Assessing Understandability of the Fault Trees through Metrics from Business Process Modeling and Exploratory Factor Analysis

Arnas Venskunas
a.venskunas@student.utwente.nl
University of Twente
Netherlands, Enschede

ABSTRACT

Fault Tree Analysis (FTA) is a widely used method for assessing the reliability and safety of the system. However, the increasing structural complexity of fault trees can make them difficult to interpret, particularly when models are reused or modified by others. To address this challenge, this study explores the understandability of fault trees by identifying complexity-related metrics that influence the ease with which a model can be interpreted. Drawing on research from related fields, particularly business process modeling, we adapt a set of candidate metrics. These metrics were then applied to a sample of fault tree models, and an Exploratory Factor Analysis (EFA) was performed to uncover latent dimensions underlying perceived complexity. The resulting factor structure provides a foundation for a systematic framework to assess and potentially improve fault tree understandability. This work contributes toward more interpretable safety models.

KEYWORDS

fault tree, complexity, understandability, factor analysis

1 INTRODUCTION

Fault trees (FT) are a widely explored and adopted tool for software verification between a variety of sectors. Fault tree analysis (FTA) provides insight into accident mitigation by identifying critical components, system failure rates, and more. Over time, numerous extensions to traditional fault trees have been developed to address specific scenarios, such as dynamic fault trees [6] and fuzzy fault trees [21], as well as efficient algorithms for their evaluation, including methods to find minimal cut sets and critical components [18, 19].

At its core, a fault tree is structured as a Directed Acyclic Graph (DAG), composed primarily of two types of nodes: base event (BE) represents a fundamental component failure, and logic gate defines how these base events combine to produce higher-level failures. The root node, located at the apex of the tree, symbolizes the overall failure of the system, and its probability or occurrence depends on various combinations of underlying base events. The logic gates utilized in static fault trees are *OR*, *AND* and K/N types [19]. An example of such a tree can be seen in Fig. 1. However, for more complex systems modeled using, for example, dynamic trees, it is characteristic to have specific custom gates [6].

Although fault trees have a relatively simple notation and structure, complex systems often result in larger and/or more intricate trees that can become difficult for humans to interpret [19]. Even though fault trees are not generally hard to understand for non-experts of the FT field, more complex models present an increased challenge for domain specialists who must evaluate fault trees for correctness, particularly when they did not create the models themselves [19]. Therefore, it is important to create fault trees that are as concise and readable to humans as possible. However, currently there is no standardized framework by which to benchmark the complexity of a fault tree. In this paper, the terms "complexity" and "understandability" will be used interchangeably.

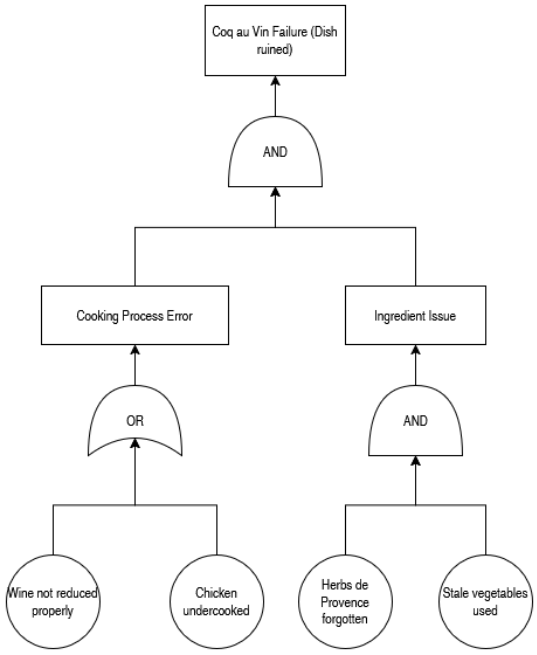

Figure 1: An example of a fault tree

Currently, there is no universally accepted set of metrics to systematically quantify the complexity of a fault tree. However, empirical research from related fields, notably Business Process Modeling (BPM), explores potential metrics that could be adapted to assess fault tree complexity. BPM involves the graphical representation of business processes [11], often using a standardized notation similar to that found in fault trees. The BPM field has been extensively studied in terms of model understandability and complexity, making it a viable resource for this work. The metrics developed in the BPM research could provide a starting point for developing a systematic method of evaluating fault tree complexity. The exploratory factor analysis (EFA) [24] could then be used on these metrics, revealing

43rd Twente Student Conference on IT, July 4th, 2025, Enschede, NL
2025.

