Mendling, Jan and Neumann, Gustaf and van der Aalst, Wil M. (2007) Understanding the Occurrence of Errors in Process Models based on Metrics. In Meersman, R. and Tari, Z., Eds. *Proceedings 15th International Conference on Cooperative Information Systems (CoopIS 2007)* LNCS 4803, pages pp. 113-130, Vilamoura, Portugal.

Understanding the Occurrence of Errors in Process Models based on Metrics

Jan Mendling[1], Gustaf Neumann[2], and Wil van der Aalst[3]

[1] BPM Cluster, Faculty of Information Technology
Queensland University of Technology, Australia
j.mendling@qut.edu.au
[2] Institute of Information Systems and New Media
Vienna University of Economics and Business Administration, Austria
neumann@wu-wien.ac.at
[3] Department of Computer Science,
Eindhoven University of Technology, The Netherlands.
w.m.p.v.d.aalst@tue.nl

Abstract. Business process models play an important role for the management, design, and improvement of process organizations and process-aware information systems. Despite the extensive application of process modeling in practice, there are hardly empirical results available on quality aspects of process models. This paper aims to advance the understanding of this matter by analyzing the connection between formal errors (such as deadlocks) and a set of metrics that capture various structural and behavioral aspects of a process model. In particular, we discuss the theoretical connection between errors and metrics, and provide a comprehensive validation based on an extensive sample of EPC process models from practice. Furthermore, we investigate the capability of the metrics to predict errors in a second independent sample of models. The high explanatory power of the metrics has considerable consequences for the design of future modeling guidelines and modeling tools.

1 Introduction

Even though workflow and process modeling have been used extensively over the past 30 years, we know surprisingly little about the act of modeling and which factors contribute to a "good" process model in terms of error probability. This observation contrasts the large body of knowledge that is available for the formal analysis and verification of desirable properties, in particular for Petri nets. While conceptual work was conducted on guidelines and quality frameworks (e.g. [1–4]), there is clearly a need for an empirical research agenda to acquire new insights on quality (cf. [5]) and usage aspects (cf. [6]) of process modeling.

A recent study provides evidence that larger process models from practice tend to have more formal flaws (such as e.g. deadlocks) than smaller models [7, 8]. One obvious hypothesis related to this phenomenon would be that human modelers loose track of the interrelations of large and complex models due to their limited cognitive capabilities (cf. [9]), and then introduce errors that they would not insert in a small model. Yet, there are further factors beyond simple count metrics such as the degrees of sequentiality, concurrency, or structuredness that need to be considered [10]. Against this background, this paper provides the following three contributions. First, we introduce a tool-based approach for detecting errors and calculating metrics for Event-driven Process Chains (EPCs), a popular business process modeling language. Second, we utilize an extensive sample of EPC models from practice to analyze the statistical connection between errors and metrics. Third, we calculate a logistic regression model and validate its ability to predict errors in a second independent sample. All these contributions relate to the formal correctness of the process model as a design artifact. Validation aspects with respect to the content of a process model, human understandability issues, ease of use of the modeling language, and modeling pragmatics are also closely related to quality, but they are not considered here.

The remainder of the paper is structured as follows. In Section 2 we give a brief overview of EPCs, EPC soundness, and the kind of metrics we calculate. In Section 3 we introduce a sample of 2003 EPCs from practice that we use to investigate the connection between errors and metrics. Moreover, we provide disaggregated descriptive statistics. In Section 4 we determine the correlation between errors and metrics, and estimate a logistic regression function. This function is validated against a second independent sample of EPCs for its capability to predict errors. Section 5 discusses our findings in the light of related research before Section 6 concludes the paper.

