Graph reduction techniques for exergy-based FDI on the Tennessee Eastman process

R Styger


orcid.org/0000-0003-1796-6672

Dissertation accepted for the degree *Master of Engineering* in Electrical and Electronic Engineering at the North-West University

Supervisor:	Prof KR Uren
Co-supervisor:	Prof G van Schoor

Graduation:	May 2022
Student number:	27013588

Abstract

As industrial processes, which form the backbone of the industrialized world, continue to become larger and more complex, control systems require fault diagnostic schemes that can maintain plant safety and product quality, even during fault conditions. As a result, graph-based exergy fault diagnostic schemes have become increasingly popular tools, especially in the petrochemical industry. However, when these graph-based schemes are applied to complex processes, the implementation becomes more complex and requires the deployment of more sensors as well as additional computational resources for the control system. Therefore, this study focuses on evaluating the concept of graph reduction by proposing several graph reduction techniques and assessing their efficacy at reducing complexity while preserving the performance of the fault diagnostic schemes. To determine the effect that the reduction techniques would have on the fault detection and isolation (FDI) methods, it is first necessary to determine the performance of these FDI methods prior to any graph reduction and use it as control data. The distance parameter, eigendecomposition, and residual-based FDI methods are used in this study. The attributed graph data used in this study is generated from the Tennessee Eastman process (TEP). Five graph reduction techniques are proposed based on three theoretical concepts, which rely on understanding the process used by FDI methods. These three concepts are concerned with finding redundant attributes and removing them from the graph. These reduction techniques are evaluated with an experimental process whereby the extent to which the technique reduces graph attributes (reduction interval) is increased, and the performance of the FDI methods using this reduced version of the graph data is recorded. Graph reduction is a viable concept when at least one reduction technique can reduce graph attributes while maintaining the level of FDI performance achieved prior to reducing any attributes. To validate this study, it is shown that graph reduction is a general solution by applying these five reduction techniques to the attributed graph data of the gas-to-liquids process (GTLP) and evaluating their effect on FDI performance. The three FDI methods are applied to the graph data of the GTLP to generate a set of control data. The reduction techniques are assessed with the same experimental process, which reduces more attributes from the GTLP graph data and measures FDI performance after each reduction increment. Since at least one reduction technique could reduce the attributed graph data of both the TEP and GTLP, while maintaining a similar level of FDI performance for at least one FDI method, graph reduction is considered a general solution, and the reduction techniques have been validated. This study clearly shows that it is possible to reduce the attributed graph data of a process and maintain or even improve upon, in some instances, the level of FDI performance achieved before reducing attributes. It will, therefore, contribute to mitigating the adverse effects resulting from applying graph-based FDI methods to large and complex

processes.

Keywords: Graph reduction, attributed graph, exergy-based FDI, Tennessee Eastman process.

Acknowledgements

I would sincerely like to thank the following people for their support during the course of this study:

- First and foremost, I would like to thank my Heavenly Father, without whom none of this would have been possible.

  "And whatever you do, whether in word or in deed, do everything in the name of the Lord Jesus, giving thanks to God the Father through Him." Colossians 3:17

- My study leaders, Prof. George van Schoor and Prof. Kenny Uren, for their valuable inputs, guidance, moral support, and for truly making this an enjoyable journey.
- My parents, Eugene and Michellé, as well as my sister, Francè, for their love, support, and contributions throughout the years.
- My amazing friends and family for all the joy they provided and for taking an interest in this study.
- The members of the McTronX research group for their inputs on this study. In particular, Johandri Vosloo and Sarita Greyling, for the guidance they provided with their respective process models.
- Any other individual who provided support along the way.
- To the National Research Foundation (NRF) of South Africa for providing me with a bursary to complete this study.

This material is based on research/work supported wholly/in part by the National Research Foundation (NRF) of South Africa (Grant Number 129084). The research findings are that of the author and not that of the NRF.

Contents

List of Figures

List of Tables

Acronyms

AEM	Active Event Management.
ATR	Autothermal Reformer.
CANAL	Categorization of Attributes with Numerical Values based on Attribute Values and Link Structures of Nodes.
CPV	Cumulative Percentage Variance.
FDI	Fault Detection and Isolation.
FN	False Negative.
FP	False Positive.
FT	Fischer-Tropsch.
GTLP	Gas-to-liquids process.
HEOM	Heterogeneous Euclidean-Overlap Metric.
MDL	Minimum Description Length.
NOC	Normal Operating Conditions.
NSM	Node Signature Matrix.
PCA	Principal Component Analysis.
PCI	Petrochemical Industry.
SGVR	Summarizing Graph based on Virtual and Real links.
SNAP	Summarization by Grouping Nodes on Attributes and Pairwise Relationships.
TEP	Tennessee Eastman process.
TN	True Negative.
TP	True Positive.

Chapter 1

Introduction

1.1 Background

A *fault* can be defined as the deviation from the acceptable range of a calculated parameter or an observed variable of an industrial process [1]. When a fault occurs in such an industrial process, an activity known as Active Event Management (AEM) must take place. AEM, which forms part of the control system, involves taking the necessary steps to ensure that the process returns to the normal state of operation and that the entire operation is safe [2].

A robust fault diagnostic system is required for the control system to implement AEM and remove the effects of a fault condition from the process or reduce these effects to an acceptable level. This critical system should first detect that a fault has occurred (Detection) and then determine which fault condition has occurred (Isolation). This type of diagnostic system employs a method known as a fault detection and isolation (FDI) scheme. Once the FDI scheme has diagnosed the fault condition, the control strategy can be adjusted accordingly to ensure that the process still satisfies the design specifications, or an alarm could be activated to alert the operators that a fault is present.

There exists a great variety of FDI methods that can be used to implement a fault diagnostic system. The type of method used will depend on the type of process as well as the availability of process knowledge, and historic process data [2]. Figure 1.1 gives a breakdown of the main categories of fault diagnostic methods.

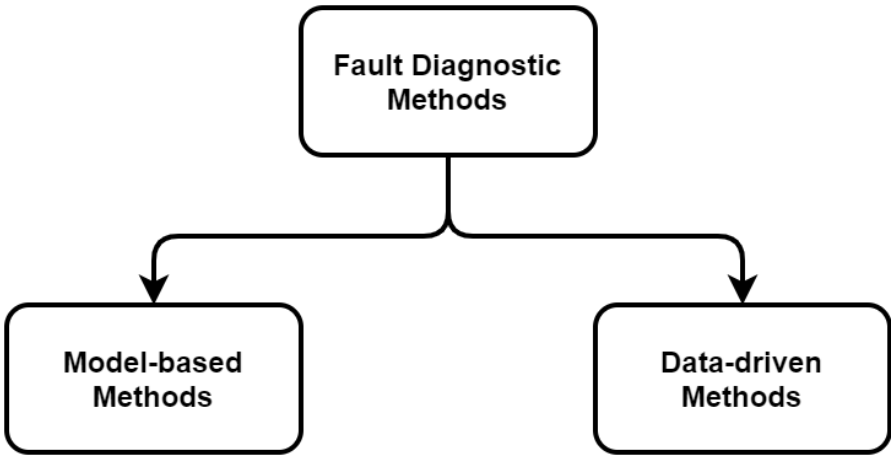


Figure 1.1: Categories of fault diagnostic methods [3].

Model-based methods rely on a fundamental understanding of how a process works to diagnose fault conditions [3]. On the other hand, data-driven methods use a large amount of historical process data and apply feature extraction during which the data are transformed and presented to the diagnostic system as theoretical knowledge [4].

The fact that so many methods exist is a clear indication that no single method satisfies all the requirements of a fault diagnostic system. This has led to the development of hybrid methods which employ a combination of methods to complement one another and overcome the limitations of individual methods [4]. These hybrid methods are generally more attractive solutions since they are better equipped to satisfy the requirements of the diagnostic system.

The hybrid approach, as outlined by Marais et al. in [5], has shown significant merit, especially in petrochemical settings. This approach abstracts physical measurements such as temperature and pressure to energy variables such as exergy. The FDI scheme uses the system exergy as a metric for diagnosing faults in the system.

In [6], Van Schoor et al. show how the approach proposed in [5] can be used to create an energy representation of a system. Van Schoor et al. further illustrate how this energy representation in the form of an attributed graph can be used in an exergy-based FDI scheme to monitor the condition of a system. An attributed graph is a graph that has information assigned to its nodes and/or links [7]. In this case, information pertinent to the FDI scheme is assigned to the nodes and/or links.

As an example, Figure 1.2 [6] illustrates the composition of the attributed graph of a coal-fired power station. Each node in the graph represents a component of the power station, and each link represents the connection between two components of the station. The attributes assigned to nodes are the exergy flow rates across components, and the attributes assigned

to links are the energy flow rates between components. By mathematically analysing these attributes, the exergy-based FDI scheme proposed in [6] can then detect and isolate process faults.

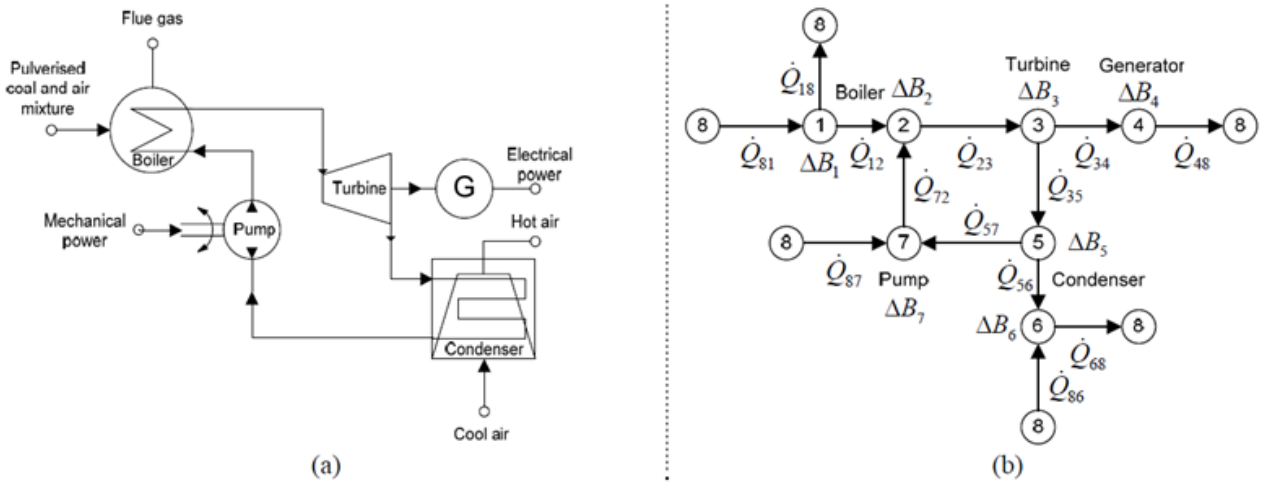


Figure 1.2: (a) Representation of the power station. (b) Corresponding attributed graph [6].

From Figure 1.2 it can be deduced that the larger a process, the larger its corresponding attributed graph will be, as these processes have more components and a more significant number of connections between the components. The more components a process has, the more attributes will be present in the attributed graph of the process, ultimately increasing the graph's complexity. It then becomes more complicated to use complex attributed graphs in the exergy-based FDI scheme since more process sensor data are required to calculate the attributes, the mathematical matrix operations become more complex and take longer to execute, resulting in faults not being detected promptly.

The FDI scheme is ultimately implemented to ensure that the plant operation remains safe, the product quality remains satisfactory, and financial losses are minimised. However, it is difficult for FDI schemes with complex attributed graphs to achieve these objectives since they take longer to operate and require more computational resources. It is, therefore, warranted to investigate techniques that can reduce the size of these attributed graphs used in the exergy-based scheme, to reduce the ultimate complexity of implementing the FDI scheme.

In practice, control systems use FDI schemes in real-time. By simplifying the attributed graph of an industrial process, it is possible for the FDI scheme to diagnose the process condition faster and to use fewer resources since fewer process variables need to be monitored and less data processing is needed.

1.2 Problem statement

Large industrial processes result in large and complex attributed graphs. These large graphs adversely affect FDI schemes' computational abilities and speed and often require extra processing resources and sensors. This study aims to formulate techniques that can reduce the graph used in exergy-based fault detection methods and then evaluate the performance of these methods, which use the reduced graph data by using the Tennessee Eastman process (TEP) as a benchmark process. The goal of the reduction techniques is to reduce the complexity of the graph data by removing nodes and/or attributes. This will ensure that only attributes which contribute to the FDI schemes are retained. The study will then determine if the FDI schemes that use reduced graph data can retain a similar level of performance achieved prior to any graph reduction. In doing so, it will ultimately be possible to determine if the concept of graph reduction is viable.

When evaluating the diagnostic performance of each of the schemes to determine their effectiveness, it is necessary to be holistic and evaluate the schemes based on their ability to detect and isolate process faults. The focus of this study is to develop graph reduction techniques for the TEP model, which are also effective in general and can be applied to the graph data of other systems. This study focuses only on graph-based FDI schemes, and the scope of this study does not include determining which FDI scheme is the optimal option for the benchmark system. All process models and the standard versions of their attributed graphs have been developed as part of separate studies.

1.3 Research objectives

1.3.1 Formulate graph reduction techniques

In order to propose viable graph reduction techniques, it is necessary to investigate how the attributed graph of a process is constructed and how this graph is used by the relevant FDI methods to diagnose fault conditions. It is further necessary to review literature pertaining to existing graph reduction techniques. Graph reduction techniques can then be developed based on the findings of this investigation. Finally, verification of each technique is required to ensure that they are correctly implemented.

1.3.2 Generate the FDI control data set

In order to evaluate the efficacy of the proposed graph reduction techniques, the performance of the FDI methods using the standard, unreduced attributed graph data of the benchmark process is first determined. Three different FDI methods are used for the evaluation. The performance of each FDI method then acts as a set of control data. The performance is considered with respect to the method's ability to detect and isolate process faults. The benchmark system in this study is the Tennessee Eastman process (TEP).

1.3.3 Develop an experimental design

An experimental process is designed to determine if it is conceptually possible to reduce the attributed graph data used by exergy-based FDI methods while still maintaining a similar level of FDI performance as achieved before reducing the graph data. In addition, this experimental process identifies how each technique should be applied to result in the maximum reduction of graph attributes while minimally deteriorating FDI performance. It can also be determined which reduction techniques complement each FDI method.

1.3.4 Evaluate the graph reduction techniques on an additional process

To validate this study, it is necessary to show that graph reduction is a general solution that is not only valid when applied to the attributed graph data of a specific process. To this end, it is required to show that the proposed graph reduction techniques can reduce graph attributes while maintaining a similar FDI performance level when applied to a different process than the TEP.

1.4 Methodology

1.4.1 Formulate graph reduction techniques

A literature study is conducted on how process data are encoded in an attributed graph, how exergy-based FDI methods use these attributed graphs, and existing graph reduction techniques. Based on how FDI methods use graph data, several graph reduction techniques are

developed to increase the chances of formulating a viable solution. Each technique's corresponding algorithm is then implemented in MATLAB® where these techniques are applied to reduce the attributed graph data relevant to this study. Finally, each technique is verified by implementing its algorithm in MATLAB® and Excel® to ensure that both implementations produce the same result.

1.4.2 Generate the FDI control data set

The distance parameter, eigendecomposition, and residual-based FDI methods are the three FDI methods used to evaluate the efficacy of the graph reduction techniques. These graph-based methods utilise the data encoded in the graph for the purpose of fault diagnosis. Each FDI method is implemented in MATLAB®. Data from a validated Simulink® model are used to generate the original attributed graph data of the TEP. The attributed graph data includes all the fault conditions as well as the normal operating condition (NOC).

Once the attributed graph data has been processed to a format compatible with the FDI methods in MATLAB®, each FDI method is applied to the graph data. The MATLAB® code then determines the FDI method's overall detection rate, overall isolation rate, and the specific isolation rates of critical fault conditions. These metrics form the control data of each FDI method. For this study, it is assumed that only one fault is introduced to the process at a time since most existing graph-based FDI schemes have not yet developed the necessary functionality to deal with multiple faults [8]. The presence of unknown faults is also not considered in this study since further research is required into how these graph-based FDI schemes, which use the standard attributed graph data, deal with unknown faults.

1.4.3 Develop an experimental design

The experimental process involves conducting an experiment on each reduction technique. First, reduction intervals are identified for each reduction technique. These reduction intervals determine to which extent the technique removes attributes from the graph data. Then, each technique is applied to the graph data iteratively, and the reduction interval is increased with every iteration.

With every reduction iteration, all three FDI methods are applied to the reduced graph data resulting from that reduction iteration. The overall detection rate, overall isolation rate, and the specific isolation rates are determined for each FDI method. By comparing FDI performance resulting from each reduction iteration with FDI performance from the control

data, it is possible to determine which reduction intervals are more effective at reducing attributes while maintaining a similar level of FDI performance as achieved before applying graph reduction. It is also possible to determine which reduction technique and FDI method combination has superior performance. A logic flow diagram of the experimental process followed for each reduction technique can be seen in Figure 1.3.

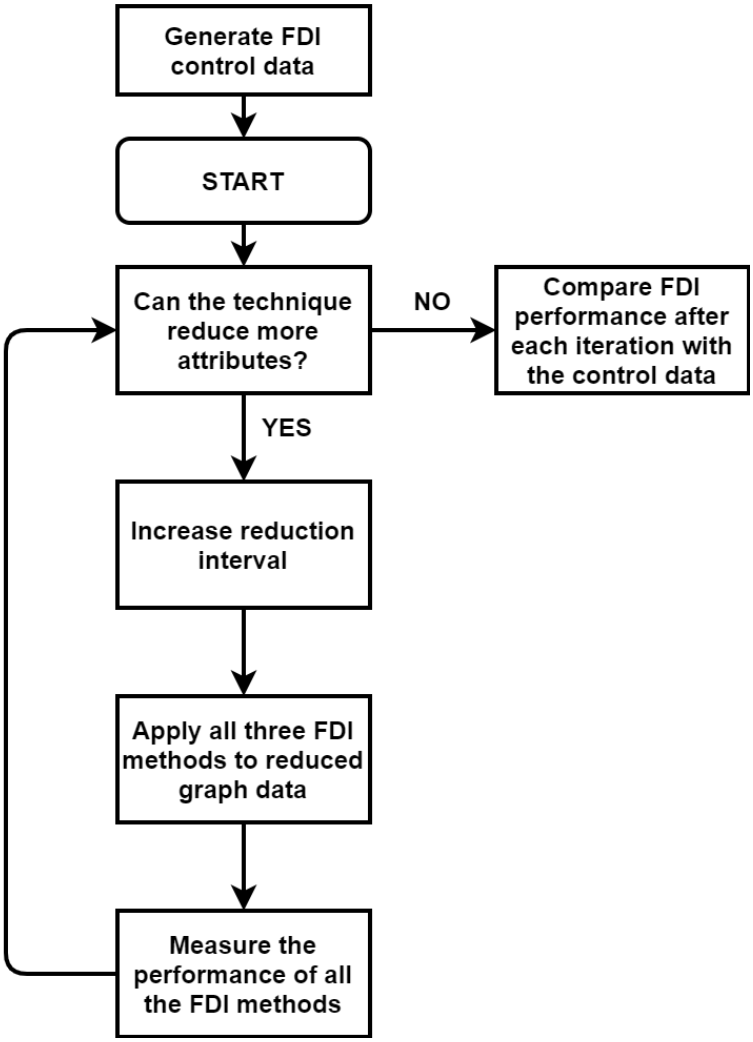


Figure 1.3: Logic flow diagram of the experimental process applied to the graph reduction techniques in this study.

For this study, no explicit specifications are provided to define when FDI performance using the reduced graph data maintains a similar level of performance as achieved before applying graph reduction. It is, therefore, not stated at what point the deterioration in FDI performance becomes unacceptable. This is simply because different processes will have different minimum requirements for the performance of the FDI method used by the control system.

1.4.4 Evaluate the graph reduction techniques on an additional process

The other system used to validate the graph reduction techniques is the gas-to-liquids process (GTLP), and the process data is generated with a static model in HYSYS®, which generates steady-state data. There are many contrasts between the TEP and GTLP, such as the fact that the TEP model is a dynamic model which generates time-based data, while the GTLP generates an instantaneous snapshot of the process data while the process is in steady-state mode. The two processes also have different levels of complexity and different types of fault conditions. These contrasts assist in proving the generality of a solution. In the context of this study, a solution is considered general when it can successfully be applied to more than one process. This shows that the solution is not limited to a specific process.

To evaluate the effect of the graph reduction techniques on the attributed graph data of the GTLP, a similar procedure to the one used on the TEP graph data is followed. All three FDI methods are first applied to the unreduced attributed graph data of the GTLP, and the performance of each FDI method is used as a set of control data. Then, the experimental process used to evaluate the reduction techniques' efficacy when applied to the TEP graph data is repeated on the GTLP graph data.

For each reduction technique, the performance of each of the three FDI methods is determined after each reduction iteration has been applied. By comparing the FDI performance at all reduction iterations with the control data, it is possible to determine if the attributed graph data of the GTLP can be reduced while maintaining the same level of FDI performance achieved before any graph reduction. If graph reduction can successfully be applied to the GTLP graph data, it can be considered a general solution, which would validate the reduction techniques.

1.5 Dissertation outline

Chapter 2 provides a discussion of the literature that forms the foundation of this study. The concept of energy visualisation is introduced, and an explanation of how energy is used to characterise a process is provided. The type of graph resulting from energy visualisation is described, and an overview of how FDI methods use this type of graph to analyse the process condition is then given. Relevant graph reduction solutions are listed, and their effectiveness is evaluated. The Tennessee Eastman process is introduced and briefly discussed. This chapter

concludes with a critical review.

Chapter 3 is reserved for the specific literature and expands on the essential elements discussed in Chapter 2. This chapter gives an overview of the TEP Simulink® model and the modelling process followed to develop it. All the fault conditions of the TEP are listed in this chapter. The methodology used to develop the attributed graph of a process is explained. The chapter concludes by introducing the attributed graph of the TEP as used in this study.

The primary focus of Chapter 4 is to determine the FDI performance when the FDI methods are applied to the unreduced attributed graph data of the TEP to use as control data. First, the detailed methodology of each of the three FDI methods is provided. Each FDI method is then applied to the TEP graph data, after which the performance of that FDI method is recorded to be used as control data when evaluating graph reduction techniques.

Chapter 5 introduces the five graph reduction techniques and gives a brief description of the premise each technique is based on. Then, the experimental process used to evaluate the reduction techniques is designed. This is followed by a discussion as to how the results should be analysed.

In Chapter 6, the methodology of each reduction technique and the experimental process used to evaluate that technique is outlined. After each methodology, the results from the experimental process are provided. Observations and interpretations about the results of the experimental process are then discussed. The effect of combining reduction techniques is also evaluated. Finally, the verification of the reduction techniques concludes this chapter.

Chapter 7 shows that graph reduction is a general solution by evaluating the effect of the graph reduction techniques on a gas-to-liquids process (GTLP) modelled in HYSYS®. The FDI control data is first generated by applying the three FDI methods to the GTLP. Then, the experimental process used in the previous chapter is repeated on the GTLP to determine the efficacy of the reduction techniques on the GTLP. Combined reduction techniques are also evaluated. The validation of this study concludes this chapter.

The final chapter in this dissertation concludes the study by summarizing the results and highlighting the most critical findings. Specific suggestions regarding the future work pertaining to the findings of this study are also included, and potential shortcomings in the research relating to the topic are identified.

Chapter 2

Literature study

2.1 Introduction

This chapter aims to provide an overview of the literature that is relevant to this study. First, the concept of energy visualisation and its role in FDI schemes is discussed. Next, an introduction to the graph theory applicable to graph-based FDI schemes is provided, followed by a brief explanation of how graphs are used in these FDI schemes. This is followed by a discussion of several identified graph reduction techniques. These techniques are also evaluated as related to the objectives of the study. The final section of this chapter introduces the benchmark model used in this study and briefly describes how this system works.

Figure 2.1 gives a clear description of the four main research topics and the subsections of each topic. All the subsections are accompanied by the relevant citations used. The "Graph comparison" and "Graph summarization" subsections branch out into subsections of their own. The coloured lines indicate the citations the two research topics have in common. This illustrates how all the research topics relate to each other.

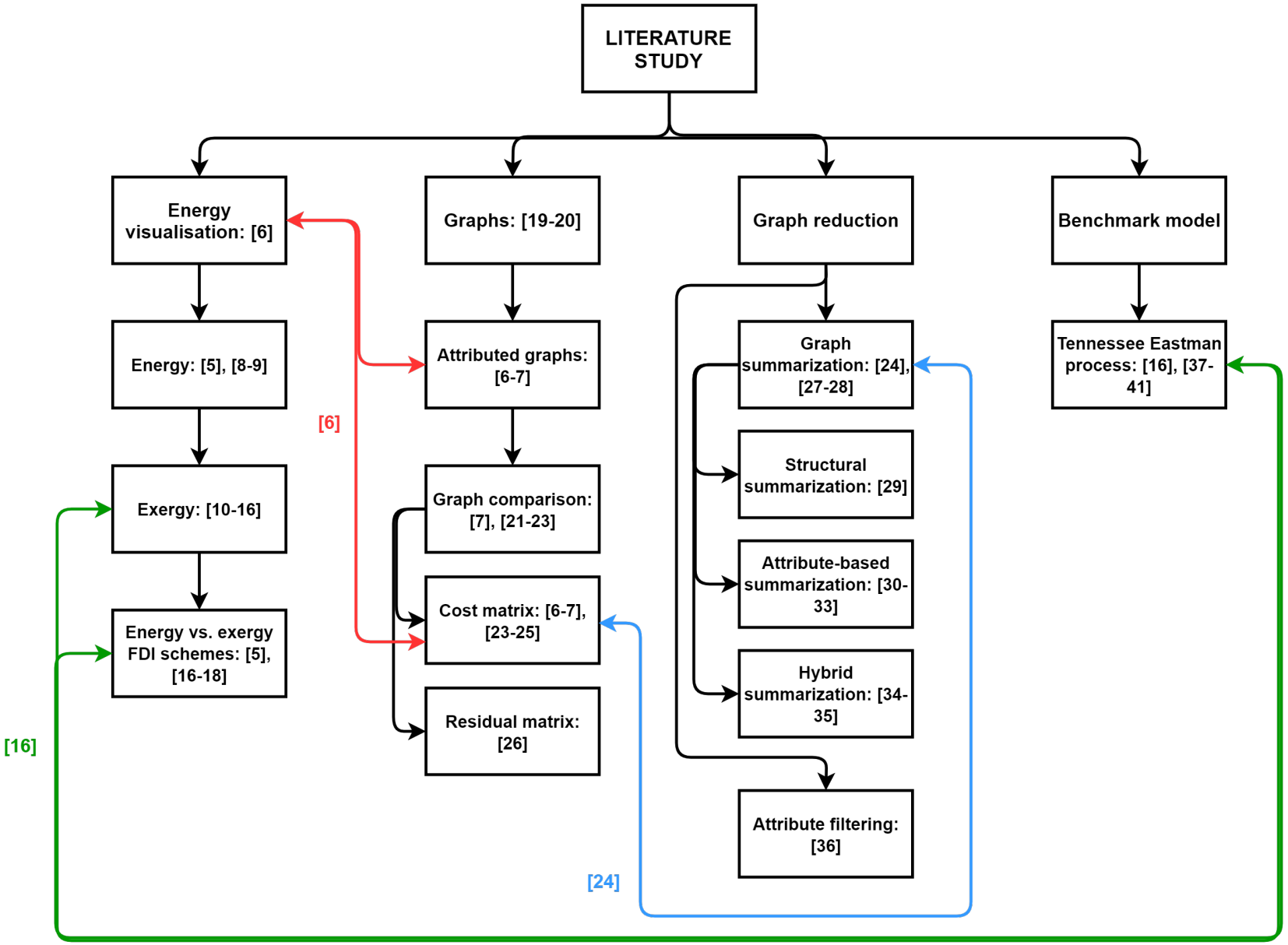


Figure 2.1: Overview of the literature study with all the references used in this chapter.

2.2 Energy visualisation

Energy visualisation can be described as the process of using a system's enthalpy and entropy attributes to characterise and assess the overall health of the system [6]. This section examines the concepts of energy and exergy and their role in the energy visualisation process used by certain hybrid fault detection schemes.

2.2.1 Energy

Energy is a unifying concept across physical domains (electrical, mechanical, chemical, and thermal) and can, therefore, be used to reduce the dimensionality of the input space in a hybrid fault detection scheme [5]. Furthermore, energy is a scalar quantity, and while it is impossible to observe it directly, indirect measurements make it possible for energy to be recorded and evaluated [9].

According to Dincer & Cengel [9], all the forms of energy in thermodynamic analysis can be classified as either macroscopic or microscopic. A macroscopic form of energy is a form of energy the system possesses relative to some external frame of reference, such as kinetic and potential energy. *Kinetic energy* is defined as the energy a system possesses because of its motion relative to a frame of reference [9]. *Potential energy* is the energy a system possesses because of its elevation in a gravitational field [9].

A microscopic form of energy is a form of energy that relates to molecular structure and the degree of molecular activity. Microscopic forms of energy are independent of external reference frames. The *Internal energy* of a system is the sum of all the microscopic forms of energy of that system [9].

According to the First Law of Thermodynamics, energy cannot be created or destroyed; it can only be converted from one form to another. The change in a system's energy is the difference in the system's energy values at the final state and initial state of the process [10]. This change in a system's energy, when electric, magnetic, and surface tension effects are absent, can be expressed as

$$\Delta E = E_2 - E_1, \tag{2.1}$$

which can also be expressed as

$$\Delta E = \Delta U + \Delta KE + \Delta PE, \tag{2.2}$$

with ΔU as the change in the system's internal energy, ΔKE as the change in the system's kinetic energy, and ΔPE as the change in the system's potential energy.

2.2.2 Exergy

Exergy can be defined as the ability to do useful work [11]. The total exergy of a system can be defined as

$$b_{tot} = b_k + b_p + b_{ch} + b_{ph}, \tag{2.3}$$

where b_k and b_p are the kinetic and potential exergy, respectively [12]. b_{ch} and b_{ph} are the chemical and physical exergy, respectively. In most cases which involve process plants, the kinetic and potential exergy can be ignored since the plant itself is unlikely to move during normal operation [13]. The total exergy expression can then be simplified to

$$b_{tot} = b_{ch} + b_{ph}. \tag{2.4}$$

Chemical exergy can be defined as the available energy to do work when a substance undergoes a reversible process from the restricted reference state of 25 °C and 1 atm to complete thermodynamic equilibrium (thermal, chemical, pressure) [14]. The chemical exergy of a substance can be expressed in its most generic form as

$$b_{ch} = \sum x_{(i)} b^0_{ch(i)}, \tag{2.5}$$

where $x_{(i)}$ is the mole fraction of substance i and $b^0_{ch(i)}$ is the standard chemical exergy of that substance [15].

The *Entropy* of a system is defined as the amount of molecular disorder in that system, and the *Enthalpy* of a system is defined as the sum of the system's internal energy and the product of its volume and pressure [16].

The *Physical exergy* of a system accounts for the mechanical exergy and thermal exergy of a system, which are associated with deviations in pressure and deviations in temperature, respectively [17]. The physical exergy of a material stream can be determined as

$$b_{ph} = (h - h_0) - T_0(s - s_0), \tag{2.6}$$

where h and s represent the enthalpy and entropy of the stream, h_0 and s_0 are the reference environment entropy and enthalpy values of the stream, and T_0 is the reference environment temperature [12]. The entropy and enthalpy of a stream can be expressed as indicated in (2.7) and (2.8).

$$(h - h_0) = \int_{T_0}^{T} C_p dT, \tag{2.7}$$

$$(s - s_0) = \int_{T_0}^{T} \frac{C_p}{T} dT - R\ln(\frac{P}{P_0}). \tag{2.8}$$

C_p is the heat capacity of the particular stream, R the universal gas constant, and P_0 the reference environment pressure. T_0 is defined as 25 °C and P_0 is defined as 101.325 kPa [15].

2.2.3 Comparison of energy-based and exergy-based methods

Energy-based fault diagnostic schemes rely on an energy representation, which closely resembles the actual process plant, to detect if a fault has occurred in the process. What makes this energy representation so valuable is the fact that energy is a universal concept and the inputs and outputs of a process can, therefore, be treated additively [5]. The energy-based scheme uses the entropy and enthalpy of a process to conduct condition monitoring.

By using an energy representation of a process plant, the number of monitored process variables can be reduced [5]. Energy-based schemes are not new and have been applied in several instances, such as the condition monitoring of a simple chemical reactor [18] and the fault diagnosis of an autothermal reformer [19].

According to Marais et al. [5] natural processes display a degree of irreversibility, which is followed by a loss of exergy. This destruction of exergy ultimately causes a process to become less useful. Therefore, the exergy-based schemes rest on the premise that the irreversibility of the process, which is linked to the destruction of exergy, indicates the condition of the process.

There are fewer examples of exergy-based schemes than energy-based schemes simply because more research has been devoted to energy-based schemes. However, some of the recent examples of exergy analysis being used as a form of condition monitoring include the exergy-based fault detection of the Tennessee Eastman process [12] and of a transcritical heat pump system [7].

There is a large degree of similarity between the exergy-based and energy-based schemes, and it is thus expected that their performance will be similar in several areas. Marais et al. [5] experimentally compared the exergy-based and energy-based methods against applicable metrics, and a summary of the results can be seen in Table 2.1. A checkmark indicates the superior scheme, while the absence of one indicates that the schemes perform equally well.

It is evident from the findings in Table 2.1 [5] that an exergy-based fault detection scheme will outperform an energy-based scheme, especially in petrochemical applications, since it has a superior ability to isolate the detected fault. The metrics listed in the table are considered by [5] to be the most important metrics to evaluate when comparing different schemes. Because chemical compositional variations also affect physical exergy, the exergy-based scheme is better than the energy-based scheme. The two schemes have similar modelling requirements as well as similar storage and computational requirements.

Table 2.1: Comparison of exergy-based and energy-based schemes

Chosen Metrics	Exergy-based schemes	Energy-based schemes
Isolability	✓	
Explanation Facility	✓	
Modelling Requirements	-	-
Storage & Computational Requirements	-	-

2.3 Graphs

A *graph* can simply be defined as a set of vertices and a set of edges, where each edge connects two vertices [20]. Any system where information can be modelled as objects and the relationship between these objects can be represented by a graph. This is achieved by representing objects as vertices and the relationship between these objects as the edges [20].

According to Balakrishnan and Ranganathan [21], a graph has the following mathematical definition: "A graph is an ordered triple $G(V(G), E(G), I_G)$, where $V(G)$ is a nonempty set, $E(G)$ is set disjoint from $V(G)$, and I_G is an 'incidence' relation that associates with each element of $E(G)$ an unordered pair of elements (same or distinct) of $V(G)$." The vertex elements set $V(G)$ represent all the vertices of the graph G, and the elements of the edge set $E(G)$ represent the edges of G. All the elements in I_G represent how the vertices and edges of graph G connect.

Figure 2.2 [21] is an illustration of such a graph. All the vertices and edges have been labelled in the figure. In Graph Theory literature, it is common to see the term vertex be used interchangeably with the term node and the term edge be used interchangeably with the term link. For the sake of consistency and to avoid confusion, this dissertation shall henceforth use the terms node and link.

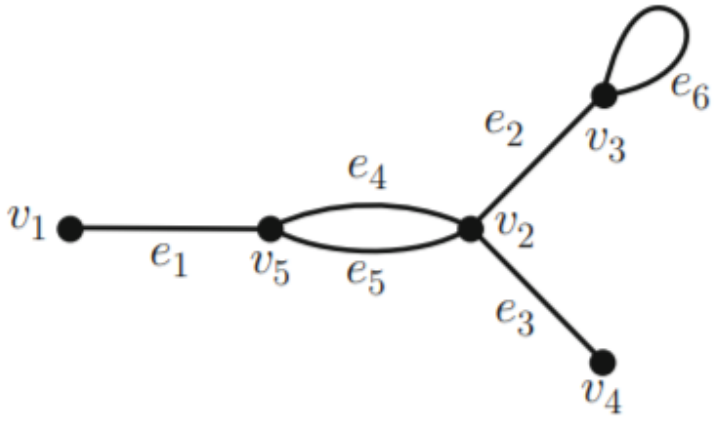


Figure 2.2: Example of a graph G [21].

For this study, graphs will be used for FDI. The types of graphs that will be used for FDI are numerical attributed graphs. In a paper by Van Schoor and Uren [6], a process is outlined which employs FDI by mathematically analysing attributed graphs. The process works by applying graph comparison operations to the attributed graph of the system in question. This section explores the graph comparison procedure and how it can be implemented in an FDI scheme.

2.3.1 Attributed graphs

Energy-visualisation, as used in exergy-based FDI schemes, requires the system to be modelled as an attributed graph representing the system's various energy and exergy flow rates. An attributed graph is useful since it can compile a node signature matrix (NSM) which is in turn used to perform graph comparison operations [7]. A *node signature matrix* is simply a matrix that represents the attributed information of a specific attributed graph [6]. An example of an attributed graph and its corresponding node signature matrix can be found in Figure 2.3 [6]. In the context of FDI methods, the phrases "attributed graph" and "node signature matrix" are often used interchangeably since they contain the same information in different formats. An attributed graph is converted into a node NSM, and an FDI scheme then uses the NSM to diagnose faults.

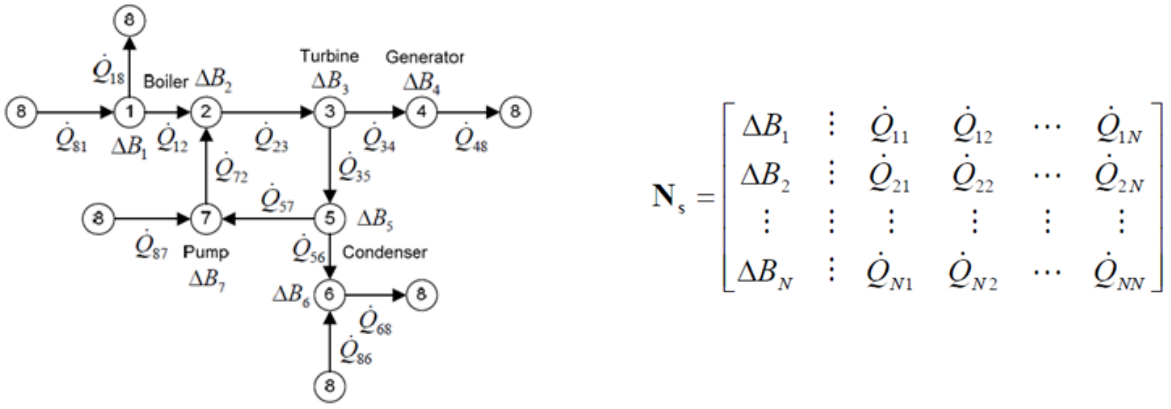


Figure 2.3: (a) An example of an attributed graph. (b) The corresponding node signature matrix [6].

2.3.2 Graph comparison for FDI

Graph comparison as used in this study, can be described as the process of mathematically comparing two or more graphs to determine how similar the attribute parameters of the different graphs are [22]. In the context of FDI, graph comparison is used to compare the NSM of the system while it is in a normal operating condition, with an operational NSM to determine if a fault condition is present or not. Whenever attributed graphs are compared in this study, the comparison refers to a mathematical graph comparison procedure.

De la Fuente [23] defines a normal operating condition (NOC) as: "A system operating under a condition where none of its defining characteristic properties or associated parameters deviates from the standard system condition." In addition, De la Fuente [23] also defines a *fault condition* as: "A system operating under a condition where an unpermitted deviation of at least one characteristic property or parameter of the system from the standard condition is present."

From the literature, there are two main strategies of employing graph comparison procedures in FDI schemes. The first strategy involves generating a cost matrix and mathematically analysing this matrix to diagnose faults as was done by [7] & [8]. The second strategy involves generating a residual matrix which can also be mathematically analysed to diagnose faults.

2.3.3 Graph comparison by generating a cost matrix

A *cost matrix* describes the matching cost (the cost of the mathematical comparison) between two node signature matrices and is denoted by C [24]. A cost matrix, therefore, represents

the degree of similarity between these two node signature matrices. Jouili & Tabbone [24] proposed a method that can be used to generate a cost matrix by comparing a node signature matrix with itself or with a node signature matrix that contains different elements.

In [8], a database is created by storing an NSM of the system in NOC and an NSM of the system for each fault condition. Next, graph comparison is applied by developing a cost matrix between the operational NSM and each one of the matrices in the database. All the cost matrices are then quantified to a distance parameter. The cost matrix with the smallest distance value thus represents the cost matrix developed between the operational NSM and the database NSM most similar to this operational NSM. Therefore, the smallest distance value identifies which condition in the database the system is currently operating in.

It has also been shown that FDI schemes can apply eigenvector and eigenvalue decomposition to cost matrices in order to diagnose faults [7], [8]. *Eigenvectors* and *eigenvalues* are mathematically defined by Lay et al. [25] as the following: "An *eigenvector* of an $n \times n$ matrix $\mathbf{A}$ is a nonzero vector $\mathbf{x}$ such that $\mathbf{Ax} = \lambda\mathbf{x}$ for some scalar λ. A scalar λ is called an *eigenvalue* of $\mathbf{A}$ if there is a nontrivial solution $\mathbf{x}$ of $\mathbf{Ax} = \lambda\mathbf{x}$; such an $\mathbf{x}$ is called an *eigenvector corresponding to* λ."

Eigenvectors of a linear transformation are vectors that remain on their original spanning line through the origin of the n-dimensional space after the transformation has taken place [25]. Thus, eigenvectors make it possible for a linear transformation in the n-dimensional space to be characterised by mathematically describing the transformation.

An eigenvalue is associated with each eigenvector, and it represents the change in magnitude, or in some instances, the change in the direction of that eigenvector. While the magnitude and direction of an eigenvector can change when a linear transformation takes place, it will always pass through the origin of the n-dimensional space and remain on its span.

Since the cost matrix is a linear transformation, calculating its eigenvectors and eigenvalues will assist in characterising the information in the cost matrix and, thus, aid with the detection and diagnosis of faults [6].

In [7], cost matrices are also developed by comparing the NSM of the system in NOC with the NSM of each fault condition. These cost matrices are known as reference cost matrices. Each reference cost matrix undergoes an eigenvalue and eigenvector decomposition and produces eigenvector and eigenvalue patterns unique to a specific fault. An operational cost matrix can then be developed by comparing the operational NSM with the NSM of the system in NOC. The fault can then be diagnosed by computing the eigenvector and eigenvalue patterns of this operational cost matrix and comparing it with all the patterns produced by the reference cost

matrices.

The study conducted in [8] also uses eigenvalue decomposition to diagnose faults. A database is once again constructed with the NOC graph and the graph of each process fault. The first step is to develop a cost matrix by comparing the operational graph with itself, known as the operational cost matrix. Then each graph in the database should be compared with the operational graph to produce several reference cost matrices. The operational cost matrix and all the reference cost matrices should undergo eigenvalue decomposition to produce an eigenvalue pattern for each cost matrix. By comparing the eigenvalue pattern of the operational cost matrix with the eigenvalue pattern of each one of the reference cost matrices, the process condition can be inferred from the database.

2.3.4 Graph comparison by generating a residual matrix

As an alternative to generating cost matrices and using them to diagnose faults, Neser [26] proposes using graph comparison operations to implement a residual-based FDI method. To generate a residual matrix, the reference graph of the system, which is just the graph of the system in NOC, is compared with the actual operational graph of the system. By extracting certain features from the residual matrix, a residual signature is generated. A database of residual fault signatures is then constructed by comparing the NSM of each fault condition with the NSM of the system in NOC. Each of these residual fault signatures is unique to a specific fault condition and can be used to diagnose fault conditions in the system.

2.4 Graph reduction

Representing a set of data or a system as a graph has many applications in many different domains [27]. A graph can either be augmented by increasing the number of nodes or edges, which increases the graph's complexity, or it can be reduced by removing nodes or edges, which reduces the graph's complexity. This section investigates methods that can reduce the size and, thus, the complexity of an attributed graph.

2.4.1 Graph reduction through graph summarization

One possible way to reduce a graph is to summarize it. The process of graph summarization removes unnecessary detail while retaining the general properties of the original graph [27].

The summarization process is performed by creating supernodes by grouping together similar nodes from the original graph [28]. The graph summarization process can be seen in Figure 2.4 [27]. The graph on the left-hand side is the original, and the graph on the right-hand side is the summarized version. Similar nodes are represented by the same colour and are grouped into a supernode.

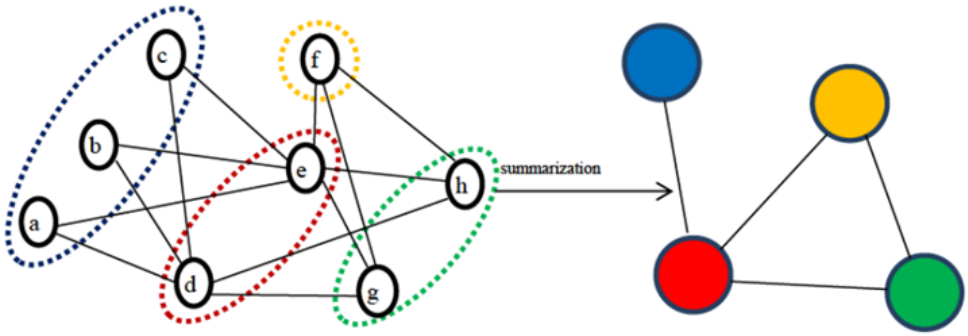


Figure 2.4: Graph summarization process [27].

All graphs can either be classified as stationary or as stream graphs [27]. A stationary graph's structure remains the same as time progresses, while a stream graph's structure changes over time due to nodes being added or removed from the graph. An example of a stationary graph would be a graph representing several cities and how they are connected, while a social media network can be seen as a stream graph since nodes are added as new members join the network over time.

Since this study is centred around the attributed graph of a plant process, summarization methods of stationary graphs will be the main focus as the structure of a process plant typically remains the same.

All the summarization methods are based on similarity, and this similarity can be structural, attribute-based, or a combination of the two. Therefore, the summarization methods can be classified as structural, attribute-based, or hybrid approaches.

2.4.1.1 Structural graph summarization

Navlakha et al. [29] discuss a graph summarization method that is based on the structural composition of the graph. The process works by merging two or more nodes with edges going to the same set (or a very similar set) of other nodes into supernodes. The edges going to each common neighbour are then replaced with a super edge. This process is illustrated in

Figure 2.5 [29], where the graph $G = (V(G), E(G))$ on the left is summarized to create the graph on the right $S = (V(S), E(S))$.

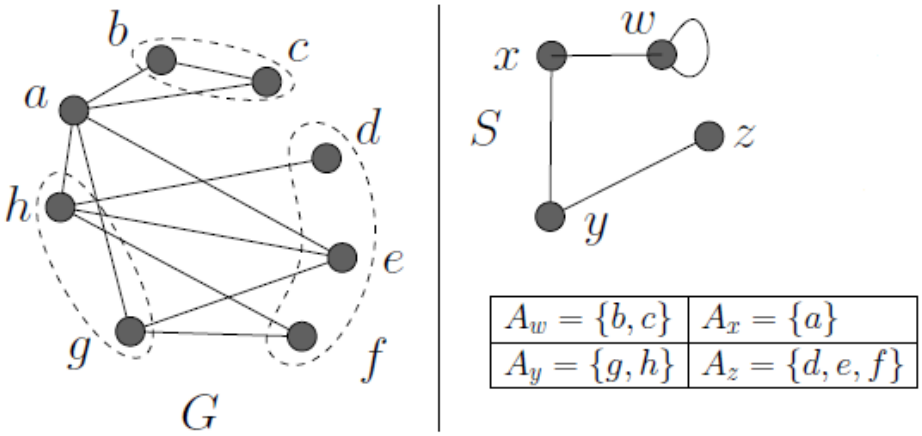


Figure 2.5: An example of structural graph summarization [29].

