Chapter 8
Diagnosis in Cyber-Physical Systems with Fault Protection Assemblies

Ajay Chhokra, Abhishek Dubey, Nagabhushan Mahadevan, Saqib Hasan, and Gabor Karsai

8.1 Introduction

The Smart Electric Grid is a CPS: it consists of networks of physical components (including generation and transmission subsystems) interacting with cyber components (e.g., intelligent sensors, communication networks, computational and control software). Reliable operation of such CPS is critical. Therefore, these systems are equipped with specialized protection devices that remove the faulty component from the system. However, if there are failures in the fault protection units, this leads to a situation where an incorrect local mitigation in a subsystem results in a larger fault cascade, leading to a blackout. This phenomenon was observed in the recent blackouts [1], where tripping of some lines by the relays (protection devices) overloaded some other parts of the system. These secondary overloaded components were again isolated by pre-defined protection schemes, leading to tertiary effects and so on. This domino effect got disseminated into the whole system, pushing it towards total collapse.

The ultimate challenge in doing fault diagnosis in these cyber-physical systems is to handle the complexity: the sheer size, large number of components, anomalies, and failure modes. Furthermore, the subsystems are often heterogenous and the typical approach is to try and understand the interactions among them, even if the subsystems are from different domains. In the past, we have used the high-level concept to model the interaction between the subsystems—(1) observable degradations,

A. Chhokra (✉) · A. Dubey · N. Mahadevan · S. Hasan · G. Karsai
Institute for Software Integrated Systems, Vanderbilt University, Nashville, 37212, TN, USA
e-mail: ajay.d.chhokra@vanderbilt.edu; abhishek.dubey@vanderbilt.edu; nag.mahadevan@Vanderbilt.Edu; saqib.hasan@vanderbilt.edu; gabor.karsai@Vanderbilt.Edu

M. Sayed-Mouchaweh (ed.), *Diagnosability, Security and Safety of Hybrid Dynamic and Cyber-Physical Systems*, https://doi.org/10.1007/978-3-319-74962-4_8

anomalies, discrepancies caused by failure modes, (2) their propagation, and (3) their temporal evolution towards system-level fault (effects). This approach called Timed Fault Propagation Graphs (TFPG) has been applied to avionics systems, fuel assemblies, and software component assemblies [2, 3] and is based on a discrete-event model that captures the causal and temporal relationships between failure modes (causes) and discrepancies (effects) in a system, thereby modeling the failure cascades while taking into account propagation constraints imposed by operating modes and timing delays. In this graphical model, nodes represent failure modes and discrepancies, edges represent the direction of causality, and attributes of edges capture the conditions (mode and temporal delays) under which the edge is active. The model-based fault diagnostics reasoner receives observations in the form of time-stamped alarms that indicate whether a discrepancy is present and, using abductive reasoning, generates a set of hypotheses about the failure modes that could explain the observed fault signature, i.e. the fault effects.

However, the approach of failure diagnosis with timed fault propagation graphs does not deal with the built-in automatic fault-protection mechanisms of the system. Such local fault protection components are designed to mask the effect of failures and thereby arrest the fault cascades. Additionally, these fault protection components introduce failure modes that are specific to the operation or lack of operation of the protection components. A classical TFPG model is not well suited for capturing the specializations that are introduced by the inherent fault protection mechanisms built into the system. For example, in power systems, there is already a fault-detection/protection system (relays and breakers) that autonomously protects elements of the network. Any protection operation performed by these systems can fall into one of these categories: (a) correct and thereby isolate the area where the fault occurred, (b) incorrect: fires incorrectly when it is not supposed to, (c) backup: accounting for lack of firing of another protection system, or (d) consequence of a previous firing which was incorrect when considering its effect on the global or regional system stability. In effect, the failure can be introduced by the physical components of the power system (e.g. cables) as well as components of the fault-protection system (e.g., breakers/sensors). Furthermore, the autonomous fault protection mechanism changes the network topology automatically (i.e., changes the mode of the system).

To solve this problem, we have developed an extension of TFPG called Temporal Causal Diagrams (TCDs). A TCD model is a behavioral augmentation of Temporal Fault Propagation Graphs (TFPGs) that can efficiently model fault propagation in various domains. The TCD-based diagnosis system is hierarchical. The lower level uses local discrete event diagnosers, called *Observers*, which are generated from the behavior specification of fault management controllers. A higher level reasoner produces system level hypotheses based upon the output of local observers. The approach does not involve complex real-time computations with high-fidelity models, but reasons using efficient graph algorithms to explain the observed anomalies. This approach is applicable to CPS that include supervisory controllers that arrest fault propagation based upon local information without considering system-wide effects. To explain TCD we use examples from power system domain.

The paper is organized as follows, Sect. 8.2 describes the background and literature review followed by brief explanation of cascade phenomenon caused by misoperation of fault management assemblies in power systems (Sect. 8.2.3). Section 8.3 gives an overview of our approach and describes the TCD modeling formalism and diagnosis methodology in detail. The fault diagnosis approach is described in the context of a power system example in Sect. 8.4, followed by concluding remarks in Sect. 8.5.

8.2 Background

8.2.1 Diagnosis in CPS

Diagnostic reasoning techniques share a common process in which the system is continuously monitored and the observed behavior is compared with the expected one to detect abnormal conditions. In many industrial systems, diagnosis is limited to signal monitoring and fault identification via threshold logic, e.g., detecting if a sensor reading deviates from its nominal value. Failure propagation is modeled by capturing the qualitative association between sensor signals in the system for a number of different fault scenarios. Typically, such associations correspond to relations used by human experts in detecting and isolating faults. This approach has been effectively used for many complex engineering systems. Common industrial diagnosis methods include fault trees [4–7], cause-consequence diagrams [8, 9], diagnosis dictionaries [10], and expert systems [11, 12].

Model-based diagnosis (see [13–15] and the references therein), on the other hand, compares observations from the real system with the predictions from a model. Analytical models such as state equations [16], finite state machines [17], hidden Markov models [18], and predicate/temporal logic [19] are used to describe the nominal system behavior. In the presence of a fault, the observed behavior of the system deviates from the nominal behavior expected by the model. The associated discrepancies can then be used to detect, isolate, and identify the fault depending on the type of model and methods used. In consistency-based diagnosis the behavior of the system is predicted using a nominal system model and then compared with observations of the actual behavior of the system to obtain the minimal set of faulty component that is consistent with the observations and the nominal model. Consistency-based diagnosis was introduced in a logical framework in [19] and was later extended in [20]. The approach has been applied to develop diagnosis algorithms for causal systems [21, 22] and temporal causal systems [23, 24].

The diagnosis approach presented here is conceptually related to the temporal causal network approach presented in [24]. However, we focus on incremental reasoning and diagnosis robustness with respect to sensor failures. The causal model presented in this paper is based on the timed failure propagation graph

(TFPG) introduced in [25, 26]. The TFPG model is closely related to fault models presented in [27–29] and used for an integrated fault diagnosis and process control system [30]. The TFPG model was extended in [31] to include mode dependency constraints on the propagation links, which can then be used to handle failure scenarios in hybrid and switching systems. TFPG modeling and reasoning tool has been developed and used successfully in an integrated fault diagnoses and process control system [30].

Additionally, the temporal aspects of the TFPG model are closely related to the domain theoretic notion of temporal dependency proposed in [32]. However, there are several major differences between the two approaches. In particular, TFPG-based diagnosis implements a real-time incremental reasoning approach that can handle multiple failures including sensor/alarm faults. In addition, the underlying TFPG model can represent a general form of temporal and logical dependency that directly incorporates the dynamics of multi-mode systems.

8.2.2 Diagnosis in Power Systems

Since power systems is our example domain, we now present a brief review of fault diagnosis approaches in that domain, which can be categorized into three main branches based on their underlying technique: expert systems [33–36], artificial neural networks [37–40], and analytical model based optimization [41–44]. In addition, approaches based on Petri nets [45] and cause-effect Bayesian networks [46–50] have also been proposed. Expert systems are one of the earliest techniques proposed to address the failure diagnosis problem in power systems. A comprehensive survey of such knowledge-based approaches is available in [51]. The expert systems, in general, suffer from limitation imposed due to the maintenance of the knowledge database and slow response time. Moreover, expert system based approaches are known to produce wrong hypothesis in presence of missing and/or spurious alarms. Artificial neural networks (ANNs) are adaptive systems inspired by biological systems. These approaches, in general, suffer from convergence problems. Further, the ANNs have to be retrained whenever there is a change in network topology as the weights are dependent upon the structure of the power system. A number of model-based analytical methods have been devised over the years for diagnosing failures by generating optimal failure hypotheses that best explain all the events and anomalies. However, these techniques rely heavily on critical and computationally expensive tasks such as the selection of an objective function, development of exact mathematical models for system actions and protective schemes, which greatly influence the accuracy of the failure diagnosis.