2 Error Detection and Metrics Calculation for EPCs

The Event-driven Process Chain (EPC) is a business process modeling language for representing temporal and logical dependencies of activities in a business process (see [11]). EPCs offer *function type* elements to capture activities of a process and *event type* elements describing pre- and post-conditions of functions. *Process interface type* elements are used to refer to subsequent processes. Furthermore, there are three kinds of *connector types* including AND (symbol $\wedge$), OR (symbol $\vee$), and XOR (symbol $\times$) for the definition of complex routing rules. Connectors have either multiple incoming and one outgoing arc (join connectors) or one incoming and multiple outgoing arcs (split connectors). The informal (or intended) semantics of an EPC can be described as follows. The AND-split activates all subsequent branches in concurrency. The XOR-split represents a choice between one of alternative branches. The OR-split triggers one, two or up to all of multiple branches based on conditions. In both cases of the XOR- and OR-split, the activation conditions are given in events subsequent to the connector. The AND-join waits for all incoming branches to complete, then it propagates control to the subsequent EPC element. The XOR-join merges alternative branches. The OR-join synchronizes all active incoming branches. This feature is called non-locality since the state of all transitive predecessor nodes has to be considered. Regarding their routing elements, EPCs are quite similar to BPMN [12] and YAWL [13]. Recently, EPC semantics have been formalized, and there is tool support for the verification of EPC soundness (see [14]). In this paper, we use EPC soundness as a correctness criterion in order to find out whether the model has errors or not. In particular, an EPC is sound if and only if for a set of initial markings I and a set of final markings O the following three properties hold:

(i) For each start-arc there exists an initial marking $i \in I$ where the arc (and hence the corresponding start event) holds a positive token.
(ii) For every marking reachable from an initial state $i \in I$, there exists a firing sequence leading from this marking to a final marking $o \in O$.
(iii) The final markings $o \in O$ are the only markings reachable from a marking $i \in I$ such that there is no node that can fire.

We use two complementary tools to check whether an EPC is sound (has no errors) or unsound (has errors): firstly, *xoEPC* that implements a fast, but not complete reduction rule approach, secondly, a ProM plug-in [15] that calculates the reachability graph which is complete, but not very performative [16]. Both tools can be coupled using the EPML interchange format [17].

Beyond verification of EPC soundness, *xoEPC* also calculates a set of process model metrics. We briefly describe them in the following list including their hypothetical connection with errors. The formulas for calculating the different metrics are given and extensively discussed in [16, Chap.5].[4] Furthermore, this reference mentions related work for each of the metrics.

Size S_N refers to the number of nodes of the process model graph. An increase in S_N should imply an increase in error probability (+). Count metrics of different node types are written as e.g. S_C for connectors.

Diameter $diam$ gives the length of the longest path from a start node to an end node in the process model. It is presumably positively connected with error probability (+).

Density Δ relates the number of arcs to the maximum number of arcs between all nodes. We presume a positive connection (+).

Coefficient of Connectivity CNC gives the ratio of arcs to nodes (+).

Average Connector Degree $\overline{d_C}$ gives the number of nodes a connector is in average connected to (+).

Maximum Connector Degree $\widehat{d_C}$ captures the maximum degree over all connectors (+).

Separability Π gives the ratio of the number of cut-vertices to the number of nodes. An increase in Π should imply a decrease in error probability (–).

Sequentiality Ξ is the number of arcs between none-connector nodes divided by the overall number of arcs (–).

Structuredness Φ of the process graph is one minus the number of nodes in the EPC reduced with structured reduction rules divided by the number of nodes in the original EPC (–).

[4] This reference is also available online at http://wi.wu-wien.ac.at/home/mendling/publications/Mendling%20Doctoral%20thesis.pdf

Depth Λ captures how deep nodes are nested between splits and joins (+).

Connector Mismatch MM gives the sum of mismatches for each connector type. The mismatch is the absolute sum of all input arcs minus output arcs over all connectors of a connector type (+).

Connector Heterogeneity CH gives the type entropy of the connectors (+).

Control Flow Complexity CFC sums up all choice of a process based on the number of splits of each type and its number of outgoing arcs (+).

Cyclicity CYC relates nodes on cycles to all nodes (+).

Token Splits TS sums up all concurrent threads that can be activated by AND- and OR-splits in the process (+).

