Selecting an Ethernet Failure Recovery Method for Protection and High-Speed Automation Applications

David Dolezilek, Colin Gordon, Dwight Anderson,
Timothy Tibbals, and David Keckalo
Schweitzer Engineering Laboratories, Inc.

Presented at the
PAC World Africa Conference
Johannesburg, South Africa
November 12–13, 2015

Selecting an Ethernet Failure Recovery Method for Protection and High-Speed Automation Applications

David Dolezilek, Colin Gordon, Dwight Anderson, Timothy Tibbals,
and David Keckalo, Schweitzer Engineering Laboratories, Inc.

Email: papers@selinc.com

USA

1 Introduction

This paper, an expanded version of [1], compares the mechanisms for quickly reconfiguring Ethernet networks to satisfy mission-critical performance metrics while providing simultaneous duplicate or redundant signal exchange. This paper also compares the attributes and complexities of duplication methods such as Parallel Redundancy Protocol (PRP) and High-Availability Seamless Redundancy (HSR) that were designed for industrial systems where personnel are available to repair failed systems. Finally, this paper describes the redundancy and duplication methods that are available in software-defined networking (SDN) with illustration of how they create lossless signal delivery with the use of Ethernet packets and address issues of teleprotection, interlocking, and automation applications based on IEC 61850 Generic Object-Oriented Substation Event (GOOSE) messaging. The conclusion summarizes the ability for each method to satisfy selection criteria for networks to satisfy mission-critical protection and automation applications.

IEC 61850-based mission-critical protection signaling applications exchange signal statuses, controls, alarms, and measurements as information within GOOSE messages.

Although communications cables can be directly connected between intelligent electronic devices (IEDs), this does not provide redundancy or reconfiguration. Through use of Ethernet technology, the messages between IEDs pass through a shared bandwidth Ethernet network. This network is made of devices in the middle of the data path between IEDs, called middleboxes. Other designs use intelligent middleboxes such as communications processors and information processors that understand the message contents. This paper discusses methods to use Ethernet middleboxes that do not understand the message contents. It also introduces software defined networking (SDN) that can understand and manipulate parts of the messages.

1.1 Relay Self-Tests Detect Network Anomalies

Relay self-tests determine the health of firmware execution and communications port functions. Other diagnostics provide status of communications interfaces and out-of-range warnings for input signals and dc power. Self-test of the receipt of protection signals via digital messages, also known as message quality, is performed by supervising the correct and constant receipt of the digital messages. Immediate detection of failed self-test of message receipt is used to modify logic to work correctly in the absence of signals. Error codes for exchange of IEC 61850 GOOSE messages via Ethernet packets include *out of sequence* when one or more packets are not delivered and the sequence number of the next received packet is not consecutive [2]. Another error code calculated by the subscriber is titled *time to live expired*. This code results when the time between messages is too long, the lifespan of the previous message is expired, and the contents of the previous message are no longer considered current. When the receiver detects that the signal exchange via GOOSE messages is not working, it sets a logic bit associated with the quality of that message subscription to the failed state (a value of one). The subscribing IED must immediately detect the failure of message quality to correctly perform communications-assisted protection by adapting the logic after failure. Fig. 1 illustrates transformer relay logic qualifying signals from the feeder relay used for breaker failure.

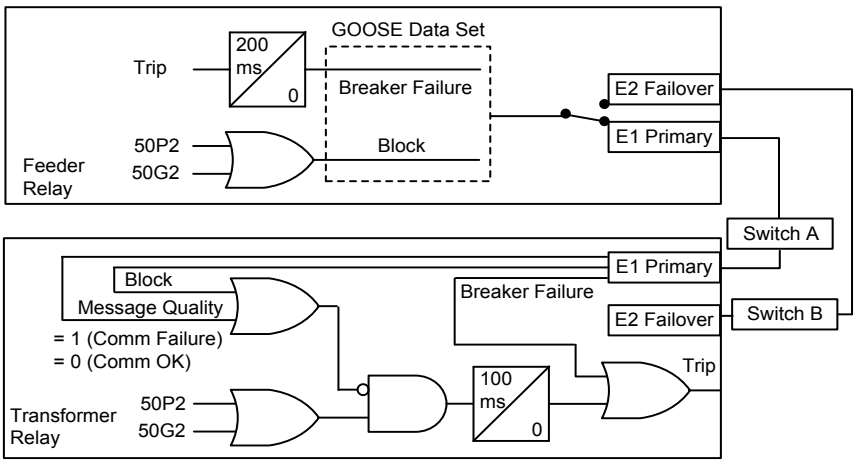

Fig. 1. Use of GOOSE Message Quality Status in IED Logic

1.2 Redundancy vs. Duplication

Redundancy to improve the availability of mission-critical applications is defined as the inclusion of extra always-active components and capabilities—not strictly necessary for the application—to provide functionality in the event of failure within the primary system. By this definition, redundancy methods overcome a failure within the primary system. Duplication methods overcome a failure by providing a simultaneous second virtual or physical system. However, duplication methods alone without recovery mechanisms only provide duplication until there is a failure. Two simultaneous failures within systems relying on duplication methods may disable communication to all or part of the two systems. Therefore, duplication alone is not an appropriate fault compensation technique and needs to be used in conjunction with, or replaced by, methods that quickly reconfigure after failure.

Immediately after an IED senses a signal change-of-state triggering event, it publishes that change of signal information in a GOOSE message. To increase the dependability of teleprotection, interlocking, and automation, IEC 61850 describes a process by which IEDs make the new signal information redundant by publishing several consecutive redundant, rather than duplicate, GOOSE messages in a burst after the initial change-of-state message.

This burst provides redundant signal information via the multiple messages. Even if a network component fails immediately before publication of the initial change-of-state GOOSE message, lossless signal transfer is attained with fast and efficient processing of the spanning tree algorithm (STA) and well-engineered Ethernet network topologies [2]. In each network, one middlebox, called the root bridge, is in charge of coordinating the distributed STAs running within each middlebox. The middlebox designated to take over in the event of failure of the root bridge is referred to as the backup root. Should that middlebox fail, the others will negotiate a new root bridge. With the ladder topology and fast STAs, network failures are detected and corrected quickly enough that one or more of the redundant GOOSE messages reaches each subscriber.

Fig. 2(a) illustrates a signal exchange in which the first physical device (PD1) senses a change via a contact input and communicates the signal status change with a message published in an Ethernet packet. The Ethernet packet is sent through the network to the second physical device (PD2), which trips an output contact.

Fig. 2(b) illustrates the misconception that a second Ethernet cable connected to an IED provides a redundant connection. To avoid data flow loops, IEEE 802.1D Ethernet mechanisms disable any second connection to an IED and force the connection into failover or hot-standby mode. This is a duplicate connection, not a redundant connection, because the failover link is not active simultaneously with the primary link. The failover link only becomes active to publish GOOSE messages or any other Ethernet packets after the primary connection has failed. When the network failure is a link directly connected to the IED, it detects the failure and communications failover to the other port. However, the protection application is unaware that a failover occurred.

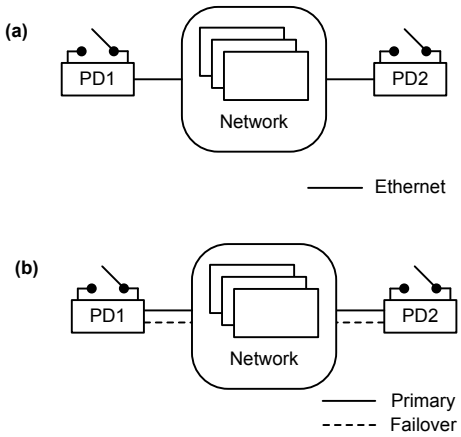

Fig. 2. System Exchanging Protection Signals in Ethernet Packets Through a Network (a) and Duplicate Ethernet Cable Serving as Hot Standby (b)

1.3 Timing Requirements for Digital Message Transport Across an Ethernet Network

In this paper we assume a communications-assisted protection scheme such as a transfer trip example requiring signal exchange via digital message transfer that takes no longer than 20 ms. From various international standards [3], we know that control blocking schemes require a 99.99 percent success rate and that direct control schemes require a 99.9999 percent success rate of receipt of digital messages, as per IEC 60834-1. IEC 61850-5 defines fast messages that meet the 3-ms transmission time as Type 1A, Performance Class P2/P3, further described in [3]. Failure to accomplish the 20-ms transfer time, which includes the subscribing IED processing the message, is defined as a delay in delivery greater than 18 ms when transferring a data signal to an IED with a 2-ms operating cycle. Therefore, the digital communications system must meet the 3-ms transmission time 99.99 percent of the time and have a delay no longer than 18 ms for the remainder. Network reconfiguration around a path failure must be fast enough to satisfy the 18-ms maximum signal transfer during communications system failure and recovery. Subtracting the 3-ms subscribing IED packet processing time from this 18-ms maximum duration leaves 15 ms for the network to reconfigure after any failure. Should the network reconfigure within 15 ms, it can deliver the Type 1A, Performance Class P2/P3 message. This delivery takes as long as 3 ms, for a total of 18 ms. Add to this the maximum time to process the signal within protection logic of 2 ms and we meet the maximum signal transfer time of 20 ms. If the reconfiguration time of 15 ms cannot be satisfied with the chosen switch network, redundant networks and redundancy protocols are necessary.