The graph representation G has a summary set S and a correlation set C, used to reconstruct the original graph. Navlakha et al. [29] define the cost of the representation $R = (S, C)$, which determines the sum of the storage costs of its two inputs. The Minimum Description Length (MDL) principle is applied to determine the best possible summary. If $\wedge R = (\wedge S, \wedge C)$ is the minimum cost representation, then according to the MDL principle, the summary $\wedge S$ is the best possible summary graph.

2.4.1.2 Attribute-based summarization

In the paper by Tian et al. [30], two summarization operations are proposed. The first operation is called SNAP (**S**ummarization by Grouping **N**odes on **A**ttributes and **P**airwise Relationships). The SNAP operation works by grouping nodes that are homogeneous in terms of relationships and attributes together. Edges are then used to show the relationship between the different groups.

An example of the SNAP operation can be seen in Figure 2.6 [30]. The original graph on the left represents students (nodes) with different attributes (gender, department) and the relationship between the students (edges). Not all the relationships are shown in the figure. The summary is created by grouping students of the same gender and department together, which results in four different groups (G_1, G_2, G_3, G_4). The edges are assigned to represent

the relationships between the different groups in the summary. The summarized graph can be seen on the right-hand side of the figure.

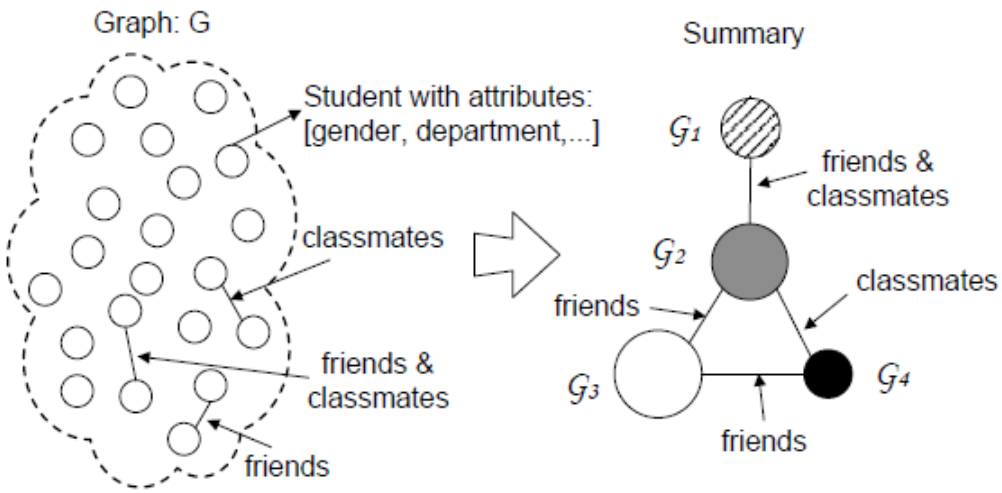


Figure 2.6: Illustration of the SNAP operation [30].

The second operation proposed by Tian et al. in [30] is called the k-SNAP operation. The k-SNAP operation relaxes the homogeneity requirements present in the SNAP operation by not requiring that every node in a group participates in a group relationship. The k-SNAP operation also allows the user to determine the size of the resulting summary. The user specifies the required number of groups in the summary, which is denoted by k.

An improvement on the k-SNAP operation is presented in a paper by Zhang et al. [31]. They propose using the CANAL (**C**ategorization of **A**ttributes with **N**umerical Values based on **A**ttribute Values and **L**ink Structures of Nodes) technique to summarize a graph. The CANAL technique automatically categorises the numerical values by assessing the similarities of the attribute values and the link structure of all the nodes in the graph.

The inputs to this algorithm are the graph $G = (V, E)$, the attributes of all the nodes denoted by a, and the number of categories required by the user denoted by C. It works by grouping nodes based on the attribute values of all the nodes in the graph. All the nodes in one group have the same numerical value. The groups are ordered numerically. The algorithm then iteratively merges groups based on the similarity of their link structures until one group remains.

During the merging process, the algorithm constantly determines the quality of the summary. If merging two groups significantly causes the summary quality to decrease, the boundary between the groups is an excellent cut-off position. A cut-off position splits two categories.

In the final step, the algorithm uses the boundaries of the $C - 1$ merging operations which produce the worst quality summaries, to categorise the numerical attributes.

The k-means clustering algorithm is another possible method that can be used to create a summarized graph based on node attributes. The algorithm sorts the data into k non-overlapping clusters. The user specifies the k value, and a cluster is represented by its centroid, which is the mean of all the points in the cluster [32]. This method is more suited for graphs that have numerical attributes.

The algorithm starts by selecting k centroids. Then, each point in the dataset is assigned to the closest centroid, and the collection of points around each centroid forms a cluster. Next, an updated centroid is assigned to the cluster according to the data points in that cluster. This process repeats until the data points stop changing clusters [32].

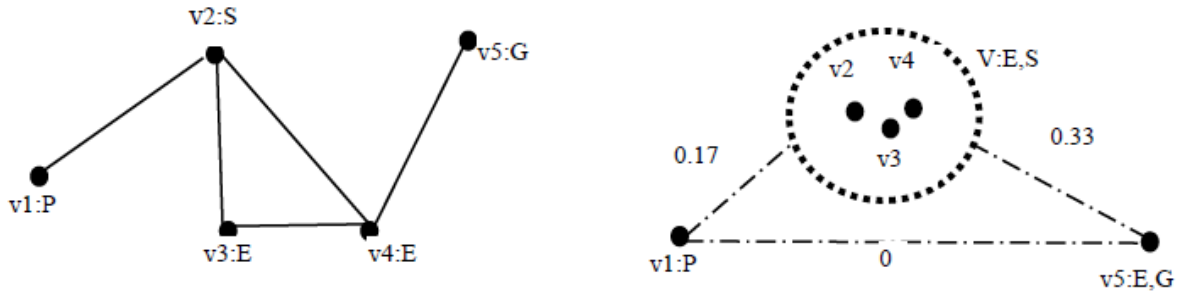
Although graph summarization and graph clustering are two distinct operations, in the paper by Riondato et al. [33] they exploit the connection between graph summarization and geometric clustering (k-means clustering) to develop an algorithm that is capable of producing a summarized graph. The clusters produced by the k-means clustering form the supernodes in the summarized graph.

2.4.1.3 Hybrid graph summarization

In [34], a method is proposed that creates a summary of a graph-based on both virtual and real edges. The technique is called SGVR (**S**ummarizing **G**raph based on **V**irtual and **R**eal links), and it works by aggregating similar nodes into non-overlapping groups using user-selected attributes. It considers both virtual edges, which represent node attributes, and actual edges, representing the graph structure.

Ashrafi & Kangavari [35] propose a new approach that generates a hybrid summary of an attributed graph by allowing the user to specify - in percentage - the contribution of the structural information to the summarized graph. This method further allows the user to specify the resulting summary's size and the importance of attributes if nodes have multi-valued similarities.

They also introduce the concepts of density and entropy, which are measures used to determine the quality of summarized graphs, depending on the type of summary. They compared their method with the method in [34] by using these two measures and found that their method results in a higher quality summary. Figure 2.7 [35] illustrates this summarization method. The summarized graph now contains a supernode and two regular nodes.



(a) (b)

Figure 2.7: Example of hybrid graph summarization:(a) Original graph. (b) Summarized graph [35].

2.4.1.4 Using a summarized graph for FDI

For graph comparison to be applied to a summarized graph, an attribute will have to be assigned to each supernode, and its value should represent all the individual nodes from the original graph that make up the supernode. Furthermore, this newly assigned attribute will have to be selected in such a way that the graph comparison process should be able to detect a fault at any of the individual nodes forming the supernode while not detecting a fault if any of the individual node attributes exhibit normal levels of variation.

While Jouili & Tabbone [24] show that it is possible to compare graphs of different sizes, comparing a reference graph with an operational graph after both have been summarized, will result in a more accurate FDI process than a comparison between one standard and one summarized graph, since this will lead to valuable data not being considered. This FDI process involves the summarized graph being compared with itself under normal operating conditions (NOC). When considering an eigendecomposition FDI method as an example, this is done by generating a cost matrix and calculating the eigenvectors and eigenvalues for the normal operation.

The attributes of the summarized graph of the system are then continually updated and monitored during operation as the plant measurements are updated. Condition monitoring then takes place, and it involves applying graph comparison operations and generating a cost matrix that determines the difference between the normal operation summarized graph and the monitored summarized graph. Finally, the eigenvectors and eigenvalues of this cost matrix are calculated and compared to those of the normal plant operation. A fault has occurred if the deviation between the two sets of eigenvectors and eigenvalues is significant enough.

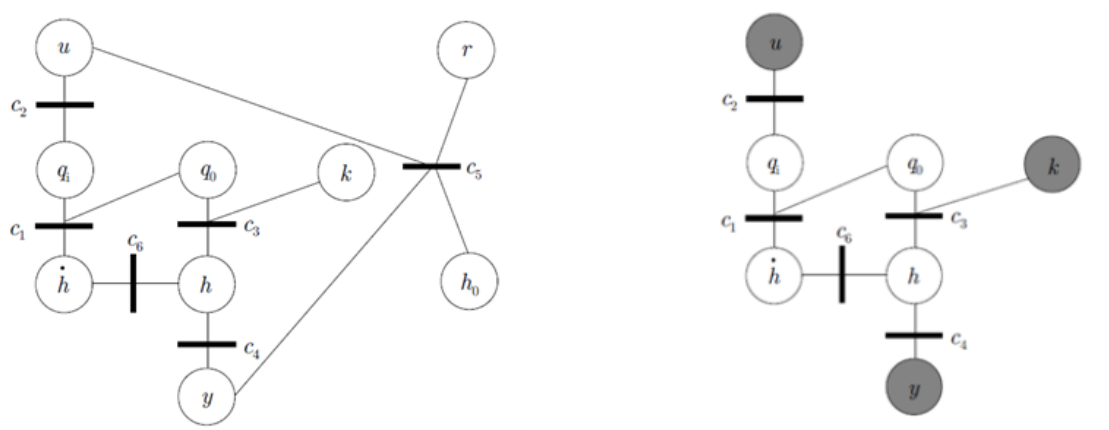
2.4.2 Graph reduction through attribute filtering

Since the reduced graph will undergo the graph comparison procedure to conduct FDI, only attributes that contribute to the FDI process are of interest in this reduced graph. Another way to reduce the attributed graph's size is to identify all the nodes and edges whose attributes can be used for fault diagnosis and filter out the remaining nodes and edges.

This filtering can be done by examining the attributed graph under fault conditions and observing how each node's attributes and edge change relative to the corresponding attributes of nodes and edges during NOC. By then contrasting each attribute's contribution to the detection of faults with the contribution of all the other attributes, it can be deduced which attributes contribute very little to the process, and their corresponding nodes or edges can then be filtered out.

This form of graph reduction is an adaptation of the graph reduction method used for structural analysis by Blanke et al. [36]. In the approach used by Blanke et al. [36], the constraints (or equations) which define a system are partitioned into a subset of constraints that contain only known variables and a subset of constraints that contain at least one unknown variable. A structure graph links together all the variables and constraints of a system. The reduced structure graph is created by removing all fixed inputs to the system and the subset of constraints that merely correspond to known variables.

An illustration of this reduction process can be seen in Figure 2.8 [36]. The particular variables and constraints involved in the figure are of lesser concern since the figure is only meant to convey the main idea behind this approach. The result of this reduction process is a graph that contains only the most important structural information while superfluous details are omitted.



(a) (b)

Figure 2.8: (a) Diagram of a structure graph. (b) Diagram of a reduced structure graph [36].

2.4.2.1 Using a graph reduced with attribute filtering for FDI

When reducing a graph by filtering the attributes of nodes and edges, none of the attributes is altered or represented by other values. The reduced graph with unaltered attributes is compared with itself under NOC, and a cost matrix is generated. Again using the eigendecomposition FDI method as an example, the eigenvectors and eigenvalues of this cost matrix under NOC are then calculated.

During plant operation, the attributes of the reduced graph are constantly monitored and updated by taking plant measurements. The monitoring process involves applying graph comparison operations and generating a cost matrix that determines the difference between the reduced graph under NOC and the monitored reduced graph. The eigenvectors and eigenvalues of this cost matrix are calculated and compared to those of the normal plant operation to determine if a fault has occurred.

2.4.3 Summary of graph reduction methods found in literature

Table 2.2 contains a detailed summary of all the methods discussed in this section as well as a comparison of the main strengths and weaknesses of all these methods. The type of reduction employed by each method is also provided in the table.

Table 2.2: Comparison of graph reduction techniques found in the literature.

Method	Type of reduction	Strengths	Weaknesses
Structural graph summarization	Navlakha et al. method [29]	Uses the MDL principle to create best-quality summary.	Does not consider graph attributes during the summarization process.
Attribute-based graph summarization	SNAP Algorithm [30]	Some structural information (links) of the original graph are retained in the summarized graph.	Difficult to determine homogeneity of different numerical values.
	k-SNAP Algorithm [30]	Homogeneity requirements are more relaxed than SNAP requirements.	Does not preserve structural information of original graph.
	CANAL Algorithm [31]	Clusters according to numerical values and summary quality is automatically verified.	Does not preserve structural information of original graph.
	Summarizing with k-means clustering [32],[33]	Ideal for graphs with numerical attributes and it is possible to verify the size of the summarized graph.	Does not preserve structural information of original graph.
Hybrid graph summarization	SGVR Method [34]	Allows user to select attributes used for summarization.	Not ideal for numerical attributes.
	Ashrafi & Kangavari Method [35]	Metrics are available to verify the quality of summarized graphs.	Technique used to determine degree of similarity of attributes is not ideal for numerical attributes.
Attribute reduction	Graph reduction through attribute filtering	Preserves structural information of original graph and ideal for numerical attributes.	No metrics available to verify quality of reduced graph.

2.5 Benchmark model - Tennessee Eastman process

Any fault detection scheme has to be evaluated to verify that it functions correctly according to the specific application and to ensure that it is efficient enough. One way of testing such a scheme is by using a benchmark process. The benchmark process will verify if the scheme

meets all the standard requirements of being a fault detection scheme, and it will allow the user to compare the scheme with other available fault detection schemes. This section investigates such a benchmark model which is specific to the petrochemical industry (PCI).

2.5.1 Tennessee Eastman process

In the paper by Downs & Vogel [37], the model of an industrial chemical plant process is described, which can be used to develop, investigate, and evaluate process control technologies. This process is called the Tennessee Eastman process. The process entails two simultaneous gas-liquid reactions as well as two additional by-product reactions. All the reactions are exothermic and irreversible.

The two gas-liquid reactions have the following form:

$$A(g) + C(g) + D(g) \rightarrow G(liq),\ Product\ 1,$$

$$A(g) + C(g) + E(g) \rightarrow H(liq),\ Product\ 2.$$

The process uses four reactants to produce two products. An inert and a by-product are also present in the process, which means that a total of eight components are present. The process also contains 12 valves that can be manipulated, and 41 measurements can be taken for the purposes of monitoring and/or control. The Downs & Vogel model has 20 disturbances making this process the ideal benchmark model for evaluating fault detection schemes.

The five main units of the Tennessee Eastman process are:

- The two-phase reactor
- The product condenser
- The vapour-liquid separator
- The recycle compressor
- The product stripper

The four reactants, all in gas form, are fed to the reactor, where they react to form liquid products. The heat from the reaction is removed by the internal cooling bundle of the reactor. The products leaving the reactor are in a vapour form and are condensed by the product condenser before passing to the vapour-liquid separator. The uncondensed components are then recycled back to the reactor feed with a compressor. The condensed components proceed to the product stripper, which removes the remaining reactants from the stream. The products G and H then exit the stripper in liquid form, but the section which separates these two

products is not included in the model. The process model as adapted for Simulink by Vosloo et al. in [12] can be seen in Figure 2.9.

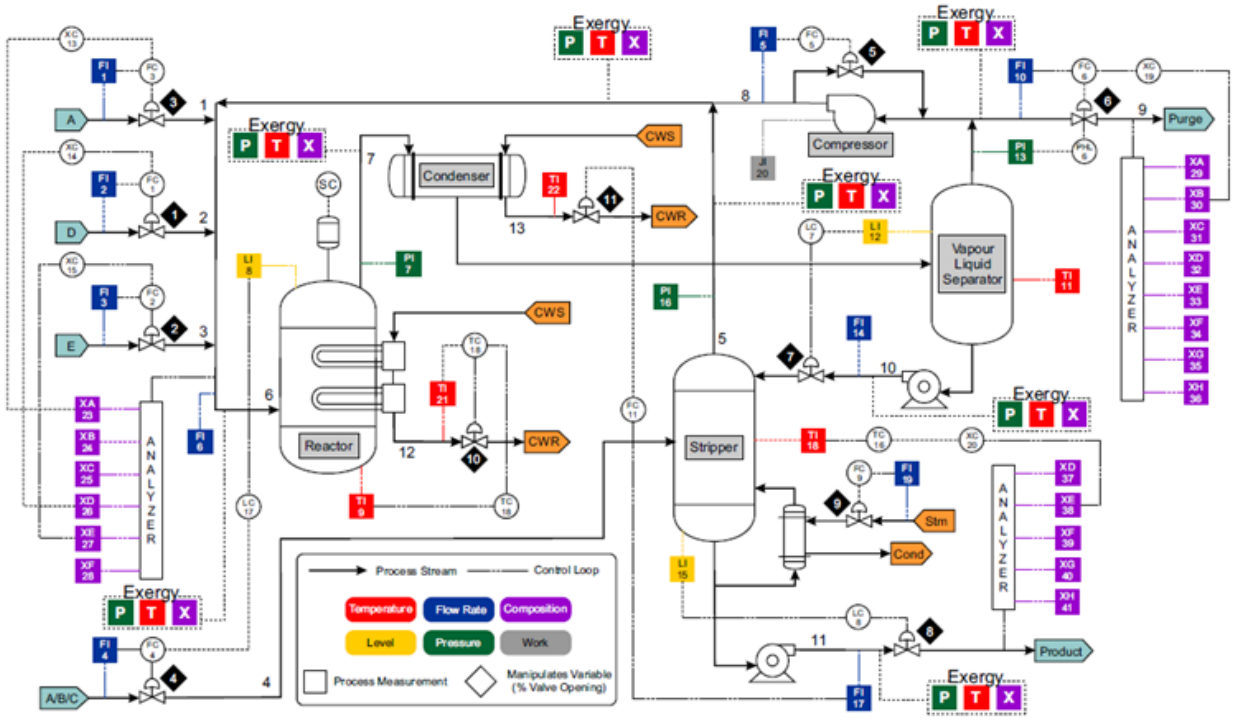


Figure 2.9: Tennessee Eastman process model [12].

The TEP's main advantage is that it has been used several times as a benchmark model to evaluate FDI schemes. It has become increasingly popular to apply data-driven schemes to the TEP with advancements in computer systems. These data-driven schemes have shown improved detection rates compared to other schemes published in literature [38], [39]. There has also been an increase in studies that implement hybrid schemes on the TEP. These studies indicate that the hybrid schemes have better diagnostic capabilities than their single method counterparts [40], [41]. These studies verify that the TEP is an excellent benchmark for evaluating hybrid and data-driven FDI schemes.

2.6 Critical literature review

It is clear from the research problem that energy visualisation, graphs, graph reduction, and the benchmark system are the four main topics of interest in this study. Based on the analysis conducted in this chapter, exergy-based FDI schemes hold significant potential and are, therefore, the FDI schemes of choice for this study.

Using graph comparison as a means of FDI as proposed by Van Schoor & Uren [6] has proven to be an effective strategy for implementing condition monitoring [7], [8] & [26]. Since using graph reduction techniques to reduce the complexity of these graph-based FDI schemes is relatively new, it is unclear which graph comparison approach will work best after the reduction process is complete. It is, therefore, prudent to not only evaluate the reduction techniques on one specific FDI scheme but to rather test a variety of schemes and determine which reduction techniques complement which schemes.

When reviewing the different techniques available to reduce the attributed graph data, it is vital to consider the effect each technique has on the structural information contained in the attributed graph. *Structural information* entails the information each graph element represents. Structural information can be seen as knowing that Node A represents component A in the process and knowing Link AB represents the connection between components A & B.

While structural information will inevitably be lost when the graph is reduced, albeit through summarization or attribute filtering, distorting the original structural information too severely results in more processing resources being needed to generate the reduced attributed graph, thus increasing graph complexity. If structural information is removed in a random and unrestricted fashion, it would, for instance, make it difficult to determine which process sensors can be removed to reduce complexity.

When comparing graph reduction through attribute filtering and graph reduction through summarization as explained in this section (see Table 2.2), it is evident that graph reduction through attribute filtering is superior at preserving and not distorting structural information since nodes of the original graph are not grouped into supernodes.

By altering the techniques and implementing certain restrictions, it is possible to reduce graphs through summarization and attribute filtering. By removing structural information in a methodical manner that is retraceable, and by restricting the extent to which structural information may be distorted (summarized), graph reduction techniques can alter and/or reduce structural information in such a way that graph complexity as well as the complexity of implementing the graph in an FDI scheme, are both reduced.

It is also worth exploring the effects of combining the graph summarization approach with the attribute filtering approach. This combined approach holds the potential of combining some of the strengths of each individual approach while also overcoming some of their weaknesses.

Selecting the Tennessee Eastman process as the primary benchmark system for this study is a logical choice, given that it is often used to assess fault diagnostic schemes. It is also a

complex process with many fault conditions, which will ensure that any evaluation performed on it is robust. The more detailed literature relevant to this study will be discussed in relevant chapters that follow.

Chapter 3

Tennessee Eastman process model and attributed graph

3.1 Introduction

This chapter reviews specific literature which relates to the exergy characterisation of the TEP model and the development of its attributed graph. First, an overview of the Simulink® model of the TEP is given, describing certain features, such as the model validation. All the fault conditions of the TEP are then introduced and briefly discussed. The chapter concludes by providing a detailed explanation of how an exergy characterisation is conducted, and the attributed graph is developed from the process model.

3.2 TEP model overview

The Simulink® model of the TEP used in this study was initially developed as part of a study conducted by Vosloo [42]. This Simulink® model was developed from the FORTRAN source code provided in the article by Downs and Vogel [37]. The control strategy used to control the TEP is the strategy proposed by Lyman and Georgakis [43] and can be seen in Figure 2.9. This control strategy also uses the controller parameters as suggested by Chiang et al. [44]. A sampling interval of 180 seconds is used to take sensor measurement imperfections into account.

The validation of the TEP model constitutes two steps. The first step involves comparing process measurements and manipulated variables obtained from the FORTRAN source code

with the data obtained from running the open-loop TEP simulation with steady-state values as the initial condition. The second step entails comparing the closed-loop TEP Simulink® simulation results with the results produced by simulating the TEP with the results from the FORTRAN code used by [44]. The comparison conducted during each step indicates that the results obtained from the Simulink® model and the results obtained from the FORTRAN code have a reasonable correlation.

3.3 Fault conditions

Table 3.1 contains a description of all 20 fault conditions of the TEP. Fault conditions 1 to 7 are all the result of a step change applied to the respective process variable of each fault. Fault conditions 8 to 12 are all the result of random variations occurring in the relevant process variables. Fault condition 13 emulates a slow drift within the kinetics of the process reactions. Both fault conditions 14 and 15 are caused by sticking valves. Fault conditions 16 to 20 are all unknown conditions.

Table 3.1: A description of the fault conditions of the TEP.

Fault ID	Disturbed process variable	Type of disturbance
Fault 1	A/C feed ratio, B composition constant (stream 4)	Step
Fault 2	B composition, A/C feed ration constant (stream 4)	Step
Fault 3	D feed temperature (stream 2)	Step
Fault 4	Reactor cooling water inlet temperature	Step
Fault 5	Condenser cooling water inlet temperature	Step
Fault 6	A feed loss (stream 1)	Step
Fault 7	C header pressure loss - reduced availability (stream 4)	Step
Fault 8	A,B,C feed composition (stream 4)	Random variation
Fault 9	D feed temperature (stream 2)	Random variation
Fault 10	C feed temperature (stream 4)	Random variation
Fault 11	Reactor cooling water inlet temperature	Random variation
Fault 12	Condenser cooling water inlet temperature	Random variation
Fault 13	Reaction kinetics	Slow drift
Fault 14	Reactor cooling water valve	Sticking
Fault 15	Condenser cooling water valve	Sticking
Fault 16	Unknown	Unknown
Fault 17	Unknown	Unknown
Fault 18	Unknown	Unknown
Fault 19	Unknown	Unknown
Fault 20	Unknown	Unknown

3.4 Calculating exergy attributes

The TEP Simulink® model calculates the exergy of the process streams by using data such as the physical properties, chemical composition, enthalpy, and entropy of those streams [12].

To calculate the physical exergy of the TEP, the formulas given by equations 2.6 - 2.8 are used. To accurately calculate the chemical exergy of the TEP requires the standard chemical exergy of all substances involved [12].

Several studies have been dedicated to developing methods which can be used to determine the standard chemical exergy of a variety of substances [45], [46], [47]. However, these methods cannot be used on the components of the TEP since these components have an unknown nature. The only information that could be obtained from the TEP source code about these components are some of the physical and thermodynamical properties such as molecular weight, heat capacities, and liquid densities [12].

Based on the work done by Gharagheizi et al. [48], [49], & [50], Vosloo et al. [12] used linear regression analysis to establish a mathematical correlation between the known physical and thermodynamical properties and the standard chemical exergy of the TEP. This linear regression analysis was done separately for substances in vapour and liquid phases, seeing as the chemical exergy differs for different phases.

For the components in the vapour phase, the molecular weight $MW_{(i)}$ and heat capacity $Cp_{v(i)}$ of the vapours are available in literature. The linear regression analysis produced an equation for standard chemical exergy of substances in the vapour phase [12], which is expressed as

$$b^0_{ch(i)v} = -510261 + 25667Cp_{v(i)} + 13745MW_{(i)}. \tag{3.1}$$

For the components in the liquid phase, the molecular weight, liquid heat capacity $Cp_{l(i)}$, as well as the liquid density $\rho_{(i)}$ are available in literature. The linear regression analysis again produced an equation for standard chemical exergy of TEP substances in the liquid phase [12], which is expressed as

$$b^0_{ch(i)l} = 1537576 + 112.65Cp_{l(i)} + 49487MW_{(i)} - 2973515\rho_{(i)}. \tag{3.2}$$

A comparison of the standard chemical exergy values of substances in the vapour phase calculated with equation 3.1, and the values obtained from literature resulted in a correlation coefficient of $R^2 = 0.92$ [12]. Similarly, a comparison of the standard chemical exergy values of substances in the liquid phase calculated with equation 3.2, and the values obtained from literature resulted in a correlation coefficient of $R^2 = 0.97$ [12]. These results indicate that both equations 3.1 & 3.2 are sufficient means of determining standard chemical exergy in both the vapour and liquid phases.

3.5 Attributed graph of the TEP

Uren et al. [51] mathematically define an attributed graph as $\mathcal{G}_{\mathcal{A}} = (\mathcal{N}, \mathcal{L}, \mathcal{A})$, where $\mathcal{N}$ is the finite set of nodes that is non-empty, $\mathcal{L}$ is the finite set of links, and $\mathcal{A}$ the set of attributes. The set of attributes is defined as $\mathcal{A} = a_{n_i}, a_{l_{j,k}}$, with $n_i \, \epsilon \, \mathcal{N}$ representing node number i, and $l_{j,k} \, \epsilon \, \mathcal{L}$ representing the link between nodes j and k.

By using the procedure as stipulated by Van Schoor et al. in [6], an attributed graph could be constructed for the TEP model, which is displayed in Figure 3.1. The diagram of this attributed graph, as developed by Vosloo [42], can be seen in Figure 3.2. The total change in exergy (chemical plus physical) over node i is represented with ΔB_i and the energy flow rate from the source node j to the destination node k, is represented with $\dot{q}_{j,k}$.

The graph is constructed by representing each system component as a node, and in the cases where the component functions as a heat exchanger, that component is represented with two nodes. These two nodes, representing the primary and secondary sides of heat exchange, are connected with a link representing heat exchange from one side to the other. The links, which represent the connection between system components, are directional and indicate from which node to which node energy is transferred. Table 3.2 details the process component corresponding to each node in the attributed graph of the TEP.

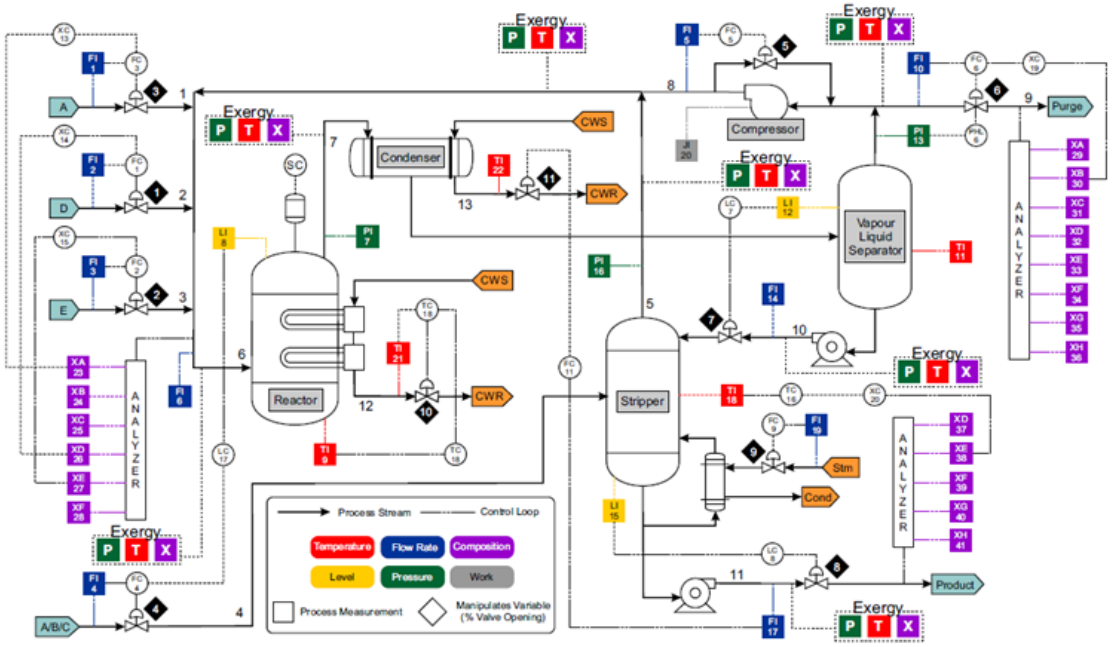


Figure 3.1: Tennessee Eastman process model [12].

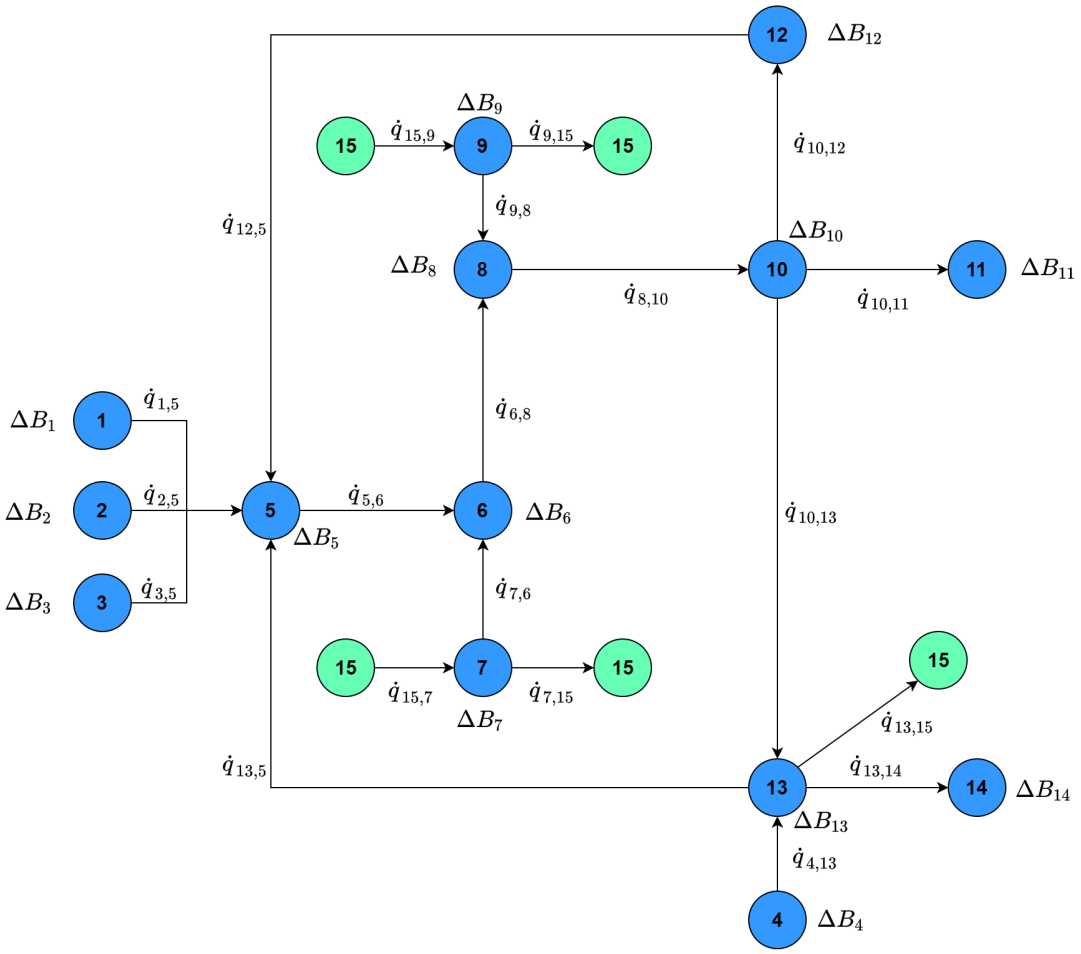


Figure 3.2: Diagram of the attributed graph of the TEP.

Table 3.2: Summary of all the nodes of the TEP attributed graph and the process components they represent.

Process component	Node number	Process component	Node number
Feed stream A	Node 1	Feed stream D	Node 2
Feed stream E	Node 3	Feed stream A/B/C	Node 4
Mixing zone	Node 5	Reactor	Node 6
Reactor cooling bundle	Node 7	Condenser	Node 8
Condenser cooling side	Node 9	Vapour-liquid separator	Node 10
Purge stream	Node 11	Compressor	Node 12
Stripper	Node 13	Product stream	Node 14
Environment	Node 15	-	

For the attributed graph to be used by an FDI scheme, it must first be converted into NSM format. For FDI, Wolmarans [52] provided a mathematical expression of an NSM. The system used in that study, however, only had physical exergy and the expression had to be modified to

$$N_s = [\mathcal{N}(G), \mathcal{L}(G)], \quad N_s \, \epsilon \, \mathcal{R}^{n \times n+2} \tag{3.3}$$

with $\mathcal{N}(G)\ \epsilon\ \mathcal{R}^{n\times 2}$ representing the matrix of node attributes, and $\mathcal{L}(G)\ \epsilon\ \mathcal{R}^{n\times n}$ representing the matrix of link attributes. The NSM of the attributed graph of the TEP as illustrated in Figure 3.2, can be expressed as

$$N_s = \left[\begin{array}{cc|ccc} \Delta B_{ch1} & \Delta B_{ph1} & \dot{q}_{1,1} & \cdots & \dot{q}_{1,14} & \dot{q}_{1,15} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \Delta B_{ch14} & \Delta B_{ph14} & \dot{q}_{14,1} & \cdots & \dot{q}_{14,14} & \dot{q}_{14,15} \\ \Delta B_{ch15} & \Delta B_{ph15} & \dot{q}_{15,1} & \cdots & \dot{q}_{15,14} & \dot{q}_{15,15} \end{array}\right]. \tag{3.4}$$

Certain rules apply when populating the NSM from an attributed graph. These rules include:

- The diagonal entries of the link attribute matrix of N_s are all zero, seeing as no node is connected to itself. This means $\dot{q}_{j,k} = 0$, where $j = k$.
- In the case where no energy is transferred between nodes and no connection exists between those nodes; the link attribute is set to zero.
- Since links are directional; the link attribute is multiplied with -1 when the link is reversed. This is expressed as $\dot{q}_{1,5} = -\dot{q}_{5,1}$. This is, however, not the case when the reverse of a link is a separate link. This can be seen in Figure 3.2 with the environmental node, where the link from node 7 to node 15 is separate and different from the link from node 15 to node 7. This is expressed as $\dot{q}_{7,15} \neq -\dot{q}_{15,7}$.

3.6 Conclusion

This chapter provides an overview of the TEP Simulink® model and touches on critical aspects such as the control strategy and the validation of the model. First, a table of all 20 fault conditions of the TEP is provided, and the type of faults are then discussed briefly. Next, more detail is provided on how the model calculates the chemical and physical exergy attributes used to populate the attributed graph. The chapter then concludes by explaining how the attributed graph and NSM of the TEP are developed.

Chapter 4

Exergy-based fault detection and isolation

4.1 Introduction

In this chapter, a set of control data is obtained by applying three FDI schemes, which utilize the unreduced attributed graph, to the graph data of the benchmark system. This set of control data is required to evaluate how effective the graph reduction techniques are by comparing the performance of the FDI schemes, which use reduced attributed graphs, with the performance of schemes in the control data.

The distance parameter method, the eigendecomposition method, and the residual-based method are used in this study. Both the distance parameter method and the eigendecomposition method require the generation of cost matrices, while the residual-based method only requires the generation of residual matrices.

This chapter provides an extensive overview of the methodology used to implement each of these three FDI schemes. The performance of each of these FDI schemes is gauged by the overall detection and isolation rates achieved by each of these methods. It is also shown how the time-series data produced by a dynamic system are sampled to implement the FDI methods.

4.2 Generating control data for FDI schemes

Using graph reduction techniques in the context of graph-based FDI schemes is a relatively novel approach, and as such, it is unclear exactly which FDI schemes will function better than others after the reduction process is applied. To this end, more than one FDI scheme is required to evaluate the reduction techniques. Furthermore, for the schemes to be used to evaluate reduction techniques, a control dataset is required where the graphs have not yet been reduced. A process flow diagram of how these FDI schemes are implemented can be seen in Figure 4.1. The FDI schemes mentioned in the literature study were all applied to static systems, while the benchmark system is a dynamic system with time-series data. In the dissertation done by Smith [53], three of the FDI methods referenced in the literature study are applied to a system with time-series data. These methods involve generating a cost matrix and then evaluating the distance parameter and eigenvalues, and generating residual matrices to diagnose faults. Figure 4.2 illustrates the different graph comparison operations that are applied to the NSM for each FDI scheme.

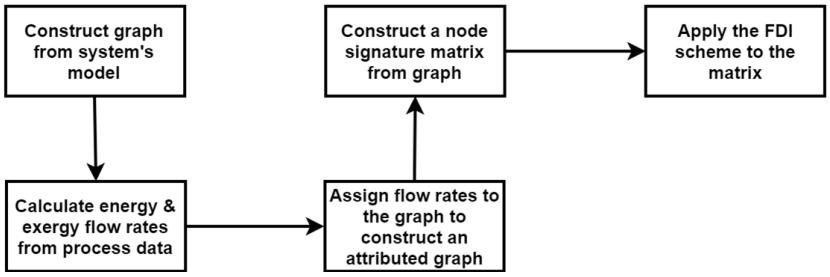


Figure 4.1: Process flow diagram of a graph-based FDI scheme.

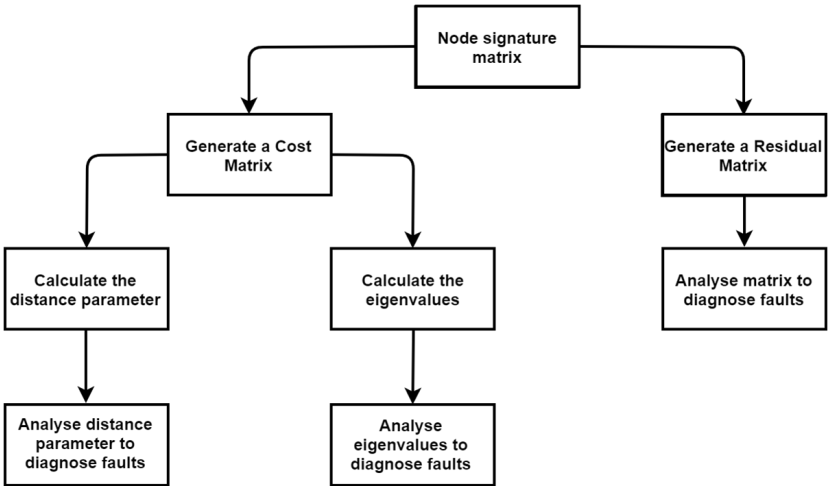


Figure 4.2: Illustration of how the FDI schemes use different graph comparison operations to diagnose faults from the NSM.

4.2.1 Generating a cost matrix

As stated in Chapter 2, a cost matrix is a way of quantifying the similarity between two matrices that are being compared. Both the distance parameter approach and the eigenvalue approach require a cost matrix. Several different methods are available to generate a cost matrix and a few variations of these methods that can be used in different applications. In [53], the chosen method for determining the cost matrix is the Heterogeneous Euclidean-Overlap Metric (HEOM) as proposed by Jouili et al. [24]. The HEOM equation proposed in [24] was adapted specifically for FDI in [51]. This adapted HEOM equation is given by

$$HEOM(N_{s1}, N_{s2}) = \sqrt{\sum_{a=1}^{k} \delta(N_{s1}(i,a), N_{s2}(j,a))^2}, \tag{4.1}$$

where $N_{s1}(i,a)$ represents the (i,a) entry of the first matrix, $N_{s2}(j,a)$ represents the (j,a) entry of the second matrix, a represents the $a-th$ column of the rows that are being considered, and k represents the row length. When the matrices contain numeric entries, the δ function is given by

$$\delta(N_{s1}(i,a), N_{s2}(j,a)) = \frac{|N_{s1}(i,a) - N_{s2}(j,a)|}{range_a}, \tag{4.2}$$

where the function $range_a = max_a - min_a$ normalizes the attributes by using both the maximum and minimum values in column a.

While the work done by Wilson & Martinez [54] offers several options for determining δ, such as using $range_a$ or substituting the range parameter with either one or four standard deviations (σ or 4σ) of the data, an inspection done in [53] showed that substituting the range yielded a negligible difference in results. Thus, $range_a$ as used in (4.2) will be used in this study.

As an example of how a cost matrix is generated, consider the illustration provided in Figure 4.3. The cost matrix C is generated by comparing matrices A and B using HEOM, with matrix A being the reference matrix and matrix B being some operational matrix. As such, the range is calculated by using the values in matrix A. In the context of FDI, matrices A and B would be two node signature matrices.

It is critical to consider which cost matrix to calculate so as to use this matrix to detect and isolate fault conditions in the system. To explain how it is determined which cost matrix to use, consider the following scenario: A system is under NOC up until $time = T_f$ when a fault

condition is introduced into the system. It can be assumed that when the system is under NOC, there are variations present in the measured process data, but these variations are all within an acceptable range as set out in the design specifications. Once a fault is introduced into the system, the energy and exergy attributes in the NSM will deviate to levels outside the bounds of a system's normal variation range under NOC.

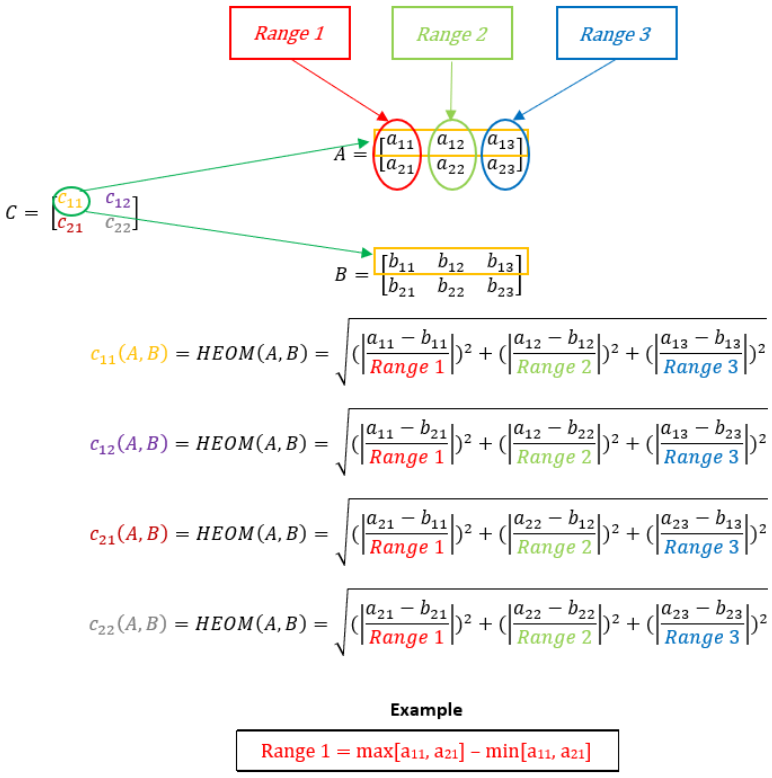


Figure 4.3: Illustration of the cost matrix generation process.

For any fault condition to be detected, the energy and exergy attributes in the NSM before T_f should not present any problems, while the attributes in the NSM at T_f and onward should be indicative of a fault condition present in the system. To differentiate between the different conditions an NSM can be in, an NSM with measurements taken under NOC is referred to as a normal NSM, and an NSM with measurements taken during a fault is referred to as a fault NSM.

When graph comparison is applied as part of an FDI scheme to generate a cost matrix analysed by the scheme to diagnose faults, measured process data in the form of energy and exergy attributes are the input to the scheme. Measurements taken while the system is operational are used to construct an operational attributed graph and, subsequently, an operational NSM. The FDI scheme diagnoses faults by comparing this operational attributed graph with a reference normal attributed graph representing the system under NOC. The rationale is that because a

cost matrix constitutes the similarity between two compared graphs, the cost matrix generated by comparing the reference normal attributed graph and the operational attributed graph when the system is under NOC will have small values. In juxtaposition, when comparing the operational attributed graph with the reference normal attributed graph while a fault condition is present, it will generate a cost matrix with large values.

For the FDI scheme to detect and isolate faults, a reference attributed graph must be generated for the system under NOC (the reference normal attributed graph) and a reference attributed graph for each fault condition. Since the TEP has 20 fault conditions, a database of 21 reference attributed graphs is constructed. The operational attributed graph is then compared with each of the reference attributed graphs, generating a cost matrix. The degree of similarity the cost matrix represents is quantified differently depending on whether the distance parameter approach or eigenvalue approach is used. By identifying the cost matrix with the slightest degree of similarity, the operational attributed graph can be linked to a specific reference attributed graph, and the fault is isolated.

This approach of comparing an operational attributed graph to a reference graph hinges on considering all possible fault conditions. In a practical setting, however, this may not be the case. For example, if an unknown and unreferenced fault occurs, the results from the graph comparison process will be inconclusive.

As mentioned earlier, the TEP model is dynamic and generates time-series data. To construct each reference attributed graph in the database, the average of each node and link attribute is calculated from the time-series data for each one of the 21 reference conditions. Each average reference attributed graph of a system condition includes 20 samples of attributed graphs while the system has no fault and is under NOC to make the reference attributed graphs more robust. These 20 samples represent the first hour in the 25-hour simulation period. This also better equips these reference attributed graphs for practical use of the FDI schemes since most process plants are dynamic and do not transition from normal operation to faulty operation instantaneously, but instead start experiencing faults and their effects over time.

In FDI applications, it is typical to split a dataset and use 70% of the data for training the FDI scheme and use the remaining 30% to test the FDI scheme. Smith [53] notes that since average values are being used, the characteristics of the average testing sample (70% of the dataset) and the average of the complete dataset will show no discernible differences. The entire dataset is therefore used to compile the average values in this study.

