

Permissive or Blocking Pilot Protection Schemes? How to Have It Both Ways

Bogdan Kasztenny, Mangapathirao V. Mynam, Normann Fischer, and Armando Guzmán
Schweitzer Engineering Laboratories, Inc.

Presented at the
47th Annual Western Protective Relay Conference
Virtual Format
October 20–22, 2020

Permissive or Blocking Pilot Protection Schemes? How to Have It Both Ways

Bogdan Kasztenny, Mangapathirao V. Mynam, Normann Fischer, and Armando Guzmán,
Schweitzer Engineering Laboratories, Inc.

Abstract—This paper reviews permissive and blocking pilot schemes for protection of transmission lines. It covers principles of operation, settings considerations, the importance of coordinating the forward and the reverse fault detectors, applications to multiterminal lines, and single-pole tripping. The paper also describes add-ons used for dependability of the permissive schemes: open breaker echo, weak-infeed logic, and channel failure logic. The paper introduces a crossover permissive-blocking scheme that eliminates the need for additional engineering required by many applications of permissive schemes, while allowing fast tripping without the coordination time. The scheme is easy to configure and test, and it uses two pilot bits in a multi-bit digital protection channel.

I. INTRODUCTION

Directional comparison protection schemes (or pilot protection schemes) are a workhorse of transmission line protection. Widely used since protection channels became reliable and affordable, pilot protection schemes deliver excellent return on investment. Operating by using only one pilot bit per line terminal, pilot protection schemes provide instantaneous, selective, and dependable protection for the entire line (unit protection).

In the last two decades, the availability of digital protection channels, combined with increasingly challenging system conditions (mostly due to proliferation of nonstandard power sources), has propelled widespread applications of line current differential (87L) schemes. 87L schemes are easier to apply from the protection settings perspective, but they put much more stringent requirements on communications networks, and they require paying more attention to the engineering and maintenance of communications networks. Overall, pilot protection schemes are simpler, more robust, and more economical, and they will continue to be used for decades to come, regardless of the increased application of 87L schemes. Often, a pilot protection scheme and an 87L scheme are applied as two independent and redundant unit protection systems for transmission lines.

A pilot protection scheme uses a set of forward-looking protection elements to detect line faults. For dependability, these elements must overreach the remote line terminal(s). Typical applications use overreaching phase and ground distance elements and zero- and negative-sequence directional overcurrent elements. A pilot protection scheme achieves selectivity by comparing fault directions at each terminal of the protected line. Several variants are possible, but the two general categories are permissive and blocking pilot protection schemes (with the permissive scheme having a few subcategories). A

permissive scheme follows the principle that “all relays in the scheme see the fault in the forward direction, therefore the fault must be internal to the protected line.” A blocking scheme follows the principle that “none of the other relays see the fault in the reverse direction, therefore the fault seen as forward at a given terminal must be internal to the protected line.”

A permissive scheme operates based on an explicit confirmation of the forward fault direction at *all* remote terminals. A blocking scheme operates based on an absence of confirmation of the reverse fault direction at *any* of the remote terminals.

These operating principles result in a different emphasis for relay and channel availability and protection element performance. A permissive pilot protection scheme operates if all relays in the scheme are operational and detect a forward fault condition and all channels in the scheme are operational and deliver the permissive pilot signals between the relays in the scheme. A relay in a blocking pilot protection scheme operates if it detects a forward fault condition and does not receive a blocking signal, indicating that the relay(s) at the remote terminal(s) has not detected a reverse fault.

Under nominal conditions, when all channels are healthy and all relays in the scheme are operational and able to detect faults, there is no difference in performance between the two types of schemes, except in speed. Permissive schemes are faster because they do not need to intentionally delay operation to account for the slowest possible channel time plus margin.

The two scheme types exhibit advantages and disadvantages only under unusual or failure conditions, and this results in different tradeoffs to address failure modes. Under failures, permissive schemes sacrifice dependability, while blocking schemes sacrifice security. The unusual conditions and failure modes include the following four categories:

- A permanent loss of function, such as relay or channel failure.
- A predictable obstacle to protection element operation, such as open breaker or weak-infeed conditions.
- A sporadic problem with protection element operation, such as infeed effect during a resistive fault, less-than-optimum or marginal settings, saturation of current transformers (CTs), and so on.
- A sporadic problem with the protection channel operation, such as carrier holes, spurious permissive pilot bit assertion, and so on.

Monitoring, combined with fast crew response times, can mitigate the impact of a permanent loss of function. A

permissive scheme can address the predictable obstacles to protection element and channel dependability by using several add-ons, including weak-infeed echo, open breaker echo, and channel fail logic (i.e., a so-called directional comparison unblocking [DCUB] scheme).

Historically, the type of channel drove the choice between permissive and blocking schemes. Blocking schemes are preferred when the protection channel shares the right of way with the protected line (power line carrier and optical ground wire [OPGW] fiber-optic applications), which exposes the scheme to the effects of the fault that it is supposed to detect. Also, blocking schemes have been preferred when the protection channel is a simple on/off channel (power line carrier). Today, the self-monitoring embedded in microprocessor-based relays and digital protection channels alleviates the permanent loss of function concern. Engineering and settings considerations, historical track record, and field personnel familiarity drive the choice of the scheme type [1]. Utilities minimize the risk and maximize their field experience and workforce skillset by adhering to the scheme type used in the past, while taking only limited advantage of new technology such as relay self-monitoring and protection channels, digital protection channels that are capable of carrying multiple pilot bits at no additional cost, and programmable logic in relays.

This paper is a tutorial on pilot protection schemes. It reviews the principles, advantages, and disadvantages of permissive and blocking schemes. The paper explains the merits of using multiple overreaching elements for sensitivity and speed and related application considerations. The paper also reviews add-ons to a permissive scheme (weak-infeed logic, open breaker logic, and channel failure logic) and discusses their applications. Finally, it presents a crossover scheme that combines characteristics of permissive and blocking schemes for the benefit of speed and dependability while reducing engineering effort and increasing simplicity.

II. PILOT SCHEME LOGIC

A. Pilot Scheme Operating Principles

Fig. 1 shows a simplified diagram of a permissive overreaching transfer trip (POTT) logic.

PILOTF is a forward-looking, overreaching protection element, or a combination of several elements, enabled and configured to detect faults on the protected line. Typically, the PILOTF condition includes the overreaching phase and ground distance Zone 2 elements, and the zero- and/or negative-sequence directional elements.

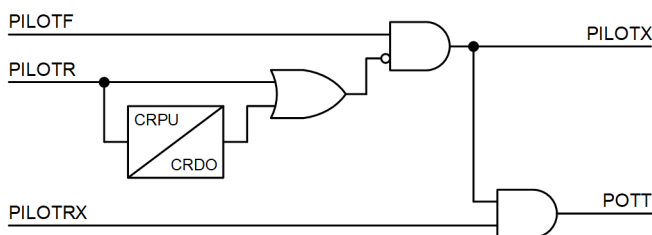


Fig. 1. Simplified POTT logic

PILOTR is a reverse-looking protection element, or a combination of several elements, enabled and configured to detect reverse faults that are within the reach of the PILOTF elements at the remote line terminal. In terms of element type and sensitivity, the PILOTR condition in the local relay must match the PILOTF condition in the remote relay (this paper refers to the local terminal or relay as the one that is closer to the fault, or the one whose logic it describes).

The primary function of the PILOTR condition in the POTT logic is to drive a current reversal timer and prepare the scheme for a race condition between various bits changing their value, which occurs when an external fault is cleared on a parallel line (or a parallel path). If the PILOTR condition asserts for a reverse fault for a time period longer than the current reversal pickup (CRPU) time (on the order of a power cycle), the current reversal logic extends the blocking signal for an additional current reversal dropout (CRDO) time (on the order of several cycles).

Some permissive schemes use the PILOTR condition to override the PILOTF condition instantaneously, even before the current reversal pickup timer asserts, often on a per-phase basis. This interlocking of the PILOTF condition with the PILOTR condition enhances the overall security of the scheme as well as selectivity of single-pole tripping applications for cross-country faults. A POTT scheme that applies such instantaneous interlocking is referred to as a *hybrid* scheme because it includes elements of both a permissive scheme and a blocking scheme. This instantaneous blocking of the PILOTF condition with the reverse PILOTR condition is critical to protection security in POTT applications that use a single permissive bit and share it among a wide range of protection elements, as explained in the *Using Multiple Directional Elements* subsection.

The scheme in Fig. 1 asserts the permissive pilot bit, PILOTX, for forward fault conditions. The scheme operates by asserting the POTT bit when the local relay detects a forward fault (PILOTX asserted), and at the same time, it receives the permissive pilot bit, PILOTRX.

Fig. 2 shows a simplified diagram of a directional comparison blocking (DCB) logic. The scheme is similar to the POTT scheme in terms of using the PILOTF condition to detect line faults. The role of the PILOTR condition is, however, more critical. In a blocking scheme, the PILOTR condition must detect all reverse faults that are within the reach of the PILOTF condition in the remote relay; otherwise, the scheme will misoperate at the remote terminal.

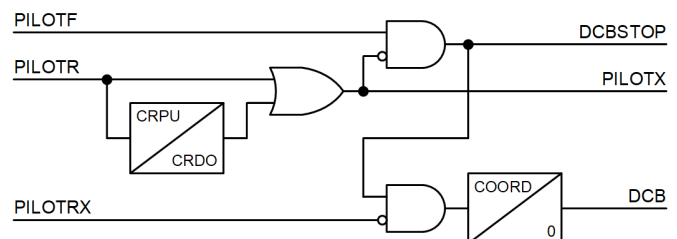


Fig. 2. Simplified DCB logic

The scheme operates by asserting the DCB bit when the local relay detects a forward fault and does not receive the blocking pilot bit, PILOTRX. A coordination timer (COORD) is used to allow time for the remote relay to detect a reverse fault and for the protection channel to deliver the blocking bit.

