Complexity Analysis of Business Processes

Martin Ibl
Institute of System Engineering and Informatics
Faculty of Economics and Administration
University of Pardubice
Pardubice, Czech Republic
Email: martin.ibl@upce.cz

Žaneta Boruchová
Faculty of Economics and Administration
University of Pardubice
Pardubice, Czech Republic

Abstract— **Complexity is a term that is currently used not only in research articles but also in the methodologies and standards used to manage information and projects. The complexity can represent the size of a system, process, program or project, the number of functions and the cost of their acquisition, operation and maintenance. Within the context of this work, the complexity is a variable that represents system properties such as legibility, clarity, comprehensibility, usability, modifiability, easiness of implementation or predictability. With increasing complexity, these characteristics deteriorate and the system becomes more difficult and less effective, for example by increasing its cost, increasing the use of resources, increasing the time and expenses needed for training or maintenance, which may result in a loss of profit for businesses. Different systems or their parts can be compared by complexity measurement. If the system contains elements or bindings that are not necessary, simpler solutions can be created so that its complexity is minimal. This paper presents the process of quantification of complexity in Petri nets and then compares it with other existing approaches. The advantage of the presented complexity measure is the possibility to examine this variable at different levels of system load.**

Keywords—***complexity; process analysis; complexity measure; Petri nets; entropy***

I. INTRODUCTION

The concept of complexity becomes one of the most important concepts of contemporary science and implies the complicacy or size of a system. Such a system is difficult to define, comprehend, modify or use. Complexity refers to the intrinsic, implicit nature of the system, which affects both the properties of the interacting components and the nature of their interaction. The complexity of the system includes aspects such as uncertainty, fluctuation, singularity, internal dynamics, connectivity, and more. Descriptive complexity is how the system appears to an observer standing outside, usually static views. Natural complexity is the intrinsic, real matter of the system and forms the essence of the system. Both types of complexity are linked to information, both the information needed to describe the system and information to clarify uncertainty are embedded in it. These two complexities are in conflict. If we want to limit one, then the other is likely to grow or at best remain the same. This mutual exchange is one of the most important methodological bases of system science.

Business process is a workflow or activity that represents a dynamic component of a system. Each organization is an essentially organized set of processes and activities that interact with each other, run across organizational units and respond to various stimuli from the internal and external environment. In processes, inputs and resources are transformed into outputs that are valued by the process's customer. Processes exist within and between organizations. There is always a flow of work and activity from one person to another. The core of processes is the creation of value or benefit for the organization's customers. The most common division of processes is according to who is their customer and the added value they bring to them. The customer of the process may be a customer of a company, its employee or manager. Processes in an organization are divided into main, supportive, and managing. Major processes create value or benefit to an organization's customer, create a product or service. Supporting processes are all processes whose sole purpose is to ensure the functioning of the main processes and organization. Management processes and activities are all activities that coordinate, manage, organize and plan everything else. Business processes have their own complexity, which, if not controlled, can continue to increase over time, making processes prone to error, difficult to understand and maintain.

In recent years, several scientists have suggested several metrics that can be used to measure and thereby manage the complexity of business processes. Processes are not static. They are constantly undergoing revisions, adaptations, changes and adjustments to meet the needs of end-users. The complexity of these processes and their continuous development make it very difficult to ensure their stability and reliability. As the simplest measures of complexity can essentially be considered the size or length of the process.

Organizations are increasingly struggling with the issue of managing business processes, workflows, and more recently with web processes. One of the important aspects of business processes that has been overlooked is their complexity. A high complexity of processes can result in poor comprehensibility, errors, defects and exceptions, resulting in processes requiring more time to develop, test, and maintain. For this reason, it is necessary to avoid excessive complexity. The measurement of business processes is the task of empirical and objective assignment of numbers, due to their characteristics and in such a way as to describe them. Required attributes include complexity, cost, maintainability, and reliability. Metrics should be evaluated by theoretical (or empirical) validation principle, for example in terms of Weyuker's properties [1], to ensure that the metric is consistent and effective. Business process management systems, referred to as BPMS, provide the basic infrastructure for defining and managing business processes. BPMS, such as workflow systems, has become a serious competitive factor for many organizations that are increasingly struggling with the issue of managing business applications, workflows, web services, and web processes. Business processes promise to mitigate several of the current challenges in infrastructure such as data, applications, and process integration. With the emergence of web services, the workflow process management system becomes crucial to support, manage and receive processes, both between businesses and within the enterprise. The measurement process deals with the derivation of a numeric value for process attributes. Measurements can be used to improve productivity and process quality. Designing and improving processes is a key aspect for businesses to remain competitive in today's market. Organizations are forced to improve their business processes because customers require better products and services. The business process consists of a series of activities, tasks or services that together lead to the goal.