Figure 1 illustrates for an example EPC taken from [18] which nodes and arcs contribute to the more elaborate metrics. Since the different count metrics, in particular for size, can be easily read from the model, we focus on those that need to be calculated from the process graph, i.e. separability, sequentiality, structuredness, depth, cyclicity, and diameter.

The *separability ratio* Π depends on the identification of cut vertices (i.e. articulation points), i.e. those nodes whose deletion breaks up the graph in two or more disconnected components. Figure 1 displays articulation points with a letter A written next to the top left-hand side of the node. For example, if the function "record loan request" is deleted, the start event is no longer connected with the rest of the process model. There are eleven articulation points in total yielding a separability ratio of $11/(27-2) = 0.440$. Note that start and end events do not belong to the set of articulation points, since their deletion does not increase the number of separate components.

The *sequentiality ratio* Ξ is calculated by relating the number of sequence arcs, i.e. arcs that connect functions and events, to the total number of arcs. Figure 1 highlights sequence arcs with an s label. There are ten sequence arcs and 29 arcs altogether which results in a sequentiality ratio of $10/29 = 0.345$. The degree of *structuredness* Φ relates the size of a reduced process model to the size of the original one. Figure 1 shows those elements with a cross on the left-hand side that are eliminated by reduction of trivial constructs. Other structured reduction rules are not applicable. Since 15 elements are deleted by reduction, the structuredness ratio is $1-15/27 = 0.556$. The *in-depth and out-depth* is also indicated for each node in Figure 1. The depth of a node is then the minimum of in-depth and out-depth. Several nodes have a depth of 1, which is a maximum, and therefore also the depth of the overall process. The *cyclicity* is based on the relation between number of nodes on a cycle and nodes in total. Figure 1 shows nodes on a cycle with a letter C written to the left-hand side bottom. There are seven such nodes yielding a cyclicity ratio of $7/27 = 0.259$. Finally, Figure 1 connects those 14 nodes that are on the *diameter* with a bold line.

3 Empirical Distribution of Errors and Metrics

As input to our analysis we use a sample of EPC business process models that are available in the XML interchange format of ARIS Toolset of IDS Scheer

