An Improved Way for Measuring Simplicity During Process Discovery

Peer-reviewed author version

LIEBEN, Jonas; JOUCK, Toon; DEPAIRE, Benoit & JANS, Mieke (2018) An Improved Way for Measuring Simplicity During Process Discovery. In: Pergl, Robert; Babkin, Eduard; Lock, Russel; Malyzhenkov, Pavel; Merunka, V. (Ed.). EOMAS 2018: Enterprise and Organizational Modeling and Simulation, Springer International Publishing,p. 49-62.

DOI: 10.1007/978-3-030-00787-4_4
Handle: http://hdl.handle.net/1942/28410

An improved way for measuring simplicity during process discovery

Jonas Lieben[1,2], Toon Jouck[1], Benoit Depaire[1], and Mieke Jans[1]

[1] Hasselt University, Martelarenlaan 42, 3500, Hasselt, Belgium
jonas.lieben@uhasselt.be,
[2] FWO, Egmontstraat 5, 1000 Brussel, Belgium

Abstract. In the domain of process discovery, there are four quality dimensions for evaluating process models of which simplicity is one. Simplicity is often measured using the size of a process model, the structuredness and the entropy. It is closely related to the process model understandability. Researchers from the domain of business process management (BPM) proposed several metrics for measuring the process model understandability. A part of these understandability metrics focus on the control-flow perspective, which is important for evaluating models from process discovery algorithms. It is remarkable that there are more of these metrics defined in the BPM literature compared to the number of proposed simplicity metrics. To research whether the understandability metrics capture more understandability dimensions than the simplicity metrics, an exploratory factor analysis was conducted on 18 understandability metrics. A sample of 4450 BPMN models, both manually modelled and artificially generated, is used. Four dimensions are discovered: token behaviour complexity, node IO complexity, path complexity and degree of connectedness. The conclusion of this analysis is that process analysts should be aware that the measurement of simplicity does not capture all dimensions of the understandability of process models.

Key words: Understandability metrics, simplicity, process models, exploratory factor analysis, BPMN

1 Introduction

Many organisations are aware of the importance of becoming process-oriented. A first step is modelling the current processes [1]. This can be done by conducting stakeholder interviews or discovering the model from event logs [1, 2]. A graphical notation is used for expressing the process models. Examples of these notations are BPMN (Business Process Model and Notation) and Petri nets. The usefulness of the process models depends among other things on how understandable they are [3]. To this end, many researchers of the business process management (BPM) domain have proposed metrics for measuring different aspects of process model understandability. These metrics belong to different perspectives such as the organisational perspective and the control-flow perspective. The control-flow

perspective takes into account everything that is related to the execution order of activities [4].

In the domain of process discovery, models discovered using an algorithm are evaluated based on four quality dimensions including the simplicity [4]. Simplicity is related to Occam's Razor's principle, which implies that the simplest process model should be chosen for explaining the behaviour observed in the event log [4]. It is most of the times measured using the process model size, the structuredness and entropy [4, 5]. Also metrics such as the control-flow complexity have been proposed to measure simplicity [5]. Simplicity is strongly associated with the understandability metrics belonging to the control-flow perspective.

It is remarkable that there are many more understandability metrics proposed in the BPM literature compared to the number of simplicity metrics. It is possible that these simplicity metrics capture not all dimensions measured by the understandability understandability metrics from the BPM field. Therefore, we define the following research question: "To what extent do the simplicity measures used in process discovery cover the whole spectrum of control flow related understandability metrics in BPM?"

There are three contributions delivered by this paper:

- the identification of existing understandability metrics belonging to the control flow perspective. For this research article, we identified 18 existing metrics for measuring aspects of the understandability using a structured literature review. The metrics were implemented using the programming language R [6] and are publicly made available as an R package on CRAN[1]. This implementation was needed, because a software implementation for a part of the metrics was not publicly available. All metrics can be found in section 2.
- the discovery of the understandability metrics' underlying dimensions by the means of an exploratory factor analysis. We performed the analysis using BPMN models, which are both manually modelled and artificially generated. The BPMN notation is chosen, because it is one of the most popular ones used in the industry. The reason is that the notation is easier to comprehend than for example Petri nets [7, 8]. Section 4 contains all results of the factor analysis. The methodology is explained in section 3.
- An analysis of which dimensions are not measured by the current simplicity metrics. Based on this analysis, we propose an alternative way for measuring the simplicity in order that all dimensions are represented. This is explained in section 5.