Figure 4.4 contains an illustration of how the energy and exergy attributes in the dataset obtained from the simulation model are processed to create a reference database that is used

in the graph comparison operation. The operational attributed graph as illustrated with G_{OP} is the graph generated from the attributes as measured while the TEP model is in operation. This operational attributed graph can be generated for each one of the 21 system conditions.

To implement graph comparison, a MATLAB® program is used to compose an operational attributed graph from the process measurement data as supplied by the Simulink® model. The attributed graph is then converted by the program into an NSM, which the program then compares the NSM with the HEOM function to each of the average reference attributed graphs in the database. Each reference attributed graph in the database is stored as an NSM to ease the process.

A single graph comparison operation performed on one sample of an operational condition produces 21 cost matrices. To detect a fault, all 21 of these cost matrices should undergo either the distance parameter approach or the eigenvalue approach to quantify the degree of similarity each matrix represents. If it is found that the cost matrix generated by comparing the operational attributed graph with the reference normal attributed graph has the highest similarity of all the other matrices, the system is under NOC, and no fault is present. If this is not the case, a fault condition is present in the system.

Figure 4.5 is an illustration of how the cost matrices are produced by comparing an operational condition with the reference conditions. The operational attributed graph is under NOC, and each one of the 501 samples is compared with each reference attributed graph in the database to produce an array of cost matrices with 21 columns representing the reference graphs and 501 rows representing each sample of the operational graph. Each operational condition produces its own array of cost matrices, so a total of 21 arrays are generated. Figure 4.5 only displays the array produced by comparing an operational attributed graph under NOC with the reference attributed graphs for demonstrative purposes.

Table 4.1 provides an example of a cost matrix generated by applying the HEOM function. One sample of an operational attributed graph under NOC is compared with the normal reference attributed graph to produce the specific cost matrix. Table 4.2 contains the cost matrix generated by comparing one sample of the operational attributed graph under NOC with the reference attributed graph of Fault 1.

It should be noted that when an operational attributed graph is compared with its corresponding reference attributed graph, as is the case in Table 4.1, the diagonal entries are all close to zero. This is in contrast with when an operational attributed graph is compared with the incorrect reference attributed graph, as is the case in Table 4.2, which results in much larger diagonal entries. The red entry in Table 4.2 is much larger than its counterpart in Table 4.1.

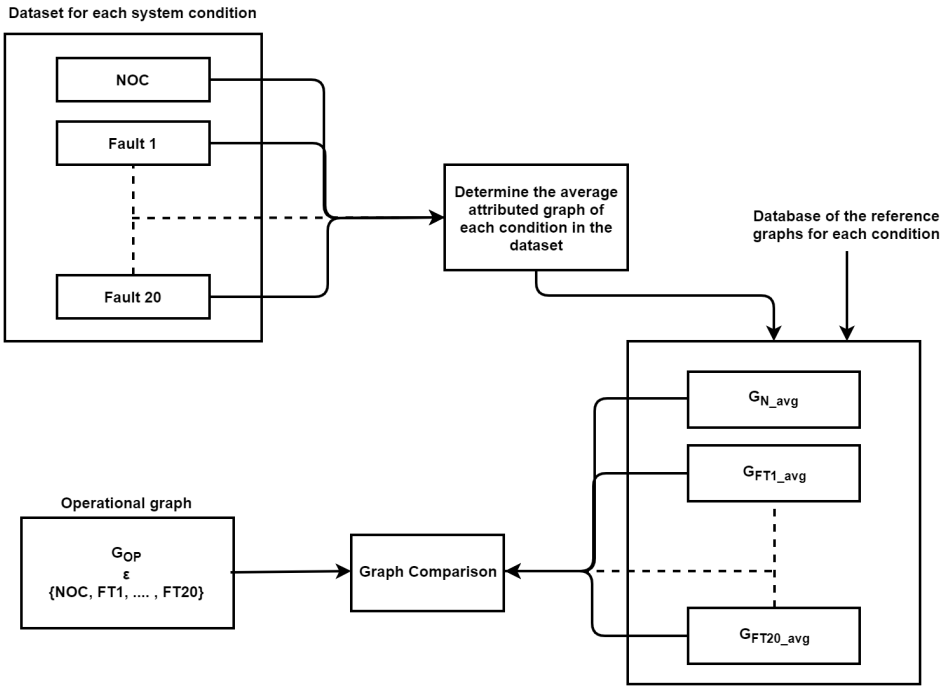
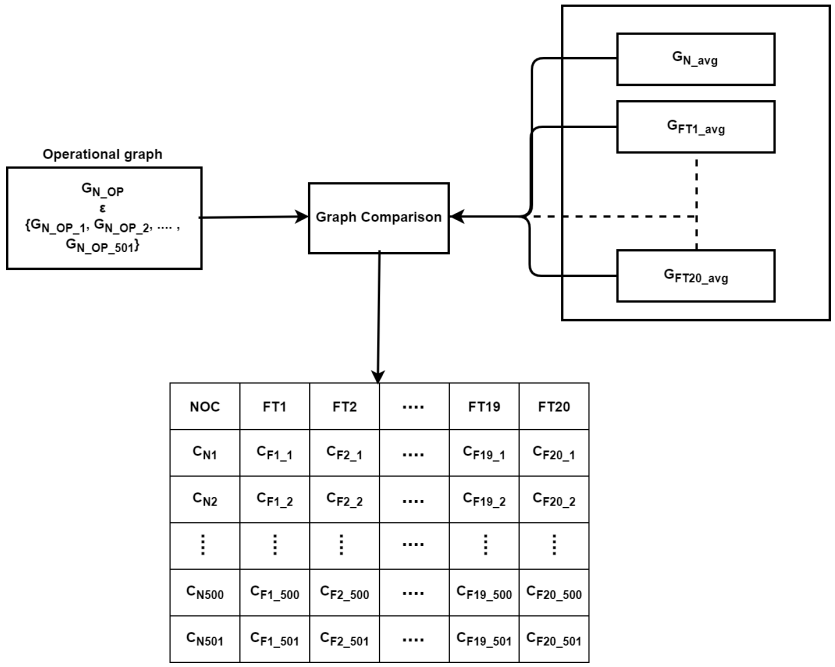


Figure 4.4: Illustration of how time-series data are used to perform graph comparison.



NOC	FT1	FT2	....	FT19	FT20
C_{N1}	C_{F1_1}	C_{F2_1}	....	C_{F19_1}	C_{F20_1}
C_{N2}	C_{F1_2}	C_{F2_2}	....	C_{F19_2}	C_{F20_2}
⋮	⋮	⋮	....	⋮	⋮
C_{N500}	C_{F1_500}	C_{F2_500}	....	C_{F19_500}	C_{F20_500}
C_{N501}	C_{F1_501}	C_{F2_501}	....	C_{F19_501}	C_{F20_501}

Figure 4.5: Generating the cost matrices for an operational condition.

Table 4.1: Cost matrix generated by comparing an operational attributed graph under NOC with the normal reference attributed graph.

C_{N1}														
0.0005	0.0289	0.0824	0.1088	1.9733	0.7574	1.1461	1.3036	0.4635	1.3407	0.0886	0.5913	1.4971	0.5312	1.4116
0.0283	0.0011	0.0906	0.1188	1.9750	0.7760	1.1467	1.3014	0.4651	1.3416	0.0961	0.5799	1.4882	0.5299	1.4121
0.0828	0.0917	0.0001	0.1258	1.9884	0.7242	1.1434	1.2640	0.4562	1.3431	0.0092	0.6544	1.5047	0.4891	1.4092
0.1085	0.1188	0.1253	0.0007	1.9966	0.7444	1.1491	1.3063	0.4723	1.2976	0.1293	0.6163	1.5092	0.6139	1.4076
1.9790	1.9806	1.9942	2.0017	0.0431	2.1595	2.3792	2.6328	2.0365	2.5971	1.9966	1.9306	2.4825	2.0246	2.4575
0.7543	0.7731	0.7210	0.7413	2.1514	0.0097	1.3595	1.3854	0.9479	1.6805	0.7195	1.2009	1.8179	0.8441	1.5435
1.1590	1.1597	1.1563	1.1620	2.3815	1.3729	0.0143	1.6026	0.8294	1.7773	1.1569	1.2945	1.8930	1.2584	1.8229
1.3058	1.3038	1.2660	1.3086	2.6298	1.3859	1.5967	0.0071	1.3614	1.8539	1.2648	1.7062	2.0428	1.2395	1.8754
0.4631	0.4648	0.4557	0.4719	2.0307	0.9498	0.8189	1.3592	0.0006	1.3620	0.4562	0.7791	1.5704	0.6701	1.4810
1.3519	1.3528	1.3537	1.3118	2.5979	1.6892	1.7773	1.8553	1.3724	0.0516	1.3538	1.5010	2.0269	1.6094	1.9259
0.0893	0.0973	0.0095	0.1300	1.9909	0.7226	1.1439	1.2628	0.4568	1.3433	0.0003	0.6612	1.5047	0.4860	1.4092
0.5937	0.5819	0.6574	0.6188	1.9244	1.2088	1.2847	1.7073	0.7818	1.4883	0.6640	0.0042	1.5071	0.9409	1.5554
1.4984	1.4895	1.5065	1.5104	2.4762	1.8218	1.8864	2.0438	1.5722	2.0205	1.5066	1.5049	0.0189	1.5884	2.0630
0.5341	0.5332	0.4922	0.6176	2.0160	0.8485	1.2477	1.2393	0.6727	1.6142	0.4892	0.9387	1.5857	0.0045	1.5203
1.4200	1.4205	1.4175	1.4160	2.4579	1.5521	1.8211	1.8803	1.4891	1.9230	1.4175	1.5617	2.0673	1.5266	0.0144

Table 4.2: Cost matrix generated by comparing an operational attributed graph under NOC with the reference attributed graph of Fault 1.

C_{F11}														
0.0004	0.0292	0.0805	0.1053	3.1858	0.7347	1.1531	1.2795	0.5092	1.1737	0.0877	0.5792	1.4498	0.5192	1.5082
0.0283	0.0016	0.0888	0.1156	3.1869	0.7531	1.1538	1.2772	0.5106	1.1748	0.0952	0.5685	1.4410	0.5179	1.5087
0.0828	0.0922	0.0019	0.1218	3.1952	0.7007	1.1505	1.2393	0.5026	1.1767	0.0077	0.6428	1.4570	0.4769	1.5059
0.1085	0.1190	0.1243	0.0035	3.1998	0.7220	1.1562	1.2822	0.5171	1.1257	0.1286	0.6038	1.4618	0.6017	1.5044
1.9791	1.9806	1.9937	2.0012	1.6864	2.1522	2.3833	2.6180	2.0459	2.5077	1.9962	1.9270	2.4556	2.0225	2.5144
0.7543	0.7735	0.7218	0.7404	3.2994	0.0235	1.3653	1.3630	0.9726	1.5457	0.7197	1.1855	1.7726	0.8377	1.6320
1.1590	1.1597	1.1563	1.1616	3.4506	1.3604	0.0074	1.5893	0.7963	1.6557	1.1568	1.2890	1.8556	1.2535	1.8986
1.3057	1.3040	1.2670	1.3072	3.6285	1.3724	1.6006	0.0330	1.3802	1.7357	1.2648	1.7011	2.0040	1.2377	1.9469
0.4631	0.4649	0.4557	0.4709	3.2217	0.9292	0.8245	1.3363	0.0492	1.2005	0.4561	0.7696	1.5249	0.6611	1.5733
1.3519	1.3528	1.3537	1.3131	3.5999	1.6752	1.7817	1.8383	1.3882	0.2479	1.3538	1.4957	1.9917	1.6016	1.9974
0.0892	0.0978	0.0110	0.1260	3.1968	0.6990	1.1510	1.2379	0.5031	1.1769	0.0019	0.6498	1.4570	0.4737	1.5059
0.5938	0.5815	0.6557	0.6194	3.1549	1.1901	1.2920	1.6886	0.8067	1.3412	0.6631	0.0219	1.4668	0.9317	1.6435
1.4984	1.4894	1.5063	1.5102	3.5194	1.8070	1.8908	2.0272	1.5860	1.9142	1.5067	1.5073	0.0532	1.5845	2.1302
0.5341	0.5334	0.4931	0.6135	3.2147	0.8270	1.2541	1.2145	0.7053	1.4746	0.4896	0.9312	1.5398	0.0159	1.6107
1.4199	1.4205	1.4175	1.4159	3.5063	1.5412	1.8256	1.8636	1.5039	1.8114	1.4175	1.5568	2.0328	1.5220	0.1268

4.2.2 Distance parameter method

When this approach was applied in [53], it was argued that the diagonal entries of a cost matrix represent the most critical information stored within that matrix; it is logical to only mathematically examine the diagonal values to diagnose faults. The distance parameter approach does this by calculating an average of the diagonal entries to condense the cost matrix to a single value, indicative of the degree of similarity between the two attributed graphs encapsulated in the cost matrix. In the article by Greyling et al. [55], the equation used to determine this distance parameter D_C is given by

$$D_C = \sum_{k=1}^{n} \frac{C_{kk}}{n}, \tag{4.3}$$

where C_{kk} is the relevant diagonal entry in the cost matrix and n is the number of diagonal entries in the cost matrix.

To implement this approach to detect faults, operational attributed graph data is generated by taking measurement samples every 180 seconds over 25 hours, producing 501 operational attributed graphs since both the first and last samples are included. For the first hour of the 25 hours, the system is still under NOC, and then one of the 21 system conditions is induced for 24-hours. The first 20 operational attributed graphs of the 501 operational attributed graph set of a specific system condition, represent NOC and the remaining 481 operational attributed graphs represent the induced system condition. When operational matrices are compared with the reference normal attributed graph in the database, the distance parameters from the resulting cost matrices are expected to be relatively small if the system is under NOC. When these operational matrices are compared with the reference normal attributed graph in the database, the distance parameters from the resulting cost matrices are expected to be much more significant if a fault condition is present in the system.

If all 501 distance parameters were to be plotted on the same axis, the contrast in the size of the distance parameters produced by NOC and fault conditions is expected to be very apparent. However, this is the case only if the distance parameters are obtained from comparing the reference normal attributed graph with the operational graph of all the 21 possible system conditions.

The procedure for detecting and isolating a particular fault condition is as follows:

1. Generate the operational attributed graphs (G_{OP}) for each one of the 21 system conditions. Each condition will have 501 operational attributed graphs due to the sampling

interval, of which the first 20 graphs are still under NOC.

2. Compare an operational attributed graph of a specific system condition (G_{OP}) with each one of the 21-reference attributed graphs in the database by using the HEOM function as described in Figure 4.4. This will produce 21 cost matrices. These cost matrices are then recorded in one row of an array of cost matrices where each row represents a specific sample number, and each column represents a reference condition. An example of this can be seen in Figure 4.5.
3. Calculate all the distance parameters of the 21 cost matrices and insert these 21 parameters into a row of the distance array displayed in Figure 4.6. Each row in the distance array represents the sample number, and each column represents one of the 21 reference conditions. It should be noted that the array of cost matrices and distance array displayed in Figure 4.6 represent a single operational condition. This means that these arrays will have to be generated for each of the 21 conditions.
4. To isolate a fault, determine the column index of the minimum distance parameter of the specific row (sample number) in the distance array. This method isolates the faulty operational condition (anyone of the last 481 operational attributed graphs in the set of 501 operational attributed graphs) to the reference condition with the same column index as the minimum distance parameter in a specific row (sample number).
5. A fault is detected whenever this method does not isolate an operational fault condition to the column representing the reference NOC. Thus, a fault is detected when the column index of the minimum distance parameter is not 1 (See Figure 4.6).
6. Repeat steps 2 - 5 for each of the 501 sampled operational attributed graphs produced for each of the 21 system conditions.

As an example of how faults are detected and isolated, consider Figure 4.7 and Figure 4.8 below. Figure 4.7 contains the plotted results of the distance parameters that are produced by comparing all 501 of the sampled operational attributed graphs of Fault 2 (G_{FT2}) with all 21 of the reference attributed graphs. It is clear from Figure 4.7 that the distance parameters produced by comparing the operational attributed graphs of Fault 2 with the reference attributed graph of Fault 2 are, for the most part, smaller than any of the other distance parameters. The method, therefore, can detect and isolate Fault 2 for most of the 501 samples.

This is, however, not the case when the results plotted in Figure 4.8 are analysed. Most of the time, the distance parameters produced by comparing the operational attributed graphs of Fault 8 with the reference attributed graph of Fault 8 are not smaller than any of the other distance parameters. This FDI method, therefore, cannot uniquely isolate Fault 8. However, since the distance parameters produced by comparing the operational attributed graphs of Fault 8 with the reference normal attributed graph are, for the most part, not smaller than

the other distance parameters, the method can detect the presence of a fault in the system.

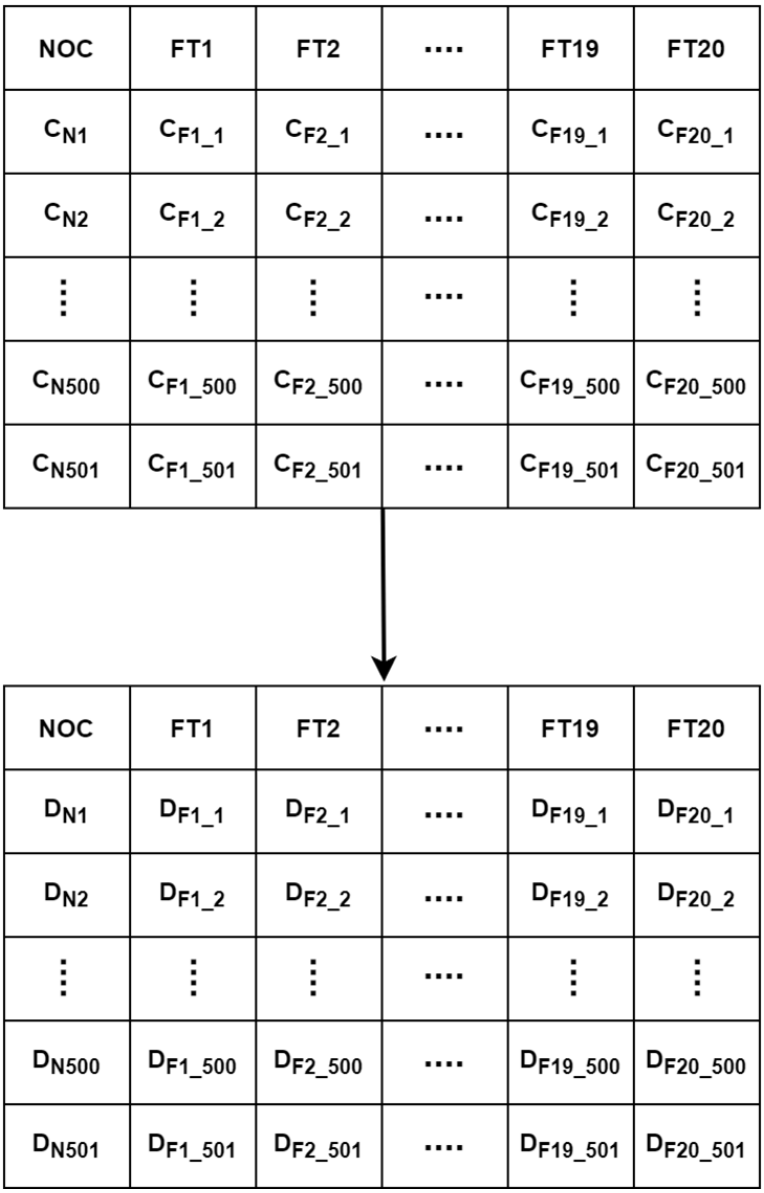


Figure 4.6: Illustration of how the distance array is produced from the array of cost matrices.

Table 4.3 contains the detection and isolation rates obtained from implementing the distance parameter FDI method. The top row contains the fault detection rates, while the rest contain the fault isolation rates. Each isolation rate entry represents the percentage of times the reference condition represented by that entry's row index was isolated to the operational condition represented by that entry's column index. The main column represents the diagonal entries of the isolation rates, representing the percentage of times an operational condition was correctly isolated to its reference condition.

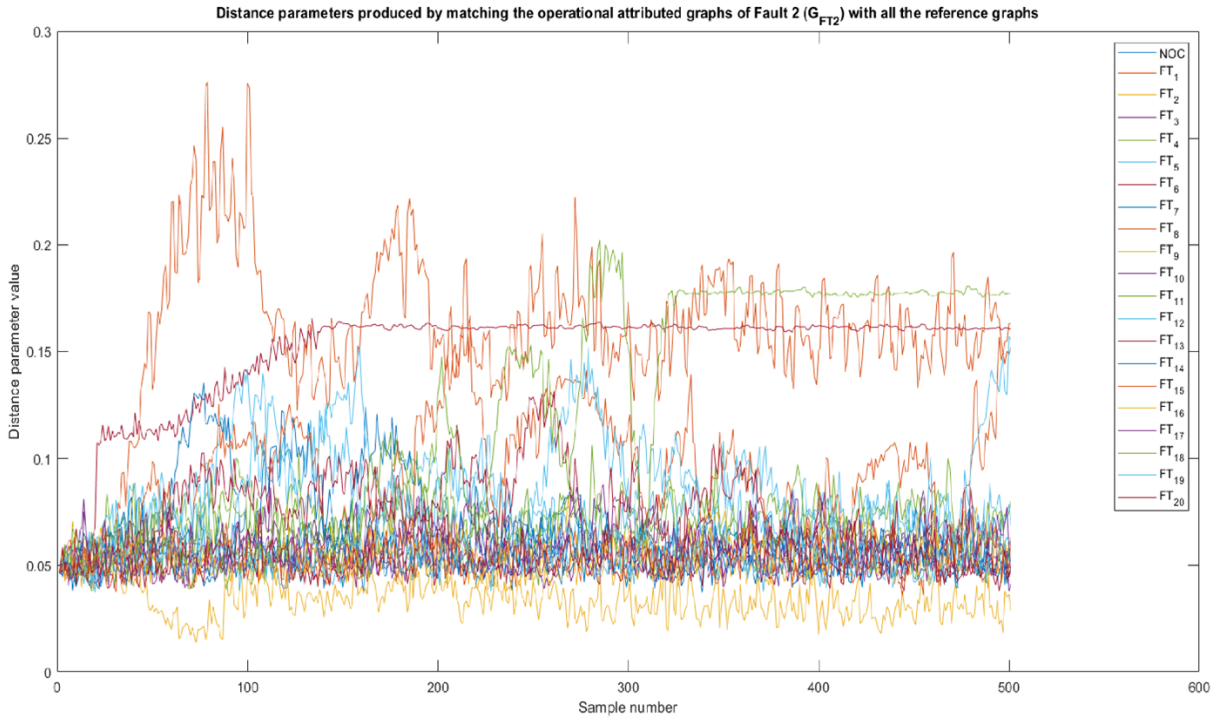


Figure 4.7: Plot of distance parameters produced by comparing the operational attributed graph of Fault 2 with all the reference graphs.

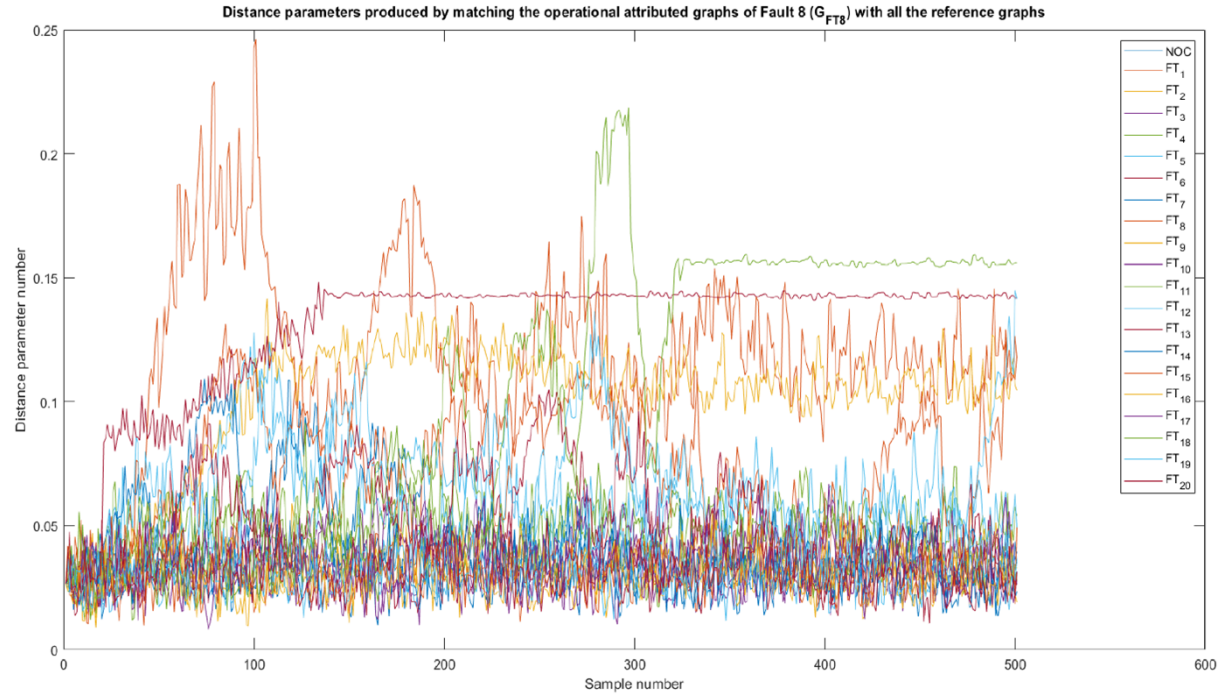


Figure 4.8: Plot of distance parameters produced by comparing the operational attributed graph of Fault 8 with all the reference graphs.

Table 4.3: Detection and isolation rates of the distance parameter FDI method.

	Main	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	F13	F14	F15	F16	F17	F18	F19	F20
DR (%)		98	99	86	95	91	100	92	89	86	87	87	90	92	84	84	84	86	96	84	85
IR (%) F1	75	75	0	2	0	0	0	3	8	1	2	0	1	0	1	1	1	1	1	0	1
IR (%) F2	85	0	85	1	0	0	0	1	0	1	0	0	0	0	1	3	2	1	0	1	1
IR (%) F3	15	0	0	15	0	0	0	3	0	11	1	1	0	0	4	12	7	7	2	10	10
IR (%) F4	33	0	0	4	33	0	0	2	0	3	1	11	0	0	9	4	3	19	0	2	4
IR (%) F5	21	0	0	8	0	21	0	2	0	9	1	1	0	1	5	11	8	6	2	8	8
IR (%) F6	59	0	0	0	0	5	59	7	9	0	0	0	11	3	0	0	0	0	6	0	0
IR (%) F7	8	0	0	9	0	1	0	8	1	10	4	1	0	5	10	10	9	5	2	7	9
IR (%) F8	1	0	0	11	0	0	0	5	1	10	5	1	1	2	5	9	9	7	2	10	11
IR (%) F9	10	0	0	14	0	0	0	4	1	10	2	2	0	0	7	11	7	5	1	9	12
IR (%) F10	2	0	0	10	0	0	0	6	0	11	2	2	1	1	7	12	8	3	4	11	9
IR (%) F11	5	0	0	8	0	0	0	3	0	6	1	5	0	0	13	10	6	18	1	6	8
IR (%) F12	8	0	0	9	0	0	0	7	0	10	2	3	8	2	8	10	7	5	2	10	6
IR (%) F13	5	0	0	6	0	0	0	8	0	12	2	1	1	5	8	9	12	5	2	8	10
IR (%) F14	7	0	0	11	0	0	0	4	0	10	1	1	0	0	7	14	7	4	1	10	11
IR (%) F15	14	0	0	9	0	0	0	6	1	14	2	1	0	1	3	14	8	4	3	11	8
IR (%) F16	10	0	0	8	0	0	0	4	0	12	2	2	0	1	5	11	10	5	4	9	10
IR (%) F17	13	0	0	9	0	0	0	4	0	10	1	4	1	0	10	11	6	13	1	6	10
IR (%) F18	8	0	0	4	0	12	0	12	8	5	1	2	10	15	4	4	5	5	8	1	2
IR (%) F19	9	0	0	12	0	0	0	5	1	10	1	2	0	1	7	13	7	4	2	9	11
IR (%) F20	10	0	0	9	0	0	0	6	1	12	2	1	0	1	4	14	8	3	3	12	10

4.2.2.1 Detection

The confusion matrix of the distance parameter FDI method can be seen in Table 4.4. All 501 sample attributed graphs of all the 21 system conditions (a total of 10521 samples) were assessed with the FDI method to produce this confusion matrix. The overall detection accuracy of the method is expressed as the percentage of times that the operational condition (the actual system condition) was a fault condition and the method successfully detected that a fault was present in the system. This is expressed as the true positive (TP) value divided by the sum of the true positive (TP) and false-negative (FN) values $\left(\frac{d}{d+b}\right)$. From the matrix, this method achieved an overall detection rate accuracy (true positive rate) of 89.81 % and a false negative rate of 10.19 %. This method is, therefore, quite proficient in detecting faults from the TEP data.

Table 4.4: Confusion matrix of distance parameter FDI method.

CONFUSION MATRIX			
		True condition	
		Fault-free	Fault
Detected condition	Fault-free	a TN 119	b FN 980
	Fault	c FP 782	d TP 8640

DETECTION RATES	
Rate	%
R_FN	10.19
R_TP	89.81
Accuracy	89.91

4.2.2.2 Isolation

To determine the overall isolation rate of this FDI method, the diagonal entries of the isolation rate rows of Table 4.3 are considered. Since each of these diagonal entries represents the rate at which a specific operational condition was correctly isolated to the condition's corresponding reference condition, the overall isolation rate of the method can be determined by calculating the average value of the diagonal entries. The overall isolation rate for the distance parameter FDI method applied to the TEP data is calculated as 19.90 %.

4.2.3 Eigendecomposition method

After examining the eigendecomposition FDI method as proposed in [53], it was found that the methodologies of the distance and eigendecomposition FDI methods produce very similar performance results. This is because the average of the diagonal entries of a cost matrix and the

average eigenvalue of said cost matrix are the same. To overcome this, the eigendecomposition FDI method was altered so that the cost matrices are generated by comparing the reference normal attributed graph with all the other reference and operational attributed graphs. Every cost matrix then acts as a vector of departure from the reference normal attributed graph, and the eigenvalues of these cost matrices can be used to detect and isolate faults.

It is clear from examining the fundamental principles of eigenvectors and eigenvalues [56], that the eigenvalues are inextricably linked to the diagonal entries of a matrix. This fact makes this approach, like the distance parameter approach, a valuable tool since the diagonal entries of that matrix represent the most pertinent information of a cost matrix.

The methodology behind this approach deviates from the methodology of the distance parameter method in that this approach requires two graph comparison operations to implement FDI, and it does not condense cost matrices to a single parameter. First, an operational attributed graph is generated by taking measurements every 180 seconds over 25 hours, producing 501 operational attributed graphs for each operational condition. Next, the system is again under NOC for the first hour, and then one of the 21 system conditions is induced for the remaining 24 hours. This results in the first 20 operational attributed graphs representing NOC while the remaining 481 operational attributed graphs all represent the induced system condition. Each operational attributed graph is then compared with all the reference attributed graphs in the database, and a cost matrix is generated with the HEOM function for each comparison operation.

The process for detecting and isolating faults is as follows:

1. Generate the operational attributed graphs (G_{OP}) for each one of the 21 system conditions. Each condition will have 501 operational attributed graphs due to the sampling interval, of which the first 20 graphs are still under NOC.
2. Compare all the reference graphs in the database with the reference normal attributed graph (G_{N_avg}) by using the HEOM function as seen in Figure 4.9. This produces 21 cost matrices. These cost matrices are then recorded in one row of an array of cost matrices (array A in Figure 4.9) where each column represents a system condition. Finally, the eigenvalues of each cost matrix in array A are calculated and stored as a vector in an array of eigenvalues, as can be seen in Figure 4.10.
3. Compare all 501 of the sampled operational attributed graphs of a specific system condition (G_{OP}) with the reference normal attributed graph (G_{N_avg}) in the database by using the HEOM function as seen in Figure 4.9. This will produce 501 cost matrices. These cost matrices are then recorded in one of the columns of an array of cost matrices (array B in Figure 4.9) where each row represents a specific sample number, and each

column represents a system condition. This implies that when the operational condition is under NOC, the cost matrix produced by comparing the operational attributed graph with the reference normal attributed graph will be assigned to the NOC column of array B as can be seen in Figure 4.11. The eigenvalues of each cost matrix in array B are calculated and stored as a vector in an array of eigenvalues, as can be seen in Figure 4.11.

4. All the vectors in the array of eigenvalues in Figure 4.10 (vectors in blue) are concatenated into a single reference matrix as can be seen in Figure 4.12 (a) and then the absolute value of each eigenvalue is determined before they are rearranged in descending order.
5. A single vector from the array of eigenvalues in Figure 4.11 is selected, the absolute value of each eigenvalue is determined before they are rearranged in descending order and subtracted from the reference matrix mentioned in the previous step $|\vec{\lambda}_{NRef} - \vec{\lambda}_{NO}|$. As an example, the eigenvalue set indicated as a red vector in (Figure 4.11), is displayed in Figure 4.12 (b). An example of the matrix produced by the subtraction is displayed in Table 4.5.
6. For each row in the matrix represented by Table 4.5, identify the column index which contains the minimum value of that row. Then, use a frequency counter vector with one row and the same number of columns as the matrix and increase the entry in the frequency vector, which corresponds to the column with the minimum value in each row.
7. A fault is isolated by identifying the column index in the frequency vector which has the highest count, as this index represents the reference condition to which the operational condition has been matched.
8. A fault is detected when the operational condition represents a fault condition (anyone of the last 481 operational attributed graphs in the set of 501 operational attributed graphs), and the method does not isolate that operational condition to the reference normal condition.
9. Repeat steps 5 – 8 for every one of the 21 $\times$ 501 vectors contained in the array of eigenvalues in Figure 4.11.

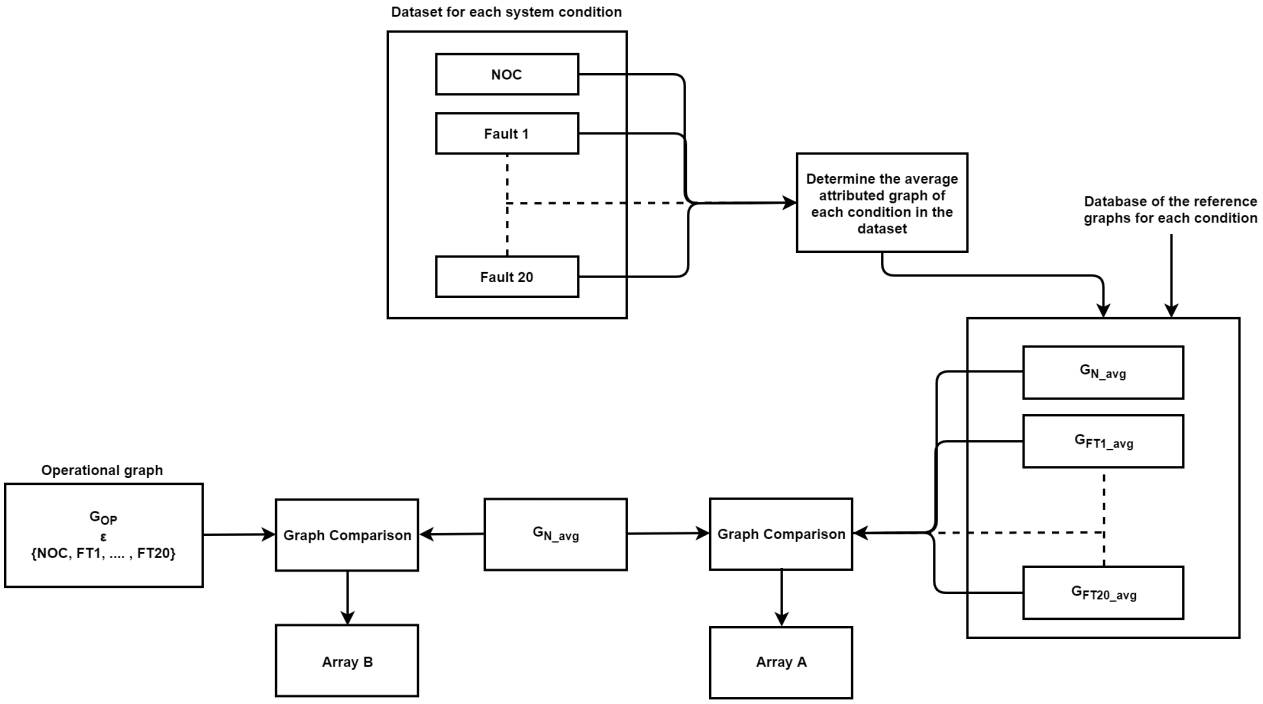


Figure 4.9: Illustration of the graph comparison operation for the eigendecomposition method.

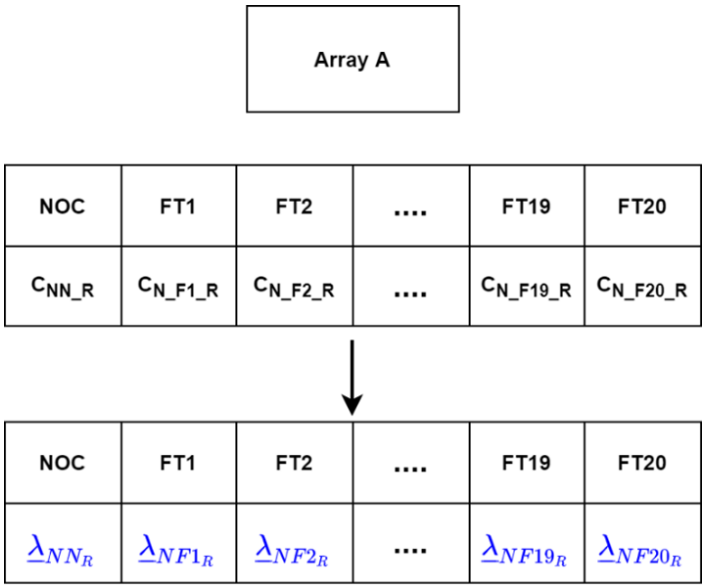


Figure 4.10: The array eigenvalues calculated from the cost matrices in array A.

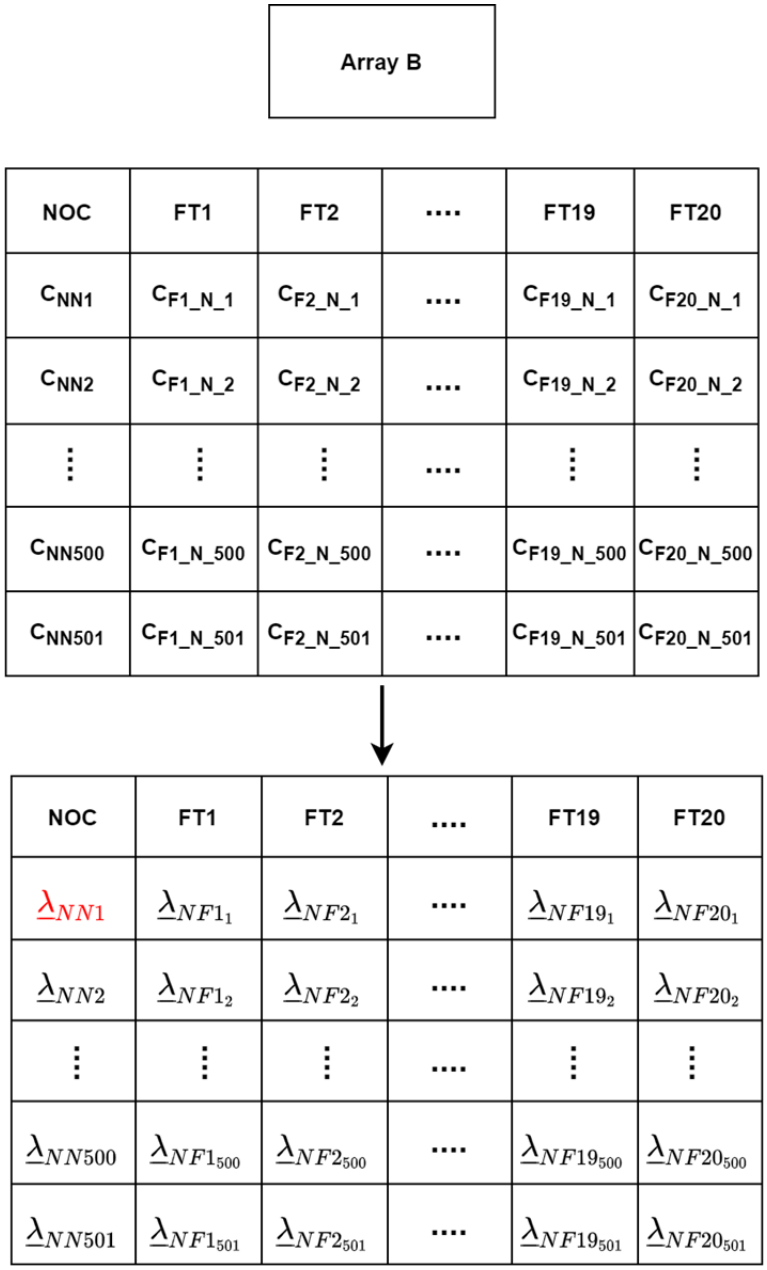


Figure 4.11: The array of eigenvalues calculated from the cost matrices in array B.

$\underline{\lambda}_{NN_R}$	$\underline{\lambda}_{NF1_R}$	$\underline{\lambda}_{NF2_R}$	....	$\underline{\lambda}_{NF19_R}$	$\underline{\lambda}_{NF20_R}$
18.9903	20.7931	19.8258	....	19.1183	18.9961
4.9579	5.4005	5.0894	....	4.9842	4.9014
2.7419	2.3587	2.6580	....	2.6960	2.6952
2.1810	2.1348	2.1260	....	2.1621	2.1682
1.9730	1.9537	1.9261	....	1.9614	1.9656
1.9362	1.9122	1.6896	....	1.9254	1.9243
1.7020	1.6915	1.7739	....	1.6988	1.6994
1.1182	1.1121	1.1144	....	1.1141	1.1144
0.8185	0.8089	0.8021	....	0.8152	0.8140
0.7329	0.7308	0.7005	....	0.7216	0.7304
0.4994	0.4878	0.4937	....	0.4929	0.4957
0.1585	0.1544	0.1559	....	0.1582	0.1584
0.1337	0.1290	0.1324	....	0.1333	0.1331
0.0279	0.0268	0.0270	....	0.0276	0.0277
0.0094	0.0091	0.0094	....	0.0094	0.0093

$\underline{\lambda}_{NN1}$
19.0353
4.9666
2.7261
2.1661
1.9652
1.9271
1.6976
1.1146
0.8145
0.7239
0.4944
0.1583
0.1334
0.0276
0.0094

Figure 4.12: (a) Matrix produced by concatenating the blue vectors from the array of eigenvalues produced by array A. (b) Eigenvalues contained in the red vector.

As an example, Fault 6 is induced as the operational condition. The set of eigenvalues produced by comparing the attributed graph of Fault 6 with the reference normal attributed graph, are subtracted from the blue vectors in Figure 4.12 (a). This subtraction results in the matrix seen in Table 4.5. Note that all the matrix entries are normalized and, thus, expressed as percentage values. The green entries represent the minimum value in each row, while the blue row at the bottom of the table represents the frequency vector. By investigating the frequency vector on the bottom row of the table, the column representing Fault 6 contained the most minimum value entries of all the rows. This means that this sampled operational condition has been isolated to Fault 6.

Table 4.6 contains the detection and isolation rates obtained from implementing the eigendecomposition FDI method. The top row contains the fault detection rates, while the rest of the rows contain the fault isolation rates. Each isolation rate entry represents the percentage of times the reference condition represented by that entry's row index was isolated to the operational condition represented by that entry's column index. The summary column represents the diagonal entries of the isolation rates, representing the percentage of times an operational condition was correctly isolated to its reference condition.

Table 4.5: Matrix expressing the percentage difference between each blue vector entry and its corresponding red vector entry.

NOC	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	F13	F14	F15	F16	F17	F18	F19	F20
1.56	11.20	6.03	1.80	2.70	2.65	0.17	1.22	2.45	2.29	1.23	2.55	0.67	1.12	2.18	1.60	1.38	2.24	0.78	2.24	1.59
12.50	22.54	15.48	12.25	13.21	11.95	2.69	8.40	13.46	12.83	10.90	13.36	10.12	9.98	13.05	11.30	11.72	12.60	5.74	13.10	11.22
23.95	6.62	20.15	23.11	23.49	22.04	4.95	16.60	18.55	22.84	21.32	22.21	19.63	18.97	22.20	21.79	22.60	23.46	8.34	21.87	21.84
11.34	8.99	8.54	9.44	10.35	10.50	5.59	10.09	10.60	9.46	10.76	10.65	9.71	9.56	11.15	10.74	9.22	10.85	7.11	10.38	10.69
12.64	11.54	9.96	11.78	11.67	11.82	4.69	11.68	9.93	11.53	11.53	12.17	8.41	8.94	12.23	12.10	11.60	11.86	7.52	11.98	12.21
16.67	15.23	6.89	15.20	13.74	12.27	2.68	15.13	13.03	14.79	14.83	14.62	4.98	9.29	15.89	16.04	15.51	15.08	6.51	16.02	15.96
24.09	23.33	23.19	24.16	23.29	24.31	12.69	23.65	23.01	23.84	23.19	23.39	20.60	22.04	23.30	23.69	22.89	23.59	20.93	23.86	23.91
10.61	10.00	10.23	10.07	10.16	9.67	4.23	9.46	9.00	10.28	10.01	10.01	8.68	7.94	10.37	9.99	9.79	10.41	3.00	10.20	10.24
8.92	7.64	6.73	7.99	8.16	7.10	2.98	6.72	7.69	8.21	8.00	7.93	6.31	6.65	8.54	8.38	8.09	8.11	0.59	8.48	8.33
4.63	4.32	0.01	1.33	3.20	3.82	1.97	3.95	4.31	1.39	4.37	3.75	3.68	3.65	4.48	4.01	1.29	4.25	2.59	3.02	4.27
15.14	12.46	13.82	11.21	12.35	8.53	4.89	13.73	11.45	10.91	12.86	14.19	3.51	10.50	13.95	14.35	13.65	14.22	9.70	13.65	14.30
1.84	0.76	0.19	1.62	1.69	1.85	0.68	1.87	1.69	1.51	1.83	1.68	1.76	1.19	1.82	1.89	1.50	1.74	1.90	1.69	1.78
3.60	0.05	2.63	3.25	3.37	3.19	1.57	3.11	3.01	3.38	3.24	3.04	3.10	1.90	3.11	3.04	3.01	3.44	1.50	3.32	3.12
15.11	10.61	11.37	11.95	13.77	13.04	4.96	12.63	12.97	13.75	13.57	12.64	12.20	12.26	13.77	13.36	12.58	14.43	9.10	13.87	14.11
88.90	82.78	88.35	88.27	88.06	87.08	30.61	81.54	79.73	88.69	85.29	88.54	75.60	68.01	88.56	87.25	87.80	88.30	60.33	88.55	87.15
0	**1**	**2**	**0**	**0**	**0**	**9**	**0**	**0**	**0**	**0**	**0**	**1**	**0**	**0**	**0**	**0**	**0**	**2**	**0**	**0**

Table 4.6: Detection and isolation rates of the eigendecomposition method.