Link failure detection, isolation, and reconfiguration is traditionally performed by an STA based on information received within Rapid Spanning Tree Protocol (RSTP) messages published by network devices on the shared Ethernet network. When the STA reconfigures the Ethernet network failure faster than 15 ms, all or most of the redundant GOOSE packets are delivered for the satisfaction of protection signal applications. Network devices among IEDs (referred to as switches or middleboxes) with slower STAs cannot be relied on for mission-critical signaling. Some middlebox manufacturers offer proprietary solutions, but in systems including devices from multiple manufacturers, proprietary solutions are not useful because they are neither standardized nor interoperable. As a result, when middleboxes rely on slowly resolved STAs, manufacturers recommend purchasing and building two duplicate STA/RSTP Ethernet networks in the hopes that while one network fails to deliver packets, the duplicate network will not fail simultaneously. Simultaneous failure is statistically unlikely, but still possible, so it is important to use network topologies that are more resilient than a simple ring network architecture [2].

2 Communications-Assisted Applications Via Digital Messages

2.1 Signaling Requirements for Teleprotection, Interlocking, and Automation Applications

The hard-wired exchange of protection information uses an analog value at the receiver to indicate the logical status of the signal from the sender. Typically, an analog value of zero indicates a status value of zero and the maximum analog value represents a status value of one. This method creates a constant signal value at the receiver. However, if the signal wire is cut or disconnected, the receiving device cannot distinguish between this failure or a legitimate zero analog value. With digital messages, the signal exchange is not constant. Each time a digital message is received, the signal

status is confirmed or a change of status is recognized. The receiver assumes that the signal status remains unchanged during the time between messages. Also, the digital message exchange can be supervised. In this case, the receiver can detect when the communications link is lost. For example, MIRRORED BITS® communications publish digital messages every 2 ms over dedicated links, so signal confirmation of a change of state is detected within 2 ms at the receiver.

Non-change-of-state GOOSE messages typically occur once per second and act as an application heartbeat, with the time between signal confirmations at the receiver growing to once per second. A signal status change of state typically triggers an immediate GOOSE publication. The IEC 61850 standard does not specify how quickly an IED must process and act on the signal information within the GOOSE message. The application designer must understand and procure appropriately designed IEDs that streamline and prioritize processing of signal information receipt via GOOSE. As with MIRRORED BITS, these subscriber IEDs receive and react to the change-of-state information within 2 ms, assuming the network performs correctly and delivers at least one of the GOOSE messages within the burst. To provide lossless signal transfer in the event of a network failure, IEC 61850 describes a method by which the source IED publishes several redundant change-of-state indications in a burst of redundant messages after the change of state occurs. When the failure is corrected before the burst is complete, lossless signal delivery is achieved.

Communications-assisted protection via GOOSE relies on fast signal information message publication and subscription as well as fast network reconfiguration.

GOOSE heartbeat messages are typically published no more rapidly than once per second to reduce traffic on the shared-bandwidth Ethernet network. This means that the time between confirmation is much longer, and the detection of failed communication is much longer than for MIRRORED BITS communications. In traditional switched Ethernet networks, increasing heartbeat message frequency shortens the time necessary to detect failure to receive GOOSE messages but produces a lot of additional Ethernet traffic. This additional Ethernet traffic may lead to oversubscription and saturation of various network links, creating a burden on both network middleboxes and the network interface of poorly designed IEDs. IEDs should be chosen that are designed to streamline message processing to avoid network processing problems, and the network should be designed so that data traffic is appropriate and not unnecessary. Traditional middleboxes lack the capability to detect or alarm for oversubscription and saturation problems, so these network problems often exist but go undetected. Alternatively, a direct serial or Ethernet connection between relays avoids the challenges of the shared purpose and bandwidth of the Ethernet network and performs more reliably. On a direct Ethernet link, GOOSE packets can be configured for publication every few milliseconds rather than once per second, improving both application reliability and subscription failure detection.

The IEC 61850 standard refers to the use of unique identifiers and virtual local-area networks (VLANs) based on IEEE 802.1Q, referred to as QVLANs, to be configured within GOOSE and Sampled Value messages. These message QVLANs, illustrated as tags in Fig. 3(a), are reviewed by the middlebox port connected to the GOOSE publisher, as shown in Fig. 3(b), to determine if the packet should be allowed to enter the network. Internet Protocol (IP) messages have no value in the message tag field and are referred to as untagged. When untagged messages enter a middlebox, they inherit the port-based VLANs (PVLANs) configured on the middlebox port. The middlebox uses both the PVLAN tag and the QVLAN tag identifiers to determine which port the packet is allowed to exit.

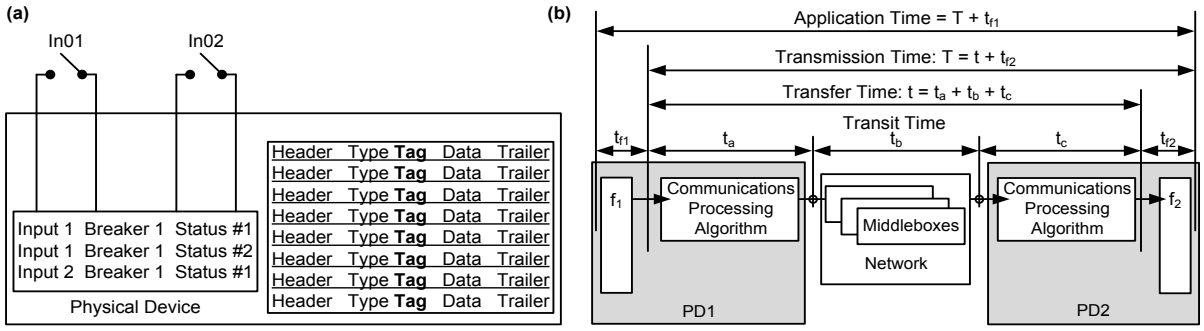

Fig. 3. IED Contact Input Information Mapped as Payload Data in GOOSE Messages (a) and Application, Transmission, Transfer, and Transit Time Based on IEC 61850-5 (b)

IEC 60834-1 requirements for security, reliability, and dependability are met for a communications-assisted protection scheme if the system meets the 3-ms transfer time 99.9999 percent of the time and has a delay no longer than 18 ms for the remainder [2]. The IEC/TR 61850-90-4 network engineering guidelines [4] for IEC 61850 Ethernet traffic suggest that GOOSE messages used for protection should be designed to have the highest priority and the shortest maximum delay.

Control blocking schemes, via GOOSE messaging or any other method, require a 99.99 percent success rate, and direct control schemes require a 99.9999 percent success rate of the receipt of digital messages (reliability). Direct tripping through delivery and processing of a GOOSE or other message is typically expected to occur within a transmission time of 20 ms [2]. Failure is defined by the absence of the message at the receiving end or, for direct control, a delay in delivery longer than 18 ms. Therefore, the use of Class P2/P3 messages requires that the system be designed to meet the 3-ms transfer time. This requires high device reliability to minimize path failures.

System availability analysis based on IEC 61850-5 measures of reliability is used to predict the ability of each system to meet IEC 60834-1 dependability and security requirements [2]. For example, a GOOSE application configured to publish confirmation messages every second publishes 86,400 messages every 24 hours. Applying the IEC 60834-1 standard to a GOOSE signal exchange for direct tripping with a 1-second heartbeat requires that the network deliver every single GOOSE message packet, no exceptions, (dependability) and deliver fewer than nine unwanted (spurious) or delayed GOOSE message packets (security) during every 24-hour period [5].