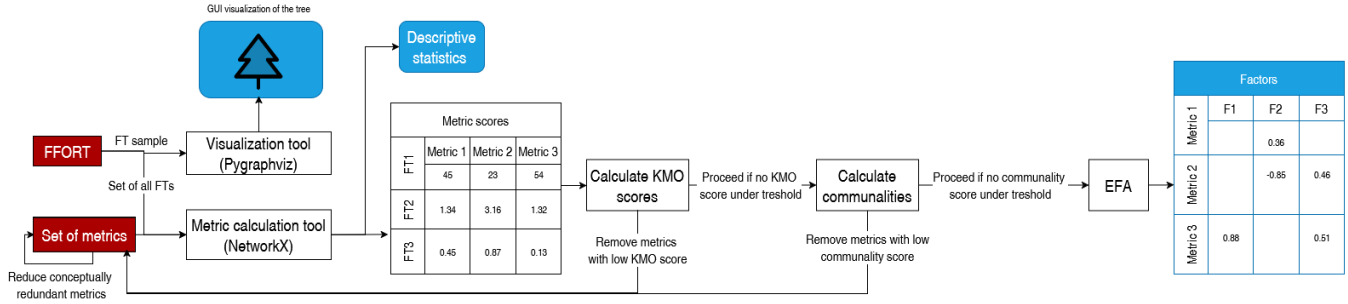

Figure 2: Diagram representing the workflow of the entire process

the underlying dimensions of complexity. A dimension represents broader concepts that simplify the complexity of the original variables. In this paper, the terms "dimension" and "factor" will be used interchangeably. By identifying these dimensions, it could provide a more high-level view of the fault tree understandability itself, while providing researchers and practitioners with guidance when creating FT models. The research question is phrased as "How can an understandability framework for fault trees be developed by adapting complexity metrics and methodologies from the field of BPM?"

1.1 Contributions

This research aims to address the lack of structured methods to approach the complexity and understandability of a fault tree. The key contributions are as follows.

(1) A Python tool to calculate the metrics defined in the methodology section using the NetworkX library and to visualize graphs using the Pygraphviz library. [23].
(2) A survey and adaptation of understandability metrics from related fields applied in the context of fault tree complexity analysis.
(3) Identifying the complexity dimensions to act as a potential base when creating a framework for fault tree complexity evaluation.
(4) A proposed methodology for the systematic development and validation of the fault tree complexity framework.

1.2 Workflow

To provide a better overview of the structure of the study, a visual representation of the workflow is presented in Fig. 2. Given the modular nature of the research, the conventional structure of background, methodology, and results is not ideally suited. Instead, the paper is organized into distinct sections, each addressing a specific component of the overall process as depicted in the diagram. The progression begins with the preliminary refinement of the selected metrics and subsequently follows the logical sequence indicated in the figure. Each section includes its own introduction, methodological approach, and corresponding results.

This paper introduces a number of technical terms that may be unfamiliar to some readers. To support clarity and ease of understanding, a terminology table is provided in **Appendix A** for reference.

2 BPM RESEARCH BASIS

Although research specifically addressing metrics for the understandability of fault trees remains limited, substantial work exists on readability metrics within similar model domains, such as BPM [9, 13, 17] or decision trees [1]. While potential biases may arise when directly applying these existing metrics to fault trees, this study will assume sufficient structural and visual similarity between fault trees and these related models. Specifically, fault trees, such as BPMN diagrams, are structured as directed, layered graphs composed of nodes, gates, and branching flow paths [10, 12]. This assumption enables the reasonable application of established metrics, thereby leveraging prior research to guide the development of fault tree complexity evaluation criteria.

In the field of BPM, several prevailing dimensions of model understandability have been identified: simplicity, fitness, precision, and generalization [14]. Among these, simplicity has shown the strongest correlation with general understandability [14]. Simplicity itself is primarily defined by three subcomponents: structuredness, model size, and entropy [14].

