Comparative study of the multiple fault isolation approaches based on the structures of residual sets

Michał Bartysiński
Institute of Automatic Control and Robotics, Warsaw University of Technology, św. Boboli 8, 02-525 Warsaw, Poland
E-mail: bartysi@mchtr.pw.edu.pl

Abstract. This paper presents a comparative study of the applicability features of the three chosen multiple fault isolation approaches based on the binary evaluated structures of residual sets. Firstly, all considered approaches are briefly described. Then, their computational effectiveness is discussed. Next, the low-level implementation of the compared approaches is characterised and the set of experimental results is presented. Finally, the results of experiments are discussed and the appropriate recommendations for diagnostic engineering practice are formulated.

1. Introduction
The goal of this paper is quite practical. It is aimed to give a theoretical background for a well documented rules of how to make reasonable choice of multiple fault isolation approach for real implementations. The scope of this paper covers three chosen multiple fault isolation approaches based on a binary evaluated structures of residual sets.

According to the definition given in [1], the fault isolation is the determination of the kind, location and time of detection of a fault. In this paper, we will principally refer to the multiple fault location abstracting from their detection time and considering a fault kind as a linguistic conjunction of a single fault labels.

The problem of multiple (simultaneous faults) arises particularly when there is no possibility or there is not enough time or capacities to fix all previously presented single faults. Intuitively, the multiple faults are that the more probable the more complex the process is. As far as the multiple faults are observed, the symptoms of single faults are overlapping each other. As long as combined symptoms of the single faults do not match any single fault signature, the hypothesis regarding multiple faults is reasonable. Otherwise, the multiple faults may introduce the effect referred to as fault cancellation. This rare phenomenon relies on the cancellation of a single fault residual as the result of the mutual overlapping of the truly existing faults. The cancellation of residual of a fault makes fault isolation process unreliable.

The multiple fault isolation problems have been studied mainly from the Fault Detection and Isolation (FDI) and Artificial Intelligence (AI) perspectives [2, 3, 4, 5]. The FDI methods are principally focused on generation and processing of residuals sensitive to particular faults and insensitive to others. The Fault Detection (FD) deals with the methods of residual generation while the Fault Isolation (FI) deals with location of faults, time of their origin and their kind. FI approaches make use from residuals that are transformed to the form of diagnostic signals in the
process referred to as residual evaluation. Different threshold approaches (constant, adaptive, fuzzy, statistic) are applied on this stage \[6, 7, 8, 9\] in order to yield the diagnostic signals. The particular diagnostic signals which are indicative for the given faults are referred to as the fault symptoms. The Artificial Intelligence (AI) or consistency based methods are principally focussed on fault isolation based on reasoning from the first principles i.e., using knowledge of structure and behavior of the system. Some of AI multiple fault approaches \[4, 5, 10, 11\] are searching for the minimal sets of faults which are consistent with the observed symptoms according to the parsimony principle. Some of them are searching for all probable multiple faults despite their probability \[12, 13\].

In this paper, we will provide a deep-in comparative study of the properties of the three chosen approaches belonging to the FDI group and based on the binary evaluated residuals.

2. Fault isolation system structure

Consider a model based diagnostic system. Let a finite set \( F \) of the faults \( f_i \) in this system be defined as:

\[
F = \{ f_i : i = 1..n \}
\]

and a finite set of diagnostic signals \( S \):

\[
S = \{ s_j : j = 1..m \}.
\]

Let us now discuss the relation between both sets in the form of Cartesian product \( R_{FS} \subset F \times S \). Here, the relation \( R_{FS} \) constitutes a set of \( n \cdot m \) ordered pairs (bi-element relations) \( (f_i, s_j) \). According to the geometrical interpretation of the Cartesian product, the relation is the set of \( n \cdot m \) points of the plane defined in \( F \) and \( S \) coordinates. It is possible to spread out a three-dimensional mesh over the \( F \times S \) plane by attributing diagnostic signal values \( v_{j,i} \) of all diagnostic signals \( s_j \) for all \( f_i \) faults. The three-dimensional mesh is easily transformable into the form of a two-dimensional \( m \cdot n \) matrix \( V \) of the diagnostic signal values \( v_{j,i} \):

\[
V = \begin{bmatrix}
v_{1,1} & \cdots & v_{1,n} \\
\vdots & \ddots & \vdots \\
v_{m,1} & \cdots & v_{m,n}
\end{bmatrix} = [v_{j,i}]_{m \times n}
\]

