High Availability IEEE 1588 nodes over IEEE 802.1aq Shortest Path Bridging Networks

P. Ferrari, A. Flammini, S. Rinaldi
Dept. of Information Engineering
University of Brescia
Via Branze, 38 – 25123 – Brescia, Italy
stefano.rinaldi@ing.unibs.it

G. Prytz
ABB Corporate Research
Bergerveien 12
NO-1396 Billingstad, Norway
Gunnar.Prytz@no.abb.com

***Abstract*—The modern industrial and substation automation Ethernet networks have demanding requirements about high availability. Recently, the IEEE 802.1 WG proposes the IEEE 802.1aq, also known as Shortest Path Bridging, to manage Ethernet redundant networks. The IEEE 1588 protocol can exploit the SPB capabilities to manage multiple paths in order to improve synchronization availability. In the paper, the possibility to send copies of IEEE 1588 messages over multiple paths has been analyzed. In particular, the attention has been focused on the combining algorithms in the slaves, which is used to recover the time from multiple sources. In details, some specific fault tolerant combining algorithms, based on synchronization quality estimation have been proposed. The considered solutions are compared to traditional approaches under fault conditions by means of a specifically developed simulation environment.**

Keywords—IEEE 1588; IEEE 802.1aq; Fault tolerant synchronization; Time Synchronization; Fault Masking; Combining algorithm

I. INTRODUCTION

Availability is a key requirement in modern industrial automation systems [1]. Industrial automation systems grow in size and complexity and the cost associated with system downtime is often very high, thus the requirements on availability become more and more challenging. Some systems found in process, factory and power automation cannot tolerate communication downtime for more than a few milliseconds. It is challenging to fulfill such requirements today [2] but several solutions specified in the IEC 62439:2010 standard can be used in some situations. PRP (Parallel Redundancy Protocol) and HSR (High Availability Seamless Redundancy) are two of such examples of solutions utilizing parallel communication along two different paths (IEC 62439-3). There are also communication protocol specific solutions, such as the PROFINET system redundancy approach (IEC 61784-2:2010), which can be used.

Recently, however, the development of new communication solutions such as SPB (Shortest Path Bridging), IEEE 802.1aq-2012, and TRILL (Transparent Interconnection of Lots of Links), IETF RFC 6325, has changed the picture significantly. SPB and TRILL are technologies coming from the high performance computing and data center networking domains. Although the base specifications are in place there are still a lot of activities in standardization bodies focusing on the further development of these technologies and how they should be applied. It can be expected that these technologies will be increasingly supported by standard components (switching silicon and network software stacks) in the near future. Due to the sales volumes and corresponding attractive pricing of standard networking components such solutions are thus very relevant for industrial automation [3]. Both SPB and TRILL are technologies that enable the simultaneous usage of multiple communication paths in a network. This is in contrast with the traditional RSTP (Rapid Spanning Tree Protocol), IEEE 802.1D-2004, technology which turns any network into a logical tree, i.e. there is only one path between any two nodes. RSTP typically does not utilize the available bandwidth in the network in a good way as all data traffic takes place in the same logical tree. This drawback has been overcome in SPB and TRILL which, when possible, define multiple paths between the nodes. Thus SPB and TRILL enable multipathing and may increase the fault tolerance and thus the availability of a system if traffic is allowed to be sent simultaneously over several paths. Although not focused upon in the initial SPB and TRILL specifications, the ability to send the same data on several paths is currently being studied. Such a feature is very interesting for industrial and smart grid automation systems as well as for telecommunication system and is a foundation for the discussions in this paper.

Alongside availability, time synchronization is another key requirement in many industrial systems. IEEE 1588-2008 is a solution for high accuracy time synchronization in modern networks. One of its main application fields is the industrial automation. The challenge is, however, to combine time synchronization and availability in an optimal and robust manner. In parallel with increasingly challenging requirements in industrial and telecommunication systems it is crucial to be able to fulfill these two requirements together. This is currently being discussed within several relevant standardization working groups. This paper is aiming to start the evaluation of IEEE 1588 devices connected to SPB networks under some typical industrial scenarios to improve the reliability of the synchronization. In particular, the paper is focused on the capabilities of an IEEE 1588 slave to handle information coming from multiple paths. Nodes in future SPB networks will potentially receive synchronization information from many sources. There are many ways how this could be handled and some possibilities are discussed here.

