Qualitative Event-based Diagnosis with Possible Conflicts: Case Study on the Fourth International Diagnostic Competition*

Matthew Daigle[1] and **Indranil Roychoudhury**[2] and **Anibal Bregon**[3]
[1]NASA Ames Research Center, Moffett Field, CA, 94035, USA
e-mail: matthew.j.daigle@nasa.gov
[2]SGT Inc., NASA Ames Research Center, Moffett Field, CA, 94035, USA
e-mail: indranil.roychoudhury@nasa.gov
[3]Department of Computer Science, University of Valladolid, Valladolid, 47011, Spain
e-mail: anibal@infor.uva.es

Abstract

The objective of the International Diagnostic Competition is to provide a platform for evaluating how different diagnostic algorithms perform and compare to one another when applied to the same problem. This paper describes three model-based diagnosis algorithms entered into the Fourth International Diagnostic Competition. We focus on the first diagnostic problem of the industrial track of the competition in which a diagnosis algorithm must detect, isolate, and identify faults in an electrical power distribution testbed in order to provide abort recommendations when warranted. We present here a general fault isolation framework that encompasses three algorithms, each of which use different residual sets for fault isolation; one based on the global system model, one based on minimal submodels computed using Possible Conflicts, and one based on the combination of the former two residual sets. We describe, compare, and contrast the three algorithms in terms of practical implementation and their diagnosis results.

1 Introduction

In this paper, we present a model-based, qualitative, event-based fault diagnosis scheme entered into the Fourth International Diagnostic Competition (DXC'13). The competition allows for a comparative study of different diagnostic approaches, and includes multiple diagnostic problems. Different diagnostic algorithms applied to the same diagnostic problem are compared to one another and ranked in terms of metrics developed in [Poll *et al.*, 2010; 2011]. In this work, we focus on diagnostic problem I (DPI) of the industrial track of the competition, which consists of fault diagnosis and recovery for a subset of the Advanced Diagnosis and Prognosis Testbed (ADAPT) [Poll *et al.*, 2007], called ADAPT-Lite, which is an electrical power distribution system. Our diagnosis scheme has three instantiations, the first two being QED (Qualitative Event-based Diagnosis) and QED-PC (QED with Possible Conflicts [Pulido and Alonso-González, 2004]), both of which were previously submitted to the Third International Diagnostic Competition (DXC'11) [Daigle *et al.*, 2012], and have been improved and updated for this new edition of the competition. In addition to these two algorithms, we present a new entry in this year's diagnostic competition, QED-PC++, which may be considered as a combination of QED and QED-PC.

QED extends the TRANSCEND qualitative diagnosis scheme described in [Mosterman and Biswas, 1999]. In this scheme, fault isolation is obtained through analysis of the transients produced by faults, manifesting as deviations in observed behavior from predicted nominal behavior. TRANSCEND was extended by including relative residual orderings, which provide a partial ordering of measurement deviations for different faults, leading to an enhanced event-based fault isolation scheme [Daigle *et al.*, 2009; 2007]. TRANSCEND deals only with abrupt faults, so in [Daigle *et al.*, 2012] we incorporated methods for incipient and intermittent fault isolation and identification.

Whereas QED uses a global model of the system for residual generation, the second algorithm, QED-PC, uses the Possible Conflicts (PCs) diagnosis approach (presented in [Pulido and Alonso-González, 2004]), in which residuals are generated from minimal single-output submodels. This approach decomposes the global system model into minimal submodels containing sufficient analytical redundancy to generate fault hypotheses from observed measurement deviations. In this work, we use the qualitative fault isolation framework of QED to perform residual analysis, using the PC-based residuals. For fault identification, we use minimal parameter estimators computed from PCs for each faulty parameter as proposed in [Bregon *et al.*, 2012].

The third algorithm, named QED-PC++, can be seen as the combination of the previous two diagnosis schemes. QED-PC++ uses the residual sets used by QED (computed using the global system model) and QED-PC (computed from the PCs) to form a diagnosis scheme which improves upon QED and QED-PC individually. In the previous competition, we found that QED excelled at isolating nonsensor faults, but needed ad hoc fault isolation rules in order to isolate sensor faults without the aid of fault identification. In contrast, for QED-PC, we found that it excelled at isolating sensor faults, but required ad hoc fault isolation rules to isolate nonsensor faults. By using residuals from both the global model and the PCs, we eliminate the weaknesses of the individual residual sets and obtain improved diagnosability for QED-PC++ over QED and QED-PC individually.