System availability is simply stated as the ratio of up time (when it is capable of operation) to down time (when it fails to operate). For reconfigurable designs such as STA, this ratio is very high and not dependent on repairs being made. STA and SDN technology employ mechanisms to detect the faulted component, isolate it, and recover service by reconfiguring data paths automatically. When necessary, these methods activate previously inactive, hot standby redundant links to become active redundant data paths. Based on the Ethernet network topology, several component failures can exist simultaneously and the system will still recover by reconfiguring data paths automatically. Active techniques such as STA and SDN use fault detection, fault location, and fault recovery to achieve fault tolerance.

Redundancy is necessary, but not sufficient for security [6]. One objective of unique GOOSE message identifiers and encoding of the information within a payload is to make each of the distinct messages as different as possible from the rest. The quantitative measure of this difference is the Hamming distance. Specifically, Hamming's formulas allow algorithms to detect and correct errors on their own. For messages, the Hamming distance is defined as the minimum number of bits that could be corrupted in one distinct message, which would result in a different distinct valid message. A different valid message represents failure of the system, and the Hamming distance represents the minimum number of failed elements that cause the application to stop functioning.

When considering communications networks, the Hamming distance is defined relative to system components [7]. The Hamming distance of redundancy and duplication mechanisms is the minimum number of devices that can fail to change a fully operational system(s) into one that fails to perform. For communications-assisted logic schemes, there are three types of component failures to be considered when contrasting the Hamming distance among multiple network choices. These include failure of the source IED(s) or cable(s), network components, and the destination IED(s) or cable(s). As with other quality measures, the system Hamming distance is the sum of components that must fail simultaneously and the product of components that cause failure individually.

The purpose of this paper is to contrast the resiliency of the network, and not the IED connection to the network. Therefore, Hamming distance of the network is considered to be the minimum number of network components, not directly connected to an IED, that must fail.

2.2 Redundant Information, Messages, Packets, and Applications

The redundancy method with the highest availability as a percentage of expected run time involves creating two complete and separate systems, isolated from one another and acting as dual primary redundant applications, as shown in Fig. 4(a). This provides complete application redundancy as well as redundant information, messages, and packets. PD2A logic uses locally calculated message quality to confirm successful receipt from PD1A, and PD2B logic uses locally calculated message quality to confirm successful receipt from PD1B. PD2A and PD2B then share these statuses so that their respective protection algorithms are aware if the dual primary fails.

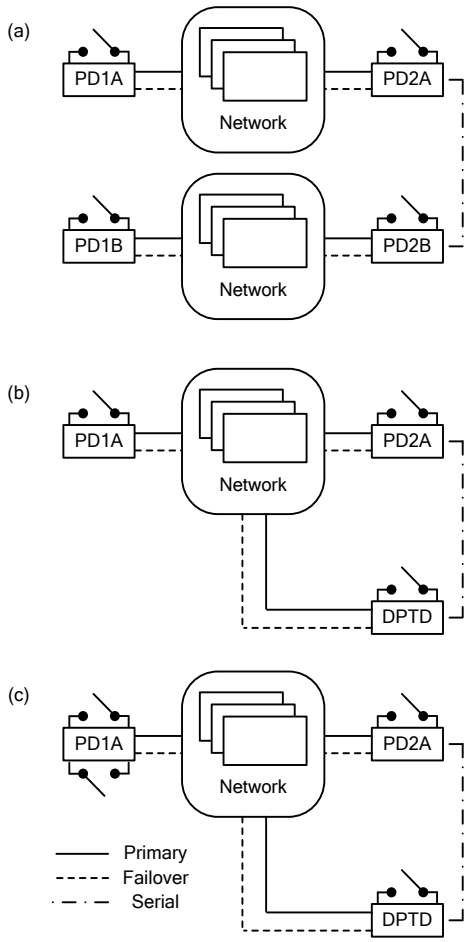

Fig. 4. Complete Dual Primary Signaling Systems (a) Application of DPTD (b) and Dual-Input Contacts (c)

Consider the case of sending a GOOSE message from PD1A to PD2A and PD1B to PD2B in Fig. 4(a). Hamming distance for this example using STA and a 10-switch ladder network topology, as shown in Fig. 5(a), is as follows.

Source IED	2 IEDs or 4 cables
Network component	16 middleboxes or 26 cables
Destination IED	2 IEDs or 4 cables

In the event that completely redundant systems are not possible, adding a dual primary trip device (DPTD) makes the mitigation application redundant via the redundant trip function, as shown in Fig. 4(b).

Hamming distance for this example in Fig. 4(b) using a 10-switch STA ladder network topology is as follows.

Source IED	1 IED or 2 cables
Network component	8 middleboxes or 13 cables
Destination IED	2 IEDs or 4 cables

The information can be singular, or it can be redundant with a second contact input, as shown in Fig. 3(a). The message can be singular, or it can be redundant if a second message is configured into the source device. According to IEEE 802.1, the packets are duplicated and rebroadcast within a middlebox when the same message is published to both PD2A and the DPTD. Therefore, additional redundancy is provided when two GOOSE messages are configured, one for PD2A and a redundant one for the DPTD.

Dual-input contacts do not influence the effect of source IED, network, or destination IED component failure. Therefore the Hamming distance for this system, shown in Fig. 4(c), is the same as that of the previous example for the system illustrated in Fig. 4(b).

Source IED	1 IED or 2 cables
Network component	8 middleboxes or 13 cables
Destination IED	2 IEDs or 4 cables

However, these dual-input contacts from different source field contacts double the Hamming distance of a substation wiring failure.

PD2A and DPTD in both Fig. 4(b) and Fig. 4(c) calculate message quality to confirm successful receipt from PD1A. DPTD sends this status to PD2A so that its protection logic can determine the status of the dual primary trip signal exchange.

2.3 Traditional Redundancy Protocols and Duplication Methods

Rather than improve the STA and network topology, many IED manufacturers have chosen to promote a variety of message duplication techniques that require a substantial amount of additional Ethernet hardware. This effort is an attempt to provide lossless signal exchange during failure and recovery of links and Ethernet middlebox hardware [5]. It is also largely the result of a common perception that middleboxes are inexpensive and subject to poor mean time between failure (MTBF) numbers. In addition to the process of redundant GOOSE publications after a change of state, four methods are currently available to quickly overcome failures or create duplicate or redundant messages:

- IEEE 802.1D STAs
- IEEE 802.1Q message redundancy
- IEC 62439-3 message duplication protocols—HSR and PRP
- OpenFlow 1.3 software-defined networking

Software-defined network (SDN) technology is a recently available technology that offers substation-hardened components and has the potential to dramatically improve digital messaging.

2.3.1 Fast STAs

Traditionally, STAs in each middlebox share data through the use of RSTP (particularly according to IEEE 802.1D-2004), which publishes Bridge Protocol Data Unit packets that communicate middlebox statuses to overcome failures. Using information from other middleboxes, each STA identifies and deactivates duplicate paths to a network address. Duplicate paths are forced into hot-standby mode and can be reactivated to reconfigure the network after a failure. The speed with which typical middleboxes resolve STAs is inadequate for protection and high-speed automation. Testing of a four-node ring of often-used middleboxes reveals periods of failed communications that grow from hundreds of milliseconds to tens of seconds during a failure recovery scenario. Without testing, most end users are unaware that their ring networks perform this poorly. This testing is necessary and required for systems as part of the IEC/TR 61850-90-4 network engineering guidelines.

Specific research and development have led to the use of ladder topology, as shown in Fig. 5(a), as the Best Known Method (BKM) to enable fast reconfiguration times [5]. During reconfiguration after a failed middlebox or link, the ladder topology with an interoperable and fast STA will quickly detect and isolate faults and then recover data communication. Reconfiguration after failure of any middlebox or cable is performed in under 15 ms. Failure of a primary switch around which the STA network topology is determined (root bridge) causes more disruption, so reconfiguration after most, but not all of these failures is performed within 15 ms as well. A small percentage of hundreds of test cases for root bridge failure did result in recovery slightly longer than 15 ms. However, any ring topology larger than three middleboxes with traditional STAs will not reconfigure any faster than 15 ms and usually takes much longer. The ladder topology with a fast STA is less expensive, simpler, and more easily understood than other methods. This topology also provides much more resiliency than other methods. For example, the devices attached to middlebox switches B3 and B5 in the ladder network in Fig. 5(a) are proven to have lossless signal exchange even if the network experiences as many as seven middlebox switches and twelve link failures [2].