The DCB scheme may be used with an on/off protection channel, which effectively uses a flip-flop logic to explicitly turn the blocking pilot bit on (PILOTX bit) and turn it off (DCBSTOP bit). To facilitate these applications, the DCB scheme in Fig. 2 asserts the DCBSTOP bit if it detects a forward-fault condition. This pilot-signaling method that uses two bits (start with the PILOTX bit and stop with the DCBSTOP bit) allows applications with nondirectional starting and improves overall scheme security by not requiring the PILOTR condition to be permanently asserted.

Some scheme designs (permissive or blocking) use the underreaching distance Zone 1 element to reset the current reversal dropout timer in order to allow the scheme to trip for an evolving external-to-internal fault. In this scenario, the external fault asserts the current reversal logic and inhibits the scheme, but when the distance Zone 1 element detects an internal fault, it resets the current reversal blocking action and allows the pilot scheme at the remote line terminal to operate.

B. Pilot Scheme Similarities

The schemes in Fig. 1 and Fig. 2 have many similarities. The figures use the same labels to emphasize the same logical conditions in both schemes. Specifically, both schemes

- Use a forward-looking overreaching condition, PILOTF, to detect line faults.
- Use a reverse-looking condition, PILOTR, to detect reverse faults.
- Use a pilot output bit, PILOTX, to transmit the permissive (POTT) or the blocking (DCB) pilot bit.
- Use a configurable input bit, PILOTRX, to receive the pilot bit.
- Assert a pilot TRIP bit (POTT in Fig. 1 or DCB in Fig. 2).

Some pilot scheme implementations, such as [2], take advantage of these similarities and provide a single pilot logic that enables either the permissive (hybrid POTT) or blocking (DCB) logic. Fig. 3 shows a functional block of such a logic (EPILOT is the enable setting and DCBSTOP applies to the DCB scheme only). In this paper, this general logic is referred to as PILOT logic (a function in a relay) or a PILOT scheme (multiple relays and associated protection channels).

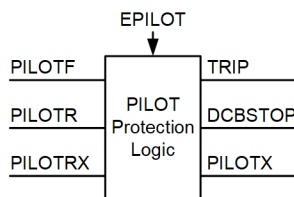


Fig. 3. PILOT logic functional block

C. Pilot Scheme Customization

Today's relays provide programmable logic equations for binary inputs to protection elements and schemes, and for binary outputs of the relay (contact outputs and communications-based outputs). Programmable logic equations allow the following additional flexibility and customization of the PILOT logic (see Fig. 4):

- The relay can make the PILOTF condition programmable as a multichoice list of selected protection elements, or the user can take full control over the forward fault condition by programming a logic equation.
- The relay can be configured to match the PILOTR condition to the PILOTF condition (protection elements of same kind), or the user can take full control over the reverse fault condition by programming a logic equation.
- The PILOTRX input pilot bit can be a programmable equation that allows implementing multiterminal line logic, adding test bits, programming redundant channel applications, and so on.
- The PILOTX output pilot bit can be programmed in equations that drive the relay binary outputs to send a signal to the remote relays in the PILOT scheme while applying additional supervision or customization, as required.

Fig. 5, Fig. 6, and Fig. 7 are examples of the flexibility that can be achieved through logical equations. Fig. 5 shows a three-terminal POTT application with a test bit, TEST, used to force the outgoing and incoming pilot bits to logical 0. Fig. 6 shows a two-terminal DCB application with a redundant channel. Fig. 7 shows a permissive underreaching transfer trip (PUTT) logic application obtained by using the overreaching elements in the PILOTF condition, but supervising the permissive output with the underreaching distance Zone 1 element (Z1). The seal-in logic ensures that chattering of the Z1 bit for faults near the reach point will not cause the permissive bit to chatter.

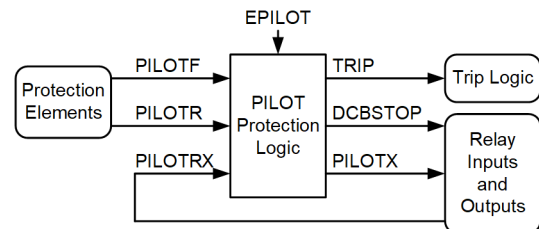


Fig. 4. PILOT logic application in a relay with programmable logic

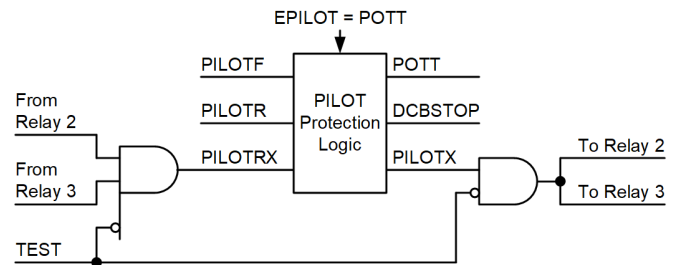


Fig. 5. Three-terminal POTT application with a test bit

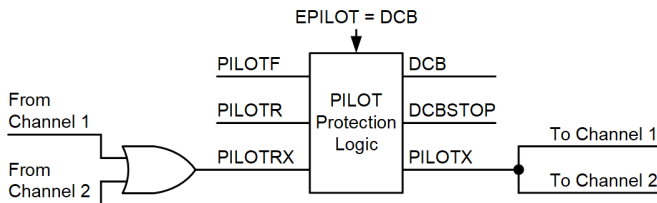


Fig. 6. Two-terminal DCB application with a redundant channel

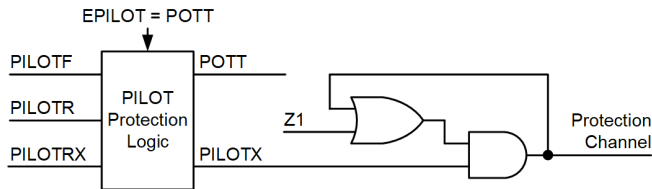


Fig. 7. PUTT application of the PILOT logic

III. SETTINGS CONSIDERATIONS

The following general settings rules apply to pilot protection schemes:

- **Rule 1:** The PILOTF condition must assert for all line faults that are expected to be cleared instantaneously under all applicable contingencies.
- **Rule 2:** The PILOTR condition must assert for all reverse faults for which the remote PILOTF condition asserts (a strict requirement for the DCB scheme; it can be relaxed for current reversal application in the POTT scheme). Preferably, the PILOTR condition in the local relay should assert faster than the PILOTF condition in the remote relay.
- **Rule 3:** The current reversal pickup timer must assert before a circuit breaker starts clearing an external fault, potentially creating race conditions when the directional elements respond to the change in current flow direction. Ultra-high-speed relays applied with fast breakers require short values of this timer, on the order of a half or even a quarter power cycle [2] [3].
- **Rule 4:** The DCB coordination timer, COORD, must be longer than the slowest channel time, plus the difference (if any) in the operating times of the local PILOTF condition and remote PILOTR condition, plus margin.

A. Setting PILOTF Protection Elements

The purpose of the protection elements configured to drive the PILOTF condition is to detect those line faults that require instantaneous clearing, including fault resistance, infeed effect, mutual coupling, line and load unbalance, and system conditions comprising all applicable contingencies. Typical PILOTF applications include overreaching phase and ground distance elements and sequence directional elements (the term *sequence* in this paper refers to zero and negative sequence).

Overreaching phase and ground distance elements have the advantage of well-controlled reach. Their sensitivity is limited to metallic (zero-resistance) faults or faults with relatively small resistance. For out-of-zone faults, a distance element is often subject to an infeed effect, which makes it less likely to assert for external faults. All these characteristics make overreaching

distance elements a very secure choice for driving the PILOTF condition.

In applications to two-terminal lines without mutual coupling, the line positive-sequence impedance can be used as the setting base and the overreaching distance element can be set to about 120 percent of the line impedance. In more advanced applications involving mutual coupling and multiterminal lines, a best practice is to use a short-circuit program for calculating the apparent impedance for end-of-line faults for all remote terminals under all applicable contingencies. For lines with mutual coupling, these contingencies must include the coupled line(s) in service and out of service. When considering the out-of-service scenario, these lines must be modeled following the utility grounding procedures (one terminal, both terminals, grounding at the place of field work, etc.). Data in short-circuit databases related to mutual coupling have limited accuracy, and therefore, larger settings margins should be applied to the ground elements for the PILOTF condition.

The sequence directional overcurrent elements are used in the PILOTF condition for their sensitivity. However, they do not have a well-controlled reach and they may assert for faults far from the protected line, or even for no fault conditions (these elements detect direction of the location of an unbalance). While distance elements typically do not need current reversal logic, sequence directional elements require current reversal logic in the pilot protection logic. The sensitivity of the sequence directional elements can be intentionally limited by using overcurrent thresholds. Changing system conditions can still allow them to reach far, especially if these elements are set to detect high-resistance faults. Another disadvantage of sequence directional elements is that they may legitimately disagree on fault direction for external faults in a meshed network. Reference [4] discusses issues with sequence directional elements in more detail. The *Using Multiple Directional Elements* subsection discusses the application of multiple elements in the PILOTF condition while sharing a common permissive pilot bit.

Typical PILOTF protection elements include the phase and ground distance elements and—for backup and additional sensitivity—either the zero- or negative-sequence directional element. The system strength and other factors, such as mutual coupling, drive the choice between the zero- and negative-sequence directional elements.

Just because the PILOTF condition is an overreaching condition, it does not mean that it should reach as far as possible or that the degree of overreach does not matter. It is a good practice to limit the reach to a value that ensures assertion for all line faults, but only with a reasonable margin. This recommendation is especially relevant for blocking schemes, where for each assertion of the PILOTF condition for an external fault in the remote relay, the local relay must assert the PILOTR condition, or else the scheme would misoperate. The PILOTF condition in one relay must coordinate with the PILOTR condition in the other relay, and limiting the reach of the PILOTF condition makes the coordination easier.