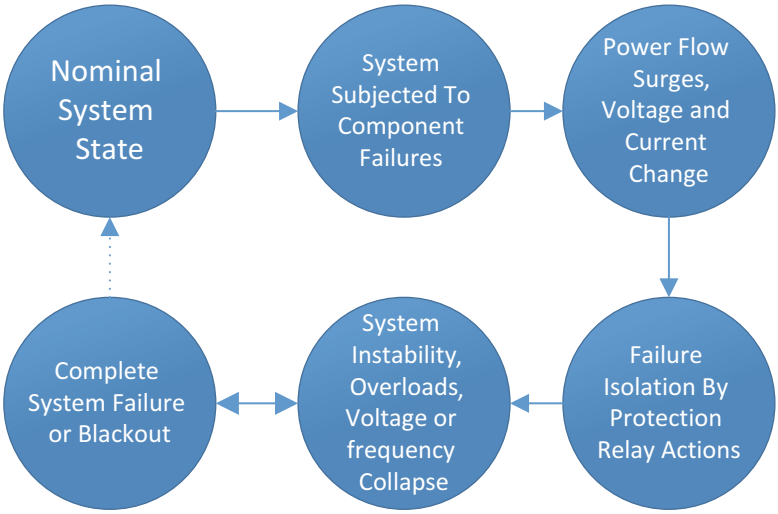

Fig. 8.1 Typical blackout progression in power systems

8.2.3 Cascade Phenomenon: When Fault Management Controllers Misoperate

Cascading failures in networked systems are defined as the set of independent events that trigger a sequence of dependent events. Such cascading failures in power grids successively weaken the system by increasing stress on other components and sometimes lead to major blackouts. According to North American Reliability Corporation (NERC), a cascading outage is defined as an uncontrolled loss of any system facilities or load, whether because of thermal overload, voltage collapse, or loss of synchronism, as a result of fault isolation.

Figure 8.1 shows a typical blackout scenario in power systems. The nominal system is subjected to failures from physical and cyber components. These failure modes change the voltage and current at different buses. Fast acting protection devices (relays) react to these changes based on predefined strategies. While these actions are intended to isolate the faulty components and arrest fault propagation, they could have unintended secondary effects such as branch overloads, voltage and/or frequency collapse that can cause instability in the system. A new set of protection elements react to arrest these secondary effects. These secondary actions may cause different tertiary effects and the cycle continues until the system reaches a blackout or there are no more consequences of protective actions.

A simple example of cascading phenomenon using a standard IEEE 14 bus system is shown in Fig. 8.2. It is a simple approximation of American electric power system as of 1960s [52, 53]. The system consists of 14 buses, 5 generators, 11 loads,

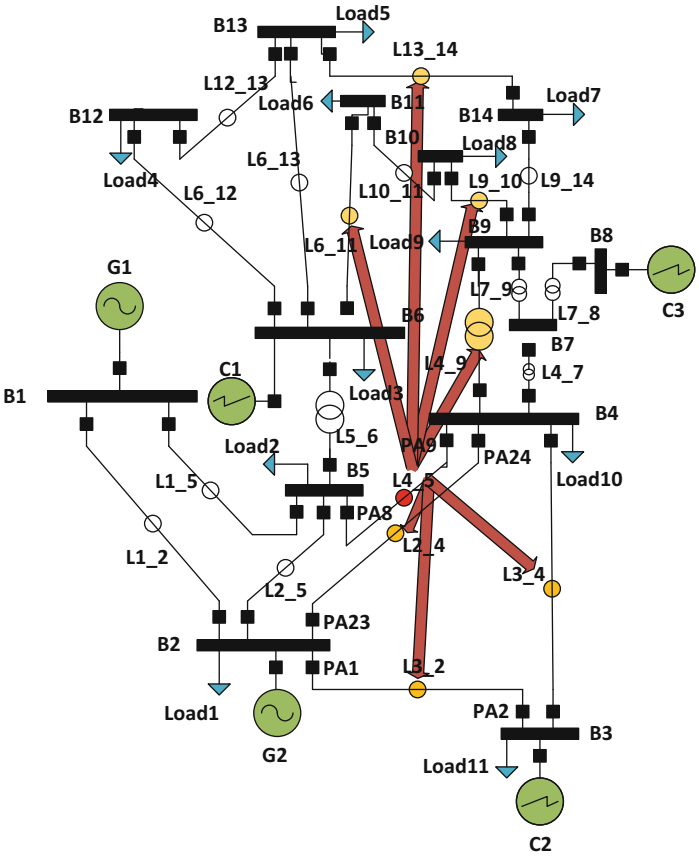

Fig. 8.2 Cascade progression in IEEE 14 Bus system with initial outage in *L*4_5 leading to outages in *L*3_4, *L*2_4, and *L*2_3 followed by outages in *L*6_11, *L*13_14, *L*9_10, *L*4_9, and ultimately leading to blackout

and 20 branches (transmission lines and transformers). Reference [52] provides bus and branch data in IEEE common data format [54] for creating OpenDSS [55] simulation models. A three phase to ground phase fault is injected in line, *L*4_5. The fault is isolated by tripping the line. This control actions of protection devices lead to overloading of lines *L*3_4, *L*2_4, and *L*2_3. These overloads are removed by tripping these lines. The removal of these secondary effects leads to overloads in lines *L*6_11, *L*13_14, *L*9_10, *L*4_9. The removal of these overloaded branches de-energizes more than 40% of the total system load and is considered as catastrophic event or blackout.

8.3 Temporal Causal Diagrams (TCD)

TCDs are discrete models that appropriately model failure modes, anomalies, and their propagation in both physical and cyber systems. TCD is a combination of Temporal Fault Propagation Graphs (TFPGs) and Time Triggered Automata (TTAs). TFPG based models and reasoning schemes have been used in the past to diagnose faults in physical systems including industrial plants [56, 57], aerospace systems [3], and software systems [58].

However, in cyber-physical systems, there are discrete controllers that try to arrest the failure effect if detected. These protection devices can cause system reconfiguration by instructing actuators to change their state. These devices can also have faults that alter their response to the detection of failure effects and control commands. TFPG based reasoning schemes are not very effective in accounting for faults in both physical system and their corresponding protection assembly (i.e., anomaly detectors, mode detectors, actuators). Failure diagnosis of protection devices is critical for cyber-physical systems, where realistic assessment of fault propagation is not possible without accounting for the behavior of the deployed sensors, controllers, and actuators. The second component of the TCD model, TTA is responsible for modeling the behavior of discrete components in both faulty and non-faulty modes.

TCD framework consists of hierarchical event-driven reasoning engines as shown in Fig. 8.3. The diagnosis system consists of multiple local diagnosers, called *Observers* that track the behavior of protection devices and estimate the presence of failures in both physical and cyber infrastructure (fault management controllers are often implemented in software). These estimates are then passed to a system level reasoner that creates system level hypotheses temporally consistent with the fault propagation graph. The observable events in the case of power transmission system are commands sent by relays to breakers, messages sent by relays to each other, state change of breakers, physical fault detection alarms, etc. The following sections describe the modeling formalism of TCD, which includes an extension to TFPG.

8.3.1 Extending TFPG with Non-deterministic Semantics

A temporal fault propagation graph is a labeled directed graph where nodes are either failure modes or discrepancies. Discrepancies are the failure effects, some of which may be observable. Edges in TFPG represent the causality of the fault propagation and edge labels capture operating modes in which the failure effect can propagate over the edge, as well as a time-interval by which the failure effect could be delayed (see Fig. 8.4). Classically, the diagnostic reasoner of TFPG assumed the correct knowledge of the system modes is always available. However, in the context of self-correcting cyber-physical systems such as power grids, the system mode or

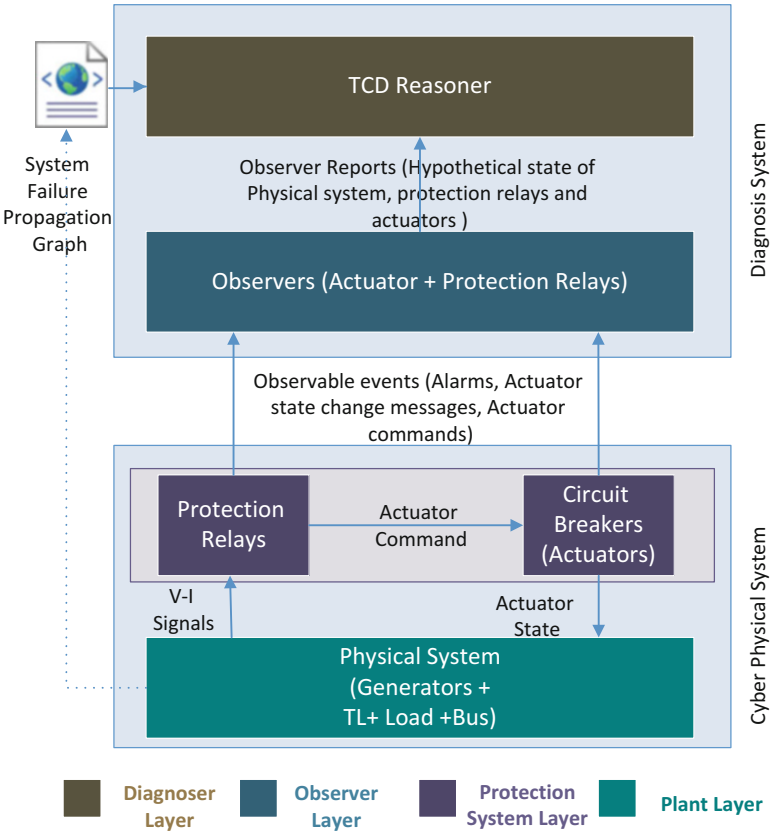

Fig. 8.3 The block-diagram of the Temporal Causal Diagram Diagnosis Framework in the Context of Power Systems

operating conditions depend upon the state of sources, sinks, and the topology of the system. Identification of all operating conditions, i.e. unique system modes is computationally very expensive. In this paper, we use the system topology dictated by the state of the actuators to map an operating condition (i.e., mode) to the fault propagation. However, while such a constraint imposed due to topology of the system is deemed necessary to identify when a fault will not propagate, it is not sufficient to state that the failures will propagate. So we need to extend the TFPG language with an additional map that associates uncertainty to failure edges.