As previously stated, the relay will only identify a failover because of a failed connection to the primary port on the relay itself. When STA resolves a failure within the network, the relay logic is not aware that a failure has occurred. However, correctly designed networks with fast reconfiguration correct any failures quickly enough that protection logic is unaffected.

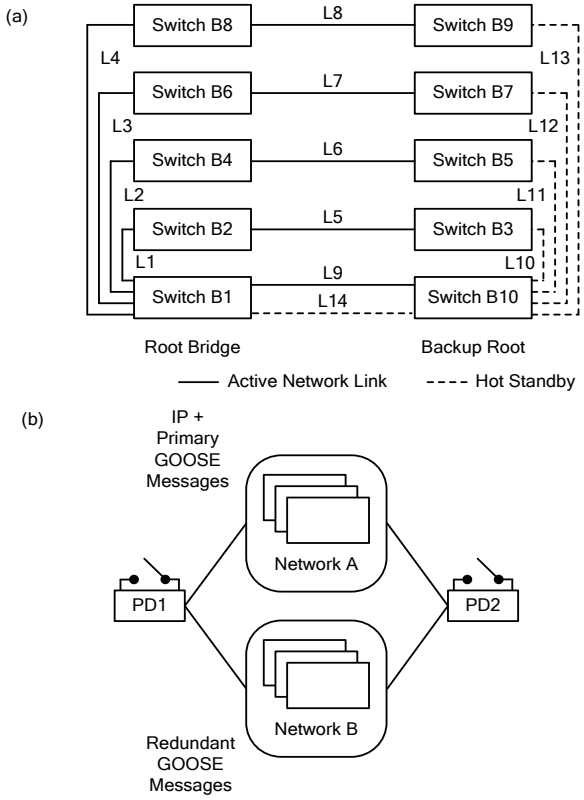

Fig. 5. Resilient Single Primary Ladder Network Topology (a) and Dual Primary Signaling Via Two Systems Segregated by Way of QVLANS (b)

2.3.2 IEEE 802.1Q VLAN Methods

Through use of integrated ingress filtering based on QVLANs for redundancy, GOOSE messages are sent into two different networks connected to two different ports on the same device, as shown in Fig. 5(b). Both IED ports are active and connected to a different network, and IEEE 802.1Q filtering prevents data flow loops. The primary network is configured to allow all untagged IP traffic and GOOSE message packets tagged with QVLAN identifiers allocated to the primary LAN. This creates a fully functional primary network with GOOSE message burst redundancy.

The second IED Ethernet port is connected to a redundant protection LAN that is configured to allow only tagged traffic with QVLAN identifiers designated for use on the protection LAN, as shown in Fig. 5(b). These QVLAN tag identifiers must each be unique from tags allocated to the primary LAN, and the middlebox ports are also configured to deny Network A IP traffic. Using this method, the IED calculates message quality independently for both of the redundant GOOSE messages. This method can provide redundant signals, packets, and messages and can inform the protection logic if one of the two signal paths fails.

Again, consider the case of sending a GOOSE message from a device connected to switch B3 to a device connected to switch B5 in Fig. 5(a). Hamming distance for this example using STA and a 10-switch ladder network topology is as follows.

Source IED	1 IED or 2 cables
Network component	16 middleboxes or 26 cables
Destination IED	1 IED or 2 cables

This redundant protection network is simpler than the primary network because it only connects to the second Ethernet port of IEDs that are sharing signals. The information can be singular, or it can be redundant with a second contact input. The messages and packets are redundant. The QVLAN method is the simplest, least expensive, and most effective way to produce redundancy through the network. However, this method does increase the engineering necessary in the IEDs. Although it is trivial to configure publication of two GOOSE messages instead on one, writing logic to subscribe to two does add complexity to the destination IED. The subscribing IED must be programmed to monitor the message quality of both incoming GOOSE messages and also perform a logical OR function using the signal status from both messages to drive the protection logic. This method of publishing two unique GOOSE messages is the only fully redundant method using single IEDs and is the only method that is capable of informing the protection logic that one or both of the two data paths has failed. Virtually all modern STA-based Ethernet middlebox hardware supports the well-established IEEE 802.1Q VLAN segregation method. However, this method does require extra IED configuration to create two GOOSE messages instead of one.

Single or redundant networks are enhanced to provide additional redundancy by way of dual primary contact inputs and DPTDs, as illustrated in Fig. 6(a) and Fig. 6(b), respectively. These can be isolated via multiple methods, including QVLANs as in Fig. 5(b).

Hamming distance for the example shown in Fig. 6(a) using STA in a 10-switch ladder network topology is as follows.

Source IED	1 IED or 2 cables
Network component	16 middleboxes or 26 cables
Destination IED	1 IED or 2 cables

Hamming distance for the example shown in Fig. 6(b) using STA in a 10-switch ladder network topology is as follows.

Source IED	1 IED or 2 cables
Network component	16 middleboxes or 26 cables
Destination IED	2 IEDs or 4 cables

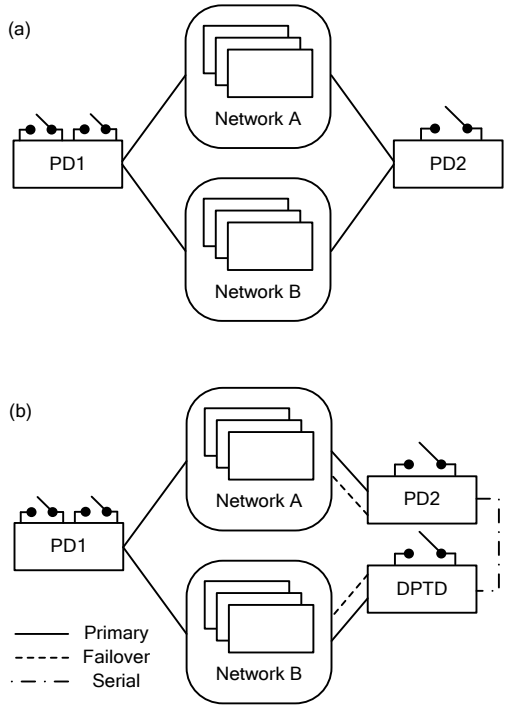

Fig. 6. Redundant Information Via Dual Primary Contact Input (a) and Redundant Application Via DPTD (b)

2.3.3 IEC 62439 Fault Masking and Packet Duplication Methods

The availability of the mechanisms described in IEC 62439 [8] relies on external error detection and repair. As described in the standard, these technologies evolved as repairable, not reconfigurable, solutions for industrial systems. Typical industrial systems, where operations and maintenance staff are present, are quite different than unmanned substations and distribution grids. The IEC 62439 technologies, including parallel redundancy protocol (PRP) and high-availability seamless redundancy (HSR), boast zero recovery time, which is misleading because they do not attempt to perform recovery. The availability of these repairable systems is directly proportional to the downtime that includes both the time to detect failure and time to repair or replace a failed component. As defined in IEC 62439, PRP and HSR are static techniques that perform fault masking. These techniques are designed to achieve limited fault tolerance without requiring any action on the part of the components within the system. The fault is unknown to the application, and the systems are only tolerant of a single fault until they are repaired [7].

2.3.4 Duplicate Ethernet Packets Via Parallel Redundancy Protocol (PRP)

IEC 62439 describes seven methods to compensate for failure of industrial communications networks within automation networks. Per Clause 4 of IEC 62439-3, PRP is an industrial Ethernet packet duplication method that performs no reconfiguration or failure recovery. This method creates redundant Ethernet packets, but it provides no redundant protection applications, signaling information, or GOOSE messages. IEDs participating in PRP make duplicate copies of each Ethernet packet and publish them out two different ports. These packets are identical in every detail and therefore are duplicates and not redundant. When an IED is producing GOOSE messages, these messages are duplicated like all other packets. The GOOSE message signal data and parameters including time and sequence and state numbers are identical and duplicated as part of the packet duplication. Therefore, even though the packets are made redundant, the GOOSE message, signal information, and protection application are not made redundant.

PRP delivers duplicate packets through two isolated networks. When one network fails, the fault is masked, or hidden, if a packet arrives via the other network. Instead of using QVLANs, PRP can be used to isolate the two networks shown in Fig. 6(a).

Hamming distance for the example shown in Fig. 6(a) using PRP and a 10-middlebox ring topology is as follows.