	Main	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	F13	F14	F15	F16	F17	F18	F19	F20
DR (%)		100	100	99	100	100	100	100	100	99	100	100	100	100	98	99	99	100	100	99	99
IR (%) F1	84	84	3	0	0	0	6	0	0	0	0	0	0	4	0	0	0	0	1	0	0
IR (%) F2	77	6	77	0	1	1	2	0	1	0	0	0	3	3	0	1	1	0	3	0	0
IR (%) F3	9	7	15	9	7	10	0	4	6	6	11	2	1	3	2	2	10	1	2	1	0
IR (%) F4	16	4	20	4	16	15	0	2	7	2	5	1	3	10	0	1	9	0	0	0	0
IR (%) F5	19	6	30	2	6	19	7	5	4	0	2	1	4	7	0	1	3	0	1	0	0
IR (%) F6	98	0	1	0	0	0	98	0	0	0	0	0	0	1	0	0	0	0	0	0	0
IR (%) F7	3	11	23	4	4	6	8	3	5	2	6	1	4	10	0	1	4	0	6	0	0
IR (%) F8	2	13	19	1	1	2	38	1	2	1	1	0	6	7	0	0	1	0	8	0	0
IR (%) F9	4	5	16	7	10	10	1	3	4	4	10	2	1	8	0	3	11	2	0	0	0
IR (%) F10	4	25	27	4	4	8	0	2	7	1	4	2	2	6	0	0	5	0	1	0	0
IR (%) F11	1	4	22	6	6	9	12	3	5	1	7	1	4	9	0	1	7	1	1	0	0
IR (%) F12	8	9	25	1	2	4	25	1	3	0	1	0	8	12	0	0	1	0	7	0	0
IR (%) F13	14	7	21	1	3	4	19	3	4	1	2	1	6	14	0	0	1	0	12	0	0
IR (%) F14	1	4	21	8	8	11	0	3	6	5	6	2	4	6	1	1	10	1	0	0	0
IR (%) F15	2	4	16	14	8	8	0	4	6	4	9	2	2	5	1	2	10	1	0	1	1
IR (%) F16	10	4	21	9	9	7	0	4	6	2	5	3	4	10	0	2	10	1	0	1	0
IR (%) F17	1	6	20	7	13	8	0	3	6	3	9	1	2	6	1	1	11	1	0	0	0
IR (%) F18	3	9	11	3	1	3	51	2	2	1	3	0	3	4	0	1	3	0	3	0	0
IR (%) F19	0	2	18	10	7	9	0	2	4	4	11	2	3	8	1	2	10	2	0	0	0
IR (%) F20	0	3	17	9	11	7	1	3	7	5	10	3	2	5	1	2	11	2	0	0	0

4.2.3.1 Detection

The confusion matrix of the eigendecomposition FDI method can be seen in Table 4.7. All 501 sample attributed graphs of all the 21 system conditions (a total of 10521 samples) were accessed with the FDI method to produce this confusion matrix. The overall detection accuracy of the method is expressed as the percentage of times that the operational condition (the actual system condition) was a fault condition and the method successfully detected that a fault was present in the system. This is expressed as the true positive (TP) value divided by the sum of the true positive (TP) and false-negative (FN) values $\left(\frac{d}{d+b}\right)$. From the matrix, this method achieved an overall detection rate accuracy (true positive rate) of 99.60 % and a false negative rate of 0.40 %. This method is, therefore, extremely proficient in detecting faults from the TEP data.

Table 4.7: Confusion matrix of the eigendecomposition FDI method.

CONFUSION MATRIX			
		True condition	
		Fault-free	Fault
Detected condition	Fault-free	a TN 15	b FN 38
	Fault	c FP 886	d TP 9582

DETECTION RATES	
Rate	%
R_FN	0.40
R_TP	99.60
Accuracy	99.60

4.2.3.2 Isolation

To determine the overall isolation rate of this FDI method, the diagonal entries of the isolation rate rows of Table 4.6 are again considered. The overall isolation rate of the method can be determined by calculating the average value of the diagonal entries since each one of these diagonal entries represents the rate at which a specific operational condition was correctly isolated to the condition's corresponding reference condition. The overall isolation rate for the eigendecomposition FDI method applied to the TEP data is calculated as 17.80 %.

4.2.4 Modified eigendecomposition FDI method

A concurrent study conducted by Wolmarans [52] found that when cost matrices are generated, and the eigenvalues of those cost matrices are calculated to detect and isolate faults, the process noise is encapsulated in the later eigenvalue entries. By removing these later eigenvalue entries,

the effect process noise has on the performance of FDI methods, is minimized and the FDI method requires less information to diagnose faults.

This aligns with a process known as principal component analysis (PCA), whereby only components (variables) that contain the bulk of process information are used to diagnose faults. In this case, the principal components are the dominant eigenvalues. Wolmarans used a cumulative percentage variance (CPV) calculation to establish how many eigenvalues account for the bulk amount of process variance. Process variance refers to the changes in the cost matrix as more time goes by and more samples are taken [52].

The equation used to calculate the CPV of selected eigenvalues is given by [52]:

$$CPV(a) = \frac{\sum_{i=1}^{a} \lambda_{ref}(i)}{\sum \lambda_{ref}} \times 100, \tag{4.4}$$

with a representing the number of selected eigenvalues from the reference eigenvalue set. Wolmarans showed that of the five eigenvalues generated from the cost matrix, the three dominant eigenvalues have a CPV of 86.9 %, meaning the last two eigenvalues represent 13.1 % of process variance and are, therefore, regarded as process noise.

This result implies that the eigendecomposition FDI method can be modified to consider less information while maintaining a similar level of performance. A trade-off is required to determine the least amount of dominant eigenvalues that will maintain FDI performance. The most common CPV values used for PCA are 75, 80, 85, & 90 % [57] and this study will, therefore, evaluate FDI performance at these four values.

The trade-off between FDI performance and the number of dominant eigenvalues used can be seen in Table 4.8. Equation 4.4 is manipulated to determine the number of selected eigenvalues (dominant eigenvalues) for a given CPV value. When a CPV of 85 % is used, the overall detection and isolation rates are respectively 3.32 % and 0.25 % lower than the rates obtained when all eigenvalues were used. Even though the FDI performance experienced a slight drop from the original rates, when a CPV of 85 % is used, the eigendecomposition FDI method has to consider nine fewer eigenvalues which greatly reduces the information required to make a diagnosis and, ultimately, the complexity of implementing the FDI method.

The eigendecomposition FDI method is now modified to no longer consider all 15 eigenvalues but to only consider the eigenvalues representing 85 % of the CPV, which are the six largest eigenvalues in this case. The confusion matrix and the specific detection and isolation rates of the modified eigendecomposition FDI method can be found in Tables 4.9 & 4.10, respectively. For the remainder of this study, the modified eigendecomposition FDI method will be used to

replace the original method.

Table 4.8: The performance of the eigendecomposition FDI method for different CPV values.

CPV (%)	Number of dominant eigenvalues	Overall detection rate (%)	Overall isolation rate (%)
75	4	94.45	18.40
80	5	95.40	17.75
85	6	96.28	17.55
90	7	96.55	17.50

Table 4.9: Confusion matrix of the modified eigendecomposition FDI method.

CONFUSION MATRIX			
		True condition	
		Fault-free	Fault
Detected condition	Fault-free	a TN 60	b FN 358
	Fault	c FP 841	d TP 9262

DETECTION RATES	
Rate	%
R_FN	3.72
R_TP	96.28
Accuracy	96.28

4.2.5 Residual-based method

As mentioned in the literature study, the residual-based method was initially promoted as a means of implementing an FDI scheme in the study done by Neser [26] and compared to the distance parameter method and an eigendecomposition method in the study done by Smith [53]. This method directly compares the indices of the NSM of the reference normal attributed graph with the NSM of the operational attributed graph to diagnose faults. Unlike the previous methods, this graph comparison method does not require a cost matrix to diagnose faults.

Table 4.10: Detection and isolation rates of the modified eigendecomposition method.

	Main	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	F13	F14	F15	F16	F17	F18	F19	F20
DR (%)		100	99	95	98	98	100	97	98	93	94	98	99	98	92	93	94	95	99	91	95
IR (%) F1	90	90	3	0	0	0	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0
IR (%) F2	75	8	75	0	1	0	4	0	1	0	1	0	4	1	0	0	0	0	3	0	0
IR (%) F3	12	5	14	12	11	3	0	4	7	4	12	3	2	4	3	1	5	2	0	2	0
IR (%) F4	26	2	28	6	26	10	0	1	10	1	6	1	0	4	0	0	1	0	0	0	0
IR (%) F5	5	3	37	2	16	5	8	7	5	0	4	0	1	7	0	0	0	0	2	0	0
IR (%) F6	100	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0
IR (%) F7	7	5	26	5	6	4	10	7	5	3	7	2	2	4	1	0	1	0	5	1	0
IR (%) F8	3	7	25	1	2	1	38	4	3	1	2	0	2	2	0	0	0	0	9	0	0
IR (%) F9	2	4	12	12	11	5	0	6	5	2	15	2	1	5	1	1	6	2	0	1	0
IR (%) F10	8	3	25	6	9	7	2	5	10	2	8	2	3	6	0	0	4	1	0	0	0
IR (%) F11	0	4	30	5	11	3	8	4	5	2	11	0	4	4	1	1	3	1	1	0	0
IR (%) F12	2	3	38	4	2	1	24	4	6	0	2	0	2	5	0	0	1	0	6	0	0
IR (%) F13	7	3	28	3	5	2	12	9	5	1	5	1	4	7	0	1	1	0	12	0	0
IR (%) F14	2	6	18	10	10	6	0	4	6	2	9	3	2	5	2	1	6	2	0	1	0
IR (%) F15	1	5	14	16	8	3	0	7	4	3	14	3	2	4	2	1	4	1	0	1	0
IR (%) F16	5	6	20	10	10	4	0	6	7	3	11	2	2	4	1	2	5	1	0	1	0
IR (%) F17	1	9	20	7	16	5	0	4	7	2	9	3	2	2	2	1	4	1	0	1	0
IR (%) F18	4	5	15	4	4	1	51	4	2	0	5	1	1	1	0	0	1	0	4	0	0
IR (%) F19	1	5	17	14	8	5	0	2	3	3	15	2	2	3	1	1	4	3	0	1	0
IR (%) F20	0	5	16	10	12	4	0	6	5	4	13	4	2	3	2	1	5	3	0	1	0

The residual matrix generated by comparing the operational attributed graph of the system in NOC with the reference normal attributed graph should theoretically be zero. This is, however, not the case when a dynamic system is used since minor variations occur even when the system is under NOC. This should, theoretically, not adversely affect fault detection or isolation since a residual generated by comparing an operational attributed graph under NOC with the reference normal attributed graph will be relatively small compared to a residual generated by comparing an operational attributed graph of a fault condition with the reference normal attributed graph. It is expected that the residuals will have distinct patterns allowing for the isolation of fault conditions. The terms "residual" and "residual matrix" are used interchangeably.

The mathematical expression of the graph comparison operation responsible for generating the residual matrix with the size $n \times m$ is expressed as

$$G_{res}(i,j) = \frac{G_{ref}(i,j) - G_{op}(i,j)}{G_{ref}(i,j)}, \tag{4.5}$$

where $G_{ref}(i,j)$ represents an entry in the NSM of the reference normal attributed graph and $G_{op}(i,j)$ represents the corresponding entry in the NSM of the operational attributed graph which are both $n \times m$ matrices. This comparison operation then produces the following residual matrix G_{res} :

$$G_{res} = \left(\begin{array}{ccc|ccc} \frac{\Delta b_{ref}(1,1)-\Delta b_{op}(1,1)}{\Delta b_{ref}(1,1)} & \cdots & | & \frac{\Delta \dot{q}_{ref}(1,3)-\Delta \dot{q}_{op}(1,3)}{\Delta \dot{q}_{ref}(1,3)} & \cdots & \frac{\Delta \dot{q}_{ref}(1,m)-\Delta \dot{q}_{op}(1,m)}{\Delta \dot{q}_{ref}(1,m)} \\ \vdots & \vdots & | & \vdots & \ddots & \vdots \\ \frac{\Delta b_{ref}(n,1)-\Delta b_{op}(n,1)}{\Delta b_{ref}(n,1)} & \cdots & | & \frac{\Delta \dot{q}_{ref}(n,3)-\Delta \dot{q}_{op}(n,3)}{\Delta \dot{q}_{ref}(n,3)} & \cdots & \frac{\Delta \dot{q}_{ref}(n,m)-\Delta \dot{q}_{op}(n,m)}{\Delta \dot{q}_{ref}(n,m)} \end{array}\right) \tag{4.6}$$

It is important to note that the first two columns of the NSM are reserved for the exergy attributes, while the third column up to column m are all reserved for the energy attributes.

The process used to generate reference attributed graph data is similar to the process used to generate the reference attributed graph data for the eigendecomposition FDI method, which can be seen in the process flow diagram in Figure 4.9. The process flow of generating the reference attributed graph data for the residual-based FDI method can be seen in Figure 4.13.

Each operational condition will again produce 501 operational attributed graphs due to the 180-second sampling interval over the 25 hours. The system is again operating under NOC for the first hour, and then one of the 21 system conditions is induced for the remaining 24 hours. The first 20 operational attributed graphs represent NOC, while the remaining 481 operational attributed graphs represent the induced system condition. Note that, for the residual-based

FDI method, the reference attributed graph of each condition is also the average of the 501 operational attributed graphs of that condition. When an operational attributed graph under NOC (G_{OP_N}) is compared with the reference normal attributed graph (G_{N_avg}), which is the average normal attributed graph from the dataset, a residual matrix for NOC is produced (G_{res_N}).

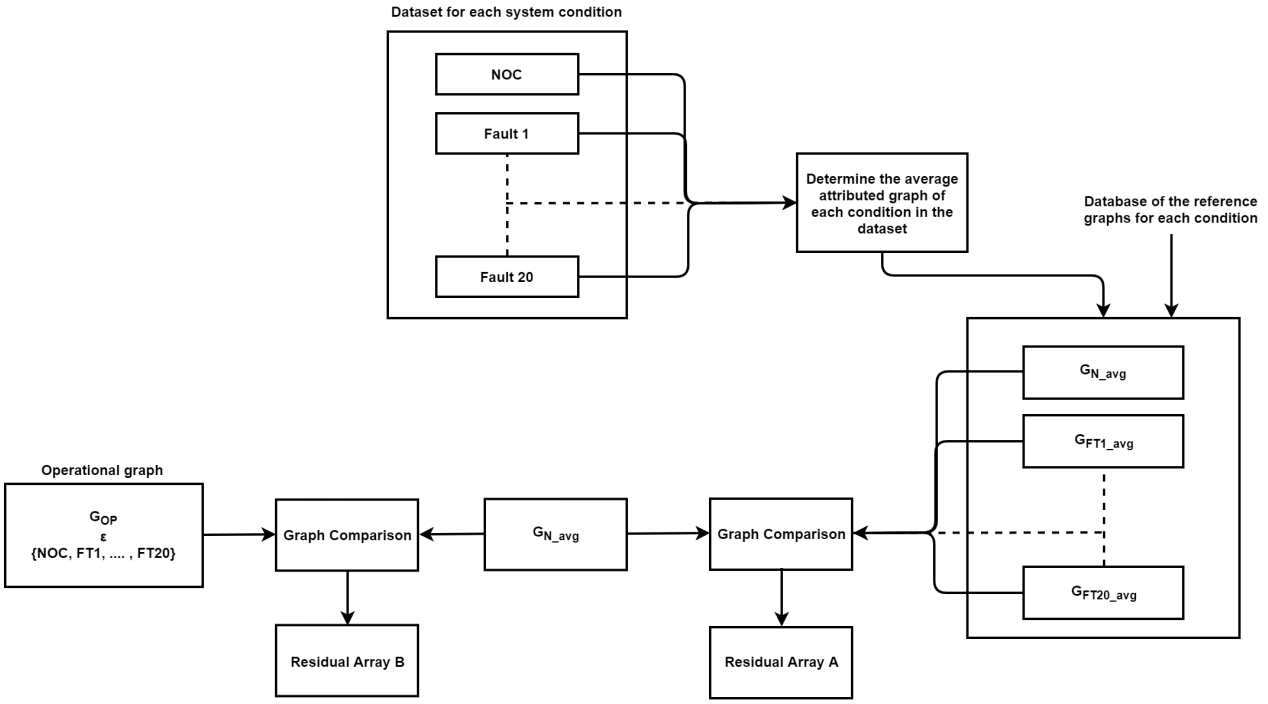


Figure 4.13: Process flow diagram of the graph comparison operation used by the residual-based FDI method.

In [26], a qualitative visual inspection approach is used to detect and isolate faults. A bar plot is composed for every node which contains the residual energy flow rate in and out of a node (R_{Qin} & R_{Qout}), the change in exergy over a node ($R_{\Delta b}$), and the energy flow rate between the node and the environmental node (R_{Qenv}). A pattern is encoded within this bar plot which can be analysed to detect if a fault is present or if the system is under NOC. Faults can also be isolated in this manner by only comparing the operational attributed graphs with the reference normal attributed graph to generate residuals which are in turn used to compose bar plots. Therefore, it is not necessary to compare the operational attributed graphs with every reference attributed graph in the database with this approach.

The attributed graph used in [26] only had 6 nodes which is a much smaller system than the one used in [53] which had 20 nodes. To address this, the approach used in [26] was slightly altered in [53]. A bar plot was generated where each node (column) receives one bar, and this single bar is a combination of the four variables of the node ($R_{Qin}, R_{Qout}, R_{\Delta b}$ & R_{Qenv}).

Because the qualitative interpretation of these bar plots proved to be a tedious and inefficient process, a quantitative method was instead pursued in [53]. This method counts the number of residual matrices that match a specific pattern of a specific system condition. This method, to its detriment, produces several unknown states, which complicate fault diagnosis. The existence of these unknown states will also render the control data ineffective when using it to compare and evaluate graph reduction techniques. To overcome this issue, the method proposed in [53] was altered by using frequency vectors in a similar fashion as with the eigendecomposition method, instead of simply counting the matching matrices. This ensures that no unknown states are produced. The procedure of this altered method is outlined below:

1. Generate the operational attributed graphs (G_{OP}) for each one of the 21 system conditions. Each condition will have 501 operational attributed graphs due to the sampling interval, of which the first 20 graphs are still under NOC.
2. Compare all the reference graphs in the database with the reference normal attributed graph (G_{N_avg}) by using (4.5). This produces 21 residual matrices. These residual matrices are then recorded in one of the rows of an array of residual matrices (array A in Figure 4.13 and Figure 4.14) where each column represents a system condition. For demonstrative purposes, a residual matrix is represented in the subsequent figures with the symbol R.
3. Compare all 501 of the sampled operational attributed graphs of all 21 system conditions (G_{OP}) with the reference normal attributed graph (G_{N_avg}) in the database by using (4.5). This will produce 21×501 residual matrices. These residual matrices are then recorded in the corresponding columns of an array of residual matrices (array B in Figure 4.13 and Figure 4.15) where each row represents a specific sample number, and each column represents a system condition. This implies that when the operational condition is under NOC, the residual matrix produced by comparing the operational attributed graph with the reference normal attributed graph will be assigned to the NOC column of array B as can be seen in Figure 4.15.
4. Calculate the average and standard deviation of the residual values of the change in exergy over each node in the system for every residual matrix. This is done by calculating the average and standard deviation of the residual values in columns one and two. ($R_{\Delta b\_avg}$ & $R_{\Delta b\_std}$)
5. Calculate the average and standard deviation of the residual values of the energy flow rates between nodes for every residual matrix. This is done by calculating the average and standard deviation of the residual values in column three until the residual matrix's final column. ($R_{\Delta Q\_avg}$ & $R_{\Delta Q\_std}$)
6. Calculate $R_{\Delta b\_avg} - 2R_{\Delta b\_std}$ as well as $R_{\Delta b\_avg} + 2R_{\Delta b\_std}$. Now, do the same for the

energy flow rates: $R_{\Delta Q\_avg} - 2R_{\Delta Q\_std}$ as well as $R_{\Delta Q\_avg} + 2R_{\Delta Q\_std}$.

7. From each residual matrix, develop a binary residual signature, which contains only binary values, by using two cases. Case 1: If any value in the first two columns is less than $R_{\Delta b\_avg} - 2R_{\Delta b\_std}$ or more than $R_{\Delta b\_avg} + 2R_{\Delta b\_std}$, that value is replaced with a 1. If any value in the remaining columns is less than $R_{\Delta Q\_avg} - 2R_{\Delta Q\_std}$ or more than $R_{\Delta Q\_avg} + 2R_{\Delta Q\_std}$ that value is replaced with a 1. Case 2: If any value in the first two columns is more than $R_{\Delta b\_avg} - 2R_{\Delta b\_std}$ or less than $R_{\Delta b\_avg} + 2R_{\Delta b\_std}$, that value is replaced with a 0. If any value in the remaining columns is more than $R_{\Delta Q\_avg} - 2R_{\Delta Q\_std}$ or less than $R_{\Delta Q\_avg} + 2R_{\Delta Q\_std}$ that value is replaced with a 0. An example of a binary residual signature can be seen in Table 4.11.
8. Repeat the previous step to transform all residual matrices into binary residual signatures in array B. This will produce 21×501 binary residual signatures in total. Next, do the same for the 21 reference residual matrices recorded in array A.
9. The number of "1's" in each column of every binary residual signature from array B (501 signatures in total) are counted and stored in its own operational frequency vector. This process is illustrated in Figure 4.15 where the red vector is a frequency vector that corresponds to a residual matrix in array B. The green row in Table 4.11 is an example of an operational frequency vector. Each entry of this vector represents the count of one column in the signature. Thus, a total of 501 vectors are generated for each one of the 21 system conditions.
10. Repeat the previous step for the 21 residual matrices contained in array A. This will produce 21 reference frequency vectors. This process is illustrated in Figure 4.14 where each blue vector is a reference frequency vector that corresponds to a residual matrix in array A.
11. The 21 reference frequency vectors (vectors in blue) are concatenated into a single matrix where each column represents a single reference frequency vector. This can be seen in Figure 4.16 (a). An operational frequency vector (vector in red) can be seen in Figure 4.16 (b).
12. The operational frequency vector (vector in red) is then subtracted from each of the 21 columns containing the concatenated reference frequency vectors (vectors in blue) $|\vec{fc}_{NRef} - \vec{fc}_{NO}|$. The total of each column is then calculated, and the column index, which has the minimum total value, is identified. This column index represents the reference condition that the operational condition has been isolated to.
13. Repeat the previous step for all the operational frequency vectors produced from the residual matrices in array B.
14. A fault is isolated by identifying which column index has the minimum total value.
15. A fault is detected when an operational fault condition (anyone of the last 481 opera-

tional attributed graphs in the set of 501 operational attributed graphs) is not isolated to the normal reference condition.

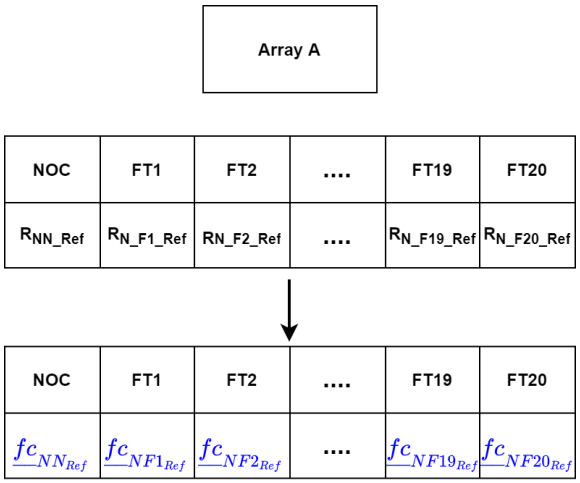


Figure 4.14: The array of frequency vectors resulting from the residual matrices in array A.

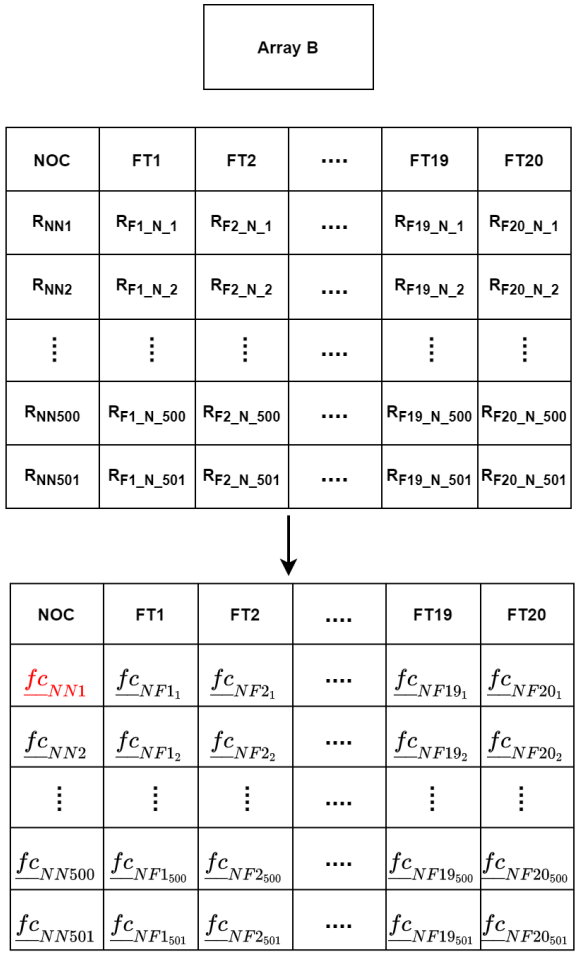


Figure 4.15: The array of frequency vectors resulting from the residual matrices in array B.

$\underline{fc}_{NN_{Ref}}$	$\underline{fc}_{NF1_{Ref}}$	$\underline{fc}_{NF2_{Ref}}$	....	$\underline{fc}_{NF19_{Ref}}$	$\underline{fc}_{NF20_{Ref}}$
0	1	2	....	1	1
0	1	1	....	1	1
0	1	1	....	1	1
0	1	1	....	1	1
0	1	1	....	1	1
0	1	1	....	1	1
0	1	1	....	1	1
0	2	1	....	2	3
0	1	1	....	1	1
0	2	1	....	1	1
0	1	1	....	1	1
0	2	2	....	2	1
0	1	1	....	1	1
0	2	2	....	2	2
0	1	1	....	2	2
0	1	1	....	1	1
0	1	1	....	1	1

$\underline{fc}_{NN1}$
1
2
1
1
1
1
1
1
2
1
3
1
1
1
2
2
1

Figure 4.16: (a) Matrix produced by concatenating the blue reference frequency vectors from array A. (b) Operational frequency vector in red.

The binary residual signature is displayed in Table 4.11 is derived from a residual matrix generated from an operational attributed graph while the operational condition was Fault 6.

Table 4.13 contains the detection and isolation rates obtained from implementing the residual-based FDI method. The top row contains the fault detection rates, while the rest of the rows contain the fault isolation rates. Each isolation rate entry represents the percentage of times the reference condition represented by that entry's row index was isolated to the operational condition represented by that entry's column index. The summary column represents the diagonal entries of the isolation rates, representing the percentage of times an operational condition was correctly isolated to its reference condition.

Table 4.11: Binary residual signature with the operational frequency vector in the green row.

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	1	1	0	0	1	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0	1	0	0	0	1	0	0
1	1	1	1	1	1	1	2	2	1	1	2	1	2	1	1	1

4.2.5.1 Detection

The confusion matrix of the residual-based FDI method can be seen in Table 4.12. All 501 sample attributed graphs of all the 21 system conditions (a total of 10521 samples) were accessed with the FDI method to produce this confusion matrix. The overall detection accuracy of the method is expressed as the percentage of times that the operational condition (the actual system condition) was a fault condition and the method successfully detected that a fault was present in the system. This is expressed as the true positive (TP) value divided by the sum of the true positive (TP) and false-negative (FN) values $\left(\frac{d}{d+b}\right)$. From the matrix, this method achieved an overall detection rate accuracy (true positive rate) of 98.86 % and a false negative rate of 1.14 %. This method is, therefore, excellent at detecting faults from the TEP data.

4.2.5.2 Isolation

To determine the overall isolation rate of this FDI method, the diagonal entries of the isolation rate rows of Table 4.13 are again considered. Each diagonal entry represents the rate at which a specific operational condition was correctly isolated to the condition's corresponding reference condition, so the overall isolation rate of the method can be determined by calculating the

average value of the diagonal entries. The overall isolation rate for the residual-based FDI method applied to the TEP data is calculated as 20.15 %.

Table 4.12: Confusion matrix of the residual-based FDI method.

CONFUSION MATRIX			
		True condition	
		Fault-free	Fault
Detected condition	Fault-free	a TN 3	b FN 110
	Fault	c FP 898	d TP 9510

DETECTION RATES	
Rate	%
R_FN	1.14
R_TP	98.86
Accuracy	98.86

4.3 Summary of results

A summary of the detection and isolation performance of the three methods can be seen in Table 4.14 below. The table contains the overall detection and isolation rates of each FDI method and the specific isolation rates of the fault conditions that each method could successfully isolate for the majority of samples. Specific detection rates of fault conditions were omitted from this table, seeing as each FDI method achieved very high detection rates for all the fault conditions.

When considering these factors, it is clear that no one of the three FDI methods can be seen as the outright best-performer when applied to the TEP data. For example, the distance parameter FDI method achieved the lowest overall detection rate and the weakest performance in terms of specific isolation rates. On the other hand, the eigendecomposition FDI method obtained the lowest overall isolation rate while also being the top performer in specific isolation rates.

The residual-based FDI method achieved the highest overall isolation and detection rates. While the specific isolation rates obtained by the residual-based FDI method are better than those obtained by the distance parameter FDI method, they are considerably weaker than the rates obtained by the eigendecomposition FDI method. An analysis of the specific fault isolation rates of each FDI method will allow for a holistic comparison between the performance of the FDI methods which use unreduced attributed graph data and the performance of the FDI methods which use reduced attributed graph data. This will, in turn, lead to reduction techniques being better equipped at reducing graph complexity while maintaining the same level of performance of the FDI methods.

If a choice had to be made, the eigendecomposition method would be selected since its overall detection and isolation rates are slightly lower than that of the residual-based FDI method, and it is the top performer in terms of specific isolation rates. What's more, it achieved a perfect specific isolation rate of Fault 6. Fault 6 involves a plant shutdown and has a very distinct effect on the process data, which should result in any effective FDI method achieving high detection and isolation rates. These results (Table 4.14) now serve as a set of control data with which the results of the FDI schemes, which use the reduced attributed graphs, can be compared. The algorithm of each FDI method, in the form of pseudocode, can be seen in Appendix A. These algorithms detail how the graph data is transformed into the final data structure used for FDI analysis. The links to the attributed graph data and all MATLAB® code used in this study, including the code of all three the FDI methods used in this chapter, can be found in Appendix B.

4.4 Conclusion

This chapter provides an elaborate explanation of how each of the three FDI methods selected for this study is implemented. Each FDI method is then applied to the attributed graph data of the TEP, and the performance of each method is measured in terms of the method's ability to detect and isolate fault conditions. An overview of the performance capabilities of the three methods can be found in Table 4.14.

It is clear, from analysing these results, that when these particular FDI methods are applied to the TEP graph data, the isolation capabilities of the FDI methods are very poor. It should, however, be noted that the isolation capabilities of graph-based methods are much weaker than the detection capabilities. This discrepancy in capabilities becomes even greater when the dataset to be considered by the FDI method is much larger. This becomes evident when comparing the results found in [8] with those found in [53].

There is no clear indication of which FDI method performed the best when applied to the graph data of the TEP since all the methods performed relatively well in at least one aspect. However, using all three FDI methods to evaluate graph reduction techniques makes it possible to make more holistic decisions about those reduction techniques.

Table 4.13: Detection and isolation rates of the residual-based FDI method.

	Main	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	F13	F14	F15	F16	F17	F18	F19	F20
DR (%)		100	100	100	98	93	100	99	99	100	100	97	100	93	100	100	100	100	100	100	100
IR (%) F1	80	80	0	0	0	1	0	1	17	0	0	0	0	0	0	0	0	0	0	0	0
IR (%) F2	61	0	61	0	0	1	18	0	0	0	0	0	1	0	16	0	0	0	0	1	0
IR (%) F3	17	13	1	17	1	5	6	2	2	9	11	0	15	0	0	5	5	0	0	1	8
IR (%) F4	23	1	2	0	23	15	2	2	1	0	1	25	6	0	0	0	0	8	0	0	11
IR (%) F5	29	14	4	0	5	29	2	10	2	0	9	0	13	2	2	0	0	0	0	0	0
IR (%) F6	88	11	0	0	0	0	88	0	0	0	0	0	0	0	1	0	0	0	0	0	0
IR (%) F7	9	12	3	2	2	5	11	9	16	4	5	0	10	0	6	3	2	0	0	2	6
IR (%) F8	19	26	2	0	0	1	14	13	19	0	1	0	2	0	16	1	0	0	0	2	1
IR (%) F9	10	11	2	10	1	9	7	2	4	10	12	0	11	1	2	4	5	0	0	1	8
IR (%) F10	8	4	2	4	4	16	5	9	11	4	8	0	5	1	1	2	3	0	0	0	22
IR (%) F11	22	6	2	1	19	11	2	1	2	1	7	22	6	0	1	1	1	5	0	1	8
IR (%) F12	13	13	3	1	4	11	14	10	10	3	4	0	13	4	8	1	0	0	0	0	0
IR (%) F13	1	1	8	1	1	4	16	11	33	1	1	0	7	1	3	1	1	0	0	0	2
IR (%) F14	1	7	2	6	8	11	3	2	3	6	14	3	13	1	1	3	3	3	0	0	11
IR (%) F15	5	12	1	4	3	8	4	3	1	8	16	0	14	2	2	5	4	0	0	2	10
IR (%) F16	3	6	4	3	2	11	7	4	9	6	13	0	11	1	2	4	3	0	0	0	13
IR (%) F17	4	10	1	3	10	10	3	2	3	4	12	6	11	1	1	4	4	4	0	1	9
IR (%) F18	0	10	1	1	1	7	47	4	5	2	4	0	4	1	6	2	1	0	0	1	3
IR (%) F19	1	11	2	3	3	10	4	3	3	9	19	0	13	1	2	5	3	0	0	1	8
IR (%) F20	9	13	2	3	2	7	5	4	4	8	17	0	13	1	2	4	4	0	0	2	9

Table 4.14: Summary of the detection and isolation performance of the three FDI schemes applied to the TEP.

Distance parameter FDI method	
Overall detection rate (%)	89.81
Overall isolation rate (%)	19.90
Isolation of Fault 1 (%)	75.00
Isolation of Fault 2 (%)	85.00
Isolation of Fault 6 (%)	59.00
Eigendecomposition FDI method	
Overall detection rate (%)	96.28
Overall isolation rate (%)	17.55
Isolation of Fault 1 (%)	90.00
Isolation of Fault 2 (%)	75.00
Isolation of Fault 6 (%)	100.00
Residual-based FDI method	
Overall detection rate (%)	98.86
Overall isolation rate (%)	20.15
Isolation of Fault 1 (%)	80.00
Isolation of Fault 2 (%)	61.00
Isolation of Fault 6 (%)	88.00

Chapter 5

Graph reduction techniques and experimental design

5.1 Introduction

This chapter outlines the graph reduction techniques that have been proposed as potentially viable options for reducing the size of the attributed graph data used in graph-based FDI methods. Then, the experimental design section provides an overview of the experimental process used to determine each proposed graph reduction technique's effectiveness. Guidelines on how the results of the experimental process should be interpreted are also discussed.

5.2 Proposed graph reduction techniques

The techniques proposed in this chapter were formulated by considering existing techniques in literature and how the FDI techniques detect and isolate faults. The five suggested techniques rely on three principal reduction approaches to reduce the attributed graph data. The first principal approach analyses the degree to which attributes vary from their values under NOC when fault conditions occur. This approach is based on the premise that attributes that experience minimal variation when fault conditions are induced will have a negligible effect on the cost matrices and/or residual matrices used in the FDI methods and can be removed from the graph data.

This first principal approach considers only the reference attributed graphs constructed by calculating the average of the time-series graph data to determine the degree of attribute

variation from NOC. Since the time-series data is averaged and fault conditions do not affect the noise experienced by graph attributes, process noise will not significantly influence the variance. The noise present in the system will, therefore, not result in this approach removing attributes.

The second principal approach analyses the size of the graph attributes while the process is under NOC. This approach is based on the premise that more minor attributes will have a much smaller effect on the cost and residual matrices than more considerable attributes and can, therefore, be removed.

The final principal approach summarizes nodes that are similar in attribute size. This approach is based on the premise that all the attributes remain in the graph data, but the structural layout of the graph changes. This means that all the attributes still contribute to the cost and residual matrices, causing negligible disruption to the FDI process while the structural size of these matrices is reduced. This final approach is largely based on the graph summarization techniques identified in Chapter 2 [27] - [35], but allows for the summarization to be restricted to prevent vital structural information from being completely distorted.

Both the first and second main approaches consider node and link attributes separately and, thus, result in two separate techniques for each approach. All three of these premises are theoretical suggestions, and that is why an experimental process is required to determine if these reduction techniques are valid, to which extent they work, and which techniques complement which of the FDI methods.

Link attribute reduction techniques and node attribute reduction techniques reduce the complexity of the attributed graph in different ways. Link attribute reduction techniques reduce complexity by setting specific attribute values in the NSM to zero, which means that the mathematical operations of the FDI methods evaluate fewer attributes. On the other hand, node attribute reduction techniques reduce complexity by removing rows and columns from the NSM and, thus, reduce the size of the NSM, to which the mathematical operations of the FDI methods are then applied. It is, therefore, also necessary to observe the effect that these different styles of complexity reduction have on the execution time of the mathematical operations of the FDI methods.

All techniques are prohibited from completely removing the environmental node to retain some of the most vital structural information. It is, however, permitted for the summarization technique to merge the environmental node with one other node. A summary of all the proposed reduction techniques, as well as a brief description of each technique, can be seen in Table 5.1.

Table 5.1: Summary of the graph reduction techniques proposed in this study.

Technique Number	Name	Reduction Type	Description
Technique 1	Node attribute filtering with variation analysis	Node attribute reduction	Removes node attributes based on their average variation from NOC over all fault conditions.
Technique 2	Link attribute filtering with variation analysis	Link attribute reduction	Removes link attributes based on their average variation from NOC over all fault conditions.
Technique 3	Link attribute filtering with a size threshold	Link attribute reduction	Removes link attributes based on the size of the attributes in NOC.
Technique 4	Node attribute filtering with a size threshold	Node attribute reduction	Removes node attributes based on the size of the attributes in NOC.
Technique 5	Summarization of similar nodes	Node attribute reduction	Merges similar nodes into a single node by summarizing their attributes.

5.3 Experimental design

5.3.1 Overview

The experimental process first has to determine if any of these techniques can reduce the attributed graph data of the TEP while maintaining a similar level of FDI performance achieved before graph reduction, for any of the three FDI methods. Then, the experimental process has to determine the efficacy of each reduction technique with regards to all three FDI methods by varying the extent to which the techniques reduce graph data in intervals, and measuring the resulting FDI performance indicators.

For each graph reduction technique, reduction intervals are selected. These reduction intervals determine to which extent the reduction technique reduces graph data. For example, the intervals for Technique 1 - 4 are the percentile threshold values these techniques use as input, and the number of mergers for Technique 5. The percentile threshold values are increased in increments of ten (10th percentile, 20th percentile, ... , 90th percentile), and the number of mergers is increased with an increment of one. These intervals only go so far as each specific technique allows.

The process starts with Technique 1. The lowest reduction interval (10th percentile threshold) is selected, and the technique accordingly reduces the TEP attributed graph data. The percentage of non-zero attributes reduced is calculated. All three FDI methods are then applied to the reduced attributed graph data. The method's overall detection rate, overall isolation rate, and three specific isolation rates are measured for each FDI method. These performance indicators are chosen to ensure the evaluation is holistic. Three specific isolation rates are again considered so as to facilitate a realistic comparison with the three specific isolation rates found in the control data. The reduction interval is now increased to the subsequent reduction interval (20th percentile threshold). All three FDI methods are again applied to the reduced graph data, and all the performance indicators are measured. This is repeated until the final reduction interval (90th percentile threshold) is reached.

Table 5.2 contains every indicator that is measured at one reduction interval for Technique 1. All these indicators have to be measured for all the reduction intervals of a reduction technique. This process is repeated for Techniques 2 - 5. It should be noted that Technique 5 uses the number of mergers as reduction intervals and not percentile threshold values.

Table 5.2: Overview of the experimental process applied to each reduction technique.

Technique #	FDI Method	Reduction interval	Measured performance indicator
Technique 1	Distance	10th percentile	Overall detection rate (%)
Technique 1	Distance	10th percentile	Overall isolation rate (%)
Technique 1	Distance	10th percentile	Non-zero attributes reduced (%)
Technique 1	Distance	10th percentile	Specific isolation rate 1 (%)
Technique 1	Distance	10th percentile	Specific isolation rate 2 (%)
Technique 1	Distance	10th percentile	Specific isolation rate 3 (%)
Technique 1	Eigen	10th percentile	Overall detection rate (%)
Technique 1	Eigen	10th percentile	Overall isolation rate (%)
Technique 1	Eigen	10th percentile	Non-zero attributes reduced (%)
Technique 1	Eigen	10th percentile	Specific isolation rate 1 (%)
Technique 1	Eigen	10th percentile	Specific isolation rate 2 (%)
Technique 1	Eigen	10th percentile	Specific isolation rate 3 (%)
Technique 1	Residual	10th percentile	Overall detection rate (%)
Technique 1	Residual	10th percentile	Overall isolation rate (%)
Technique 1	Residual	10th percentile	Non-zero attributes reduced (%)
Technique 1	Residual	10th percentile	Specific isolation rate 1 (%)
Technique 1	Residual	10th percentile	Specific isolation rate 2 (%)
Technique 1	Residual	10th percentile	Specific isolation rate 3 (%)

5.4 Analysis of the results

Once the experimental process has been applied to all five reduction techniques, the results obtained from this process must be evaluated to determine the viability and efficacy of the techniques. For each reduction technique, all the performance indicators discussed in Table 5.2 should be recorded in a table for all the reduction intervals. The performance indicators from the control data should also be included in this table.

For each reduction technique, two plots must also be generated. The first plot should contain the overall detection rate of each FDI method at every reduction interval and the overall detection rates of the three FDI methods contained in the control data. The second plot should contain the overall isolation rate of each FDI method at every reduction interval and the overall isolation rates of the three FDI methods contained in the control data. No plot will be made of specific isolation rates, seeing as they are recorded in the result tables, and the overall detection and isolation rates indicators carry more weight when evaluating FDI performance.

By using the tables and plots, it is possible to determine, for each technique, how the FDI performance obtained when using original graph data (control data) changes as the reduction

interval increases. It is then possible to determine the trajectory of this change in performance, the reduction intervals at which the reduction techniques work well, and which reduction techniques work better than others. This analysis will detail whether or not graph reduction is a viable solution and how effective the different techniques are.

5.5 Conclusion

This chapter introduces all five of the proposed graph reduction techniques used in this study. The premise on which each of the reduction techniques is based is also discussed. An outline of the experimental process used to evaluate the viability and efficacy of graph reduction is then provided. The final section of this chapter describes how the results from the experimental process should be analysed.

Chapter 6

Results

6.1 Introduction

In this chapter, the experimental process as outlined in Chapter 5 is implemented to evaluate the proposed graph reduction techniques. The results from the experimental process will be analysed as discussed in the previous chapter.

The detection and isolation capabilities of the FDI methods which use the reduced graph data will be compared to the capabilities of the methods in the control data recorded in Chapter 4. By comparing these capabilities, it will be possible to determine whether attributed graph data can be reduced by removing non-vital structural information while maintaining the level of performance achieved by the FDI methods when using unreduced graph data. Furthermore, it will also be possible to determine the extent to which each reduction technique can reduce the graph data before the performance of the FDI methods starts to deteriorate too extensively.

The individual reduction techniques are also applied to the attributed graph data in different combinations to determine if it is possible to overcome the shortcomings of the individual techniques. The implementation of each of the five reduction techniques is also verified by using Excel® to reduce the graph data and comparing the result with that produced by the MATLAB® code.

6.2 Evaluating the graph reduction techniques

This section provides, in numerical order, the methodology of each graph reduction technique as well as the experimental process used to evaluate that technique. The results from this

evaluation are also discussed.

6.2.1 Node attribute filtering with variation analysis

This technique uses the nodes of the attributed graph of the system under NOC as a reference and analyses the variation of each node for all the 20 process faults. The methodology used for evaluating this technique is as follows:

- Generate the attributed graph data for all 20 fault conditions as well as the normal condition. Then, for each of the 15 nodes, express how much the chemical exergy attribute value varies relative to the node's attribute value while the system is under NOC, as a percentage value. This produces a 20×15 matrix where each entry represents the percentage with which a specific node's chemical exergy value for a given fault condition has varied relative to the attribute value of the node in NOC.
- For each of the 15 nodes, express how much the physical exergy attribute value varies relative to the node's attribute value while the system is under NOC, as a percentage value. This also produces a 20×15 matrix where each entry represents the percentage with which a specific node's physical exergy value for a given fault condition has varied relative to the attribute value of the node in NOC.
- Calculate the mean of each column of both the chemical exergy variation and physical exergy variation matrices. Summate the mean column value of the chemical exergy variation matrix with its corresponding value in the physical exergy variation matrix. This results in a summated variation vector containing 15 values. Each value represents the average physical and chemical exergy variation of a node over the 20 process faults relative to the physical and chemical exergy values in the NOC.
- Use the 'isoutlier(A,'percentiles',threshold)' function in MATLAB® to identify the outliers found to be less than the 10th percentile threshold of the variation vector. These outliers represent the nodes whose attribute values vary very little for all 20 process faults relative to the NOC and are highly unlikely to contribute to the detection and isolation of faults.
- Remove the nodes identified as lower outliers in the vector from the attributed graph data of the TEP.
- Use the reduced graph data and apply the three FDI methods used in Chapter 4 to determine the detection and isolation capabilities of all three FDI methods.
- Increase the percentile threshold in increments of 10 until the 90th percentile has been reached. After each increase, repeat the previous three steps.

Figures 6.1 & 6.2 show the respective overall detection and isolation rates, relative to the percentage of attributes removed from the graph data. From the overall detection rates illustrated in Figure 6.1, it can be observed that the distance FDI method's detection capabilities improved as more attributes were removed. The detection rate of the eigendecomposition FDI method remained relatively constant for the first few iterations, after which it experienced a dip in performance. At the 40th and 70th percentile threshold, the eigendecomposition FDI method experienced anomalies, whereby the detection rate decreased and then reversed course by increasing again with the following iteration. The detection rate of the residual-based FDI method remained relatively constant until the 80th percentile threshold was used, at which point it achieved a perfect detection rate. Once the 90th percentile threshold was used, its detection capabilities collapsed altogether.

All three FDI methods experienced a deterioration of overall isolation capabilities, with the residual-based method deteriorating to zero percent. However, the distance FDI method experienced the most gradual deterioration of all three methods. Several anomalies occurred in the isolation rate data, with the first occurring when the isolation capability of both the eigendecomposition and residual-based FDI methods decreased rapidly once the 30th percentile threshold was exchanged with the 40th percentile threshold. Another anomaly occurred when the isolation capability of the eigendecomposition method increased rapidly once the 70th percentile threshold was exchanged with the 80th percentile threshold, after which it decreased again.

The isolation rates of the specific faults which each FDI method could successfully isolate for the majority of sample measurements are recorded in Table 6.1. From these specific isolation rates, it can be noted that all three FDI methods are capable of retaining relatively high specific isolation rates up until the 30th percentile, after which at least one of the specific isolation rates of all three methods became insufficient. The eigendecomposition FDI method experienced a poor specific isolation rate for Fault 2 when the 10th percentile threshold was used, after which that specific isolation rate increased substantially, creating another anomaly.