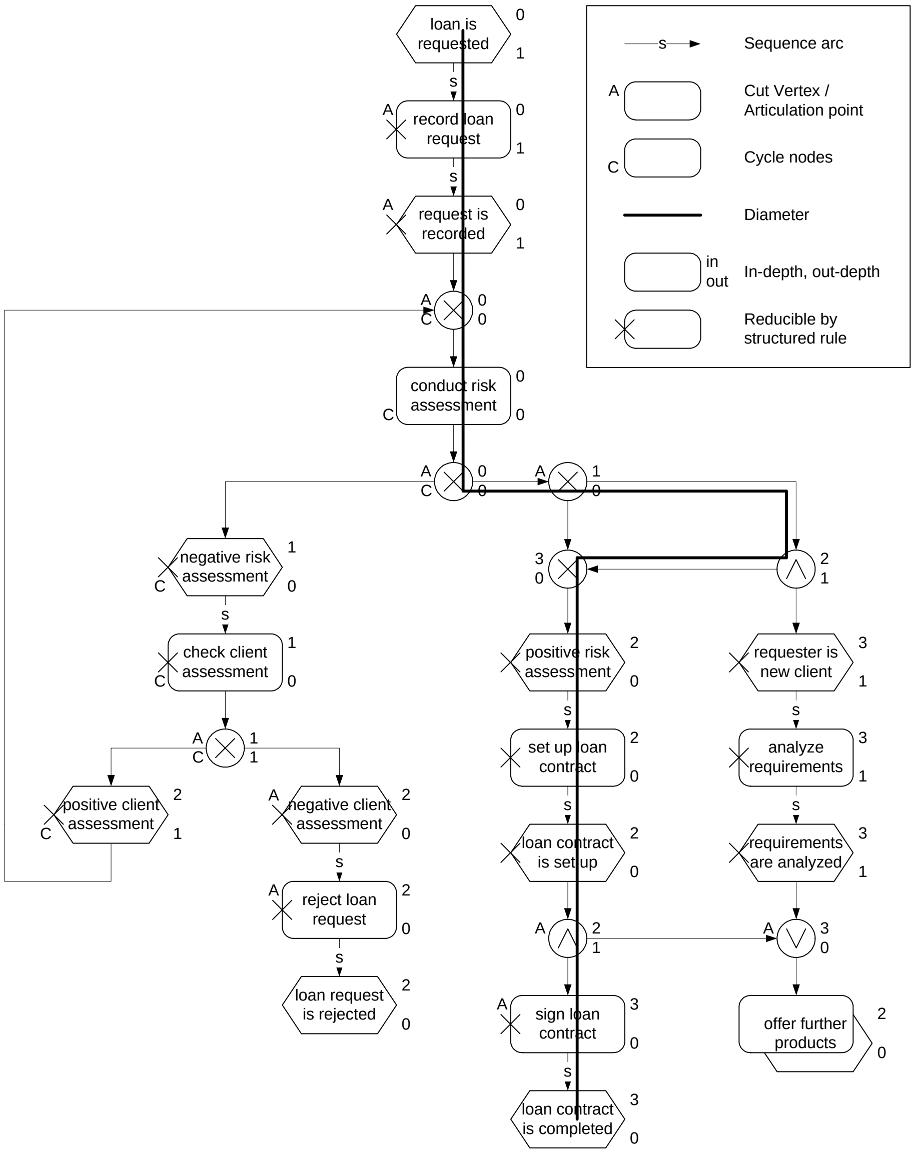

Fig. 1. EPC example with sequence arcs, articulation points, cycle nodes, diameter, depth, and reducible nodes

AG. The sample includes four collections of EPCs with a total of 2003 process models. All EPCs of the four groups were developed by practitioners.

1. *SAP Reference Model:* The first collection of EPCs is the SAP Reference Model. The development of the SAP reference model started in 1992 and first models were presented at CEBIT'93 [19, p.VII]. We use the SAP reference model in its version from 2000 that includes 604 non-trivial EPCs.
2. *Service Model:* The second collection of EPCs stems from a German process reengineering project in the service sector. The project was carried out in the late 1990s. The models of this project include 381 non-trivial EPCs.
3. *Finance Model:* The third model collection contains EPCs of a process documentation project in the Austrian financial industry. It includes 935 EPCs.
4. *Consulting Model:* The fourth collection covers in total 83 EPCs from three different consulting companies.

As a first step, the set of ARIS XML files is read and processed by *xoEPC* to generate information on errors and values for all the metrics. Furthermore, each EPC is then checked by the help of the reachability analysis plug-in for ProM. The results of this analysis are added to an analysis table. We use the software package for the statistical analysis of this table. In particular we present descriptive statistics disaggregated by group and error in Sections 3.1 and 3.2.

3.1 Descriptive Statistics Disaggregated by Group

In this section we characterize the overall EPC sample and its four sub-groups by the help of mean values μ and standard deviation σ for each metric. Several of the disaggregated mean values are quite close to each other, but in particular the Finance Model shows a striking differences: it has the highest mean in structuredness and sequentiality. Figures 2 and 3 illustrates the distribution of both these metrics as box plots disaggregated by group. In this type of diagram invented by *Tukey* [20] the median is depicted as a horizontal line in a box that represents the interval between lower and upper quartile, i.e. the EPCs ranked by the metric from 25% to 75%. The upper and lower wicks define a one and a half multiple of the respective quartile. Values outside these two intervals are drawn as individual points and are considered to be outliers. From this observation on structuredness and sequentiality we might conclude that the Finance Model contains the more structured EPCs and thus might have less error models.