The paper is organized as follows. First, Section 2 overviews the general diagnosis approach. Section 3 dis-

*M. Daigle and I. Roychoudhury's funding for this work was provided by the NASA System-wide Safety and Assurance Technologies (SSAT) Project. A. Bregon's funding for this work was provided by the Spanish MCI TIN2009-11326 grant.

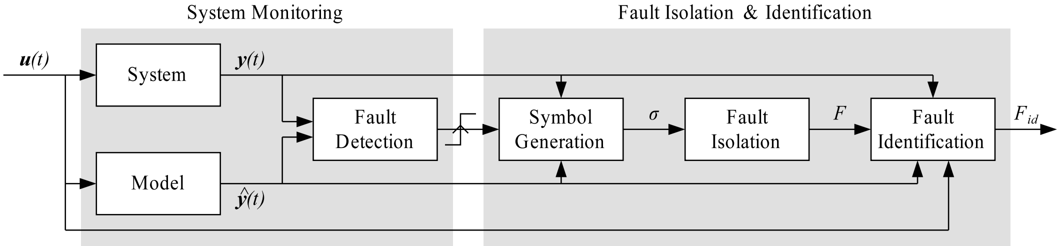

Figure 1: Diagnosis architecture.

cusses diagnosability within our fault isolation framework. Section 4 describes three diagnosers that are instantiated from the general diagnosis approach. Section 5 presents diagnosis results, and Section 6 concludes the paper.

2 Diagnosis Approach

Our diagnosis approach uses the model of the system, and performs the tasks of (*i*) fault detection, (*ii*) fault isolation, and (*iii*) fault identification. The diagnosis architecture is shown in Fig. 1, and reflects the implementation of the three algorithms. Next, we briefly introduce each of the main blocks within our architecture. Additional details on each module may be found in [Daigle *et al.*, 2012].

2.1 System Modeling

ADAPT-Lite consists of relays, circuit breakers, AC and DC resistive loads, a fan, a battery, and a DC to AC inverter, along with 11 sensors measuring current, voltage, relay position, fan speed, and battery temperature (see [Poll *et al.*, 2011]). Our diagnosis approach is model-based, requiring a model of both nominal and faulty behavior for use throughout the diagnosis process. The three algorithms implement the nominal model in a different way. For QED, the nominal model is a global model of the system, $\mathcal{M}$, and its inputs are those of the global system. For QED-PC, the nominal model is composed of a set of 11 minimal submodels computed from PCs, with each submodel $\mathcal{M}_i$ estimating the value of sensor i using a subset of the system measurements as input variables. QED-PC++ uses both the global model and the 11 submodels, thus having 22 total outputs and 22 residuals.

2.2 Fault Detection

The three algorithms use the same approach for fault detection. Each residual is assigned a fault detector. For each output $y(t)$, we define the residual as $r(t) = y(t) - \hat{y}(t)$, where $\hat{y}(t)$ is the model-predicted output signal. As previously described, for QED, $\hat{y}(t)$ is computed using the global model, for QED-PC, it is computed using the corresponding PC, and for QED-PC++, it is computed using each. Fault detection works by applying a Z-test to the residual values. The Z-test detects statistically significant nonzero residual signals, which indicate the presence of a fault.

The real sensor data contains some intermittent spikes which are not to be classified as faults. In order to avoid false alarms and incorrect fault isolation due to data spikes, we implement a median filter, which takes the last three values of a given sensor and reports the median of these values as the current sensor value. As a result, fault detection will be delayed and its sensitivity decreased, however this is preferred over false alarms and misdiagnoses.

2.3 Fault Isolation

In our proposal, we use a qualitative diagnosis methodology that isolates faults based on the transients they cause in system behavior, manifesting as deviations in residual values [Mosterman and Biswas, 1999]. The transients produced by faults are abstracted using qualitative + (increase), - (decrease), and `0` (no change) values and `N` (zero to nonzero), `Z` (nonzero to zero), and `X` (no discrete change) values to form *fault signatures*. Fault signatures represent these measurement deviations from nominal behavior as the immediate (discontinuous) change in magnitude, the first nonzero derivative change, and discrete zero/nonzero value changes in the measurement from the estimate caused by discrete faults. These symbols are computed from the residuals using symbol generation, as described in [Daigle and Roychoudhury, 2010]. In addition we use the relative order of residual deviation, termed relative residual orderings, to isolate faults [Daigle *et al.*, 2007]. Fault isolation fundamentals for the algorithms will be detailed in Section 3.

