2024 Conference on Systems Engineering Research

Graph Complexity Measures as Indicators of Verification Complexity

Sukhwan Jung[a], Alejandro Salado[a,*]
shjung@arizona.edu, alejandrosalado@arizona.edu

[a]*The University of Arizona, 1127 E James E Rogers Way, Tucson, AZ, USA*

Abstract
Increasing system complexity affects the complexity of their verification strategies, cognitively overloading engineers throughout the system development lifecycle. A graph-based approach has been developed to allow a scalable verification strategy complexity analysis through the use of fundamental patterns. Through the verification strategies graphs, this paper proposes a possibility of a mathematical measurement for verification strategies independent from the system complexity in terms of requirements and their verification. Two verification strategies on real world systems were used to showcase the scalability and resource efficiency of the approach. Ordinal comparison between the two graphs revealed that there were mathematical graph complexity measures correlated with the complexity of verification strategies they represent, with their fundamental patterns providing additional information on their differences. These correlations indicate that the verification strategy complexity is connected to their graph complexity, validating the feasibility of a quantitative measurement applicable to existing real-world applications and future system developments.

Keywords: verification; modelling; knowledge graphs; complexity;

1. Introduction

Verification is the process of assessing the fulfillment of a number of system requirements by a system using various verification methods including inspection, analysis, demonstration, or testing (Engel, 2010). These verification activities are performed in a specific sequence to satisfy relevant requirements, forming a verification strategy (Walden

* Corresponding author.

et al., 2015). These sequences are designed to achieve certain functional confidence in target systems while maximizing resource efficiency. We suggest that the complexity of these verification strategies is dictated by the number of requirements, verification activities, and their interconnections. Verification strategies can also be conceptualized as a sequential information transaction, since a verification activity is defined as "the collection of information about a specific aspect of the system under development" (Jung and Salado, 2023a). Such information, in the form of verification evidence, forms an information influence between verification activities and requirements (Salado and Kannan, 2018). Since information processing requires cognition (Piccinini and Scarantino, 2011), more complex verification strategy likely requires an increased cognitive load of engineers, which is known to be positively associated with human errors (Ayres, 2001; Dörner and Güss, 2022) when there is cognitive overload (Kirsh, 2000). We hypothesize that, as a result of verification complexity, engineers could experience a range of cognition bias errors from overgeneralizations, information misinterpretations, false assumptions, or priority mismatches (Dörner and Güss, 2022).

Reducing the cognitive load of engineers may therefore be crucial to reduce possible human errors during verification planning and assessment. Providing a more computational foundation to the field is one of the possible approaches to such problem, converting the verification strategy into a machine-readable data. This conversion allows a mathematical approach to verification strategy analysis that can provide better data visualization (Anderson et al., 2011) or isolating elements for task complexity reduction (Ayres, 2006).

The goal of this paper is to assess the feasibility of a quantifiable complexity metric for verification strategies. This was because we only had access to two real world verification strategies eliminating the possibility of non-linear interpolation in the previous research (Jung and Salado, 2023a). On top of that, there is no publicly available numerical complexity metrics for verification strategies to our knowledge. Linear regression cannot be done without an existing numerical measure, be it manual or automated. Expanding on the previous research on graph-based verification strategy analysis, this paper searched differences in graph complexity of the two datasets. These differences were then compared to manually determined verification complexity, where one verification strategy was deemed more complex than the other by the domain experts.

The paper identified a set of graph complexity measures as verification strategy complexity indicators based on their ordinal similarity between two datasets and the manually perceived verification strategy complexity order. A series of graph complexity measures were found and their relative differences between two datasets were analyzed to detect possible complexity differences between them. The measures with adequate relative differences are considered as the verification strategy complexity indicators, showing possible correlations between the graph complexity and verification strategy complexity in contextually rich real-world verification strategies.

This paper is organized as follows. Section 2 discusses the research design, including the scale of verification strategies, knowledge graph construction, descriptions of two data sources, and the metrics used during the analysis. Section 3 shows the ordinal comparison between two datasets based on a size-based heuristic and expert opinions about their perceived complexities, on the whole graphs as well as their subgraphs specific to a number of fundamental verification strategy patterns. Section 4 provides discussions and concluding remarks on the study, accompanied by a description of planned future work on the verification complexity measure proposal.