This research activity has been partially funded by research grant MIUR 2009ZTT5N4 -PRIN 2009, “New generation hybrid networks in measurement and industrial automation applications - Characterization and wired/wireless performance measurement”, and by ABB research grant “AIRnetS”, whose support the authors gratefully acknowledge.

978-1-4799-0242-2/13/$31.00 ©2013 IEEE

II. IEEE 1588 OVER IS-IS SPB NETWORK

A. Shortest Path Bridging

The SPB, as specified in IEEE 802.1aq, is a networking technology which enable the management and the forming of multipath network. SPB can be considered as the replacement of the older Spanning Tree Protocols, which are able to create a single communication tree from an Ethernet mesh network, blocking some of the ports of the switches. On the contrary, SPB allows the entire path to be active with multiple equal cost paths, allowing a faster recovery time and improving the bandwidth usage, because traffic can be shared across all the paths of the mesh network. The SPB is able to create logical Ethernet networks over native Ethernet infrastructures by means of a link state protocol IS-IS (Intermediate System to Intermediate System), used to identify the topology and membership to the network. The packets are encapsulated at the edge and transported only to other nodes of the logical network. The SPB supports unicast as well as multicast. All the paths defined by the protocol are on symmetric shortest paths. Among the main advantages of this protocol: the possibility to use all the available physical link; fast recovery time after failure; only the nodes directly affected by the fault are impacted during the recovery phase; the traffic on the other paths continues without any problem; rapid recovery of broadcast and multicast connections.

B. IEEE 1588 Multipath Synchronization over IS-IS SPB

In the recent years, IEEE 1588 and its profiles, like IEEE C37.238-2011 and IEEE 802.1AS-2011, represent the leader technologies for accurate time synchronization in LAN (Local Area Network) and it is more and more adopted in new plant installation [4]. IEEE 1588 standard defines several mechanisms to improve the synchronization performance in case of network reconfiguration: alternate master mechanism; support to redundant network topologies. Nevertheless, the application of IEEE 1588 protocol in critical environment, like smart grid or industrial automation, requires an increase in the availability requirements. Currently, Sync packets can be lost during network reconfiguration, a behavior not acceptable in some critical systems. Therefore, fault masking mechanism, able to mask network failure from the synchronization point of view, are required. Recently, the research community proposes the transmission of the Sync message over multiple paths: in the case of a link failure, the other copies of Sync can reach the IEEE 1588 slaves following alternative paths. The management of multiple active paths is done with the IS-IS SPB link state protocol. The transmission of multiple copies of Sync on different paths requires a mechanism to identify them both in the switches and in the IEEE 1588 slave. The IEEE 1588 standard is technology independent, which means the protocol can be mapped over different network technologies. Some of them already provide mechanism to identify Syncs transmitted over multiple paths. For example, PRP identifies the packet coming from different paths using a dedicated LAN-ID. Nevertheless, these solutions cannot be generalized. Given the relevance of these issues, several solutions have been proposed [5]. More in general, a first approach could be the use of a unique MAC address for each synchronization path, but this requires a list of multicast MAC address in each intermediate nodes. An alternative solution could be the use of the IEEE 802.1Q tag to identify each path but VLAN-ID tag is explicitly forbidden by the IEEE 802.1AS profile. In the case IEEE 1588 is mapped over IP, different IP addresses can be assigned to each slaves and used to identify the packets coming from different paths. The main drawback of this solution is the use of unicast Syncs instead of multicast. A more interesting approach could be the definition of an additional PATH ID in the field of the PTP packets, to identify the path. In the rest of the work, this solution is considered in the implementation of the model of IEEE 1588 node although it has not yet been defined in the standard. As mentioned at the beginning of this section, the transmission of a Sync message over multiple paths requires the configuration of the network infrastructure. As proposed by IEEE 802.1 WG[6], the IS-IS SPB link state protocol can be used to properly configure the switches to create multiple IEEE 1588 paths. The configuration of the network requires several information about the synchronization topology (e.g. the number of switches and IEEE 1588 nodes, and more) and about end to end connectivity. This information can be exchanged among switches by means of link state PDU, supporting additional TLV dedicated to IEEE 1588 synchronization. Using this packet, the switches can define the IEEE 1588 topology and identify the best multiple paths.