There is some evidence for such a hypothesis when we look at the number of errors in each of the four groups. Table 1 gives a respective overview. It can be seen that there are 2003 EPCs in the overall sample and 215 of them have at least one error. Accordingly, there is an overall error ratio of 10.7%. 154 of the 215 errors were found by *xoEPC*. 156 EPCs could not be reduced completely and were analyzed with ProM. This analysis revealed that 115 of the unreduced EPCs still had errors. Please note that there are EPCs for which both *xoEPC* and ProM found errors. Therefore, the number of EPCs with errors is less than the sum of EPCs with *xoEPC* and ProM errors. The comparison of the

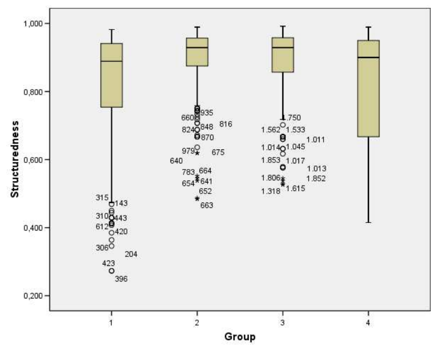

Fig. 2. Box plot for structuredness disaggregated by group (1=SAP, 2=Service, 3=Finance, and 4=Consulting)

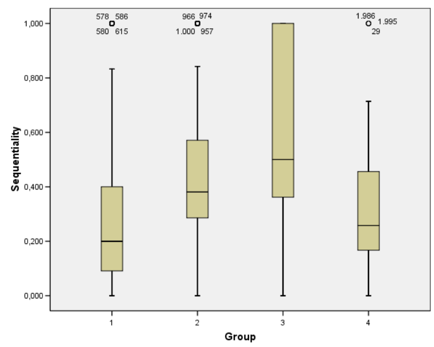

Fig. 3. Box plot for sequentiality disaggregated by group (1=SAP, 2=Service, 3=Finance, and 4=Consulting)

Table 1. Errors in the sample models

Parameter	Complete Sample	SAP Ref. Model	Services Model	Finance Model	Consulting Models
xoEPC errors	154	90	28	26	10
Unreduced EPCs	156	103	18	17	18
ProM error EPCs	115	75	16	7	17
EPCs with errors	215	126	37	31	21
EPCs in total	2003	604	381	935	83
Error ratio	10.7%	20.9%	9.7%	3.3%	25.3%

groups shows that the error ratio is quite different. In the previous paragraph we hypothesized that the finance model group might have less errors since its models are more structured. This suggests that metrics could be able to explain the low error ratio of only 3.3 %. We search further evidence in the next section.

3.2 Descriptive Statistics Disaggregated by hasErrors

In this section we discuss the distribution of the different metrics disaggregated by the variable *hasErrors*. Table 2 shows that there are quite large differences in the mean values of the sub-samples with and without errors. It is interesting to note that the error mean μ_e is higher than the non-error mean μ_n for most metrics where we assumed a positive connection with error probability in Section 2 and smaller for those metrics with a presumably negative connection. The only case where this does not hold is the density metric; it seems that it more accurately works as a counter-indicator for size than as an indicator for the density of connections in the model. The two columns on the right hand side of Table 2 might provide the basis for proposing potential error thresholds. The first of these columns gives a double σ_n deviation upwards from the non-error mean μ_n. Given a normal distribution only 2.5% of the population can be ex-