2. Research Design

2.1. Data sources and Knowledge Graphs

A graph-based approach to verification strategy complexity analysis has been proposed to allow a scalable complex verification strategy analysis. A verification strategy graph is an acyclic directed knowledge graph representing the verification strategy, having verification activity entities as nodes and information flow as edges (Salado and Kannan, 2018). An initial analysis of these knowledge graphs revealed that the graph-based approach was capable of representing complex verification strategies used in two real-life industrial applications. The complex interconnections between requirements and verification activities were represented as a single graph, successfully representing large-scale verification strategies exceeding traditional verification engineering research use cases in terms of problem sizes and entity interdependencies (Engel, 2010). Quantifiable measurements with visualizations provided evidence of

analysis that can be performed quickly on verification strategies regardless of their sizes and the interconnections within them while the graph visualizations offered manually interpretable summary of the verification strategies (Jung and Salado, 2023a). The graph-based approach was then expanded with eight fundamental patterns that were based on fundamental patterns of verification strategies identified in past research (Jung and Salado, 2023b; Salado and Kannan, 2019). Eight fundamental patterns were proposed as substitutes for the elemental patterns to cover a wider range of verification behaviors, each with distinct graph structure characteristics. The retrieved subgraphs were manually interpreted to reveal the informative nature of fundamental pattern subgraphs, reducing the cognitive load of engineers by presenting a more focused view with a specific goal in smaller entity sizes. In this research, graphical quality measures were assumed to have connections to the quality and complexity of the verification strategies.

Two verification strategy datasets, corresponding to two different systems, were independently provided by two international private companies in the form of Excel files containing requirements and verification matrices. The first project *pharma* was aimed at providing a non-pharmacological physical treatment device for a specific medical condition, while the second project *defense* developed a defense system product. In accordance with non-disclosure agreements made between the University of Arizona and both companies, any competitively sensitive information was protected through anonymization and sanitation. The verification matrix for the *pharma* project contained system requirements, associated verification activities, associated verification evidence in the form of verification closeout documents, and additional information about the model used in the verification activities. Traceability between the different objects was provided by having them in the same row in one sheet or manually assigning unique identifiers across different sheets. The artifacts for the *defense* project were produced in a DOORS database, which was exported to a single file before deliberation. The verification matrix contained system requirements and associated verification events and/or activities. Traceability between the different objects was managed in DOORS and was exported as objects in the same row in one sheet or assigning unique identifiers across different sheets.

Both strategies encompass complex, real-world problems with requirement and verification sizes far exceeding other data sources used in the existing literature; *defense* contains 5,779 requirements and 3,115 verification activities as shown in Table 1 while the largest toy problem used in existing literature has 15 of each (Salado and Kannan, 2019). They underwent graph a construction process we previously employed (Jung and Salado, 2023a). The *parameter* and *verification* nodes were read as the basis of respective verification strategy graphs, while other external entities such as *model* and *document* were added when direct connections to the former node types were found within the datasets. The relationships are divided into four categories: a verification *verifies* a parameter, a parameter *satisfies* another parameter, a parameter *requires* a model, and a verification *generates* a document. There are clear size differences between the two datasets, both in terms of entities (nodes) and relationships (edges). Following the heuristic of size correlating with complexity as well as manual domain expert determination, the *defense* graph is assumed to have higher cognitive loads compared to the *pharma* graph due to their differences in verification strategy complexity.

Table 1 The number of entities and relationships in *pharma* and *defense* verification strategy graphs

Nodes	Verification	Parameter	Document	Model
pharma	179	129	48	48
defense	3,115	5,779	-	28

Edges	*verifies*	*satisfies*	*requires*	*generates*
pharma	179	61	129	194
defense	11,036	3,768	2,515	-

2.2. Fundamental Patterns Used for Analysis

The elemental patterns of verification strategies represent seven distinctive verification scenarios in their basic format (Salado and Kannan, 2019). The culmination of a single pattern results in a set of connected subgraphs representing each of the pattern's verification scenarios. We hypothesize that these subgraphs would exhibit quantitatively distinguishable graphical characteristics containing human-readable contexts. The complexity of human cognition in understanding these pattern subgraphs would therefore be correlated with graph complexity measures.

The structure of the available dataset resulted in three elemental patterns inaccessible. Patterns II and IV were excluded due to the lack of hierarchical variation activities in the dataset, while the lack of temporal variations in the

source verification matrices rendered Pattern VII inapplicable. We previously proposed eight *fundamental* patterns (Jung and Salado, 2023b) to cover more variations in the verification scenarios in order to observe the similarities and differences between graphical structures and verification scenarios in more detail. Five of these fundamental patterns in Table 2 were used in this research. First, a *generic* pattern S^1 represents a structural characteristic baseline. S^3, S^4, and S^7 are slightly altered versions of *elemental* patterns. S^8 examines the orthogonality in the verification strategy, which is defined as parameter subsets independently verified without accessing each other. A *semi-orthogonal* pattern was found as a variation of S^8, where the definition of orthogonality is relaxed resulting in more frequent pattern matches. Each pattern is found within the graph, and all edges satisfying it are combined to generate a *pattern graph*, a set of connected subgraphs representing a verification strategy specific to the given pattern. S^2 was not utilized as the cycle bases overlap each other heavily, while S^5 and S^6 were skipped for their lack of appearances on the *pharma* dataset.