Formally, the extended TFPG is represented as a tuple $\{F_{physical}, D_{physical}, E, M, ET, EM, ND\}$, where

- $F_{physical}$ is a nonempty set of fault nodes in physical system. A fault node can be in two states either present denoted by ON state or absent represented by OFF state. A fault node represents a failure mode of the system or a component, and

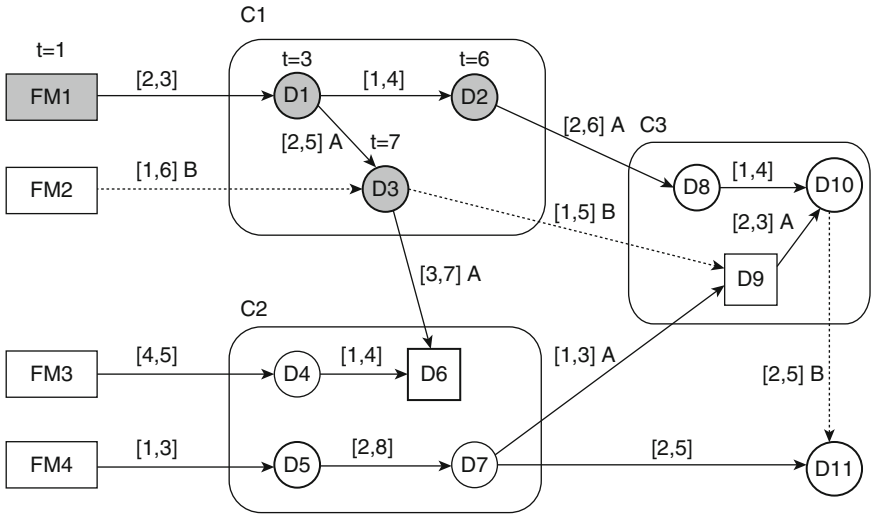

Fig. 8.4 TFPG Model with Failure Modes (FM), Discrepancies (D), and fault propagation links (edges). Labels on edges indicate delay (min,max) values and modal dependencies (letters)

its state represents whether the failure mode is present or not. In the subsequent discussion we will use the terms fault node and failure mode interchangeably.

- $D_{physical}$ is a nonempty set of discrepancy nodes related to fault effects of physical faults.
- $E \subseteq V \times V$ is a set of edges connecting the set of all nodes $V = F_{physical} \cup D_{physical}$.
- M is a nonempty set of system modes. At each time instance t the system can be in only one mode.
- $ET : E \rightarrow I$ is a map that associates every edge in E a time interval $[t_{\min}, t_{\max}] \in I$ that represents the minimum and maximum time for fault propagation over the edge.
- $EM : E \rightarrow M$ is a map that associates every edge in E with a set of modes in M when the edge is active. For any edge $e \in E$ that is not mode-dependent (i.e., active in all modes), $EM(e) = \varnothing$.
- $ND : E \rightarrow \{True, False\}$ is a map that associates an edge, $e \in E$ to *True* or *False*, where *True* implies the propagation along the edge, e **Will** happen, whereas *False* implies the propagation is uncertain and **Can** happen. The destination node of any uncertain edge is referred to as secondary discrepancy while primary discrepancy implies a certain edge. These labels are defined with respect to edges as same discrepancy can act as a destination node of both uncertain and certain edge.

8.3.2 Modeling the Behavior of Fault Management Controllers

The TCD framework relies on the use of an extended time triggered automaton [59] to model the interaction between the fault management controllers and the plant model (TFPG model). Then, given these behaviors we can synthesize the observers that are used in diagnosis step.

Mathematically, the extended time triggered automaton is represented as tuple $(\Sigma, Q, q_0, Q_m, F_{cyber}, D_{cyber}, \mathbb{M}, \alpha(F), \Phi, T)$.[1]

- **Event Set**: Σ is a finite set of events that consists of observable and unobservable events partitioned as $\Sigma = \Sigma_{obs} \cup \Sigma_{unobs}$ such that $\Sigma_{obs} \cap \Sigma_{unobs} = \phi$. Observable events are alarms, commands, and messages exchanged between discrete components, whereas unobservable events are related to introduction of faults in system components.
- **Locations**: Q is a finite set of locations. $q_0 \in Q$ is the initial location of the automaton and $Q_m \subset Q$ is a finite set of marked locations.
- **Discrepancy Set**: D_{cyber} is a finite set of discrepancies associated with the component behavior, partitioned into the sets of observable and unobservable discrepancies.
- **Failure Mode Set**: F_{cyber} is a finite set of unobservable failure modes associated with the component. Similar to a fault node in TFPG, failure mode also has ON and OFF states. δ_t is a function defined over $F_{cyber} \times \mathbb{R}_+$ that maps a failure mode $f \in F_{cyber}$ at time $t \in \mathbb{R}_+$ to *True* if the state of failure mode is ON and to *False* if the state is OFF.
- **Failure Mode Constraints**: $\alpha(F_{cyber})$ represents the set of all constraints defined over members of set F_{cyber}. An individual failure mode constraint, $\omega_t \in \alpha(F_{cyber})$, is a Boolean expression defined inductively as

$$\omega_t := \delta_t(f) \quad | \quad \neg\delta_t(f) \quad | \quad \omega_{1,t} \quad \wedge \quad \omega_{2,t} \tag{8.1}$$

  where $f \in F_{cyber}$ is a failure mode and ω_1, ω_2 are failure mode constraints. A failure mode constraint is True if the Boolean expression is evaluated to be True and False otherwise.
- **Timing Constraints**: Φ is a set of timing constraints defined as $\Phi = [n], (n) | n \in \mathbb{N}_+$, where $[n]$ denotes instantaneous constraints and (n) represents periodic constraints. The timing constraints specify a pattern of time points at which the automaton checks for events and failure node constraints. For instance, periodic constraint, (4), on any outgoing transition from the current state forces the automaton to periodically look for events specified by the edge, every 4 units of time whereas in the case of instantaneous constraint, [4], automaton checks only once.
- **Mode Map**: $\mathbb{M} : Q \rightarrow 2^m$ is a function that maps location $q \in Q$ to mode $m \in M$ defined in the fault propagation graph.

[1] The extension includes sets of failure modes and failure mode guards.

- **Edge**: $T \subset Q \times p(\Sigma) \times \Phi \times \alpha(F_{cyber}) \times p(\Sigma) \times Q$ is a finite set of edges. An edge represents a transition between any two locations. The activation conditions of an edge depend upon the timing, failure mode constraints, and an input event. For example, an edge $< q_1, \sigma_1, [n], \delta(f_1) \wedge \neg\delta(f_2), \sigma_2, q_2 >$ represents a transition from location q_1 to q_2 with an instantaneous time constraint of n units of time and failure mode constraint $\delta(f_1) \wedge \neg\delta(f_2) \in \alpha(F_{cyber})$ defined over the failure modes $f_1, f_2 \in F_{cyber}$. $\sigma_1 \in \Sigma$ is the required input event for this transition to be valid. $\sigma_2 \in \Sigma$ represents the event generated when the transition is taken. Syntactically, a transition is represented as *Event(timing constraint){failure constraint}/Event*. If no event is mentioned, then the transition is valid only if the failure mode constraint evaluates to true as per the timing constraints.

8.3.3 Observers for Postulating the Failures of Controllers

Observers are discrete, finite state machines that consume events produced by their respective tracked devices in order to diagnose faults in their behaviors. There exist a number of approaches for generating discrete diagnosers for dynamic systems based on [60] and [61]. However, the observers presented here are created manually. The events produced by the various observers fall into two categories; an estimation of a state change in discrete components, and a discrepancy detection. The detected anomalies and the local estimate of the state of different components in the plant and protection layer are passed by the observer to the next layer for system level diagnosis.

8.3.4 Combined Diagnosis and Reasoning Approach

The TCD reasoner relies on the fault propagation graph and the output of various observers to hypothesize about the anomalies observed in the system.[2] The reasoner attempts to explain the observations in terms of consistency relationship between the states of the nodes and edges in the fault propagation graph. The states of a node in a fault propagation graph can be categorized as *Physical* (Actual), *Observed*, and *Hypothetical* state [57].