In addition to qualitative fault isolation, some heuristic fault isolation rules are introduced to improve fault isolation times and overall diagnosability. For example, if a relay fault occurs, we should observe a change in the residual for its corresponding sensor almost immediately. Most rules are of the form where we give an emprical time limit to an expected deviation for a fault, such that if the deviation is not observed, we eliminate that fault from the candidate list. Specific rules are provided in [Daigle *et al.*, 2012]. For QED-PC++, due to its improved diagnosability from the combined residual set, most rules are eliminated. For example, for QED we need a rule to rule out nonsensor faults if after a significant amount of time only one residual has been observed to deviate; QED-PC++ does not require that rule because multiple PC-based residuals will deviate due to a sensor fault.

2.4 Fault Identification

Fault identification takes the set of isolated faults, and computes the fault parameters for each, producing a new fault set F_{id} including this information. Identification is initiated immediately after the initial set of fault candidates is produced after fault detection. An identification routine is run for each fault candidate, which updates at each time step. All of the algorithms use minimal submodels for fault identification, except for sensor faults in which QED and QED-PC++ use the global model.

2.5 Recovery

At the end of the scenario, the decision whether to abort or continue the mission must be made. The fault identification module computes a candidate set F_{id}, with each $f \in F_{id}$

being defined by the component, its fault mode, and the associated fault parameters. The oracle is viewed as a function $O(f)$ which, for a given fault, computes a recommended set of commands C. For DPI, either $C = \{abort\}$ or $C = \varnothing$.

Each command set has an associated cost. The cost is zero when the correct command is chosen. If the algorithm recommends $abort$ when the mission should be continued, the associated cost is that of the mission (25). If the algorithm recommends to continue when it should have been aborted, the associated cost is that of the mission and the vehicle (125). Therefore, we take the conservative approach and recommend $abort$ if $O(f) = \{abort\}$ for at least one $f \in F_{id}$. In the case that a fault was detected but all candidates were eliminated, then one may assume either a false positive, or a true positive with incorrect fault isolation. We assume the latter, and in this case, we again conservatively recommend an abort.

3 Diagnosability

For a given model, through the qualitative fault isolation framework we can generate a set of fault signatures and relative residual orderings, which form, based on a set of residuals, the diagnostic information of the qualitative approach. Using the general model decomposition framework described in [Roychoudhury *et al.*, 2013], we can generate, given a global system model, a number of independent submodels for the purposes of residual generation. A submodel is defined by its subset of the variables and constraints of the global model. For defining residuals, the outputs of the submodels are the important variables. Given a set of (measured) outputs, we can generate a minimal submodel. For a set of m total outputs, we can define m single-output submodels, $\binom{m}{2}$ double-output submodels, and so on, and one submodel with all m outputs (i.e., the global model). For a system with m measurements the number of possible submodels is $2^m - 1$, and the number of unique residuals over all possible submodels is $m \times 2^{m-1}$.

In our qualitative fault isolation framework, deviations in residuals resulting from faults are abstracted into qualitative symbols that can be reasoned over. These symbolic abstractions are termed fault signatures [Mosterman and Biswas, 1999; Daigle *et al.*, 2009]. In order to define diagnosability, we must first formalize the fault isolation framework. The framework was originally presented in [Daigle *et al.*, 2009], and we extend and generalize it here to account for residuals from a set of submodels.

Definition 1 (Fault Signature)**.** A *fault signature* for a fault f and residual r, denoted by $\sigma_{f,r}$, is set of symbols representing potential qualitative changes in r caused by f at the point of the occurrence of f. The set of fault signatures for f and r is denoted as $\Sigma_{f,r}$.

The temporal order of the residual deviations can also be used as discriminatory information. The temporal order of residual deviations for a given model, termed *relative residual orderings*, are based on the intuition that fault effects will manifest in some parts of the system before others, and can be computed based on analysis of the transfer functions from faults to residuals [Daigle *et al.*, 2007].