The goal of this contribution is to compare existing measures of complexity with a previously defined measure [2] and evaluate its advantages and disadvantages. It is essentially an empirical validation of this measure. For the comparison process itself, the two most widely used measures were used, namely the McCabe's cyclomatic measure and Cardoso's CFC measure.

II. State of the Art

Analysing complexity at all stages of process lifecycle helps to avoid the disadvantages associated with high complexity of processes. At present, organizations have not accepted complexity metrics as part of their process management projects. As a result, simple processes can be designed unnecessarily complex. Using complexity analysis helps design and implement processes and workflows that are more simple, reliable, and robust. In-depth analysis is needed to correct defects in high complexity process parts. There are three questions that are often asked when measuring the complexity of a process [3]:

How difficult is the process to describe?

How difficult is the process to create?

What is the level of organization?

Complexity measurements can be grouped into the following categories depending on which question they are trying to answer:

- The difficulty of the description, typically measured in bits, such as information, entropy, algorithmic complexity, minimum length of description, Fisher information, Rényi entropy, length of code, Chernoff's information, Lempel-Ziv complexity, dimension and fractal dimension;
- The difficulty of creating, working with time, currency, or energy, such as computational complexity, time computational complexity, spatial computational complexity, information-based complexity, logical depth, thermodynamic depth and cost;
- A degree of organization that can be divided into the difficulty of describing the organizational structure and the amount of information divided between the parts of the system as a result of this organizational structure. This category includes, for example, stochastic entropy, sophistication, effective complexity, real complexity, ideal complexity, hierarchical complexity, schema length, homogeneous complexity, grammar complexity, information exchange algorithm, channel capacity or correlation.

Measurement has a long tradition and is a basic discipline in any type of engineering. Engineers have to be experienced in estimating and valuing, which means understanding the activities and risks associated with process development, forecasting and asset management, risk management, reliable delivery and proactive management to avoid a crisis. There is no single metric that would measure the complexity of the process. One of the most sophisticated methodologies to analyse complexity of processes has been created by Cardoso [4], which identifies four main views of complexity levels, namely complexity of activities, called AC, flow control complexity, also called control-flow, data stream complexity denoted as DFC and complexity of resources, labelled RC. The complexity of the AC simply counts the number of activities that the process has. While this metric is very simple, it is important to complement other forms of complexity. While control-flow complexity can be very low, the complexity of AC can be very high. For example, a sequencing process that has thousands of activities has control-flow complexity equal to zero, while its AC complexity is 100. The control-flow complexity is influenced by the design process. It is necessary to consider the existence of XOR, OR and AND operators. The complexity of DFC increases with the complexity of data structures, the number of formal activity parameters, and mapping between activity data. The metric may consist of several sub-metrics that include data complexity, complexity of the interface, and complexity of the integration interface. While the first two sub-metrics relate to static data aspects, the third metric is more dynamic in nature and is focused on data dependencies between different process activities. The RC complexity concerns process activities that need access to resources. Source is defined as any entity (e.g. human resources, IS resources, IT resources) that the activity requires during execution, such as document, database, printer, external application or role. Resources can be structured into the organization context. The structure that is used to form different types of resources can be analysed to determine its complexity. This analysis can help managers reduce administrative costs and optimize resource usage. The CFC metric can be used to analyse the complexity of business processes, as well as the workflows and processes associated with the website. The metric is validated using Weyuker's properties [1, 5], which provide an important basis for classifying complexity measures to determine whether they can be qualified as good, structured, and complex.