2 Related work

Many metrics exist for the measurement of the understandability of process models. Both the field of business process management and the field of process discovery proposed several metrics in the literature. In the field of process discovery, simplicity is one of the four proposed quality dimensions and is strongly

[1] `https://cran.r-project.org/web/packages/understandBPMN/index.html`

associated with process model understandability. The other three are fitness, precision and generalisation [4]. The process model that is the easiest to comprehend while explaining all the observed behaviour should be chosen, if one wants to optimize for simplicity [4]. Simplicity is most of the times calculated using process model size, structuredness and entropy [4]. The process model size is equal to the number of nodes in a model [9, 10, 11]. The structuredness is related to the mismatches in gateways. If a model has a parallel split gateway combined with an exclusive join gateway, it scores worse in terms of structuredness than a model with matching gateways such as a parallel split gateway combined with a parallel join gateway [5]. The entropy refers to the distribution and use of different components of a process model. An example of this entropy is the connector heterogeneity [3, 11, 12, 13]. In addition to these metrics, the control flow complexity and the cyclomatic metric of McCabe are proposed for measuring simplicity [5]. The control flow complexity is a measure which takes into account the complexity of the behaviour of a process model stemming from the use of different gateways and the number of outgoing sequence flows of these gateways [3, 14, 5, 15, 11]. The cyclomatic metric takes into account the number of activities and the complexity of the behaviour resulting from exclusive gateways [5].

In the field of BPM, understandabillity is defined as the extent to which the reader can make correct conclusions about the process model [13]. Many more metrics are proposed for measuring the understandability of process models in the field of BPM in comparison with the field of process discovery. Not all metrics are related to the control-flow perspective. Examples are the connectivity level between pools and the number of swimlanes and pools [16, 17, 12], which belong to the organisational perspective. In our study, we identified 18 understandability metrics related to the control-flow perspective. These metrics (column 1) together with their references (column 2) can be found in table 1.

3 Research methodology

In order to reach the results of our research paper, we performed several steps. The first step was the identification of existing understandability metrics belonging to the control-flow perspective (section 3.1). The second step was gathering the data, which consisted of BPMN models (section 3.2). The third and last step was conducting the factor analysis (section 3.3).

3.1 Metrics identification

We conducted a literature review of academic journals and conference papers for identifying existing understandability metrics of the BPM domain. To search for the scientific articles, Google Scholar was used with the keywords "understandability", "complexity", "BPMN", "process models", "metrics" and "influence". This resulted in 68 articles from journals and conference proceedings and one

Metric	References
Process model size	[9, 10, 11, 18]
Number of empty sequence flows	[19]
Number of duplicate tasks	[20]
Density	[3, 11]
Coefficient of network connectivity	[3, 11, 12]
Average connector degree	[3, 11]
Maximum connector degree	[3, 11]
Sequentiality	[3, 11, 12, 15]
Cyclicity	[11, 13]
Diameter	[3, 11, 12, 13]
Depth	[9, 11, 15, 21]
Token split	[3, 11, 13]
Control flow complexity	[3, 14, 15, 11]
Connector mismatch	[3, 11, 15, 21]
Connector heterogeneity	[3, 11, 12, 13]
Separability	[3, 11, 13]
Structuredness	[3, 11, 13]
Cross-connectivity	[22]

Table 1. Understandability metrics and references with definition

doctoral thesis. All found articles were read and the proposed understandability metrics were listed. Only the understandability metrics belonging to the control flow perspective were selected. The journals and conference proceedings which had a relevant metric can be found in table 2. Since a software implementation of some metrics lacked, we implemented all metrics as an R package [6].

Journals
Business & Information Systems Engineering
Decision Support Systems
IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans
IEEE Transactions on Industrial Informatics
International Journal of Computer Science and Network Security
Conference Proceedings
Advanced information systems engineering
Proceedings of IEEE International Conference on Cognitive Informatics
Proceedings of International Conference on Business Information Systems

Table 2. Journals and conference proceedings with relevant metrics

3.2 Data

The sample of observations used for the factor analysis consisted of two types of models. Models of the first type were made by people. The source is BPM AI

(Business Process Management Academic Initiative). This repository contains a large number of models created by students and staff of academic institutions [23]. We selected the models which were made in BPMN 2.0 and which had a connectedness of at least 50%. Afterwards, the models were filtered on the occurrence of boundary events and on modelling mistakes such as unconnected activities. These models are not within the scope of this analysis, because we are focussing on the understandability of models within the context of a process discovery algorithm. The current process discovery algorithms cannot yet discover processes with boundary events or processes with many unconnected activities. The number of resulting models after filtering is 4150.