Source IED	1 IED or 2 cables
Network component	2 middleboxes or 2 cables
Destination IED	1 IED or 2 cables

Hamming distance for the example shown in Fig. 6(a) using PRP, STA, and a 10-middlebox ladder network topology is as follows.

Source IED	1 IED or 2 cables
Network component	16 middleboxes or 26 cables
Destination IED	1 IED or 2 cables

PRP enables source devices to create duplicate packets and publish them out of two different Ethernet ports. PRP uses a redundancy control trailer (RCT) tag added to the end of an Ethernet frame payload to identify outgoing packets and keep track of incoming packets. The RCT sequence counter is used to identify the sequential order of the message, and the line identifier is used to determine whether the message is destined for Network A or Network B.

The receiver uses the PRP RCT tag to identify the packet as belonging to Network A or Network B. If one network fails and the packet arrives via the other network, this masks (or hides) the fault. When both networks are active, devices process one of the duplicate messages they receive—usually the first one—and discard the other. This creates additional publication and subscription processing and the need for a second duplicate Ethernet network. A PRP Redundancy Box (RedBox) receives packets from non-PRP devices, tags them, and publishes them into Networks A and B. It also receives the duplicate messages, strips the PRP tags, and forwards the messages to the non-PRP devices. The PRP method is unable to inform the protection logic if any paths fail. The use of PRP RedBoxes interferes with the Hamming distance comparison, because they become a true single point of failure.

Hamming distance for the example shown in Fig. 7 using PRP, STA, and a ladder network topology is as follows.

Source IED	1 IED or 1 cable
Network component	1 RedBox or 2 cables
Destination IED	1 IED or 2 cables

Hamming distance for the example shown in Fig. 8 using PRP, STA, and a ladder network topology is as follows.

Source IED	1 IED or 2 cables
Network component	1 RedBox or 2 cables
Destination IED	1 IED or 2 cables

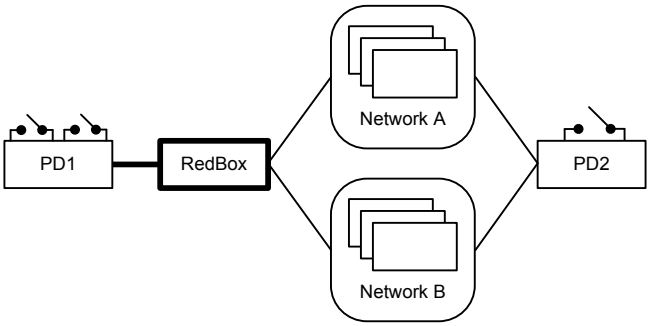

Fig. 7. Single Cable, Singly Attached Node

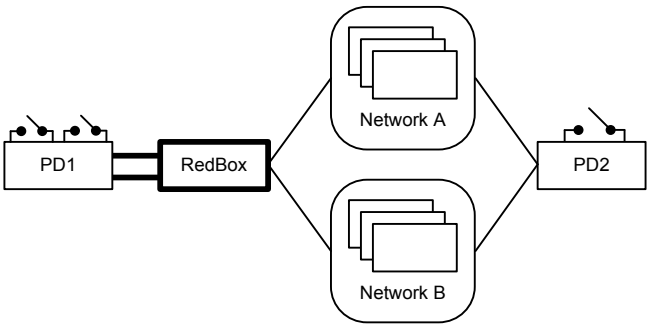

Fig. 8. Primary/Failover Cables, Singly Attached Node

PRP methods deliver a single duplicate packet in the event of any failure within either Network A or B, but not both. PRP includes no recovery mechanisms, so packet redundancy is disabled while a failure exists. If the same failure, such as damage to a cable tray or conduit, affects both Networks A and B, signaling fails completely altogether and no recovery occurs.

Again, PRP does not provide redundant protection applications, signaling information, or GOOSE messages. PRP only provides duplicate Ethernet packet delivery until the first network failure and therefore is often combined with traditional STA recovery methods. However, because two separate STA networks separated by VLANs instead of PRP provide actual redundancy instead of duplication, once two separate networks are designed, PRP adds complexity but no value and is unnecessary.

2.3.5 Duplicate Ethernet Packets Via High-Availability Seamless Redundancy (HSR)

Per Clause 5 of IEC 62439-3, HSR is an industrial Ethernet packet duplication method created for use with IEC 61784 protocols that performs no reconfiguration or failure recovery. As with PRP, the HSR method creates redundant packets, which are no longer Ethernet, and provides no redundant applications, information, or messages. This method duplicates and delivers each IED message in two directions around a single ring network, as shown in Fig. 9(a).

When one direction fails, the fault is masked, or hidden, if a packet arrives via the other direction.

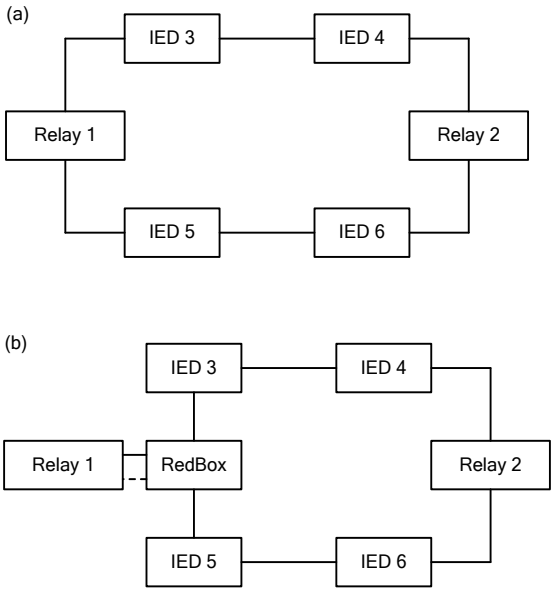

Fig. 9. Redundant Packets Within Single HSR Network Ring (a) and Non-HSR-Enabled Relay Connected Via RedBox (b)

HSR-enabled source devices create duplicate packets and publish them out of two different Ethernet ports. Also, because HSR uses a tag added to the beginning of the message, it is no longer a traditional Ethernet message and cannot travel through traditional Ethernet switches. The HSR tag aids in identification of outgoing packets and tracking of incoming packets, and each device processes one of the duplicate messages it receives—usually the first one—and discards the other. This creates additional publication and subscription processing in each IED, slowing the application and delivery because each packet must be processed and passed through each IED. This method delivers a single duplicate packet in the event of any single failure within the ring. However, in the event of a second network failure, redundant messages and all communications will fail to reach the isolated segment.

Hamming distance for the example shown in Fig. 9(a) using a HSR network topology is as follows.

Source IED	1 IED or 2 cables
Network component	2 devices or 2 cables
Destination IED	1 IED or 2 cables

The forwarding delay of every device in a HSR ring adds to the total network latency, so special hardware is often required to reduce the per-hop latency to a reasonable value. These device requirements can make HSR very specialized, expensive, and cause additional packet latency. Also, by duplicating every message onto a single network, HSR reduces the effective network bandwidth and available device processing by 50 percent, even when there is no active failure condition. As with PRP, non-HSR devices can be added through connections to a RedBox, which converts non-HSR packets into HSR and vice versa, as illustrated in Fig. 9(b).The use of HSR RedBoxes interferes with the Hamming distance comparison, because they become a true single point of failure.

Hamming distance for the example shown in Fig. 9(b) using a HSR network topology is as follows.

Source IED	1 IED or 2 cables
Network component	1 RedBox or 2 cables
Destination IED	1 IED or 2 cables

HSR negatively affects signal exchange because each device in the ring adds signal delivery latency, and every device must process and pass on each message in the network. The most difficult challenge is that, because there is no recovery method when a fault occurs on the ring, the network link is masked and remains broken until someone performs repair. The HSR method cannot inform the protection logic if a path fails.

Both HSR and PRP can be connected together using RedBox interconnections. However, no combination provides redundant applications, information, or messages.

2.4 SDN-Based Redundancy Methods

2.4.1 Introducing Software-Defined Networking (SDN)

SDN essentially allows management of networks as a single asset, giving network operators extremely granular levels of control over network functionality while simultaneously abstracting the complexity into a more traditional and functional programmatic interface [9]. The effects of the abstraction and granular control are the simplification of network operation; the ability for continuous monitoring in more detail; and holistic, centralized network control over the programming of individual middleboxes.