From this perspective, it may be beneficial to separate the overreaching distance element used in the pilot scheme and the overreaching distance element used for step distance protection. These two applications, even though similar, have slightly different settings criteria. Reference [5] discusses this aspect in more detail.

The sensitivity of the PILOTF condition should not be considered in absolute terms, but in conjunction with the weak-infeed logic. If the protection elements in the PILOTF condition in a relay at a given terminal are able to detect all line faults for which instantaneous tripping is required under all required contingencies, then this terminal is strong, and the permissive pilot scheme does not require the weak-infeed logic (protection elements in the PILOTF condition are based on current). Otherwise, the terminal is weak (under all or some contingencies), and the permissive pilot scheme requires the weak-infeed logic.

A given terminal being strong or weak depends on both the system short-circuit capacity and on the elements configured in the PILOTF condition and their settings. In many applications, a protection engineer draws an arbitrary line between faults that can be detected by using the PILOTF protection elements and faults that require the weak-infeed logic. The division of fault detection duties between the PILOTF elements and the weak-infeed logic is yet another consideration when applying permissive pilot schemes. See *Section V* for more information about the weak-infeed logic.

In single-pole tripping and reclosing applications, the relay blocks the sequence directional elements during the single-pole reclosing interval. In those applications, the PILOTF condition must rely on the distance element alone, or a phase directional element may additionally be used instead of the sequence directional element. In any case, protection sensitivity during the single-pole reclosing interval is reduced, and high-resistance internal-to-internal evolving faults may be cleared only after reclosing following the first fault.

B. Setting PILOTR Protection Elements

This section focuses on blocking pilot schemes because these schemes have coordination requirements that are more stringent than the hybrid permissive schemes. Applying the coordination requirements of blocking schemes in the hybrid permissive schemes is a good practice because it contributes to protection system security.

In reference to Fig. 8, the PILOTR condition in the local relay must assert for every external fault for which the PILOTF condition in the remote relay asserts.

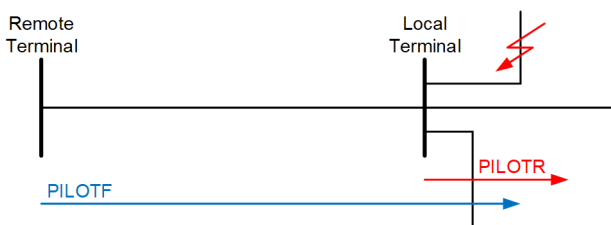


Fig. 8. Illustration of the PILOTF and PILOTR coordination principle

This general coordination rule has the following facets:

1. It is a best practice to use the same set of elements in the PILOTF and PILOTR conditions. Some implementations [2] match the PILOTR elements to the PILOTF elements that the user selected. If there is a need to deviate from this rule, users should not include more elements in the PILOTF condition than in the PILOTR condition.
2. It is a best practice to use like elements in the PILOTF and PILOTR conditions. The term *like elements* refers to the operating principle, built-in restraints, internal element thresholds such as directional limit angle, and so on. Using elements that are dissimilar may lead to miscoordination, even when the user gives the reverse PILOTR elements a very generous settings margin. Transient simulation testing is recommended when dissimilar elements or relays are used in the blocking scheme.
3. The inductive reach of the PILOTR element (i.e., ability to detect faults located further away from the terminal) in the local relay must be larger than the PILOTF element reach in the remote relay.
4. The resistive reach of the PILOTR element (i.e., ability to detect faults with resistance while the infeed effect maintains the fault-point voltage) in the local relay must be larger than the PILOTF element reach in the remote relay.

Settings can be used to address items 3 and 4 in this list (PILOTR elements are set to have larger inductive reach, larger resistive reach, lower overcurrent thresholds, wider comparator limit angles, and so on). When in doubt, perform short-circuit studies or transient simulation testing to ensure proper coordination.

Regarding distance element coordination, mho distance elements are considered first and then coordination between the forward distance Zone 2 element at the remote terminal and reverse distance Zone 3 element at the local terminal. For an external fault in a two-terminal line application, the apparent impedance seen by the forward zone in the remote relay (ZREM), minus the line impedance (ZILINE), equals the inverted reverse zone apparent impedance in the local relay ($-ZLOC = ZREM - ZILINE$). Fig. 9 illustrates this observation by plotting—on the impedance plane that is common for the local and remote relays—the forward Zone 2 at the remote relay and the inverted Zone 3 at the local relay.

Fig. 9 is plotted for self-polarized mho elements. It points to a potential for miscoordination. A resistive external fault may appear outside of Zone 3 at the local relay but inside Zone 2 at the remote relay. When considering memory polarizing (mho characteristic expansion), as shown in Fig. 10, the Zone 3 element characteristic expands more than the Zone 2 element characteristic, and the two zones coordinate properly.

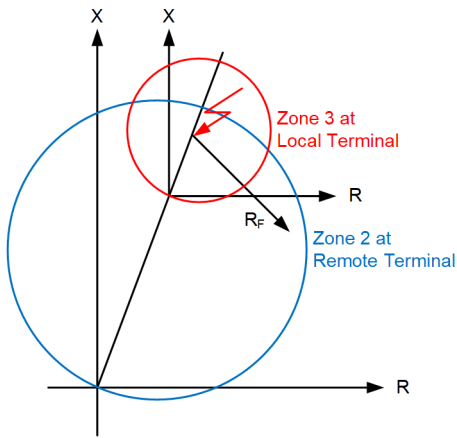


Fig. 9. Forward Zone 2 and reverse Zone 3 coordination (self-polarized mho)

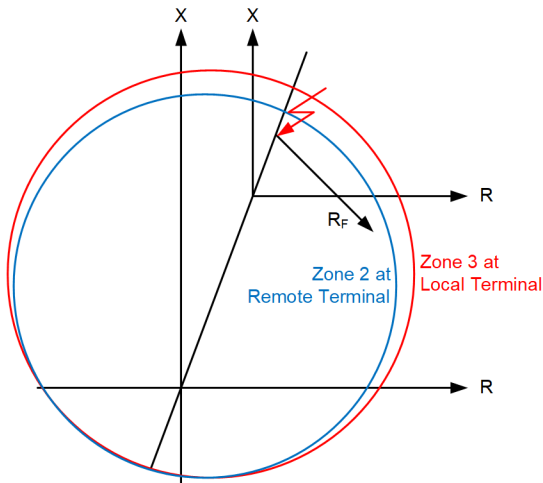


Fig. 10. Forward Zone 2 and reverse Zone 3 coordination (memory-polarized mho)

Coordination of quadrilateral distance elements deserves more attention. The quadrilateral characteristic does not expand the way the memory-polarized mho element characteristic does. Fig. 11 shows a quadrilateral Zone 3 characteristic properly coordinated by encompassing the upper part of the Zone 2 characteristic. However, depending on the type of Zone 3 directional supervision, a coordination problem for heavy infeed conditions may still occur.

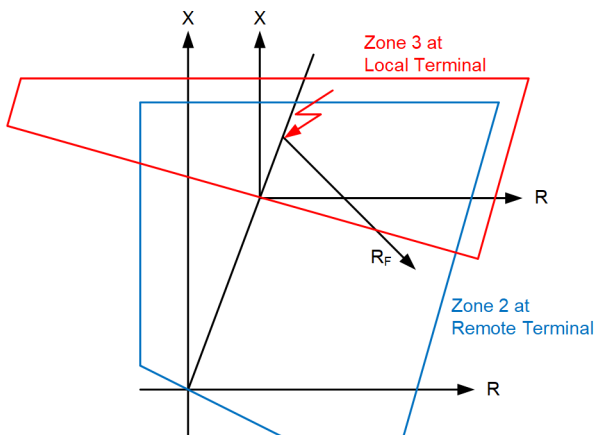


Fig. 11. Forward Zone 2 and reverse Zone 3 coordination (quadrilateral)

The following solutions help coordinating quadrilateral distance elements in the PILOTF and PILOTR conditions:

- The negative-sequence directional element in the PILOTR condition can be depended on to provide blocking for all unbalanced reverse faults. Three-phase balanced faults typically do not include high resistance, and the coordination challenge shown in Fig. 11 does not apply to the three-phase balanced faults.
- The distance Zone 2 element [2], when used in the PILOTF condition, uses sequence currents to polarize its reactance comparator in order to stop the element from operating for out-of-reach faults.
- A nondirectional (offset) distance element can be used to provide nondirectional starting (blocking) in the blocking scheme, as explained in the *Nondirectional DCB Starting* subsection.

Application to series-compensated lines or lines adjacent to series capacitors creates additional coordination challenges for blocking schemes. Capacitors bypassing operation with pole scatter and unequal conduction of the metal-oxide varistors (MOVs) add series unbalance to the shunt unbalance of the external fault. This series unbalance may upset the traditional relationships between sequence components and may create significant challenges for directional elements [6]. For this reason, series-compensated lines are protected with permissive pilot schemes, and blocking schemes are typically avoided. Using nondirectional starting (blocking) in the blocking schemes can solve some of the problems related to series compensation.

C. Nondirectional DCB Starting

Fig. 12 shows the application from Fig. 11, but with the local relay using a nondirectional quadrilateral distance Zone 5 element in the PILOTR condition. The operating characteristic of a nondirectional distance element extends in both directions: forward and reverse. The nondirectional distance element has two independent reactive reach settings that control the shape of its operating characteristic.