The matrix \( V \) forms a fault isolation system structure \[12\]. More comprehensively, this matrix is referred to as a diagnostic matrix. If the diagnostic signals are bi-valued then the matrix \( V \) is called as a Binary Diagnostic Matrix (BDM) \[9\]. For convenience, the diagnostic matrix \( V \) will be further represented as a block matrix of \( n \) vectors \( V_i \):

\[
V = [V_i]_{n \times 1},
\]

such that each vector \( V_i \) contains the set of diagnostic signals values associated exclusively with the single \( i \)-th fault:

\[
V_i = [v_{1,i}, v_{2,i}, \ldots v_{j,i}]^T.
\]

The specific vector \( V_i \) that contains all diagnostic signals associated with a particular fault builds up a numerical or symbolic signature (pattern) of this fault.

Consider a diagnostic system for which a binary diagnostic matrix is known. Let the output of this system in a time instant \( t \) be a set of residual values \( R_t = \{ r_j^t : j = 1..m \} \). Let now the set of residuals \( R_t \) be transformed onto the set of values of diagnostic signals \( V_t = \{ v_j^t : j = 1..m \} \) in result of binary evaluation of residuals. The ordered set of diagnostic values \( V_t \) in the time instant \( t \) will be referred to as the vector of actual diagnostic signal values.
The vector $V^t$ can match one or even more single fault signatures in the diagnostic matrix. If this is the case, than this specific vector may simply indicate single faults. However, please pay attention that the vector of actual diagnostic signals values $V^t$ may be understood as a vector that points out multiple faults too. All of the multiple fault isolation approaches considered in this paper [12, 13, 14] assume that probability of the multiple faults compared to the single faults is negligible. Hence, if the vector of actual diagnostic signals values $V^t$ matches the single faults then the multiple faults are not considered and searched.

However, the question arises, what happens if the vector $V^t$ does not match any fault signature in a fault isolation structure. The possible answers are as follows:

(i) there is no fault;
(ii) there are unknown faults;
(iii) there are multiple faults.

Case (i) is trivial. It takes place when all values of the actual vector of diagnostic signals are equal to 0. This however, does not exclude a case of unknown faults. Case(ii) indicates poor design of the diagnostic system and will not be further discussed. Case (iii) is the main concern of this paper.

3. The classic approach to multiple fault isolation

The approach to multiple fault isolation proposed by Gertler in [12] will be further referred to as a classic approach. Let us take into consideration the following set of assumptions:

1° a bi-valued diagnostic matrix of the diagnosed system is known;
2° all single faults are defined;
3° the signatures of all single faults are known;
4° the residual cancellation effects do not take place;
5° the signature of a multiple fault is the logical alternative of single fault signatures.

The above given assumptions are acceptable to some extent in practice because the residual cancellation effects rather have marginal meaning. Moreover, the diagnosed systems having duly prepared diagnostic matrices are not exceptional. Obviously, if the assumptions are not met than the diagnostic system should notify users that fault isolation results may be false.

The starting point of this approach is quite a trivial statement that the maximal fault multiplicity does not exceed the number $n$ of single faults. Therefore, searching for the multiple faults might be arranged as an iterative process of searching for single, double, triple, ..., and $n$–multiple faults. In fact, each search needs to identify all combinations of single fault signatures which are consistent with the vector $V^t$ of actual diagnostic signal values. Therefore, the total number of all possible combinations of all single fault signatures equals:

$$N = 2^n - 1.$$  \hfill (6)

From (6), it follows that the number of searches for multiple faults grows exponentially with the cardinality of the set of faults $F$. Therefore, the substantial drawback of this approach is a huge computational effort that must be paid to find out all multiple faults. On the other hand, its most important advantage is an extremely easy implementation. It is obvious, that this approach may not be considered as a fast if the number of fault signatures is significant.

Conclusion 1: The efficiency of a classic multiple fault isolation approach depends exclusively on the number of columns of the $BDM$ matrix.

The classic approach is based on matching logical alternatives of signatures of single faults with the $V^t$ pattern. Clearly, the zero values of a diagnostic signals do not influence the
logic alternatives of signatures. Therefore, the binary diagnostic matrix may be compressed dynamically in horizontal direction by removing this \( V_i \) columns for which holds:

\[
\bigvee_{j=1}^{m} (v_{j,i} \land \neg v_{j}^{t}) = 1; \quad \forall i = \{1,..,n\}. \quad (7)
\]

This, in general, decreases the number of signatures of the diagnostic matrix. Let the symbol \( c \) denotes the number of rejected columns. Hence, the number of combinations of all single fault signatures in relation to the number \( N \) from (6) will drop down approximately \( 2^c \) times:

\[
\eta = \frac{2^N - 1}{2^{n-c} - 1} \approx 2^c. \quad (8)
\]

**Observation 1:** The horizontal compression of the binary diagnostic matrix is beneficial in that sense that may speed up the multiple fault isolation.