The fundamental shift in networking brought by SDN is the decoupling of the systems that decide where the traffic is sent (i.e., the control plane) from the systems that perform the forwarding of the traffic in the network (i.e., the data plane). For traditional networks, packets must flow on links determined by the various distributed STAs in the middleboxes supported by information published in RSTP packets. Middleboxes must contain additional traffic control methods, including IEEE 802.1Q VLANs and IEEE 802.1p Class of Service tags, to maximize reliability. In large networks, trying to match the STA-discovered path with an application-desired data path may involve changing configurations in hundreds of devices with a variety of features and configuration parameters. This complexity in management arises from the fact that each middlebox internally integrates a combination of control logic and data-forwarding logic. Fig. 10 illustrates the SDN building blocks, which provide logical separation of creating and executing the data flow rules [10].

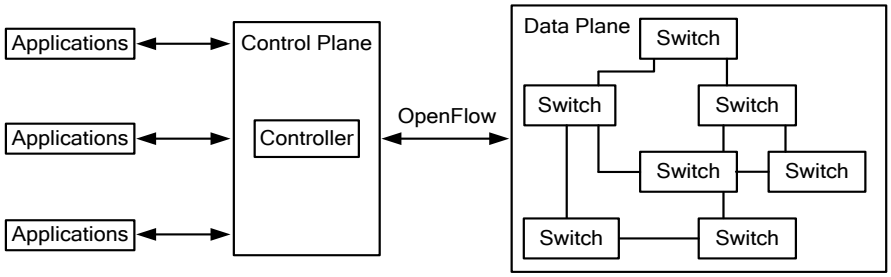

Fig. 10. SDN Architecture Overview

The SDN controller understands the entire network and sends rules to the SDN middleboxes, which are programmed simply to execute the flow rules as directed by the controller. SDN reduces or eliminates the need to use spare, inactive links to IEDs or other middleboxes. By defining the behavior of network paths, SDN allows otherwise inactive Ethernet paths to be actively used simultaneously. Therefore, all Ethernet connections to the IEDs and middleboxes can be used as designed and none are forced to hot standby mode. With SDN, the middleboxes no longer need to send RSTP messages or run STAs and can make fast and deterministic decisions about where to send packets based on logical paths through the network, as computed by the controller. Without STAs, the network can manage more than one active connection to each end device. When two cables are connected to an IED, they can both function simultaneously and pass different or redundant messages on each link. As with STA, the relay will only identify a failover because of a failed connection to the primary port on the relay itself. However, using SDN the network can be designed to deliver two different GOOSE messages from the source relay to the destination relay by using different paths in the same network. Using this method, the IED calculates message quality independently for both of the redundant GOOSE messages. As with the redundant physical LAN example shown in Fig. 4(b), this method provides redundant signals, packets, and messages and informs the protection logic if one or both of the two signal paths has failed. Also, SDN enables us to create redundant data paths in the same network.

2.4.2 Introducing OpenFlow™

SDN controls an entire network as a single operating environment and allows administrators to implement and operate the network in a way that is targeted directly at the data exchange needs of each application. SDN gives operators high-level control over individual message types by separating the control plane (or brain) that decides traffic flow from the data plane (or hardware) that pushes packets. OpenFlow is an increasingly popular protocol that acts as the interface between the SDN

brain and the packet-pushing hardware. OpenFlow was developed by the Open Networking Foundation (ONF) to fulfill the need for a communications interface for SDN [11]. OpenFlow enables one or more OpenFlow controllers to define the path of packets through an Ethernet network by manipulating the packet flow tables of OpenFlow-enabled hardware. When using STA, each Ethernet middlebox is responsible for its own data flow decisions. This leads to complications during dynamic conditions because of all the simultaneous independent decision-making. SDN (when implemented with OpenFlow) solves this problem because the SDN controller precomputes all states of the network system, including those with host, link, and middlebox failures, and implementation occurs in OpenFlow-enabled middleboxes, eliminating the need for packet-pushing hardware to discover the state of the system from neighboring middleboxes.

OpenFlow provides three functions that help the controller make decisions on packets that enter an Ethernet middlebox: matches, actions, and counters (or statistics). Combinations of matches and actions form flow tables, while counters (collected by the controller via polls) detail information such as byte counts, flow table matches, and more.

- OpenFlow matches: When an Ethernet frame first enters the middlebox hardware, the OpenFlow-enabled middlebox examines the packet in search of match fields. Match fields can be physical and can include physical ingress ports and packet headers found in OSI layers 2-4.
- OpenFlow actions: When a match is found for a particular flow table, actions can be performed. Actions may include traditional packet-handling methods, such as forwarding or dropping. OpenFlow also allows copying of frames to multiple end ports, the application of metering or rate-limiting functions, and direct manipulation of packet headers.
- OpenFlow counters: OpenFlow-enabled middleboxes keep counters for every port, flow, flow table, and other logical and physical ports and actions performed. Examples of counter data can be the number of dropped frames, total byte count for a flow table, and so on.

OpenFlow can define the logical or physical data path for an Ethernet frame based on any matched packet field. OpenFlow can detect teleprotection messages by their Ethertype and treat them as critical traffic with unique data paths.

2.4.3 OpenFlow Controller Interaction

OpenFlow controllers interact with OpenFlow middleboxes in two ways. The first method of interaction is called reactive flow instantiation, where the OpenFlow middlebox forwards all frames that do not match any flow tables to the controller, which then reacts and decides how to process them. The controller updates flow entries in the middleboxes regularly as different flows cross the network. This method is inadequate for teleprotection networks, because the time necessary to react to a failure is dependent on latency to/from the controller, the processing time necessary for the controller to calculate a new network path, and the time it takes to instantiate a new flow rule into the OpenFlow middlebox. The second method of interaction between controller and middlebox is called proactive flow instantiation, in which the OpenFlow controller has already pre-calculated and instantiated the flow rules that match all normal and failure possibilities that the middleboxes should experience. This is an ideal scenario for teleprotection communication, because the OpenFlow middleboxes pass traffic much more quickly under dynamic circumstances and react to possible failure scenarios at or near line speed.

OpenFlow packets between controllers and middleboxes can be optionally encrypted with Transport Layer Security (TLS) with controller and middlebox identities verified using X.509 certificates. The enforcement of the confidentiality and integrity of OpenFlow communication represents a new shift towards stronger cybersecurity controls for control plane communication.

2.4.4 Examples of OpenFlow Failure Recovery

By programming an OpenFlow controller, users develop flow table logic, pre-engineer high-availability scenarios per middlebox and per link, and then update the flow tables in all OpenFlow middleboxes with this precalculated logic (proactive flow instantiation). Using a variety of methods, OpenFlow automatically calculates advanced failover scenarios that would effectively reroute traffic with only the loss of either the Ethernet frame currently being transmitted on the affected link or the frames currently in the output buffer for the particular port. With OpenFlow, the loss of a network link only

affects the OpenFlow middleboxes that are directly connected to it. Redundancy of links is easily performed by grouping ports together so that, on packet egress, the highest priority port currently available is used for the outgoing packet to the next hop or its final destination. By forwarding a packet to a group of ports, an OpenFlow-enabled middlebox will use the first available port to forward the frame to its destination. If the port closest to the destination is unavailable, flow tables previously designed by a traffic engineer forward the packet to the next middlebox closest to the relay (see Fig. 11).

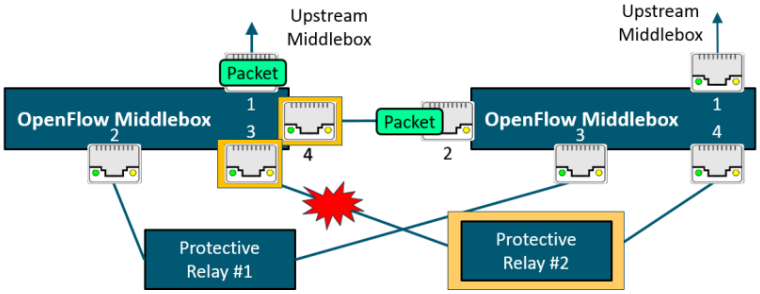

Fig. 11. OpenFlow Group Port Failover

In cases where ports are not grouped, logic can send a packet back out the original port from which it ingressed (see Fig. 12). This practice, impossible with STA but possible with OpenFlow, redirects traffic back into a middlebox to resend it out an alternate port. With OpenFlow, by using granular-enough rules, packets will always be delivered except for scenarios in which a link fails while the packet is on the link itself or when packets are queued in the outgoing port of the middlebox hardware.