- *Physical state* corresponds to the actual state of the nodes and edges.
- An *Observed state* is the same as the *Physical state*, but defined only for observable nodes.
- A *Hypothetical state* is an estimate of the node's physical state and the time since the last state change happened by the TCD reasoner.

[2]In order to relate to the alarms generated by observers with the failure graph few modifications are performed. The alarms signaled by relays are replaced by their corresponding observers.

Every reasoner hypothesis $h_f \in HSet_t$ consists of a map, $HNode_t$ that associates to every node in the failure graph an evaluation, (*ON*, *OFF*) and time estimate (t_1, t_2). The time estimate (t_1, t_2) denotes the earliest and latest time estimates for the state changes of node v, i.e. from ON to OFF or vice versa. The structure of a hypothesis is described as follows: Hypothesis is a tuple, where elements are related based on temporal consistency. Formally, hypothesis h_f=*{f, terl, tlat, S, C, I, M, E, U}* where:

- $f \in F_{physical}$ is a physical failure mode projected by the hypothesis, h_f and F is the set of physical failure modes defined in Sect. 8.3.1. We are using single physical fault hypothesis which lists only one fault per element of the physical system along with multiple faults in protection system.
- $S \subseteq F_{cyber}$ is a set of faults active in the system. These faults are related to components in the protection system layer as defined in Sect. 8.3.2.
- The interval $[terl, tlat]$ is the estimated earliest and the latest time during which the failure mode f could have been activated. The time estimate for protection layer faults is not supported in the current implementation.
- $C \subseteq D_{physical}$ is the set of discrepancies that are consistent with the hypothesis h_f, where $D_{physical}$ is the set of physical discrepancies described in Sect. 8.3.1. These discrepancies are referred to as consistent discrepancies. We partition the set C into two disjoint subsets, *C1*, *C2* where *C1* consists of primary discrepancies and *C2* contains secondary discrepancies. A discrepancy d w.r.t hypotheses h_f is called primary if the fault propagation linking the discrepancy, d, is certain otherwise it's termed secondary as defined in Sect. 8.3.1.
- $E \subseteq D_{physical}$ is the set of discrepancies which are expected to be activated in the future according to h_f. This set is also partitioned into *E1* and *E2* that contain primary and secondary discrepancies, respectively.
- $M \subseteq D_{physical}$ is the set of discrepancies that are missing according to the hypothesis h_f, i.e. alarms related to these discrepancies should have been signaled. This set is also composed of two disjoint sets *M1* and *M2* based on primary and secondary discrepancies.
- $I \subseteq D_{physical}$ is the set of discrepancies that are inconsistent with the hypothesis h_f. These are the discrepancies that are in the domain of f but cannot be explained in the current mode.
- $U \subseteq D_{physical}$ is the set of discrepancies which are not explained by this hypothesis h_f as there is no fault propagation link between $d \in U$ and $s \in f \cup S \cup C$, i.e. the discrepancy is not in the domain of f.

For every scenario, the reasoner creates one special hypothesis (conservative), **H0** that associates a spurious detection fault with each of the triggered alarms.

The quality of the generated hypotheses is measured based on four metrics defined as follows:

- ***Plausibility***: It is a measure of the degree to which a given hypothesis explains the current fault and its failure signature. Mathematically, it's defined as

$$Plausibility = \frac{|C1| + |C2|}{|C1| + |C2| + |M1| + |I|}$$

- ***Robustness***: It is a measure of the degree to which a given hypothesis will remain constant. Mathematically, it's defined as

$$Robustness = \frac{|C1| + |C2|}{|C1| + |C2| + |M1| + |E1| + |E2| + |I|}$$

- ***Rank***: It is a measure that a given hypothesis (a single physical fault along with multiple cyber faults) completely explains the system events observed. Mathematically, it is defined as $Rank = |C1| + |C2| - |M1| - |U|$
- ***Failure Mode Count***: is a measure of how many failure modes are listed by the hypothesis. The reasoner gives preference to hypotheses that explain the alarm events with a limited number of failure modes (i.e., it follows the parsimony principle). This metric plays an important role while pruning out **H0** from the final hypothesis report.

There are three types of events that invoke the reasoner to update the hypotheses. The first two are external physical events related to a change in the physical state of a monitored discrepancy and system mode. The third event is an internal timeout event that corresponds to the expectation of an alarm. A physical event is formally defined as a tuple $e = (\text{da}, t)$, where $\text{da} \in D_0 \cup M$ is either an observable discrepancy or a system mode. The timeout event is described as a tuple $e = < h_f, \text{da}, t >$ which implies as per hypotheses h_f, any alarm related to discrepancy *da* should have been signaled by time *t*. Figures 8.5 and 8.6 give an overview of the underlying algorithm of reasoner response to three different type of events.

Timeout Event Whenever the observed state of a discrepancy does not change as expected by the reasoner, an internal timeout event, (h, da, t) is generated, where h denotes the set of hypotheses to be updated and da is the expected discrepancy and t is the current time. This event causes reasoner to update the expected sets of all hypotheses, h. If the expected sets, E1(E2), of any hypothesis in h, list da, then it is moved to missing sets M1(M2).

Mode Change Event If any actuator component in the protection layer changes its state, a mode change event is triggered by the corresponding observer. This event causes reasoner to update the expected sets of all hypotheses as the new actuator state might influence the operating modes and disable or enable failure propagation edges.

Discrepancy Mode Change Event This event is triggered if any observer detects appearance or disappearance of failure effects in both plants and protection devices. The event is denoted by (da, t), where da is a discrepancy that activated or deactivated at time t. If the observed state of this alarm is ON (activated), then reasoner iterates over all the hypotheses at time, t, to find hypotheses that explain this discrepancy (which lists da in expected sets). If found, expected and consist