Model size can be quantified using metrics such as the number of nodes [7, 17], which directly reflects the overall scale of the fault tree. Structuredness is captured by metrics such as sequentiality [17], which measures the degree to which the model follows a horizontal and straightforward path structure. Entropy reflects unpredictability and disorder within the model and is associated with metrics such as gate diversity (a wider range of gate types increases unpredictability) [13] and connector heterogeneity [17] (greater variation in the number of incoming connections).

Following these principles, a set of 11 complexity metrics was compiled, each supported by existing literature. However, not all of these metrics will be included in the Exploratory Factor Analysis. In the Results section, the selection will be refined to include only those metrics deemed most relevant and applicable to the context of fault trees specifically.

3 LITERATURE REVIEW AND METRICS

After researching the understandability metrics, 11 complexity metrics were identified. The articles were selected through separate searches conducted using a combination of keywords in Google Scholar and the University of Twente Library. Some of the keywords used are "business process modeling", "process model understandability", "complexity metrics", "BPMN", and "cognitive factors".

From the resulting articles, the most relevant metrics were chosen. They can be seen in Table 1, with their names, descriptions, and appropriate sources indicated next to their name. The last column indicates whether the variable was included in the EFA analysis. The selection process will be explained in the following sections of this paper.

Table 1: List of Graph Metrics and Inclusion Status for EFA (The latter is explained in sections 5.2, 6.1 and 6.2)

Metric	Description	EFA
Number of Nodes (Size) [7, 17]	The total count of all nodes in the graph, namely events and gates (connectors).	✓
Number of Levels [17]	The length of the longest path from the root node to any leaf node in the graph.	
Avg. Connector Degree [17]	The average number of input connections (in-degree) per connector in the graph. A higher average indicates denser junctions.	
Sequentiality [17]	A measure of the linearity of paths; higher values indicate more straight-line sequences of nodes. High sequentiality corresponds to more straightforward paths.	
Connector Heterogeneity [17]	The degree of inconsistency in the number of inputs between connectors of the same type.	✓
Branching Factor [22]	The average number of child nodes per parent node. High branching increases the perceived width of the graph.	
Path Complexity [1, 13]	The average length of all paths from the root to each of the leaf nodes.	✓
Max. Connector Degree [17]	The highest in-degree observed on a single connector in the graph.	
Label Density [5]	The ratio of text labels to available visual space or number of nodes.	
Gate Diversity [13]	The count of unique logic gate types used in the graph.	✓
Graph Density [17]	The density of connections between elements divided by maximum possible connections (value in [0, 1])	✓

4 SAMPLE

4.1 Sample Dataset

To ensure the applicability of the exploratory factor analysis in actual scenarios, it is preferred that the sample accurately reflects real-world fault tree structures. For this purpose, a set of graphs from the FFORT repository [20] was selected. The FFORT (the Fault tree FOResT) dataset consists of fault trees and was developed by academic staff and students of the University of Twente. Tree samples were gathered mainly from academic publications and the rest were developed for other applications, such as experiments.

The possibility of including automatically generated fault trees in the sample was initially considered, but ultimately rejected due to several limitations. To generate fault trees, Olzhas Rakhimov's [16] fault tree generation algorithm was used. One key issue was the difficulty of generating models small enough to be suitable for human interpretability. Automatically generated fault trees typically contain more than 40 basic events and are primarily intended for computational analysis, rather than for evaluation by human experts. Moreover, these models generally rely on a narrow set of gate types, primarily *AND*, *OR*, and K/N, which limits the structural variety and does not reflect the complexity and nuance of specialized systems.

Given that the focus of this study is on the human understandability of fault trees, it was deemed more appropriate to rely on manually created models. These are typically tailored to a specific system and generally incorporate a broader range of specific structural elements, and are the most similar to those that domain specialists encounter in their work. It makes them better suited for evaluating the understandability aspects of fault tree analysis. As Berres and Schumann note [2], automatically generated fault trees often result in complex models that, while useful for computational safety assessments, still require substantial manual effort to verify and refine and are best used as support tools.

From this repository, a representative sample of 45 fault trees was chosen for analysis. An example of such a graph can be visualized using a Python-based tool built with the PyGraphviz library (Fig. 3). According to established guidelines for EFA, a ratio of five observations per variable is considered minimum, while a ratio of ten is recommended [24]. After the final selection of variables, the variable-to-observation ratio becomes 9:1.