Table 2. Mean and Standard Deviation of the sample models disaggregated by error

Parameter	Complete Sample		Non-Error EPCs		Error EPCs		2 σ dev. up	2 σ dev. down
	μ	σ	μ_n	σ_n	μ_e	σ_e	$\mu_n + 2\sigma_n$	$\mu_n - 2\sigma_n$
S_N	20.71	16.84	18.04	13.48	42.97	24.08	44.99 $\approx \mu_e$	
S_E	10.47	8.66	9.06	6.69	22.17	13.19	22.45 $\approx \mu_e$	
S_F	5.98	4.94	5.67	4.65	8.53	6.33	14.97	
S_C	4.27	5.01	3.30	3.47	12.26	7.89	10.24 $< \mu_e$	
S_A	21.11	18.87	18.14	15.20	45.79	26.78	48.54 $\approx \mu_e$	
$diam$	11.45	8.21	10.63	7.71	18.25	9.01	26.06	
Δ	0.09	0.07	0.09	0.07	0.03	0.02	0.23	
CNC	0.96	0.13	0.95	0.13	1.05	0.08	1.21	
$\overline{d_C}$	3.56	2.40	2.80	1.66	3.57	0.68	6.11	
$\widehat{d_C}$	2.88	1.60	3.31	2.28	5.64	2.41	7.87	
Sep. Π	0.56	0.27	0.59	0.27	0.35	0.13		0.06
Seq. Ξ	0.46	0.31	0.49	0.30	0.18	0.14		-0.12
Strct. Φ	0.88	0.11	0.90	0.09	0.70	0.16		0.72 $> \mu_e$
Depth Λ	0.70	0.74	0.61	0.69	1.45	0.73	1.98	
MM	3.31	4.55	2.54	3.45	9.71	6.92	9.44 $< \mu_e$	
CH	0.28	0.35	0.22	0.32	0.75	0.19	0.85	
CFC	382.62	8849.48	202.19	6306.23	1883.17	19950.26	12814.64	
CYC	0.01	0.08	0.01	0.06	0.07	0.17	0.12	
TS	1.82	3.53	1.28	2.46	6.26	6.62	6.20 $< \mu_e$	

pected to have a metric value greater than this. The comparison of this value to the mean μ_e of the error EPCs gives an idea how good the two subparts of the sample can be separated by the metric. In several cases the mean μ_e is outside the double σ_n interval around μ_n. The box plots in Figures 4 and 5 illustrate the different distributions. It can be seen that correct EPCs tend to have much higher structuredness values and lower connector heterogeneity values. The next section investigates this observation with inferential statistics.

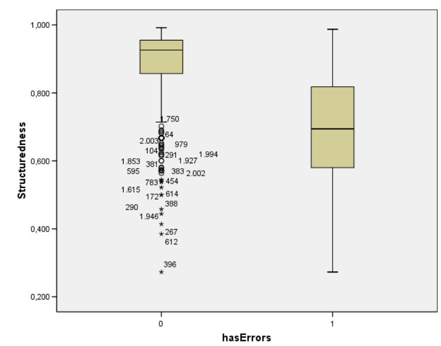

Fig. 4. Box plot for structuredness disaggregated by error

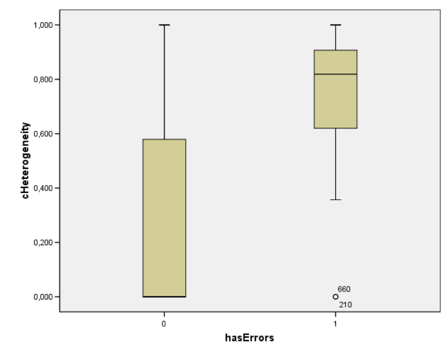

Fig. 5. Box plot for connector heterogeneity disaggregated by error

4 Inferential Statistics

4.1 Correlation Analysis

This section approaches the connection between error probability and metrics with a correlation analysis. We use the Spearman rank correlation coefficient for ordinal data. As a confirmation of the previous observation all variables have the expected direction of influence besides the density metric. Table 3 presents the Spearman correlation between *hasErrors* and the metrics ordered by strength of correlation. It can be seen that several correlations are quite considerable with absolute values between 0.30 and 0.50. The significance of all correlations is good with 99% confidence.

Table 3. Spearman correlation between hasError and metrics ordered by absolute correlation

	hasError		hasError
cHeterogeneity	0.46	Sequentiality	-0.35
C	0.43	Depth	0.34
MM	0.42	MaxCDegree	0.33
CFC	0.39	CYC	0.30
A	0.38	diameter	0.30
tokenSplit	0.38	Separability	-0.29
N	0.38	CNC	0.28
E	0.38	AvCDegree	0.23
Density	-0.37	F	0.19
Structuredness	-0.36		