Models of the second type were created using the PTandLogGenerator [24]. One sample of 300 models was generated. The maximum number of activities was set to 20. The minimum and mode were respectively 2 and 10 activities. This resulted in process trees which were converted to BPMN models. As process trees have a different notation for representing process models than BPMN, the translated BPMN models can have a representational bias [25]. Mismatches in gateways can for example not be represented in process trees. The bias will not pose a problem, because the first type of models is directly made in BPMN.

We chose to use two types of models, in order to increase the generalisation of our analysis. As we want to get an insight into the underlying dimensions of the understandability of models discovered using a process discovery algorithm, it is not a good approach to only use models made by people. However, not all possible constructs can be generated using the PTandLogGenerator such as mismatches in gateways. Hence, we chose to combine BPMN models created by process modellers with automatically generated models.

After having gathered all the models, the metrics were calculated for each model and some descriptive statistics such as the mean and standard deviation were calculated. These statistics were useful for getting a first overview of the data. The descriptive statistics can be found in table 3. We included the mean, median, standard deviation, minimum and maximum.

A couple of interesting patterns can be discovered from the descriptive statistics. There are some models which can be considered an outlier in terms of size, because the average is bigger than the median. Not many models have empty sequence flows or duplicate tasks. The models from the PTandLogGenerator have even no empty sequence flows, but some of them have duplicate tasks.

The density is rather small, because the density ranges from zero to one with one representing a dense model. This metric represents the percentage of sequence flows compared with the theoretical maximum number of sequence flows [11, 3]. The density is correlated with the sequentiality which determines how sequential the process is. As the density is rather small for most models, the processes are to some extent sequential.

Processes do have non-sequential parts which are gateways or activities with more than three connected sequence flows. On average, a gateway or activity with multiple incoming or outgoing sequence flows has three connected sequence flows. In addition, the models have sometimes a mismatch in gateways, even

Table 3. Descriptive statistics for each metric of all process models included in the sample

Metric	Mean	Median	Std	Min	Max
Size	21.624	18	15.378	1	134
# empty sequence flows	0.021	0	0.223	0	7
# duplicate tasks	0.531	0	1.861	0	41
Density	0.09	0.062	0.112	0.001	1
Coefficient of network connectivity	0.976	1	0.213	0.037	2.95
Average connector degree	2.727	3	1.062	0	8
Maximum connector degree	3.112	3	1.454	0	12
Sequentiality	0.336	0.286	0.26	0	1
Token split	1.208	0	1.984	0	42
Control flow complexity	6.26	4	7.178	0	129
Connector mismatch	1.369	1	1.819	0	34
Connector heterogeneity	0.387	0	0.444	0	1
Cyclicity	0.069	0	0.151	0	0.897
Diameter metric	10.929	8	9.507	0	85
Depth metric	1.392	1	1.352	0	13
Separability metric	0.183	0.077	0.246	0	1
Cross-connectivity	0.104	0.077	0.103	0	0.75
Structuredness	0.847	0.966	0.286	0	1

though this number is also quite small. The number of activities being part of an explicit loop is limited. Even more than 50% of the processes do not have any explicit loops. Some parts of the processes are also separable, because the separability metric is on average 0.183. The combination of this metric, with the results from the depth and sequentiality indicate that there are several gateways used in the process models. Even though there are several gateways used, they are most of the times modelled in the right way. This conclusion is derived from the structuredness, which is on average 0.847. This means that the gateways most of the time match and that most explicit loops are modelled using exclusive gateways.

3.3 Factor analysis

A first step when performing a factor analysis is choosing the type of factor analysis. Several types of factor analyses exist and are used for different purposes. We chose an R-type common factor analysis, because the main goal is the discovery of the underlying dimensions of the metrics. This type only considers the shared variance, which defines the structure of the variables [26].

The sample size of 4450 observations is sufficient to perform an exploratory analysis. It is recommended to have at least 50 observations and at least 20 observations per variable to perform a factor analysis [26]. Therefore, this requirement is fulfilled.