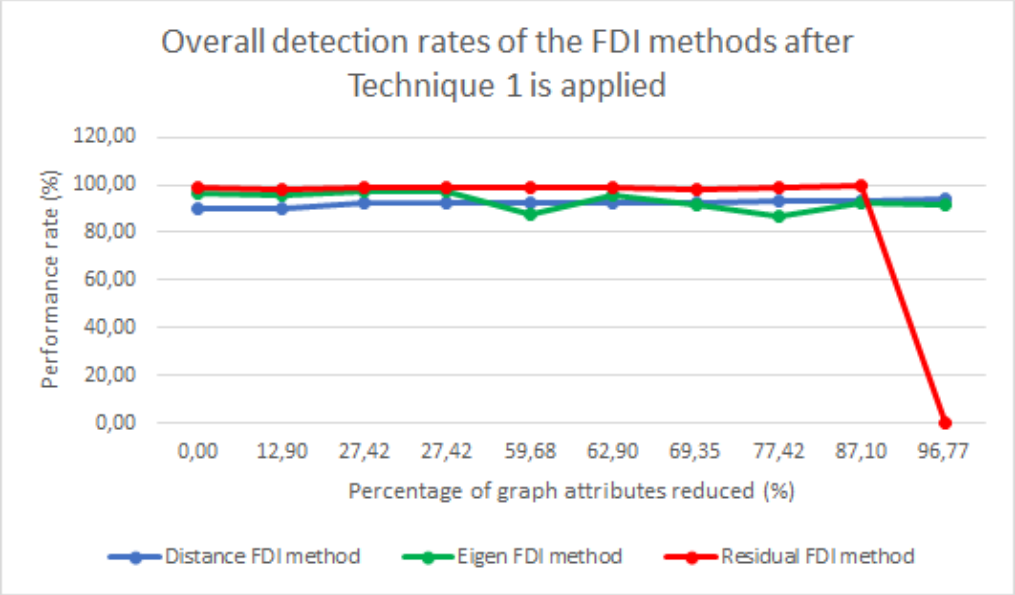


Figure 6.1: The overall detection rates after applying Technique 1.

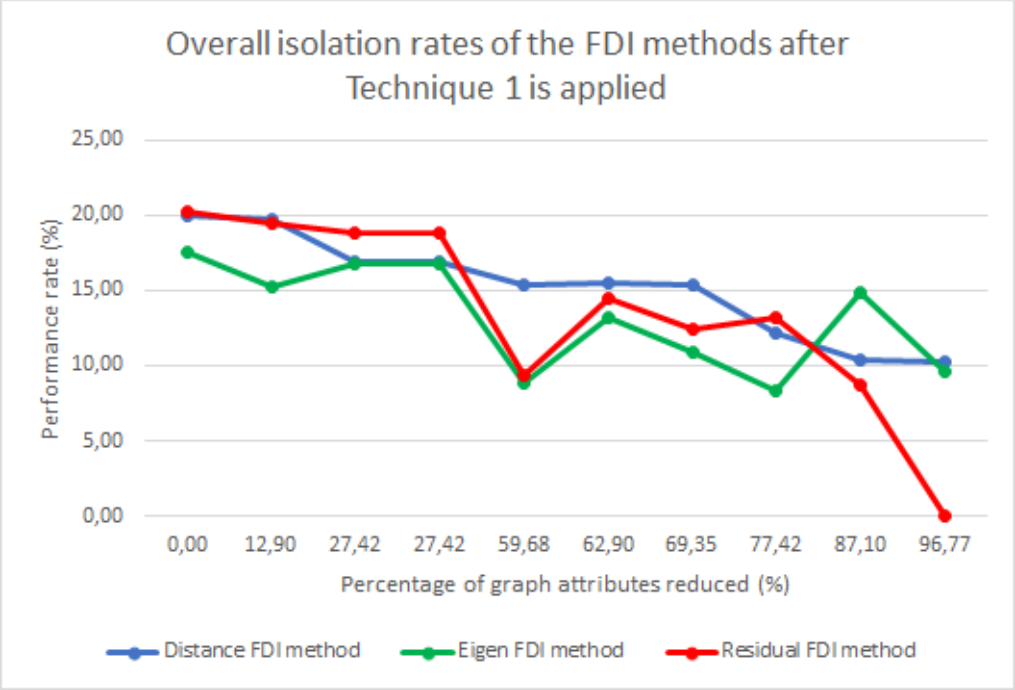


Figure 6.2: The overall isolation rates after applying Technique 1.

6.2.2 Link attribute filtering with variation analysis

The methodology of this technique is similar to that of the previous technique, with the main difference being that link attribute values are filtered out instead of node attribute values. The attributed graph of the system under NOC remains the reference graph. The methodology used for evaluating this technique is as follows:

- Generate the attributed graph data for all 20 fault conditions as well as the normal condition. For all the non-zero link attributes, express how much the attribute value varies for each fault condition relative to that link's attribute value while the system is under NOC, as a percentage value.
- Calculate the average percentage value with which each non-zero link attribute value varies for all the 20 fault conditions. This produces a 15×15 variation matrix where each non-zero entry represents a specific link attributes average percentage variation for all the fault conditions.
- Use the 'isoutlier(A,'percentiles',threshold)' function in MATLAB® to identify the non-zero outliers found in the 10th percentile of the variation matrix. These outliers represent the non-zero links whose attribute values vary very little for all 20 process faults relative to the NOC and are highly unlikely to contribute to the detection and isolation of faults.
- Reduce the attributed graph data of the TEP by setting the link attribute values identified as lower outliers in the variation matrix to zero.
- Use the reduced graph data and apply the three FDI methods used in Chapter 4 to determine the detection and isolation capabilities of all three FDI methods.
- Increase the percentile threshold in increments of 10 until the 90th percentile has been reached. After each increase, repeat the previous three steps.

The overall detection and isolation rates of the FDI methods relative to the percentage of attributes removed from the graph data can be seen in Figures 6.3 & 6.4, respectively. The detection rate of the residual-based FDI method remained relatively constant as more attributes were reduced, indicating a robust reaction to the type of reduction performed by the technique. On the other hand, the detection rate of the distance FDI method showed an upward trajectory. While the initial and final values of the overall detection rate of the eigendecomposition FDI method were very close, its detection rate had a turbulent response to the attribute reduction performed by Technique 2.

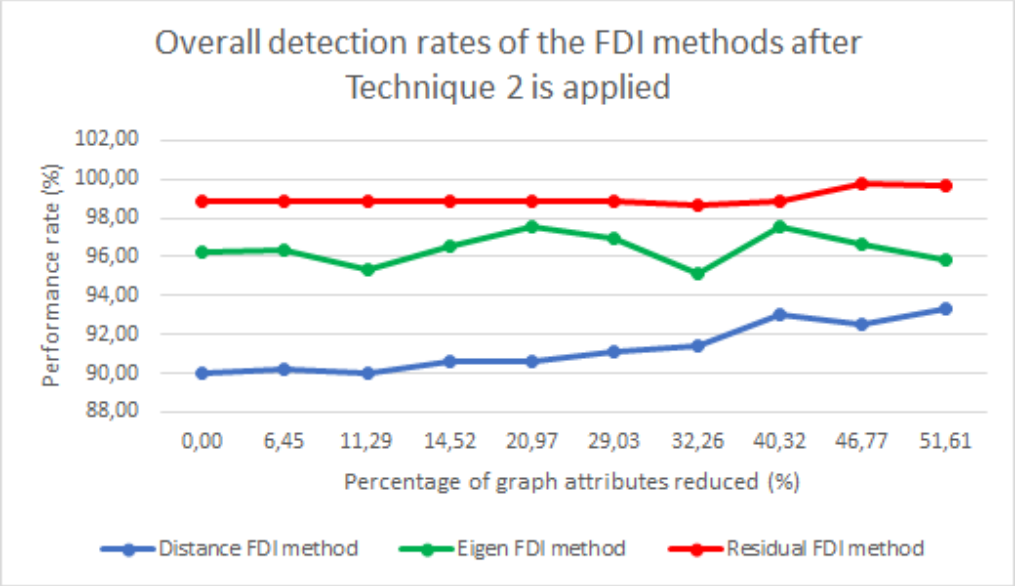


Figure 6.3: The overall detection rates after applying Technique 2.

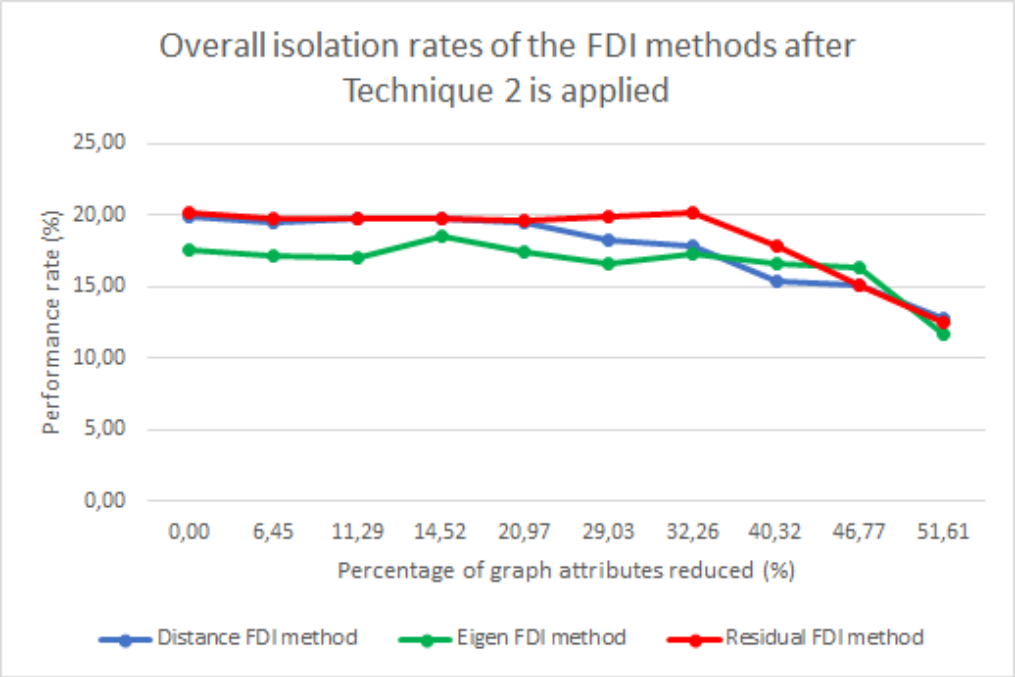


Figure 6.4: The overall isolation rates after applying Technique 2.

The overall isolation rates of all three FDI methods experienced a downward trajectory as more attributes were reduced by this technique, with the residual-based FDI method's rate being largely unaffected until the 60th percentile threshold, after which it deteriorated rapidly. In addition, both the overall detection and isolation rates experienced anomalies, whereby the

rates changed their trajectories at a specific interval of attribute reduction and then once again changed trajectory as the reduction interval increased.

Table 6.2 contains the specific isolation rates of fault conditions which each FDI method was capable of successfully isolating for a majority of sample measurements. The specific isolation rates achieved by the distance parameter, eigendecomposition, and residual-based FDI methods all remained relatively high up until the 60th, 70th, & 80th percentile thresholds were respectively applied to each method. After those respective intervals, the specific isolation rates became subpar.

6.2.3 Link attribute filtering with a size threshold

The rationale behind this technique is that smaller attribute values are likely to contribute less to the fault detection and isolation process than larger attribute values. This technique analyses the size of the link attribute values of the attributed graph while the system is under NOC to determine which links should be removed from the attributed graph data. The methodology used for evaluating this technique is as follows:

- Generate the attributed graph data for the normal condition.
- Apply the 'isoutlier(A,'percentiles',threshold)' function in MATLAB® to the non-zero link attribute values obtained in the previous step to identify the outliers found in the 10th percentile. These outliers represent the non-zero links whose attribute values are so small that they are unlikely to contribute to the detection and isolation of faults.
- Reduce all the attributed graph data of the TEP by setting the link attribute values identified as lower outliers in the previous step to zero.
- Use the reduced graph data and apply the three FDI methods used in Chapter 4 to determine the detection and isolation capabilities of all three FDI methods.
- Increase the percentile threshold in increments of 10 until the 90th percentile has been reached. After each increase, repeat the previous three steps.

The overall detection and isolation rates of the FDI methods relative to the percentage of attributes removed from the graph data are illustrated in Figures 6.5 & 6.6, respectively. The overall detection rates of all three FDI methods after being reduced by Technique 3 are similar to those of the three FDI methods after being reduced by Technique 2. There are, however, a few stark exceptions, such as the final overall detection rate of the eigendecomposition FDI method being much lower than the initial value and the distance parameter FDI method experiencing a dip in detection rate during the early reduction iterations before continuing on a slight upward trajectory.

Also similar to the results after Technique 2 was applied, the overall isolation rates of all three FDI methods experienced a downward trajectory as this technique reduced more attributes. The residual-based FDI method's rate experienced the steepest downward trajectory, while the overall isolation rate of the eigendecomposition FDI method had a turbulent response to the reduction technique. Anomalies could be identified in both the overall detection and isolation rates since some methods experienced a decrease in rates followed by an increase in rates as more attributes were reduced. The opposite effect also occurred.

The specific isolation rates of fault conditions which each FDI method was capable of successfully isolating for a majority of sample measurements can be found in Table 6.3. From examining these results, it is evident that all three FDI methods experienced a significant drop in at least one of their specific isolation rates once the 10th percentile threshold was applied. Once the percentile threshold was increased, the deterioration in specific isolation rates became worse in most instances. The isolation rate of Fault 6, which is a very distinct fault, was high for all three methods once the 80th percentile threshold was applied. This indicates that the most vital structural information needed to diagnose Fault 6 is encapsulated in the remaining graph data once Technique 3 is applied to the original graph data with an 80th percentile threshold.

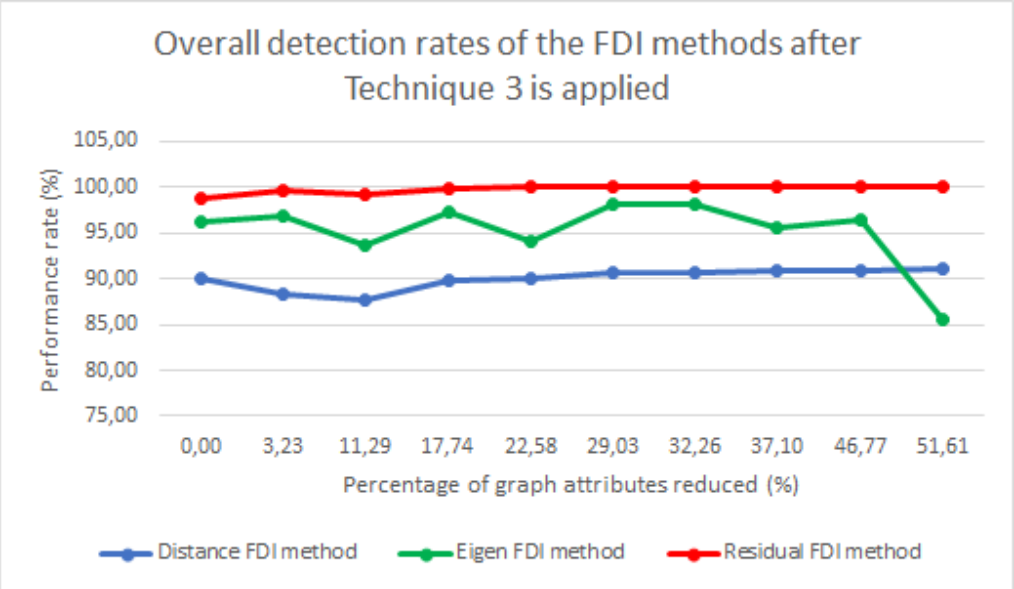


Figure 6.5: The overall detection rates after applying Technique 3.

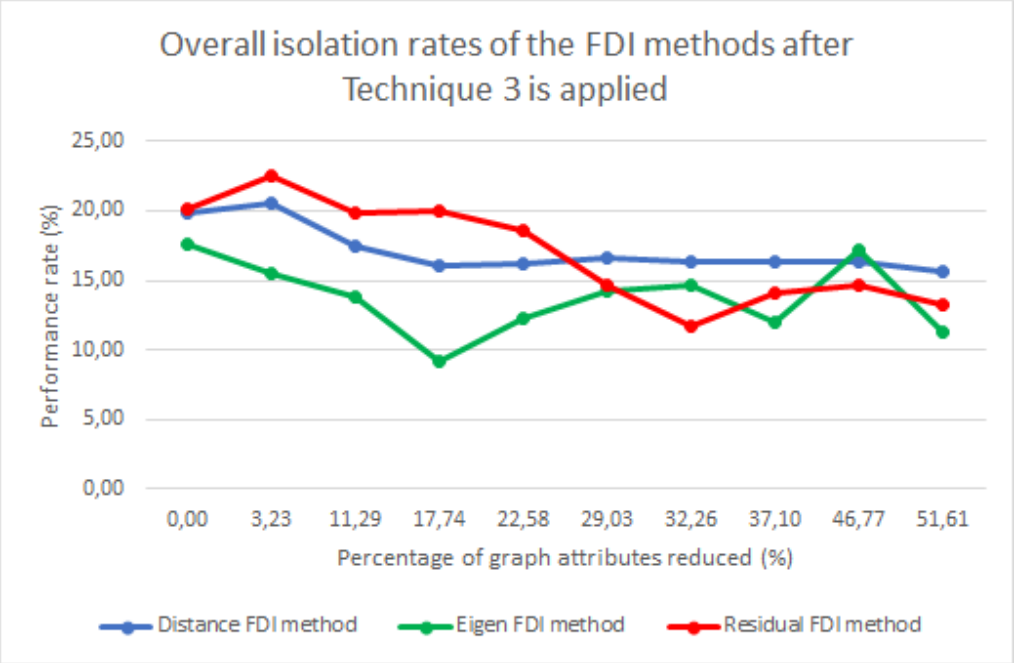


Figure 6.6: The overall isolation rates after applying Technique 3.

6.2.4 Node attribute filtering with a size threshold

The methodology and the rationale behind this technique are both similar to that of the previous technique. This technique, however, analyses the size of the node attribute values of the attributed graph while the system is under NOC to determine which nodes should be removed from the attributed graph data. The methodology used for evaluating this technique is as follows:

- Generate the attributed graph data for the normal condition.
- For each node, summate the chemical and physical exergy attribute values of that node.
- Apply the 'isoutlier(A,'percentiles',threshold)' function in MATLAB® to all the summated node attribute values obtained in the previous step to identify the outliers found in the 10th percentile. These outliers represent the nodes whose attribute values are so small that they are unlikely to contribute to the detection and isolation of faults.
- Reduce all the attributed graph data of the TEP by removing the nodes identified as outliers.
- Use the reduced graph data and apply the three FDI methods used in Chapter 4 to determine the detection and isolation capabilities of all three FDI methods.
- Increase the percentile threshold in increments of 10 until the 90th percentile has been reached. After each increase, repeat the previous three steps.

Figures 6.7 & 6.8 show the respective overall detection and isolation rates, relative to the percentage of attributes removed from the graph data. From Figure 6.7, it can be seen that the overall detection rate of the distance FDI method experienced a slight upward trajectory, while the detection rate of the eigendecomposition FDI method experienced a slight downward trajectory. Like in the case of Technique 1, the detection rate of the residual-based FDI method remained relatively constant until the 70th percentile threshold was used, at which point it achieved a perfect detection rate. Once the 90th percentile threshold was used, its detection capabilities collapsed altogether.

A possible explanation for this occurrence is that each entry in a residual matrix represents only one attribute, while an entry in a cost matrix depends on more than one attribute. Therefore, when node reduction techniques, which remove entire rows and columns from an NSM such as Techniques 1 & 4, are applied to the graph data at a 90th percentile thresholds, the reduced graph's NSM is so small that it becomes easier to diagnose faults from cost matrices than residual matrices.

The overall isolation rates of all three FDI methods experienced a downward trajectory as more attributes were reduced with Technique 4. The distance FDI method and the eigendecomposition FDI method both had a very similar trajectory. The final reduction resulted in the total diminishing of the residual-based FDI method's overall isolation rate. There were a few instances where anomalies took place with both the overall detection and isolation rates.

The isolation rates of the specific faults which each FDI method could successfully isolate for the majority of sample measurements are recorded in Table 6.4. After applying the first attribute reduction iteration to the graph data, all three FDI methods severely lost the specific isolation capacity of at least one of their specific fault conditions. The distance FDI method was the most effective at maintaining a high isolation rate for Fault 6 as the reduction iterations increased. Several anomalies can be observed in the specific isolation rates of all three methods.

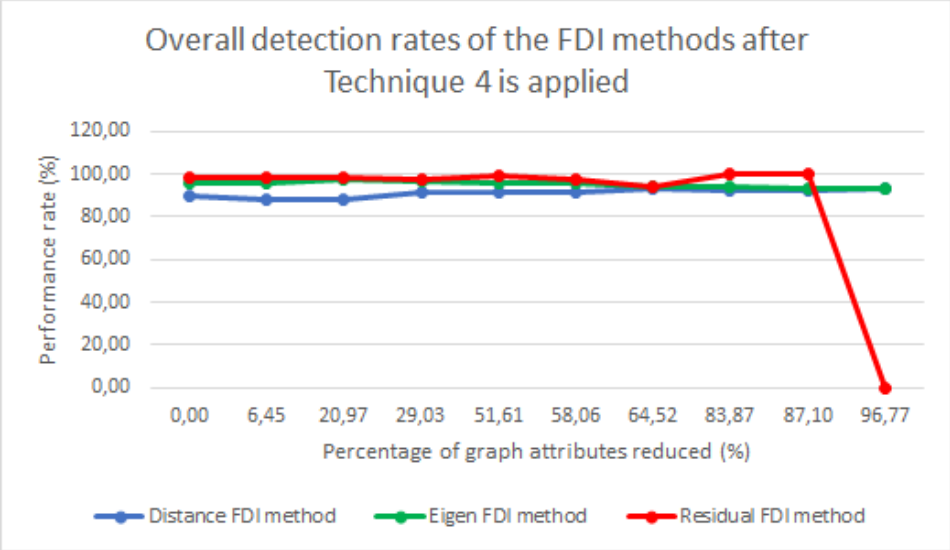


Figure 6.7: The overall detection rates after applying Technique 4.

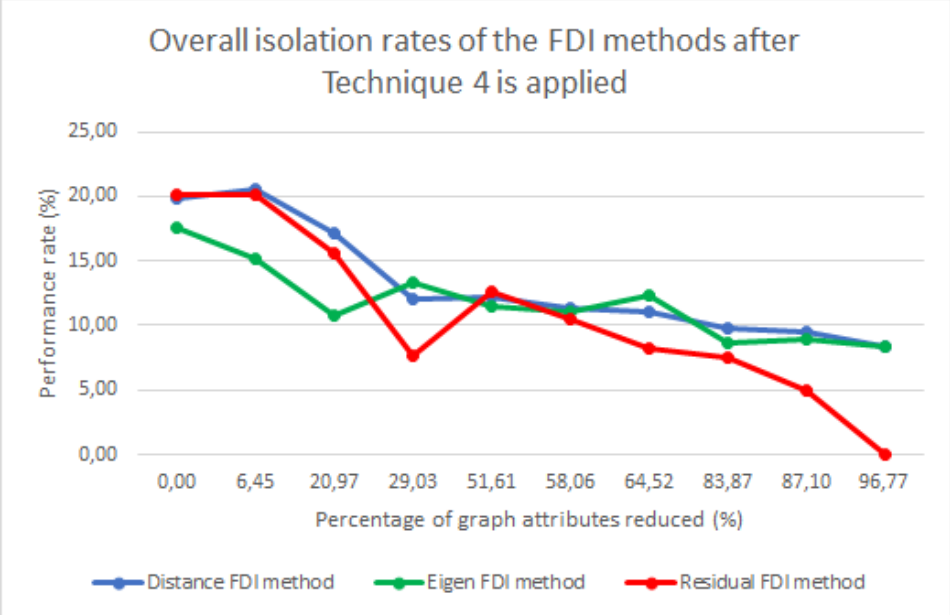


Figure 6.8: The overall isolation rates after applying Technique 4.

6.2.5 Summarization of similar nodes

This technique identifies two or more nodes with similar attribute profiles and summarizes these nodes into a single node. Thus, this technique preserves all the original attribute information while reducing the structural complexity and, thus, the complexity of the mathematical

operations applied to the graph data. The methodology used for evaluating this technique is as follows:

- Generate the attributed graph data for the normal condition.
- Summate the physical and chemical exergy attribute values for each node in the attributed graph.
- For each node in the attributed graph, calculate the difference between the summated attribute value of that node and the summated attribute value of all the other nodes in the graph.
- Identify two nodes with similar attribute size profiles and, thus, a relatively small difference between them. One of these nodes is selected as the primary node and will remain in the graph, while the other node is the secondary node and will be removed from the graph once the summarization process is complete.
- Add the physical and chemical exergy attribute values of the secondary node to the primary node's physical and chemical exergy attribute values, respectively.
- Take all the links between the secondary node and all the other nodes in the graph and connect those links to the primary node. If there exists a link between the primary and secondary nodes, simply omit it from the new summarized graph.
- Remove the secondary node and all the links that connect to that node from the attributed graph to form the new summarized attributed graph.
- Repeat this process until there are no more nodes with similar attribute profiles.

The overall detection and isolation rates of the FDI methods relative to the percentage of attributes removed from the graph data can be seen in Figures 6.9 & 6.10, respectively. The overall detection rate of all three FDI methods remained relatively constant as more attributes were reduced with Technique 5. However, the final overall detection rate of the eigendecomposition FDI method was slightly lower than the initial rate. Conversely, the final overall detection rate was slightly higher than the initial rate for both the distance and residual-based FDI methods.

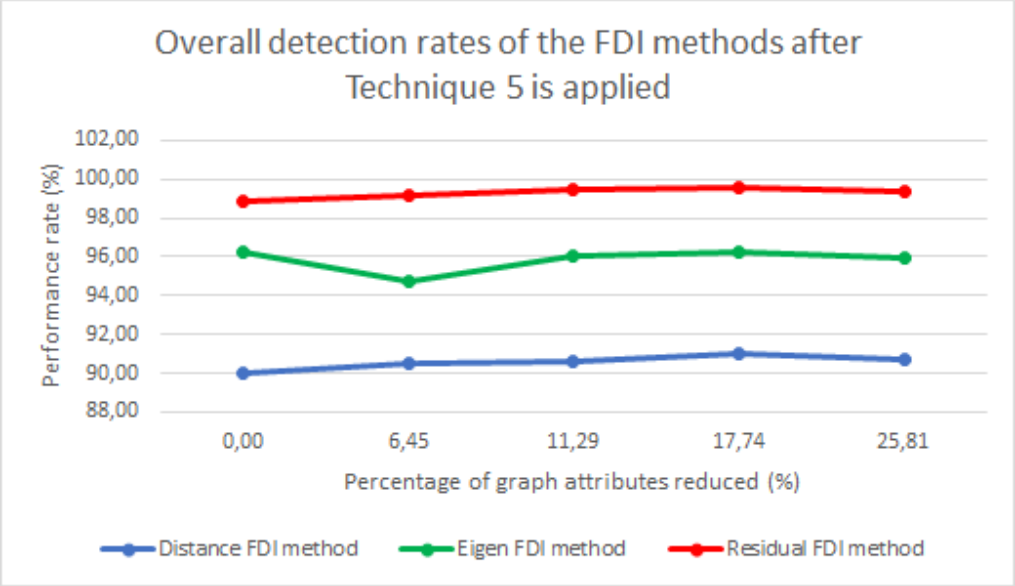


Figure 6.9: The overall detection rates after applying Technique 5.

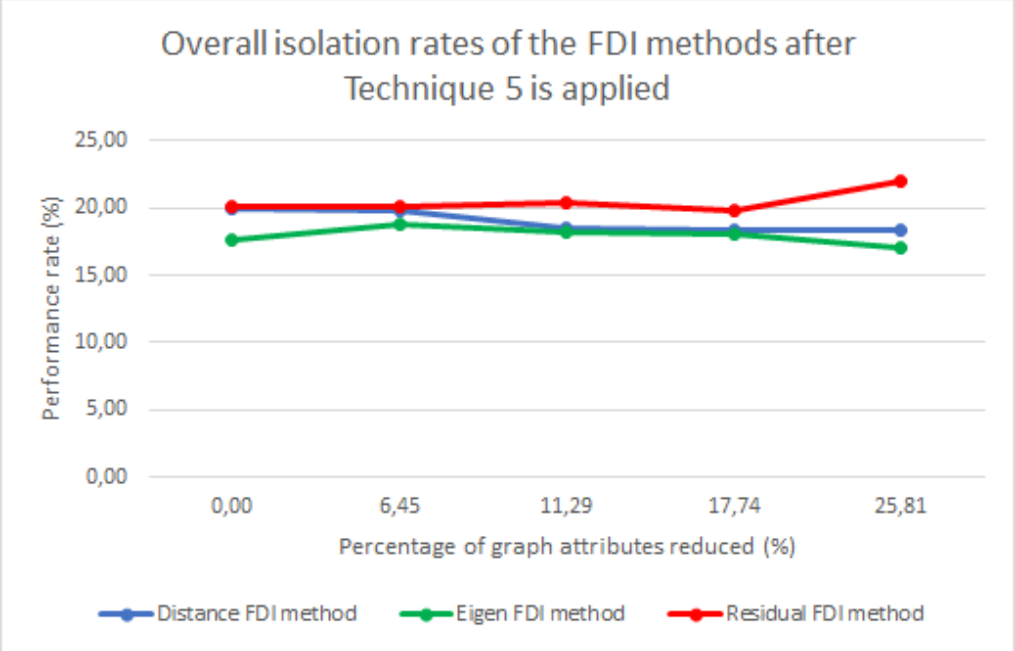


Figure 6.10: The overall isolation rates after applying Technique 5.

The overall isolation rates of the three FDI methods also remained relatively stable as more attributes were reduced. The initial and final overall isolation rates were very close for the eigendecomposition FDI method. The final overall isolation rate of the distance FDI method was slightly lower than the initial rate, while the final overall isolation rate of the residual-based FDI method was slightly higher than the initial rate.

When looking at the specific isolation rates of the FDI methods as recorded in Table 6.5, it can be seen that the specific isolation rates of both the eigendecomposition and residual-based FDI methods remained relatively high and even improved, as more node summarizations were conducted. Only one of the distance FDI method's specific isolation rates dropped slightly below 50 %. This drop occurred when the third pair of nodes were summarized. This is, however, not a significant drop since the specific isolation rate before any nodes were summarized was 59 %.

The relatively consistent performance of all three FDI methods after being reduced by Technique 5 can be ascribed to the fact that no attributes are discarded from the graph data. Instead, Technique 5 simply reorganizes the location of attributes within the structure of the process graph data.

Table 6.1: Performance of all three FDI methods using the attributed graph data as reduced by Technique 1.

Distance FDI Method										
Percentiles	0th	10th	20th	30th	40th	50th	60th	70th	80th	90th
Overall Detection Rate (%)	90.00	89.85	92.15	92.15	92.74	92.78	92.72	93.43	93.66	94.46
Overall Isolation Rate (%)	19.90	19.65	16.95	16.95	15.40	15.45	15.30	12.10	10.35	10.25
Attribute Reduction (%)	0.00	12.90	27.42	27.42	59.68	62.90	69.35	77.42	87.10	96.77
Isolation rate of Fault 1 (%)	75.00	74.00	70.00	70.00	4.00	4.00	3.00	8.00	16.00	75.00
Isolation rate of Fault 2 (%)	85.00	87.00	80.00	80.00	79.00	79.00	78.00	84.00	32.00	10.00
Isolation rate of Fault 6 (%)	59.00	59.00	56.00	56.00	79.00	79.00	79.00	38.00	55.00	14.00
Eigen FDI Method										
Percentiles	0th	10th	20th	30th	40th	50th	60th	70th	80th	90th
Overall Detection Rate (%)	96.28	96.01	97.29	97.29	87.84	95.38	91.78	86.58	92.51	91.90
Overall Isolation Rate (%)	17.55	15.25	16.80	16.80	8.90	13.20	10.85	8.35	14.90	9.55
Attribute Reduction (%)	0.00	12.90	27.42	27.42	59.68	62.90	69.35	77.42	87.10	96.77
Isolation rate of Fault 1 (%)	90.00	87.00	94.00	94.00	8.00	11.00	9.00	8.00	63.00	78.00
Isolation rate of Fault 2 (%)	75.00	26.00	50.00	50.00	64.00	62.00	89.00	92.00	79.00	18.00
Isolation rate of Fault 6 (%)	100.00	99.00	99.00	99.00	10.00	87.00	9.00	10.00	78.00	20.00
Residual FDI Method										
Percentiles	0th	10th	20th	30th	40th	50th	60th	70th	80th	90th
Overall Detection Rate (%)	98.86	98.41	98.66	98.66	98.95	98.73	98.30	98.66	100.00	0.00
Overall Isolation Rate (%)	20.15	19.50	18.85	18.85	9.40	14.50	12.40	13.20	8.65	0.00
Attribute Reduction (%)	0.00	12.90	27.42	27.42	59.68	62.90	69.35	77.42	87.10	96.77
Isolation rate of Fault 1 (%)	80.00	83.00	89.00	89.00	23.00	52.00	60.00	80.00	99.00	0.00
Isolation rate of Fault 2 (%)	61.00	72.00	79.00	79.00	45.00	58.00	80.00	93.00	0.00	0.00
Isolation rate of Fault 6 (%)	88.00	87.00	80.00	80.00	0.00	85.00	0.00	0.00	0.00	0.00

Table 6.2: Performance of all three FDI methods using the attributed graph data as reduced by Technique 2.

Distance FDI Method										
Percentiles	0th	10th	20th	30th	40th	50th	60th	70th	80th	90th
Overall Detection Rate (%)	90.00	90.20	89.98	90.57	90.57	91.11	91.43	92.99	92.56	93.34
Overall Isolation Rate (%)	19.90	19.50	19.80	19.80	19.45	18.20	17.85	15.35	15.05	12.75
Attribute Reduction (%)	0.00	6.45	11.29	14.52	20.97	29.03	32.26	40.32	46.77	51.61
Isolation rate of Fault 1 (%)	75.00	74.00	74.00	73.00	72.00	69.00	70.00	78.00	67.00	74.00
Isolation rate of Fault 2 (%)	85.00	85.00	87.00	86.00	86.00	80.00	77.00	60.00	92.00	38.00
Isolation rate of Fault 6 (%)	59.00	58.00	60.00	59.00	59.00	56.00	53.00	41.00	26.00	29.00
Eigen FDI Method										
Percentiles	0th	10th	20th	30th	40th	50th	60th	70th	80th	90th
Overall Detection Rate (%)	96.28	96.33	95.36	96.57	97.55	96.98	95.11	97.60	96.68	95.88
Overall Isolation Rate (%)	17.55	17.20	17.05	18.55	17.40	16.60	17.25	16.55	16.30	11.70
Attribute Reduction (%)	0.00	6.45	11.29	14.52	20.97	29.03	32.26	40.32	46.77	51.61
Isolation rate of Fault 1 (%)	90.00	93.00	83.00	94.00	88.00	88.00	89.00	88.00	94.00	96.00
Isolation rate of Fault 2 (%)	75.00	70.00	58.00	72.00	63.00	61.00	66.00	68.00	49.00	34.00
Isolation rate of Fault 6 (%)	100.00	100.00	100.00	100.00	99.00	91.00	92.00	93.00	90.00	3.00
Residual FDI Method										
Percentiles	0th	10th	20th	30th	40th	50th	60th	70th	80th	90th
Overall Detection Rate (%)	98.86	98.88	98.89	98.84	98.86	98.85	98.67	98.87	99.81	99.64
Overall Isolation Rate (%)	20.15	19.70	19.75	19.70	19.60	19.90	20.20	17.85	15.15	12.45
Attribute Reduction (%)	0.00	6.45	11.29	14.52	20.97	29.03	32.26	40.32	46.77	51.61
Isolation rate of Fault 1 (%)	80.00	80.00	80.00	80.00	80.00	80.00	80.00	80.00	81.00	80.00
Isolation rate of Fault 2 (%)	61.00	61.00	61.00	61.00	61.00	61.00	61.00	61.00	60.00	22.00
Isolation rate of Fault 6 (%)	88.00	87.00	87.00	87.00	87.00	87.00	87.00	87.00	88.00	85.00

Table 6.3: Performance of all three FDI methods using the attributed graph data as reduced by Technique 3.

Distance FDI Method										
Percentiles	0th	10th	20th	30th	40th	50th	60th	70th	80th	90th
Overall Detection Rate (%)	90.00	88.35	87.71	89.72	90.09	90.62	90.68	90.94	90.86	91.09
Overall Isolation Rate (%)	19.90	20.55	17.45	16.10	16.15	16.60	16.40	16.30	16.30	15.60
Attribute Reduction (%)	0.00	3.23	11.29	17.74	22.58	29.03	32.26	37.10	46.77	51.61
Isolation rate of Fault 1 (%)	75.00	10.00	16.00	11.00	11.00	10.00	10.00	10.00	8.00	8.00
Isolation rate of Fault 2 (%)	85.00	93.00	23.00	18.00	15.00	10.00	10.00	10.00	12.00	13.00
Isolation rate of Fault 6 (%)	59.00	87.00	88.00	84.00	83.00	82.00	82.00	83.00	77.00	64.00
Eigen FDI Method										
Percentiles	0th	10th	20th	30th	40th	50th	60th	70th	80th	90th
Overall Detection Rate (%)	96.28	96.81	93.57	97.32	94.19	98.15	98.12	95.64	96.50	85.60
Overall Isolation Rate (%)	17.55	15.50	13.85	9.20	12.30	14.30	14.60	12.05	17.20	11.25
Attribute Reduction (%)	0.00	3.23	11.29	17.74	22.58	29.03	32.26	37.10	46.77	51.61
Isolation rate of Fault 1 (%)	90.00	3.00	15.00	13.00	10.00	7.00	8.00	5.00	10.00	7.00
Isolation rate of Fault 2 (%)	75.00	93.00	15.00	35.00	9.00	5.00	4.00	8.00	4.00	11.00
Isolation rate of Fault 6 (%)	100.00	89.00	56.00	5.00	65.00	9.00	10.00	14.00	85.00	9.00
Residual FDI Method										
Percentiles	0th	10th	20th	30th	40th	50th	60th	70th	80th	90th
Overall Detection Rate (%)	98.86	99.56	99.32	99.91	99.97	100.00	100.00	100.00	100.00	100.00
Overall Isolation Rate (%)	20.15	22.50	19.80	19.95	18.60	14.70	11.65	14.10	14.60	13.20
Attribute Reduction (%)	0.00	3.23	11.29	17.74	22.58	29.03	32.26	37.10	46.77	51.61
Isolation rate of Fault 1 (%)	80.00	24.00	9.00	15.00	11.00	11.00	11.00	3.00	28.00	50.00
Isolation rate of Fault 2 (%)	61.00	95.00	34.00	52.00	21.00	8.00	17.00	4.00	14.00	15.00
Isolation rate of Fault 6 (%)	88.00	79.00	84.00	83.00	82.00	64.00	0.00	67.00	78.00	77.00

Table 6.4: Performance of all three FDI methods using the attributed graph data as reduced by Technique 4.

Distance FDI Method										
Percentiles	0th	10th	20th	30th	40th	50th	60th	70th	80th	90th
Overall Detection Rate (%)	90.00	88.36	88.21	91.64	91.88	91.89	93.31	92.38	92.57	93.12
Overall Isolation Rate (%)	19.90	20.50	17.20	12.00	12.20	11.35	11.00	9.75	9.45	8.35
Attribute Reduction (%)	0.00	6.45	20.97	29.03	51.61	58.06	64.52	83.87	87.10	96.77
Isolation rate of Fault 1 (%)	75.00	9.00	17.00	23.00	12.00	11.00	9.00	10.00	10.00	14.00
Isolation rate of Fault 2 (%)	85.00	93.00	48.00	37.00	28.00	11.00	13.00	13.00	11.00	12.00
Isolation rate of Fault 6 (%)	59.00	87.00	90.00	61.00	88.00	84.00	82.00	69.00	63.00	27.00
Eigen FDI Method										
Percentiles	0th	10th	20th	30th	40th	50th	60th	70th	80th	90th
Overall Detection Rate (%)	96.28	96.23	97.73	96.42	95.91	96.30	93.80	94.50	93.31	93.16
Overall Isolation Rate (%)	17.55	15.15	10.75	13.25	11.50	11.00	12.35	8.65	8.95	8.30
Attribute Reduction (%)	0.00	6.45	20.97	29.03	51.61	58.06	64.52	83.87	87.10	96.77
Isolation rate of Fault 1 (%)	90.00	16.00	25.00	8.00	33.00	22.00	22.00	34.00	28.00	12.00
Isolation rate of Fault 2 (%)	75.00	92.00	19.00	68.00	13.00	8.00	32.00	19.00	17.00	6.00
Isolation rate of Fault 6 (%)	100.00	89.00	92.00	84.00	82.00	55.00	79.00	9.00	30.00	73.00
Residual FDI Method										
Percentiles	0th	10th	20th	30th	40th	50th	60th	70th	80th	90th
Overall Detection Rate (%)	98.86	98.79	98.20	97.95	99.10	97.85	94.39	100.00	100.00	0.00
Overall Isolation Rate (%)	20.15	20.05	15.60	7.65	12.65	10.50	8.25	7.45	4.95	0.00
Attribute Reduction (%)	0.00	6.45	20.97	29.03	51.61	58.06	64.52	83.87	87.10	96.77
Isolation rate of Fault 1 (%)	80.00	11.00	14.00	13.00	9.00	7.00	31.00	86.00	85.00	0.00
Isolation rate of Fault 2 (%)	61.00	95.00	0.00	0.00	17.00	0.00	0.00	0.00	0.00	0.00
Isolation rate of Fault 6 (%)	88.00	78.00	76.00	2.00	85.00	85.00	0.00	0.00	0.00	0.00

Table 6.5: Performance of all three FDI methods using the attributed graph data as reduced by Technique 5.

Distance FDI Method					
Number of Mergers	0	1	2	3	4
Overall Detection Rate (%)	90.00	90.48	90.57	91.00	90.68
Overall Isolation Rate (%)	19.90	19.80	18.50	18.35	18.40
Attribute Reduction (%)	0.00	6.45	11.29	17.74	25.81
Isolation rate of Fault 1 (%)	75.00	73.00	75.00	75.00	77.00
Isolation rate of Fault 2 (%)	85.00	83.00	87.00	87.00	87.00
Isolation rate of Fault 6 (%)	59.00	58.00	50.00	49.00	47.00
Eigen FDI Method					
Number of Mergers	0	1	2	3	4
Overall Detection Rate (%)	96.28	94.70	96.02	96.29	95.94
Overall Isolation Rate (%)	17.55	18.75	18.20	18.05	17.05
Attribute Reduction (%)	0.00	6.45	11.29	17.74	25.81
Isolation rate of Fault 1 (%)	90.00	91.00	90.00	93.00	92.00
Isolation rate of Fault 2 (%)	75.00	73.00	88.00	82.00	77.00
Isolation rate of Fault 6 (%)	100.00	100.00	98.00	96.00	100.00
Residual FDI Method					
Number of Mergers	0	1	2	3	4
Overall Detection Rate (%)	98.86	99.16	99.45	99.58	99.37
Overall Isolation Rate (%)	20.15	20.10	20.45	19.75	21.95
Attribute Reduction (%)	0.00	6.45	11.29	17.74	25.81
Isolation rate of Fault 1 (%)	80.00	83.00	87.00	89.00	90.00
Isolation rate of Fault 2 (%)	61.00	72.00	77.00	77.00	76.00
Isolation rate of Fault 6 (%)	88.00	87.00	85.00	79.00	76.00

6.3 Interpretation of results

The results provided in the previous section mainly indicate a negative linear relationship between the overall isolation capabilities of the FDI methods and the increase in the number of attributes reduced by the various reduction techniques. The overall detection capabilities of the FDI methods, in contrast, remain relatively constant as more attributes are reduced. This can be ascribed to the fact that for a method to detect a fault, the method must simply discern between two options - faulty or normal. To isolate a fault successfully, the method must discern between 20 conditions. When attributes are removed, the FDI method has fewer data to analyse, and it is more manageable for the method to detect (decide between 2 options) than isolate (decide between 20 options).

It is unclear, at this stage, if the direction and slope of the trajectories observed in the results provided in the previous section are the rules or the exceptions when it comes to graph reduction. It is, therefore, necessary to apply these reduction techniques to the attributed graph data of another industrial process to compare the nature of the relationship between the overall detection and isolation capabilities and an increase of the number of attributes reduced of the TEP with that of the other industrial process.

Several anomalies occurred in the overall detection and isolation rates, as well as in the specific isolation rates. Two types of anomalies could be identified from the results in the previous section. The first type occurs when a downward trajectory changes to an upward trajectory as the reduction interval is increased and then back to a downwards trajectory again once the reduction interval is increased again, creating a local maximum. The second type occurs when an upward trajectory changes to a downwards trajectory as the reduction interval is increased and then back to an upwards trajectory again once the reduction interval is increased again, creating a local minimum.

From these anomalies, specific observations can be made. The most important observation is that there exists structural information in the graph data that is vital to FDI performance as well as structural information that is non-vital to its performance. For instance, when the overall detection and isolation rates, as well as the specific isolation rates, experience a local maximum or a local minimum, it can be observed that the reduction iteration removes attributes (non-vital structural information) that obfuscate the FDI method on the upward trajectory part of the local maximum or minimum. It follows logically that the downward trajectory part of the local maximum or minimum results from the reduction iteration removing attributes (vital structural information) that bolster the FDI method.

When looking at the specific isolation rates of all the FDI methods as more attributes are reduced by the various reduction techniques, it can further be seen that specific attributes can obscure or bolster the isolation capability of the FDI method with regards to a particular fault condition, while having the opposite or even no effect on the capability of the method with regards to other fault conditions. Such is the case with the specific isolation rates of the distance FDI method, which are recorded in Table 6.1. For example, once the 30th percentile threshold was increased to the 40th percentile threshold, the isolation rate of Fault 1 fell from 70 % to 4 %, the isolation rate of Fault 2 only dropped by one percent, and the isolation rate of Fault 6 increased by 23 %.

This can be ascribed to the fact that a specific reduction operation can alter the structural information so that the attribute features that the FDI method uses to diagnose a particular fault are accentuated. In contrast, the attribute features that the same method uses to diagnose a different fault are attenuated.

By conducting an intertechnique comparison, it is possible to determine which reduction techniques better complement each of the three FDI methods. This comparison involves assessing the overall detection and isolation rates, as well as the specific isolation rates of each FDI method as stated in Tables 6.1 - 6.5. For example, for the distance FDI method, Techniques 1, 2, and 5 at specific thresholds were the most complementary reduction techniques when examining the performance of the FDI method after these techniques were applied to the graph data.

When looking at the performance of the eigendecomposition FDI method after the various reduction techniques were applied to the graph data, Techniques 2 and 5 at specific thresholds were also able to maximally reduce attributes while minimally affecting performance. Techniques 1, 2, and 5 at specific thresholds were once again able to maximally reduce the attributed graph data while minimally deteriorating the performance of the residual-based FDI method. While Techniques 3 & 4 positively affected the overall detection and isolation rates at certain reduction intervals, the specific isolation rates were adversely affected in most instances.

This intertechnique comparison also allows for the evaluation of the different types of techniques used to reduce the attributed graph data. For this evaluation, it is essential to note the scale of attribute reduction, seeing as the techniques that reduce nodes can reduce far more non-zero attributes than the techniques that are link attribute orientated. It should also be taken into account that the scale of the x-axes in Figures 6.1 - 6.10 are not necessarily displayed linearly. This evaluation serves to determine which techniques are better at maintaining the level of performance achieved before reduction while still reducing non-vital

structural information.

From assessing the trajectories of the performance indicators of the three FDI methods for each of the five reduction techniques, it is evident that the techniques which rely on attribute variation analysis (Techniques 1 & 2) outperformed the techniques which rely on attribute size analysis (Techniques 3 & 4) across all three FDI methods when the same range of reduction is used for the evaluation. This proves the validity of the argument, which stated that the process noise does not significantly influence the variation analysis performed by Techniques 1 & 2.