Table 2 List of the five fundamental patterns and one variation used in the analysis

Pattern	Name	Description
S^1	*Connected Subgraphs*	Reflects how centralized verification strategies are.
S^3	*1V→1P*	Stricter Pattern I, allowing only one-to-one relationships.
S^4	*nV→1P*	Stricter Pattern III, allowing only many-to-one relationships.
S^7	*nP–1P*	Pattern VI, calculates the maximum spanning trees between parameters.
S^8	*Orthogonal*	Captures verification-parameter orthogonality in verification strategies.
$S^{8'}$	*Semi-Orthogonal*	Adds possibly orthogonal relationships with longer paths to S^8.

2.3. Metrics Used for Analysis

There is no single universal measure for graph complexity as the graph can manifest in various ways. For example, spanning tree count (Brown et al., 1996) would fail to capture the accurate graph complexity when the graph is disconnected. Multiple graph measures are therefore calculated to determine measures correlated with the verification strategy complexities. Table 3 lists the ten graph complexity measures used for analysis in this paper.

Table 3 List of the ten graph complexity measures used in the analysis

ID	Name	Description
m_0	*Node count*	Number of nodes.
m_1	*Edge count*	Number of edges.
m_2	*Density*	The ratio of edges over all possible edges, $2m_2 / m_1(m_1 - 1)$.
m_3	*Subgraph count*	Number of connected components, or subgraphs.
m_4	$Modularity_{Label\_propagation}$	Modularity score based on the label propagation community detection algorithm.
m_5	$Centrality_{closeness}$	Average of the inverse of average shortest path lengths starting from each node.
m_6	$Centrality_{information}$	Variation of closeness centrality, measured as the contribution to global efficiency.
m_7	*Efficiency*	Average of the inverse shortest path lengths between all node pairs
m_8	*Communicability*	The sum of all possible walks in the graph using adjacency matrix exponentiation.
m_9	*Graph energy*	The sum of absolute eigenvalues of the adjacency matrix.

Basic graph attributes such as the *node count*, *edge count*, and *density* were first calculated. These graph complexity measures are based on the simple heuristic of 'bigger, heavier, and dense graphs are more complex', and could be considered as the baseline estimation of the verification strategy complexity. Density decreases when the graph grows with non-exponential edge growth, therefore the larger *defense* graph is expected to show lower density. *Subgraph count* was added as another basic graph attribute. This was introduced under the assumption that the disconnected verification strategy subsets require individual attention; each subgraph could be interpreted as an independent task, therefore bigger subgraph would increase the overall cognitive load of the involved engineers. Connectivity also represents the graph complexity by measuring the minimum number of nodes or edge removals required to disconnect the given graph. The use of the minimal connection between components is not suitable for analyzing the verification strategy graph as a one-on-one relationship (S^3) is one of their major elemental patterns. Pilot experiments showed that 93.76% out of the total of 1,122 graphs had node and edge connectivity of one; the lack of variations resulted in

both measures not being considered during the analysis. The average shortest paths and eccentricity were not used as the verification strategies are not expected to be represented as connected graphs.

Modularity measures the strength of modules within a graph, or how easily communities can be formed and how pronounced their boundaries are. This is done by detecting communities with statistically significant edge abundance. Three sets of community detection algorithms, Clauset-Newman-Moore greedy modularity, Louvain community detection, and label propagation algorithms were used to detect three community sets. All three algorithms generated similar modularity values on both graphs and their *pattern graph* pairs, with their relative ratios ranging from 1.0197 to 1.1139. Among these, the former two showed almost identical modularity values with comparatively less distinction between the two datasets. Modularity based on *label propagation* communities (Newman, 2004) was used, where communities are found by iteratively populating community labels based on a small number of randomly assigned seeds. The community labels are diffused throughout the graph at each iteration, utilizing neighboring nodes as the source. Clique numbers as well as a number of cliques were tested to show different perspectives of modular subsets, but the preliminary experiment showed that there are no cliques of size over two. There is no triangle in either graph, therefore the clique measures were not analyzed further.