Definition 2 (Relative Residual Ordering)**.** If fault f always manifests in residual r_i before residual r_j, then we define a *relative measurement ordering* between r_i and r_j for fault f, denoted by $r_i \prec_f r_j$. We denote the set of all residual orderings for f as $\Omega_{f,R}$.

Signatures and orderings can be generated by manual analysis of the system model, by simulation, or automatically from certain types of models, e.g., as presented in [Daigle, 2008]. Together, they establish an event-based form of diagnostic information. For a given fault, the combination of all fault signatures and residual orderings yields all the possible ways a fault can manifest in the residuals. Each of these possibilities is a *fault trace* [Daigle *et al.*, 2009].

Definition 3 (Fault Trace)**.** A *fault trace* for a fault f over residuals R, denoted by $\lambda_{f,R}$, is a sequence of fault signatures, of length $\leq |R|$ that includes, for every $r \in R$ that will deviate due to f, a fault signature $\sigma_{f,r}$, such that the sequence of fault signatures satisfies $\Omega_{f,R}$.

We group the set of all fault traces into a *fault language*:

Definition 4 (Fault Language)**.** The *fault language* of a fault $f \in F$ with residual set R, denoted by $L_{f,R}$, is the set of all fault traces for f over the residuals in R.

In our diagnosis framework, distinguishability between faults is characterized using fault traces and languages.

Definition 5 (Distinguishability)**.** Given a residual set, R, a fault f_i is *distinguishable* from a fault f_j, if there does not exist a pair of fault traces $\lambda_{f_i,R} \in L_{f_i,R}$ and $\lambda_{f_j,R} \in L_{f_j,R}$, such that $\lambda_{f_i} \sqsubseteq \lambda_{f_j}$.

One fault will be distinguishable from another fault if it cannot produce a fault trace that is a prefix[1] (denoted by $\sqsubseteq$) of a trace that can be produced by the other fault. If this is not the case, then when that trace manifests, the first fault cannot be distinguished from the second.

Distinguishability is used to define the diagnosability of a diagnosis model under a given fault isolation framework. A diagnosis model is an abstraction of a system model with only diagnosis-relevant information.

Definition 6 (Diagnosis Model)**.** A *diagnosis model* $\mathcal{S}$ is a tuple $(F, R, L_{F,R})$, where $F = \{f_1, f_2, \ldots, f_n\}$ is a set of faults, R is a set of residuals, and $L_{F,R} = \{L_{f_1,R}, L_{f_2,R}, \ldots, L_{f_n,R}\}$ is the set of fault languages.

If a diagnosis model is diagnosable, then we can make guarantees about the unique isolation of every fault in the diagnosis model.

Definition 7 (Diagnosability)**.** A diagnosis model $\mathcal{S} = (F, R, L_{F,R})$ is *diagnosable* if and only if $(\forall f_i,\ f_j \in F) f_i \neq f_j \implies f_i \not\sim_R f_j$.

If $\mathcal{S}$ is diagnosable, then every pair of faults is distinguishable using the residual set R. Hence, we can uniquely isolate all faults of interest. If $\mathcal{S}$ is not diagnosable, then ambiguities will remain after fault isolation, i.e., after all possible fault effects on the residuals have been observed.

4 Diagnosers

A qualitative fault diagnoser, in our framework, is then defined by the set of faults and a set of residuals. From the large residual space, any subset of residuals can, in theory, be selected. In practice, there is much redundant information over the residuals, and, therefore, only a subset are required to achieve an appropriate level of diagnosability. The three diagnostic algorithms we develop are based on three different qualitative fault diagnosers with different diagnosability properties.

[1] A fault trace λ_i is a prefix of fault trace λ_j if there is some (possibly empty) sequence of events λ_k that can extend λ_i such that $\lambda_i \lambda_k = \lambda_j$.