An important assumption when performing a factor analysis is that there is enough intercorrelation between the variables [26]. We performed two tests to validate this assumption: the Bartlett's Test of Sphericity and the Measure of Sampling Adequacy (MSA) values [26]. When the Bartlett's Test of Sphericity is statistically significant, it indicates that there is enough overall correlation between the variable to perform a factor analysis. The overall and individual MSA values indicate also whether there is enough intercorrelation. As MSA values increase, when the sample size increases [26], we used a cut-off of 0.6. Variables which had a lower MSA value were discarded, because they do not exhibit enough intercorrelation with the other variables. This is unacceptable in the context of a factor analysis and reduces the quality of the factor solution.

The number of factors is determined using the scree test. After having determined the number of factors, the factor analysis was performed iteratively. During each iteration, the communalities of the variables were assessed. The communality determines the amount of variance which is included in the factor solution [26]. Some variables were again left out, because their communalities were smaller than 0.3 and this can decrease the reliability of the analysis [27].

To make the interpretation easier, a factor rotation was performed. The chosen rotation was the varimax rotation. As this is an orthogonal rotation, the factors are not correlated anymore, which is in contrast with the oblique factor rotation methods [26]. We chose an orthogonal factor rotation, because this allows the development of new uncorrelated scales for measuring the understandability of process models with regards to the control-flow perspective [26].

The removal of the variables due to low communalities and low MSA values does not pose a problem in the context of this factor analysis, because we are interested in the underlying dimensions of the understandability metrics. The metrics which are not included into the factor analysis can still be regarded as separate dimensions for each omitted variable.

4 Results

The standard deviations of the variables in table 3 indicate that we can perform a useful factor analysis, because a necessary condition for a factor analysis is a sufficient amount of variance in the data [26]. In addition, there is enough intercorrelation between the variables, because the Bartlett's Test of Sphericity is statistically significant at 0.01. The overall MSA value is 0.77 and the individual MSA values range from 0.61 to 0.87. We discarded no variable due to a low MSA value.

We determined the number of factors using the scree test by making a plot. The eigenvalues are assigned to the Y-axis, while the factor number is plotted on the X-axis. The point where the curve starts to straighten out indicates the number of factors [26]. The scree test criterion implies that four factors should be chosen (figure 1). Note that the latent root criterion, which states that the number of factors should equal to the number of factors with eigenvalues bigger

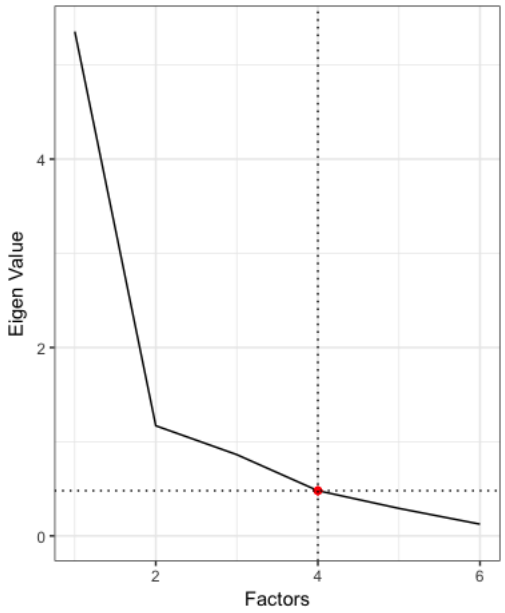

Fig. 1. Scree plot for determining number of factors

than one [26], implies that two factors should be chosen. However, this number of factors is deemed too low to make a useful interpretation.

Table 4. Factor communalities

Metrics	Communalities
Size	0.706
Density	0.613
Coefficient of network connectivity	0.756
Average connector degree	0.995
Maximum connector degree	0.859
Sequentiality	0.527
Token split	0.483
Control flow complexity	0.676
Connector mismatch	0.355
Connector heterogeneity	0.331
Cyclicity	0.460
Diameter	0.995
Depth	0.558
Separability	0.352
Cross-connectivity	0.337

Not all variables are used for the factor analysis, because some variables had communalities which were too low. However, the factor analysis was again executed using the 15 variables which had communalities bigger than 0.3. These variables with their corresponding communalities can be found in table 4. The variables with a communality lower than 0.3 are the number of empty sequence flows, the number of duplicate tasks and the structuredness.