**Observation 2:** The binary diagnostic matrix may be further compressed in vertical direction. It is easy to note that irrelevant are all those rows of the diagnostic matrix for which holds:

\[
v_{j}^{t} = 0; \quad \forall j = \{1,..,m\} \quad (9)
\]

**Observation 3:** As results from (6), the vertical compression of the binary diagnostic matrix does not influence the total count of all possible combinations of single fault signatures.

4. The **MUFIA** approach

Generally speaking, the classical algorithm of multiple fault isolation is ineffective. Much more effective is Multiple Fault Isolation Approach (**MUFIA**) algorithm[14]. Principally, the **MUFIA** approach consists of two parts. In the first part, the set of possible faults is determined. In the second part, the classic approach to multiple fault isolation is applied with the use of horizontal and vertical squeezing of the **BDM** matrix which are described in Sect. 3.

4.1. The first part of the approach: identification of possible faults

The following assumptions are adopted for the **MUFIA** approach:

1° all signatures of all single faults are known;
2° if a fault occurs then all symptoms related to this particular fault will appear;
3° multiple faults cause a mismatch in the set of diagnostic signal values;
4° there is no match between vector of diagnostic signals and signatures of single faults.

At the beginning, the presence of single faults is checked based on a classic fault isolation algorithm described in Sect. 3. Here, the computational effort is relatively low and moreover straight proportional to the total number of the **BDM** columns. In case if the single faults are localised then the **MUFIA** algorithm is considered as completed. Otherwise, according to assumption (3°) a search for a multiple faults starts-up.

The search for multiple faults begins with the decomposition of the diagnostic matrix into dynamic subsystems according to Dynamic Table of States approach (**DTS**) proposed by Kościelny in [15]. Each dynamic subsystem is defined by a set of possible faults and a set of diagnostic signals sensitive to these faults. The subsystem is created and updated immediately after the appearance of any subsequent fault symptom.

Let us assume that the first fault symptom \( s_{j}^{1} = 1 \) will trigger the procedure that creates a subset \( F^{1} \) consisting of all faults for which \( s_{j}^{1} = 1 \):

\[(s_{j}^{1} = 1) \Rightarrow F^{1}; \quad j \in [1..m]. \quad (10)\]
Next, a specific subset \( S^1 \) of diagnostic signals is created. This subset contains all diagnostic signals that are necessary for the isolation of the faults belonging to the subset \( F^1 \):

\[
S^1 = \{ s_j \in S : \forall f_i \in F^1 \} ; j \in [1..m].
\]  

(11)

Clearly, the Cartesian product \( F^1 \times S^1 \) finally defines a dynamic binary diagnostic subsystem. Usually, the size of this subsystem is significantly reduced in comparison to the primary \( BDM \). A reduced diagnostic matrix contains only \( |S^1| \) rows and \( |F^1| \) columns. The next fault symptom will trigger the next procedure that similarly assigns faults and diagnostic signals to the appropriate subsets \( F^2 \) and \( S^2 \). If the subsets \( S^1 \) and \( S^2 \) of the diagnostic signals are disjunctive:

\[
S^1 \cap S^2 = \emptyset
\]

(12)

then multiple faults can be isolated correctly in separate fault isolation processes. In this case, the subsets of possible faults \( F^1 \) and \( F^2 \) are also disjunctive. This allows for significant acceleration of multiple fault isolation because of two reasons: lower size of diagnostic subsystems and possibility of application of parallel processing of multiple fault isolation in each subsystem.

However, truly, there is no reason to assume that dynamically created subsets of diagnostic signals must always be disjunctive. Moreover, the benefits of application of DDS approach are disputable in case of some applications, where the size of \( BDM \) matrices is low. This applies for example for the embedded diagnostics of smart final control devices.

In this case it is still possible to apply the horizontal and vertical compression of the size of the \( BDM \), according to (7) and Obs. 2. As results from (8), particularly valuable are those actions, that reduce the maximal number of columns of \( BDM \) without the loss in the set of multiple faults. Originally there is the horizontal as well as vertical compression applied in the \( MUFIA \) approach. According to (7), the total count \( c \) of removed columns is equal to the cardinal number of the set \( C \) consisting of indices of excluded columns.

\[
c = |C| ; C = \{ i : i \in \{1, \ldots, n\} : \bigvee_{j=1}^{m} (v_{j,i} \land \neg v^j) = 1 \}
\]

(13)

**Conclusion 2:** The number of columns which may be removed from \( BDM \) matrix without the loss of a multiple fault information depends on the distribution of its entries equalling 1 \((v_{j,i} = 1)\) and on the distribution of the entries equalling 0 \((v^j = 0)\) of the vector of actual diagnostic signal values \( V^t \).