Centrality on the other hand ranks the nodes by their importance in graph traversal, assigning higher values to the hub nodes at the center of communication. Five centrality measures were calculated and averaged in the pilot experiment: betweenness, closeness, degree, Kats, and information centrality. All five measures shared high correlations with each other, with the lowest correlation coefficient of 0.7812 between closeness and betweenness centralities which was expected due to the two measures having conceptually different perspectives of centrality. Therefore, only two measures were selected for analysis. *The closeness* centrality of a node is measured as the inverse of the average length of all shortest paths starting from it, showing the most deviation from the other four centrality measures. *Information* centrality is a variation of closeness centrality and efficiency (Fortunato et al., 2004), focusing on the information propagation efficiency of the graph. The information centrality of a node is measured as the drop in global graph efficiency with its removal, representing how much in-graph communications were affected (Latora and Marchiori, 2007). Cognitive complexity is closely correlated with the information flow during the mearing process hence this measure was expected to represent complexity more tailored to human cognition. Average *efficiency* is calculated as the average of inverse shortest distances between all node pairs and was also considered in the analysis.

Communicability (Estrada and Hatano, 2008) was selected to reflect the philosophy of cyclomatic complexity measure developed for the software engineering domain. The procedural nature of software implementation and verification strategies made cyclomatic complexity a candidate, measuring the number of independent execution paths in the flow graph (Watson et al., 1996). A sum of various length walks was calculated with the communicability measure instead in this paper as datasets used in the experimented lacked information on their verification sequences. The verification strategy graphs therefore had no single entry points rendering the original measure inapplicable. This measure can be viewed as an antithesis of *efficiency*, where paths of all lengths are considered together instead of the shortest path. Lastly, *graph energy* was used for its unique approach to measuring graphs as it showed positive correlation to the system complexity in real applications (Gutman, 2001; Sinha and de Weck, 2013). It applies the concept of the chemical energy of electrons, measuring the energy of a graph by summing the absolute graph eigenvalues. Binary measures showed that the graphs are not planar nor Eularian as with most of their *pattern graphs*; there were fewer comparative values therefore the binary graph measures were not used.

3. Analyzing the Connection Between Verification Strategy Complexity and Graph Complexity Measures

The traditional verification strategy complexity measures required manual interpretations of detailed verification implementation records. Direct complexity calculations were therefore unfeasible for both *pharma* and *defense* using existing methods, as both datasets were too large for manual interpretations and their verification matrices did not track the verification implementation records. Instead of numerically comparable complexity measures, the size-based heuristic was used to find a complexity difference between the two graphs; *pharma* was considered to have a lower verification strategy complexity compared to *defense*. A series of graph complexity measures were ordinally compared between the two graphs to find possible complexity hierarchy in the two verification strategies they represent. Their

absolute values, large or small in their relative value range, were inconsequential to ordinal comparisons and therefore were not considered during the analysis.

3.1. Ordinal Indicators for Verification Strategy Complexity

The comparison was done by reviewing the measure ratio, which is defined as:

$$r(m_n) = m_n^{defense} / m_n^{pharma} \; where \; n = [0, \dots, 9] \quad (1)$$

for all graph complexity measures, dividing the measured value for *defense* ($m_n^{defense}$) by the same measure calculated for *pharma* (m_n^{pharma}). Both $r(m_0)$ and $r(m_1)$ in Table 4 reflected the size differences between the two graphs, showing that the *defense* had 21.9298 times more entities and nearly 30.7620 times more relationships than the *pharma* graph. These results naturally aligned with the size-based complexity heuristic, rendering them the most basic verification strategy complexity indicators. The relatively small difference between $r(m_0)$ and $r(m_1)$ indicated a non-exponential edge growth in the verification strategy graph. This proportionate increase in edge count made the *defense* a significantly sparse graph compared to *pharma* reflected by a density ratio $r(m_2) = 0.0638$. This was an expected behavior of verification strategies sharing a set of elemental patterns for additional edges. While being one of the fundamental graph complexity measures, density is mostly used for distinguishing graphs with different fundamental structures. The verification strategies used in the experiment rarely showed a significant density difference independent of differences in nodes and edge counts; it was therefore deemed as a less informative indicator when used in conjunction with the former two.

Table 4 The relative ratio of complexity measures $r(m_n)$ using the whole graphs

$r(m_0)$	$r(m_1)$	$r(m_2)$	$r(m_3)$	$r(m_4)$	$r(m_5)$	$r(m_6)$	$r(m_7)$	$r(m_8)$	$r(m_9)$
21.93	30.76	0.06	15.30	1.11	0.88	0.06	0.80	1.00E+09	18.91