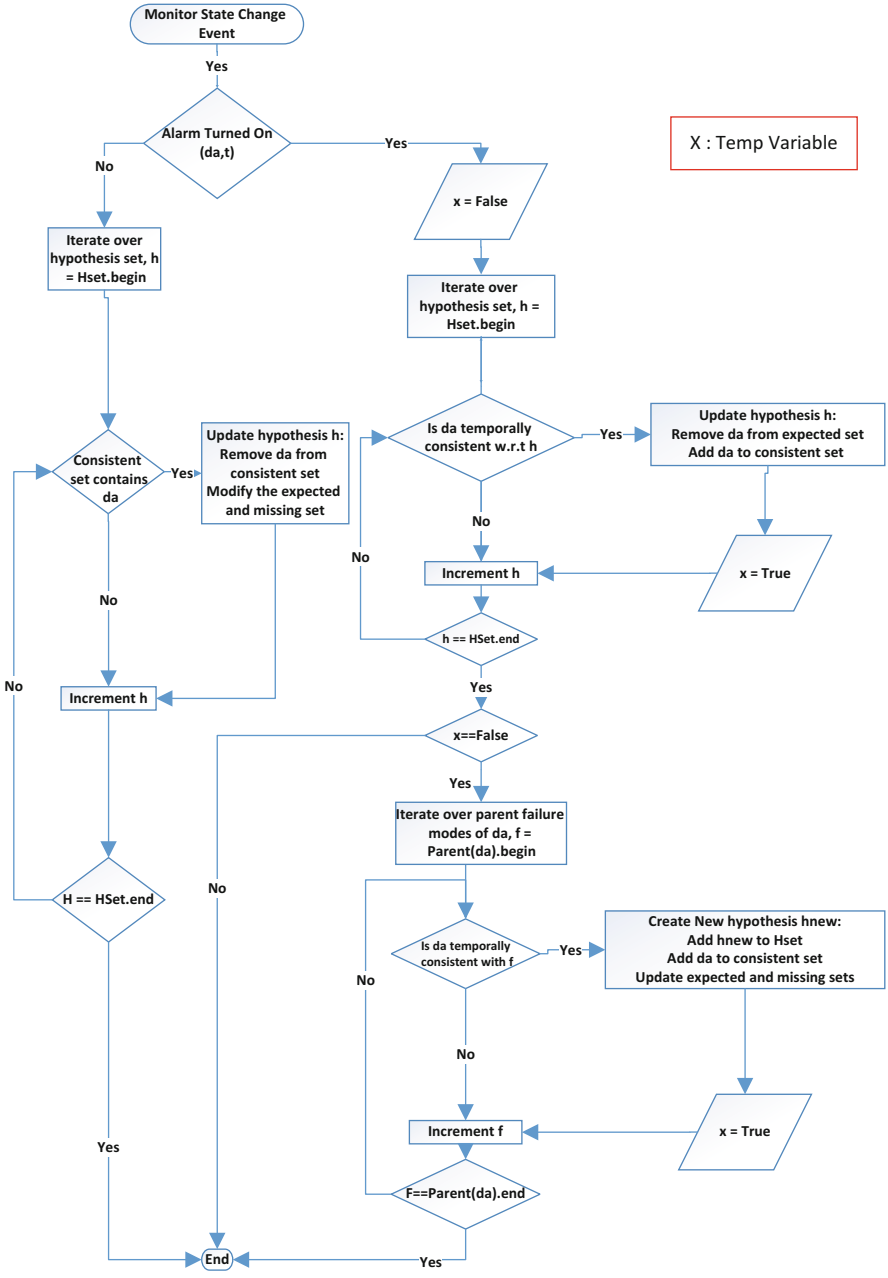

Fig. 8.5 Flowchart for handling Monitor or Discrepancy State Change Event

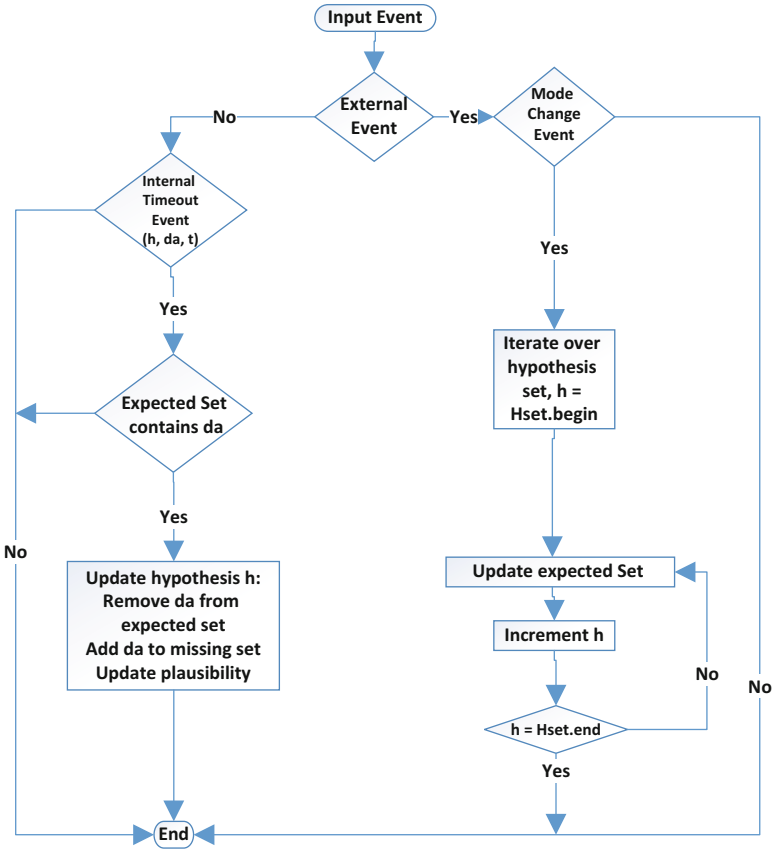

Fig. 8.6 Flowchart for handling Timeout and Mode Change Event

sets of those hypotheses are updated. In case, no hypothesis is discovered, a new hypothesis is generated and added to the hypothesis set. On the other hand, if the observed state of the discrepancy is OFF (deactivated), reasoner iterates over all hypotheses and update the consistent and expected sets of all hypotheses that list da in their consistent sets.

8.4 Example System: Electric Transmission Network

8.4.1 System Under Test

An electric power system can be considered as a tripartite graph with sources at one end and loads at the other with a complex transmission and distribution system in the middle. Figure 8.7 shows a segment of a transmission network where a load, L1

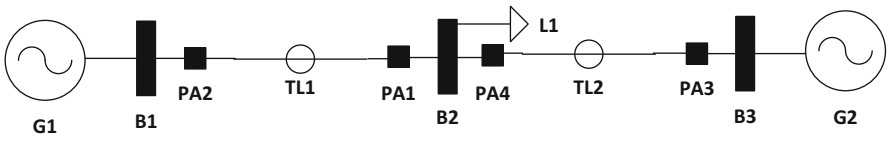

Fig. 8.7 A simple two transmission line system

is being fed by two generators G1, G2 through transmission lines TL1, TL2. The transmission lines are connected by buses B1, B2, B3. All these components are protected by specialized relays and breaker assemblies. In this work, we are focusing on transmission lines only, each transmission line is protected by a set of distance relays and breaker assemblies, installed at each end, collectively represented as a protection assembly labeled as PA1, PA2, PA3, PA4 in Fig. 8.7.

Distance relays are used for detecting two types of faults in transmission lines: (1) phase to phase faults, and (2) phase to ground faults. Both phase to phase and phase to ground faults cause an increase in current flowing through the conductor and decrease in voltage at the buses connected on both ends of the transmission line. This decrease in impedance (V/I) is detected as physical fault and typically categorized by the relay into the following three categories depending upon the calculated impedance:

- **Zone 1 Fault**: If the measured impedance is less than $(0.7 - 0.8) * Z_{TL}$ and the phase angle is between 0 and $\pi/2$, where Z_{TL} is the impedance of the line. The distance relay acts as a primary protection device and instructs the corresponding breaker to open immediately.
- **Zone 2 Fault**: If the measured impedance is greater than $(0.7 - 0.8) * Z_{TL}$ but less than $1.25 * Z_{TL}$ with phase angle being in first quadrant. After detecting a zone 2 fault, distance relay waits for 0.05–0.1 s before sending trip signal to the breaker. This wait time ensures the distance relay to act as a secondary or back-up protection element. If the fault is in any neighboring transmission line, then the wait time ensures the primary protection associated with that line to engage first. In case, the primary distance relays fail, then secondary protection kicks in after the waiting period expires.
- **Zone 3 Fault**: If the measured impedance is in the range $(1.25 - 2) * Z_{TL}$ with phase angle between 0 and $\pi/2$, then the fault considered as zone 3 fault. Similar to zone 2, the protection device acts as a back-up element in case primary device fails to engage. The wait time is of the order of $1 \ldots 1.5$ s.

The time to detect fault depends upon the sampling period of the relay and is of the range 16–30 ms.

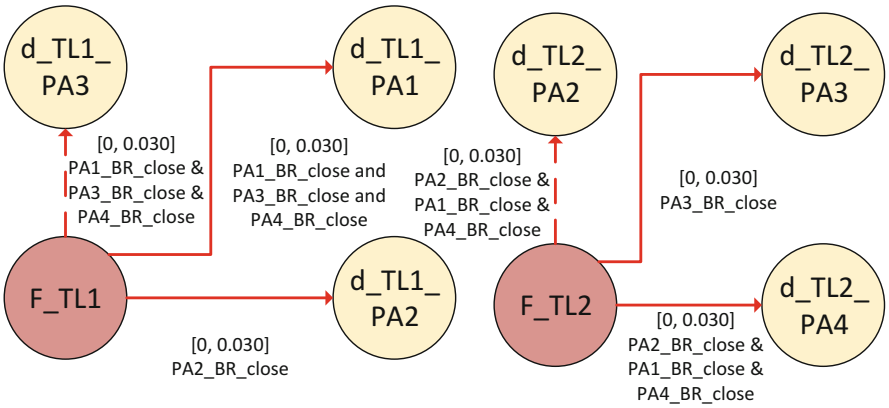

Fig. 8.8 Fault propagation graph for faults in two different transmission lines

Table 8.1
Discrepancy–Alarm Association Map

Discrepancy	Alarms
d_TL1_PA1	PA1_DR_Z1, PA1_DR_Z2
d_TL1_PA2	PA2_DR_Z1, PA2_DR_Z2
d_TL1_PA3	PA3_DR_Z2, PA3_DR_Z3
d_TL2_PA3	PA3_DR_Z1, PA3_DR_Z2
d_TL2_PA4	PA4_DR_Z1, PA4_DR_Z2
d_TL2_PA2	PA2_DR_Z2, PA2_DR_Z3

8.4.2 TCD: Fault Propagation Graph

The fault detection events are recorded by Sequence Event Recorders installed at substations. Using these events as alarms fault propagation graph can be created. Figure 8.8 shows such a graph for the segment of transmission network. The set of nodes labeled as F_TLn represents physical fault in transmission line, TLn. The discrepancy d_TLn_PAk represents the effect of failure F_TLn and the node represents the decrease in impedance as detected by relay in PAk. The edge between nodes represents the fault propagation and is constrained by the timing and operating conditions. The operating conditions are modeled in terms of the physical state of the breakers. The distance relay in PA4 will detect the failure mode F_TL1 as long as all the breakers in the path between G2 and TL1 are in close state. Table 8.1 lists the alarms that can signal discrepancies shown in Fig. 8.8, where the columns identify discrepancies, alarms, and the uncertainty associated to it. The failure edges that link failure source and discrepancy related to secondary protection relay are marked uncertain, i.e. ND(e) = false, depicted as dotted lines in Fig. 8.8.

A primary protection element will always signal Zone 1 or Zone 2 alarm for fault injected at any point in the transmission line. The secondary protection devices will always signal either Zone 2 or Zone 3 alarm depending upon the location of the fault.