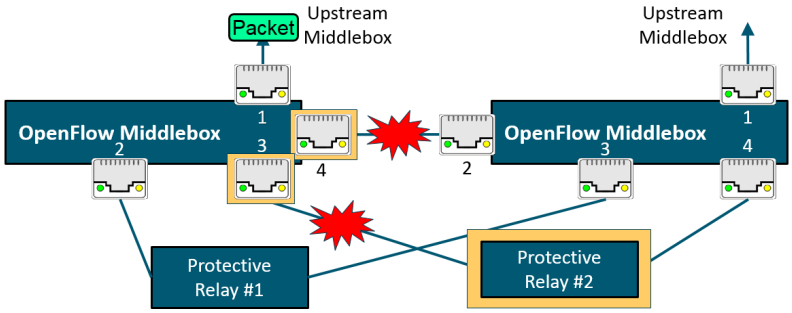

Fig. 12. OpenFlow Outputs Packet to Original Ingressing Port

OpenFlow can also copy ingressing critical frames and forward out multiple ports, or it can dedicate physical links for specific flows to minimize possible queueing or latency. Because OpenFlow can operate on either a blacklisting or whitelisting premise, it can also effectively firewall all communications. OpenFlow neither consumes additional bandwidth nor disables backup links, because only specific traffic flows need to be redundant and OpenFlow hardware can use all links. OpenFlow rules can be initiated on a schedule to enable maintenance modes where OpenFlow flow tables redirect all traffic around affected portions of the network without loss of data.

Consider the case of sending a GOOSE from a device connected to switch B3 to a device connected to switch B5 in the SDN Ladder topology illustrated in Fig. 13. Hamming distance for the example shown in Fig. 6(a) using an SDN 10-switch Ladder network middlebox topology is as follows.

Source IED	1 IED or 2 cables
Network component	18 middleboxes or 30 cables
Destination IED	1 IED or 2 cables

Hamming distance for the example shown in Fig. 6(b) using an SDN 10-switch ladder network middlebox topology is as follows.

Source IED	1 IED or 2 cables
Network component	18 middleboxes or 30 cables
Destination IED	2 IEDs or 4 cables

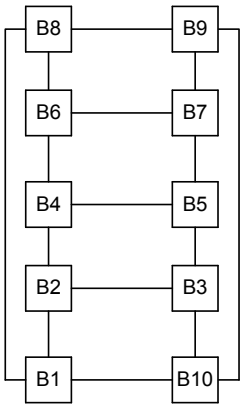

Fig. 13. Resilient Single Primary SDN Ladder Network Topology

2.5 Protection Considerations for Communications Failure Modes

Teleprotection schemes rely on digital status bits to and from remote devices. These signals are often used in pilot schemes as permissive or blocking signals and are used in some cases as transfer trip signals.

When the communications system functions properly, the states of these digital signals are constantly updated.

Communications failure modes include loss of outgoing data, the loss of incoming data, or both. As described previously, the communications system can provide status of the channel, preferably covering the health of the link in either direction.

2.5.1 Systems Where Loss of Incoming or Outgoing Communications Can be Detected

The protection system design should include a predefined fallback method in each destination IED for when signal information communications is lost. Fallback methods include the following:

- Taking the communications-dependent scheme out of service in the affected IED.
- Providing a redundant scheme in another IED, preferably connected to a separate communications network.
- Providing a backup scheme that does not rely on communication (e.g., step distance or directional overcurrent elements).

  The result will sacrifice selectivity and dependability, with increased operating time. For example, the scheme may trip for faults outside the protected zone, and may trip slower than the fully enabled scheme.
- Tripping the protected line, or deenergizing protected equipment.
  - This scheme may be suitable for private systems or industrial feeders.
  - For safety reasons, operator intervention may be required to restore service.

2.5.2 Systems Where Loss of Incoming or Outgoing Communication Cannot Be Detected

The protection system design must cover cases in which the communications interface silently fails and then resumes. A single protection scheme is applied for all cases because the communications system status is unknown.

Considerations can include the following:

- Using permissive tripping schemes that will still operate without communication, with reduced selectivity, dependability, and with increased operating time in some cases.
- Providing a redundant scheme in another IED, preferably connected to a separate communications network.

2.5.3 Systems Where Loss of a Dual Primary Application Can be Detected

The destination IED trip devices in both the dual primary protection application and the dual primary trip application exchange message quality. Knowing that communications necessary for the dual primary protection or trip has failed is not cause to change local logic. However, because this is learned so quickly via this method, the healthy destination IEDs can alert SCADA and engineering immediately that the dual primary has failed.

2.6 Composite System Availability Based on Hamming Distances

As with other quality measures, the system Hamming distance is the sum of components that must fail simultaneously and the product of components that cause failure individually.

The use of Hamming distance values to contrast the various methods illustrates the inadequacy of both PRP and HSR. This is expected because they both mask instead of correct failures. If we consider only network component failures, STA is 5.75 times and SDN is 7.5 times more available and secure than PRP. Also, STA is 11.5 times and SDN is 15 times more available and secure than HSR.

Table I
Network Component Hamming Distance Comparison of Recovery Methods Within Networks Built With 10 Middleboxes

Method	Network Middlebox, cable
Single Primary, Dual Paths	
PRP Ring	4, 4
HSR Ring	2, 2
STA Ladder	16, 23
SDN Ladder	18, 30
Dual Primary	
STA Ladder	16, 23
SDN Ladder	18, 30

However, when we consider the entire system, failure of the source IED(s), network components, or destination IED(s) cause system failure. Therefore, the product of individual Hamming distances is the system Hamming distance. If we consider devices only, and not cable failures, the system Hamming distances are as follows.

Table II
System Hamming Distance Comparison of Recovery Methods Within Networks Built With 10 Middleboxes

Method	Source IEDs	Middlebox	Destination IEDs	Total
Single Primary, Dual Paths				
PRP Ring	1	4	1	4
HSR Ring	1	2	1	2
STA Ladder	1	16	1	16
SDN Ladder	1	18	1	18
Single Primary, Dual Trip Device				
STA Ladder	1	16	2	32
SDN Ladder	1	18	2	36
Dual Primary				
STA Ladder	2	16	2	64
SDN Ladder	2	18	2	72

If we consider component failures, Dual Primary STA is 16 times and Dual Primary SDN is 18 times more available and secure than PRP. Also, Dual Primary STA is 32 times and Dual Primary SDN is 36 times more available and secure than HSR.

3 Conclusion

To provide lossless signal transfer in the event of a network failure, IEC 61850 describes a method for an IED to send several redundant change-of-state indications in a burst via republications after a change of state occurs. If the failure is corrected before the burst is complete, lossless signal delivery is achieved. Protection application operations are also made redundant through DPTDs in the same network consuming the GOOSE message or another based on unique contact input information.

The most reliable application redundancy method is to provide two complete and separate systems, physically isolated from one another and acting as dual primary redundant applications. The second most reliable method is the use of redundant dual primary networks isolated by way of QVLANs. PRP provides duplicate information and GOOSE messages delivered by way of redundant Ethernet packets in duplicate networks. HSR provides no redundancy of application or information, but it does provide duplicate messages and redundant packets in a single-ring network. HSR and PRP can be combined in various topologies and connect traditional devices by way of RedBoxes. While PRP, HSR, and any combination of the two can provide redundant packets, they cannot provide redundant applications, information, or messages. Actual times of message transit during normal and failed network situations require specific staging and testing. Table III compares the different methods based on capability and complexity.

OpenFlow promises to be a future “best in class” choice for teleprotection systems because of its flexibility and because OpenFlow middleboxes make efficient use of existing links and simultaneously provide extremely granular control over passing Ethernet traffic. OpenFlow can support previously unsupported reliability scenarios, and it can even be programmed to emulate the primary benefits of protocols such as HSR, PRP, and others. Note that the number of flow rules required to support large or complex teleprotection networks may be a limiting factor. Because line-speed failure recovery requires proactive flow instantiation, it is unknown whether fully redundant flow-table rulesets can be efficiently reduced to fit within OpenFlow middlebox hardware currently existing on the marketplace, which can currently support several thousand flow table entries. However, the creativity that OpenFlow inspires will enable further development that will increase the reliability of protection networks while decreasing cost and complexity.

Finally, considering component failures, Dual Primary STA is 16 times and Dual Primary SDN is 18 times more available and secure than PRP. Also, Dual Primary STA is 32 times and Dual Primary SDN is 36 times more available and secure than HSR.