The ratio between the *subgraph count* $r(m_3) = 15.3000$ followed another size-based heuristic of a larger system having more components, showing a significant difference between the two graphs following m_0 and m_1. This finding was in sync with the verification strategy's characteristics, where partition in orthogonal sets was considered one of the verification strategy complexity indicators; the more orthogonal partitions there are, the more independent verifications are required. S^8 represented the partition count in orthogonal sets, therefore it was considered as the main analysis focal point for m_3. Like the density, both $r(m_4) = 1.1139$ and $r(m_5) = 0.8849$ visualized the effects of sharing a set of elemental graph patterns. Modularity and closeness centrality showed significantly smaller differences between the graph than density as well, again indicating that both graphs shared similar graph structures independently of their size differences. The purpose and roles of the hub nodes were similar in both graphs (being high-level Parameters and Models) sharing a pseudo-tree structure, therefore similar degrees of modularizations were expected. The goal of verification strategy complexity indicators was to distinguish the difference within the similarly structured verification strategies, therefore both m_4 and m_5 were deemed not suitable indicators.

A significant difference could be observed between the information centralities with $r(m_6) = 0.0638$, returning significantly lower values for *defense* similar to density m_3. The value reflected the relatively higher degree of information stored in *pharma* edges, resulting in lower information resistance and larger information coverages in its paths. The verification strategy complexity comes from the complexity engineers experience during the verification process, therefore strategies with higher complexity can be described as cognitively complex, as well. The *defense* graph had higher cognitively complexity due to the relative information shortage in their modules, forcing engineers to traverse more entities to reach the same degree of information compared to the number of entities required in a cognitively less complex *pharma* dataset. As a variation of closeness centrality, the information centrality (m_6) was not fully dependent on graph structures and therefore was deemed as an effective cognitive complexity indicator. The same cannot be said for the efficiency measure $r(m_7) = 0.8016$ as it utilizes the shortest paths between node pairs; it shares the same pitfalls of other structure-dependent complexity measures, failing to detect significant differences

between the two verification strategies. The ratio differences between m_6 and m_7 were attributed to the node count differences; *pharma* had fewer nodes to measure the average on. The information centrality was measured by calculating how much global efficiency decreased when each node was removed from the graph therefore a smaller graph, with shared elemental patterns, would have nodes each with relatively more role in the global information dissemination effort.

The communicability measure was where the differences were the most pronounced with $r(m_8)$ = 1.0038E+09, as the number of possible walks grew exponentially with increasing graph sizes. The measure was intended to compare graphs with smaller size differences; the ratio was over-emphasizing the difference between the two graphs as it was determined that one was realistically not a billion times more complex than another. Communicability was considered a successful but less desirable complexity indicator that could be utilized when other measures fail. Graph energy is a lesser used complexity measure with the recent surge in its usage. It represents a graph as a chemical molecule, calculating the electron energy level from nodes. $r(m_9)$ = 18.9136 is slightly lower than the value of $r(m_1)$, indicating that the graph energy could also become an effective complexity indicator.

3.2. Ordinal Indicators for Verification Strategy Complexity in Pattern Graphs

The use of *pattern graphs* with m_3 showed that the variations in fundamental patterns are significant enough to warrant further analysis of pattern-specific complexity measures. This was done to test whether the different fundamental patterns affect existing indicators differently, potentially offering their pattern-specific variations as better indicators. Previously discarded graph complexity measures were also tested to discover potentially viable variations based on specific patterns. Defining m_n^k as the complexity measure m_n on the *pattern graph* for S^k, Table 5 shows how each of the six *pattern graphs* affected the relative graph complexity measure ratio, marking the previously selected indicators (m_0, m_1, m_3, m_6, m_8, and m_9) in bold. The first pattern S^1 showed identical values to those in Table 4 as the connected subgraphs (S^1) pattern graph is identical to the whole graph. This row was added for comparative presentation and was not analyzed duplicitously.

Table 5 The relative ratio of complexity measures ($r(m_n^k)$) using their respective *pattern graphs* for the five fundamental patterns and a variation

$r(m_n^k)$	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9
S^1	**21.93**	**30.76**	0.06	**15.30**	1.11	0.88	**0.06**	0.80	**1.00E+09**	**18.91**
S^3	**10.51**	**11.15**	0.10	**10.51**	0.93	0.18	**0.10**	0.20	**3.29E+00**	**10.54**
S^4	**14.50**	**15.41**	0.07	**10.09**	1.05	0.16	**0.07**	0.16	**7.11E-01**	**12.77**
S^7	**38.83**	**62.53**	0.04	**12.91**	0.93	0.66	**0.02**	0.69	**1.55E+03**	**44.08**
S^8	**92.33**	**211.28**	0.02	**10.67**	0.93	1.18	**0.01**	1.17	**8.52E+04**	**117.35**
$S^{8'}$	**23.68**	**36.50**	0.06	**16.50**	1.15	0.84	**0.08**	0.75	**1.18E+09**	**20.52**
Average	**33.63**	**61.27**	0.06	12.66	1.02	0.65	0.06	0.63	**3.64E+08**	**37.36**