**Conclusion 3:** Because the number of combinations of all single fault signatures in relation to the number \( N \) (8) drop down approximately \( 2^c \) times, then the \( MUFIA \) approach promises faster handling of multiple faults compared to conventional approach presented in Sect. 3.

**Conclusion 4:** The \( MUFIA \) approach is more effective the more rich the \( BDM \) matrix is and the more sparse the vector \( V^t \) is.

**Observation 5:** It is easy to see that in the case of absence of faults i.e. \((v^j = 0 ; \forall j \in \{1 \ldots m\})\) the number of rejected columns is maximal and \( c = n \) because:

\[
\exists ! v_{j,i} = 1 ; \forall j \in \{1 \ldots m\}, \forall i \in \{1 \ldots n\}
\]

(14)

In order to meet requirements of mathematical formalism let us now define conversion \((v \Rightarrow d)\) of the binary number \( v \) into integer number \( d \):

\[
\begin{align*}
&v_{j,i} = 0 \Rightarrow d_{j,i} = 0 ; \quad v^j = 0 \Rightarrow d^j = 0 \\
&v_{j,i} = 1 \Rightarrow d_{j,i} = 1 ; \quad v^j = 1 \Rightarrow d^j = 1
\end{align*}
\]

(15)
Let us consider a BDM according. Usually not.

Observation 6. Despite the underlaying theory, presented in [14] and its extensions given above, it is obvious that BDS is at least a compressed dynamically BDM matrix in horizontal (7) as well as in vertical (9) direction.

Example 1. The first part of an approach.

Let us consider a BDM matrix and a vector \( V^t \) as in Tab. 1. Firstly, we apply the dynamic decomposition approach. The following subsets of faults and diagnostic signals are obtained according to (10–11):

\[
F^2 = \{ f_1, f_3, f_4, f_5 \}, \quad S^2 = \{ s_1, s_2, s_3, s_4, s_6 \},
\]

\[
F^3 = \{ f_1, f_3, f_4 \}, \quad S^3 = \{ s_1, s_2, s_3, s_4, s_6 \},
\]

\[
F^4 = \{ f_2, f_3, f_5 \}, \quad S^4 = \{ s_1, s_2, s_3, s_4, s_5, s_6 \},
\]

\[
F^6 = \{ f_1, f_3, f_4 \}, \quad S^6 = \{ s_1, s_2, s_3, s_4, s_6 \}.
\]

As it can be easily seen, all subsets of diagnostic signals \( S^j \) are not disjunctive. Therefore, according to condition (12), the application of the dynamic decomposition approach is not possible. Let us now create a fault isolation subsystem based on formulas (7 and 9). Firstly, the columns are removed in accordance with (7). The result is depicted in the intermediate matrix in Tab. 1. Next, the rows of intermediate matrix are rejected in accordance with (9). The result is depicted in the right hand matrix in Tab. 1. Please note that \( V^* \) is a column vector for which holds: \( \bigwedge_{j=1}^m v^*_j = 1 \). □

4.2. The second part of the approach: isolation of multiple faults

This part of the MUFIA relies on searching for multiple faults in the BDS submatrix with the use of classic approach described in Sect. 3. Theoretically, all possible multiple faults can
Table 1. An example of a formation of a BDS.

| S/F | f_1 | f_2 | f_3 | f_4 | f_5 | V^t | S/F | f_1 | f_3 | f_4 | V^t | S/F | f_1 | f_3 | f_4 | V^t |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| s_1 | 0   | 1   | 0   | 0   | 1   | 0   | s_1 | 0   | 0   | 0   | 0   | s_2 | 1   | 1   | 1   | 1   |
| s_2 | 1   | 0   | 1   | 1   | 1   | 1   | s_2 | 1   | 1   | 1   | 1   | s_3 | 1   | 1   | 1   | 1   |
| s_3 | 1   | 0   | 1   | 1   | 0   | 1   | s_3 | 1   | 1   | 1   | 1   | s_4 | 1   | 0   | 1   | 1   |
| s_4 | 0   | 1   | 1   | 0   | 1   | 1   | s_4 | 0   | 1   | 0   | 1   | s_5 | 1   | 0   | 0   | 1   |
| s_5 | 0   | 1   | 0   | 0   | 0   | 0   | s_5 | 0   | 0   | 0   | 0   | s_6 | 1   | 1   | 1   | 1   |
| s_6 | 0   | 0   | 1   | 1   | 0   | 1   | s_6 | 1   | 1   | 1   | 1   |

be searched. However, as stated in [14] the searching may be stopped by any predefined fault multiplicity \( K \leq n^* \). In this case, the number \( N^* \) of possible multiple faults equals:

\[
N^* = \sum_{k=1}^{K} \binom{n^*}{k}.
\]