Other very popular complexity measure is the so-called cyclomatic complexity (MCC) defined by McCabe [6]. Since its development, MCC has been one of the most promising software metrics. The resulting empirical knowledge base has enabled software developers to calibrate their own software measurements and gain some understanding of its complexity. Software metrics are often used to obtain a quantitative expression of program complexity. They cannot be confused with the complexity of algorithms that aim to compare the performance of the algorithm. It has been found that software metrics are useful in reducing software maintenance costs by assigning a numeric value that reflects the ease or difficulty with which the program module can be understood. MCC is a measure of the number of linearly independent paths in the program. The intention is independence of language and language format. The MCC bears an indication of the complexity of the program flow. From the module control representation graph, it was found that MCC is a reliable indicator of complexity in large software projects. This metric is based on the assumption that the complexity of the program relates to the number of control channels within the program. For example, a ten-line program with ten assignment commands is more comprehensible than a ten-line program with ten if-then commands. The MCC is defined for each module as e - n +2, where e and n is the number of edges and nodes in the control-flow graph. These graphs describe the logical structure of the software modules. Nodes represent computational commands or expressions, and the edges represent handover between nodes. Each possible realizable path of the software module has a corresponding path from the input to the output node of the control-flow graph of the module. An MCC value 10 indicates a simple program without a high risk, a value between 11-20 indicates a more complex program with a moderate risk, and a value between 21 to 50 indicates a complex high risk program.

Gruhn and Laue [7] have suggested a cognitive weight for business process models. This metric, referred to as CFS, is an adaptation of cognitive functional size. Cognitive degrees of complexity are based on cognitive informatics. Cognitive metrics suggest that there are three factors that lead to the complexity of software architecture, model input data, and model output data. This means that cognitive complexity is a function of these three factors. This metric is intended for use with enterprise-class business process models that emphasize visual communication with users but offer minimal formal semantics. The main limitation of this metric is that it ignores two of the three factors that involve cognitive complexity, namely inputs and outputs, and focuses only on flow control. They also suggested customizing the metric of the flow of information for business processes, and unlike Cardoso's IC, this metric does not include the length of the process.

Lassen and van der Aalst [8] have suggested three levels of complexity for the Petri net subclass, called a workflow network. Extension of Cardoso metric ECaM, extended ECyM cycling metrics, and structured SM metrics. ECaM extends CFCs by being tailored to support Petri nets. These metrics were implemented within Prom, a business process measure that focuses on monitoring BAM's business activities.

Vanderfeesten [9] proposed a metric called Cross-Connectivity, labelled CC, based on cognitive complexity. It is the predictive error rate that measures the strength of the bonds among the elements of the process model. It is based on the hypothesis that process models are more understandable and contain fewer errors if they have a high cross-linking CC. In addition to predicting errors, it can also measure the comprehensibility of the business process model. This metric has been empirically evaluated using Spearman's correlation coefficient and multidimensional logistic regression.

Mendling and Neumann [10] have suggested six metrics for errors that are closely related to complexity. These metrics are based on graph theory and include size, separability, context, structure, cyclicality and parallelism. Increasing the size increases the probability of an error. Increasing separability, context, and structure means decreasing the probability of error. Increasing cyclicality and parallelism also increases the likelihood of error.

III. MEASUREMENT OF COMPLEXITY IN PETRI NETS

The Petri net is a mathematical tool for modelling and simulating discreet dynamic event-driven systems and consists of places, transitions, and oriented edges connecting places and transitions. Places may contain tags that are called tokens. The number of tokens at the given places indicates the current state of the system. Transitions represent possible activities that can change the state of the system. Transitions triggers tokens from input to output. The Petri Net provide a visual method for examining the properties of the system.

The following types of Petri nets have been created successively:

- Condition / Event Petri nets, referred to as C / E;
- Place / Transition Petri nets, referred to as P / T;
- P / T Petri nets with inhibiting edges;
- P / T Petri nets with priorities;
- Timed Petri nets, referred to as TPNs;
- Colored Petri nets, referred to as CPN;
- The hierarchical Petri nets, referred to as HPN.

The Petri P/T network, which will be used in the context of this work, consists of places, transitions, oriented edges, capacities, weights, and initial markings. The places are graphically represented by a circle and transitions by a rectangle. Oriented edges point either from a place to a transition or from a transition to a place. Place capacity indicates the maximum number of tokens that may be present at one time. The complexity calculated by the Petri nets is expressed by entropy and represents the uncertainty of the system. The greater the entropy value, the more the model is complex.

Generalized P/T Petri net is a 5-tuple, $PN = (P, T, F, W, M_0)$ where:

- $P = \{p_1, p_2, p_{3,}, \ldots, p_m\}$ – a finite set of places,
- $T = \{t_1, t_2, t_3, \ldots, t_n\}$ – a finite set of transitions,
- $P \cap T = \emptyset$ – places and transitions are mutually disjoint sets,
- $F \subseteq (P \times T) \cup (T \times P)$ – a set of edges (arcs), defined as a subset of the set of all possible connections,
- $W: F \rightarrow N_1$ – a weight function, defines the multiplicity of edges (arcs),
- $M_0: P \rightarrow N_0$ – an initial marking

Let $\boldsymbol{PN}$ be a Petri net and $\boldsymbol{A}$ its transition matrix, vector $\mathbf{u}$:$\boldsymbol{uA} = \boldsymbol{u}$ represents the stationary probabilities of all markings in $\boldsymbol{PN}$. Entropy of $\boldsymbol{PN}$ is then defined as:

$$H(PN) = -\sum_{i=1}^{|R(M_0)|} u_i \log_2 u_i \qquad (1)$$

where $|R(M_0)|$ is the number of all reachable markings for PN.