Table III
Summary of Metrics for Duplication and Redundancy Methods

Duplication and Redundancy Methods	Lossless Signal Delivery	Redundancy					Coexist Standard Ethernet	Require Custom Device Software	Require Custom Device Hardware	Increase Message Delivery Latency	Increase Failure Recovery Latency	Logic Aware Duplication Failed	Network Bandwidth Available	Added Network Cost
		Message Burst	Application	Signal Data	GOOSE Message	Ethernet Packet								
Single application and ladder network, fast STA (Fig. 2[a], Fig. 2[b])	Y	Y	N	N	N	N	Y	N	N	N	N	N	All	None
Single application, single ladder network, fast STA, dual input contacts	Y	Y	N	Y	N	N	Y	N	N	N	N	N	All	None
Dual application, single ladder network, fast STA, DPTD (Fig. 4[b])	Y	Y	Partial	N	Y	Y	Y	N	N	N	N	Y	All	None
Dual application, single ladder network, fast STA, dual input contacts, DPTD (Fig. 4[c])	Y	Y	Partial	Y	Y	Y	Y	N	N	N	N	Y	All	None
Dual primary applications, physically separate networks (Fig. 4[a])	Y	Y	Total	Y	Y	Y	Y	N	N	N	N	Y	All	Double
Single application, duplicate GOOSE, dual primary networks isolated via QVLANs (Fig. 5[b])	Y	Y	N	N	N	Y	Y	N	N	N	N	Y	All	Double
Single application, dual inputs, redundant GOOSE, dual primary networks isolated via QVLANs (Fig. 6[a])	Y	Y	N	Y	Y	Y	Y	N	N	N	N	Y	All	Double
Dual application, dual contact inputs, redundant GOOSE, dual networks isolated via QVLANS (Fig. 6[b])	Y	Y	Partial	Y	Y	Y	Y	N	N	N	N	N	All	Double
Single application, duplicate messages, fast STA ladder networks isolated via PRP (Fig. 6[a])	Y	Y	N	N	N	Y	Y	Y	N	N	N	N	All	Double
Single application, duplicate messages, single ring network via HSR (Fig. 9[a])	Y	Y	N	N	N	Y	N	Y	Y	Y	Y	N	One-half	> Double
Single application, duplicate GOOSE, dual primary paths within single SDN network	Y	Y	N	N	Y	Y	N	N	N	N	N	Y	All	None
Single application, dual contact inputs, redundant GOOSE, dual primary paths within single SDN network	Y	Y	N	Y	Y	Y	N	N	N	N	N	Y	All	None

4 References

[1] D. Dolezilek, C. Gordon, D. Anderson, and T. Tibbals, “Modern Ethernet Failure Recovery Methods for Teleprotection and High-Speed Automation,” proceedings of the 5th International Scientific and Technical Conference: Actual Trends in Development of Power System Relay Protection and Automation, Sochi, Russia, June 2015.

[2] P. Franco, G. Rocha, and D. Dolezilek, “Case Study: Increasing Reliability, Dependability, and Security of Digital Signals Via Redundancy and Supervision,” proceedings of the 5th International Scientific and Technical Conference: Actual Trends in Development of Power System Relay Protection and Automation, Sochi, Russia, June 2015.

[3] S. Chelluri, D. Dolezilek, J. Dearien, and A. Kalra, “Understanding and Validating Ethernet Networks for Mission-Critical Protection, Automation, and Control Applications,” March 2014. Available: https://www.selinc.com

[4] IEC/TR 61850-90-4, Communication Networks and Systems for Power Utility Automation – Part 90-4: Network Engineering Guidelines.

[5] D. Bekker, T. Tibbals, and D. Dolezilek, “Defining and Designing Communication Determinism for Substation Applications,” proceedings of the 40th Annual Western Protective Relay Conference, Spokane, WA, October 2013.

[6] E. O. Schweitzer III, K. Behrendt, and T. Lee, “Digital Communications for Power System Protection: Security, Availability, and Speed,” proceedings of the 25th Annual Western Protective Relay Conference, Spokane, WA, October 1998.

[7] H. Kirrmann, “Highly Available Automation Networks Standard Redundancy Methods: Rationales behind the IEC 62439 standard suite,” ABB Switzerland Ltd, Corporate Research, ABB 2012.

[8] IEC 62439-3, Industrial Communication Networks – High-Availability Automation Networks – Part 3: Parallel Redundancy Protocol (PRP) and High-Availability Seamless Redundancy (HSR).

[9] D. Dolezilek, C. Gordon, D. Anderson, S. McCreery, and W. Edwards, “Simplifying Teleprotection Communications With New Packet Transport Technology,” proceedings of the 5th International Scientific and Technical Conference: Actual Trends in Development of Power System Relay Protection and Automation, Sochi, Russia, June 2015.

[10] R. Bobba, D. R. Borries, R. Hilburn, J. Sanders, M. Hadley, and R. Smith, “Software-Defined Networking Addresses Control System Requirements,” April 2014. Available: https://www.selinc.com.

[11] *Open Networking Foundation*. Available: https://www.opennetworking.org

5 Biographies

David Dolezilek received his BSEE from Montana State University and is the international technical director at Schweitzer Engineering Laboratories, Inc. He has experience in electric power protection, integration, automation, communication, control, SCADA, and EMS. He has authored numerous technical papers and continues to research innovative technology affecting the industry. David is a patented inventor and participates in numerous working groups and technical committees. He is a member of the IEEE, the IEEE Reliability Society, CIGRE working groups, and two International Electrotechnical Commission (IEC) technical committees tasked with the global standardization and security of communications networks and systems in substations.

Colin Gordon is an application engineer at Schweitzer Engineering Laboratories, Inc. (SEL) in the wired communications division, specializing in communication and cybersecurity solutions and services for critical infrastructure. His work experience includes secure network design, implementation, testing, and regulatory compliance consultation for utilities and asset owners in North America and abroad. He joined SEL in January, 2008, as a product management intern, and he holds a bachelor’s degree in computer engineering from the University of Idaho.

Dwight Anderson received his B.S. in electrical engineering from Steven's Institute of Technology. He is now a security engineer for Schweitzer Engineering Laboratories, Inc. (SEL) in Pullman, Washington. Prior to joining SEL in 2005, he worked 20 years for Hewlett-Packard as a business development manager and systems engineer, working on projects ranging from signal intelligence systems to SCADA system programming. He is an active member of the FBI InfraGard team. He is a professional engineer in Texas and a Certified Information Systems Security Professional (CISSP).

Timothy Tibbals received his BSEE from Gonzaga University in 1989. After graduation, he joined Schweitzer Engineering Laboratories, Inc. (SEL) as an application engineer, performing system studies and relay testing. Tim has also worked as a development engineer and as part of the development team for many of the communications features and functions of SEL products. He subsequently worked as an application engineer for protection, integration, and automation products, assisting customers through product training, seminars, and phone support. Tim served as the automation services supervisor in the SEL systems and services division for several years before returning to the research and development division as a product engineer for automation and communications engineering products. He is currently a senior automation system engineer in the research and development division.

David Keckalo received his Bachelor of Applied Science from the University of British Columbia in 1987. He joined Schweitzer Engineering Laboratories, Inc. (SEL) in 1998 and is a Lead Power Engineer in Protection Systems. He has worked on the design and development of many of SEL's protective relay products, including product literature. Prior to SEL, David held various positions at BC Hydro, concluding his 10 years of service as Senior Distribution Engineer. He holds one U.S. patent, is a Registered Professional Engineer (B.C.), and is a member of the IEEE.

Abstract—Ethernet networks are increasingly used for protection and high-speed automation as part of the trend to combine all substation communications onto a single shared network. A primary question now is how do we ensure the dependability and security of these applications when using Ethernet for signaling?

There are several mechanisms available to deal with different communications network failure scenarios. This paper compares and contrasts the following methods and their abilities to satisfy performance requirements expected for protection and high-speed automation:

- Classic dual-primary architectures.
- Spanning tree algorithms (Spanning Tree Protocol/Rapid Spanning Tree Protocol).
- Parallel Redundancy Protocol.
- High-Availability Seamless Redundancy.
- Software-defined networking.

Specifically, this paper addresses how each failure-handling method works and selection criteria for determining the failure-handling method to apply given the particular objectives of the protection or automation application. The paper also provides actionable information on the recovery time for any given failure-handling protocol for any specific failure scenario.

20151019 • TP6716