Furthermore, when comparing the trajectories of the overall detection rates, overall isolation rates, and the specific isolation rates of the three FDI methods for Technique 5 with that of the other techniques over the same reduction range, it is also evident that Technique 5 resulted in the most stable and consistent response. This indicates that Technique 5 is highly effective at reducing graph complexity while not significantly affecting FDI performance.

Seeing as the aim of graph reduction is to reduce the complexity of the attributed graph, and subsequently the complexity of implementing the FDI methods, it is also necessary to compare the influence that different styles of reduction (node attribute or link attribute) have on the execution time of these methods. Table 6.6 contains a comparison of the execution times of the unreduced graph data, the graph data reduced by Technique 1 with a 10th percentile threshold, and the graph data reduced by Technique 2 with a 20th percentile threshold. The percentage attributes reduced by the node attribute reduction technique (Technique 1) and the link attribute reduction technique (Technique 2) at those respective thresholds are approximately the same. The distance FDI method is used to diagnose graph data sets since it has the longest execution time.

From the table, it is evident that while these two techniques reduced approximately the same percentage of attributes, the node attribute reduction technique (Technique 1) far outperformed the link attribute reduction technique (Technique 2) in terms of reducing the execution time. This is because the node attribute reduction technique reduces the size of the NSM while the link attribute reduction technique only sets attributes within the NSM to zero. To emphasize the usefulness of graph reduction, it is noted that after applying Technique 1 at that specific reduction interval, the FDI execution time decreased by 17 %, while none of the performance indicators decreased by more than 1 %.

Table 6.6: Comparison of the execution times of the distance FDI method after applying different reduction techniques.

Reduction Type	Attribute Reduction (%)	Execution Time (s)
Unreduced	0.00	1365.42
Technique 1	12.90	1133.10
Technique 2	11.29	1323.04

6.4 Combining reduction techniques

It is of particular interest to determine how the performance and execution time of the FDI methods are affected when node and link reduction techniques are applied to the attributed graph data in combination with one another. This approach is akin to using an evolutionary algorithm (EA) to iteratively and randomly combine favourable features from a population of solutions to end up with a better solution.

Unlike evolutionary algorithms, this approach will not be applied randomly. This is because this study is not necessarily focused on developing an optimized reduction technique but rather determining if the performance of the FDI methods achieved when applying well-performing techniques individually, can be improved by combining these well-performing techniques. Furthermore, after applying these combined reduction techniques, it is also essential to gauge the complexity of implementing the FDI methods. This is done by observing the execution times of the FDI methods.

The combined reduction techniques proposed in this section are specific to each FDI method. The combined techniques are formulated by identifying the individual complementary reduction techniques for each FDI method and the reduction interval at which these FDI methods maintained the same level of performance achieved prior to any attributes being reduced. A reduction technique is considered complementary to an FDI method when the reduction technique can maximally reduce the size of the attributed graph data while resulting in a minimal deterioration of FDI performance. To be clear, the combination techniques are not combinations of the algorithms of individual techniques. Instead, the combined techniques simply combine the outcomes of applying the individual techniques at specific reduction intervals. For example, if Technique 1 removes node 2 and Technique 4 removes nodes 5 & 8, the combined approach removes nodes 2, 5, & 8.

Once the FDI method's specific combined reduction technique has been applied to the attributed graph data, the percentage of attributes reduced is calculated, and the method's

performance is determined. The performance is again evaluated in terms of overall detection rates, overall isolation rates, and specific isolation rates. The execution time of each FDI method is also recorded after applying the combined reduction techniques. It should be noted that some of the reduction operations performed by the complementary techniques, which are combined to form the combined techniques, may overlap.

6.4.1 Combining techniques for the distance FDI method

When Technique 1 uses the 10th percentile threshold, Technique 2 uses the 30th percentile threshold, and Technique 5 summarizes a single pair of nodes, they are all individually capable of reducing the graph data in such a way that the distance FDI method maintains relatively the same level of performance as when unreduced graph data is used. The performance is evaluated in terms of the overall detection rate, overall isolation rate, and the specific isolation rates of this FDI method.

Using the 10th percentile threshold, Technique 1 removes node 8 and all the links connected to it from the graph data. By using the 30th percentile threshold, Technique 2 sets the links between nodes 1 & 5, nodes 6 & 7, nodes 6 & 8, nodes 8 & 10, and nodes 4 & 13 to zero. By conducting only one summarization, Technique 5 summarizes nodes 4 & 10. Table 6.7 contains the performance results of the distance FDI method after the combined reduction technique, specifically tailored for the method, was applied to the attributed graph data. These results are compared with the results obtained when the FDI method uses unreduced graph data.

Table 6.7: Performance of the distance FDI method after applying the combined reduction technique.

	Unreduced Graph Data	Reduced Graph Data
Attribute Reduction (%)	0.00	25.81
Overall Detection Rate (%)	89.81	90.75
Overall Isolation Rate (%)	19.90	19.50
Isolation Rate Fault 1 (%)	75.00	72.00
Isolation Rate Fault 2 (%)	85.00	85.00
Isolation Rate Fault 6 (%)	59.00	57.00
FDI Execution Time (s)	1365.42	853.36

From the results recorded in Table 6.7, it is evident that the combined approach as proposed for the distance FDI method can reduce 25.81 % of non-zero attributes while maintaining the level of performance achieved prior to any reduction. Furthermore, when comparing these performance results with the results of the distance FDI method after applying the individual

complementary techniques to reduce approximately the same percentage of attributes, it is clear that the combined technique outperforms each complementary reduction technique.

The combined reduction technique proposed for the distance FDI method also severely decreases the execution time of the FDI method. This indicates that this combined reduction technique is highly effective at reducing graph complexity. The attributed graph diagram of the TEP after being reduced by this combined technique can be seen in Figure 6.11. Graph elements that have been removed are represented with dotted lines and have no colour. Some reduction operations may overlap.

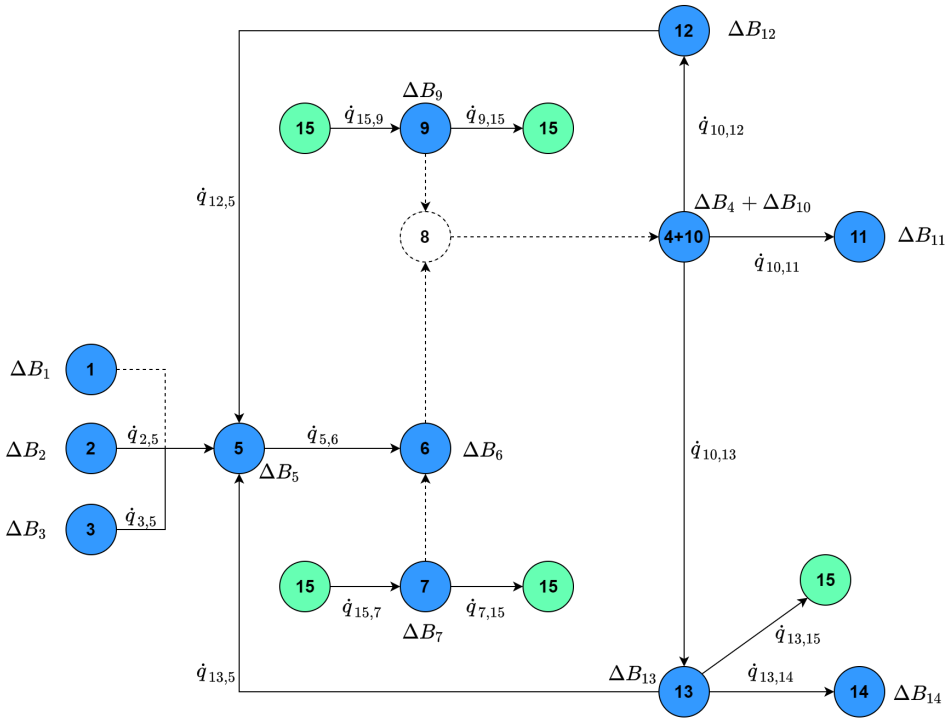


Figure 6.11: The attributed graph diagram after applying the combined reduction technique specific to the distance FDI method.

6.4.2 Combining techniques for the eigendecomposition FDI method

To compile a combined reduction technique which results in the eigendecomposition FDI method maintaining a similar level of performance as it did prior to the graph data being reduced, Technique 2 uses a 70th percentile threshold, and Technique 5 summarizes four pairs of nodes. When Technique 2 is applied, the links between nodes 1 & 5, 2 & 5, 4 & 13, 5 & 1, 5 & 2, 5 & 6, 5 & 12, 5 & 13, 6 & 5, 6 & 7, 6 & 8, 7 & 6, 7 & 15, 8 & 6, 8 & 9, 8 & 10, 9 & 8, 10 & 8, 10 & 12, 13 & 4, 13 & 5, 13 & 14, 14 & 13, and 15 & 7 are all set to zero. Technique

5 again summarizes nodes 4 & 10, 7 & 9, 1 & 2, and 5 & 12. Technique 5 summarizes nodes 4 & 10, 7 & 9, 1 & 2, and 5 & 12.

The performance of the eigendecomposition FDI method after applying the combined reduction technique can be seen in Table 6.8. This combined reduction technique is capable of reducing 51.61 % of non-zero attributes. The performance of the eigendecomposition FDI method after applying this combined technique to the attributed graph data is very similar, and in certain regards better, than the method's performance prior to any graph reduction taking place. There is, however, a stark difference in the execution time of the method before graph reduction takes place and the execution time of the method after this combined technique was applied to the graph data. The execution time of the method using graph data reduced with the combined technique is substantially lower.

When the 90th percentile threshold is applied, Technique 2 is on its own also capable of reducing 51.61 % of non-zero attributes. The FDI method's performance once Technique 2 is applied to the graph data at a 90th percentile threshold is much weaker than the method's performance after the combined technique is applied to the graph data. From the FDI execution times recorded in Table 6.9, it is clear that the combined reduction technique far outperformed Technique 2 with a 90th percentile threshold. Therefore, the combined reduction technique is superior at reducing graph complexity and should be used instead of the individual complementary reduction techniques. The attributed graph diagram of the TEP after applying this combined technique can be seen in Figure 6.12.

Table 6.8: Performance of the eigendecomposition FDI method after applying the combined reduction technique.

	Unreduced Graph Data	Reduced Graph Data
Attribute Reduction (%)	0.00	51.61
Overall Detection Rate (%)	96.28	98.00
Overall Isolation Rate (%)	17.55	16.65
Isolation Rate Fault 1 (%)	90.00	91.00
Isolation Rate Fault 2 (%)	75.00	62.00
Isolation Rate Fault 6 (%)	100.00	98.00
FDI Execution Time (s)	61.94	24.54

Table 6.9: Comparison of the execution times of the eigendecomposition FDI method after applying different reduction techniques.

Reduction Type	Attribute Reduction (%)	Execution Time (s)
Combined	51.61	24.54
Technique 2	51.61	63.47

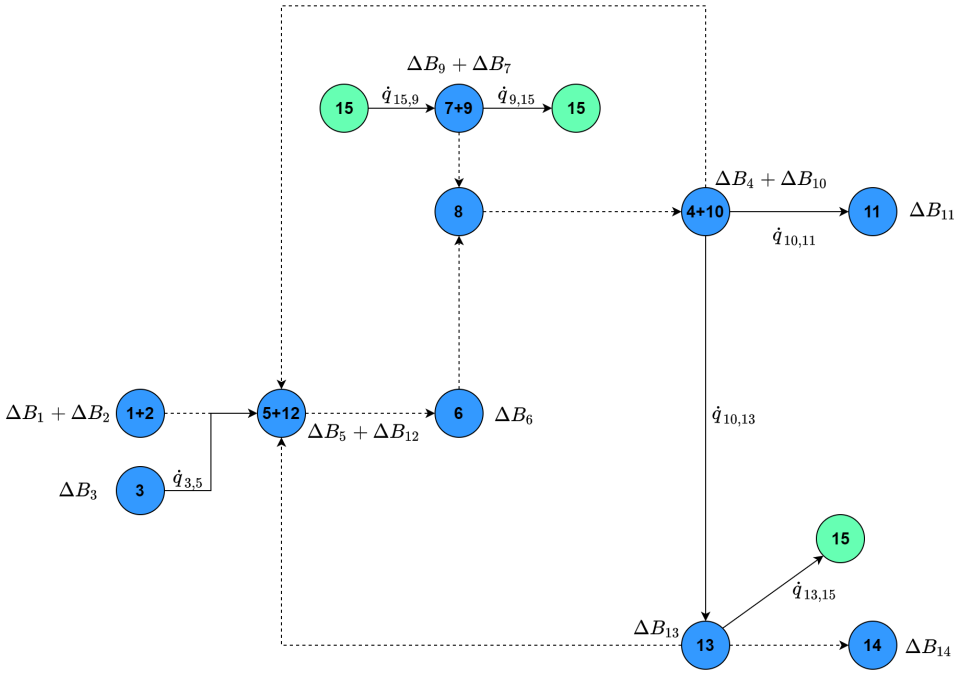


Figure 6.12: The attributed graph diagram after applying the combined reduction technique specific to the eigendecomposition FDI method.

6.4.3 Combining techniques for the residual-based FDI method

The combined technique applied to the graph data used by the residual-based FDI method is a combination of Technique 1 using the 10th percentile threshold, Technique 2 using a 60th percentile threshold, and Technique 5 summarizing four pairs of nodes. When Technique 1 is applied to the graph data with a 10th percentile threshold, node 8 and all its links are removed.

When Technique 2 is applied, the links between nodes 1 & 5, 2 & 5, 4 & 13, 5 & 1, 5 & 2, 5 & 6, 6 & 5, 6 & 7, 6 & 8, 7 & 6, 7 & 15, 8 & 6, 8 & 9, 8 & 10, 9 & 8, 10 & 8, and 13 & 4, 13 & 5, 13 & 14, and 14 & 13 are all set to zero. Technique 5 again summarizes nodes 4 & 10, 7

& 9, 1 & 2, and 5 & 12. This leads to a 51.61 % reduction in non-zero attributes.

A comparison of the performance of the residual-based FDI method using unreduced graph data and the performance of the method using graph data as reduced by this combined technique can be seen in Table 6.10. All but one of the indicators in the table experienced an enhanced performance once the FDI method was applied to the graph data as reduced by the combined technique. None of the individual complementary reduction techniques could reach a similar reduction percentage while resulting in the same level of FDI performance as the combined technique. The resulting attributed graph diagram of the TEP after applying this combined technique can be seen in Figure 6.13.

Table 6.10: Performance of the residual-based FDI method after applying the combined reduction technique.

	Unreduced Graph Data	Reduced Graph Data
Attribute Reduction (%)	0.00	51.61
Overall Detection Rate (%)	98.86	99.31
Overall Isolation Rate (%)	20.15	20.65
Isolation Rate Fault 1 (%)	80.00	93.00
Isolation Rate Fault 2 (%)	61.00	93.00
Isolation Rate Fault 6 (%)	88.00	86.00
FDI Execution Time (s)	5.36	3.09

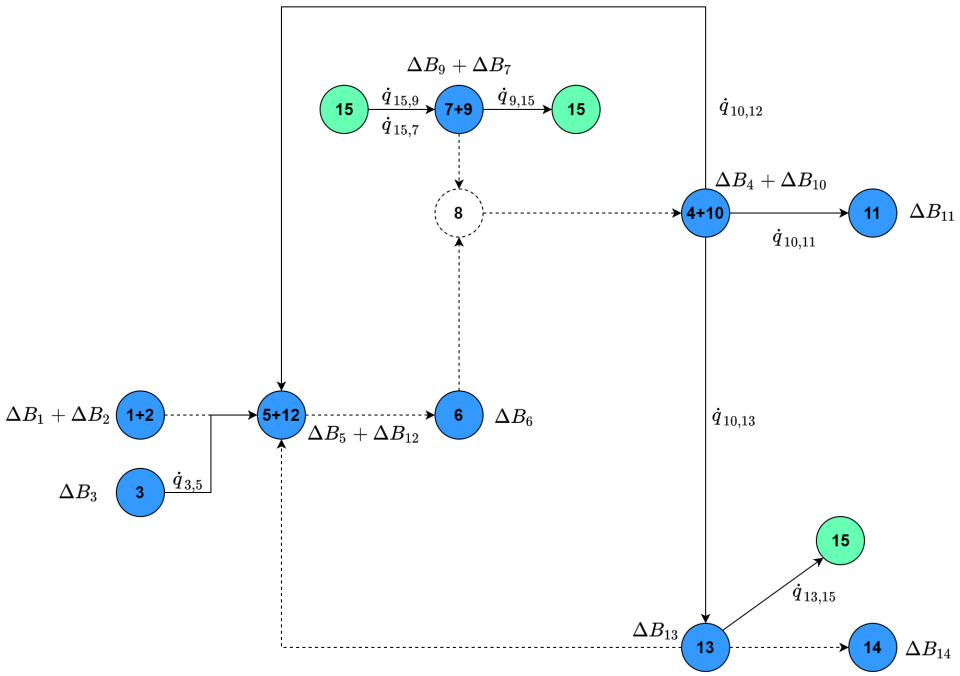


Figure 6.13: The resulting attributed graph diagram after applying the combined reduction technique specific to the residual-based FDI method.

6.5 Verification

The final section of this chapter focuses on the verification process of the proposed reduction techniques. The verification of the graph reduction techniques entails determining if each technique has been correctly implemented in MATLAB® according to that technique's original design by also manually implementing the technique and comparing the results. This is accomplished with Excel®. The Excel® environment requires each step of a technique's methodology to be implemented manually, which provides a great deal of user oversight. While the implementation of the techniques in Excel® is quite laborious, it ensures that the techniques are correctly implemented and that the attributed graphs are reduced as intended.

For each reduction technique, the technique is implemented with MATLAB® code which reduces the attributed graph data to a certain extent. The same technique is then implemented in Excel® which reduces the attributed graph to the same extent. Finally, the implementation of a technique is verified when the reduced graph produced by the MATLAB® code corresponds to the reduced graph produced with Excel®.

6.5.1 Verification of Technique 1

The reference attributed graph of all 21 process conditions is exported to Excel® where the percentage of physical exergy variation for each node over the 20 fault conditions is calculated. In addition, the percentage of chemical exergy variation for each node over the 20 fault conditions is also calculated. Finally, the average percentage of physical exergy variation and the average percentage of chemical exergy variation over the 20 fault conditions are determined and summated for each node. This produces a summated variation vector which can then be divided into percentiles.

The Excel® function 'PERCENTILE.EXC()' is applied to the summated variation values to determine the 30th percentile threshold. All the nodes with summated variation values less than the 30th percentile threshold are removed from the attributed graph data. The MATLAB® code is also used to reduce nodes with summated variation values below the 30th percentile threshold, to compare the resulting reduced attributed graphs.

Table 6.11 contains the NSM, as reduced by Technique 1 in Excel®, of the reference attributed graph of the process under NOC. Table 6.12 contains the NSM, as reduced by Technique 1 in MATLAB®, of the reference attributed graph of the process under NOC. Since these reduced matrices are identical, it is verified that Technique 1 was correctly implemented in MATLAB®.

For the sake of brevity, the verification of the remaining reduction techniques can be found in Appendix C.

Table 6.11: Reduced attributed graph produced by implementing Technique 1 in Excel.

-5.36E+07	-2.89E+05	0.00	0.00	0.00	5.50E+05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-7.77E+05	-2.51E+04	0.00	0.00	0.00	3.89E+03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-5.16E+07	-9.70E+05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.55E+05	0.00	0.00
-3.04E+08	-3.90E+07	-5.50E+05	-3.89E+03	0.00	0.00	4.92E+06	0.00	0.00	0.00	-3.51E+06	-5.85E+05	0.00	0.00
6.42E+07	7.37E+06	0.00	0.00	0.00	-4.92E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	-3.43E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.19E+08
-5.94E+07	6.40E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	3.93E+04	3.51E+06	8.45E+05	0.00	0.00
3.82E+06	3.85E+05	0.00	0.00	0.00	0.00	0.00	0.00	-3.93E+04	0.00	0.00	0.00	0.00	0.00
-3.05E+08	-3.07E+07	0.00	0.00	0.00	3.51E+06	0.00	0.00	-3.93E+04	0.00	0.00	0.00	0.00	0.00
-1.45E+07	-6.85E+06	0.00	0.00	-1.55E+05	4.92E+06	0.00	0.00	-8.45E+05	0.00	0.00	0.00	6.48E+05	-1.43E+05
1.69E+08	2.56E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-6.48E+05	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	6.18E+07	0.00	0.00	0.00	1.43E+05	0.00	0.00

Table 6.12: Reduce attributed graph produced by implementing Technique 1 in Matlab.

-5.36E+07	-2.89E+05	0.00	0.00	0.00	5.50E+05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-7.77E+05	-2.51E+04	0.00	0.00	0.00	3.89E+03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-5.16E+07	-9.70E+05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.55E+05	0.00	0.00
-3.04E+08	-3.90E+07	-5.50E+05	-3.89E+03	0.00	0.00	4.92E+06	0.00	0.00	0.00	-3.51E+06	-5.85E+05	0.00	0.00
6.42E+07	7.37E+06	0.00	0.00	0.00	-4.92E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	-3.43E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.19E+08
-5.94E+07	6.40E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	3.93E+04	3.51E+06	8.45E+05	0.00	0.00
3.82E+06	3.85E+05	0.00	0.00	0.00	0.00	0.00	0.00	-3.93E+04	0.00	0.00	0.00	0.00	0.00
-3.05E+08	-3.07E+07	0.00	0.00	0.00	3.51E+06	0.00	0.00	-3.93E+04	0.00	0.00	0.00	0.00	0.00
-1.45E+07	-6.85E+06	0.00	0.00	-1.55E+05	4.92E+06	0.00	0.00	-8.45E+05	0.00	0.00	0.00	6.48E+05	-1.43E+05
1.69E+08	2.56E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-6.48E+05	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	6.18E+07	0.00	0.00	0.00	1.43E+05	0.00	0.00

While verifying Technique 2, a discrepancy was noticed between the graph reduced with MATLAB® and the graph reduced with Excel®. This could be ascribed to the incorrect application of an absolute value function in the MATLAB® code. After correcting this, the graphs reduced with MATLAB® matched all the corresponding graphs reduced with Excel®. Thus, the implementation of all five graph reduction techniques has been verified. Therefore, the verification mechanism used for this study could ensure that all the reduction techniques were implemented as initially designed.

6.6 Conclusion

This chapter provides the detailed methodology of each of the five proposed graph reduction techniques. The effect each technique ultimately has on the performance of the three FDI methods is evaluated by conducting an experimental process that increases the number of attributes reduced by each technique and then measures the subsequent performance of the FDI methods. From the experiment results, it could be determined which FDI method responds better to which reduction technique. The results also confirmed that it is possible to reduce the complexity of the attributed graph data and maintain the level of performance achieved before reduction.

It was found that by combining individual complementary reduction techniques, the performance and/or execution time of all three FDI methods improved relative to the FDI performance obtained when the complementary techniques were applied individually. After applying the respective combination techniques, the performance of FDI methods was maintained, and in some cases, even improved upon the level of performance achieved prior to any graph reduction taking place. The implementation of all five graph reduction techniques in MATLAB® was successfully verified by implementing all the techniques in the Excel® environment.

Chapter 7

Validation of graph reduction techniques

7.1 Introduction

The purpose of this chapter is to validate graph reduction as a concept when it is applied to attributed graph data used by FDI schemes to diagnose faults. The validation is done by determining if the graph reduction techniques proposed in this study are general solutions capable of reducing the graph data while maintaining the same level of FDI performance, regardless of the process used to generate the graph data.

To this end, the graph data of a gas-to-liquids process is generated, and the reduction techniques are applied to the data. The performance of each FDI method after applying these techniques is then evaluated to determine if the successes achieved with the TEP are achieved when using this gas-to-liquids process.

7.2 Gas-to-liquids process overview

The gas-to-liquids process (GTLP) is a chemical process that converts natural gas into synthetic crude, which can, in turn, be upgraded and separated into various types of hydrocarbon fractions. The three core processing sections of the GTLP can be seen in Figure 7.1 (Adapted from [58]). In the first section, feedstock in the form of a gas is converted into synthesis gas. The second section uses the Fischer-Tropsch (FT) reaction to convert the synthesis gas into hydrocarbon products in liquid form. The final section upgrades the products to the suitable

specifications by using cracking and hydro-processing [58], [59].

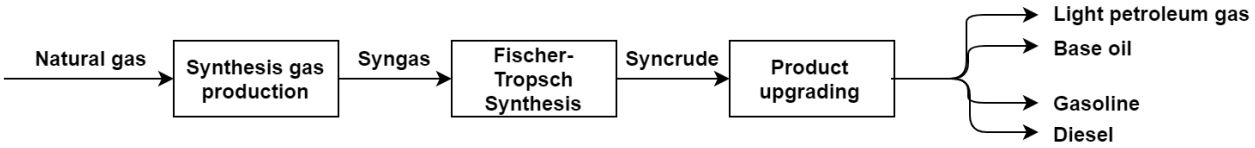


Figure 7.1: The main processing sections of a GTLP.

7.2.1 Synthesis gas production

The synthesis gas, also referred to as syngas, is a mixture of hydrogen (H_2) and carbon monoxide (CO) and chemical reforming is usually employed to produce it. The syngas production section generally requires pre-reforming, reforming, and cleaning or conditioning [8]. Pre-reforming is used to prevent the heavier hydrocarbon products in the first section from cracking [60].

After pre-reforming is employed, a reforming process, which is the primary mechanism for producing syngas, is used [8]. Syngas then usually undergoes a cleaning process to remove sulphur and nitrogen-containing compounds since these compounds degrade the catalysts of the FT reactor. If required, the syngas may also undergo conditioning whereby aspects of the syngas, such as composition or temperature, are altered [59], [61].

7.2.2 Fischer-Tropsch synthesis

The syngas product is fed into the FT reactor, where a catalytic transformation occurs to produce syncrude. There exist several different variations of the FT technology used to convert the syngas to syncrude. The syncrude produced is influenced by the reactor type, catalyst, and operating conditions of the FT technology used for this synthesis [8]. The composition of the syncrude can thus be customized to certain specifications by varying the combination of these three aspects used in the FT technology [59], [61].

7.2.3 Product upgrading

The last step of the GTLP is the syncrude-to-product conversion which is done by upgrading or refining the syncrude [8]. There are generally three types of syncrude-to-product conversion, with the conversion type being determined by whether the products obtained are final

or intermediate [61]. With upgrading, all the products must be refined to obtain the final products (Intermediate). With partial refining, some products have to be refined while others have to be blended to obtain the final products (Intermediate and Final). Finally, with stand-alone refining, all the products that are obtained are final products (Final).

7.3 GTLP model overview

The GTLP model used in this study is the model developed by Greyling in [8]. In that study, Greyling provides a detailed explanation of how the model works and the assumptions made to develop the model. Since this study is more concerned with the data produced by the model than the development thereof, only a short overview of the model will be provided. The interested reader is referred to [8] for more details.

7.3.1 Modelling assumptions

Aspen HYSYS® is the software package used to simulate the GTLP. For practical reasons, several assumptions were made while implementing the GTLP model in HYSYS®. Natural gas is primarily made up of pure methane (CH_4) but usually contains impurities [8]. For the sake of simplicity, pure methane is used as feedstock to negate the need for pre-reforming. All the other feedstock inputs are used in their pure form.

The GTLP model only contains the reformer used for syngas production, the reactor used for the Fischer-Tropsch synthesis, and the recycling of any unreacted gas back to the FT reactor. The upgrading section is too complex for the scope of the study conducted by Greyling and is, therefore, not considered. An Equation of State (EOS) thermodynamic package is used to create the model, seeing as this package is the most fitting package for dealing with hydrocarbons. The thermodynamic package is the foundation of determining the physical properties of the components and mixtures in the process as functions of pressure and temperature.

When considering noise, the two most prominent sources of noise present in literature are numerical and sensor noise. Numerical noise usually occurs in simulation models which use differential equations, seeing as these equations are adaptive and do not necessarily converge [62]. Sensor noise occurs when the output of the sensor deviates without the measurand changing, but since this is a simulation model, it can be assumed that sensor noise is not present [8]. Greyling noted that the numerical noise observed in HYSYS®, was of such a

small scale that it had a negligible effect.

7.3.2 Modelled process flow

In [8], Greyling implements an autothermal reformer (ATR) since it offers several advantages for single process streams and is suitable to be used in a GTLP fed by natural gas. Natural gas, steam, oxygen, and carbon dioxide (CO_2) are fed to the ATR in specific ratios to produce syngas with the desired temperature and composition. The syngas is then cooled and cleaned by separating some of the water. The syngas is then heated up substantially before being fed into the FT reactor. The FT reactor then produces syncrude in the form of various types of hydrocarbons. The syncrude is then cooled and sent to the 3-phase separator, which produces vapour, light liquid, and heavy liquid products. Finally, the unreacted products are recycled back to undergo the process again, while the liquid products are sent to for upgrading.

In order to cool the syngas produced by the ATR to an acceptable temperature before applying FT synthesis, a cooler was used. This cooler resulted in the steam being converted to water. A separator was then used to remove the water from the process. The cooling process produces heat which is returned to the environment for the sake of simulation simplicity. The simulation also includes a heater to heat the clean syngas before feeding it to the FT reactor.

To validate the model, Greyling evaluated the distribution of the product before including any recycling in the simulation. This evaluation is concerned with the weight fractions of the components after exiting the FT reactor. First, the logarithm values of all the weight fractions divided by their corresponding carbon numbers were calculated and plotted. This was done for both the theoretical and modelled products. Next, the slopes of both the theoretical and modelled plots were calculated. The theoretical and modelled slopes only differed with 1.2 %, which is a slight enough deviation for the simulation model to be deemed a valid representation of the theoretical model.

The completed GTLP model as developed by Greyling [8] in Aspen HYSYS® can be seen in Figure 7.2. The FDI methods used in Chapter 4 will be applied to the process data generated by this model.

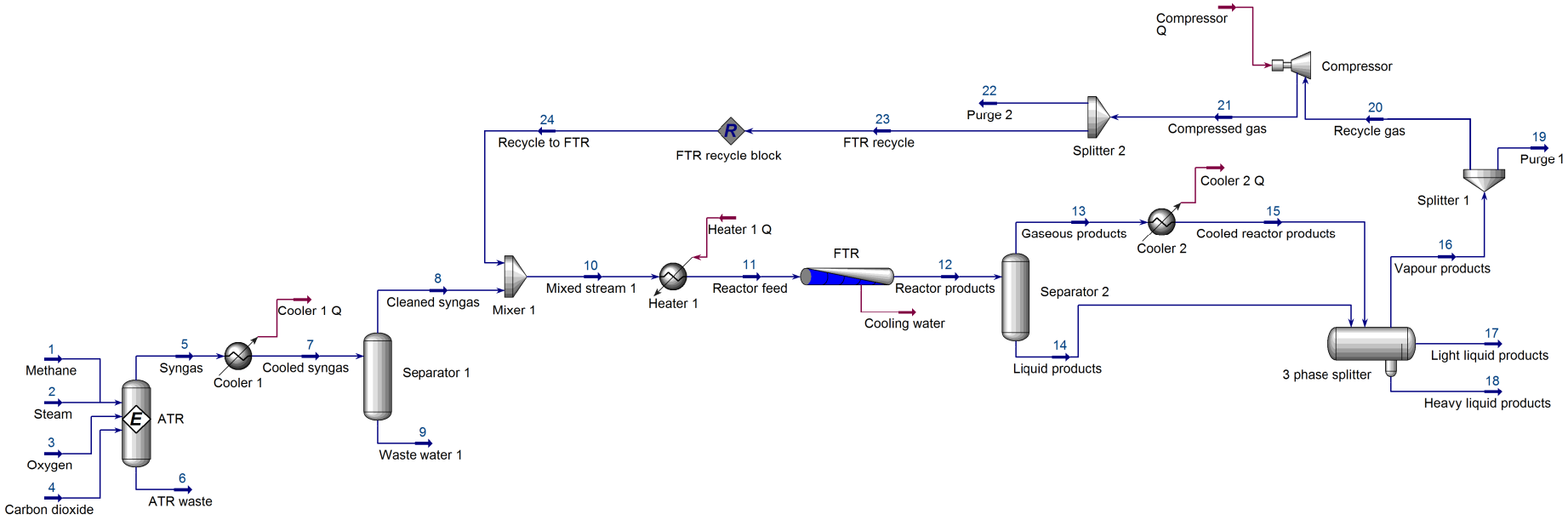


Figure 7.2: Simulation model of the GTLP as developed by Greyling in Aspen HYSYS®.

7.4 Fault conditions

7.4.1 Selecting faults

The fault conditions selected to evaluate the FDI methods were chosen by Greyling [8] by first identifying the most vital process units of the GTLP. After reviewing the relevant literature, possible process failures were identified, and the causes of these possible failures were then used to define fault conditions. For a specific fault condition to be selected, it should also be possible to emulate that condition within HYSYS®.

7.4.2 Fault sets

To compile the fault sets, the GTLP is divided into its three main processing sections: the syngas production section, Fischer-Tropsch section, and the recycling section. For each section, four-fault conditions were induced, resulting in 12 fault conditions being induced. In addition, several variations of fault condition magnitudes were also induced.

When considering the fault conditions occurring in the syngas production section, it can be observed that there are only two fault types at play: molar flow and pressure faults. Faults F_{11} and F_{12} both represent deviations in the feed molar flow rate of the methane stream while differing in the direction of the deviation. Faults F_{13} and F_{14} both represent low pressure in the methane stream and ATR, respectively.

In the Fischer-Tropsch section, the reactor feed stream can either have a low temperature (F_{21}), or damaged pipes can cause the feed stream to leak (F_{22}). Damaged pipes could also result in leakages within the FT reactor (F_{23}), or problems in the cooling unit of the reactor could result in increased reaction temperatures (F_{24}).

In the recycling section, the recycling compressor can degrade over time, resulting in a loss of pressure (F_{31}). A blockage in splitter one would result in less gas being recycled, which will increase the volume of the purge gas (F_{32}). A blockage in splitter two would have the opposite effect, resulting in more gas being recycled back (F_{33}). Damaged pipes could also result in leakages within the recycling stream (F_{34}).

Table 7.1 contains a summary of the 12 fault conditions, as well as the description and location of each condition. Figure 7.3 shows the locations of all 12 fault conditions. The conditions are indicated with yellow error triangles. The figure also shows which fault conditions have

the same location within the GTLP.

7.5 Exergy-based graph of the GTLP

The procedure for developing the exergy-based attributed graph of the GTLP remains the same as the procedure used to develop the attributed graph of the TEP. The components or input/output streams are modelled as nodes, while the streams between components are modelled as links. Node attributes represent the change in exergy over the representative component or input/output stream, and link attributes represent the energy flow rate between the different components (nodes).

The GTLP also has both physical and chemical exergy. All the changes in exergy and energy flow rate values were calculated within HYSYS®. The NSM of the GTLP is constructed in the same fashion as that of the TEP, with the node attributes (both physical and chemical exergy values) being recorded in the first two columns and the energy flow rate values being recorded in the remaining columns of the matrix. Since the GTLP has 18 nodes, the NSM has 18 rows and 20 columns.

Figure 7.4 illustrates the attributed graph of the GTLP as developed by Greyling [8]. Table 7.2 contains a summary of which nodes represent which process components or input/output streams. Unlike the simulation model of the TEP, the model of the GTLP does not model environmental exchanges, and the environmental nodes are, therefore, omitted from the attributed graph.

Table 7.1: Summary of the simulated 12 fault conditions of the GTLP.

Fault #	Fault ID	Location	Description
Syngas production section			
Fault 1	F_{11}	Methane stream	Molar flow
Fault 2	F_{12}	Methane stream	Molar flow
Fault 3	F_{13}	Methane stream	Pressure
Fault 4	F_{14}	ATR	Pressure
Fischer-Tropsch section			
Fault 5	F_{21}	Reactor feed stream	Temperature
Fault 6	F_{22}	Reactor feed stream	Leakage
Fault 7	F_{23}	FT reactor	Pressure
Fault 8	F_{24}	FT reactor	Temperature
Recycling section			
Fault 9	F_{31}	Compressor	Pressure
Fault 10	F_{32}	Splitter 1	Lower split rate
Fault 11	F_{33}	Splitter 2	Higher split rate
Fault 12	F_{34}	Recycle to FTR	Leakage

7.6 Applying the FDI methods to the GTLP

Just like in Chapter 4, the FDI control data of the GTLP is required to evaluate the performance of the graph reduction techniques once they are applied to the attributed graph data of the GTLP.

Unlike the TEP, the GTLP simulation is static and, therefore, generates steady-state data. As a result, measured process data is not generated by sampling over time but rather by using these steady-state measurements. Also, unlike the TEP model, it is possible to simulate different magnitudes of the fault conditions. In the case of the TEP, all 21 process conditions were sampled 501 times over the 25 hours to produce 10521 samples (10521 node signature matrices). The 21 reference attributed graphs were generated by calculating the average of the 501 samples for each process condition.

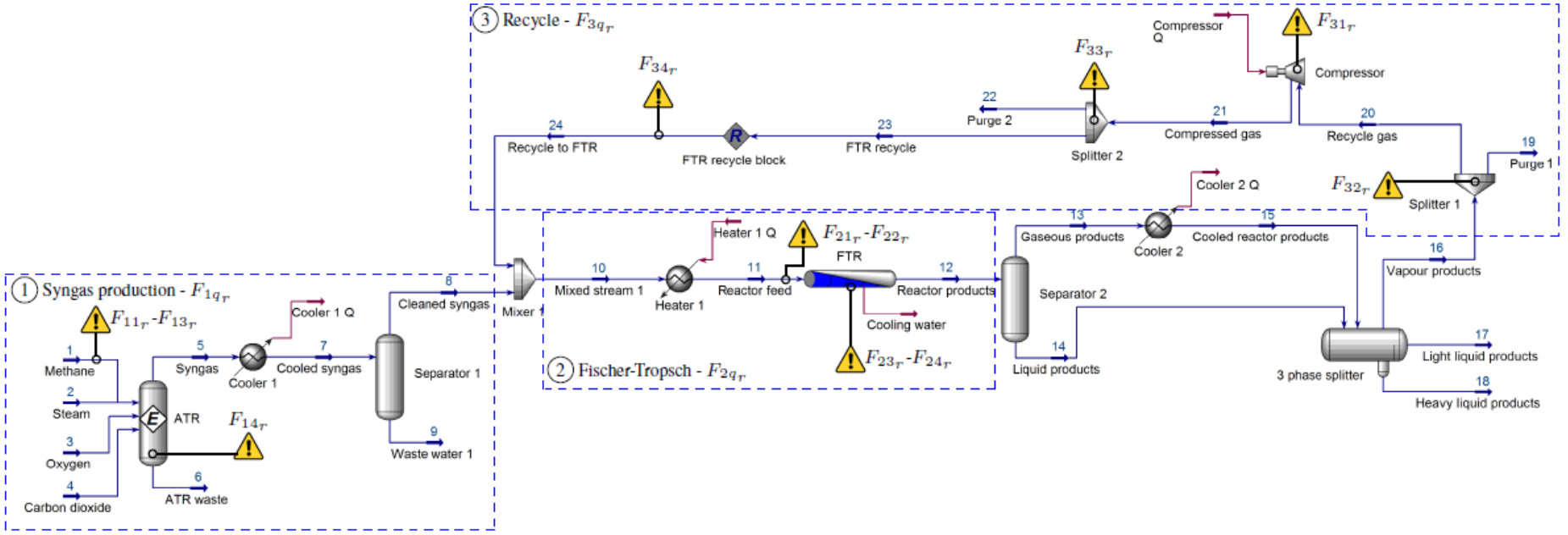


Figure 7.3: Illustration of fault locations within the GTLP.

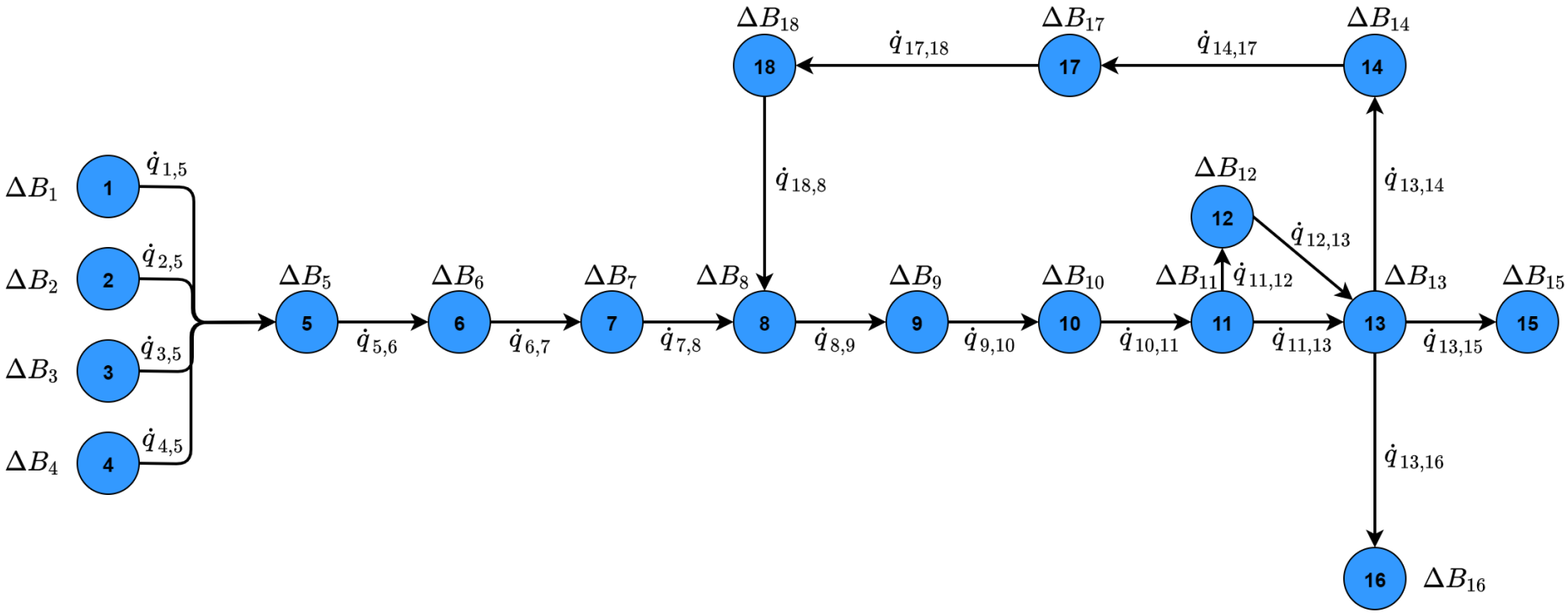


Figure 7.4: Exergy-based attributed graph of the GTLP.

Table 7.2: Summary of the components and input/output streams representing each node.

Node number	1	2	3	4	5	6	7	8	9
Process component	-	-	-	-	ATR	Cooler 1	Separator 1	Mixer 1	Heater 1
Input/output stream	Methane	Steam	Oxygen	Carbon dioxide	-	-	-	-	-
Node number	10	11	12	13	14	15	16	17	18
Process component	FTR	Separator 2	Cooler 2	3 phase separator	Splitter 1	-	-	Compressor	Splitter 2
Input/output stream	-	-	-	-	-	Light liquids	Heavy liquids	-	-

The GTLP, in turn, has 12 fault conditions and, together with the normal condition, has 13 process conditions in total. The reference attributed graph data for all 13 conditions is generated by inducing each fault condition with a 10 % magnitude, as done by Greyling in [8]. The reference normal attributed graph is simply the steady-state attributed graph while no fault is present. To generate the operational attributed graph data of each fault condition, the induced fault is adjusted to different magnitudes, and the steady-state attributed graph is recorded for each magnitude. These magnitudes are 5 %, 8 %, 12 %, and 25 %.

With the TEP, each fault condition has 501 operational attributed graphs, while in the case of the GTLP, each fault condition has four operational attributed graphs representing the four different fault magnitudes. The attributed graph of the process in NOC remains the same, irrespective of the different magnitudes. The four normal operational attributed graphs are, thus, all the same as the reference normal attributed graph. The difference in the number of graphs for each condition in the TEP and GTLP datasets (501 vs four graphs) prevent the direct comparison of TEP FDI performance with GTLP FDI performance. This is, however, no problem since the study aims to compare the response graph reduction has on the FDI performance of each process and not the FDI performance itself.

The rationale behind using this configuration for the GTLP dataset is to mimic the data configuration used by the FDI methods when applied to the TEP data. Using the attributed graph data at different fault magnitudes, the operational attributed graph data of the GTLP mirrors that of the TEP, which includes sampled measurements as the fault transitions from a minimal magnitude to its rated magnitude. Using the graph data generated at a relatively central fault magnitude as the reference graph data of the GTLP, an effect similar to using the averages of the sampled measurements to construct the reference graph data of the TEP, is achieved.

Seeing as the attributed graph of the GTLP has 18 nodes, the cost matrices generated by the distance parameter and eigendecomposition FDI methods will be an 18×18 matrix, which will produce 18 eigenvalues. The eigendecomposition FDI method is once again modified according to the study conducted by Wolmarans [52], whereby only dominant eigenvalues are used by the FDI method to diagnose faults. This is done to reduce the amount of information required by the FDI method to accurately diagnose faults, which reduces the complexity of implementing the FDI method.

Just like with the TEP, a trade-off is done between the number of dominant eigenvalues selected and the FDI performance by evaluating this performance using a range of different CPV values. Equation 4.4 is once again manipulated to determine the number of dominant eigenvalues for a specific CPV value. Table 7.3 contains the results of the trade-off study.

Using only five dominant eigenvalues (CPV of 75 %), the method achieved a higher overall isolation rate while maintaining a very high overall detection rate. The eigendecomposition FDI method is now modified to only consider the number of dominant eigenvalues with a CPV of 75 %. Any future references to the eigendecomposition FDI method refer to this modified version of the eigendecomposition FDI method.

Table 7.3: The performance of the eigendecomposition FDI method applied to the GTLP for different CPV values.

CPV (%)	Number of dominant eigenvalues	Overall detection rate (%)	Overall isolation rate (%)
75	5	95.83	45.83
80	7	97.92	43.75
85	9	97.92	43.75
90	11	97.92	50.00
100	18	97.92	43.75

Seeing as this chapter explains how the attributed graph data and the FDI methods differ from those applied to the TEP, only the results of each FDI method will be provided here. Table 7.4 contains a summary of the overall detection and isolation rates, as well as the isolation rates of specific fault conditions when the FDI methods are applied to the GTLP. A table containing the specific detection and isolation rates of each fault condition for each of the three FDI methods can be found in Appendix D.

In the case of the TEP, the isolation rates of specific fault conditions, which the FDI methods were highly capable of isolating, were also included in the control data to evaluate the graph reduction techniques. This was done to determine if the graph reduction techniques deteriorate the high-quality isolation capabilities of these specific faults. The same approach is used when generating the control data of the GTLP, except for the specific fault conditions used for the distance parameter FDI method. In the case of this FDI method, all but two fault conditions experienced the same high level of performance. Therefore, one high performing fault condition and the bottom two fault conditions are selected as specific fault conditions to determine if the graph reduction techniques could improve these two faults' capabilities.

The results, as captured in Table 7.4, now represent the control data that will be used to evaluate the graph reduction techniques when they are applied to the GTLP.

Table 7.4: Summary of the detection and isolation performance of the FDI methods applied to the GTLP.