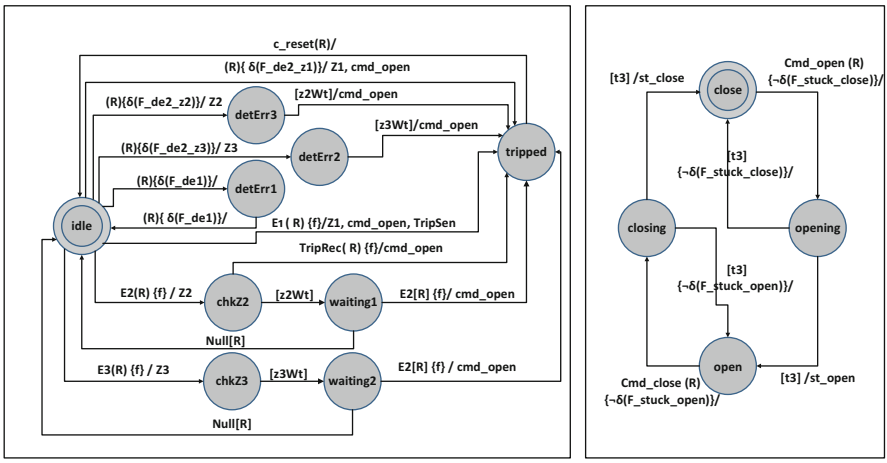

Fig. 8.9 Protection System Behavior Components (Left: Distance Relay; Right: Breaker), where f is a failure mode constraint defined as f:$\neg\delta$(F_de1) $\wedge\neg\delta$(F_de2_z1) $\wedge\neg\delta$(F_de2_z2) $\wedge\neg\delta$(F_de2_z3)

The failure graph captures the propagation of failures under different conditions (breaker states) but does not contain any information to diagnose faults related with the behaviors of breakers and relays. Figure 8.9 shows the TTA model of a protection assembly (distance relay and breaker).

8.4.3 TCD: Distance Relay Behavioral Model

Modern relays are reactive devices that monitor the health of the physical devices at a fixed rate, R secs. Figure 8.9 shows a time triggered model of a distance relay configured to detect Zone 1, 2, 3 faults. The time triggered automaton appropriately models the behavior of a relay under both faulty and non-faulty conditions. The model considers two types of faults, $F = f1 \cup f2$, where $f1 = \{Fde1\}$ is a set of missed detection faults and $f2 = \{Fde2z1, Fde2z2, Fde2z3\}$ is the collection of spurious detection faults related to three zones. As the name implies, a missed detection fault forces the relay to skip the detection of any fault conditions and a spurious detection fault, $Fde2zk$, ensues incorrect inference of zone k fault by the relay. Figure 8.9 lists five different failure mode constraints, namely, $\delta(Fde1)$, $\delta(Fde2z1)$, $\delta(Fde2z2)$, $\delta(Fde2z3)$, $\neg\delta(Fde1) \wedge \neg\delta(Fde2z1) \wedge \neg\delta(Fde2z2) \wedge \neg\delta(Fde2z3)$, where the first four imply the presence of a failure mode, i.e. its state is ON while the last means none of the failure modes in F are present.

There are a total of nine events used to model the behavior of the relay. Out of nine events, three are unobservable, labeled as E1, E2, and E3. These events represent the presence of zone 1, 2, 3 fault conditions. The state machine consists of nine locations, with idle being the initial location. In the idle location, automaton

check for events—E1, E2, E3, and the status of failure modes every R seconds. If the distance relay detects zone 1 fault (modeled by the presence of the event E1), then the distance relay moves to the tripped location and issues a Z1 alarm and commands the breaker to open by emitting event, `cmd_open`. For zone 2 and zone 3 faults conditions (E2, E3), the protection relay does not issue an open command after moving to the `chkZ2` or `chkZ3` locations. The state machine waits for predefined time, $zn2wt, zn3wt \in \mathbb{R}_+$ and confirms again the presence of the fault conditions, once the time expires. If the fault is still present, the relay commands the breaker to open and transitions to tripped location, otherwise moves back to idle location. Additionally, distance relays may be configured with overreach trip transfer protocols. In this case, the primary relays associated with a transmission line send permissive trip signals to each other, `TripSen`, in order to avoid zone 2 wait time.

The deviation in the normal behavior of the relay is caused if any of the failure mode constraints evaluates to true. For instance, if the current location of the automaton is idle and failure mode $Fde1$ is present then automaton jumps to detErr1 location and stays there until the fault is persistent. Similarly if any of the spurious detection faults are present, then irrespective of the presence of E1, E2, and E3, the state machine jumps to `detErr2` or `detErr3` and finally transitions to tripped state. In this model, the faults (`F_de1`, `F_de2_z1`, `F_de2_z2`, `F_de2_z3`) are assumed to be mutually exclusive, i.e. one of the cyber faults can be present at a given time.

8.4.4 TCD: Breaker Behavioral Model

Figure 8.9 also shows TTA model of a breaker with two failure modes, F = *{F_stuck_close, F_stuck_open}*. The breaker automaton has four states labeled as `open`, `opening`, `close`, and `closing`, with `close` being the initial state. All the events used in the state machine are observable. The events cmd_open, cmd_close represent the commands received by the breaker assembly and st_open, st_close signify change in the physical state of the breaker. The transition from open to close and vice verse is not instantaneous. The lag is caused due to mechanical nature of the breaker and zero crossing detection, which is modeled by parameter t3. Automaton consists of two failure mode constraints, $\neg\delta(Fstuckclose)$, $\neg\delta(Fstuckopen)$, which evaluates to true when respective failure modes are not present.

The breaker is also modeled as reactive component which is periodically checking for commands. While in the `close` location, the automaton looks for event cmd_open and evaluates the failure constraint every R secs. If the event is present and F_stuck_close fault is absent, the state machine transitions to opening state. After t3 secs, the automaton moves to open state if failure mode constraint still evaluates to false. Similarly in open location, the presence of the event cmd_close and validity of failure constraint is checked.

8.4.5 TCD Diagnosis System: Observers

The TCD based diagnosis system employs a hierarchical framework as shown in Fig. 8.3. The lower layer includes observers that track the operation of cyber components (distance relays and circuit breakers) to detect and locally diagnose faults in physical and protection systems. The observers feed their results to the reasoning engine as explained in previous section. The TCD reasoning engine produces a set of hypotheses that explain the current system states as per the output of various observers by traversing the fault propagation graph. The traversal is constrained by the state of the protection system as predicted by observers tracking it. The following sections provide a detailed description of the model and operation of the observers related with power system protection devices.

8.4.5.1 Observer: Distance Relay

The TTA model of a distance relay observer can be seen in Fig. 8.10. The state machine has eight locations with `idle` being the initial state. The observer machine consumes the observable zone alarms (Z1, Z2, Z3), commands sent to breaker (cmd_open) and reset events and produce h_Z1, h_Z2, h_Z3 to indicate or confirm the presence of zone 1, 2, 3 faults. The observer also produces h_Z1′, h_Z2′ and h_Z3′ to indicate absence of zone 1, 2, 3 fault conditions. The observer remains in the `idle` position until zone fault conditions are reported by the corresponding distance relay. Once the distance relay fires a *Z1* event, the observer machine jumps to the `chkZ1` location while emitting h_Z1 event. The observer machine waits for *t2* seconds for open command (cmd_open event). If received, the observer moves

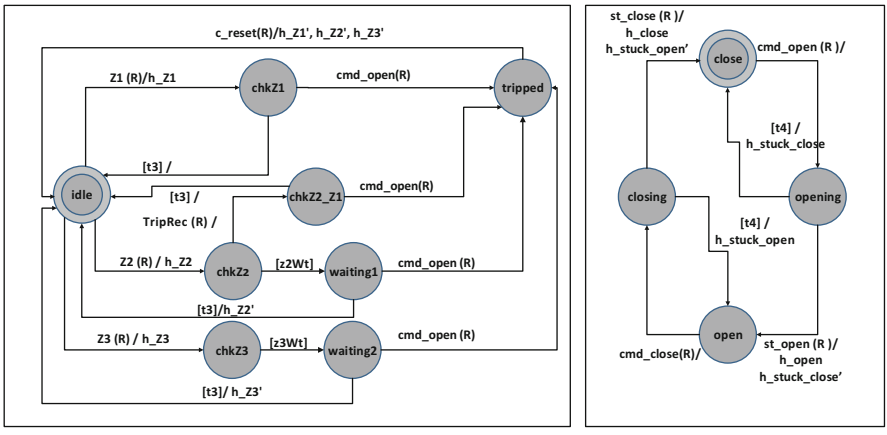

Fig. 8.10 Protection System Observer Models, Distance Observer Model (Left); Breaker Observer Model (Right)

to the `tripped` state, otherwise transitions back to `idle` state. *t2* is a parameter of the distance relay observer machine that models propagation delay and relay frequency. Please note that the transition from `chkZ1` state to the *idle* state implies a communication channel fault, but in this paper we are not considering such faults.

Similarly, the observer machine moves to the `chkZ2` state when the distance relay reports a Z2 event after detecting zone 2 fault conditions. Upon the confirmation of zone 2 fault, the observer waits *t3* seconds for the arrival of the *cmd_open* command. *t3* is a parameter which is equal to the sum of zone 2 wait time and t2. If the *cmd_open* event is not observed within *t3* seconds the automaton moves back to the `idle` state and concludes that the zone 2 fault condition has disappeared by generating h_Z2′ event. The observer machine moves from `chkZ2` state to `chkZ2_Z1` state if the event *TripRec* occurs and waits for the *cmd_open* event and concludes the presence of fault by producing h_Z2 event. In a similar fashion, the distance relay observer diagnoses zone 3 faults.