Table 5. Factor loadings

	Factor 1	Factor 2	Factor 3	Factor 4
Size	0.810			
Density				0.745
Coef. of network conn.		0.531		0.450
Avg. connector degree		1.174		
Max. connector degree		0.888		
Sequentiality		−0.621		
Token split	0.705			
CFC	0.846			
Connector mismatch	0.613			
Connector heterogeneity	0.493			
Cyclicity			0.693	
Diameter			0.982	
Depth			0.489	
Separability	−0.389			
Cross-connectivity	−0.440			

In table 5, the factor loadings can be found. The factor loadings determine how the metrics correlate with a given factor [26]. The higher the absolute value of the loading, the bigger the correlation. We removed the loadings which had an absolute value lower than 0.35. These loadings are considered insignificant, given our big sample size [26].

5 Discussion

Four main dimensions are discovered as a result of the factor analysis. It should be emphasized that some metrics were not included in the factor analysis, because they measure different things than what is captured by the main dimensions. These metrics are the number of empty sequence flows, the number of duplicate tasks and the structuredness. Each metric can be considered a different dimension for the purpose of this factor analysis.

We give each main dimension a label and explain why the metrics belong to the factors. Also an intuitive interpretation of what the dimension captures is given. Afterwards, the dimensions are related to the current simplicity metrics.

5.1 Dimension 1: Token behaviour complexity

The size, token split, control flow complexity, connector mismatch, connector heterogeneity, seperability and cross-connectivity all belong to dimension one. We label it token behaviour complexity. The token behaviour complexity includes the number of tokens which are consumed by activities. A token consumption happens when an activity of the model is executed. The bigger the size of a model, the more token consumptions can occur, because there are more activities. The token split measure indicates in which activities and gateways the tokens are split [3, 11, 13]. If tokens are split, more token consumptions can occur. The control flow complexity is closely related to the token split but assigns different weights to the type of gateways [14]. When a mismatch occurs, the token behaviour can become more complex as well. If for example a parallel split gateway is matched with an exclusive join gateway, multiple tokens will continue flowing through the model and many activities will be repeated and executed in parallel. The connector heterogeneity also belongs to this dimension, because if there are several types of connectors, the complexity of the token behaviour increases [3, 11, 12, 13]. The higher the heterogeneity, the higher the probability that there are inclusive and parallel gateways present in the model. The separability measures how easy it is to split the model into several independent parts [3, 11, 13]. If it is easier to split the model, the token behaviour is less complex, and therefore this metric is negatively related to the token behaviour complexity. Also the cross-connectivity is negatively related, because the token behaviour becomes more complex when the value of this metric decreases. In that case, there is a higher probability of several types of gateways.

The token behaviour complexity captures **the number of token consumptions** for one process execution. In addition, it takes into account **the complexity of the routing of the tokens**. When exclusive gateways are used, the routing becomes more complex, which reduces for example the separability and cross-connectivity. The token behaviour complexity is high, when the process model contains many gateways and activities.

5.2 Dimension 2: Node IO complexity

The coefficient of network connectivity, the average and maximum connector degree and the sequentiality belong to dimension two. We give this dimension the label node incoming outgoing complexity or node IO complexity. The node IO complexity takes into account the number of incoming and outgoing sequence flows for a given node. If there are more incoming and outgoing sequence flows on average for activities and gateways, the overall number of sequence flows increases and hence the coefficient of network connectivity increases. The same reasoning can be applied for the average and maximum connector degree. The sequentiality negatively loads on dimension two, because if a process is sequential, it consists only of activities with exactly one incoming and outgoing sequence flow [3, 11, 12, 15]. This means that there are less outgoing and incoming flows of nodes.

The node IO complexity captures the **number of connected sequence flows with activities and gateways**.

5.3 Dimension 3: Path complexity

Dimension three is labelled the path complexity. This dimension includes three metrics: cyclicity, diameter and depth. We define a path in a similar way as a path in graph theory [28] and not as a trace. This means that if there is a parallel or inclusive split gateway, only one of the outgoing sequence flows is chosen and becomes part of the path. In addition, each path starts with a start event and ends with an end event. The diameter is the length of a path [3, 11, 12, 13] and the bigger the length, the larger a path. When there are loops, the path becomes longer and more complex [11, 13]. The complexity of the paths also increases when the depth increases. If the depth increases, there are more gateways, which make the path longer.