Distance parameter FDI method	
Overall detection rate (%)	87.50
Overall isolation rate (%)	70.83
Isolation of Fault 1 (%)	75.00
Isolation of Fault 10 (%)	50.00
Isolation of Fault 12 (%)	50.00
Eigendecomposition FDI method	
Overall detection rate (%)	95.83
Overall isolation rate (%)	45.83
Isolation of Fault 1 (%)	75.00
Isolation of Fault 5 (%)	75.00
Isolation of Fault 8 (%)	75.00
Residual-based FDI method	
Overall detection rate (%)	97.92
Overall isolation rate (%)	39.58
Isolation of Fault 1 (%)	75.00
Isolation of Fault 2 (%)	100.00
Isolation of Fault 11 (%)	100.00

7.7 Evaluating the graph reduction techniques on the GTLP graph data

The proposed graph reduction techniques now have to be evaluated on a different process to show that graph reduction as a concept works and that it is a general solution and not confined to only operating on a specific process. The methodology of each reduction technique, as outlined in Chapter 6, remains unchanged when the techniques are applied to the GTLP graph data. In the case of the TEP, a requirement was imposed that prevented the environmental node from being removed to preserve practical structural information. Seeing as the GTLP graph does not include environmental nodes, this restriction is not considered.

For the GTLP, the experimental evaluation is repeated, whereby the extent to which the reduction techniques reduce attributes is incrementally increased, and the FDI performance is recorded. Just like in Chapter 6, this experimental procedure will first determine if it is possible to reduce graph attributes while maintaining the same level of FDI performance, and secondly, identify which reduction intervals remove structural information that confuses the specific FDI method.

The overall detection and overall isolation rates of each reduction technique as the reduction interval increases can be seen in Figures 7.5 - 7.14 below. The tables containing the overall detection and overall isolation rates, as well as the specific isolation rates for each reduction interval of every reduction technique, can be found in Appendix E.

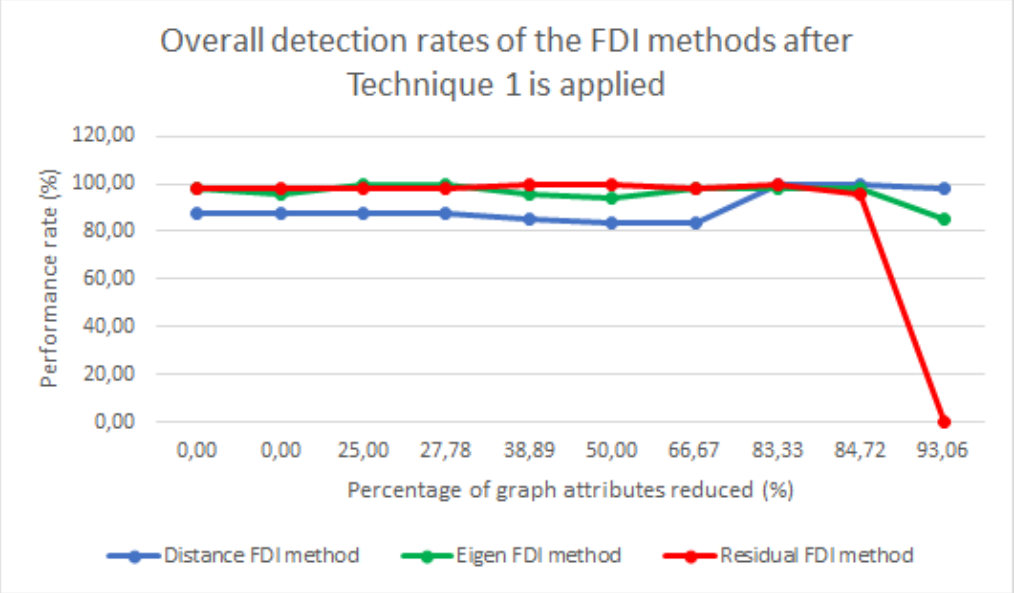


Figure 7.5: The overall detection rates after applying Technique 1 to the GTLP graph data.

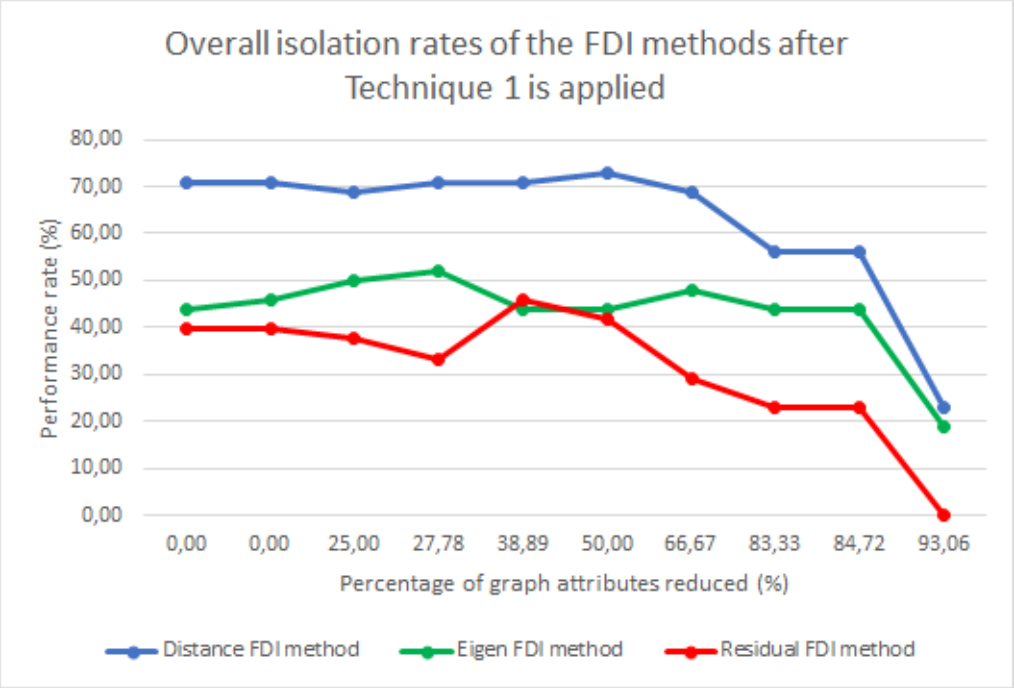


Figure 7.6: The overall isolation rates after applying Technique 1 to the GTLP graph data.

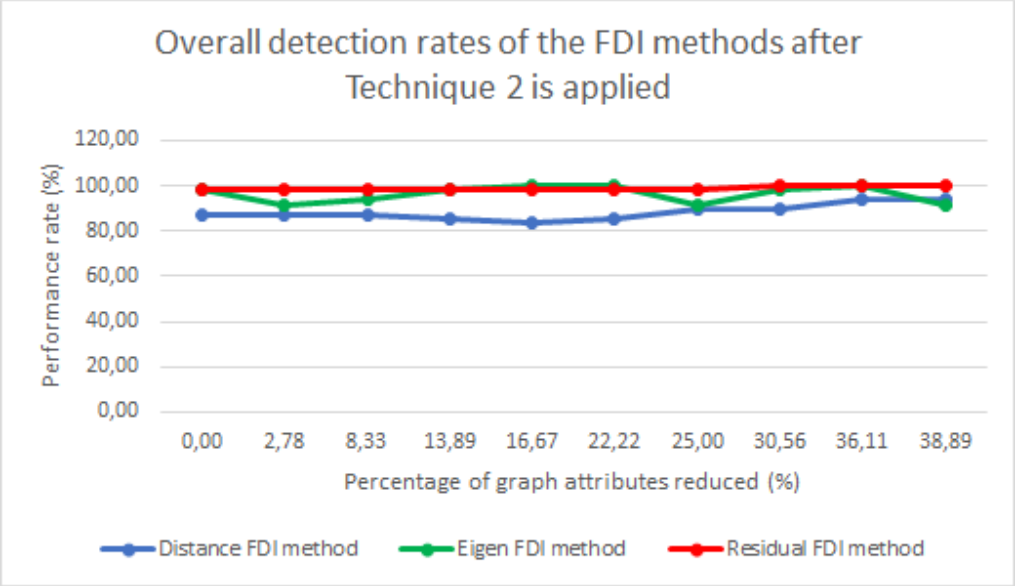


Figure 7.7: The overall detection rates after applying Technique 2 to the GTLP graph data.

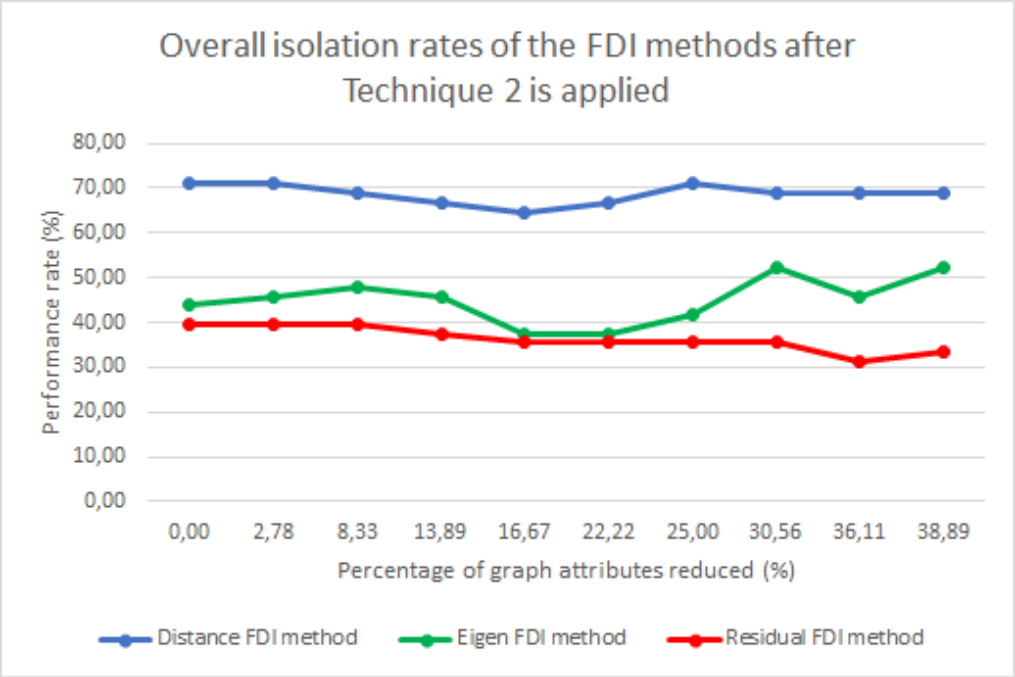


Figure 7.8: The overall isolation rates after applying Technique 2 to the GTLP graph data.

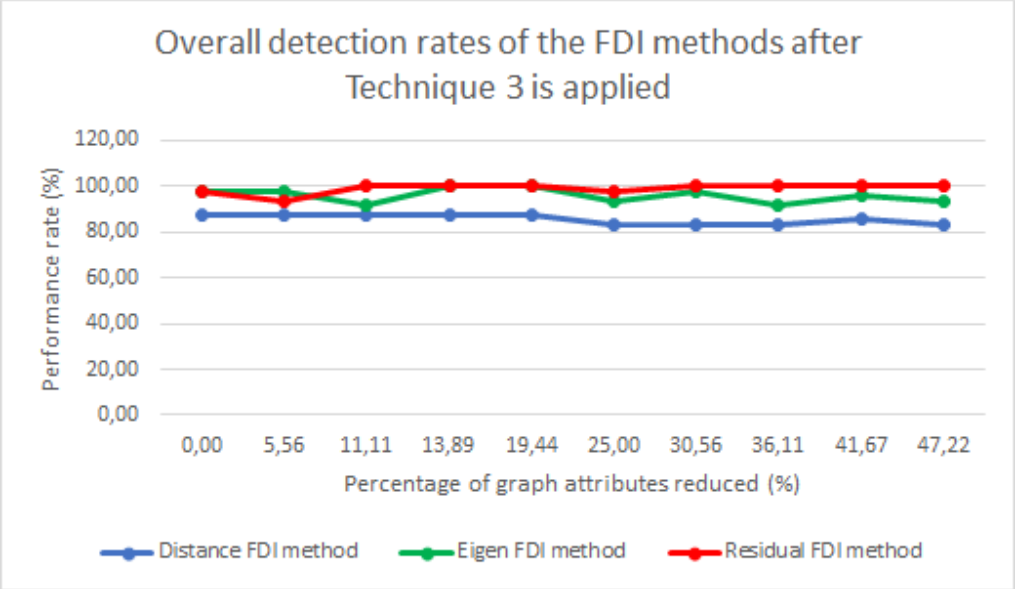


Figure 7.9: The overall detection rates after applying Technique 3 to the GTLP graph data.

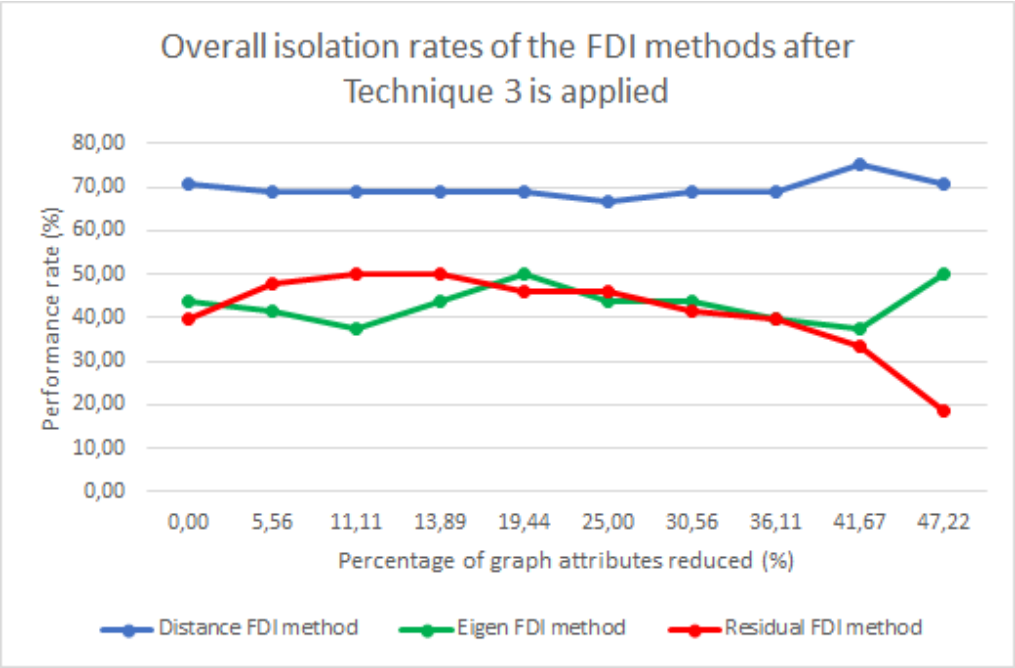


Figure 7.10: The overall isolation rates after applying Technique 3 to the GTLP graph data.

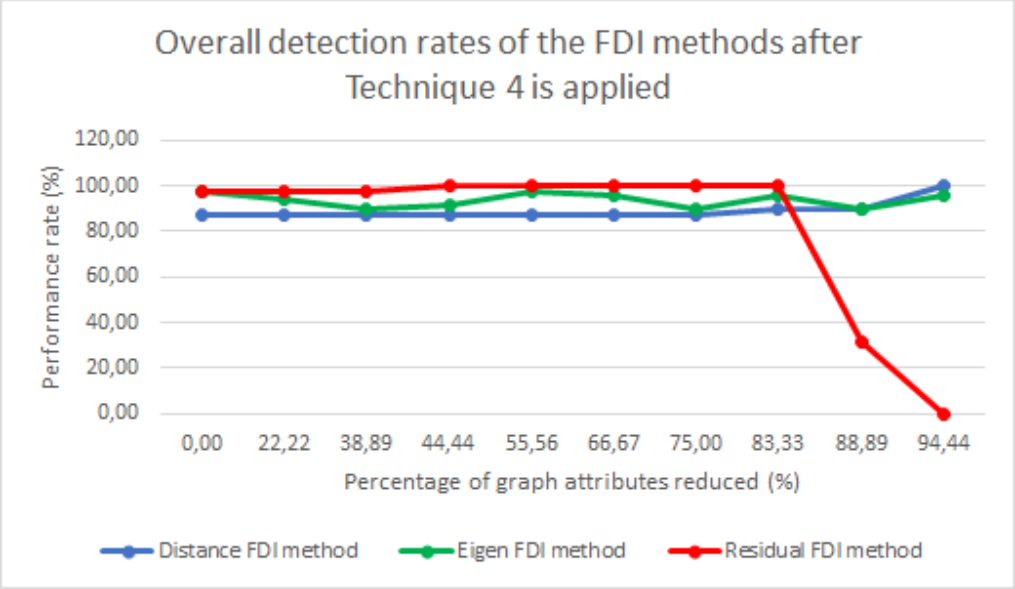


Figure 7.11: The overall detection rates after applying Technique 4 to the GTLP graph data.

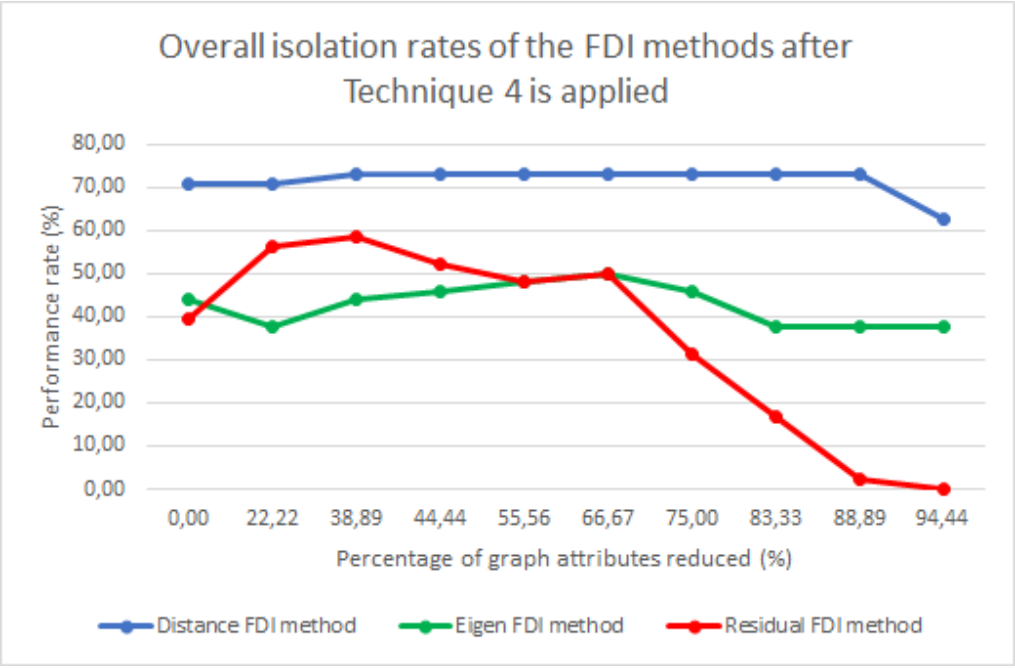


Figure 7.12: The overall isolation rates after applying Technique 4 to the GTLP graph data.

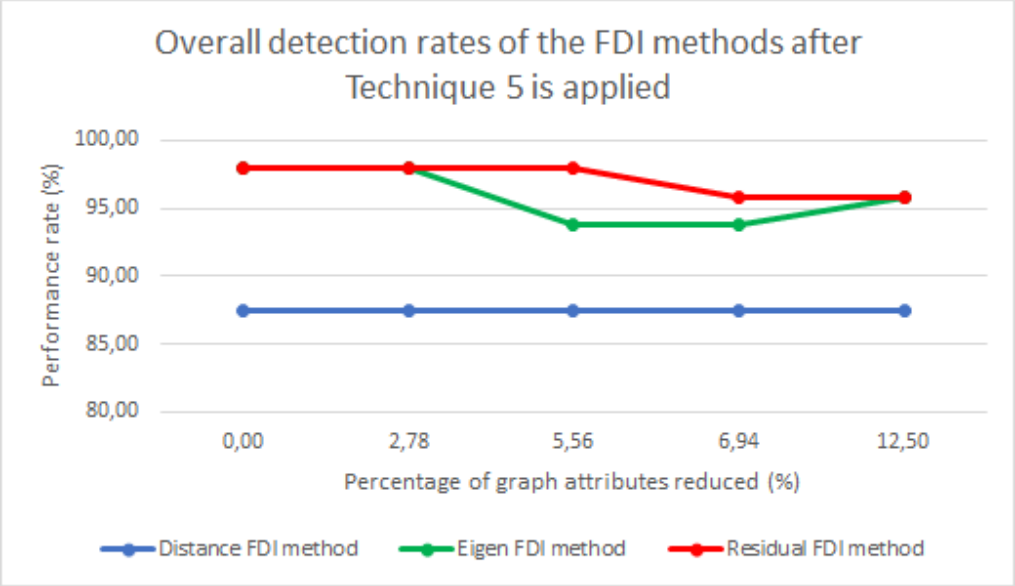


Figure 7.13: The overall detection rates after applying Technique 5 to the GTLP graph data.

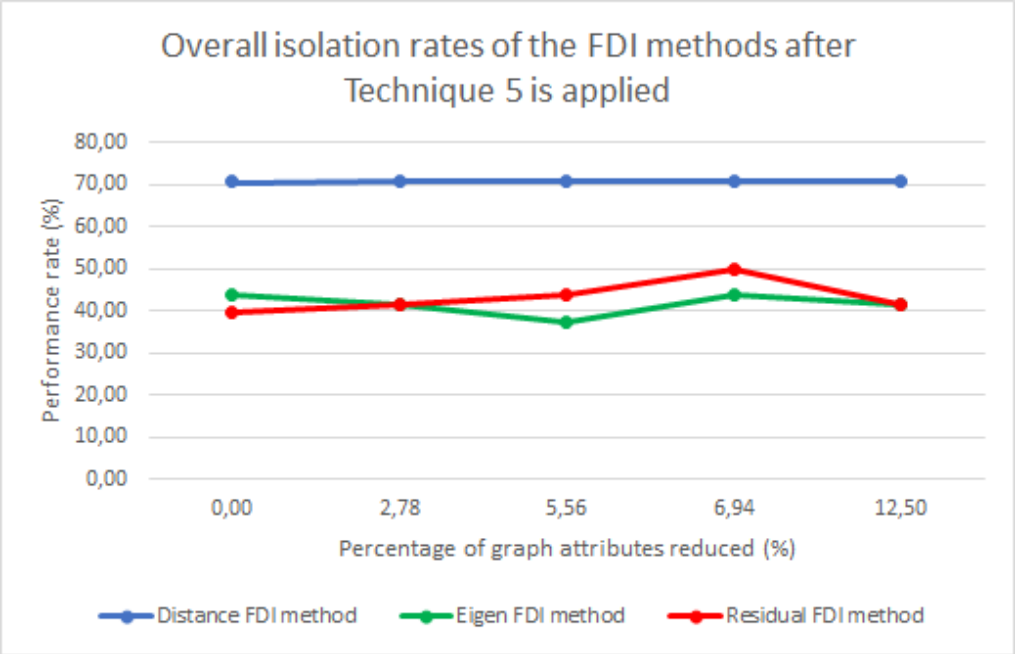


Figure 7.14: The overall isolation rates after applying Technique 5 to the GTLP graph data.

7.8 Observations from the results

The most important observation from these results is that the graph reduction techniques proposed in this study are all capable of reducing graph complexity by removing unnecessary

attributes from the graph data or reorganizing the structural information using summarization. Furthermore, these reduction operations can reduce the graph data so that the FDI methods maintain or improve on the level of performance achieved when unreduced graph data are used.

When comparing the results obtained from evaluating the reduction techniques on the TEP graph data with the results obtained from evaluating the techniques on the GTLP graph data, some stark contrasts can be observed. One of the most notable contrasts is that the overall isolation rates of the distance parameter and eigendecomposition FDI methods applied to the GTLP data experience a much more constant trajectory as reduction intervals of all the techniques increase. On the other hand, when looking at the overall isolation rates when these two methods are applied to the TEP data, the rates generally experience downward trajectories.

The overall detection rates of all three FDI methods applied to the GTLP data remained relatively constant for most reduction intervals. However, the reduction techniques identified as complementary to each FDI method applied to the GTLP data are not the same techniques identified as complementary to each FDI method applied to the TEP data. This can be ascribed to the fact that the structural composition of the two graphs differs, and an identical reduction operation may remove information that is non-critical to the FDI method from one graph while removing information that is critical to the method from the other. This results in an FDI method performing very differently on two different graphs reduced by the same technique. Therefore, the specific reduction intervals at which point each reduction technique is most effective are of interest.

To determine each FDI method's complementary reduction technique, the FDI method's overall detection rate, overall isolation rate, and specific fault isolation rates are evaluated using the control data as a benchmark. For example, when considering the residual-based FDI method, Technique 2 (using the 60th percentile threshold), Technique 4 (50th percentile), and Technique 5 (1 merger) could all maximally reduce the graph attribute data while minimally deteriorating FDI performance.

For the distance parameter FDI method, Technique 1 (50th percentile), Technique 2 (60th percentile), Technique 3 (80th percentile), Technique 4 (80th percentile), and Technique 5 (4 mergers) are all complementary reduction techniques as they are all capable of reducing attributes while maintaining and/or even improving on the results in the control data. Likewise, for the eigendecomposition FDI method, Technique 1 (70th percentile), Technique 2 (70th percentile), Technique 3 (40th percentile), and Technique 5 (3 mergers) are all capable of reducing the graph data while maintaining and/or even improving on the FDI performance

when unreduced graph data is used.

From analysing each FDI method's complementary reduction techniques, it is evident that the reduction techniques which rely on attribute size analysis (Techniques 3 & 4) are more effective when applied to the FDI methods using the GTLP attributed graph data than the FDI methods, which use the TEP attributed graph data. It is further evident that the reduction techniques that rely on attribute variation analysis (Techniques 1 & 2) are effective when applied to the attributed graph data of both the GTLP and the TEP.

Certain similarities can also be observed between the performance of FDI methods when reducing the TEP graph data and the performance of the methods when reducing GTLP graph data. The main similarity is the occurrence of anomalies in the form of local minima and maxima in FDI performance as more attributes are removed from the GTLP graph data. This reinforces the argument that specific attributes, which represent the structural information of the process, are vital to FDI performance (vital structural information) and bolster the performance while others are non-vital (non-vital structural information) and obfuscate FDI performance.

Also similar to the TEP, a reduction operation can result in the increase of the specific isolation rate of one fault while resulting in the deterioration of the rate of another specific fault at the same time. An example of this can be seen when the residual-based FDI method is reduced with Technique 1 using the 70th percentile threshold.

7.9 Applying combination reduction techniques to the GTLP

The effect of combining the individual complementary reduction techniques of each FDI method and applying these combined techniques to the attributed graph data is evaluated on the GTLP as well. This is done to determine if the advantages of combined reduction techniques are only applicable to specific processes like the TEP or applicable in general.

This evaluation revealed that, unlike with the TEP, at least one reduction operation performed by an individual complementary reduction technique resulted in a performance by each FDI method that could not be outdone with any combined technique. These reduction operations are Technique 4 with a 70th percentile threshold for the distance parameter FDI method, Technique 1 with a 70th percentile threshold for the eigendecomposition FDI method, and Technique 4 with a 70th percentile threshold for the residual-based FDI method.

However, it was evident from the evaluation that it is possible to improve the FDI performance resulting from individual complementary reduction techniques by combining these techniques, albeit not in all instances. Therefore, for each of the three FDI methods, complementary techniques were combined, and the resulting FDI performance was compared with the FDI performance resulting from those individual complementary techniques.

From the performance of the distance parameter FDI method recorded in Table 7.5, it can be seen that the combination of Technique 1 (50th percentile) and Technique 2 (60th percentile) resulted in improved specific isolation rates, percentage attributes reduced, as well as FDI execution time compared to the performance resulting from the complementary techniques being applied individually.

For the eigendecomposition FDI method (Table 7.6), the combination of Technique 3 (40th percentile) and Technique 5 (3 mergers) also yielded better specific isolation rates, attribute reduction, and execution time compared to that of the individual techniques. For the residual-based FDI method (Table 7.7), the combination of Technique 2 (60th percentile) and Technique 5 (1 merger) outperformed the individual reduction techniques in every metric used for the evaluation.

While no combined technique could outperform every individual complementary reduction technique for a given FDI method, there are instances where it is possible to increase the efficacy of individual reduction techniques by combining them. The reduced attributed graph diagrams resulting from the combined techniques in Tables 7.3, 7.5, and 7.6 can be seen in Figures 7.15, 7.16, and 7.17, respectively.

Table 7.5: A comparative evaluation of combined reduction techniques for the distance parameter FDI method.

	Technique 1 (50th percentile)	Technique 2 (60th percentile)	Combined Technique
Attribute Reduction (%)	50.00	25.00	61.11
Overall Detection Rate (%)	83.33	89.58	83.33
Overall Isolation Rate (%)	72.92	70.83	72.92
Isolation Rate Fault 1 (%)	75.00	75.00	75.00
Isolation Rate Fault 10 (%)	75.00	75.00	75.00
Isolation Rate Fault 12 (%)	75.00	50.00	75.00
FDI Execution Time (s)	0.86	6.84	0.82

Table 7.6: A comparative evaluation of combined reduction techniques for the eigendecomposition FDI method.

	Technique 3 (40th percentile)	Technique 5 (3 mergers)	Combined Technique
Attribute Reduction (%)	19.44	6.94	26.39
Overall Detection Rate (%)	100.00	93.75	91.67
Overall Isolation Rate (%)	50.00	43.75	47.92
Isolation Rate Fault 1 (%)	75.00	100.00	75.00
Isolation Rate Fault 5 (%)	75.00	75.00	100.00
Isolation Rate Fault 8 (%)	50.00	50.00	75.00
FDI Execution Time (s)	0.62	0.44	0.42

Table 7.7: A comparative evaluation of combined reduction techniques for the residual-based FDI method.

	Technique 2 (60th percentile)	Technique 5 (1 merger)	Combined Technique
Attribute Reduction (%)	25.00	2.78	27.78
Overall Detection Rate (%)	97.92	97.92	97.92
Overall Isolation Rate (%)	35.42	41.67	41.67
Isolation Rate Fault 1 (%)	75.00	75.00	100.00
Isolation Rate Fault 2 (%)	100.00	100.00	100.00
Isolation Rate Fault 11 (%)	100.00	100.00	100.00
FDI Execution Time (s)	0.35	0.33	0.29

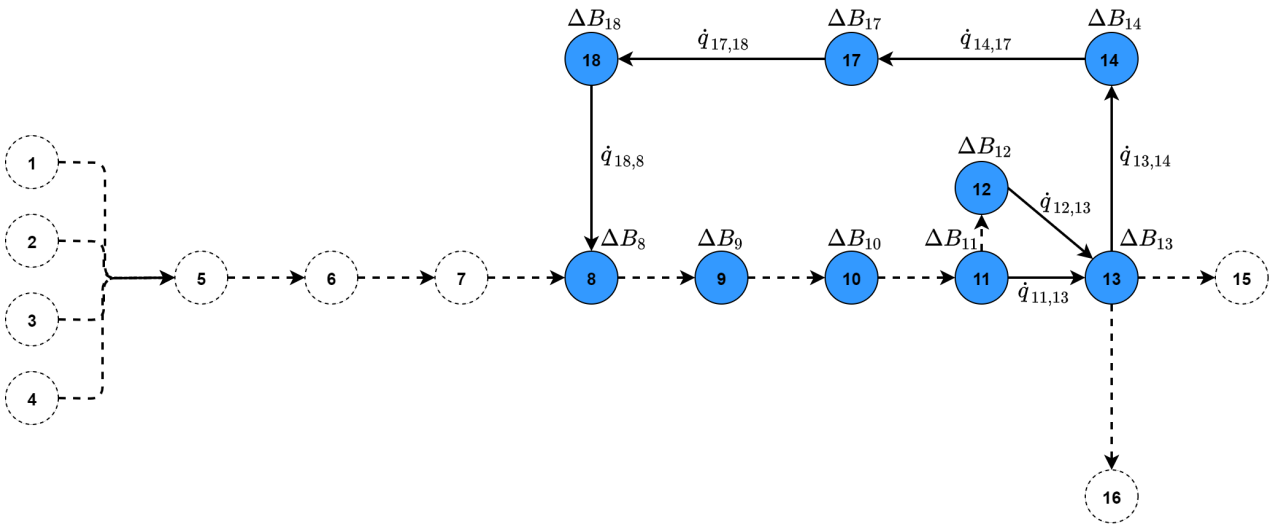


Figure 7.15: Attributed graph reduced by the combined reduction technique for the distance parameter FDI method.

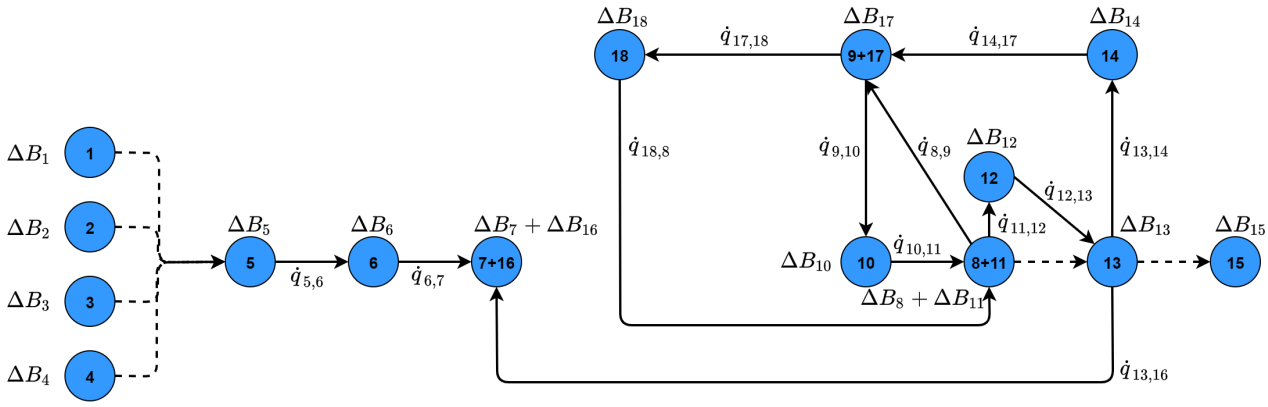


Figure 7.16: Attributed graph reduced by the combined reduction technique for the eigendecomposition FDI method.

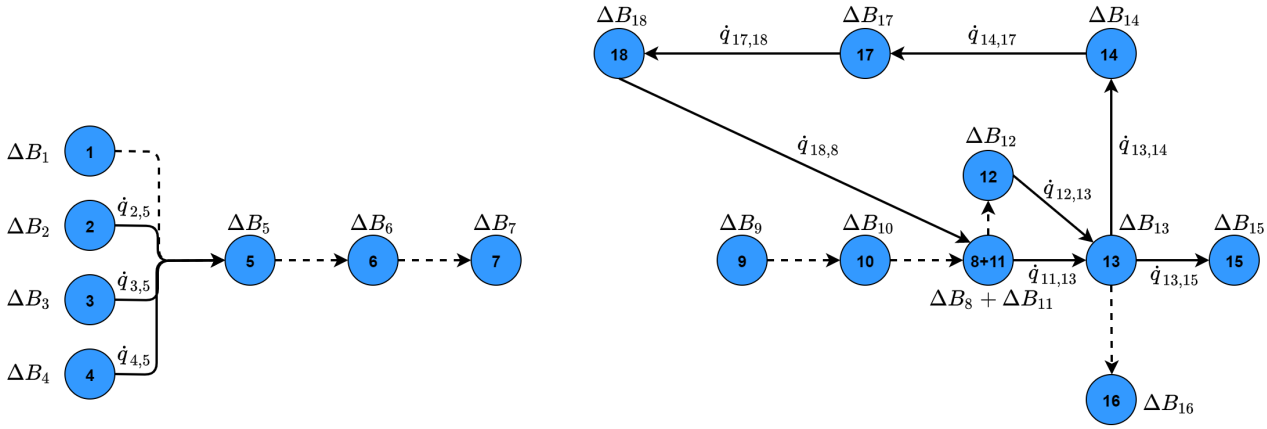


Figure 7.17: Attributed graph reduced by the combined reduction technique for the residual-based FDI method.

7.10 Validation

To validate this study, it is necessary to show that graph reduction can successfully be applied to the attributed graph data of any process as a general solution that is not restricted to a specific process. To accomplish this, the GTLP was used as a secondary process to the TEP in order to evaluate the effect of the proposed graph reduction techniques on FDI performance. By analysing the results of both Chapters 6 & 7, it is clear that the proposed reduction techniques are capable of reducing graph complexity by reducing attributes from the graph while maintaining the same level of FDI performance achieved prior to any graph reduction, regardless of the process data used.

While each proposed reduction technique did not affect the FDI methods in precisely the same manner when different processes were used, for both the TEP and GTLP, it was possible to formulate at least one reduction technique for each FDI method which resulted in similar FDI performance to that achieved prior to graph reduction. The concept of a general solution is further supported by the fact that the two processes differ substantially in terms of graph structure, fault conditions, and composition of graph data (dynamic vs steady-state system), as well as the fact that three different FDI methods were used for the evaluation of the techniques. It can, therefore, be concluded that graph reduction is a general solution and can be considered validated.

7.11 Conclusion

This chapter introduces the gas-to-liquids process (GTLP) model and provides a brief overview of how the model was created. Next, the differences in graph data between the TEP and GTLP are explained, after which the three FDI methods are applied to the GTLP graph data to generate the control data used to evaluate the reduction techniques. The method used to evaluate the graph reduction techniques applied to the graph data is the same as that of the TEP's graph data.

It is found that it is possible to reduce the attributed graph data of the GTLP while maintaining the level of FDI performance achieved before any reduction takes place. Furthermore, when combination techniques are evaluated, it is shown that while no combined technique could outperform the top individual technique for each FDI method, it is still possible to derive significant value from combined techniques. Lastly, it is shown that since graph reduction is a general solution, the study has been validated.

Chapter 8

Conclusion

8.1 Introduction

This chapter serves as the conclusion of this study. First, the concluding remarks are provided, which highlight the most important results and findings from this study. Then, the last section of this chapter contains several recommendations on how these results and findings can be used in future works.

8.2 Concluding remarks

The ultimate objective of this study was to identify and/or propose graph reduction techniques that can be used to reduce the complexity of the attributed graph data of the Tennessee Eastman process. These techniques had to reduce graph complexity while ensuring that the FDI methods applied to the TEP graph data retain the level of performance achieved before any reduction had taken place, thereby proving that graph reduction as a concept is a viable solution.

Before any form of graph reduction could take place, control data of the performance of the FDI methods when the standard, unreduced attributed graph data are used to diagnose faults, had to be generated. The control data are used to evaluate the efficacy of the reduction techniques. The distance parameter, eigendecomposition, and residual-based FDI methods are the three FDI methods used to diagnose faults from the TEP graph data. The eigendecomposition FDI method was updated according to the findings of a concurrent study to reduce the complexity of the FDI method by only considering dominant eigenvalues.

The indicators used to quantify the performance of each FDI method are the overall detection rate, overall isolation rate, and the specific isolation rate of critical fault conditions. These indicators were selected to facilitate a holistic evaluation while not overwhelming the evaluation process with excessive amounts of data. When looking at these indicators for the three FDI methods, it is clear that there is great variety in the performance indicators across the different methods, making it challenging to select a top-performing method for the TEP. However, this variety in performance indicators across the different methods in the control data set will contribute to a robust evaluation of graph reduction techniques.

The graph reduction techniques proposed for this study are based on three theoretical concepts. The first concept is that attributes experience slight variation from the NOC for all fault conditions, have a negligible impact on the FDI process and can be removed. The second concept is that attributes with small values contribute less to the FDI process and can be removed. The final concept is that graph complexity decreases by methodically summarizing nodes with similar attribute values with imposed limitations. This preserves attribute information.

From these three concepts, five graph reduction techniques were formulated. The first two concepts have a technique based on node attributes and a technique based on link attributes. The final concept has one technique based on node attributes only. The five techniques were formulated to remove structural information from the graph data without distorting the remaining structural information and resulting in additional processing resources being required. Furthermore, structural information is removed methodically, making it easier to reconstruct the original graph. This was done to overcome the shortfalls of the techniques identified in the literature.

An experimental process had to be used to determine if the proposed reduction techniques formulated from the three theoretical concepts could reduce graph complexity without deteriorating FDI performance. This experimental process involves applying each reduction technique iteratively and increasing the extent to which the technique reduces the TEP graph data with each iteration. After each iteration, the FDI performance indicators are recorded. This is done until there are no more attributes to reduce or the limitations and/or restrictions prevent the technique from reducing any further. Then, the performance indicators obtained after each reduction iteration were compared to the control data set indicators by plotting some of these indicators as more attributes were reduced. This was done to determine the efficacy of each technique.

Several interesting observations could be made from the performance of each FDI method's response to the reduction iterations of each reduction technique. However, the most important

observation that can be made is the fact that for every FDI method applied to the TEP graph data, it was possible to reduce the graph data with at least one reduction technique using a specific reduction iteration while maintaining a similar level of performance as achieved in the control data set.

Another important observation that could be made is the occurrence of anomalies in the form of local minima or maxima as the reduction interval increases. These anomalies indicate that there exists structural information in the attributed graph data that can be either vital or non-vital to FDI performance. This observation inferred that the downward part of either the local minima or maxima results from vital information that bolsters FDI performance, being removed from the graph data. Conversely, the upward part of either the local minima or maxima results from non-vital information that obfuscates FDI performance, being removed from the graph data.

To determine which reduction techniques are more effective than others, the complementary techniques of every FDI method were identified. Each complementary technique effectively reduces graph data while maintaining a similar level of FDI performance achieved in the control data set. For the TEP, the reduction techniques based on attribute variation (Techniques 1 & 2) were much more effective than those based on attribute size analysis (Techniques 3 & 4). Across all three FDI methods, Technique 5 resulted in the most stable response of FDI performance for its reduction range.

To bypass the shortcomings of some of the individual reduction techniques, it was proposed that specific individual techniques be combined by applying them to the graph data simultaneously. Thus, for each FDI method, that method's complementary reduction techniques at specific reduction intervals were applied simultaneously, and the standard performance indicators and the execution time of the FDI method were recorded. By doing this, it was found that for the TEP graph data, the performance and/or execution time of each FDI method improved relative to the performance and/or execution time achieved when the individual complementary reduction techniques were applied. The effect some of the individual and combined reduction techniques had on the execution time of FDI methods, can be observed in Sections 6.3 & 6.4. In some instances, the combined techniques resulted in a superior level of FDI performance than that achieved in the control data.

For the verification of this study, it had to be verified that all five reduction techniques were implemented as proposed. Therefore, each technique was first implemented in the MATLAB® to reduce an attributed graph to a certain extent. The technique was then implemented in Excel® to reduce the same attributed graph to the same extent. Finally, these two attributed graphs were compared to each other in their NSM format. Since both matrices were identical

for each technique, it can be concluded that the study correctly implemented and verified each proposed reduction technique.

The validation of this study entails showing that the proposed solution, in this case, the concept of graph reduction, is a general solution. This involves showing that at least one of the reduction techniques proposed in this study can reduce the attributed graph data of more than one validated process model while maintaining a similar level of FDI performance achieved prior to the graph data of those process models being reduced. To this end, a validated GTLP model was used as an additional model to evaluate the graph reduction techniques. The GTLP model differs from the TEP model with regards to the type of fault conditions, complexity and structure of the attributed graph, and the composition of graph data (steady-state model vs dynamic model). These differences help to enhance the generality of the solution.

To evaluate the proposed reduction techniques on the GTLP, the same chain of events used with the TEP was repeated. First, the control data set was generated by applying the FDI methods to the standard, unreduced attributed graph data of the GTLP. Then, each reduction technique's response on each FDI method's performance was evaluated using an experimental process that applies the techniques iteratively, increasing the reduction interval with each iteration. Thus, the performance of each FDI method was determined at every reduction increment, and the indicators of each FDI method at these various reduction intervals could be compared to the indicators in the control data set.

This evaluation yielded that it is possible to reduce the attributed graph data of the GTLP while maintaining the FDI performance achieved prior to any graph reduction taking place. Furthermore, since graph reduction could successfully be applied to both the TEP and the GTLP, which are both validated models, it can be concluded that graph reduction is a general solution, which ultimately validates this study. It should be stressed that this study does not aim to compare the performance of FDI methods using TEP data with the performance obtained using GTLP data. Instead, the aim is to investigate and compare the response graph reduction has on the TEP FDI performance with the response it has on GTLP FDI performance to determine if it is a viable and general solution.

When comparing the response on FDI performance after applying the reduction techniques to the GTLP graph data to the response after applying the techniques to the TEP graph data, anomalies in the form of local minima and maxima were again detected. This supports the argument that certain reduction operations remove vital structural information (causing the downward part of either the local minima or maxima). In contrast, certain other reduction operations remove non-vital structural information (causing the upward part of either the

local minima or maxima). This results in FDI performance being obfuscated or bolstered, respectively.

When comparing the efficacy of the five different techniques applied to the GTLP graph data, it can be noted that the techniques based on attribute size analysis (Techniques 3 & 4) were more effective when applied to the GTLP data than they were applied to the TEP data. However, seeing as the variance-based reduction techniques (Techniques 1 & 2) were effective when applied to the data of both models, it can be inferred that they are more effective in general than the techniques based on attribute size analysis.

Combining reduction techniques and applying them to the GTLP graph data did not yield the same successful response as achieved when techniques were combined and applied to the TEP graph data. While it could be shown that combining specific complementary techniques for each FDI method had the possibility of being more effective than some of the individual complementary techniques, an individual complementary technique at a specific reduction interval was identified for each FDI method that could not be outdone.

8.3 Recommendations on future work

The purpose of this section is to recommend possible improvements to graph reduction and to recommend potential future implementations of the findings of this study. These recommendations include:

- For this study, it was deliberately decided not to use optimization algorithms to reduce the attributed graph data since this would have produced an optimized solution. If this were the case, the observations about the effects that different types of reductions techniques have on the performance of the different FDI methods would have been lost. The observations from this study can now be used as guidelines for selecting an optimization algorithm and correctly formulating its cost function to reduce attributed graph data optimally.
- Both the TEP and GTLP are large and complex systems. It is, therefore, of particular interest to determine the effect of graph reduction of smaller systems with fewer fault conditions. In addition, by applying these reduction techniques to smaller systems, it would be possible to determine if any adjustments to the techniques are required for these smaller systems.
- Once graph reduction has successfully been applied to the attributed graph of a process; it will be possible to determine the most effective sensor placement. The sensors should

be deployed so that they collect the majority of process data from components and streams that remain in the graph after reduction. This will also assist in reducing the cost of implementing FDI schemes.

Bibliography

[1] D. Himmelblau, *Fault Detection and Diagnosis in Chemical and Petrochemical Processes.* Amsterdam: Elsevier Scientific Pub. Co., 1978.

[2] V. Venkatasubramanian, R. Rengaswamy, K. Yin, and S. N. Kavuri, "A review of process fault detection and diagnosis: Part I: Quantitative model-based methods," *Computers & Chemical Engineering*, vol. 27, no. 3, pp. 293–311, 2003.

[3] M. Jung, O. Niculita, and Z. Skaf, "Comparison of Different Classification Algorithms for Fault Detection and Fault Isolation in Complex Systems," *Procedia Manufacturing*, vol. 19, pp. 111–118, Jan. 2018.

[4] V. Venkatasubramanian, R. Rengaswamy, and S. N. Kavuri, "A review of process fault detection and diagnosis Part III: Process history-based methods," *Computers & Chemical Engineering*, vol. 27, no. 3, pp. 327–346, 2003.

[5] H. Marais, G. Van Schoor, and K. R. Uren, "The merits of exergy-based fault detection in petrochemical processes," 2019.

[6] G. Van Schoor and K. R. Uren, "A vision of energy-based visualisation of large scale industrial systems for the purposes of condition monitoring," pp. 337–346, 2018.

[7] J. De Bruin, "Energy-based visualisation of a transcritical CO2 heat pump cycle," Ph.D. dissertation, North-West University, 2019.

[8] S. Greyling, "Graph-based fault detection for a gas-to-liquids process: An exergy approach," Doctor of Philosophy, North-West University, Potchefstroom, 2020.

[9] I. Dincer and Y. A. Cengel, "Energy, Entropy and Exergy Concepts and Their Roles in Thermal Engineering," *Entropy*, vol. 3, no. 3, pp. 116–149, 2001.