(21)

If \( K = n^* \) then the total number of possible multiple faults equals \( N^* = 2^{n^*} - 1 \).

Example 2. The second part of the MUFIA approach.
Let us continue reasoning regarding the multiple faults from Ex. 1. The set of possible faults \( F^* = \{f_1, f_3, f_4\} \) and a set of diagnostic signals \( S^* = \{s_2, s_3, s_4, s_6\} \) were obtained in the first part of the MUFIA approach. The possible multiple faults together with their hypothetic signatures are shown in Tab. 2. The hypothetic signatures were created in accordance with assumption (2o) in Sect. 4.1. As can be seen from the Tab. 2, vector \( V^* \) matches one single fault \( \{f_3\} \), two double faults \( \{f_1f_3\} \) and \( \{f_3f_4\} \) and one triple fault \( \{f_1f_3f_4\} \). □

Table 2. Matrix of all possible combinations of single fault signatures in BDS.

| S^*/F^* | f_3 | f_1f_3 | f_1f_4 | f_3f_4 | f_1f_3f_4 | V^* |
|---------|-----|--------|--------|--------|-----------|-----|
| s_2     | 1   | 1      | 1      | 1      | 1         | 1   |
| s_3     | 1   | 1      | 1      | 1      | 1         | 1   |
| s_4     | 1   | 1      | 0      | 1      | 1         | 1   |
| s_6     | 1   | 1      | 1      | 1      | 1         | 1   |

5. The MFI approach
The MUFIA algorithm is very effective, particularly if condition (12) is met. If not, it is still effective by implementation of a powerful workbench of the BDM matrix compressing tools. Particularly effective is the horizontal compression which significantly cuts down computational effort (16). But the MUFIA approach has at least one serious drawback. Still, its numerical complexity depends on a combinatory manner on the number of columns compressed in the BDM (19). This allows to think that the MUFIA approach is particularly useful in cases where a relatively low fault multiplicities are of concern or where the number of the BDM matrix columns is low. This tends to be an indication of its usefulness for applications in large scale systems, where higher fault multiplicities and rich sets of faults are not exceptional.
In this section, we will briefly introduce and discuss a multiple fault isolation approach \textit{MFI} [13] which takes inspiration from the \textit{MUFIA} approach and propose an alternative solution. The main objective of this section is to present, prove and define conditions of the computational efficiency of the \textit{MFI} approach.

Firstly, the main savings in computational effort have to be achieved in the \textit{MFI} by the application of an approach in which checking of all hypothetic multiple faults is not necessary at all, as in \textit{MUFIA} or classic approaches. Instead of, the \textit{MFI} simply generates the sets of multiple faults.

Secondly, the \textit{MFI} approach is based on an idea of composition (synthesis) of multiple faults instead of analysis of possible multiple faults as in \textit{MUFIA} and classic approaches.

Thirdly, the \textit{MFI} approach is based on a stepwise composition of the multiple faults in vertical instead of horizontal direction of the \textit{BDS} submatrix.

\subsection*{5.1. Basics of \textit{MFI} approach}

In fact, the \textit{MFI} rises from the same root as the \textit{MUFIA} approach. The assumptions presented in Sect 4.1 as well as in the first part of the approach are identical. Completely different is the second part of the approach. Even the sketchy analysis indicates that this part has dominant influence on the effectiveness of the approach. Therefore, we will only refer to this part of the \textit{MFI} approach.

The \textit{MFI} generates multiple fault tracks making use of the features of the matrix of paths introduced in [13]. Each path of this matrix points out exactly one multiple fault. Therefore, it is sufficient to define this matrix and determine all its paths.