III. FAULT-TOLERANT MULTIPATH SYNCHRONIZATION ALGORITHMS

Generally speaking, when multiple synchronization packets are received by an IEEE 1588 slave over different paths, a separate delay and offset computation per each path is executed. The algorithm adopted to merge multiple synchronization information is known as combining algorithm. IEEE 1588 nodes receiving multiple Syncs from different paths should implement combining algorithms to improve synchronization performance. In the following section, a brief overview of typical combining algorithms is provided and enhanced high availability algorithms are proposed and described in details.

A. Combining algorithms: overview

The combining algorithms proposed in the literature are typically designed to improve the synchronization accuracy, composing in the proper way multiple synchronization information [5]. The different combining algorithms can be classified into the following categories [7]: Averaging; Switching; Filtering–Clustering-Combining Algorithm. In the first class of algorithms, the slave simply averages the time offset estimation obtained on the different paths and use the mean of the time offset to update the local clock. An improved version of this algorithm weights each time offset by the inverse of the one way delay estimated on the related path before the averaging. In the second class, the slave periodically chooses a primary path and updates its local clock using only the Sync coming from the primary path. Typically, the primary path is the path with the lowest packet delay variation. For example, this algorithm is adopted for the synchronization on PRP/HSR networks. The third class includes the algorithms derived from the filtering-clustercombining algorithm used by NTP (Network Time Protocol), IETF RFC 5905, to compose time information coming from different servers. Each of these solutions has advantages and disadvantages, and therefore each of them is suitable for being used in different situations, but typically does not focus on the identification and filtering of synchronization coming from faulty paths.

B. Fault-Tolerant combining algorithms

In the literature, several fault tolerant clock synchronization algorithms have been proposed [8][9]. Typically, a fault tolerant combining algorithm should take into account the quality of the synchronization provided by each path. In the following section, an improved fault tolerant combining algorithm, derived from fault tolerant synchronization system described in [10], has been proposed for high availability multipath IEEE 1588 systems. An estimation of the quality of synchronization information provided by each path is performed (Path Quality Estimation). On the basis of the estimator, the convergence function filters and combines the multiple incoming synchronization information. Thus, the servo clock update the local clock using the combined time offset. It should be noted as different servo clock algorithms may lead to different synchronization performance. In this paper, a Proportional Integral (PI) controller, derived from [11] has been considered. A detailed analysis of different servo clock algorithms is out of the scope of this paper.

Consider a multipath IEEE 1588 systems, where each IEEE 1588 slave receives Sync messages from M different paths. Each slave, using the information provided by the Sync, estimates the time offset from the reference time, T_{OFFm}, where m is the mth path. In addition, the slave estimates an indicator of the quality of synchronization, δ_m, provided by the mth path, and defines the time offset interval $\tilde{T}_m = [T_{OFFm} - \delta_m^-; T_{OFFm} + \delta_m^+]$, which can be used to quantify and compare the quality of each synchronization path. After the reception of the last Sync, the PTP slave applies to the estimated $\tilde{T}_m$ a convergence function, to obtain an estimation of the time offset not affected by faulty paths. The performance of the algorithm relies on the filtering and combining actions defined by the convergence function. Two different approaches can be followed: the Overlapping and the Discarding. In the Overlapping, which convergence function is described in the flow diagram of Fig. 1.a, the not overlapping $\tilde{T}_m$, which could be affected by fault on the network, are removed. This algorithm is able also to remove systematic error, like path asymmetry, which affect only some of the paths. In the Discarding, the synchronization information coming from the path with the larger $\tilde{T}_m$ is always discarded (flow diagram shown in Fig. 1.b). The output of the convergence function is the interval $\tilde{T}$:

$$\tilde{T} = [T'_{OFF} - \Delta^-; T'_{OFF} + \Delta^+] \tag{1}$$

where T'_{OFF} is considered the best estimation of the time offset. As described by the diagram of Fig. 1.a and Fig. 1.b, the T'_{OFF} is defined as the mean of the filtered T_{OFFm}. It should be noted as to obtain a meaningful estimation of T'_{OFF} using the discarding algorithm, the number M of paths has to be at least 3, since the information coming from a path is always discarded. Then, this interval is used to compensate the local clock, adopting traditional servo clock mechanism. In the case the T_{OFFm} estimated during the observation interval (N samples) are normally distributed, the standard deviation, $\sigma_{m,N}$, can be considered a good indicator of the quality of the synchronization provided by the mth path. The unbiased estimator of $\sigma_{m,N}$ is the sample standard deviation:

$$s_N = \sqrt{\frac{1}{(N-1)} \sum_{n=1}^{N} \left(T_{OFF} - \overline{T}_{OFF}\right)^2} \tag{2}$$

where N is the number of samples considered in the estimation of the sample standard deviation and $\overline{T}_{OFF}$ is the sample mean of the time offset (over the last N samples). Note as N is an important parameters, which definition is the result of a tradeoff between the computational and storage requirements (the greater the number of sample considered, the larger is the dimension required by buffers) and an accurate estimation.

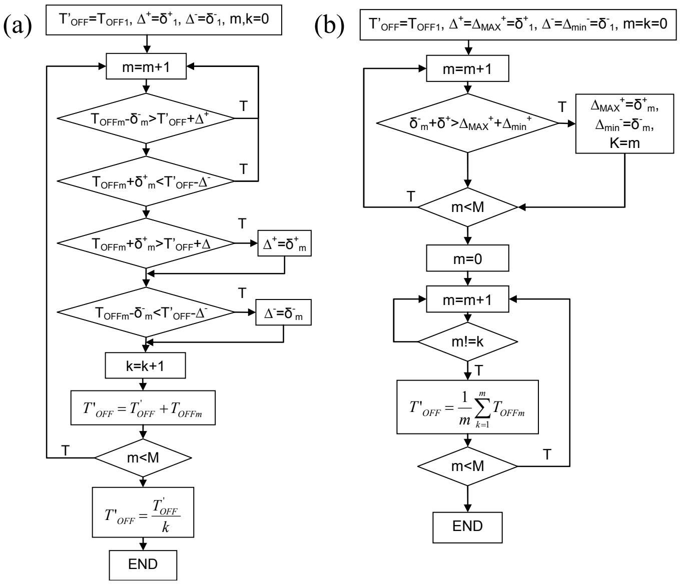

Fig. 1. The flow diagrams of the fault-tolerant combining algorithms: overlapping (a) and discarding algorithm (b).

In the rest of the analysis, N is set to 30 samples, enough to provide a meaningful estimation of s_N. Therefore, considering the time offset normally distributed ($\delta_m^- = \delta_m^+ = s_N$), the quality indicator of each time offset obtained on the mth path can be expressed as $\tilde{T}_m = [T_{OFFm} - 3s_N; T_{OFFm} + 3s_N]$.

IV. THE SIMULATION ENVIRONMENT

The evaluation of fault tolerant algorithms requires the use of simulation environment since failure conditions are difficult to reproduce in a real network environment under controlled conditions. Thus, a simulation environment for testing industrial and electrical substation automation, has been developed using OMNeT++ simulation environment [12]. The simulation of Real-Time Ethernet network requires the development of suitable network device models, like transparent clock switches (TC switches) and end-nodes, supporting the most recent protocols, like SPB and IEEE 1588, which are not already included in the available

Ethernet libraries. Each model can be easily configured to support the required protocols. In addition, several failure conditions, like link or switches faults, can be reproduced. The development of a simulation environment for the evaluation of synchronization system requires also an accurate modeling of hardware related components, like timestamping units and, in particular, clock, which key role is well demonstrated in [13]. Therefore, particular attention has been paid in the development of the clock model. A two state clock model [14], based on offset (θ) and skew (γ) has been implemented. This model is able to correctly model two of the most important noise contributions of a quartz crystal oscillator: the white frequency modulation noise (WFM) and random walk frequency modulation noise (RWFM). The main parameters of the clock model are summarized in TABLE I. The $\mathcal{U}$[a,b] represents an uniform distribution with support in [a,b]. High performance clocks are considered in the simulations, as clearly indicated by the value of σ_θ, σ_γ reported in the table. The initial clock offset and skew of each node are uniformly distributed, as indicated in TABLE I.

TABLE I. THE CLOCK PARAMETERS OF THE CLOCK MODEL.

