs /ros/2.8/system/gps

[41] J. Laine, S. Saaristo, and R. Prior, *RUDE & CRUDE: Real-time UDP Data Emitter and Collector*. [Online]. Available: http://www.atm.tut.fi/rude/

[42] *Precision clock synchronization protocol for networked measurement and control systems*, IEEE 1588, 2004 IEC 61588.

[43] M. Rowe, “IEEE 1588 keeps time in sync,” *Test & Measurement World*, 2005. [Online]. Available: http://www.reed-electronics.com/tmworld/article/CA529830?text=synthetic+instruments

[44] *IEEE 1588 Tutorial*. [Online]. Available: http://ines.zhwin.ch/index.php

[45] A. Dopplinger and J. Innis, “Using IEEE 1588 for synchronization of network-connected devices,” *Industrial Control Design Line*, 2007. [Online]. Available: http://www.embedded.com

[46] J. Bennington, *IEEE 1588 Precision Timing Protocol (PTP)*, 2007. [Online]. Available: http://eepn.com/locator/products/ArticleID/34270/34270.html, Electronic Design

[47] A. Adhikari, L. Denby, C. Mallows, J. Meloche, and B. Rao, *Measuring Network One-Way Transit Time*. [Online]. Available: www.research.avayalabs.com/ techreport/ALR-2003-051-paper.pdf

[48] P. Sharma, *IEEE 1588 In Network Processors for Next-Generation Industrial Automation Solutions*. [Online]. Available: www.intel.com/technology/magazine/communications/IXP465-NPU-0505.htm

[49] H. Weibel, “High precision clock synchronization according to IEEE 1588: Implementation and performance issues,” in *Proc. Embedded World*, Nurenberg, Germany, Feb. 2005, pp. 22–24.

Luca De Vito received the M.S. degree (*cum laude*) in software engineering and the Ph.D. degree in information engineering from the University of Sannio, Benevento, Italy, in 2001 and 2005, respectively. His M.S. thesis dealt with the automatic classification and characterization of digitally modulated signals.

He is currently with the Laboratory of Signal Processing and Measurement Information, Department of Engineering, University of Sannio, working on digital processing for measurement in telecommunications and data converters. He is also with the Benevento Research Laboratory, Telsey Telecommunications S.p.A, San Giorgio del Sannio, Italy, developing software for residential gateways.

Sergio Rapuano (M'00) received the M.S. degree (*cum laude*) in electronic engineering and the Ph.D. degree in computer science, telecommunications, and applied electromagnetism from the University of Salerno, Salerno, Italy, in 1999 and 2003, respectively.

In 2002, he joined the Department of Engineering, University of Sannio, Benevento, Italy, as an Assistant Professor of electric and electronic measurements. His current research interests include digital signal processing for measurement in telecommunications, data converters, and distributed measurement systems.

Prof. Rapuano is a member of the Waveform Generation, Measurement, and Analysis Technical Committee (TC-10) of the IEEE Instrumentation and Measurement (I&M) Society and is the Secretary of the Working Group on E-Tools for Education in Instrumentation and Measurement formed within the Technical Committee on Education for Instrumentation and Measurement (TC-23) of the I&M Society.

Laura Tomaciello was born in Benevento, Italy, on August 18, 1978. She received the master's degree in software engineering from University of Sannio, Benevento, in 2004, presenting a thesis dealing with the analysis and optimization of a fiber-optic measurement system based on Bragg gratings. Since 2005, she has been pursuing the Ph.D. degree in information technology with the University of Sannio.

At the present time, she is with the Laboratory of Signal Processing and Measurement Information, University of Sannio, developing her research in the network measurement field. She is also with Telsey Telecommunications, San Giorgio del Sannio, Italy, where she is developing research in network access device testing and validation.