Let path $p_p$ be any set of non-zero elements of the matrix $BDS[m^*; n^*]$ leading through its $m^*$ rows and such that each element of this path belongs to each separate row of the \textit{BDS}.

$$p_p = \{v_{j,i} \in BDS: v_{j,i} \neq 0; \forall i \in \{1, n^*\}; \forall j \in \{1, m^*\}; |p_p| = m^* \}$$

The total number of all paths $p_p$ in the \textit{BDS} is equal to:

$$N_p = \prod_{j=1}^{m^*} \sum_{i=1}^{n^*} d_{j,i} \quad (23)$$

In order to minimize the number of searched paths, the non-zero elements of the \textit{BDS} matrix may shift left. Thus in general, the number of elements in each row of a squeezed matrix might be different and is equal to the number of ones in each row of the \textit{BDS}. Let us denote by $n_p$ the maximal number of ones in any row of the squeezed \textit{BDS} matrix:

$$n_p = \max_{j=1}^{m^*} \left( \sum_{i=1}^{n^*} d_{j,i} \right) \quad (24)$$

The theoretical maximal number $N_p$ of all paths is easy to obtain from (23) by substitution $\sum_{i=1}^{n^*} d_{j,i} = n_p$. Hence:

$$N_p = n_p^{m^*} \quad (25)$$

At least for a part of \textit{BDS} matrices holds: $n_p < n^*$. Those matrices might be further squeezed. This allows for additional horizontal compression of the previously compressed \textit{BDS} matrix and thus, promises further reduction of computational effort. The only condition is, that squeezing operation cannot lose information about multiple faults which is, in fact, encoded in the structure of the \textit{BDS}. Therefore, before being squeezed, all non-zero values from each column of the \textit{BDS} will be replaced by their corresponding fault labels. Each sequence of $m^*$
single fault labels belonging to different rows forms a multiple fault label. Clearly, the order of single fault labels in the multiple fault label is irrelevant. This enables a significant reduction of the maximum number of relevant paths in comparison to (25). The maximal number of all multiple fault labels equals:

$$N_p^r = n_p m^*$$  \hfill (26)

The formula (26) indicates proportional dependence of the maximal number of paths to be searched to the product of the number of columns \(n_p\) and the number of rows \(m^*\) of the squeezed matrix of paths. In contrast, the number of possible paths to be searched, that is indicated by the same BDS matrix rise exponentially in respect to the number of maximal theoretical fault multiplicity \(n^*\) as indicate (21) and (25). Please keep in mind that, the fault symbol which is repeated in the same path does not increase fault multiplicity i.e. \(\{f_1f_1f_2f_2\} \equiv \{f_1f_2\}\). On the contrary, each repeated fault symbol reduce fault multiplicity in particular path by one. In other words, the real count of multiple faults may be significantly lower then indicated in (26).

In the worst case the BDS matrix cannot be squeezed. But, if the BDS matrix is not squeezable, i.e. \(n_p = n^*\), then it does not give an assumption to state that the MUFIA approach is more effective. It is obvious, that in this case:

$$N_p^r \leq N^*.$$

Therefore, as can be derived from (26), it is sufficient to hold condition:

$$n^* m^* \leq 2^{n^*} - 1.$$

(28)

to expect better effectiveness of the MFI over the MUFIA approach.

**Conclusion 5:** As results from (28), the MFI in comparison to the MUFIA approach, is particularly effective if the fault multiplicity is greater or equal to 3 because usually \(m^* \approx n^*\).

**Example 3.** An example of the MFI approach.

Firstly let us consider the BDS matrix as in Tab. 1 (right hand matrix). The operation of horizontal squeezing is not possible for this matrix because \(n_p = n^* = 3\). The maximal theoretical number of paths according to (26) equals \(N_p^r = 3 \cdot 4 = 12\). The maximal number of searches of the multiple faults (21) for the MUFIA approach equals \(N^* = 2^3 - 1 = 7\). In this case, the MFI approach is expected to have worse effectiveness than the MUFIA.

Let us now apply the MFI approach. Firstly, a matrix of fault paths is generated by simple replacement of the non-zero entries of the BDS matrix by the appropriate fault labels. All non-zero BDS entries \(v_{j,i}\) are replaced by the labels \(f_i\) as shown below and all zero elements are removed. This matrix is referred as to matrix of fault paths.

| S/F | \(f_1\) | \(f_3\) | \(f_4\) |
|-----|--------|--------|--------|
| \(s_2\) | 1      | 1      | 1      |
| \(s_3\) | 1      | 1      | 1      |
| \(s_4\) | 0      | 1      | 0      |
| \(s_6\) | 1      | 1      | 1      |

\[\Rightarrow\]

| \(f_1\) | \(f_3\) | \(f_4\) |
|--------|--------|--------|
| \(f_1\) | \(f_3\) | \(f_4\) |
| \(f_3\) |        |        |