Parameter	Description	Range
f_k	Nominal clock frequency	100 Mhz
γ_k	Clock Skew	$\mathcal{U}$[-10ppm,10 ppm]
$\theta_k(0)$	Initial Clock Offset	$\mathcal{U}$[-500 ms, 500 ms]
σ_θ	WFM noise	10^{-8} $s^{1/2}$
σ_γ	RWFM noise	10^{-9} $s^{-1/2}$

A. The simulation scenario

The models briefly described in the previous section, are used to implement the IEEE 1588v2 network (Fig. 2), in which the synchronization multipath tree is provided by IS-IS SPB (three parallel paths), as explained in section II.B. The network is used to analyze the behavior of combining algorithms, described in section III, in the case of faults. The IEEE 1588 nodes and TCs are supporting the peer to peer working mode. In a typical IEEE 1588 synchronization network, the failure points are [15]: the grand master clock, the slaves and the network. In particular, the analysis presented in this paper is focused on the last class of failures, whose impact can be mitigated by multiple copies of Sync transferred over parallel paths. In details, the different path failure conditions are classified in two groups: the path interruption and path performance degradation. The first class includes all the faults that cause the loss of the communication over an IEEE 1588 path: cable breakout, switch failure, power failure, fall within this class. The use of multiple paths masks the failure during the establishing of an alternative IEEE 1588 link. On the contrary, the second class includes all the faults which affect the quality of the synchronization information provided by the network: examples of this situation are congested network link, malicious attacks to TCs, timestamping devices with low accuracy, temporary decrease of synchronization quality of TCs. In this case, the network still works properly therefore no path reconfiguration is required. Nevertheless, the time information quality provided by the faulty path can severely affect the synchronization accuracy of the end-node. In the following analysis, three scenarios have been considered: no fault (ideal condition), path interruption and performance degradation. In the first scenario no fault is present on the network. The synchronization performance obtained by the different algorithms is considered as term of comparison.

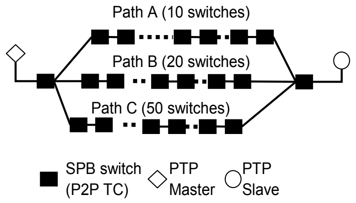

Fig. 2. The multi-path PTP synchronization tree considered in the analysis.

In the second scenario, a link failure (e.g. cable breakout) has been simulated: after the fault, an alternative IEEE 1588 connection is recovered over a longer path, but with similar behavior (from the synchronization point of view). In the third scenario, the failure of the syntonization algorithm of a TC on a single path has been simulated: after the fault, the TC stops to compensate the bridge delay of sync messages. The behavior of different algorithms implemented in the IEEE 1588 slave are analyzed and compared during the three scenarios. The parameters adopted to configure the simulation are obtained from measurement on a real industrial network, as explained in [16]. The most important parameters are summarized in TABLE II.

TABLE II. THE SIMULATION PARAMETERS.

Parameter	Description	Range
T_{sync}	Synchronization Interval	2 s
TS_G	Timestamp noise	$\mathcal{U}$[0 ns,10 ns]
b_i	Bridge delay	$\mathcal{E}$[K,λ], with K=1 and λ=2
Ld	Line Delay	$\mathcal{U}$[1602, 1608] ns
N_A	Switches on path A	10 before, 25 after link failure
N_B	Switches on path B	20
N_C	Switches on path C	50

The $\mathcal{U}$[a,b] represents an uniform distribution with support in [a,b], while $\mathcal{E}$[K,λ] represents an Erlang distribution, where K is the shape and λ the rate, typically adopted to model queuing systems. The timestamp noise is uniformly distributed, to model an IEEE 1588 hardware assisted device. Similarly, the Line delay can be modeled by a constant component, which causes an asymmetric path delay, and by a variable component which has been modeled using uniform distribution. For sake of simplicity, the Bridge delay has been modeled as an Erlang distribution, by extrapolating the experimental results reported in [17]. As an alternative approach, the traffic can be described by an analytical model. Typically the network traffic exhibits a strong correlation, also known as long range dependence (LRD), which led to a self-similar model of traffic [18].