We define the path complexity as **the length of a path in the process model allowing one repetition of a loop**. This definition gives the most weight to the diameter, while taking into account cyclicity and depth just as the factor loadings do. When the path complexity is high, there is a long path present in the model, or a path with many and long loops.

5.4 Dimension 4: Degree of connectedness

Dimension four gets the label degree of connectedness. This dimension includes the density and the coefficient of network connectivity. A process model diagram is more connected, when it has more sequence flows cet. par. the process model size. The density of a model captures this, because it is calculated by dividing the number of sequence flows with the process model size [3, 11]. Also the coefficient of network connectivity is linked with the dimension. This metric is calculated in a similar way, but the denominator is replaced with the theoretical maximum number of sequence flows. The theoretical maximum is calculated by multiplying the process model size with the process model size minus one [3, 11, 12].

We define the degree of connectedness in the same way as the **density**. When there are many **flows given the process models size**, the degree of connectedness is high.

5.5 Relation of understandability dimensions with simplicity metrics

The simplicity of a process model is often measured using the size of the process model, the structuredness and the entropy [4]. Both the size and the entropy, which is named the connector heterogeneity in table 5 belong to dimension one, token behaviour complexity. The structuredness of the process model is not included in the factor analysis, because the communality is too low. The structuredness can be regarded as a separate dimension.

In addition to the three most used simplicity metrics, the control flow complexity and the cyclomatic metric of McCabe are proposed [5]. The control-flow

complexity is part of dimension one as well. The cyclomatic metric of McCabe takes into account the number of activities and the complexity in behaviour resulting from exclusive gateways. These concepts are mainly related to the token behaviour complexity.

We can conclude from the previous paragraphs that most simplicity metrics are associated with the token behaviour complexity. The current simplicity metrics all omit the node IO complexity, the path complexity and the degree of connectedness. Also empty sequence flows and the number of duplicate tasks are omitted. As empty sequence flows are not part of a process model resulting from a discovery algorithm, this metric can be considered irrelevant in this context.

To calculate the simplicity, we propose that at least 6 metrics are used: one for each dimension and two for the metrics which are not part of the factor analysis. We recommend to use the metrics with the highest loadings until further research is conducted. These metrics are control-flow complexity for token behaviour complexity, average connector degree for node IO complexity, diameter for path complexity and density for the degree of connectedness. In addition, the number of duplicate tasks and structuredness should be used.

6 Conclusion

When someone has discovered a process model using a process discovery algorithm, the model should be evaluated using several quality dimensions. The quality dimension simplicity is often measured with the size of a process model, the structuredness and the entropy. In addition, metrics such as the control-flow complexity and the cyclomatic metric of McCabe are proposed to measure simplicity. The current metrics do not take all dimensions of the process model understandability into account even though simplicity is closely related to the process model understandability. This conclusion can be made based on the results of an exploratory factor analysis of 18 understandability metrics, which all belong to the control-flow perspective.

Four dimensions were discovered: token behaviour complexity, node IO complexity, path complexity and degree of connectedness. Three of the 18 metrics were not included in the analysis, because these variables had too low communalities. These metrics were the number of empty sequence flows, the number of duplicate tasks and the structuredness. As they measure different things than the discovered dimensions, they can each be considered a separate dimension.

Most simplicity metrics only load on the token behaviour complexity dimension. This is the case for the process model size, the entropy, the control flow complexity. The cyclomatic metric of McCabe is also related to the token behaviour complexity. Structuredness was not part of the factor analysis. A better approach for calculating the simplicity is using six metrics, of which four belong to the discovered dimensions.

There is still room for further research. The discovered dimensions can be validated using a confirmatory factor analysis. This analysis can be done with models resulting from several discovery algorithms. It is also not yet clear how big

the correlation is between the dimensions and the simplicity metrics. Moreover, this paper is a starting point for the development of a single understandability metric which captures all dimensions belonging to the control-flow perspective.