Table 1: Selected Fault Signatures for the QED algorithm for ADAPT-Lite

Fault	E240	E242	IT281	IT267	ST516
AC483 $\Delta p > 0$	+0X	+0X	+0X	-0X	00X
DC485 $\Delta p > 0$	+0X	+0X	-0X	00X	00X
E240 $\Delta p > 0$	+0X	00X	00X	00X	00X
E240 $m > 0$	0+X	00X	00X	00X	00X
E240 $\mu_{\Delta p} > 0$	+0X	00X	00X	00X	00X
EY244 stuck open	+0X	-0Z	-0Z	-0Z	0-X
FAN416 underspeed	+0X	+0X	00X	-0X	-0X

Table 2: Selected Fault Signatures for the QED-PC algorithm for ADAPT-Lite

Fault	E240	E242	IT281	IT267	ST516
AC483 $\Delta p > 0$	00X	00X	00X	-0X	00X
DC485 $\Delta p > 0$	00X	00X	-0X	00X	00X
E240 $\Delta p > 0$	+0X	-0X	00X	00X	00X
E240 $m > 0$	0+X	0-X	00X	00X	00X
E240 $\mu_{\Delta p} > 0$	+0X	-*X	00X	00X	00X
EY244 stuck open	00X	-0Z	00X	00X	00X
FAN416 underspeed	00X	00X	00X	-0X	-0X

4.1 QED

The first algorithm, QED, uses the set of residuals defined from the global model [Daigle and Roychoudhury, 2010], and is based on the extended Transcend approach [Daigle *et al.*, 2009]. Fault signatures for selected faults are shown in Table 1.

A diagnosability analysis reveals several instances where one fault cannot be distinguished from another. The first set of indistinguishable faults is for the four pairs of faults that produce exactly the same quantitative behavior on the given measurements: failures in CB262 and INV2, failures in EY281 and DC485, failures in EY272 and AC483, and failures in EY275 and FAN416. Therefore, based on only qualitative information they cannot be distinguished either. The second set of indistinguishable faults is between offset and intermittent faults, which produce the same initial transients. It is only through fault identification that they can be distinguished. The third set of inidistinguishable faults deals with sensors. For example, consider an offset in E240 (see Table 1). An abrupt increase will be observed in E240, and at this point, an offset in AC83, DC483, EY244 stuck, and FAN416 underspeed are all consistent. We have to wait infinitely long to verify that no other residuals will deviate in order to eliminate these faults.

So, when sensor faults occur, qualitative fault isolation alone cannot uniquely distinguish them, and fault identification is needed to resolve the ambiguities. Here, we take advantage of knowledge that if a nonsensor fault occurs, it should cause deviations in multiple residuals within finite time. Therefore, we create a heuristic fault isolation rule that implements this idea, e.g., if after 60 seconds only one residual has deviated, we eliminate all nonsensor faults. In this way, we can uniquely isolate sensor faults without fault identification, and improve fault isolation time.

4.2 QED-PC

The second algorithm, QED-PC, uses the set of residuals defined from the minimal single-output submodels [Daigle *et al.*, 2012], and is based on the Possible Conflicts approach [Pulido and Alonso-González, 2004], augmented with the qualitative fault isolation framework [Bregon, 2010]. Signatures for selected faults are shown in Table 2.

The main advantage of the PC-based residuals is that, like ARRs and MSOs, they decouple faults from residuals, so faults affect only a subset of the residuals [Armengol *et al.*, 2009]. In most cases, the decoupling increases diagnosability. For example, if each fault affects only a single unique residual, the system is diagnosable even in the multiple fault case. For QED-PC, sensor faults can now be easily distinguished, because they affect mutliple residuals. The reason is that sensed values are used as inputs, so a sensor fault will cause a deviation in the residual for the PC that computes the output as well as any PCs that use the sensor's values as an input. However, now there is a diagnosability problem with nonsensor faults. For example, a fault in AC483 affects only a single residual. So, like sensor faults with QED, we have to wait infinitely long to ensure it is not some other fault that produces a deviation in that residual.

Similarly to QED, we can improve diagnosability by introducing some heuristic fault isolation rules for these faults. In this case, we can assume that sensor faults would produce deviations on more than one residual in finite time. This will allow faults like those in AC483 and DC485 to be distinguished without resorting to fault identification, and improve fault isolation times.

4.3 QED-PC++

The diagnosability analyses of QED and QED-PC reveal the weaknesses of the individual residual sets. When sensor faults occur, QED cannot distinguish them from nonsensor faults. When some nonsensor faults occur, on the other hand, QED-PC cannot distinguish them from sensor faults. This suggests that an algorithm using the combined residual sets of QED and QED-PC can resolve these ambiguities and lead to improved fault isolation. This is the residual set used by the third algorithm, QED-PC++.