Both the node and edge count ratios $r(m_0)$ and $r(m_1)$ increased when *pattern graphs* were utilized, from 21.9298 and 30.7620 to 33.6310 and 61.2704 respectively. This indicates that the *pattern graphs* on average were larger in the *defense* graph even when their size differences were accounted for. This size disparity was also observed with graph energy m_9, with the average ratio $r(m_9)$ increased to 37.3601 from 18.9136. The detailed analysis revealed that S^8 and its variation were the main source of size disparity between $r(m_n)$ and $r(m_n^k)$ $where\ n = [0,1,8,9]$, reflecting the increasing ratio of orthogonal subsets in the larger verification strategy. S^8 showed outlier maximum ratios for m_0, m_1, and m_9 and an outlier minimum ratio for m_6. This was in sync with the assumption that the verification strategy orthogonality is correlated with its complexity; higher orthogonality is assumed to result in lower complexity. The *Pharma* dataset had 343 orthogonal subgraphs out of 404 nodes while the *Defense* has 2,451 orthogonal subgraphs out of 8,922 nodes. The relative ratio of 0.8490 versus 0.2747 indicated that the presumably less complex *Pharma* had relatively higher orthogonality, satisfying the aforementioned assumption. Increasingly greater differences

between graphs were also seen relative to their differences in orthogonal subset counts as well. The semi-orthogonal subsets $S^{8'}$ was the sole contributor to the extremely high $r(m_8)$ value, having $r(m_8^{8\prime})$ =1.1775E+09. This was because the semi-orthogonal pattern included several paths with long distances, exponentially increasing the number of possible paths within each semi-orthogonal subset. $S^{8'}$ showed the maximum complexity ratio for partition count m_3, albeit with less significant differences from other *pattern graphs*. It is also worth noting that $S^{8'}$ did not share the outlier characteristics of S^8 in m_0, m_1, and m_9; the ratio for these measures stayed similar to those of the whole graphs, indicating that the semi-orthogonality should be used for measuring verification strategy complexity paired with a specific set of measures.

To verify the similarity between the two graphs in terms of independent components, m_3 was measured for *pattern graphs* where a verification strategy graph was filtered by edges satisfying a specific fundamental pattern. Table 6 shows the number of pattern-specific partitions found in the *pharma* and *defense* graphs. The original connected subgraph measure m_3 was identical to m_3^1 while m_3^8 represented the orthogonal subset count. The difference between the two datasets was clearly visible with all six patterns having a minimum of 10.09 relative ratio. This was in sync with the assertion that *defense* requires more human cognition than *pharma* due to its complexity. It is also worth noting that the $S^{8'}$, a semi-orthogonality pattern, showed the largest difference between the datasets when $r(m_3^8)$ was in a lower end of the ratio spectrum. This could be interpreted as the weaker orthogonal connections exacerbating the gaps by providing more bridges to less complex orthogonal sets, which would more frequently be found in less complex datasets. This suggested that m_3 could be replaced with $m_3^{8\prime}$ for a more contextually relevant complexity indicator for human cognition in verification strategies.

Table 6 The number of partitions made by the fundamental patterns for each dataset and their ratio

Pattern	m_3^1	m_3^3	m_3^4	m_3^7	m_3^8	$m_3^{8\prime}$
Pharma	10	97	32	11	9	4
Defense	153	1,019	323	142	96	66
$r(m_3^k)$	15.30	10.51	10.09	12.91	10.67	16.50

The effects of fundamental patterns in the complexity measures were the most consistent in S3 and S4 through four out of the six indicators (m_0, m_1, m_8, and m_9). *Pattern graphs* from both patterns returned the lowest ratio between the two graphs with significant gaps against the others. The shared effect of S^3 and S^4 could be attributed to them having a single parameter; parameters are the basic building blocks of the verification requirements and excluding the possibility of multiple parameter interconnections minimized the difference between the two varying-sized verification strategies. This was most noticeable with the almost negligible differences in communicability (m_8), where the patterns do not have enough structural variations to warrant significant differences in their paths. The parameter hierarchy pattern S^7 was positioned as a midpoint between the verification-based patterns (S^3, S^4) and the orthogonality pattern (S^8) with $r(m_n^{3,4}) < r(m_n^7) < r(m_n^8)$ and was therefore deemed a less distinctive pattern for comparing verification strategy complexities.