V. THE SIMULATION RESULTS

A. Estimation of time information

The fault tolerant algorithms introduced in Section III are based on the estimation of the quality of synchronization provided by each IEEE 1588 path. As mentioned in the Section III, an interval-based quality indicator has been considered in this work. The width of this interval depends on the standard deviation of last 30 samples of the time offset. In the Fig. 3, the time offset evaluated (T_{OFF}) on each path has been plotted (solid line) with the interval, $\tilde{T}_m$, representing the quality of synchronization information (dotted line). As clearly indicated by the results, the longer is the path the greater is the interval, i.e. the greater the noise contributions. Consider, for example, the time quality interval, $\tilde{T}_C$ (dotted line) of the path C (Fig. 3.c), which includes 50 TC switches: $\tilde{T}_C$ is larger than $\tilde{T}_A$ (Fig. 3.a). Nevertheless, the difference between the information provided by the different paths (i.e. the interval $\tilde{T}_m$) is rather limited since IEEE 1588 nodes are supporting hardware synchronization. It should be noted as the quality estimator considered in the paper, based on standard deviation, does not quickly detect the path fault but require the collection of statistical data (i.e. 60 s with a synchronization interval of 2 s).

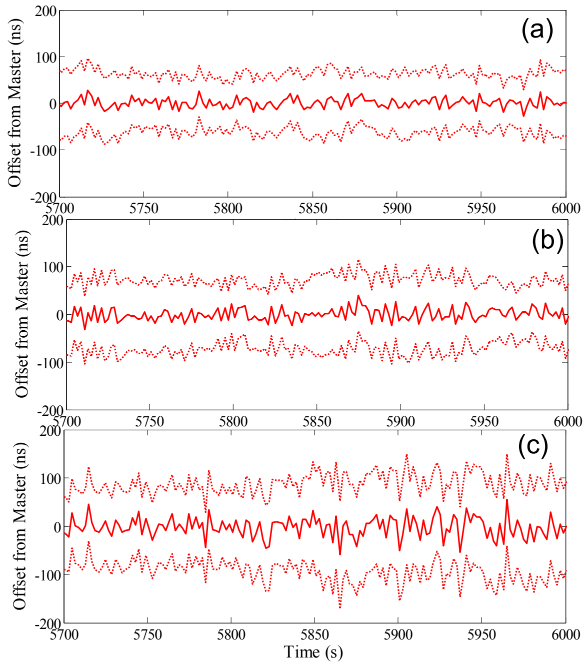

Fig. 3. The time offset (solid line) and the quality of the time (dotted line) synchronization provided by path A (a), by path B (b) and path C (c).

B. *No failure scenario*

In the first simulation scenario, no failure conditions are simulated. The Average and Switching algorithm are compared with fault tolerant combining algorithms: Overlapping and Discarding. The density estimations of the time offset obtained using the different algorithms over 3000 samples are compared in Fig. 4. As expected, the overall noise contribution is normally distributed, because of the composition of timestamping uncertainty (uniformly distributed) introduced by each TC. In this condition, the switching algorithm, which considers only the time offset coming from the shortest path (path A), provides the best results. In fact, the Averaging, Overlapping and Discarding algorithms work, in this situation, in a similar way, calculating the mean of time offset values provided by different paths. Nevertheless, paths B and C are affected by a larger noise contribution if compared to path A, limiting the estimation of time offset.

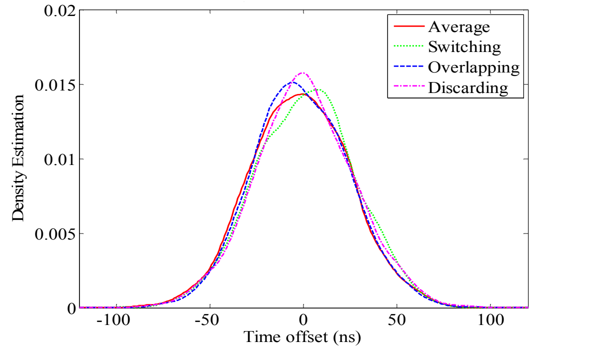

Fig. 4. Comparison of the density estimations of time offset evaluated using different combining algorithms. No Failure scenario.

C. *Path interrupt scenario*

In the second simulation scenario, a cable breakout affects the path A in t=6000 s. The fault causes the recovery of the synchronization on an alternative path, longer than the original (25 switches). The different algorithms are used to recover the time offset. The comparison of density estimations of time offset obtained using the different algorithms is shown in Fig. 5. Also in this case, the differences in the algorithms performance are limited because the noise contribution introduced by a chain of hardware assisted TCs is limited: the noise due to a longer path differs little from that of a shorter path.