8.4.5.2 Observer: Circuit Breaker

The breaker observer model is shown in the right side of Fig. 8.10. It consists of four states labeled as `open`, `close`, `opening`, and `closing` and correlate directly to the four states of the breaker automaton. Initially the state machine is in the `close` state and jumps to the `opening` state after observing *cmd_open* event. The breaker observer transitions to the `open` state if it receives an *st_open* event from the breaker assembly within *t4* seconds. *t4* is a model parameter that is equal to the sum of propagation time and the maximum time required to open the breaker. If the event is observed in the time limit, the observer concludes the physical state of breaker is open and stuck close fault is not present by producing an event, h_stuck_close′. Otherwise it hypothesizes that the breaker has the stuck close fault. The fault is signaled by generating an event, *h_stuck_close*. Similarly, when the breaker is in the `open` state it has the same timed behavior and an *h_stuck_open* event is generated if an *st_close* event is not observed within *t4* seconds of receiving the *cmd_close* event.

8.4.6 Results

Figure 8.11 shows the sequence of events generated by protection devices, observers, and reasoning engine when a three phase to ground fault is injected in transmission line TL2 along with the presence of missed detection fault in PA4_DR and stuck close fault in PA2_BR. At $t = 0.501$, PA3_DR_OBS and PA2_DR_OBS report h_Z1 and h_Z3 alarms. These alarms produce two hypotheses H0, H1. H1 lists faults in line TL2 with two consistent discrepancies and expects an alarm from PA4_DR_OBS (h_Z1 or h_Z2). At $t = 0.531$, timeout forces the expected discrepancy to shift to the missing set. H1 and H0 both list two failure modes.

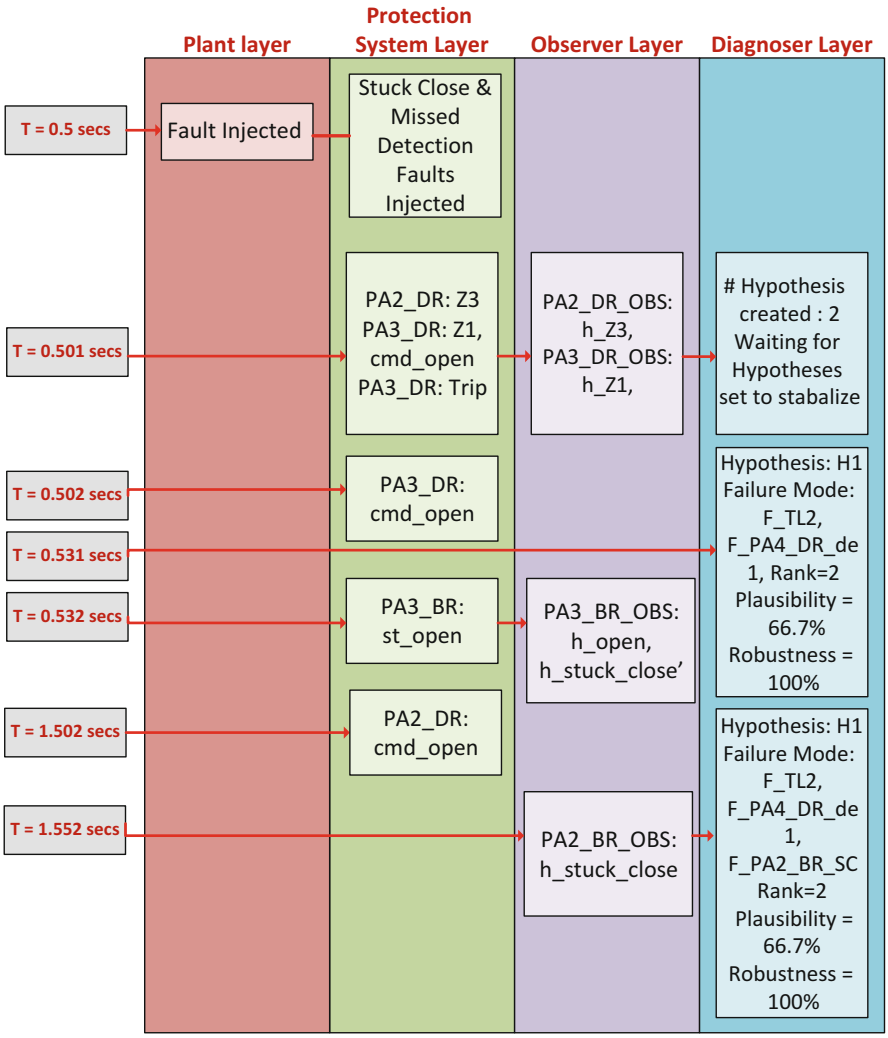

Fig. 8.11 Diagnosis results for scenario 4

H1 lists physical faults associated with line TL2 along with a missed detection fault in PA4_DR whereas H0 blames both the distance relays for having spurious detection faults. At $t = 1.552$, PA2_BR_OBS concludes a stuck fault in breaker PA2_BR after failing to receive a state change event (st_open). Both hypotheses are updated to reflect the breaker fault. The hypothesis H1 is given preference over H0 as the probability of two cyber faults is less than a physical and a cyber fault [62]. Figure 8.11 shows the events sequence and hypotheses evolution.

8.5 Conclusion

We have presented a new formalism: Temporal Causal Diagrams with the aim of diagnosing failures in cyber-physical systems that include local fast-acting protection devices. Specifically, we have demonstrated the capability of the TCD model to capture the discrete fault propagation and behavioral model of a segment of a power transmission system protected by distance relays and breakers. The paper also presented hierarchical TCD-based reasoner to diagnose faults in the physical system and its protection elements.

Acknowledgements This work is funded in part by the National Science Foundation under the award number CNS-1329803. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of NSF. The authors would like to thank Rishabh Jain, Srdjn Lukic, Saqib Hasan, Scott Eisele, and Amogh Kulkarni for their help and discussions related to the work presented here.

References

1. North American Electric Reliability Corporation, 2012 state of reliability, Tech. Rep. (2012). Available: http://www.nerc.com/files/2012_sor.pdf
2. S. Abdelwahed, G. Karsai, G. Biswas, A consistency-based robust diagnosis approach for temporal causal systems, in *The 16th International Workshop on Principles of Diagnosis* (2005), pp. 73–79
3. N. Mahadevan, A. Dubey, G. Karsai, Application of software health management techniques, in *Proceedings of the 6th International Symposium on Software Engineering for Adaptive and Self-managing Systems*, ser. SEAMS '11 (ACM, New York, 2011), pp. 1–10. Available: http://doi.acm.org/10.1145/1988008.1988010
4. P. Seifried, Fault detection and diagnosis in chemical and petrochemical processes, Bd. 8 der Serie „Chemical Engineering Monographs". Von D. M. Himmelblau, herausgegeben von S. W. Churchill, Elsevier Scientific Publishing Company, Amsterdam – New York 1978. 1. Aufl., X, 414 S., 137 Abb., 66 Tab., DM 145,–. Chem. Ing. Tech. **51**, 766 (1979). https://doi.org/10.1002/cite.330510726
5. N. Viswanadham, T.L. Johnson, Fault detection and diagnosis of automated manufacturing systems, in *27th IEEE Conference on Decision and Control* (1988)
6. R. Hessian, B. Salter, E. Goodwin, Fault-tree analysis for system design, development, modification, and verification. IEEE Trans. Reliab. **39**(1), 87–91 (1990)
7. Y. Ishida, N. Adachi, H. Tokumaru, Topological approach to failure diagnosis of large-scale systems. IEEE Trans. Syst. Man Cybern. **15**(3), 327–333 (1985)
8. S.V.N. Rao, N. Viswanadham, Fault diagnosis in dynamical systems: a graph theoretic approach. Int. J. Syst. Sci. **18**(4), 687–695 (1987)
9. S.V.N. Rao, N. Viswanadham, A methodology for knowledge acquisition and reasoning in failure analysis of systems. IEEE Trans. Syst. Man Cybern. **17**(2), 274–288 (1987)
10. J. Richman, K.R. Bowden, The modern fault dictionary, in *International Test Conference* (1985), pp. 696–702
11. W.T. Scherer, C.C. White, A survey of expert systems for equipment maintenance and diagnostics, in *Knowledge-Based System Diagnosis, Supervision and Control*, ed. by S.G. Tzafestas (Plenum, New York, 1989), pp. 285–300