The process of generation of multiple faults from matrix of fault paths rely on searching for all possible combinations of all fault labels originating from different rows of this matrix. Appropriate rules are given in [13]. These rules define the principles of the combination and reduction of the rows of the matrix of paths. In fact, the rules allow for transformation
of the matrix of paths into the one row matrix of multiple fault labels. In order to exemplify this process, let us consider the first two rows of the right hand matrix shown above. Incidentally, both rows are identical. Possible combinations without repetitions of the fault labels from these both rows are: \( \{f_1, f_3\} \), \( \{f_1, f_4\} \), \( \{f_3, f_4\} \), \( \{f_4, f_4\} \). Thus, first two rows of the matrix of fault paths may be replaced by one row containing: \( \{f_1\} \), \( \{f_1, f_3\} \), \( \{f_1, f_4\} \), \( \{f_3\} \), \( \{f_3, f_4\} \), \( \{f_4\} \). Analogously, third and fourth row of this matrix may be replaced by the one row containing: \( \{f_1, f_3\} \), \( \{f_3\} \), \( \{f_3, f_4\} \). Result of combinations of rows (1, 2) and (3, 4) of the matrix of fault paths are depicted below.

\[
\begin{array}{cccc}
\{f_1\} & \{f_1, f_3\} & \{f_1, f_4\} & \{f_3\} & \{f_3, f_4\} & \{f_4\} \\
\{f_1, f_3\} & \{f_3\} & \{f_3, f_4\} \\
\end{array}
\]

The row reduction process is repeated until the matrix will be reduced to the single row. This row contains sequenced labels of all single and multiple faults.

\[
\begin{array}{cccc}
\{f_3\} & \{f_1, f_3\} & \{f_3, f_4\} & \{f_1, f_3, f_4\} \\
\end{array}
\]

6. Implementation

All three multiple fault isolation approaches were considered and implemented in a 16-bit microcontroller based platform (MSP430). The software was coded by means of a low level symbolic address programming language. The best programming practices were used to guarantee the highest performance of the code. Wherever it was possible, the same common procedures were applied for all these implementations. The execution time of each individual implementation was captured by means of the machine cycle counter. In the MSP430 family of micro-controllers, each machine cycle lasts reverse proportionally to the machine clock frequency. The counting of machine cycles allows direct comparison of the computational effectiveness of the investigated algorithms despite the frequency of the oscillator foreseen for the final application.

The diverse binary diagnostic matrices were examined in the experimental investigations. The outputs of all implementations were standardised. A standardised output record contain: total number of isolated multiple faults, index of isolated faults, distribution of fault multiplicities and the number of machine cycles used. In fact, the number of the machine cycles expresses indirectly the effectiveness of the implementation. In this study, the relative execution time indices \( t_r \) are defined as the ratio of the number of machine cycles used by the currently tested implementation to the number of either minimum or maximum number of machine cycles used by the reference implementation. The reference number of counts is updated with each change of \( BDM \) matrix. The analysis of the values of the efficiency indices allows the decision making regarding reasonability of the final choice of the multiple fault approach. These indices are particularly indicative for embedded applications in the smart field devices. As a rule, these devices have strongly limited computational power. Obviously, the results of the experimental tests are \( BDM \) matrix case sensitive. Chosen results of experimental tests are presented in Figs. 1–4.

Fig. 1 depicts the relative execution time \( t_r \) versus fault multiplicity \( k \) for \( MFI \) approach implementation. In this case, a diagonal \( BDM_{[k \times k]} \) matrix was tested. As can be seen, the relative execution time rise quadratic with the number of the fault multiplicity. Hopefully, this is not in contradiction to quadratic relationship laid down by the formula (26).

Fig. 2. refers also to the implementation of the \( MFI \) approach. It depicts the relative execution time versus the number of rows and columns of a diagonal \( BDM \) matrix. As can
be seen, the relative execution time is proportional to the number of rows as well as to the number of columns. Hence, the expected execution time is proportional to the product $n \cdot m$. This observation further strengthens the credibility of the formula (26). The execution time is increasingly dependent on the number of rows than the number of columns. Fig. 3 depicts the relative execution time $t_r$ versus maximal declared fault multiplicity $k$ for all studied multiple fault isolation approaches. In this case, a diagonal, constant size [10;10] BDM matrix was tested. As can be seen, the relative execution time values for diagonal matrix are almost equal for the classic and the MUFIA approaches. It comes from the adopted assumption that the condition (12) regarding disjunctive sets of diagnostic signals is not considered in this test. In this case the MFI approach is much faster beginning from fault multiplicities higher than 1. This is even more optimistic result than expected in conclusion 5.