The structurally limited $V \rightarrow 1P$ patterns (S^3 and S^4) had the advantage of discovering relatively smaller differences in graph complexity measures, shown by distinctive variations in their values compared to other fundamental patterns. Such disparities were also observed in closeness centrality m_5 and efficiency m_7 while their relative indifferences between graphs were explained by the shared use of the fundamental pattern set. Unlike the information centrality ratio $r(m_6^k)$ ranging from 0.0145 to 0.0985, $r(m_5^k)$ had a larger variance across different *pattern graphs* with the ratio ranging from 0.1628 to 1.1809. The closeness centrality was sensitive to the specific patterns more so than to the graphs, with $r(m_5^3)$ and $r(m_5^4)$ averaging at 0.1732 near the minimum threshold. This was also the case for efficiency m_7 with $r(m_7^3)$ and $r(m_7^4)$ averaging at 0.1820 against the maximum $r(m_7^k) = 1.1695\ when\ k = 8$. Both measures showed some degree of indicative power when used on the two *pattern graphs*, making them conditional auxiliary verification strategy complexity indicators.

Table 7 summarizes the eight complexity measures considered as the ordinal indicators for verification strategy complexity. The first three (m_0, m_1, and m_3) were baseline graph complexity measures based on heuristics of graph and verification strategy complexities. Size differences in *nodes* and *edges* were the basis of size-based heuristics and therefore considered as basic indicators of cognition complexity in verification strategies. The *subgraph count* was

conceptually related to the number of independent verification subtasks and therefore also considered as a viable indicator. The next three (m_6, m_8, and m_9) were the graph complexity measures cognizant of the verification strategy implementation process. *Information* centrality measured the information propagation efficiency effectively reflecting the information transfer during the verification process, showing significant differences between the two datasets while not being fully dependent on graph structures. *Communicability* was conceptually similar to cyclomatic complexity measuring the complexity of procedural processes such as verification strategies. The exponential nature of the measure made the difference between datasets exuberant, making this a less desirable complexity indicator when numerical accuracy is necessary. The *graph energy* measured information energy stored in the graph with similarities to the node count differences therefore also considered one of the effective indicators. The last two (m_5 and m_7) were conditional indicators that would be effective when calculated on *pattern graphs* for S^3 and S^4. The structurally simple $V \rightarrow 1P$ relationships revealed structurally innate differences in *closeness centrality* and *efficiency*, which were considered not significant when the whole graphs were compared. The orthogonality subsets S^8 and its variation $S^{8'}$ could also be utilized to fine-tune these indicators, while the use of fundamental patterns was not found to be non-essential in comparing the verification strategy complexity.

Table 7 List of the eight graph complexity measures considered as the verification strategy complexity indicators

ID	Name	ID	Name	ID	Name
m_0	*Node count*	m_6	$Centrality_{information}$	m_5	$Centrality_{closeness}$
m_1	*Edge count*	m_8	*Communicability*	m_7	*Efficiency*
m_3	*Subgraph count*	m_9	*Graph energy*		

4. Conclusions and Limitations

This paper showed verification strategies can be numerically analyzed in knowledge graph format, ordinally comparing two real-world industrial systems in terms of complexity. The smaller medical device system and the larger defense system both captured the complete set of requirements and verification activities employed in the development of the system. The larger system had a total of 8,922 verification entities connected by 17,319 relationships showcasing the necessity of the scale-free graph-based approach for verification planning and assessment. Due to its larger size and heavier interconnections, the defense system was determined by domain experts to have a more complex verification strategy; engineers would experience higher cognitive loads verifying the system.

Two verification strategy graphs were drawn from their respective verification matrix documents, and differences in their graph complexity measures were ordinally compared against the manually asserted premise of difference in verification complexity. This was done to detect possible indicators for verification strategy complexity; a graph complexity measure having clear differences between two strategies was assumed to be ordinally correlated to their cognitive complexity as well. A total of 25 measures were considered for the experiment, discarding 15 due to the structural characteristics of the verification strategy graphs. Ten remaining measures were calculated in two graphs and their *pattern graph* variations, measuring complexities on different verification scenarios in both. The experiment showed that there were five effective indicators of verification strategy complexities (*node*, *edge*, *subgraph* counts, *information centrality*, and *graph energy*) with communicability as a substitute measure. Traditionally dominant graph quality measures such as modularity and betweenness centrality were unable to distinguish the two verification strategies as both shared a set of elemental patterns, limiting their structural variability. Focusing on each pattern at a time, however, revealed that two fundamental patterns contributed to additional information gained from closeness centrality and efficiency. The (semi-)orthogonality subsets also provided a strong distinction between the two verification strategies. In summary, the fundamental patterns provided additional information but were not found to be essential in distinguishing two verification strategies with a significant complexity difference.