We can see now that the diagnosability weaknesses of QED and QED-PC are resolved by a combined residual set. When a sensor fault occurs, only one global model residual will deviate, but multiple PC-based residuals will deviate. For the indistinguishable nonsensor faults in QED-PC, only one PC-based residual will deviate but multiple global model residuals will deviate. This eliminates the need for most of the heuristic fault isolation rules and improves fault isolation times.

5 Results

In order to demonstrate the differences between the diagnosers, we describe a demonstration scenario in which a drift fault in E242 is injected at 175 s with a slope of 0.075. Measured and predicted values for relevant outputs are shown in Figs. 2–5. QED detects the fault at 176.7 s, with an increase in the E242 residual. The possible faults are in AC483, CB262, DC485, E242, EY260, EY272, EY275, EY284, FAN416 (failed off or underspeed), and INV2. At 176.9 s the stuck mode of E242 is eliminated since consecutive measurements were of different values. At 177.1 s the discrete change symbol is computed as X, which does not change the candidate list. At 179.2 s, the slope of E242

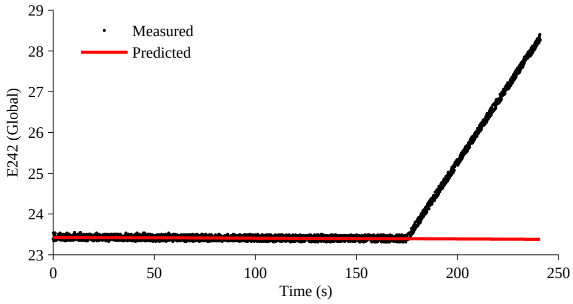

Figure 2: Measured and predicted values of E242 (global model) for E242 drift fault.

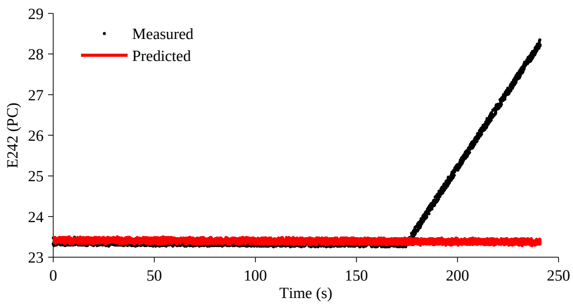

Figure 3: Measured and predicted values of E242 (PC) for E242 drift fault.

is computed as +, eliminating all candiates but AC483 resistance drift, DC485 resistance drift, and E242 drift. At 220 s, since only one residual had deviated, it is concluded to be a sensor fault and E242 drift is isolated. The injection time is computed as 169.8 s and the magnitude as 0.069. The recommended action is ∅, which is correct.

QED-PC detects the fault at 178.1 on the PC for E242. The thresholds for the PCs are larger, so fault detection is slower than with QED. The initial candidate list consists only of faults in E240 and E242, since the remaining faults are decoupled from the E242 residual by the PC design. This is in constrast to QED, where most components except sensors were implicated with the first residual deviation. At 178.3, it is determined that neither E240 or E242 are stuck. At 182.1 the discrete change symbol for E242 is computed as X, which does not change the candidate list. At 182.1, a decrease in the E281 PC residual and increase in the IT240 PC residual are detected, isolating the fault to

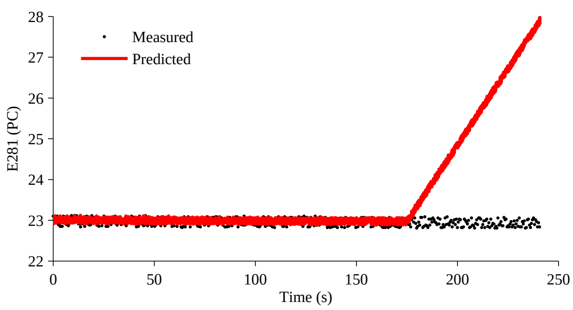

Figure 4: Measured and predicted values of E281 (PC) for E242 drift fault.

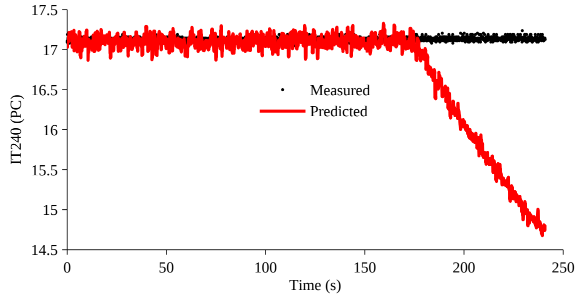

Figure 5: Measured and predicted values of IT240 (PC) for E242 drift fault.