More details on the quantification of entropy in Petri nets can be found in [2, 11].

IV. CASE STUDY - GRANTING A LOAN PROCESS

Large banks have realized that a new, modern infrastructure information system needs to be adopted in order to be competitive and efficient. Therefore, the first step in this direction was the adoption of the Workflow Management System (WfMS) to support its business processes. Given that the bank provides a number of services to its customers, the adoption of the WfMS has enabled the logic of the banking processes to be captured in the scheme. As a result, part of the services provided are stored and implemented through a workflow management system. One of the services offered by the bank is the process of providing a loan. The process of granting the loan to the client consists of 18 nodes representing activities marked A to R and twenty-four transitions. Four XOR operators and one AND operator are used.

The first activity is the client's entry into the bank's internet application. In order for the client to enter the application, he/she must fill in the password and enter the certificate. Then the client chooses to apply for a loan. The Bank offers three types of loans. Housing loan, education loan or car loan. The client may request only one loan within the process. The bank accepts the client's request and decides whether to approve or reject it. After the bank decides, the client is informed by e-mail, and then the credit application with the resulting decision is stored in the bank's database and the credit application process is completed.

The list of process activities is as follows:

- A - access to the bank's internet application;
- B – inserting passwords;
- C - inserting a certificate;
- D - selection of service;
- E - filling in the loan application and selecting the type of loan;
- F - housing loan;
- G - education loan;
- H - car loan;
- I - housing loan approval;
- J - rejection of housing loan;
- K - approval of credit for education;
- L - rejection of credit for education;
- M - approval of car credit;
- N - rejection of car loan;
- O - informing the client of the decision on the loan for housing;
- P - informing the client of the decision on the loan for education;
- Q - informing the client of the decision on the car loan;
- R - save the request to the bank database and end the process.

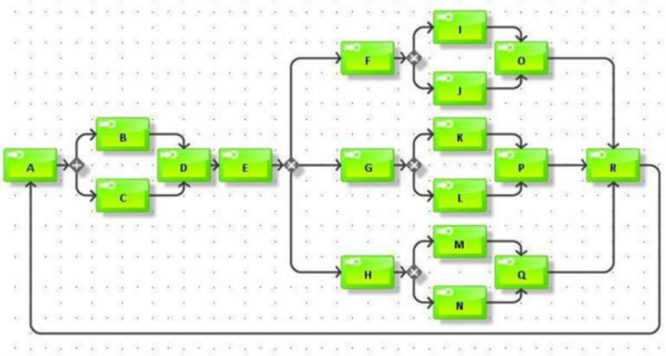

Figure 1. Process of obtaining loan - a case study.

Consequently, the complexity is computed for the process, firstly using a simple McCabe's MCC metric that ignores the used operators, then the Cardoso's CFC metric, which takes operators into account and ultimately entropy through the Petri nets.

MCC is computed using the formula e - n + 2. Where e is the number of edges and n is the number of nodes. The complexity calculated by this metric is equal to 8, which, according to the established limits, points to a simple process without great risk.

For each AND operator, the CFC complexity is equal to one. For the XOR operator, the CFC complexity of activity x is determined by the number of activities that follow from this activity, in other words by the number of outputs from activity x. For the presented process, individual CFC calculations are as follows:

- CFC_{AND} for A = 1;
- CFC_{XOR} for E = 3;
- CFC_{XOR} for F = 2;
- CFC_{XOR} for G = 2;
- CFC_{XOR} for H = 2.

By adding these individual complexities, an absolute CFC is obtain (equal to 10); the relative CFC is equal to 2.

The value of the process entropy modelled in the Petri nets is equal to 3.26.

Figure 2 and Figure 3 illustrate simplified versions of the original process.