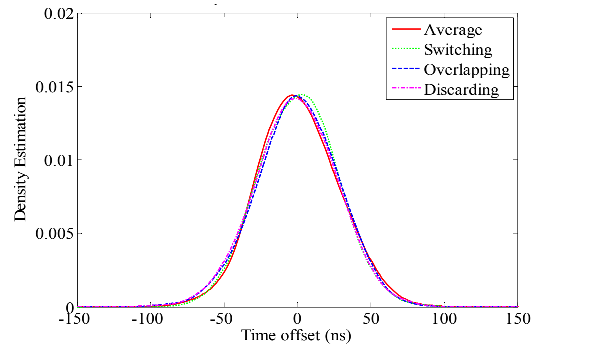

Fig. 5. Comparison of the density estimations of time offset evaluated using different combining algorithms after cable breakout on path A.

The Switching algorithm is still providing the best behavior since, after the failure on path A, the shortest, and the less noisy, path B is selected by the algorithm. The other algorithms, as in the previous case, estimate the time offset using the mean of the three paths.

D. *Path performance degradation scenario*

In the third scenario, a path performance degradation scenario has been evaluated, by simulating a TC over path A which stop, in t=6000 s, to compensate the bridge delay of the sync messages. The fault does not affect the network topology but introduces an additional noise contribution, which depend on the oscillator quality of the TC, on the bridge delay and on the network traffic. Compared to the previous case, the fault deeply affects the synchronization performance. The comparison of the density estimations of the time offset obtained using the different algorithms over 3000 sample is shown in Fig. 6. In this case, the Switching algorithm recovers the time synchronization information only from the shortest path (path A), affected by the fault, decreasing the performance: the synchronization jitter, i.e. the maximum variation of the time offset during the experiment, is on the order of 800 ns. The Average algorithm mitigates the effect of the fault, since the faulty information coming from path A are averaged with the correct time information coming from path B and path C. Nevertheless, also in this case the mean of the time offset is on the order of 100 ns and jitter on the order of 300 ns. In this scenario, the Discarding algorithm is able to identify the noisy path (path A) and to discard the corresponding time offset. The Discarding algorithm allows to obtain a time offset mean on the order of -3 ns, with a synchronization jitter in the order of 300 ns. Note as the Overlapping algorithm provides synchronization performance comparable to Average algorithm. In fact, the noise introduced by each TC is limited (and with zero mean) in a IEEE 1588 system with hardware support, and therefore the quality of the time information provided by different path is comparable also in case of a fault. Extending the previous fault scenario and considering an additional failure causing the performance degradation on the path B, the density estimations obtained using the different algorithms are compared in Fig. 7. As expected, the performance of the Switching algorithm remains the same, since the IEEE 1588 slave continues to recover the time information from the path A. On the contrary, the performance provided by the Discarding algorithm drops significantly because this algorithms is tolerant to one fault.

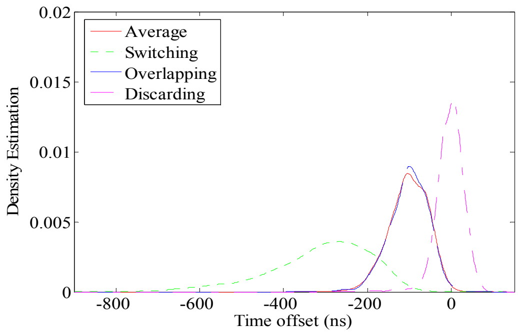

Fig. 6. Time offset density estimations obtained using different combining algorithms after performance degradation over paths A.

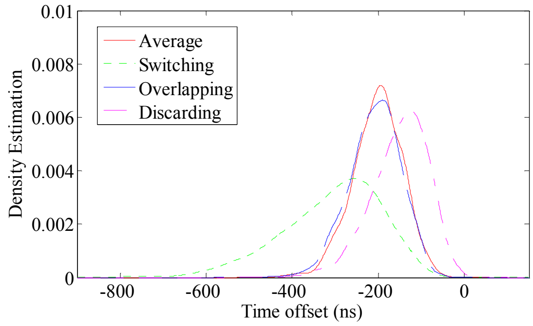

Fig. 7. Ttime offset density estimations obtained using different combining after failures causing performance degradation over two paths together.