An example of application to the diagnostic case of the steam-water line of the power station [14] based on the [9:6] BDM matrix is shown in Fig. 4. The condition (12) fails for this matrix. The only two multiple faults are present in the diagnosed system (one double and one triple). Therefore, the relative execution time of the MUFIA is flat for all $k \geq 3$. The relative execution time of the MFI is also flat for all $k \geq 1$, because the number of rows and columns of the BDM matrix is constant. Here, the MFI approach is slower for the low fault multiplicities and at least ca. 20% faster starting from the fault multiplicities higher than 2. This observation further strengthens conclusion 5.
7. Summary

A theoretical analysis dealing with the applicability features of the three chosen multiple fault isolation approaches based on a binary diagnostic matrix is presented in this paper. The classic, MUFIA and MFI approaches are discussed in order to elaborate practical hints and recommendations. All discussed approaches were implemented and examined in order to verify the correctness of the theoretical analysis. The following general practical recommendations can be withdrawn from presented above discussion:

(i) the classic approach should be used only in exceptional cases, where the number of all single faults in the system is very low (e.g. \( n \leq 3 \)) and the expected implementation burden would be minimal;

(ii) the MUFIA approach might be used, where the number of columns in each squeezed BDS matrix for each dynamically decomposed subsystem is very low (e.g. \( n^* \leq 3 \));

(iii) otherwise, the application of the MFI approach is strongly recommended.

Potentially, it is beneficial to use above given sets of recommendations for the development of the smart multiple fault isolation systems making dynamic decisions concerning the choice of the currently called multiple fault isolation procedure.

Acknowledgments

This work was supported by the Polish National Science Centre under the grant No DEC-2011/01/B/ST7/06183 entitled: “Methods of solution of fundamental problems of fault diagnostics in complex dynamic systems”.

References

[1] Isermann R and Ballé P 1997 Trends in the Application of Model Based Fault Detection and Diagnosis of Technical Processes (Control Engineering Practice) 5(5) pp 709–719

[2] Trave-Massuyès L 2014 Bridges between diagnosis theories from control and AI perspectives. In: Korbicz J and Kowal M (eds.) Intell. Syst. in Technical and Medical Diagnostics (Heidelberg: Springer) 230 pp 3–28

[3] Venkatasubramanian V and Rengaswamy R and Yin K and Kavuri S N 2003 A review of process fault detection and diagnosis: Part I: Quantitative model-based methods (Computers & Chemical Engineering) 27 pp 293–311

[4] Nyberg M and Krysander M 2003 Combining AI, FDI, and statistical hypothesis-testing in a framework for diagnosis. Proc. of 5th IFAC Symp. SafeProcess, (Washington DC) pp 891–896

[5] de Kleer J and Williams B 1987 Diagnosing multiple faults (Artificial Intelligence) 32(1) pp 97–130

[6] Isermann R 2006 Fault Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance (Berlin, Heidelberg: Springer) p 475

[7] Patton R and Frank P and Clark R. (Eds) 2000 Issues of Fault Diagnosis for Dynamic Systems (New York: Springer-Verlag) p 599

[8] Blanke M and Kinnaert M and Lunze J and Staroswiecki M 2006 Diagnosis and Fault-Tolerant Control (New York: Springer-Verlag) p 684

[9] Korbicz J and Kościelny J M and Kowalcuk Z and Cholewa W. (eds.) (2004). Fault Diagnosis. Models, Artificial Intelligence, Applications (Berlin: Springer) p 920

[10] Reiter R A 1987 Theory of Diagnosis from First Principles (Artificial Intelligence) 32(1) pp 57–95

[11] Nyberg M 2006 A Fault Isolation Algorithm for the Case of Multiple Faults and Multiple Fault Types (Proc. of IFAC Safeprocess, Beijing) pp 679–689

[12] Gertler J 1998 Fault Detection and Diagnosis in Engineering Systems (New York: Marcel Dekker Inc) p 484

[13] Bartyś M 2014 Multiple Fault Isolation Algorithm Based on Binary Diagnostic Matrix in Korbicz J and Kowal M. (eds.) Intell. Syst. in Technical and Medical Diagnosis (Heidelberg: Springer) 230 pp 441–452

[14] Kościelny J M and Bartyś M and Syfert M 2012 Methods of Multiple Fault Isolation in Large Scale Systems IEEE Transactions On Control Systems Technology, 20(5) pp 1302–1310

[15] Kościelny J M 1995 Fault Isolation in Industrial Processes by Dynamic Table of States Method, Automatica 31(5) pp 747–753