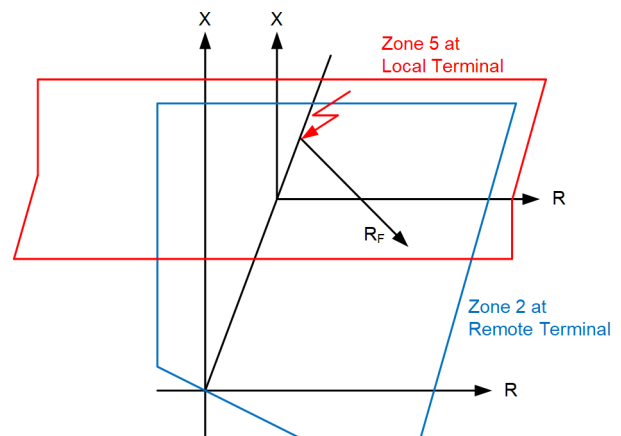


Fig. 12. Forward Zone 2 and nondirectional Zone 5 coordination

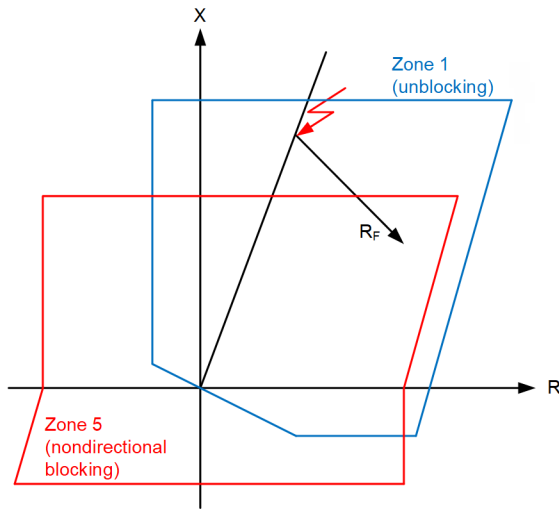


Fig. 13. Forward Zone 1 and nondirectional Zone 5 coordination

The nondirectional distance zone provides dependable blocking even for reverse faults with very heavy infeed. The nondirectional zone will assert for forward faults, but the forward elements also assert for forward faults and remove the blocking action (the Zone 1 distance element resets the current reversal dropout timer, and the PILOTF elements assert, allowing the scheme to operate).

When using the nondirectional zone for starting (blocking) in the pilot scheme, coordination between the forward part of the nondirectional distance element and the underreaching distance Zone 1 element must be ensured: for every internal fault that asserts the nondirectional zone, the Zone 1 element must also operate in order to remove the blocking action and allow the blocking pilot scheme to operate. Fig. 13 illustrates this coordination requirement.

D. Using Multiple Directional Elements

In meshed networks, fault direction is not an absolute concept. Advancing in the measuring direction of the CT, a path from the relay to an external fault can be found, but a path from the relay to an external fault can also be found by advancing in the direction opposite to that of the CT (see Fig. 14). An external fault is simultaneously forward and reverse.

Directional protection elements do not have a notion of absolute fault direction but only respond to their operating and polarizing quantities. Depending on the current flow, a directional element can declare an external fault as forward or reverse. Moreover, there may be a fault location for which a directional element will not operate at all (the forward and reverse current components cancel; the fault location that leads to this condition is sometimes referred to as *electrical center*).

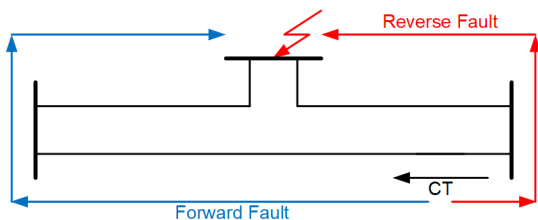


Fig. 14. External fault direction in a meshed network is not absolute

Pilot schemes can be configured to use the following directional elements:

- Phase and ground distance elements (forward Z2 and reverse Z3).
- Zero- and negative-sequence directional elements (32G, 32Q).
- Phase directional element (32P).
- Incremental-quantity directional element, TD32 [3].
- Traveling-wave directional element, TW32 [3].

The TD32 directional element is proven to be dependable [7] [8] [9]. Therefore, the TD32 element can be used in permissive schemes like other more traditional elements (Z2, 32G, and 32Q). However, it is a good practice to use the TD32 element in the blocking scheme only for issuing the blocking signal and not for allowing the TD32 element to initiate a pilot trip [2]. The TW32 directional element faces natural dependability limits imposed primarily by the poor frequency response of voltage transformers [2] [7] and during faults occurring at times near the voltage zero crossing.

When using multiple directional elements in a pilot protection scheme and allowing them to share (key) a common permissive bit, their response to external faults must be considered. Fig. 15 shows a case of an external fault in a meshed network on a parallel line or a parallel path. Because the Z2 / Z0 ratios are assumed to be very different for the two equivalent systems, the sequence currents in the healthy line flow in opposite directions. As a result, the distance (Z2) element, the zero-sequence directional (32G), and the negative-sequence directional (32Q) elements may declare different fault directions.

Each directional element in Fig. 15 works correctly: when a particular element at one terminal declares the forward direction, the same element at the other terminal declares the reverse direction. However, if these elements are combined through an OR gate, the PILOTF condition will assert at both terminals, resulting in the permissive scheme misoperation. One solution to this security challenge is to avoid combining the permissive signals from various elements and to send the permissive signals separately for each element instead. This separation is relatively easy to implement with digital protection channels because these channels carry multiple bits (such as four or eight).

Note, however, that the hybrid permissive scheme is secure when multiple directional elements send the permissive signal on a common pilot bit because at least one of these elements operates in the reverse direction and asserts the PILOTR condition.

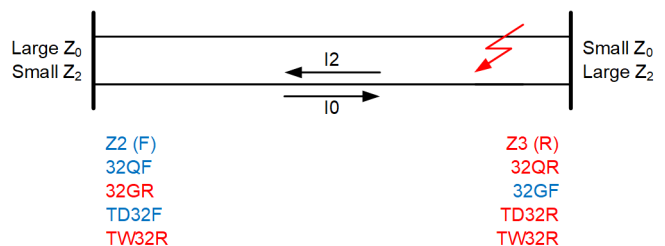


Fig. 15. PILOTF elements disagreeing on fault direction

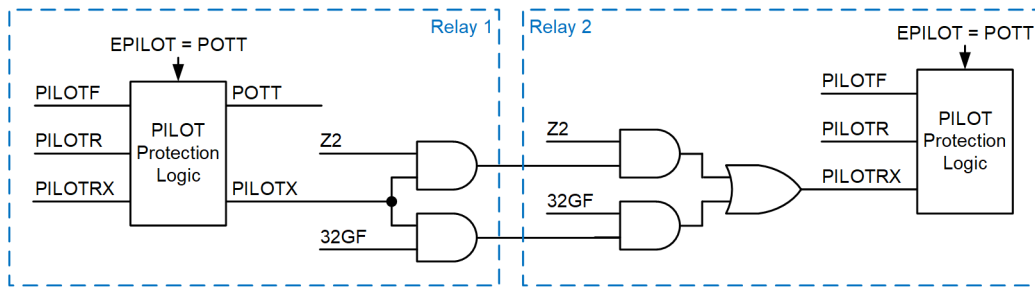


Fig. 16. Example of using separate pilot bits for Z2 and 32G conditions

Applications that use multiple directional elements to key the common permissive bit are secure as long as the used elements are fully dependable: if a given element asserts forward for an external fault at one terminal, the matching element at the other line terminal must assert reverse for the hybrid (or blocking) scheme to remain secure.

During external faults, the TW32 element may also disagree with other directional elements. The TW32 element asserts the reverse direction if the very first TW arrived from the system behind the relay. Often, both terminals assert the TW32 in the reverse direction for an external fault (see Fig. 15), which is an advantage compared with other elements. However, there may be external fault cases, especially when cable lines are involved, where the very first TW arrives through the protected line and causes the TW32 element to assert forward. At the same time, however, the TW32 element in the relay at the opposite line terminal may experience dependability issues, and it may fail to assert in the reverse direction. Therefore, if used in a permissive scheme, the TW32 element must use a separate bit to key permission so that the TW32 forward assertion at one line terminal can be confirmed by the TW32 forward assertion at the other terminal.

The TD32 and TW32 directional elements benefit pilot protection, but they should be used in a way that recognizes their strengths (speed and sensitivity) and addresses their limitations (dependability limits and temporary assertion). Implementation [2] uses the TW32 and TD32 elements as follows:

- Both the TW32 and TD32 elements are allowed to send the blocking bit.
- The TW32 element can be used to send the permissive bit, but only by using a separate pilot bit.
- The TD32 element is allowed to initiate the pilot trip, but only after supervision by the incremental-quantity overcurrent element.
- The TW32 element is allowed to initiate a permissive pilot trip, but only if another directional element confirms forward fault direction and the permissive bit from the TW32 element arrives by using a separate pilot bit.
- The TW32 element is not allowed to initiate a blocking scheme trip.

The PILOT logic implementation [2] with the functional block, shown in Fig. 3 and Fig. 4, can be used to separate permissive pilot bits, if needed. Fig. 16 shows an example of separating the permissive signal originating from the distance

Zone 2 element (the Z2 bit) and the 32G element (the 32GF bit). Fig. 16 illustrates the application of programmable logic for conditioning the sent (PILOTX) and received (PILOTRX) bits using programmable logic equations. Separating the permissive bits adds additional flexibility. For example, the 32GF-driven permissive bit may use a pickup timer in the sending relay and the receiving relay for additional security when the 32G direction element is set to be very sensitive.

IV. OPEN BREAKER AND STUB-BUS LOGIC

An open circuit breaker prevents the PILOTF protection elements from detecting line faults, and therefore, it prevents the permissive pilot schemes from operating. A blocking pilot scheme operates without the need for the terminal with an open breaker to detect the fault, and therefore only permissive schemes need to address the open breaker scenario.