E242 (drift or offset). At 183.1, the slope on E242 is computed as +, therefore isolating E242 drift as the fault. The injection time is computed as 169.7, and the slope as 0.069. The recommended action is ∅.

QED-PC++ detects the fault at 176.7 with the E242 global model residual. The initial candidate list is the same as with QED - faults are in AC483, CB262, DC485, E242, EY260, EY272, EY275, EY284, FAN416 (failed off or underspeed), and INV2. At 176.9 it is determined that E242 is not stuck. At 177.1 the discrete change symbol for E242 is computed as X, and the candidate list remains unchanged. At 178.1, an increase in the residual for the E242 PC is detected, thus eliminating all candidates except for faults in E242. Thus, fault isolation is completed far earlier than QED (about 30 s), and also earlier than QED-PC (about 5 s). The fault is computed as a drift with an injection time of 169.8 and a slope of 0.069. The recommended action is ∅.

Overall results are shown in Table 3. The metrics consist of the mean time to detect faults M_{fd}, the mean false negative rate M_{fn}, the mean false positive rate M_{fp}, the detection accuracy M_{da}, the mean time to isolate faults M_{fi}, the number of classification errors M_{err}, the mean CPU time M_{cpu}, the mean peak memory usage M_{mem}[2], and the overall recovery cost M_{rc}. Both QED and QED-PC are improved from their performance in DXC'11 (see [Poll *et al.*, 2011]). One source of previous errors was data spikes, which caused false alarms when occurring before the fault and incorrect symbol generation when occurring after the fault. The solution was to use a median filter over 3 samples. As a result, residual thresholds had to be increased slightly for some sensors. Other changes in included minor fixes to the fault signatures, and further tuning to residual thresholds to decrease sensitivity and avoid false alarms and incorrect symbol generation.

Overall, all algorithms do very well. QED and QED-PC++ have errors on the same scenarios. There are 12 errors due to the cases for the nondistinguishable faults (e.g., DC485 failing vs its relay failing). The remaining errors are due to incorrect fault mode identification (e.g., identifying as intermittent instead of drift) and missed detections. The missed detections are acceptable, because the faults are small enough that the correct action is ∅. The incorrect mode identification scenarios also did not result in a bad recommendation. The one scenario that did was one where the identified fault parameters were off just enough so that

[2]Over multiple runs, CPU time and memory usage will vary and within the statistical deviation, M_{cpu} and M_{mem} can be considered equivalent for all three algorithms.

Table 3: QED Diagnosis Results

DA	M_{fd} (s)	M_{fn}	M_{fp}	M_{da}	M_{fi} (s)	M_{err}	M_{cpu} (ms)	M_{mem} (kb)	M_{rc}
QED	9.38	0.0152	0.0	0.987	125.61	19	22.72	7675	25
QED-PC	14.66	0.025	0.0	0.978	127.70	37	23.70	7743	275
QED-PC++	9.32	0.0152	0.0	0.987	124.94	19	24.49	7835	25

an abort was recommended when the correct action was $\varnothing$.

QED-PC had similar errors, and also some additional ones in which the DC485 and IT281 faults were confused. This is a difficult situation to correct, because the thresholds were difficult to tune. DC485 can only be uniquely isolated if only IT281 deviates, so if we see additional deviations we conclude it is an IT281 fault. However, sometimes IT281 did not cause large enough deviations in other residuals so it was misdiagnosed as DC485. These scenarios did not result in a bad recommendation, though. In two cases, QED-PC recommended $\varnothing$ when the correct action was to abort. In one case, an ST516 fault was misdiagnosed as an E265 fault. This error is due to threshold sensitivities (for more discussion, see [Daigle *et al.*, 2012]). In the other case, an ST516 intermittent offset fault was misdiagnosed as an offset fault, for which the identified offset was not enough to trigger an abort recommendation.

6 Conclusions

In this paper, we presented three diagnostic algorithms based on a common qualitative fault isolation framework. The algorithms differ in which residual generators are used for fault detection and isolation. We showed that a combination of global model and PC-based residuals offers the best diagnosability and fault isolation performance, as confirmed with experimental results.