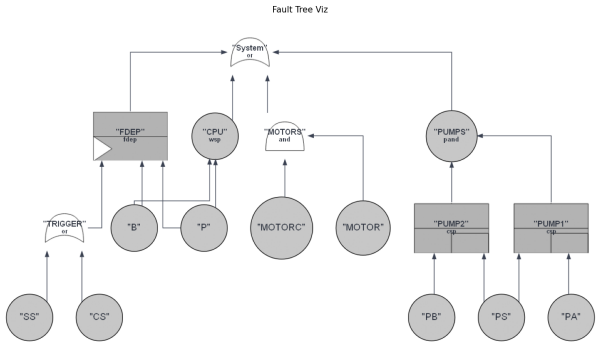

Figure 3: An example of a visualized fault tree from the ffort sample library

4.2 Limitations of the Sample

Although small sample sizes ($N < 50$) are often criticized in factor analysis [3], exceptions can be made under specific conditions. De Winter et al. [4] demonstrate that meaningful factor solutions can be achieved with small samples, provided that

(1) variables are well-defined;
(2) communalities are high (preferably > 0.8).

A well-defined variable refers to one that is conceptually independent (by portraying a measure substantially different from the rest). The definition and explanation of communalities can be found in Section 6.

Similarly, MacCallum et al. [15] argue that the rules of thumb for the sample size should not be applied rigidly, as the adequacy of the EFA results depends more heavily on data characteristics, such as the level of communality and variable overdetermination. When communalities exceed 0.6 and variables are well-specified, the sample size becomes substantially less critical.

5 PRELIMINARY DATA SAMPLE ANALYSIS AND METRIC FILTERING

5.1 Sample Analysis Using Descriptive Statistics

For the preliminary evaluation, descriptive statistics of the sample were calculated. Descriptive statistics can be useful to acquire information about the data and uncover some preliminary patterns before performing a more sophisticated analysis. It was chosen to include only the five-number summary (mean, median, standard deviation, minimum, and maximum). The descriptive statistics of the sample can be seen in Table 2.

Table 2: Descriptive Statistics of Graph Metrics

Metric	Mean	Median	Std. Dev.	Min	Max
Number of Nodes	58.56	39.00	53.47	10.00	219.00
Number of Levels	4.93	5.00	1.16	3.00	8.00
Avg. Connector Degree	0.90	0.90	0.18	0.50	1.36
Sequentiality	0.03	0.01	0.06	0.00	0.40
Connector Heterogeneity	1.27	0.99	1.03	0.28	5.26
Branching Factor	2.86	2.67	0.74	2.00	5.29
Path Complexity	3.98	3.86	0.79	2.74	5.89
Max. Connector Degree	6.89	6.00	5.44	2.00	30.00
Label Density	8.33	7.13	5.53	3.23	38.12
Gate Diversity	0.13	0.09	0.10	0.02	0.41
Graph Density	0.04	0.03	0.03	0.01	0.1

From the descriptive statistics, several interpretations of the model were made, and some of the more interesting patterns were described in the following text. The number of nodes ranges widely from 10 to 219, with a mean of 58.56, highlighting significant differences in model size. However, the median (39) is well below the average, indicating that the data consists mainly of smaller-size samples, and there may be outliers of large size. The avg. connector degree remain relatively stable with the coefficient of variation (standard deviation / mean) being 0.2. This indicates a relatively similar junction complexity on average between the sample instances. The same can be said about the max. connector degree, however, there may be some outliers present, since maximum is as high as 30. Most models show very low sequentiality values (mean = 0.03), indicating that they are not predominantly linear in structure. Notably, label density and gate diversity display skewed distributions, with a few models being significantly more text-heavy or logic-diverse than others. Some metrics show exceptionally high variability, namely the number of nodes (CV = 0.913), connector heterogeneity (CV = 0.811), max. connector degree (CV = 0,79) and gate diversity (CV = 0.77). Sequentiality even attains a CV value of 2, however, it may be extremely sensitive due to very low values attained from most models. In conclusion, the preliminary inspection of descriptive statistics indicates that the small sample size encompasses very diverse models with possible outliers from the perspective of previously indicated metrics that showed greater variability.

5.2 Filtering of variables

Due to the limited sample size, it was preferred to select the smallest set of metrics that still retained substantial explanatory power. During this stage, metrics that showed excessively high correlations with others were excluded to avoid conceptual redundancy.