A breaker can be open temporarily when the line protection system trips the line, and the autoreclosing logic tests the line from one terminal by closing the lead breaker. When the line is being re-energized, the follower breaker is still open. The line protection system typically uses a switch-onto-fault logic to trip for permanent faults when reclosing, and the loss of dependability of the permissive scheme during reclosing is not relevant (the key challenge when energizing the line is not the pilot signal, but the polarizing of the distance elements, hence the application of the switch-onto-fault logic).

A breaker can be open for a prolonged time—typically for maintenance—while the line continues to be energized and carries power to tapped load or from tapped generators, or when a multiterminal line connects other terminals. If the line protection system uses a permissive pilot scheme, that scheme would need to address the prolonged open breaker scenario.

A stub-bus situation occurs in dual-breaker applications when the two breakers that connect the line are closed to keep the ring bus intact or to tie the two buses together in the breaker-and-a-half arrangement. At the same time, the line disconnect switch is open while the line is energized from the remote terminal(s). The relay with the stub-bus connection uses an instantaneous overcurrent element (or a current differential element) to detect faults on the stub bus. If the line protection system uses a permissive pilot logic, that logic needs to address the stub-bus scenario in order to protect the line.

The two solutions to the open breaker or open disconnect switch scenario are the *open breaker key* logic and the *open breaker echo* logic. As their names imply, the open breaker key logic permanently keys permission to the remote relay(s) for as

long as the breaker is open. The open breaker echo logic echoes received permission if it receives the permissive signal while the breaker is open.

Detection of an open breaker condition can be performed based on the 52a breaker status signals or based on both the 52a status and the measured current (according to the open-pole detection logic). In the case of monitoring a disconnect switch rather than a breaker, both the 89a and 89b status signals can be used in a dual-point status logic for security. To avoid spurious open breaker or disconnect switch declaration, a time delay is also used.

Before allowing the open breaker echo logic to echo the received permissive signal, a time delay can be used to make sure the received permissive signal is legitimate and is not a spurious signal induced by noise in the protection channel.

The open breaker key logic is simpler and faster than the open breaker echo logic but is less secure: any PILOTF assertion combined with a permanently asserted permission from the terminals with an open breaker would result in a trip. The open breaker echo logic is slower because it applies a short time delay to verify the received permissive signal, and it delivers permission to the other line terminals after twice the channel time (it must receive permission before it can echo it back).

At the same time, the open breaker echo logic is more secure because it applies a time delay to verify that the remote relay truly detected a forward fault. Also, the open breaker echo logic does not key the channel permanently, but only for a short time, which is a challenge for some analog protection channels (digital channels are not concerned with permanently asserted pilot bits and can equally send bit 1 or bit 0 for an indefinite time).

The open breaker echo logic is applicable to permissive pilot schemes, and therefore it can be built into those schemes [2]. Fig. 17 shows a simplified diagram of the open breaker echo logic.

The ECHORX bit is the received *request to echo*. In two-terminal applications, this bit is typically the same as the permissive bit PILOTRX, but in applications to multiterminal lines, this bit must be separated from the PILOTRX bit, as explained and illustrated later. The ECHORX bit can be common for the open breaker echo and the weak-infeed echo. Timer PU1 adds security against noise in the protection channel and a momentary spurious assertion of the PILOTF condition in the remote relay.

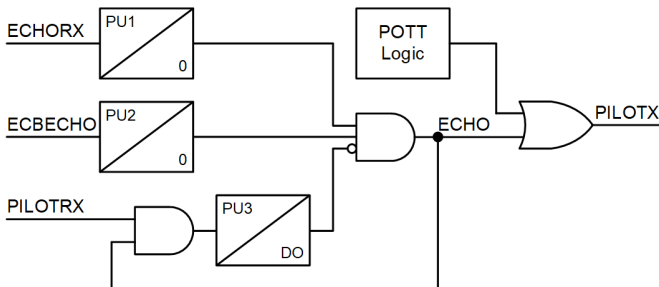


Fig. 17. Open breaker echo logic

The ECBECHO bit defines a condition when an open switch (open breaker or stub-bus condition) prevents the PILOTF protection elements from detecting line faults. Timer PU2 adds security in detecting the open switch condition and priming the open breaker echo logic to echo back.

The ECHO bit is added to the permissive bit PILOTRX and sent out to the remote relays.

In applications to multiterminal lines, more than one relay may have an echo function enabled. In order to prevent echo lockup (i.e., the two relays echoing each other indefinitely), Timer PU3 shuts down the echo signal once the relay simultaneously sends the echo (ECHO asserted) and receives permission from all other terminals (PILOTRX asserted). The Timer PU3 delay on the order of 100 ms and the associated AND gate establish positive proof that all relays in the scheme received the permissive signals, regardless of whether these signals originated from the PILOTF protection elements or as a result of an echo.

Often, the ECHORX and ECBECHO conditions are programmable logic equations that allow additional supervision and customization.

The echo request input, ECHORX, allows flexible applications to multiterminal lines. The permissive signal PILOTRX follows the “all remote relays detected a forward fault” logic, and therefore is an AND combination of the received permissive signals (see Fig. 5). The echo request signal, ECHORX, follows the “any remote relay detected a forward fault” logic, and therefore is an OR combination of the received permissive signals (see Fig. 18). However, depending on the application and security preferences, the echo operation can be restricted by programming the ECHORX equation differently in each relay. For example, if the breaker at Terminal 1 is always closed when the three-terminal line is energized, Terminal 2 and Terminal 3 echo may be allowed only if they receive the permissive signal from Terminal 1 and prevent them from sending an echo in response to the permissive signal from the other terminal (see Fig. 19).

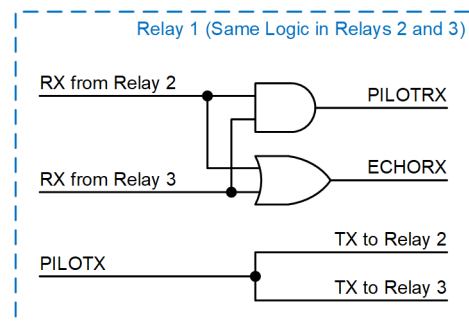


Fig. 18. Three-terminal POTT application with symmetrical open breaker echo logic

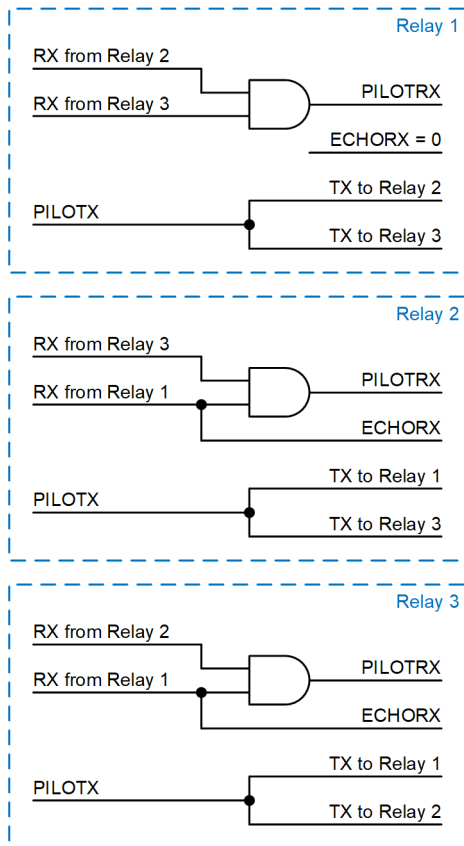


Fig. 19. Three-terminal POTT application with asymmetrical open breaker logic (only Terminal 1 is allowed to request an echo from Terminal 2 and Terminal 3)

By having an explicit request to echo input (ECHORX) that is separate from the pilot bit received input (PILOTRX), Implementation [2] allows using a separate pilot bit for requesting an echo. For example, an echo may only be requested when the ground or phase distance element asserts and refrains from requesting an echo signal when the sensitive sequence element(s) asserts. This solution increases security and can be programmed by using similar logic to that shown in Fig. 16.

Single-pole tripping and reclosing applications are concerned with a partially open breaker when reclosing after a single-pole trip. Such applications can either use the switch-onto-fault logic or a pilot scheme for instantaneous tripping when reclosing on a permanent fault. *Section VII* provides more details.

V. WEAK-INFEED LOGIC

The weak-infeed challenge to permissive pilot schemes is similar to that of an open circuit breaker. Instead of an open breaker preventing *any* current flow toward the line fault, a weak-infeed condition is characterized by a current flow that is *insufficient* to allow the PILOTF elements to dependably assert for a line fault. As explained in *Section III*, a weak-infeed scenario is as much a function of the local system short-circuit capacity as the type and settings of the protection elements used in the PILOTF condition. Application of incremental-quantity (TD32) and ground (32G) directional elements alleviates the

weak-infeed scenario. These elements are very sensitive, and they operate even if the fault current consists of only the zero-sequence component that flows between the fault and the grounding points of transformers, autotransformers, or shunt reactors.

The weak-infeed logic operates as follows:

- A reverse fault would draw fault current from the remote strong terminal(s) and would, therefore, activate the reverse-looking protection elements in the PILOTR condition.
- An internal fault would impact voltage at the weak terminal and cause phase undervoltage and sequence overvoltage conditions at the weak terminal.
- If the relay receives the request to echo (ECHORX) and the voltage is affected, then there is a fault present. If, additionally, the PILOTR condition is not asserted, the fault must be internal, and the weak terminal relay should echo the received permission, allowing the strong terminal relay(s) to trip.
- Optionally, the weak terminal relay trips the local breaker after sending the weak-infeed echo.

Fig. 20 shows a simplified weak-infeed echo logic similar to the open breaker logic of Fig. 17. The UV condition signifies abnormal voltage and typically comprises the phase undervoltage and the zero- and negative-sequence overvoltage protection elements.