The limitation of small data points limited the findings to ordinal comparison; numerical correlations between the verification strategy complexity and the graph complexity measures were not found. Additional analysis with a larger number of verification strategies is planned for future work. Based on the findings of this paper, future work would focus on a set of smaller, artificial projects where manual complexity calculations are feasible. These values would be used as a golden answer set with machine learning models using the previously found complexity indicators as the

input, producing a regression formula for the automatic calculation of verification strategy complexities. The artificially generated datasets with verification sequence information would be used to compare and distinguish verification strategies with varying degree of data availability. This measure is expected to propose an optimal number of engineers required to implement the given verification strategy without overloading their cognitive capacities which can result in possible human errors.

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. CMMI-2205468.

References

Anderson, E.W., Potter, K.C., Matzen, L.E., Shepherd, J.F., Preston, G.A., Silva, C.T., 2011. A User Study of Visualization Effectiveness Using EEG and Cognitive Load. Computer Graphics Forum 30, 791–800. https://doi.org/10.1111/j.1467-8659.2011.01928.x

Ayres, P., 2006. Impact of reducing intrinsic cognitive load on learning in a mathematical domain. Applied Cognitive Psychology 20, 287–298. https://doi.org/10.1002/acp.1245

Ayres, P.L., 2001. Systematic Mathematical Errors and Cognitive Load. Contemporary Educational Psychology 26, 227–248. https://doi.org/10.1006/ceps.2000.1051

Brown, T.J.N., Mallion, R.B., Pollak, P., Roth, A., 1996. Some methods for counting the spanning trees in labelled molecular graphs, examined in relation to certain fullerenes. Discrete Applied Mathematics, Chemistry and Discrete Mathematics 67, 51–66. https://doi.org/10.1016/0166-218X(96)85158-4

Dörner, D., Güss, C.D., 2022. Human error in complex problem solving and dynamic decision making: A taxonomy of 24 errors and a theory. Computers in Human Behavior Reports 7, 100222. https://doi.org/10.1016/j.chbr.2022.100222

Engel, A., 2010. Verification, Validation, and Testing of Engineered Systems. John Wiley & Sons.

Estrada, E., Hatano, N., 2008. Communicability in complex networks. Phys. Rev. E 77, 036111. https://doi.org/10.1103/PhysRevE.77.036111

Fortunato, S., Latora, V., Marchiori, M., 2004. Method to find community structures based on information centrality. Phys. Rev. E 70, 056104. https://doi.org/10.1103/PhysRevE.70.056104

Gutman, I., 2001. The Energy of a Graph: Old and New Results, in: Betten, A., Kohnert, A., Laue, R., Wassermann, A. (Eds.), Algebraic Combinatorics and Applications. Springer, Berlin, Heidelberg, pp. 196–211. https://doi.org/10.1007/978-3-642-59448-9_13

Jung, S., Salado, A., 2023a. Verification Complexity: An Initial Look at Verification Artifacts, in: Conference on Systems Engineering Research (CSER).

Jung, S., Salado, A., 2023b. (Submitted) Emergent Knowledge Patterns in Verification Artifacts. Systems Engineering.

Kirsh, D., 2000. A Few Thoughts on Cognitive Overload. Intellectica 1, 19–51.

Latora, V., Marchiori, M., 2007. A measure of centrality based on network efficiency. New J. Phys. 9, 188. https://doi.org/10.1088/1367-2630/9/6/188

Newman, M.E.J., 2004. Detecting community structure in networks. Eur. Phys. J. B 38, 321–330. https://doi.org/10.1140/epjb/e2004-00124-y

Piccinini, G., Scarantino, A., 2011. Information processing, computation, and cognition. J Biol Phys 37, 1–38. https://doi.org/10.1007/s10867-010-9195-3

Salado, A., Kannan, H., 2019. Elemental patterns of verification strategies. Systems Engineering 22, 370–388. https://doi.org/10.1002/sys.21481

Salado, A., Kannan, H., 2018. A mathematical model of verification strategies. Systems Engineering 21, 593–608. https://doi.org/10.1002/sys.21463

Sinha, K., de Weck, O.L., 2013. Structural complexity quantification for engineered complex systems and implications on system architecture and design, in: International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. American Society of Mechanical Engineers, p. V03AT03A044.

Walden, D.D., Roedler, G.J., Forsberg, K., 2015. INCOSE Systems Engineering Handbook Version 4: Updating the Reference for Practitioners. INCOSE International Symposium 25, 678–686. https://doi.org/10.1002/j.2334-5837.2015.00089.x

Watson, A.H., Wallace, D.R., McCabe, T.J., 1996. Structured Testing: A Testing Methodology Using the Cyclomatic Complexity Metric. U.S. Department of Commerce, Technology Administration, National Institute of Standards and Technology.