12. S. Tzafestas, K. Watanabe, Modern approaches to system/sensor fault detection and diagnosis. J. A. IRCU Lab. **31**(4), 42–57 (1990)
13. P. Frank, Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy – a survey and some new results. Automatica **26**(3), 459–474 (1990)
14. W. Hamscher, L. Console, J. de Kleer, *Readings in Model-Based Diagnosis* (Morgan Kaufmann Publishers Inc., San Francisco, 1992)
15. R. Patton, Robust model-based fault diagnosis: the state of the art, in *IFAC Fault Detection, Supervision and Safety for Technical Processes*, Espoo (1994), pp. 1–24
16. R. Patton, P. Frank, R. Clark, *Fault Diagnosis in Dynamic Systems: Theory and Application* (Prentice Hall International, Englewood Cliffs, 1989)
17. M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, D. Teneketzis, Failure diagnosis using discrete event models. IEEE Trans. Control Syst. Technol. **4**, 105–124 (1996)
18. A.N. Srivastava, Discovering system health anomalies using data mining techniques, in *Proceedings of the Joint Army Navy NASA Air Force Conference on Propulsion* (2005)
19. R. Reiter, A theory of diagnosis from first principles. Artif. Intell. **32**(1), 57–95 (1987)
20. J. de Kleer, A. Mackworth, R. Reiter, Characterizing diagnoses and systems. Artif. Intell. **56**, 197–222 (1992)
21. A. Darwiche, Model-based diagnosis using structured system descriptions. J. Artif. Intell. Res. **8**, 165–222 (1998)
22. A. Darwiche, G. Provan, Exploiting system structure in model-based diagnosis of discrete-event systems, in *Proceedings of the Seventh International Workshop on Principles of Diagnosis* (1996), pp. 95–105
23. J. Gamper, A temporal reasoning and abstraction framework for model-based diagnosis systems. Ph.D. dissertation, RWTH, Aachen, 1996
24. L. Console, P. Torasso, On the co-operation between abductive and temporal reasoning in medical diagnosis. Artif. Intell. Med. **3**(6), 291–311 (1991)
25. A. Misra, Sensor-based diagnosis of dynamical systems. Ph.D. dissertation, Vanderbilt University, 1994
26. A. Misra, J. Sztipanovits, J. Carnes, Robust diagnostics: structural redundancy approach, in *SPIE's Symposium on Intelligent Systems* (1994)
27. S. Padalkar, J. Sztipanovits, G. Karsai, N. Miyasaka, K.C. Okuda, Real-time fault diagnostics. IEEE Expert **6**(3), 75–85 (1991)
28. G. Karsai, J. Sztipanovits, S. Padalkar, C. Biegl, Model based intelligent process control for cogenerator plants. J. Parallel Distrib. Syst. **15**, 90–103 (1992)
29. P.J. Mosterman, G. Biswas, Diagnosis of continuous valued systems in transient operating regions. IEEE Trans. Syst. Man Cybern. **29**(6), 554–565 (1999)
30. G. Karsai, G. Biswas, S. Abdelwahed, Towards fault-adaptive control of complex dynamic systems, in *Software-Enabled Control: Information Technology for Dynamical Systems*, ch. 17, ed. by T. Samad, G. Balas (IEEE Publication, Piscataway, 2003)
31. S. Abdelwahed, G. Karsai, G. Biswas, System diagnosis using hybrid failure propagation graphs, in *The 15th International Workshop on Principles of Diagnosis*, Carcassonne, 2004
32. V. Brusoni, L. Console, P. Terenziani, D.T. Dupre, A spectrum of definitions for temporal model-based diagnosis. Artif. Intell. **102**(1), 39–79 (1998)
33. Z. Yongli, Y.H. Yang, B.W. Hogg, W.Q. Zhang, S. Gao, An expert system for power systems fault analysis. IEEE Trans. Power Syst. **9**(1), 503–509 (1994)
34. Y.-C. Huang, Fault section estimation in power systems using a novel decision support system. IEEE Trans. Power Syst. **17**(2), 439–444 (2002)
35. G. Cardoso, J.G. Rolim, H.H. Zurn, Identifying the primary fault section after contingencies in bulk power systems. IEEE Trans. Power Deliv. **23**(3), 1335–1342 (2008)
36. J. Jung, C.-C. Liu, M. Hong, M. Gallanti, G. Tornielli, Multiple hypotheses and their credibility in on-line fault diagnosis. IEEE Trans. Power Deliv. **16**(2), 225–230 (2001)
37. G. Cardoso, J.G. Rolim, H.H. Zurn, Application of neural-network modules to electric power system fault section estimation. IEEE Trans. Power Delivery **19**(3), 1034–1041 (2004)

38. R.N. Mahanty, P.B.D. Gupta, Application of RBF neural network to fault classification and location in transmission lines. IEE Proc. Gener. Transm. Distrib. **151**(2), 201–212 (2004)
39. D. Thukaram, H.P. Khincha, H.P. Vijaynarasimha, Artificial neural network and support vector machine approach for locating faults in radial distribution systems. IEEE Trans. Power Delivery **20**(2), 710–721 (2005)
40. T. Bi, Z. Yan, F. Wen, Y. Ni, C. Shen, F.F. Wu, Q. Yang, On-line fault section estimation in power systems with radial basis function neural network. Int. J. Electr. Power Energy Syst. **24**(4), 321–328 (2002)
41. Y.-X. Wu, X.N. Lin, S.H. Miao, P. Liu, D.Q. Wang, D.B. Chen, Application of family eugenics based evolution algorithms to electric power system fault section estimation, in *Transmission and Distribution Conference and Exhibition: Asia and Pacific, 2005 IEEE/PES* (2005), pp. 1–5
42. F. Wen, C. Chang, Probabilistic approach for fault-section estimation in power systems based on a refined genetic algorithm. IEE Proc. Gener. Transm. Distrib. **144**(2), 160–168 (1997)
43. Z. He, H.-D. Chiang, C. Li, Q. Zeng, Fault-section estimation in power systems based on improved optimization model and binary particle swarm optimization, in *IEEE Power & Energy Society General Meeting, 2009. PES'09* (IEEE, Piscataway, 2009), pp. 1–8
44. W. Guo, F. Wen, G. Ledwich, Z. Liao, X. He, J. Liang, An analytic model for fault diagnosis in power systems considering malfunctions of protective relays and circuit breakers. IEEE Trans. Power Deliv. **25**(3), 1393–1401 (2010)
45. J. Sun, S.-Y. Qin, Y.-H. Song, Fault diagnosis of electric power systems based on fuzzy petri nets, IEEE Trans. Power Syst. **19**(4), 2053–2059 (2004)
46. W.-H. Chen, C.-W. Liu, M.-S. Tsai, Fast fault section estimation in distribution substations using matrix-based cause-effect networks. IEEE Trans. Power Deliv. **16**(4), 522–527 (2001)
47. W.H. Chen, S.H. Tsai, H.I. Lin, Fault section estimation for power networks using logic cause-effect models. IEEE Trans. Power Deliv. **26**(2), 963–971 (2011)
48. W. Guo, L. Wei, F. Wen, Z. Liao, J. Liang, C.L. Tseng, An on-line intelligent alarm analyzer for power systems based on temporal constraint network, in *International Conference on Sustainable Power Generation and Supply, 2009. SUPERGEN '09* (2009), pp. 1–7
49. W.H. Chen, Online fault diagnosis for power transmission networks using fuzzy digraph models. IEEE Trans. Power Deliv. **27**(2), 688–698 (2012)
50. Z. Yongli, H. Limin, L. Jinling, Bayesian networks-based approach for power systems fault diagnosis. IEEE Trans. Power Deliv. **21**(2), 634–639 (2006)
51. Y. Sekine, Y. Akimoto, M. Kunugi, C. Fukui, S. Fukui, Fault diagnosis of power systems. Proc. IEEE **80**(5), 673–683 (1992)
52. 1962. Available: http://www2.ee.washington.edu/research/pstca/pf14/pg_tca14bus.htm
53. 1962. Available: http://icseg.iti.illinois.edu/ieee-14-bus-system/
54. 2016. Available: http://www2.ee.washington.edu/research/pstca/formats/cdf.txt
55. R. Dugan, *OpenDSS Manual*. Electrical Power Research Institute, 2016. Available: http://sourceforge.net/apps/mediawiki/electricdss/index.php
56. S. Padalkar, G. Karsai, C. Biegl, J. Sztipanovits, K. Okuda, N. Miyasaka, Real-time fault diagnostics. IEEE Expert **6**(3), 75–85 (1991)
57. S. Abdelwahed, G. Karsai, Notions of diagnosability for timed failure propagation graphs, in *2006 IEEE Autotestcon*, Sept 2006, pp. 643–648
58. A. Dubey, G. Karsai, N. Mahadevan, Model-based software health management for real-time systems, in *2011 IEEE Aerospace Conference* (IEEE, Piscataway, 2011), pp. 1–18
59. P. Krčál, L. Mokrushin, P. Thiagarajan, W. Yi, Timed vs. time-triggered automata, in *CONCUR 2004-Concurrency Theory* (Springer, Berlin, 2004), pp. 340–354
60. M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, D. Teneketzis, Diagnosability of discrete-event systems. IEEE Trans. Autom. Control **40**(9), 1555–1575 (1995)
61. S. Tripakis, Fault diagnosis for timed automata, in *International Symposium on Formal Techniques in Real-Time and Fault-Tolerant Systems* (Springer, Berlin, 2002), pp. 205–221
62. E. Schweitzer, B. Fleming, T.J. Lee, P.M. Anderson et al., Reliability analysis of transmission protection using fault tree methods, in *Proceedings of the 24th Annual Western Protective Relay Conference* (1997), pp. 1–17