The weak-infeed logic, despite being a part of a permissive pilot scheme, inherits some characteristics of a blocking scheme. When the weak-infeed logic is enabled, the scheme security depends on the ability of the PILOTR condition to detect all external faults for which the remote PILOTF condition asserts. Therefore, a best practice is to strictly follow the setting rules for the PILOTR protection elements for line terminals where the weak-infeed logic is enabled, as described in the *Setting PILOTR Protection Elements* subsection.

Using the voltage signal to supervise the weak-infeed echo brings additional security. It does not penalize the pilot scheme speed or dependability because forward faults near weak terminals significantly depress the voltage.

Very weak terminals (e.g., an ultimate case of no other power system elements connected to the bus but the protected line) may challenge the loss-of-potential (LOP) logic. A line fault will change the voltage while the current remains zero.

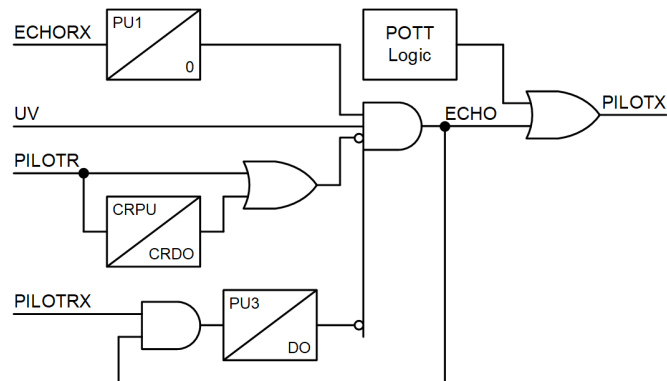


Fig. 20. Weak-infeed echo logic

Some LOP designs are immune to this scenario—they require the current to be at the level that ensures detecting a change in current for an LOP condition before allowing the LOP logic to operate [2]. In general, the weak-infeed logic designs tend to allow the weak-infeed echo before accepting the LOP signal and blocking the weak-infeed logic.

Some applications use the weak-infeed logic—instead of making the PILOTF protection elements more sensitive—to detect high-resistance faults. This is a valid approach. However, it is not much different than using a blocking scheme. It shares the main disadvantage of relying on the blocking condition, PILOTR, for security. If the logic is applied at the strong terminal, a high-resistance fault would not depress the voltage, and the UV condition would have to be set very sensitively or be disabled. The security of a true permissive scheme should not be expected if the weak-infeed logic is enabled at all line terminals. A permissive scheme with the weak-infeed logic enabled as a rule, regardless of the system strength, is a crossover scheme with many similarities to the blocking scheme.

Having the request to echo condition, ECHORX, as a programmable equation, allows more deliberate application of the weak-infeed logic. For example, in applications to multiterminal lines, the ECHORX condition may be programmed to echo only in response to a permissive signal(s) that came from a strong terminal(s). Also, an additional pilot bit can be used and the weak-infeed echo can be requested only when very secure protection elements—such as a distance Zone 2 element—detect a line fault.

VI. ADDRESSING CHANNEL FAILURES

A hidden protection channel failure challenges the dependability of a permissive scheme and the security of a blocking scheme. Channel monitoring is a preferred way to improve the channel uptime and gradually eliminate hidden failures. Channel monitoring allows detecting and addressing issues such as failing equipment, marginal receive power, incorrect network configuration, and so on. Channel monitoring is commonly available in today’s digital networks but is also available in the analog power line carrier sets (e.g., the guard signal in a frequency shift keying scheme).

A temporary protection channel failure that coincides with a line fault is a concern when the channel media shares the same right-of-way as the protected line (e.g., power line carrier, optical fiber in the ground wires on the same towers as the protected line, and radio signal with the line-of-sight at least partially overlapping with the protected line right-of-way).

A blocking pilot scheme may lose security during channel issues for external faults. Typical DCB scheme implementations use integrating timer logic for the DCB coordination timer and/or other ride-through methods to address a temporary dropout of the blocking pilot bit during external faults (historically called *carrier holes*). On the other hand, blocking schemes are dependable for internal faults coinciding with channel issues. In addition, blocking schemes perform well for external faults with channels that share the right-of-way with the protected line. These observations were

the key driving force for adoption of DCB schemes with power line carrier sets applied as protection channels.

A permissive pilot scheme may lose dependability during channel issues that occur for the same reasons as a line fault or due to a line fault. The key assumption is that when the channel is lost, it is much more likely that an internal fault caused the channel failure than an external fault or some other event. The scheme based on this assumption is historically referred to as a “directional comparison unblocking scheme.” Many practitioners today find this name misleading because it uses the word *blocking* in relation to a permissive scheme, and it mentions *unblocking* with no prior blocking action present. To avoid this confusion, this paper uses the terms “channel failure key” (CFK) and “channel failure echo” (CFE). The distinction between the CFK and echo logic is similar to the distinction between open breaker key and echo logic.

The CFK logic keys the permissive signal for a short time period, on the order of 100 to 200 ms upon the loss of signal (LOS), as Fig. 21 shows. The CFE logic opens a short time window on the order of 100 to 200 ms upon LOS, and echoes back the permissive signal if that signal asserts, as Fig. 22 shows. The CFE logic can apply an additional time delay to the PILOTX signal before connecting it to the AND gate and allowing it to echo the signal back to the relay.

The CFE logic makes the sequence of events record easier to analyze: if the channel fails when there is no fault, no permissive signal received is recorded in the sequence of events record.

The CFK and CFE logic can reside in the communications equipment or in the relay. Placing the logic in the relay provides more options. For example, when using redundant channels, the logic may issue the CFK or echo only when both channels are lost (i.e., when the second channel fails while the first channel is lost, as Fig. 23 shows). When using an echo function, it is a good practice not to send the echo based on the CFE signal but only based on the permissive signal truly received (see Fig. 24).

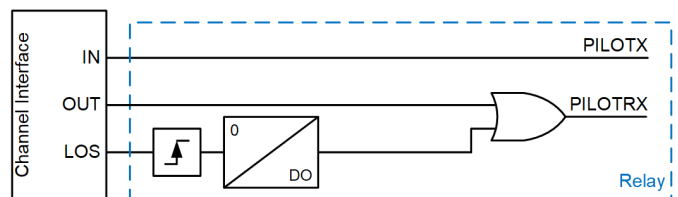


Fig. 21. Channel failure key (CFK) logic (also referred to as DCUB)

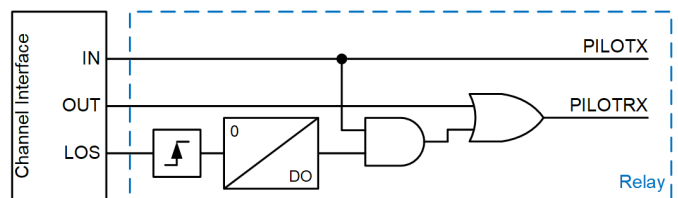


Fig. 22. Channel failure echo (CFE) logic

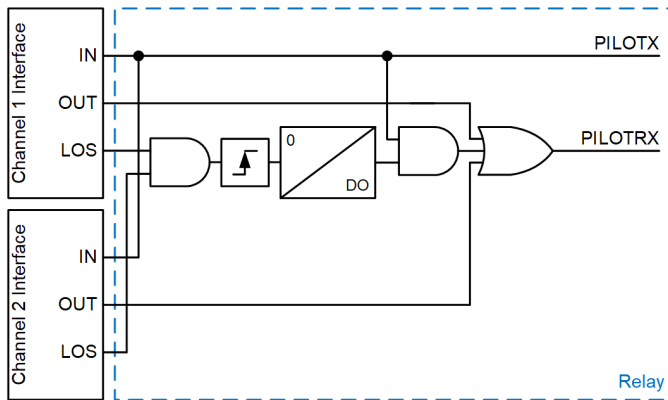


Fig. 23. CFE logic applied with a redundant channel

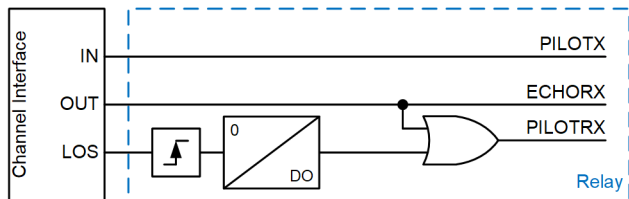


Fig. 24. CFK logic applied with the weak-infeed echo logic

VII. SINGLE-POLE TRIPPING CONSIDERATIONS

Single-pole tripping and reclosing applications require the protection elements and schemes that initiate instantaneous tripping for line faults to be able to identify the fault type. This phase selectivity is inherent in some protection elements (e.g., distance elements). Other elements (e.g., the negative-sequence directional overcurrent element in the pilot protection logic or negative-sequence overvoltage element in the weak-infeed logic) require an explicit fault-type identification logic.

In the context of a pilot protection scheme, it is convenient to distinguish between the local fault-type identification and the remote fault-type identification. A relay may issue a trip signal based on a pilot protection scheme (either permissive or blocking) and use the local fault-type identification to decide which pole(s) to trip. From this perspective, single-pole tripping does not limit applicability of any type of pilot scheme. A pilot protection scheme performs single-pole tripping as a two-step process. The first step is to detect if the fault is internal to the protected line. This decision is based on fault direction measured at both (all) terminals of the line and shared using a single—permissive or blocking—pilot bit. The second step is to identify the fault type in order to trip the correct circuit breaker poles. This latter fault-type identification step can be done based on local currents and voltages, independently of the pilot trip decision.

However, using both local and remote fault-type identification improves selectivity of single-pole tripping for cross-country and external-to-internal evolving faults. Fig. 25 shows an example of a cross-country fault. An AG fault occurs on one transmission line near-simultaneously with a BG fault on the other transmission line. It is desirable that both lines are tripped in a single-pole fashion. Three-pole tripping of both lines would entirely stop the power flow and defeat the purpose of single-pole tripping: maintaining partial power transfer when isolating temporary single-line-to-ground faults.