Although the diagnostic algorithms do fairly well, there are several areas that can be improved. For one, it was found that tuning thresholds for slope generation was very difficult. It would most likely be easier to avoid using that symbol, and rely only on fault identification to sort through the possible component modes. That would make the algorithms easier to use up front and eliminate the chances of incorrect symbol generation. Mode identification was also an issue, and it was difficult to tune the discrete-valued tests that determined which fault mode was present. A more general, probability-based approach would be more favorable, that, for instance, ranks the possible modes according to how best they fit the data.

References

[Armengol *et al.*, 2009] J. Armengol, A. Bregon, T. Escobet, E. Gelso, M. Krysander, M. Nyberg, X. Olive, B. Pulido, and L. Travé-Massuyès. Minimal structurally overdetermined sets for residual generation: A comparison of alternative approaches. In *Proc. of the 7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, Barcelona, Spain, 2009.

[Bregon *et al.*, 2012] A. Bregon, G. Biswas, and B. Pulido. A decomposition method for nonlinear parameter estimation in TRANSCEND. *IEEE Trans. Syst. Man. Cy. Part A*, 42(3):751–763, 2012.

[Bregon, 2010] A. Bregon. *Integration of FDI and DX Techniques within Consistency-based Diagnosis with Possible Conflicts*. PhD thesis, University of Valladolid, 2010.

[Daigle and Roychoudhury, 2010] M. Daigle and I. Roychoudhury. Qualitative event-based diagnosis: Case study on the second international diagnostic competition. In *Proceedings of the 21st International Workshop on Principles of Diagnosis*, pages 371–378, October 2010.

[Daigle *et al.*, 2007] M. J. Daigle, X. D. Koutsoukos, and G. Biswas. Distributed diagnosis in formations of mobile robots. *IEEE Trans. on Robotics*, 23(2):353–369, April 2007.

[Daigle *et al.*, 2009] M. J. Daigle, X. Koutsoukos, and G. Biswas. A qualitative event-based approach to continuous systems diagnosis. *IEEE Trans. on Control Systems Technology*, 17(4):780–793, July 2009.

[Daigle *et al.*, 2012] M. Daigle, A. Bregon, and I. Roychoudhury. Qualitative event-based diagnosis with possible conflicts applied to spacecraft power distribution systems. In *Proceedings of the 8th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, pages 265–270, August 2012.

[Daigle, 2008] M. Daigle. *A Qualitative Event-based Approach to Fault Diagnosis of Hybrid Systems*. PhD thesis, Vanderbilt University, 2008.

[Mosterman and Biswas, 1999] P. J. Mosterman and G. Biswas. Diagnosis of continuous valued systems in transient operating regions. *IEEE Trans. on Systems, Man and Cybernetics, Part A*, 29(6):554–565, 1999.

[Poll *et al.*, 2010] S. Poll, J. de Kleer, A. Feldman, D. Garcia, T. Kurtoglu, and S. Narasimhan. Second international diagnostics competition – DXC'10. In *Proceedings of the 21st International Workshop on Principles of Diagnosis*, October 2010.

[Poll *et al.*, 2011] S. Poll, J. de Kleer, R. Abreau, M. Daigle, A. Feldman, D. Garcia, A. Gonzalez-Sanchez, T. Kurtoglu, S. Narasimhan, and A. Sweet. Third international diagnostics competition – DXC'11. In *Proc. of the 22nd International Workshop on Principles of Diagnosis*, pages 267–278, October 2011.

[Poll *et al.*, 2007] S. Poll *et al.* Evaluation, selection, and application of model-based diagnosis tools and approaches. In *AIAA Infotech@Aerospace 2007 Conference and Exhibit*, May 2007.

[Pulido and Alonso-González, 2004] B. Pulido and C. Alonso-González. Possible conflicts: a compilation technique for consistency-based diagnosis. *IEEE Trans. on Systems, Man, and Cybernetics, Part B*, 34(5):2192–2206, 2004.

[Roychoudhury *et al.*, 2013] I. Roychoudhury, M. Daigle, A. Bregon, and B. Pulido. A structural model decomposition framework for systems health management. In *Proc. of the 2013 IEEE Aerospace Conf.*, March 2013.