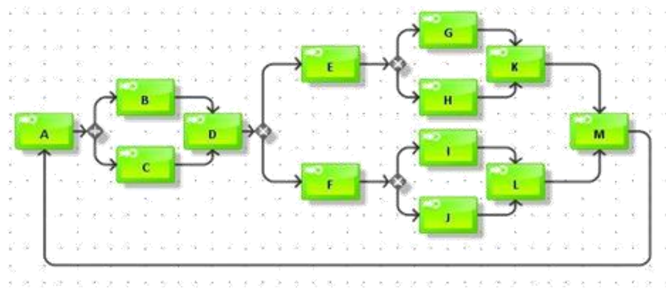

Figure 2. Simplified process A.

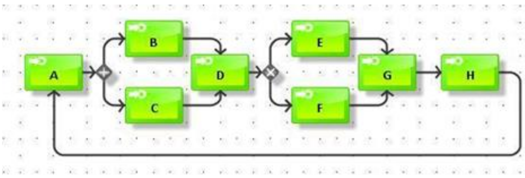

Figure 3. Simplified process B.

Figure *4* represents a more complicated version of the original process.

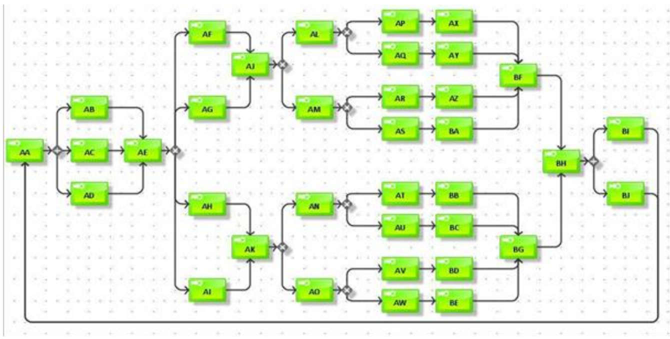

Figure 4. More complicated process.

The Figure 5 shows how the values of the individual complexities for different processes change. Interestingly, for the original process and its simplified variant A, the MCC value is lower than the CFCabs value. This is otherwise for simplified variant B, where the CFCabs value drops below the MCC, which is due to the fact that only two operators are used for this variant.

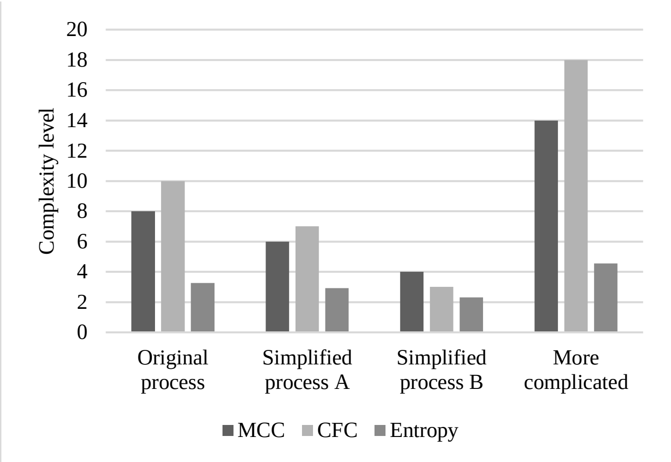

Figure 5. Comparison of different measures of complexity

The analysis shows that when modelling processes it is good to consider the number of activities and operators and try to minimize them to make the process as effective as possible.

A. *Correlation Analysis*

Based on the calculated values for the original process, its simplified variants A and B and for the more complicated process, a statistical correlation analysis of the individual complexity measures was performed in the SPSS Statistics program. Table 1 shows that there is a positive correlation between all metrics, the higher the value of one metric, the higher the value of the second metric. In all cases, this is a significant dependence, with the largest one being between CFC and entropy.

TABLE I. CORRELATION ANALYSIS

		MCC	**CFC**	**Entropy**
MCC	Pearson Correlation	1	,996**	,997**
	Sig. (2-tailed)		,004	,003
	N	4	4	4
CFC	Pearson Correlation	,996**	1	,999**
	Sig. (2-tailed)	,004		,001
	N	4	4	4
Entropy	Pearson Correlation	,997**	,999**	1
	Sig. (2-tailed)	,003	,001	
	N	4	4	4

**. Correlation is significant at the 0.01 level (2-tailed).

B. *Process Load Analysis*

The analysis is carried out in Petri nets and examines the load of the original process and its simplified variants, namely how the complexity, in this case expressed by entropy, is changing, with the growing number of users who are applying for a loan at the same time. The increase in load of the processes is shown in Figure 6, which shows that with the increasing number of registered users the complexity of the processes is increasing but gradually the growth is slowing down.