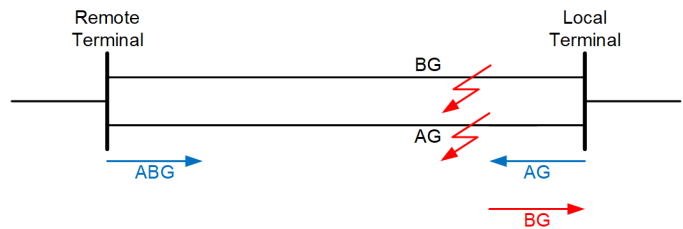


Fig. 25. Cross-country fault example

The two faults in Fig. 25 blend when seen from the remote terminal, and the remote relay will (correctly) identify the fault type as ABG. Note that both relays at the remote terminal (the top line and the bottom line) will identify the fault type as ABG and will be inclined to trip both lines in all three poles. The situation can be resolved by using distance elements at the local terminal. Distance elements are both directional and phase-selective. This allows the local relay protecting the bottom line to simultaneously identify the AG fault as a forward fault and the BG fault as a reverse fault. The relay protecting the top line would identify the AG fault as a reverse fault and the BG fault as a forward fault.

To trip the bottom-line breaker Pole A only and the top line breaker Pole B only, the local and remote relays share their identified fault types and apply a selection logic that optimizes selectivity. For example, a combination of the AG and ABG fault types may result in tripping Pole A only; a combination of the BG and ABG fault types may result in tripping Pole B only.

Pilot protection schemes that provide enhanced single-pole tripping selectivity use multiple pilot bits to convey both the fault direction and the fault type and implement the current reversal logic on a per-phase basis. Fig. 26 shows an example of a pilot scheme functional block with fault-type pilot bits PILOTXA, PILOTXB, and PILOTXC. The scheme retains the main pilot bit, PILOTX, and treats the fault-type bits as optional.

The fault-type bits are permissive by nature. They assert for forward faults. Therefore, it is commonly assumed that only permissive schemes are suitable for phase-segregated keying and that single-pole tripping and reclosing can only be done with permissive schemes. While permissive schemes are indeed more secure, the logic in Fig. 26 applies to both permissive and blocking schemes.

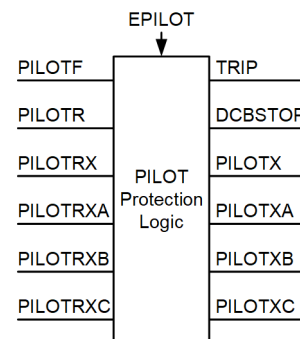


Fig. 26. PILOT logic with phase-selective (fault-type) pilot bits

When configured to be a blocking scheme, the logic trips if it does not receive the blocking signal on the PILOTRX input. If, at the same time, the PILOTRXA, PILOTRXB, and PILOTRXC input bits are present, the logic makes use of both the local and remote fault-type identification. If these phase-segregated bits are not available, the blocking logic selects the breaker pole(s) to trip based on the local fault-type identification only. This solution makes the multi-bit blocking scheme a crossover scheme: it retains dependability of a blocking scheme by using the blocking pilot bit, and it improves single-pole tripping selectivity by using three permissive fault-type identification bits.

Fig. 27(a) shows the basic connection of a multi-bit permissive scheme, and Fig. 27(b) shows the basic connection of a multi-bit blocking scheme. In the era of multi-bit digital protection channels, dedicating three or four bits to the pilot protection scheme is affordable.

The trip matrix that combines the local and remote fault-type identification is subject to a compromise between dependability and selectivity. For example, if one terminal identifies the AG fault type and the other the ABG fault type, it can be assumed that there is a Phase-A-to-ground fault present on the protected line and trip the Pole A of the breaker. However, what is the optimum trip decision if one relay identifies an AG fault and the other identifies a BCG fault? Tripping all three poles maintains dependability. Restraining tripping (no common phase selected) would increase selectivity, but it would also violate the basic premise of a pilot protection scheme: if both relays see a forward fault, the scheme must operate.

Implementation [2] is biased toward dependability: it trips the phase that is common in both local and remote fault-type identifications, and it trips three poles for fault-type combinations that have no common phase. Fig. 28 shows how to program Implementation [2] to bias it for single-pole tripping selectivity at the expense of dependability. The scheme in Fig. 28 accepts the permissive signal only if the remote and local phase-selection bits agree.

As explained earlier, single-pole tripping is possible when using a blocking scheme. Single-pole tripping is also possible when using weak-infeed logic. In this application, the fault-type identification logic must use voltages in order to identify the fault type when the current is very low or when the current contains only the zero-sequence component.

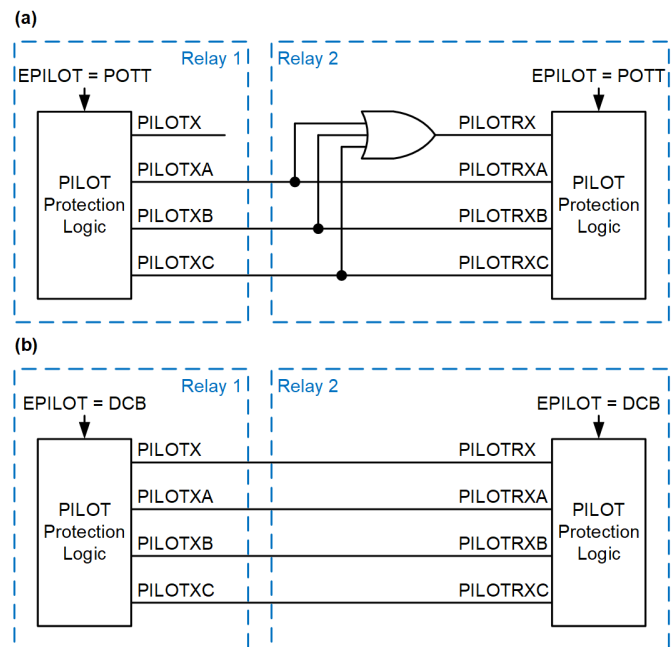


Fig. 27. Pilot scheme connection: (a) basic multi-bit permissive (b) blocking

Implementation [2] uses angle relationships between the composite zero- and negative-sequence signals for fast, accurate, and dependable fault-type identification in the pilot protection scheme. The composite signal combines both voltage and current.

When reclosing after a single-pole trip, the line protection system can apply either the pilot logic or the switch-onto-fault logic for instantaneous and dependable clearance of permanent faults. Because two conductors are energized during the single-pole autoreclose dead time, the distance elements receive polarizing voltage and are fully dependable. This favors application of a pilot scheme over the switch-onto-fault logic for detecting permanent faults when reclosing. The permissive scheme requires the open breaker echo logic enabled at the follower breaker terminal. In order to enhance security for external faults that may occur during the autoreclose dead time, a single-phase version of the open breaker logic is typically used. The logic echoes the phase-selective bit only when the breaker is opened in the corresponding phase. Fig. 29 shows a sample implementation of the single-phase open breaker logic (OPA, OPB, and OPC are open-pole bits for Phases A, B, and C, respectively).

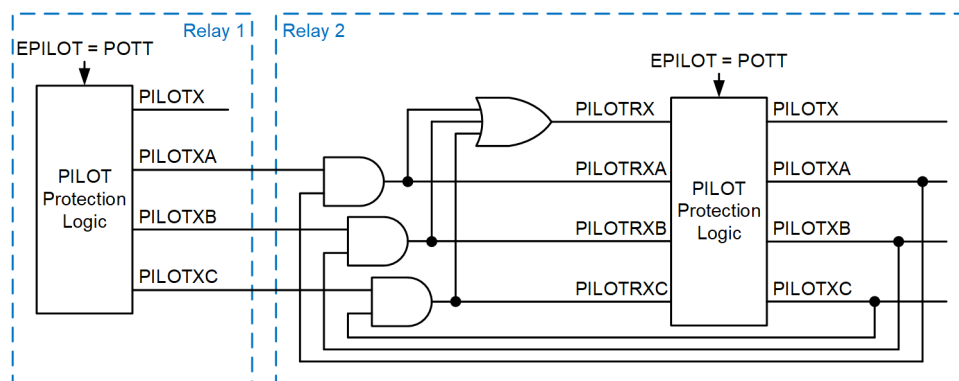


Fig. 28. Biasing the multi-bit POTT scheme for ultimate single-pole tripping selectivity

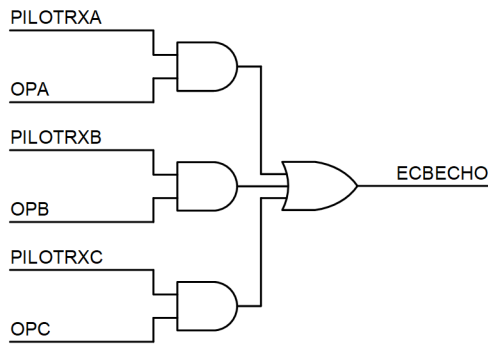


Fig. 29. Single-phase open breaker echo logic example

VIII. CROSSOVER PERMISSIVE AND BLOCKING LOGIC

So far, this paper has given numerous examples of customizing the classical permissive (Fig. 1) and blocking (Fig. 2) schemes to alleviate their weaknesses while benefiting from their inherent advantages. It is fair to say that practical schemes are crossover schemes with elements of both permissive and blocking logic. Permissive and blocking schemes only truly differ in the following ways:

- Response to equipment failures (permissive schemes retain security at the expense of dependability, and blocking schemes retain dependability at the expense of security).
- Degree and complexity of engineering (permissive schemes need add-ons for dependability, and blocking schemes need very carefully set blocking conditions).