References

[1] van der Aalst, W.M.P.: Business Process Management: A Comprehensive Survey. ISRN Software Engineering **2013** (2013) 1–37

[2] van der Aalst, W., Reijers, H., Weijters, A., van Dongen, B., Alves de Medeiros, A., Song, M., Verbeek, H.: Business process mining: An industrial application. Information Systems **32**(5) (July 2007) 713–732

[3] Reijers, H.A., Mendling, J.: A study into the factors that influence the understandability of business process models. IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans **41**(3) (2011) 449–462

[4] van der Aalst, W.M.P.: Process Mining: Discovery, Conformance and Enhancement of Business Processes. Springer-Verlag, Berlin Heidelberg (2011)

[5] Lassen, K.B., van der Aalst, W.M.: Complexity metrics for Workflow nets. Information and Software Technology **51**(3) (2009) 610–626

[6] Ihaka, R., Gentleman, R.: R: A Language for Data Analysis and Graphics. Journal of Computational and Graphical Statistics **5**(3) (September 1996) 299–314

[7] Sarshar, K., Loos, P.: Comparing the Control-Flow of EPC and Petri Net from the End-User Perspective. In: Business Process Management. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg (September 2005) 434–439

[8] Recker, J.C., Dreiling, A.: Does it matter which process modelling language we teach or use? An experimental study on understanding process modelling languages without formal education, Toowoomba (2007)

[9] Laue, R., Gruhn, V.: Complexity Metrics for business Process Models. In: Business Information Systems, Klagenfurt, Austria (January 2006)

[10] Petrusel, R., Mendling, J., Reijers, H.A.: How visual cognition influences process model comprehension. Decision Support Systems **96**(Supplement C) (April 2017) 1–16

[11] Mendling, J.: Detection and prediction of errors in EPC business process models. PhD thesis, Wirtschaftsuniversitt Wien Vienna (2007)

[12] Fernndez-Ropero, M., Prez-Castillo, R., Caballero, I., Piattini, M.: Quality-driven business process refactoring. In: International Conference on Business Information Systems (ICBIS 2012). (2012) 960–966

[13] Mendling, J., Strembeck, M.: Influence Factors of Understanding Business Process Models. In: Lecture Notes in Business Information Processing. Volume 7. (May 2008) 142–153

[14] Cardoso, J.: Control-flow complexity measurement of processes and Weyukers properties. In: 6th International Enformatika Conference. Volume 8. (2005) 213–218

[15] Figl, K.: Comprehension of Procedural Visual Business Process Models: A Literature Review. Business & Information Systems Engineering **59**(1) (February 2017) 41–67
[16] Polani, G., Cegnar, B.: Complexity metrics for process models A systematic literature review. Computer Standards & Interfaces **51**(Supplement C) (March 2017) 104–117
[17] Gschwind, T., Koehler, J., Wong, J.: Applying patterns during business process modeling. Business process management (2008) 4–19
[18] Pavlicek, J., Hronza, R., Pavlickova, P., Jelinkova, K.: The Business Process Model Quality Metrics. Lecture Notes in Business Information Processing, Springer, Cham (June 2017) 134–148
[19] Gruhn, V., Laue, R.: Reducing the cognitive complexity of business process models. (June 2009) 339–345
[20] La Rosa, M., Wohed, P., Mendling, J., Ter Hofstede, A.H., Reijers, H.A., van der Aalst, W.M.: Managing process model complexity via abstract syntax modifications. IEEE Transactions on Industrial Informatics **7**(4) (2011) 614–629
[21] Muketha, G.: Complexity Metrics for Measuring the Understandability and Maintainability of Business Process Models using Goal-Question-Metric (GQM). International Journal of Computer Science and Network Security **8**(5) (2008) 219–225
[22] Vanderfeesten, I., A. Reijers, H., Mendling, J., Aalst, W.M.P., Cardoso, J.: On a Quest for Good Process Models: The Cross-Connectivity Metric, Montpellier, France (June 2008)
[23] Kunze, M., Berger, P., Weske, M., Lohmann, N., Moser, S.: BPM Academic Initiative-Fostering Empirical Research. In: BPM (Demos). (2012) 1–5
[24] Jouck, T., Depaire, B.: Generating Artificial Data for Empirical Analysis of Control-flow Discovery Algorithms: A Process Tree and Log Generator. Business & Information Systems Engineering (March 2018) 18
[25] van der Aalst, W.: On the Representational Bias in Process Mining, IEEE (June 2011) 2–7
[26] Joseph Hair, William Black, Barry Babin, Rolph Anderson: Multivariate Data Analysis. Number Seventh edition. Pearson Education Limited
[27] Child, D.: The Essentials of Factor Analysis. A&C Black (June 2006) Google-Books-ID: rQ2vdJgohH0C.
[28] Sim, K.A., Tan, T.S., Wong, K.B.: On the shortest path in some k-connected graphs. (2016) 050010