VI. Conclusions

In the paper, the analysis of a high availability synchronization system based on IS-IS SPB has been presented. In particular, the analysis focuses on the algorithms required to compose multiple time information to tolerate path faults. In the paper two classes of failure have been considered: the path interruption and path performance degradation. In the case of a path interruption fault, the alternative (and longer) path recovered by the SPB protocol, introduces only negligible contributions. In the case of a failure which causes the performance degradation of a path, the Averaging and Switching algorithms show a sensible decrease of the performance because does not take into account the quality of the information received by each path. On the contrary, the proposed approach, based on estimation of the quality synchronization of each path and on the filtering of the time information provided by faulty paths is able to provide low synchronization jitter also in the case of path performance degradation.

References

[1] G. Prytz, L. Thrybom, "On the Usage of SPB and TRILL in Automation Systems", in *Proc. of ETFA2013*, 2013.

[2] G. Prytz, "Network recovery time measurements of RSTP in an ethernet ring topology", in *Proc. of ETFA,* pp. 1247-1253, 25 September 2007.

[3] J. Imtiaz, et al. "A proposal to integrate process data communication to IEEE 802.1 Audio Video Bridging (AVB)", in *Proc. of ETFA*, pp.1,8, 2011.

[4] C. M. De Dominicis, P. Ferrari, A. Flammini, S. Rinaldi, M. Quarantelli, "On the Use of IEEE 1588 in Existing IEC 61850-Based SASs: Current Behavior and Future Challenges", *IEEE Trans. Instrumentation and Measurement*, September, 2011, Vol. 60, N. 9, pp. 3070-3081.

[5] T. Mizrahi, "Slave diversity: Using multiple paths to improve the accuracy of clock synchronization protocols", in *Proc. of ISPCS* , pp.1,6, 2012.

[6] F.J. Goetz, "High Available Synchronization with IEEE 802.1AS bt",http://www.ieee802.org/1/files/public/docs2013/asbt-goetz-HighAvailableSync-0319-v02.pdf, 2013.

[7] A. Shpiner, R. Tse, C. Schelp, T. Mizrahi, "Multi-Path Time Synchronization", *Draft*, IETF, February 17, 2013.

[8] J. Welch-Lydelius, N. Lynch,"A new fault-tolerant algorithm for clock synchronization", Inf. Comput. Vol 1, 1988 pp.1-36.

[9] P. Ramanathan, et al. "Fault-tolerant clock synchronization in distributed systems". Computer , vol.23, no.10, pp.33,42, Oct. 1990.

[10] G. Gaderer, S. Rinaldi, N. Kero, "Master failures in the Precision Time Protocol", *in Proc. ISPCS 2008,* Sept. 22-26, 2008, pp. 59 - 64.

[11] K. Correll, et al. "Design Considerations for Software Only Implementations of the IEEE 1588 Precision Time Protocol", in Proc. of ISPCS, 2006.

[12] P. Ferrari, A. Flammini, S. Rinaldi, G. Prytz, "Analysis of Synchronization Performance in Redundant Networks for Substation Automation", in *Proc. of* I2MTC 2013, May 2013.

[13] G. Gaderer, et al. "A novel, high resolution oscillator model for DES systems" in *Proc. ISPCS 2008,* pp.178,183, 19-21 May 2008.

[14] G. Giorgi, et al. "Performance analysis of Kalman filter-based clock synchronization in IEEE 1588 networks", in *Proc. ISPCS*, pp.1-6, 2009.

[15] A. Bondavalli, F. Brancati, A. Flammini, S. Rinaldi, "Master Failure Detection Protocol in Internal Synchronization Environment", *IEEE Trans. Instrumentation and Measurement,* Vol. 62, N. 1, pp. 4-12, 2013.

[16] D. Fontanelli, D. Macii, S. Rinaldi, P. Ferrari, A. Flammini, "Performance Analysis of a Clock State Estimator for PROFINET IO IRT Synchronization", in *Proc. of I2MTC*, pp. 1828-1833, 2013.

[17] P. Ferrari, A. Flammini, D. Marioli, S. Rinaldi, A. Taroni, "Testing coexistence of different RTE protocols in the same network", in *Proc. of WFCS 2008*, pp. 179-187, 2008

[18] G. Giorgi, C. Narduzzi, "Modeling and Simulation Analysis of PTP Clock Servo", in *Proc. of ISPCS 2007*, pp.155,161, 1-3 Oct. 2007.