With multi-bit digital channels widely available today, a crossover scheme that eliminates the need for engineering required by a permissive scheme and retains the dependability of a blocking scheme can be used. Additionally, the scheme described next improves the trip time compared with the blocking scheme.

Fig. 30 shows the crossover scheme in a three-pole tripping application. To implement this scheme, both relays are enabled as a DCB scheme (EPILOT = DCB) and two pilot bits are used. The primary pilot bit (PILOTX) is a blocking bit and is connected for the DCB scheme. The PILOTF and PILOTR settings need to be selected for a DCB scheme (i.e., the importance of the blocking condition and the coordination timer selected for the slowest channel time plus margin). The second pilot bit is a permissive bit (i.e., DCBSTOP; the DCBSTOP bit asserts when the relay detects a forward fault). In the receiving

relay, the permissive pilot bit is AND-ed with the DCBSTOP signal to effectively obtain a POTT logic. The resulting bit (XPOTT in Fig. 30) is programmed into the trip equation together with the DCB bit.

The foundation of the crossover scheme is simple: the scheme is selective by either receiving an explicit permissive bit or not receiving a blocking bit within a predetermined time. The permissive bit improves speed. The blocking bit ensures dependability and adds simplicity. The scheme can also be understood as a blocking scheme with permissive acceleration.

The scheme responds to faults and failure conditions as follows:

- During external reverse faults, the local relay asserts the blocking bit and continues to keep the permissive bit deasserted.
- During internal faults, the relays continue to keep the blocking bit deasserted and they assert the permissive bit. Once the permissive bit is received, the logic trips the circuit breaker without having to wait for the coordination timer to expire.
- If the relay or the channel have failed prior to the fault, the scheme trips based on the blocking logic.
- If a relay fails to detect a line fault due to settings errors or sensitivity limitations, the other relay(s) trips based on the blocking logic.

The scheme does not need weak-infeed logic, open breaker echo logic, or channel failure logic. Having characteristics of a blocking logic, the scheme requires the PILOTR blocking condition to be set carefully.

The benefit of faster operation is twofold. First, fast trip times are obtained, especially when using fast directional elements, such as the TD32 incremental-quantity element. The permissive part operates immediately after the permissive bit is received. The blocking scheme must wait to account for the slowest possible channel time and slowest assumed assertion of the blocking PILOTR condition in the other relay(s). The difference between the normal (fast) channel time and the slowest possible channel time can be significant when using redundant channels or switching communications paths or media. Second, benefiting from fast operation due to the permissive part of the logic, users can apply more generous coordination time in the blocking part of the logic, simplifying engineering and increasing security.

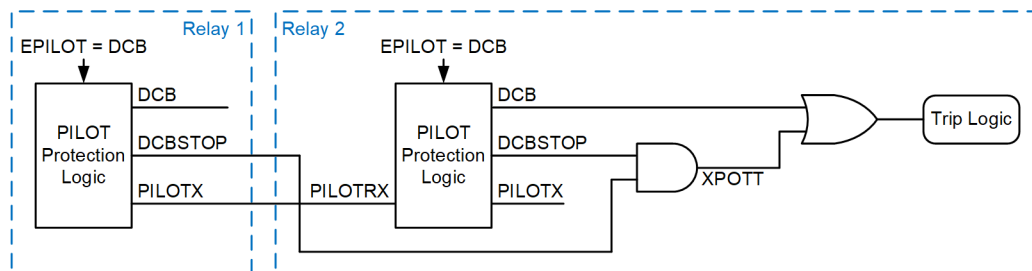


Fig. 30. Crossover permissive-blocking pilot scheme

IX. CONCLUSION

This paper reviews permissive and blocking pilot schemes for protection of transmission lines. It covers principles of operation, settings considerations, importance of coordinating the forward and the reverse fault detectors, applications to multiterminal lines, and single-pole tripping. It also describes add-ons for the dependability of permissive schemes: open breaker echo, weak-infeed logic, and channel failure logic.

The paper follows Implementation [2], in which a common PILOT scheme is available and can be enabled as a permissive logic or a blocking logic. Implementation [2] uses carefully selected pilot input and output bits that are available as programmable logical equations. By using the programmable equations, the user can easily customize the scheme and apply test bits, implement redundant channel logic, address channel failure conditions, and achieve a desired balance between dependability and single-pole tripping selectivity.

Finally, the paper introduces a crossover permissive-blocking scheme that eliminates the need for the additional engineering required by most permissive scheme applications, while allowing fast tripping without coordination time. The scheme is easy to configure and test, and it uses two pilot bits in a multi-bit digital protection channel.

X. REFERENCES

- [1] I. Stevens, N. Fischer, and B. Kasztenny, "Performance Issues With Directional Comparison Blocking Schemes," proceedings of the Texas A&M Conference for Protective Relay Engineers, College Station, TX, March 2010.
- [2] *SEL-T401L Ultra-High-Speed Line Relay Instruction Manual*. Available: selinc.com.
- [3] E. O. Schweitzer, III, B. Kasztenny, A. Guzmán, V. Skendzic, and M. V. Mynam, "Speed of Line Protection – Can We Break Free of Phasor Limitations?" proceedings of the 68th Annual Conference for Protective Relay Engineers, College Station, TX, March 2015.
- [4] B. Kasztenny, M. V. Mynam, and N. Fischer, "Sequence Component Applications in Protective Relays – Advantages, Limitations, and Solutions," proceedings of the 72nd Annual Conference for Protective Relay Engineers, College Station, TX, March 2019.
- [5] R. Abboud, J. Bell, and B. Smyth, "Considerations and Benefits of Using Five Zones for Distance Protection," proceedings of the 72nd Annual Georgia Tech Protective Relaying Conference, Atlanta GA, May 2018.
- [6] Y. Xue, B. Kasztenny, D. Taylor, and Y. Zia, "Series Compensation, Power Swings, and Inverter-Based Sources and Their Impact on Line Current Differential Protection," proceedings of the 66th Annual Conference for Protective Relay Engineers, College Station, TX, April 2013.
- [7] E. O. Schweitzer, III, B. Kasztenny, and M. V. Mynam, "Performance of Time-Domain Line Protection Elements on Real-World Faults," proceedings of the 42nd Annual Western Protective Relay Conference, Spokane, WA, October 2015.
- [8] A. Guzmán, M. V. Mynam, V. Skendzic, J. L. Eternod, and R. M. Morales, "Traveling-Wave and Incremental Quantity Directional Elements Speed Up Directional Comparison Protection Schemes," proceedings of the 14th International Conference on Developments in Power System Protection, Belfast, UK, March 2018.
- [9] A. Guzmán, M. V. Mynam, V. Skendzic, J. L. Eternod, and R. M. Morales, "Directional Elements – How Fast Can They Be?" proceedings of the 44th Annual Protective Relay Conference, Spokane, WA, October 2017.

XI. BIOGRAPHIES

Bogdan Kasztenny has specialized and worked in power system protection and control since 1989. In his decade-long academic career, Dr. Kasztenny taught power system and signal processing courses at several universities and conducted applied research for several relay manufacturers. Since 1999, Bogdan has designed, applied, and supported protection, control, and fault locating products with their global installed base counted in thousands of installations. Since 2009, Bogdan has been with Schweitzer Engineering Laboratories, Inc. where he works on product research and development. Bogdan is an IEEE Fellow, a Senior Fulbright Fellow, a Canadian representative of the CIGRE Study Committee B5, and a registered professional engineer in the province of Ontario. Bogdan has served on the Western Protective Relay Conference Program Committee since 2011 and on the Developments in Power System Protection Conference Program Committee since 2015. Bogdan has authored over 220 technical papers and holds over 50 patents.

Mangapathirao (Venkat) Mynam received his MSEE from the University of Idaho in 2003 and his BE in electrical and electronics engineering from Andhra University College of Engineering, India, in 2000. He joined Schweitzer Engineering Laboratories, Inc. (SEL) in 2003 as an associate protection engineer in the engineering services division. He is presently working as a principal research engineer in SEL research and development. He was selected to participate in the U. S. National Academy of Engineering (NAE) 15th Annual U. S. Frontiers of Engineering Symposium. He is a senior member of IEEE and holds patents in the areas of power system protection, control, and fault location.

Normann Fischer received a Higher Diploma in Technology, with honors, from Technikon Witwatersrand, Johannesburg, South Africa, in 1988; a BSEE, with honors, from the University of Cape Town in 1993; an MSEE from the University of Idaho in 2005; and a PhD from the University of Idaho in 2014. He joined Eskom as a protection technician in 1984 and was a senior design engineer in the Eskom protection design department for three years. He then joined IST Energy as a senior design engineer in 1996. In 1999, Normann joined Schweitzer Engineering Laboratories, Inc., where he is currently a distinguished engineer in the Research and Development division. He was a registered professional engineer in South Africa and a member of the South African Institute of Electrical Engineers. He is currently a senior member of IEEE and a member of the American Society for Engineering Education (ASEE). Normann has authored over 60 technical and 10 transaction papers and holds over 20 patents related to electrical engineering and power system protection.

Armando Guzmán M '95, SM '01) received his BSEE with honors from Guadalajara Autonomous University (UAG), Mexico. He received a diploma in fiber-optics engineering from Monterrey Institute of Technology and Advanced Studies (ITESM), Mexico, and his masters of science and PhD in electrical engineering and masters in computer engineering from the University of Idaho, USA. He served as regional supervisor of the Protection Department in the Western Transmission Region of the Federal Electricity Commission (the electrical utility company of Mexico) in Guadalajara, Mexico for 13 years. He lectured at UAG and the University of Idaho in power system protection and power system stability. Since 1993 he has been with Schweitzer Engineering Laboratories, Inc. in Pullman, Washington, where he is a distinguished engineer. He holds numerous patents in power system protection and metering. He is a senior member of IEEE.