In total, three such metrics were removed: *number of nodes*, *average connector degree* and *maximum connector degree*. *Number of nodes* was dismissed due to its strong correlation with path complexity, which captures a similar aspect of depth and hierarchical complexity. *Average connector degree* was omitted because in fault trees this metric is functionally equivalent to the branching factor due to the structure: the base events only appear at the lowest level and the intermediate nodes (including the top event or root) consist entirely of connectors. Although it displayed different values, it produced equivalent factor loadings during the execution of EFA. *Maximum connector degree* was dropped due to high correlation with the *average connector degree* and its limited additional explanatory power.

6 EXPLORATORY FACTOR ANALYSIS

6.1 Methodology

To discover the latent structure of the complexity metrics of the fault tree, an exploratory factor analysis was performed, using the R-type common factor analysis approach [8]. In R-type CFA, the analysis is conducted on the correlation matrix of variables, treating variables as the units of analysis and aiming to extract latent factors that account for the covariance between them [8]. This method focuses exclusively on the shared variance among observed variables, making it suitable for identifying underlying dimensions that influence multiple complexity metrics.

Given that the objective of this study is to investigate interrelationships among fault tree understandability metrics rather than individual model properties, the R-type analysis was selected. This method contrasts with Q-type analysis, which focuses on correlations among sample instances rather than variables [8].

Before factor analysis, a re-evaluation of variables was performed. The less clearly defined variables were discarded from the analysis. Thereafter, two tests were performed, namely the Bartlett's Sphericity Test and the Kaiser-Meyer-Olkin (KMO) Sampling Adequacy Measure [8] to check which variables would prove a good basis for a meaningful analysis. Bartlett's Test of Sphericity measures the

extent to which the correlation matrix is different in comparison to the identity matrix, signifying the possibility of deriving clear patterns. The KMO test evaluates how well the variables are suited for factor analysis by measuring the proportion of common variance between the variables (that is, the variance that could be attributed to underlying latent factors) [8]. KMO scores range from 0 to 1. Variables with a KMO score lower than 0.5 were cut off due to their inadequacy for this experiment, as they are below the established threshold [8].

After preprocessing, the communalities of each variable were calculated. Communality ranges from 0 to 1 and measures the degree to which the variance of the variables can be explained by the potential underlying factors [8]. During each iteration, items with a communality value below the threshold of 0.3 were removed because of their low explainability.

The number of factors (dimensions) was derived using a Scree plot. In Scree plot, dimension values lie on the X-axis, while the eigenvalues lie on the Y-axis. The general rule is to select the value where the graph forms an elbow-like structure [24]. A widely used but debated rule is that the selected point must lie above the eigenvalue of one (Kaiser criterion) [8], as anything below could have an increased risk of overfitting.

To make interpretability easier, a Varimax rotation was applied, an orthogonal rotation technique. Varimax maximizes the variance of squared loadings within each factor, yielding a simpler and more interpretable factor structure where each variable loads strongly on one factor and minimally onto others[8].

6.2 Results

Since EFA evaluates the correlation of variances, it is necessary that there is enough variability in the data. The standard deviations from Table 2 are enough to conduct the EFA. Bartlett's Test of Sphericity was significant ($\chi^2 = 113.87$, $p < .001$), indicating a sufficient overall correlation between variables to justify factor analysis. The overall KMO score is 0.686, which is mediocre to acceptable and lies in the range of 0.577-0.791 between variables (see Table 3). The branching factor was dropped due to its low KMO score.

Table 3: KMO Measure of Sampling Adequacy and Communality Score for Each Variable

Variable	KMO Score	Communality
Number of Nodes	0.791	0.582
Path Complexity	0.735	0.334
Connector Heterogeneity	0.577	0.584
Gate Diversity	0.682	0.827
Graph Density	0.644	0.957

Sequentiality and label density were dropped because of their low communality scores. All remaining variables have a mediocre to high communality score, except for path complexity, which is slightly above the threshold of 0.3. The final communality scores can be seen in Table 3.

After analyzing the scree plot (Fig. 4), an elbow-like structure was detected at factors=2, it was significantly above the eigenvalue of 1. Based on the small number of variables and the small sample size, two dimensions were deemed appropriate to discover the initial underlying structure.