[10] M. Kanoğlu, Y. A. Çengel, and I. Dincer, *Efficiency Evaluation of Energy Systems*, ser. SpringerBriefs in Energy. New York: Springer-Verlag, 2012.

[11] E. Sciubba and G. Wall, "A brief Commented History of Exergy From the Beginnings to 2004," *Int. J. Thermodyn*, vol. 10, p. 26.

[12] J. Vosloo, K. R. Uren, G. Van Schoor, and L. Auret, "Exergy-based fault detection on the Tennessee Eastman process," 2019.

[13] I. Dincer and M. A. Rosen, *Exergy: Energy, Environment and Sustainable Development.*, 2nd ed. Elsevier, 2013.

[14] S. d. O. Junior, *Exergy: Production, Cost and Renewability*, ser. Green Energy and Technology. London: Springer-Verlag, 2013.

[15] E. Querol, B. Gonzalez-Regueral, and J. L. Perez-Benedito, *Practical Approach to Exergy and Thermoeconomic Analyses of Industrial Processes*, ser. SpringerBriefs in Energy. London: Springer-Verlag, 2013.

[16] C. Borgnakke and R. E. Sonntag, *Fundamentals of Thermodynamics*, 8th ed. Hoboken, NJ: Wiley, Dec. 2012.

[17] G. Tsatsaronis, "Definitions and nomenclature in exergy analysis and exergoeconomics," *Energy*, vol. 32, no. 4, pp. 249–253, 2007.

[18] H. J. Marais, G. van Schoor, and K. R. Uren, "An Energy-based approach to condition monitoring of industrial processes," *IFAC-PapersOnLine*, vol. 48, no. 21, pp. 772–777, Jan. 2015.

[19] ——, "Energy-based Fault Detection for an Autothermal Reformer," *IFAC-PapersOnLine*, vol. 49, no. 7, pp. 353–358, Jan. 2016.

[20] M. S. Rahman, *Basic Graph Theory*, ser. Undergraduate Topics in Computer Science. Springer International Publishing, 2017.

[21] R. Balakrishnan and K. Ranganathan, *A Textbook of Graph Theory*, 2nd ed., ser. Universitext. New York: Springer-Verlag, 2012.

[22] S. Van Graan, "Graph matching as a means to energy-visualisation of a counter-flow heat exchanger," Ph.D. dissertation, North-West University, Potchefstroom, 2017.

[23] M. De la Fuente, "5th EN/ICE Automation Seminar: Fault detection techniques in the process industry," 2015.

[24] S. Jouili and S. Tabbone, *Attributed Graph Matching Using Local Descriptions*, J. Blanc-Talon, W. Philips, D. Popescu, and P. Scheunders, Eds. Bordeaux, France: Springer, 2009, vol. 5807.

[25] D. Lay, S. Lay, and J. McDonald, *Linear Algebra and Its Applications*, 5th ed. Boston: Pearson, Dec. 2014.

[26] H. Neser, "Energy-based visualisation of a Brayton cycle power conversion unit for the purpose of condition monitoring," Doctor of Philosophy, North-West University, 2019.

[27] N. Ashrafi-Payaman, M. R. Kangavari, and A. M. Fander, "A new method for graph stream summarization based on both the structure and concepts," *Open Engineering*, vol. 9, no. 1, pp. 500–511, Jan. 2019.

[28] Y. Liu, T. Safavi, A. Dighe, and D. Koutra, "Graph Summarization Methods and Applications: A Survey," *arXiv:1612.04883 [cs]*, Jan. 2018.

[29] S. Navlakha, R. Rastogi, and N. Shrivastava, "Graph summarization with bounded error," in *In SIGMOD 2008: Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*. ACM, 2008, pp. 419–432.

[30] Y. Tian, R. A. Hankins, and J. M. Patel, "Efficient aggregation for graph summarization," in *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, ser. SIGMOD '08. New York, NY, USA: Association for Computing Machinery, Jun. 2008, pp. 567–580.

[31] N. Zhang, Y. Tian, and J. M. Patel, "Discovery-driven graph summarization," in *2010 IEEE 26th International Conference on Data Engineering (ICDE 2010)*. Long Beach, CA, USA: IEEE, 2010, pp. 880–891.

[32] J. Wu, *Advances in K-Means Clustering: A Data Mining Thinking*, ser. Springer Theses. Berlin Heidelberg: Springer-Verlag, 2012.

[33] M. Riondato, D. García-Soriano, and F. Bonchi, "Graph summarization with quality guarantees," *Data Min Knowl Disc*, vol. 31, no. 2, pp. 314–349, Mar. 2017.

[34] Y. Bei, Z. Lin, and D. Chen, "Summarizing scale-free networks based on virtual and real links," *Physica A: Statistical Mechanics and its Applications*, vol. 444, no. C, pp. 360–372, 2016.

[35] N. A. Payaman and M. R. Kangavari, "Graph Hybrid Summarization," *Journal of AI and Data Mining*, vol. 6, no. 2, pp. 335–340, 2018.

[36] M. Blanke, M. Kinnaert, J. Lunze, and M. Staroswiecki, *Diagnosis and Fault-Tolerant Control*, 2nd ed. Berlin Heidelberg: Springer-Verlag, 2006.

[37] J. Downs and E. Vogel, "A plant-wide industrial process control problem," *Computers & Chemical Engineering*, vol. 17, no. 3, pp. 245–255, 1993, industrial challenge problems in process control.

[38] A. Ragab, M. El-Koujok, B. Poulin, M. Amazouz, and S. Yacout, "Fault diagnosis in industrial chemical processes using interpretable patterns based on Logical Analysis of Data," *Expert Systems with Applications*, vol. 95, pp. 368–383, 2018.

[39] H. Wu and J. Zhao, "Deep convolutional neural network model based chemical process fault diagnosis," *Comput. Chem. Eng.*, pp. 185–197, 2018.

[40] M. T. Amin, F. Khan, and S. Imtiaz, "Fault detection and pathway analysis using a dynamic Bayesian network," *Chemical Engineering Science*, vol. 195, pp. 777–790, 2019.

[41] K. Tidriri, T. Tiplica, N. Chatti, and S. Verron, "A generic framework for decision fusion in Fault Detection and Diagnosis," *Engineering Applications of Artificial Intelligence*, vol. 71, pp. 73–86, 2018.

[42] J. Vosloo, "Exergy-based fault detection and diagnosis of the Tennessee Eastman process," Doctor of Philosophy, North-West University, Potchefstroom, 2021.

[43] P. R. Lyman and C. Georgakis, "Plant-wide control of the Tennessee Eastman problem," *Computers & Chemical Engineering*, vol. 19, no. 3, pp. 321–331, Mar. 1995.

[44] L. H. Chiang, E. L. Russell, and R. D. Braatz, *Fault Detection and Diagnosis in Industrial Systems*, softcover reprint of the original 1st ed. 2001 edition ed. London ; New York: Springer, Dec. 2000.

[45] J. Szargut, D. R. Morris, and F. R. Steward, *Exergy Analysis of Thermal, Chemical, and Metallurgical Processes*. New York: Hemisphere, 1988.

[46] T. J. Kotas, *The Exergy Method of Thermal Plant Analysis*. Elsevier, Oct. 2013.

[47] R. Rivero and M. Garfias, "Standard chemical exergy of elements updated," *Energy*, vol. 31, no. 15, pp. 3310–3326, Dec. 2006.

[48] F. Gharagheizi, P. Ilani-Kashkouli, and R. C. Hedden, "Standard molar chemical exergy: A new accurate model," *Energy*, vol. 158, no. C, pp. 924–935, 2018.

[49] F. Gharagheizi, P. Ilani-Kashkouli, A. H. Mohammadi, and D. Ramjugernath, "A group contribution method for determination of the standard molar chemical exergy of organic compounds," *Energy*, vol. 70, pp. 288–297, Jun. 2014.

[50] F. Gharagheizi and M. Mehrpooya, "Prediction of standard chemical exergy by a three descriptors QSPR model," *Energy Conversion and Management*, vol. 48, no. 9, pp. 2453–2460, Sep. 2007.

[51] K. R. Uren, G. van Schoor, and L. Auret, "An energy-attributed graph approach for the purposes of FDI in a heated two-tank system," *IFAC-PapersOnLine*, vol. 52, no. 14, pp. 159–164, Jan. 2019.

[52] W. Wolmarans, "A comparison of PCA- and energy-based FDI in a physical heated two-tank process," Ph.D. dissertation, North-West University, Potchefstroom, 2021.

[53] J. H. Smith, "Energy-based fault detection and isolation in an industrial steam turbine system," Ph.D. dissertation, North-West University, 2020.

[54] D. R. Wilson and T. R. Martinez, "Improved heterogeneous distance functions," in *Journal of Artificial Intelligence Research*, 1997, pp. 1–34.

[55] S. Greyling, G. Van Schoor, K. R. Uren, and H. Marais, "Exergy graph-based fault detection and isolation of a gas-to-liquids process," 2019.

[56] L. W. Johnsonn, R. D. Riess, and J. T. Arnold, *Introduction to Linear Algebra*, 5th ed. Virginia Polytechnic Institute and State University: Ed. Boston: Addison Wesley Longman Limited, 2017.

[57] C. Reinartz, M. Kulahci, and O. Ravn, "An extended Tennessee Eastman simulation dataset for fault-detection and decision support systems," *Computers & Chemical Engineering*, vol. 149, p. 107281, Jun. 2021.

[58] B. Bao, M. M. El-Halwagi, and N. O. Elbashir, "Simulation, integration, and economic analysis of gas-to-liquid processes," *Fuel Processing Technology*, vol. 91, no. 7, pp. 703–713, Jul. 2010.

[59] K. T. Knutsen, "Modelling and optimization of a Gas-to-Liquid plant," Master's thesis, Norwegian University of Science and Technology, 2013.

[60] M. Panahi, A. Rafiee, S. Skogestad, and M. Hillestad, "A Natural Gas to Liquids Process Model for Optimal Operation," *Ind. Eng. Chem. Res.*, vol. 51, no. 1, pp. 425–433, Jan. 2012.

[61] A. De Klerk, *Fischer-Tropsch Refining*, 1st ed. Wiley-VCH, 2011.

[62] K. Vugrin, "On the effect of numerical noise in simulation-based optimization," Master's thesis, Virginia Tech, 2003.

Appendices

Appendix A

Algorithms of every FDI method

In this appendix, the algorithms of all three FDI methods are listed. Each algorithm describes the process used to transform the input parameters into various data structures, such as cell arrays, matrices, and vectors, which are ultimately used to analyse the performance of each FDI method.

For the distance parameter method, operational and reference attributed graphs (G_{OP} & G_{Ref}) are used to generate cost matrices (C), which are subsequently used to determine distance parameter values (DC).

Algorithm 1: Distance parameter FDI method. Apply algorithm for $i \in [1, 2, ..., n]$ and $j \in [1, 2, ..., 501]$, where n is the number of process conditions (faulty and normal).

Algorithm input: Operational attributed graphs $G_{OP}(i, j)$ and reference attributed graph $G_{Ref}(1, n)$.

$C\{i, j\}(n, 1) \leftarrow (G_{Ref}(n, 1), G_{OP}(i, j))$
$DC\{i, j\}(n, 1) \leftarrow C\{i, j\}(n, 1)$

For the eigendecomposition method, the reference normal and reference faulty attributed graphs (G_{N_avg} & G_{F_avg}) are used to generate an array of cost matrices ($ArrayA$), which is then used to calculate the set of eigenvalues for each cost matrix ($\vec{\lambda}_{NRef}$). The reference normal and operational attributed graphs (G_{N_avg} & G_{OP}) are used to generate an array of cost matrices ($ArrayB$), which is again used to calculate the set of eigenvalues for each cost matrix ($\vec{\lambda}_{NOP}$).

Algorithm 2: Eigendecomposition FDI method. Apply algorithm for $i \in [1, 2, ..., n]$ and $j \in [1, 2, ..., 501]$, where n is the number of process conditions (faulty and normal).

Algorithm input: Operational attributed graphs $G_{OP}(i, j)$, reference normal attributed graph G_{N_avg}, and the faulty reference attributed graphs $G_{F\_avg}(n)$.

$C(1, n) \leftarrow (G_{N\_avg}, G_{F\_avg}(n, 1))$
$ArrayA(1, n) \leftarrow C(1, n)$
$\vec{\lambda}_{NRef}(1, n) \leftarrow ArrayA(1, n)$
&
$C(i, j) \leftarrow (G_{N\_avg}, G_{OP}(i, j))$
$ArrayB(i, j) \leftarrow C(i, j)$
$\vec{\lambda}_{NOP}(i, j) \leftarrow ArrayB(i, j)$

For the residual-based method, the reference normal and reference faulty attributed graphs (G_{N_avg} & G_{F_avg}) are used to generate an array of residual matrices ($ArrayA\ of\ G_{res}$), which is then used to calculate the binary residual matrices (BIN_{ref}) from which the frequency vector ($\vec{fc}_{NRef}$) is derived. The reference normal and operational attributed graphs (G_{N_avg} & G_{OP}) are used to generate an array of residual matrices ($ArrayB\ of\ G_{res}$), which is then used to calculate the binary residual matrices (BIN_{OP}) from which the frequency vectors ($\vec{fc}_{NO}$) are derived.

Algorithm 3: Residual-based FDI method. Apply algorithm for $i \in [1, 2, ..., n]$ and $j \in [1, 2, ..., 501]$, where n is the number of process conditions (faulty and normal).

Algorithm input: Operational attributed graphs $G_{OP}(i, j)$, reference normal attributed graph G_{N_avg}, and the faulty reference attributed graphs $G_{F\_avg}(n)$.

$G_{res}(1, n) \leftarrow (G_{N\_avg}, G_{F\_avg}(n, 1))$
$ArrayA(1, n) \leftarrow G_{res}(1, n)$
$BIN_{ref}(1, n) \leftarrow ArrayA(1, n)$
$\vec{fc}_{NRef}(1, n) \leftarrow BIN_{ref}(1, n)$
&
$G_{res}(i, j) \leftarrow (G_{N\_avg}, G_{OP}(i, j))$
$ArrayB(i, j) \leftarrow G_{res}(i, j)$

$BIN_{OP}(i,j) \leftarrow ArrayB(i,j)$
$\vec{fc}_{NO}(i,j) \leftarrow BIN_{OP}(i,j)$

Appendix B

Data and MATLAB® code used in this study

These are the links to various folders containing the datasets and code used for this study:

- TEP graph data: A folder containing the TEP data as extracted from the Simulink® model and processed into attributed graph format.
- TEP FDI methods: This folder contains all the MATLAB® code used to generate the cost matrices and residual matrices used by the FDI methods applied to the TEP and the code used to analyze the performance of each FDI method as applied to the TEP. It should be noted that the code used to generate cost matrices for the Distance Parameter FDI method was partially developed as part of the study done by Vosloo [42] and was adapted for this study. All other code was developed as part of this study.
- TEP reduction techniques: The code of all the graph reduction techniques applied to the TEP graph data can be found in this folder.
- GTLP graph data: The attributed graph data in the form of node signature matrices can be found here.
- GTLP FDI methods: All the MATLAB® code used to generate the cost matrices and residual matrices used by the FDI methods applied to the GTLP and the code used to analyze the performance of each FDI method as applied to the GTLP, can be found in this folder. The code only differs slightly from that used for the TEP to accommodate the differences in the data structures of the two processes.
- GTLP reduction techniques: The code of all the graph reduction techniques applied to the GTLP graph data can be found in this folder. The code only differs slightly from that used for the TEP to accommodate the differences in the data structures of the two processes.

Appendix C

Additional reduction technique verification

This appendix is dedicated to verifying the implementation of the remaining graph reduction techniques. In each verification process, the reduction technique reduces the NSM of the reference normal attributed graph.

C.1 Verification of Technique 2

The reference attributed graph data exported to Excel® is used to determine with what percentage each non-zero link attribute varies over the 20 fault conditions. First, the percentage variation from that specific link attribute's value in the reference normal attributed graph is calculated for each link attribute. Each link attribute has 20 variation values which are then averaged to obtain an average variation percentage for a given link attribute. Next, the Excel® function 'PERCENTILE.EXC()' is applied to all the link attributes' average percentage variation values to determine the 50th percentile threshold. Finally, all the links with average percentage variation values below the 50th percentile threshold are set to zero in the attributed graph data.

The MATLAB® code is used to remove links with average percentage variation values below the 50th percentile threshold to compare the resulting reduced attributed graphs. Table C.1 contains the NSM reduced by implementing Technique 2 in Excel® and Table C.2 contains the NSM reduced by implementing Technique 2 in MATLAB®.

C.2 Verification of Technique 3

Technique 3 only requires data from the reference normal attributed graph to determine which attributes must be reduced. First, the absolute values of all the link attributes in the NSM of the reference normal attributed graph are determined. Next, the Excel® function 'PERCENTILE.EXC()' is applied to these absolute link attribute values to determine the 40th percentile threshold. Finally, all link attributes with absolute values less than the 40th percentile threshold are set to zero in the attributed graph data.

The MATLAB® code is then used to remove links with absolute values below the 40th percentile threshold, to compare the resulting reduced attributed graphs. Table C.3 contains the NSM reduced by implementing Technique 3 in the Excel® environment and Table C.4 contains the NSM reduced by implementing Technique 3 with the MATLAB® code.

C.3 Verification of Technique 4

Like Technique 3, Technique 4 only uses the sizes of the attributes in the reference normal attributed graph to determine which nodes should be removed from the attributed graph data. First, the absolute values of the node attributes in the reference normal attributed graph are calculated. Then, 'PERCENTILE.EXC()' is applied to these absolute node attributes to determine the 30th percentile threshold. Finally, all the node attributes with absolute values less than the 30th percentile threshold are removed from the attributed graph data.

Once again, MATLAB® is used to remove node attributes with absolute values below the 30th percentile threshold in order to compare the resulting reduced attributed graphs. The NSM reduced by implementing Technique 4 in the Excel® environment can be seen in Table C.5 and the NSM reduced by implementing Technique 4 with the MATLAB® code can be seen in C.6.

C.4 Verification of Technique 5

To verify the implementation of Technique 5 in Excel®, nodes 4 & 10 are summarized. To do this, all the attributes in the row associated with node 10 are added to their corresponding attributes in the row associated with node 4. Next, all the attributes in the column associated with node 10 are added to their corresponding attributes in the column associated with node

4. Finally, both the row and column associated with node 10, are removed from the graph. This is done to all the graphs in the attributed graph data.

Nodes 4 & 10 are also summarized with the MATLAB® code. Table C.7 contains the NSM reduced by implementing Technique 5 in Excel® and Table C.8 contains the NSM reduced by implementing Technique 5 in MATLAB®.

Since the reduced matrices produced by Excel® match the reduced matrices produced with the MATLAB® code, the implementation of all five graph reduction techniques have been verified.

Table C.1: Reduced attributed graph produced by implementing Technique 2 in Excel.

-5.80E+07	-3.06E+05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-5.36E+07	-2.89E+05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-7.77E+05	-2.51E+04	0.00	0.00	0.00	0.00	3.89E+03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-5.16E+07	-9.70E+05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-3.04E+08	-3.90E+07	0.00	0.00	-3.89E+03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-3.51E+06	-5.85E+05	0.00	0.00
6.42E+07	7.37E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	-1.03E+07	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
4.34E+08	4.44E+07	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	-3.43E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.19E+08
-5.94E+07	6.40E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	3.93E+04	3.51E+06	8.45E+05	0.00	0.00
3.82E+06	3.85E+05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-3.93E+04	0.00	0.00	0.00	0.00	0.00
-3.05E+08	-3.07E+07	0.00	0.00	0.00	0.00	3.51E+06	0.00	0.00	0.00	0.00	-3.93E+04	0.00	0.00	0.00	0.00	0.00
-1.45E+07	-6.85E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-8.45E+05	0.00	0.00	0.00	6.48E+05	-1.43E+05
1.69E+08	2.56E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-6.48E+05	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.04E+08	0.00	6.18E+07	0.00	0.00	0.00	1.43E+05	0.00	0.00

Table C.2: Reduced attributed graph produced by implementing Technique 2 in Matlab.

-5.80E+07	-3.06E+05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-5.36E+07	-2.89E+05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-7.77E+05	-2.51E+04	0.00	0.00	0.00	0.00	3.89E+03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-5.16E+07	-9.70E+05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-3.04E+08	-3.90E+07	0.00	0.00	-3.89E+03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-3.51E+06	-5.85E+05	0.00	0.00
6.42E+07	7.37E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	-1.03E+07	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
4.34E+08	4.44E+07	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	-3.43E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.19E+08
-5.94E+07	6.40E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	3.93E+04	3.51E+06	8.45E+05	0.00	0.00
3.82E+06	3.85E+05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-3.93E+04	0.00	0.00	0.00	0.00	0.00
-3.05E+08	-3.07E+07	0.00	0.00	0.00	0.00	3.51E+06	0.00	0.00	0.00	0.00	-3.93E+04	0.00	0.00	0.00	0.00	0.00
-1.45E+07	-6.85E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-8.45E+05	0.00	0.00	0.00	6.48E+05	-1.43E+05
1.69E+08	2.56E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-6.48E+05	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.04E+08	0.00	6.18E+07	0.00	0.00	0.00	1.43E+05	0.00	0.00

Table C.3: Reduced attributed graph produced by implementing Technique 3 in Excel.

-5.80E+07	-3.06E+05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-5.36E+07	-2.89E+05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-7.77E+05	-2.51E+04	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-5.16E+07	-9.70E+05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-3.04E+08	-3.90E+07	0.00	0.00	0.00	0.00	0.00	4.92E+06	0.00	0.00	0.00	0.00	0.00	-3.51E+06	0.00	0.00	0.00
6.42E+07	7.37E+06	0.00	0.00	0.00	0.00	-4.92E+06	0.00	6.47E+06	6.57E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	-1.03E+07	0.00	0.00	0.00	0.00	0.00	-6.47E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	2.82E+08
4.34E+08	4.44E+07	0.00	0.00	0.00	0.00	0.00	-4.92E+06	0.00	0.00	2.15E+06	6.57E+06	0.00	0.00	0.00	0.00	0.00
0.00	-3.43E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-2.15E+06	0.00	0.00	0.00	0.00	0.00	0.00	1.19E+08
-5.94E+07	6.40E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-6.57E+06	0.00	0.00	0.00	3.51E+06	8.45E+05	0.00	0.00
3.82E+06	3.85E+05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-3.05E+08	-3.07E+07	0.00	0.00	0.00	0.00	3.51E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-1.45E+07	-6.85E+06	0.00	0.00	0.00	0.00	4.92E+06	0.00	0.00	0.00	0.00	-8.45E+05	0.00	0.00	0.00	6.48E+05	0.00
1.69E+08	2.56E+06	0.00	0.00	0.00	0.00	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-6.48E+05	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.04E+08	0.00	6.18E+07	0.00	0.00	0.00	0.00	0.00	0.00

Table C.4: Reduced attributed graph produced by implementing Technique 3 in Matlab.

-5.80E+07	-3.06E+05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-5.36E+07	-2.89E+05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-7.77E+05	-2.51E+04	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-5.16E+07	-9.70E+05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-3.04E+08	-3.90E+07	0.00	0.00	0.00	0.00	0.00	4.92E+06	0.00	0.00	0.00	0.00	0.00	-3.51E+06	0.00	0.00	0.00
6.42E+07	7.37E+06	0.00	0.00	0.00	0.00	-4.92E+06	0.00	6.47E+06	6.57E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	-1.03E+07	0.00	0.00	0.00	0.00	0.00	-6.47E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	2.82E+08
4.34E+08	4.44E+07	0.00	0.00	0.00	0.00	0.00	-4.92E+06	0.00	0.00	2.15E+06	6.57E+06	0.00	0.00	0.00	0.00	0.00
0.00	-3.43E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-2.15E+06	0.00	0.00	0.00	0.00	0.00	0.00	1.19E+08
-5.94E+07	6.40E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-6.57E+06	0.00	0.00	0.00	3.51E+06	8.45E+05	0.00	0.00
3.82E+06	3.85E+05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-3.05E+08	-3.07E+07	0.00	0.00	0.00	0.00	3.51E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-1.45E+07	-6.85E+06	0.00	0.00	0.00	0.00	4.92E+06	0.00	0.00	0.00	0.00	-8.45E+05	0.00	0.00	0.00	6.48E+05	0.00
1.69E+08	2.56E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-6.48E+05	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.04E+08	0.00	6.18E+07	0.00	0.00	0.00	0.00	0.00	0.00

Table C.5: Reduced attributed graph produced by implementing Technique 4 in Excel.

-5.80E+07	-3.06E+05	0.00	0.00	0.00	2.76E+05	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-5.36E+07	-2.89E+05	0.00	0.00	0.00	5.50E+05	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-5.16E+07	-9.70E+05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.55E+05	0.00	0.00
-3.04E+08	-3.90E+07	-2.76E+05	-5.50E+05	0.00	0.00	4.92E+06	0.00	0.00	-3.51E+06	-5.85E+05	0.00	0.00
6.42E+07	7.37E+06	0.00	0.00	0.00	-4.92E+06	0.00	6.57E+06	0.00	0.00	0.00	0.00	0.00
4.34E+08	4.44E+07	0.00	0.00	0.00	0.00	-4.92E+06	0.00	6.57E+06	0.00	0.00	0.00	0.00
-5.94E+07	6.40E+06	0.00	0.00	0.00	0.00	0.00	-6.57E+06	0.00	3.51E+06	8.45E+05	0.00	0.00
-3.05E+08	-3.07E+07	0.00	0.00	0.00	3.51E+06	0.00	0.00	-3.93E+04	0.00	0.00	0.00	0.00
-1.45E+07	-6.85E+06	0.00	0.00	-1.55E+05	4.92E+06	0.00	0.00	-8.45E+05	0.00	0.00	6.48E+05	-1.43E+05
1.69E+08	2.56E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-6.48E+05	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.43E+05	0.00	0.00

Table C.6: Reduced attributed graph produced by implementing Technique 4 in Matlab.

-5,80E+07	-3,06E+05	0,00	0,00	0,00	2,76E+05	0,00	0,00	0,00	0,00	0,00	0,00	0,00
-5,36E+07	-2,89E+05	0,00	0,00	0,00	5,50E+05	0,00	0,00	0,00	0,00	0,00	0,00	0,00
-5,16E+07	-9,70E+05	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	1,55E+05	0,00	0,00
-3,04E+08	-3,90E+07	-2,76E+05	-5,50E+05	0,00	0,00	4,92E+06	0,00	0,00	-3,51E+06	-5,85E+05	0,00	0,00
6,42E+07	7,37E+06	0,00	0,00	0,00	-4,92E+06	0,00	6,57E+06	0,00	0,00	0,00	0,00	0,00
4,34E+08	4,44E+07	0,00	0,00	0,00	0,00	-4,92E+06	0,00	6,57E+06	0,00	0,00	0,00	0,00
-5,94E+07	6,40E+06	0,00	0,00	0,00	0,00	0,00	-6,57E+06	0,00	3,51E+06	8,45E+05	0,00	0,00
-3,05E+08	-3,07E+07	0,00	0,00	0,00	3,51E+06	0,00	0,00	-3,93E+04	0,00	0,00	0,00	0,00
-1,45E+07	-6,85E+06	0,00	0,00	-1,55E+05	4,92E+06	0,00	0,00	-8,45E+05	0,00	0,00	6,48E+05	-1,43E+05
1,69E+08	2,56E+06	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	-6,48E+05	0,00	0,00
0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	1,43E+05	0,00	0,00

Table C.7: Reduced attributed graph produced by implementing Technique 5 in Excel.

-5.80E+07	-3.06E+05	0.00	0.00	0.00	0.00	2.76E+05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-5.36E+07	-2.89E+05	0.00	0.00	0.00	0.00	5.50E+05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-7.77E+05	-2.51E+04	0.00	0.00	0.00	0.00	3.89E+03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-1.11E+08	5.43E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-6.57E+06	0.00	3.93E+04	3.51E+06	1.00E+06	0.00	0.00
-3.04E+08	-3.90E+07	-2.76E+05	-5.50E+05	-3.89E+03	0.00	0.00	4.92E+06	0.00	0.00	0.00	0.00	-3.51E+06	-5.85E+05	0.00	0.00
6.42E+07	7.37E+06	0.00	0.00	0.00	0.00	-4.92E+06	0.00	6.47E+06	6.57E+06	0.00	0.00	0.00	0.00	0.00	0.00
0.00	-1.03E+07	0.00	0.00	0.00	0.00	0.00	-6.47E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	2.82E+08
4.34E+08	4.44E+07	0.00	0.00	0.00	6.57E+06	0.00	-4.92E+06	0.00	0.00	2.15E+06	0.00	0.00	0.00	0.00	0.00
0.00	-3.43E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-2.15E+06	0.00	0.00	0.00	0.00	0.00	1.19E+08
3.82E+06	3.85E+05	0.00	0.00	0.00	-3.93E+04	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-3.05E+08	-3.07E+07	0.00	0.00	0.00	-3.93E+04	3.51E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-1.45E+07	-6.85E+06	0.00	0.00	0.00	-1.00E+06	4.92E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	6.48E+05	-1.43E+05
1.69E+08	2.56E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-6.48E+05	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.04E+08	0.00	6.18E+07	0.00	0.00	1.43E+05	0.00	0.00

Table C.8: Reduced attributed graph produced by implementing Technique 5 in Matlab.

-5.80E+07	-3.06E+05	0.00	0.00	0.00	0.00	2.76E+05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-5.36E+07	-2.89E+05	0.00	0.00	0.00	0.00	5.50E+05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-7.77E+05	-2.51E+04	0.00	0.00	0.00	0.00	3.89E+03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-1.11E+08	5.43E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-6.57E+06	0.00	3.93E+04	3.51E+06	1.00E+06	0.00	0.00
-3.04E+08	-3.90E+07	-2.76E+05	-5.50E+05	-3.89E+03	0.00	0.00	4.92E+06	0.00	0.00	0.00	0.00	-3.51E+06	-5.85E+05	0.00	0.00
6.42E+07	7.37E+06	0.00	0.00	0.00	0.00	-4.92E+06	0.00	6.47E+06	6.57E+06	0.00	0.00	0.00	0.00	0.00	0.00
0.00	-1.03E+07	0.00	0.00	0.00	0.00	0.00	-6.47E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	2.82E+08
4.34E+08	4.44E+07	0.00	0.00	0.00	6.57E+06	0.00	-4.92E+06	0.00	0.00	2.15E+06	0.00	0.00	0.00	0.00	0.00
0.00	-3.43E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-2.15E+06	0.00	0.00	0.00	0.00	0.00	1.19E+08
3.82E+06	3.85E+05	0.00	0.00	0.00	-3.93E+04	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-3.05E+08	-3.07E+07	0.00	0.00	0.00	-3.93E+04	3.51E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
-1.45E+07	-6.85E+06	0.00	0.00	0.00	-1.00E+06	4.92E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	6.48E+05	-1.43E+05
1.69E+08	2.56E+06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-6.48E+05	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.04E+08	0.00	6.18E+07	0.00	0.00	1.43E+05	0.00	0.00

Appendix D

Performance of the FDI methods applied to the GTLP

This appendix contains the detection and isolation rates of all three FDI methods applied to the attributed graph data of the GTLP. The detection and isolation rates of all the fault conditions after applying the distance parameter, the eigendecomposition, and the residual-based FDI methods to the GTLP can be seen in Tables D.1, D.2, & D.3 respectively.

Table D.1: Detection and isolation rates of the distance parameter FDI method applied to the GTLP.

	Main	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12
DR (%)		100	100	75	75	75	100	75	75	75	100	100	100
IR (%) F1	75	75	0	0	0	25	0	0	0	0	0	0	0
IR (%) F2	75	0	75	0	0	0	0	25	0	0	0	0	0
IR (%) F3	75	0	0	75	0	0	0	0	0	0	0	0	0
IR (%) F4	75	0	0	0	75	0	0	0	0	0	0	0	0
IR (%) F5	75	0	0	0	0	75	0	0	0	0	0	0	0
IR (%) F6	75	0	0	0	0	0	75	0	0	0	0	0	25
IR (%) F7	75	0	0	0	0	0	0	75	0	0	0	0	0
IR (%) F8	75	0	0	0	0	0	0	0	75	0	0	0	0
IR (%) F9	75	0	0	0	0	0	0	0	0	75	0	0	0
IR (%) F10	50	0	0	0	0	0	25	0	0	0	50	0	25
IR (%) F11	75	0	0	0	0	25	0	0	0	0	0	75	0
IR (%) F12	50	0	0	0	0	0	0	25	0	0	25	0	50

Table D.2: Detection and isolation rates of the (modified) eigendecomposition FDI method applied to the GTLP.

	Main	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12
DR (%)		100	100	100	100	75	100	100	100	75	100	100	100
IR (%) F1	75	75	0	0	0	0	0	0	0	0	25	0	0
IR (%) F2	25	25	25	0	0	0	25	0	0	0	0	0	25
IR (%) F3	25	0	0	25	25	25	0	0	0	0	0	25	0
IR (%) F4	25	0	0	50	25	0	0	25	0	0	0	0	0
IR (%) F5	75	0	0	0	0	75	0	0	0	0	0	0	0
IR (%) F6	50	0	25	0	0	0	50	0	0	0	25	0	0
IR (%) F7	25	0	0	0	25	50	0	25	0	0	0	0	0
IR (%) F8	75	0	0	0	0	0	0	0	75	25	0	0	0
IR (%) F9	50	0	0	0	0	0	0	25	0	50	0	0	0
IR (%) F10	25	0	0	0	25	0	25	0	0	0	25	0	25
IR (%) F11	50	0	25	25	0	0	0	0	0	0	0	50	0
IR (%) F12	50	25	25	0	0	0	0	0	0	0	0	0	50

Table D.3: Detection and isolation rates of the residual-based FDI method applied to the GTLP.

	Main	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12
DR (%)		100	100	100	100	100	100	100	100	75	100	100	100
IR (%) F1	75	75	0	0	0	0	0	25	0	0	0	0	0
IR (%) F2	100	0	100	0	0	0	0	0	0	0	0	0	0
IR (%) F3	0	0	75	0	0	0	0	25	0	0	0	0	0
IR (%) F4	0	0	75	0	0	0	0	25	0	0	0	0	0
IR (%) F5	0	0	100	0	0	0	0	0	0	0	0	0	0
IR (%) F6	75	25	0	0	0	0	75	0	0	0	0	0	0
IR (%) F7	25	0	75	0	0	0	0	25	0	0	0	0	0
IR (%) F8	0	0	100	0	0	0	0	0	0	0	0	0	0
IR (%) F9	25	25	0	0	0	0	0	0	0	25	0	0	25
IR (%) F10	50	25	0	0	0	0	0	0	0	0	50	25	0
IR (%) F11	100	0	0	0	0	0	0	0	0	0	0	100	0
IR (%) F12	25	25	0	0	0	0	0	0	0	0	0	50	25

Appendix E

Performance of the FDI methods using the GTLP graph data as reduced by all the reduction techniques

The purpose of this appendix is to provide the results of the experimental process conducted in Chapter 7. Each graph reduction technique is applied incrementally to the graph data of the GTLP, and the FDI performance is recorded with each increment. The percentage attribute reduction increases with each increment. The results of Technique 1 to 5 can be seen in Tables E.1 - E.5.

Table E.1: Performance of all three FDI methods using the GTLP graph data as reduced by Technique 1.

Distance FDI Method										
Percentiles	0th	10th	20th	30th	40th	50th	60th	70th	80th	90th
Overall Detection Rate (%)	87.50	87.50	87.50	87.50	85.42	83.33	83.33	100.00	100.00	97.92
Overall Isolation Rate (%)	70.83	70.83	68.75	70.83	70.83	72.92	68.75	56.25	56.25	22.92
Attribute Reduction (%)	0.00	0.00	25.00	27.78	38.89	50.00	66.67	83.33	84.72	93.06
Isolation rate of Fault 1 (%)	75.00	75.00	75.00	75.00	75.00	75.00	75.00	75.00	75.00	0.00
Isolation rate of Fault 10 (%)	50.00	50.00	50.00	75.00	75.00	75.00	75.00	0.00	0.00	0.00
Isolation rate of Fault 12 (%)	50.00	50.00	50.00	75.00	75.00	75.00	75.00	50.00	50.00	0.00
Eigen FDI Method										
Percentiles	0th	10th	20th	30th	40th	50th	60th	70th	80th	90th
Overall Detection Rate (%)	97.92	95.83	100.00	100.00	95.83	93.75	97.92	97.92	97.92	85.42
Overall Isolation Rate (%)	43.75	45.83	50.00	52.08	43.75	43.75	47.92	43.75	43.75	18.75
Attribute Reduction (%)	0.00	0.00	25.00	27.78	38.89	50.00	66.67	83.33	84.72	93.06
Isolation rate of Fault 1 (%)	75.00	75.00	50.00	75.00	50.00	50.00	75.00	75.00	25.00	25.00
Isolation rate of Fault 5 (%)	75.00	75.00	75.00	50.00	75.00	75.00	100.00	75.00	50.00	50.00
Isolation rate of Fault 8 (%)	75.00	75.00	50.00	50.00	75.00	50.00	0.00	75.00	50.00	25.00
Residual FDI Method										
Percentiles	0th	10th	20th	30th	40th	50th	60th	70th	80th	90th
Overall Detection Rate (%)	97.92	97.92	97.92	97.92	100.00	100.00	97.92	100.00	95.83	0.00
Overall Isolation Rate (%)	39.58	39.58	37.50	33.33	45.83	41.67	29.17	22.92	22.92	0.00
Attribute Reduction (%)	0.00	0.00	25.00	27.78	38.89	50.00	66.67	83.33	84.72	93.06
Isolation rate of Fault 1 (%)	75.00	75.00	50.00	25.00	0.00	100.00	100.00	100.00	100.00	0.00
Isolation rate of Fault 2 (%)	100.00	100.00	100.00	100.00	100.00	0.00	0.00	0.00	0.00	0.00
Isolation rate of Fault 11 (%)	100.00	100.00	0.00	0.00	100.00	100.00	0.00	0.00	0.00	0.00

Table E.2: Performance of all three FDI methods using the GTLP graph data as reduced by Technique 2.

Distance FDI Method										
Percentiles	0th	10th	20th	30th	40th	50th	60th	70th	80th	90th
Overall Detection Rate (%)	87.50	87.50	87.50	85.42	83.33	85.42	89.58	89.58	93.75	93.75
Overall Isolation Rate (%)	70.83	70.83	68.75	66.67	64.58	66.67	70.83	68.75	68.75	68.75
Attribute Reduction (%)	0.00	2.78	8.33	13.89	16.67	22.22	25.00	30.56	36.11	38.89
Isolation rate of Fault 1 (%)	75.00	75.00	75.00	75.00	75.00	75.00	75.00	75.00	75.00	75.00
Isolation rate of Fault 10 (%)	50.00	50.00	50.00	50.00	50.00	50.00	75.00	50.00	50.00	50.00
Isolation rate of Fault 12 (%)	50.00	50.00	50.00	50.00	50.00	50.00	50.00	50.00	50.00	50.00
Eigen FDI Method										
Percentiles	0th	10th	20th	30th	40th	50th	60th	70th	80th	90th
Overall Detection Rate (%)	97.92	91.67	93.75	97.92	100.00	100.00	91.67	97.92	100.00	91.67
Overall Isolation Rate (%)	43.75	45.83	47.92	45.83	37.50	37.50	41.67	52.08	45.83	52.08
Attribute Reduction (%)	0.00	2.78	8.33	13.89	16.67	22.22	25.00	30.56	36.11	38.89
Isolation rate of Fault 1 (%)	75.00	75.00	75.00	50.00	75.00	75.00	25.00	50.00	100.00	75.00
Isolation rate of Fault 5 (%)	75.00	50.00	75.00	50.00	50.00	75.00	50.00	50.00	50.00	50.00
Isolation rate of Fault 8 (%)	75.00	50.00	75.00	50.00	50.00	25.00	50.00	75.00	50.00	50.00
Residual FDI Method										
Percentiles	0th	10th	20th	30th	40th	50th	60th	70th	80th	90th
Overall Detection Rate (%)	97.92	97.92	97.92	97.92	97.92	97.92	97.92	100.00	100.00	100.00
Overall Isolation Rate (%)	39.58	39.58	39.58	37.50	35.42	35.42	35.42	35.42	31.25	33.33
Attribute Reduction (%)	0.00	2.78	8.33	13.89	16.67	22.22	25.00	30.56	36.11	38.89
Isolation rate of Fault 1 (%)	75.00	75.00	75.00	50.00	50.00	75.00	75.00	25.00	100.00	100.00
Isolation rate of Fault 2 (%)	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00
Isolation rate of Fault 11 (%)	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	75.00

Table E.3: Performance of all three FDI methods using the GTLP graph data as reduced by Technique 3.

Distance FDI Method										
Percentiles	0th	10th	20th	30th	40th	50th	60th	70th	80th	90th
Overall Detection Rate (%)	87.50	87.50	87.50	87.50	87.50	83.33	83.33	83.33	85.42	83.33
Overall Isolation Rate (%)	70.83	68.75	68.75	68.75	68.75	66.67	68.75	68.75	75.00	70.83
Attribute Reduction (%)	0.00	5.56	11.11	13.89	19.44	25.00	30.56	36.11	41.67	47.22
Isolation rate of Fault 1 (%)	75.00	75.00	75.00	75.00	75.00	75.00	75.00	75.00	75.00	75.00
Isolation rate of Fault 10 (%)	50.00	50.00	50.00	50.00	50.00	50.00	50.00	50.00	50.00	50.00
Isolation rate of Fault 12 (%)	50.00	50.00	50.00	50.00	50.00	50.00	50.00	50.00	100.00	75.00
Eigen FDI Method										
Percentiles	0th	10th	20th	30th	40th	50th	60th	70th	80th	90th
Overall Detection Rate (%)	97.92	97.92	91.67	100.00	100.00	93.75	97.92	91.67	95.83	93.75
Overall Isolation Rate (%)	43.75	41.67	37.50	43.75	50.00	43.75	43.75	39.58	37.50	50.00
Attribute Reduction (%)	0.00	5.56	11.11	13.89	19.44	25.00	30.56	36.11	41.67	47.22
Isolation rate of Fault 1 (%)	75.00	25.00	75.00	75.00	75.00	75.00	75.00	50.00	50.00	75.00
Isolation rate of Fault 5 (%)	75.00	50.00	75.00	50.00	75.00	50.00	75.00	75.00	75.00	75.00
Isolation rate of Fault 8 (%)	75.00	75.00	25.00	50.00	50.00	50.00	50.00	25.00	50.00	50.00
Residual FDI Method										
Percentiles	0th	10th	20th	30th	40th	50th	60th	70th	80th	90th
Overall Detection Rate (%)	97.92	93.75	100.00	100.00	100.00	97.92	100.00	100.00	100.00	100.00
Overall Isolation Rate (%)	39.58	47.92	50.00	50.00	45.83	45.83	41.67	39.58	33.33	18.75
Attribute Reduction (%)	0.00	5.56	11.11	13.89	19.44	25.00	30.56	36.11	41.67	47.22
Isolation rate of Fault 1 (%)	75.00	50.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00
Isolation rate of Fault 2 (%)	100.00	50.00	50.00	50.00	100.00	0.00	0.00	0.00	0.00	0.00
Isolation rate of Fault 11 (%)	100.00	100.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Table E.4: Performance of all three FDI methods using the GTLP graph data as reduced by Technique 4.

Distance FDI Method										
Percentiles	0th	10th	20th	30th	40th	50th	60th	70th	80th	90th
Overall Detection Rate (%)	87.50	87.50	87.50	87.50	87.50	87.50	87.50	89.58	89.58	100.00
Overall Isolation Rate (%)	70.83	70.83	72.92	72.92	72.92	72.92	72.92	72.92	72.92	62.50
Attribute Reduction (%)	0.00	22.22	38.89	44.44	55.56	66.67	75.00	83.33	88.89	94.44
Isolation rate of Fault 1 (%)	75.00	75.00	75.00	75.00	75.00	75.00	75.00	75.00	75.00	75.00
Isolation rate of Fault 10 (%)	50.00	50.00	50.00	50.00	50.00	50.00	50.00	50.00	50.00	50.00
Isolation rate of Fault 12 (%)	50.00	50.00	50.00	50.00	100.00	100.00	100.00	100.00	100.00	25.00
Eigen FDI Method										
Percentiles	0th	10th	20th	30th	40th	50th	60th	70th	80th	90th
Overall Detection Rate (%)	97.92	93.75	89.58	91.67	97.92	95.83	89.58	95.83	89.58	95.83
Overall Isolation Rate (%)	43.75	37.50	43.75	45.83	47.92	50.00	45.83	37.50	37.50	37.50
Attribute Reduction (%)	0.00	22.22	38.89	44.44	55.56	66.67	75.00	83.33	88.89	94.44
Isolation rate of Fault 1 (%)	75.00	50.00	75.00	75.00	75.00	50.00	75.00	75.00	50.00	75.00
Isolation rate of Fault 5 (%)	75.00	25.00	50.00	75.00	100.00	75.00	75.00	50.00	25.00	75.00
Isolation rate of Fault 8 (%)	75.00	25.00	25.00	25.00	0.00	25.00	50.00	75.00	25.00	50.00
Residual FDI Method										
Percentiles	0th	10th	20th	30th	40th	50th	60th	70th	80th	90th
Overall Detection Rate (%)	97.92	97.92	97.92	100.00	100.00	100.00	100.00	100.00	31.25	0.00
Overall Isolation Rate (%)	39.58	56.25	58.33	52.08	47.92	50.00	31.25	16.67	2.08	0.00
Attribute Reduction (%)	0.00	22.22	38.89	44.44	55.56	66.67	75.00	83.33	88.89	94.44
Isolation rate of Fault 1 (%)	75.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	0.00	0.00
Isolation rate of Fault 2 (%)	100.00	50.00	75.00	75.00	75.00	100.00	100.00	0.00	0.00	0.00
Isolation rate of Fault 11 (%)	100.00	100.00	25.00	0.00	0.00	100.00	0.00	0.00	0.00	0.00

Table E.5: Performance of all three FDI methods using the GTLP graph data as reduced by Technique 5.

Distance FDI Method					
Number of Mergers	0	1	2	3	4
Overall Detection Rate (%)	87.50	87.50	87.50	87.50	87.50
Overall Isolation Rate (%)	70.83	70.83	70.83	70.83	70.83
Attribute Reduction (%)	0.00	2.78	5.56	6.94	12.50
Isolation rate of Fault 1 (%)	75.00	75.00	75.00	75.00	75.00
Isolation rate of Fault 10 (%)	50.00	50.00	50.00	50.00	50.00
Isolation rate of Fault 12 (%)	50.00	50.00	50.00	50.00	50.00
Eigen FDI Method					
Number of Mergers	0	1	2	3	4
Overall Detection Rate (%)	97.92	97.92	93.75	93.75	95.83
Overall Isolation Rate (%)	43.75	41.67	37.50	43.75	41.67
Attribute Reduction (%)	0.00	2.78	5.56	6.94	12.50
Isolation rate of Fault 1 (%)	75.00	50.00	50.00	100.00	50.00
Isolation rate of Fault 5 (%)	75.00	50.00	50.00	75.00	75.00
Isolation rate of Fault 8 (%)	75.00	50.00	50.00	50.00	50.00
Residual FDI Method					
Number of Mergers	0	1	2	3	4
Overall Detection Rate (%)	97.92	97.92	97.92	95.83	95.83
Overall Isolation Rate (%)	39.58	41.67	43.75	50.00	41.67
Attribute Reduction (%)	0.00	2.78	5.56	6.94	12.50
Isolation rate of Fault 1 (%)	75.00	75.00	50.00	50.00	50.00
Isolation rate of Fault 2 (%)	100.00	100.00	50.00	50.00	50.00
Isolation rate of Fault 11 (%)	100.00	100.00	100.00	100.00	0.00