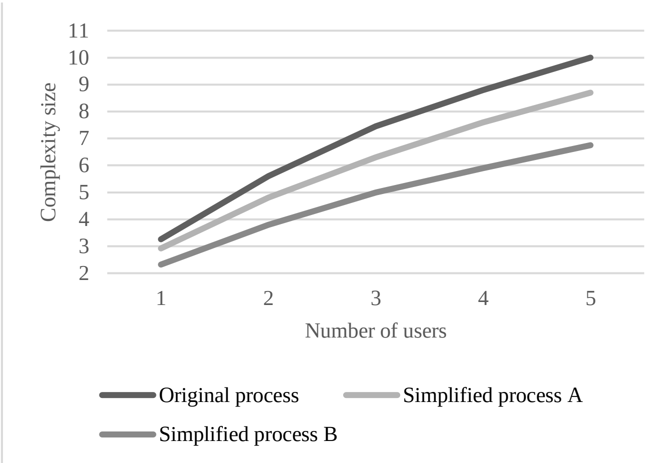

Figure 6. Comparison of different workload three processes.

V. DISCUSSION

From the empirical results, it is possible to see that the various complexity measures show a highly correlated dependence. It infers that the use of in this work specified measures is interchangeable. The main advantage of the complexity measure based on quantification of entropy in Petri nets is the possibility to simulate the increase/decrease of the load of individual states and monitor the response. The entropy measure of complexity in Petri nets therefore extends standard measures to the dynamic component. This makes it possible to achieve a more precise decision-making in general.

VI. Conclussion

Based on the comparative analysis, it can be stated that the individual complexity measures are comparable with statistical significance. In addition, it is possible to recommend the use of entropy in Petri nets as it extends the other measures with dynamic complexity analysis.

Acknowledgment

The paper was supported by the University of Pardubice, Faculty of Economics and Administration, Project SGS_2017_017.

References

[1] E. J. Weyuker, "Evaluating Software Complexity Measures," *IEEE Trans. Softw. Eng.*, vol. 14, pp. 1357-1365, 1988.

[2] M. Ibl, "Uncertainty Measure of Process Models using Entropy and Petri Nets," in *ICSOFT 2013 - Proceedings of the 8th International Joint Conference on Software Technologies*, 2013, pp. 542-547.

[3] S. Lloyd, "Measures of complexity: a nonexhaustive list," *IEEE Control Systems*, vol. 21, pp. 7-8, 2001.

[4] J. Cardoso, "Business Process Control-Flow Complexity: Metric, Evaluation, and Validation," *International Journal of Web Services Research (IJWSR)*, vol. 5, pp. 49-76, 2008.

[5] J. Cardoso, "Control-flow Complexity Measurement of Processes and Weyuker's Properties," *International Journal of Mathematical, Computational, Physical, Electrical and Computer Engineering*, vol. Vol: 1, pp. 366-371, 2007.

[6] T. J. McCabe, "A Complexity Measure," *IEEE Trans. Softw. Eng.*, vol. 2, pp. 308-320, 1976.

[7] V. Gruhn and R. Laue, "On Experiments for Measuring Cognitive Weights for Software Control Structures," in *6th IEEE International Conference on Cognitive Informatics*, 2007, pp. 116-119.

[8] K. B. Lassen and W. M. P. van der Aalst, "Complexity metrics for Workflow nets," *Information and Software Technology*, vol. 51, pp. 610-626, Mar 2009.

[9] I. Vanderfeesten, H. A. Reijers, J. Mendling, W. M. P. van der Aalst, and J. Cardoso, "On a Quest for Good Process Models: The Cross-Connectivity Metric," in *Advanced Information Systems Engineering: 20th International Conference, CAiSE 2008 Montpellier, France, June 16-20, 2008 Proceedings*, Z. Bellahsène and M. Léonard, Eds., ed Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 480-494.

[10] J. Mendling, M. Moser, G. Neumann, H. M. W. Verbeek, B. F. Dongen, and W. M. P. Aalst, "Faulty EPCs in the SAP Reference Model," in *Business Process Management*. vol. 4102, S. Dustdar, J. Fiadeiro, and A. Sheth, Eds., ed: Springer Berlin Heidelberg, 2006, pp. 451-457.

[11] M. Ibl and J. Čapek, "Measure of Uncertainty in Process Models Using Stochastic Petri Nets and Shannon Entropy," *Entropy*, vol. 18, p. 33, 2016.