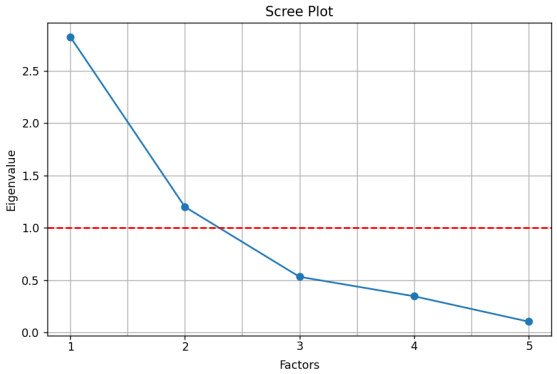

Figure 4: The scree plot resulting from five proposed variables

Table 4: Factor Loadings for Selected Complexity Metrics

Metric	F1	F2
Number of Nodes	-0,629	0,433
Path Complexity	-0,498	
Connector Heterogeneity		0,754
Gate Diversity	0,866	
Graph Density	0,956	

The factor loadings can be found in Table 4. The higher the absolute value, the stronger the correlation. Values with an absolute value of less than 0.35 were removed because they were deemed insignificant [8].

7 DISCUSSION

Based on the analysis conducted, two dimensions were derived, to which the labels and descriptions were given as follows:

7.1 Factor 1: Density–Size Complexity Axis

Factor 1 appears to capture a trade-off between size-related complexity and information density. The negative correlation between the number of nodes and both graph density and gate diversity suggests that larger fault trees tend to be less densely interconnected and show less gate diversity. In contrast, smaller trees often contain a higher concentration of information, expressed through specialized and densely connected components. This indicates that for smaller fault trees, it is not necessarily true that they contain less information; instead, they may reflect systems with greater internal coupling and more sophisticated gates, implying a more specialized connection that would otherwise be reached through a combination of multiple gates. In contrast, larger trees may represent more hierarchical or modular systems in nature, where functionality is distributed across broader structures with less immediate interconnectivity. However, these two concepts are fundamentally different; therefore, this dimension could hypothetically be split into two

subdimensions, namely Density Complexity and Size or Depth Complexity, each measuring their respective complexity concept.

7.2 Factor 2: Variability Complexity

Factor 2 appears to represent a dimension of complexity related to the variability or uniformity of information structures within fault trees, which may be referred to as "Variability Complexity". The "information structures" in this case are primarily represented by connector heterogeneity, which captures the variability of connectors between each connector type. A moderate positive loading from number of nodes suggests that larger fault trees tend to have more uniform gate inputs between types, potentially due to structural repetition or modular design. In contrast, smaller trees tend to be more irregular and show higher variability in how components are connected and interact, likely due to specialized connectors. This form of complexity can be conceptually linked to entropy in the sense that systems with greater connector and their respective input variability and less predictable structure may be seen as being more unpredictable and harder to analyze for the human eye.

8 CONCLUSION

This study investigated the complexity and understandability dimensions of fault trees, with the objective of providing a systematic approach and a methodology to evaluate their understandability. The research already conducted on the subject of understandability in related domains was explored and a list of potential understandability metrics for fault trees was derived. Then, to discover the relationships of these variables, an exploratory factor analysis was performed, revealing two dimensions: *Density–Size Complexity* and *Variability Complexity*.

The *Density–Size Complexity* dimension represents a trade-off between model size and information density, suggesting that larger fault trees often exhibit less dense interconnectivity and gate diversity compared to smaller, more densely interconnected models. This indicates that complexity in smaller trees may originate from concentrated and specialized components, whereas larger trees might represent more modular or hierarchical designs. This dimension could be hypothetically split into density complexity and size or depth complexity, each measuring their respective complexity.

The second dimension, *Variability Complexity*, expresses the complexity arising from inconsistent or varied connector structures within fault trees. Higher connector heterogeneity indicates models that are structurally less predictable.

Based on these findings, practitioners are advised to consider these complexity dimensions when designing fault trees. Reducing positively-correlated or increasing negatively-correlated metric scores of a specific dimension may help reduce the respective dimension's complexity, resulting in a better understandability of the model. Practitioners could also experiment with the metrics defined in this study and incorporate the ones that were not present in the EFA as additional dimensions of complexity.

For further research, it might be valuable to utilize the proposed methodology with larger data samples. Larger data samples might reveal different patterns and dimensions, because the analysis would not be bound by the limitations present in the current study. The variables excluded from the EFA could also be included in such future research either in the EFA or as a standalone dimension itself. In addition, an empirical study could be conducted with human participants, measuring the actual applied impact of discovered complexity factors.

REFERENCES

[1] R. C. Arbiv, L. Lovat, A. Rosenfeld, and D. Sarne. 2024. Optimizing decision trees for enhanced human comprehension. In *Communications in Computer and Information Science*. 366–381. https://doi.org/10.1007/978-3-031-50396-2_21

[2] A. Berres and H. Schumann. 2016. Automatic generation of fault trees: A survey on methods and approaches. In *CRC Press eBooks*. 2485–2492. https://doi.org/10.1201/9781315374987-377

[3] A. L. Comrey and H. B. Lee. 2013. *A First Course in Factor Analysis*. Psychology Press. https://doi.org/10.4324/9781315827506

[4] J. C. de Winter, D. Dodou, and P. A. Wieringa. 2009. Exploratory Factor Analysis With Small Sample Sizes. *Multivariate Behavioral Research* 44, 2 (2009), 147–181. https://doi.org/10.1080/00273170902794206

[5] A. Dikici, O. Turetken, and O. Demirors. 2017. Factors influencing the understandability of process models: A systematic literature review. *Information and Software Technology* 93 (2017), 112–129. https://doi.org/10.1016/j.infsof.2017.09.001

[6] J. Dugan, S. Bavuso, and M. Boyd. 1992. Dynamic fault-tree models for fault-tolerant computer systems. *IEEE Transactions on Reliability* 41, 3 (1992), 363–377. https://doi.org/10.1109/24.159800

[7] K. Figl. 2017. Comprehension of procedural visual business process models. *Business & Information Systems Engineering* 59, 1 (2017), 41–67. https://doi.org/10.1007/s12599-016-0460-2

[8] J. F. Hair, W. C. Black, B. J. Babin, and R. E. Anderson. 2014. *Multivariate Data Analysis* (7 ed.). Pearson Education Limited.

[9] C. Haisjackl, P. Soffer, S. Y. Lim, and B. Weber. 2016. How do humans inspect BPMN models: An exploratory study. *Software & Systems Modeling* 17, 2 (2016), 655–673. https://doi.org/10.1007/s10270-016-0563-8

[10] L. Jimenez-Roa, T. Heskes, and M. Stoelinga. 2021. Fault Trees, decision trees, and Binary Decision Diagrams: A Systematic comparison. In *Proceedings of the 31st European Safety and Reliability Conference (ESREL 2021)*. 673–680. https://doi.org/10.3850/978-981-18-2016-8_241-cd

[11] M. Kocbek, G. Jost, M. Hericko, and G. Polancic. 2015. Business process model and notation: The current state of affairs. *Computer Science and Information Systems* 12, 2 (2015), 509–539. https://doi.org/10.2298/csis140610006k

[12] T. Kräuter, A. Rutle, H. König, and Y. Lamo. 2024. A higher-order transformation approach to the formalization and analysis of BPMN using graph transformation systems. *Logical Methods in Computer Science* 20, 4 (2024). https://doi.org/10.46298/lmcs-20(4:4)2024

[13] R. Laue and V. Gruhn. 2006. Complexity metrics for business process models. In *Lecture Notes in Informatics (LNI)*. 1–12.

[14] J. Lieben, T. Jouck, B. Depaire, and M. Jans. 2018. An improved way for measuring simplicity during process discovery. In *Lecture Notes in Business Information Processing*. 49–62. https://doi.org/10.1007/978-3-030-00787-4_4

[15] R. C. MacCallum, K. F. Widaman, S. Zhang, and S. Hong. 1999. Sample size in factor analysis. *Psychological Methods* 4, 1 (1999), 84–99. https://doi.org/10.1037/1082-989x.4.1.84

[16] Olzhas Rakhimov. 2020. *SCRAM: Probabilistic Risk Analysis Tool (fault tree analysis, event tree analysis, etc.).* https://github.com/rakhimov/scram Accessed via GitHub.

[17] H. A. Reijers and J. Mendling. 2010. A study into the factors that influence the understandability of business process models. *IEEE Transactions on Systems, Man, and Cybernetics - Part A* 41, 3 (2010), 449–462. https://doi.org/10.1109/tsmca.2010.2087017

[18] L. Rosenberg. 1996. Algorithm for finding minimal cut sets in a fault tree. *Reliability Engineering & System Safety* 53, 1 (1996), 67–71. https://doi.org/10.1016/0951-8320(96)00034-8

[19] E. Ruijters and M. Stoelinga. 2015. Fault tree analysis: A survey of the state-of-the-art in modeling, analysis and tools. *Computer Science Review* 15–16 (2015), 29–62. https://doi.org/10.1016/j.cosrev.2015.03.001

[20] E. J. J. Ruijters, C. E. Budde, M. Chenariyan Nakhaee, M. I. A. Stoelinga, D. Bucur, D. Hiemstra, and S. Schivo. 2019. FFORT: A benchmark suite for fault tree analysis. In *ESREL 2019: Proceedings of the 29th European Safety and Reliability Conference*. 878–885. https://doi.org/10.3850/978-981-11-2724-3_0641-cd

[21] H. Tanaka, L. T. Fan, F. S. Lai, and K. Toguchi. 1983. Fault-Tree analysis by fuzzy probability. *IEEE Transactions on Reliability* R-32, 5 (1983), 453–457. https://doi.org/10.1109/tr.1983.5221727

[22] J. Torres-Sospedra, B. Martínez-Salvador, C. C. Sancho, and M. Marcos. 2021. Process Model Metrics for Quality Assessment of Computer-Interpretable Guidelines in PROforma. *Applied Sciences* 11, 7 (2021), 2922. https://doi.org/10.3390/app11072922

[23] Arnas Venskunas. 2025. *experiments_trees: Fault Tree Complexity Metrics and Analysis Tools.* https://github.com/4rnelis/experiments_trees Accessed via GitHub.
[24] A. G. Yong and S. Pearce. 2013. A beginner's guide to factor analysis: Focusing on exploratory factor analysis. *Tutorials in Quantitative Methods for Psychology* 9, 2 (2013), 79–94.

AI STATEMENT

During the preparation of this work the author(s) used ChatGPT in order to make their writing expression better. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the content of the work.

APPENDIX A: TERMINOLOGY

Term	Definition
Fault Tree (FT)	A graphical model representing the logical relationships between system failures and component faults, typically used in reliability analysis.
Base Event (BE)	A fault or failure at the level of the fundamental component, located at the base of a fault tree.
Logic Gate	An operator (for example: AND, OR, K/N) that connects base events or intermediate events to represent logical dependencies.
Top Event	The root node of a fault tree that represents the overall system failure being analyzed.
Dynamic Fault Tree (DFT)	An extension of traditional fault trees incorporating dynamic gates or time-based behavior.
Directed Acyclic Graph (DAG)	A type of graph with no cycles, often used to structure fault trees.
Understandability	The ease with which a fault tree can be interpreted by humans; used interchangeably with "complexity" in this paper.
Complexity Metrics	Quantitative measures used to assess various aspects of the structure of the fault tree that can affect human comprehension.
Business Process Modeling (BPM)	A domain where business workflows are modeled graphically; its complexity metrics are adapted in this study for FTs.
Exploratory Factor Analysis (EFA)	A statistical method for identifying latent dimensions that explain the observed correlations between variables.
Factor / Dimension	A latent construct that underlies a group of related complexity metrics, identified via EFA.
Communality	The proportion of a variable's variance that is explained by common factors in factor analysis.
Varimax Rotation	A method used in EFA to simplify the interpretation of factors by maximizing the variance of squared loadings.
Kaiser-Meyer-Olkin (KMO) Score	A measure of sampling adequacy that assesses the suitability of variables for factor analysis.
Bartlett's Test of Sphericity	A statistical test assessing whether the correlation matrix is significantly different from an identity matrix.
Scree Plot	A graph plotting eigenvalues to help determine the number of factors to retain in EFA.
Density–Size Complexity	A factor capturing the trade-off between graph size (number of nodes) and information density.
Variability Complexity	A factor reflecting the structural irregularity or inconsistency within the fault tree.