The ability of a metric to separate error from non-error models by ranking is illustrated in Figures 6 and 7. For Figure 6 all models are ranked according to their size. A point (x, y) in the graph relates a size x to the relative frequency of error models in a subset of models that have at least size x, i.e. $y = |\{\frac{|errorEPCs|}{|allEPCs|} \mid S_N(EPC) > x\}|$. It can be seen that the relative frequency of error EPCs increases by increasing the minimum number of nodes. In particular, the relative frequency of error EPCs is higher than 50% for all EPCs of at least 48 nodes. In Figure 6 all models are ranked according to their structuredness and (x, y) relates the structuredness x to the subset of models that have at most structuredness x. Here, the graph decreases and drops below 50% at a structuredness value of 0.80. Similar observations can be made for some of the other metrics, too. The values could be used as candidate thresholds. Altogether the relative frequency of error models above 50% is reached if

number of nodes $S_N > 48$	number of arcs $S_A > 62$
number of connectors $S_C > 8$	token splits $TS > 7$
number of events $S_E > 22$	connector mismatch $MM > 9$
number of functions $S_F > 40$	structuredness $\Phi < 0.8$

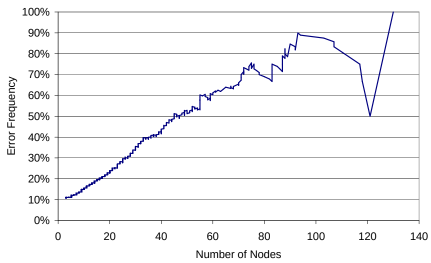

Fig. 6. Error frequency to ordered number of nodes

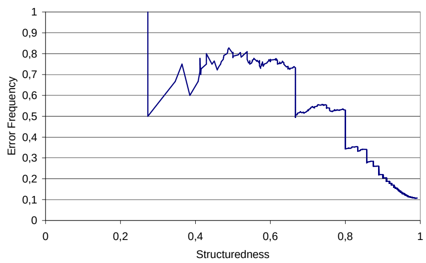

Fig. 7. Error frequency to ordered structuredness

4.2 Logistic Regression Estimation

This section provides an introduction to logistic regression analysis and presents the result of its application for estimating the prediction model for error probability based on metrics. Logistic regression is a statistical model to estimate the probability of binary choices. It is perfectly suited to deal with dependent variables such as *hasErrors* with its range *error* and *no error*. The idea of binary choice models is to describe the probability of a binary event by its odds, i.e., the ratio of event probability divided by non-event probability. In the *logistic regression* (or *logit*) model the odds are defined as $logit(p_i) = ln(\frac{p_i}{1-p_i}) = \beta_0 + \beta_1 x_{1,i} + \ldots + \beta_k x_{k,i}$ for k input variables and i observations, i.e. i EPCs in our context. From this follows that

$$p_i = \frac{e^{\beta_0 + \beta_1 x_{1,i} + \ldots + \beta_k x_{k,i}}}{1 + e^{\beta_0 + \beta_1 x_{1,i} + \ldots + \beta_k x_{k,i}}}$$

The relationship between input and dependent variables is represented by an S-shaped curve of the logistic function that converges to 0 for $-\infty$ and to 1 for ∞. The cut value of 0.5 defines whether event or non-event is predicted. $Exp(B)$ gives the multiplicative change of the odds if the input variable is increased by one unit, i.e. $Exp(B) > 1$ increases and $Exp(B) < 1$ decreases error probability. The actual value $Exp(B)$ cannot be interpreted in isolation since its impact depends upon the position on the non-linear curve [21, p.791]. The significance of a logistic regression model is assessed by the help of two statistics. First, the *Hosmer & Lemeshow Test* should be greater than 5% to indicate a good fit based on the difference between observed and predicted frequencies. Second, *Nagelkerke's* R^2 ranging from 0 to 1 serves as a coefficient of determination indicating which fraction of the variability is explained. Furthermore, each estimated coefficient of the logit model is tested using the *Wald statistic* for being significantly different from zero. The significance should be less than 5%. We calculate the logistic regression model based on a stepwise introduction of those variables that provide the greatest increase in likelihood. For more on logistic regression see [22].

Before calculating a multivariate logistic regression model for error probability we carry out two preparatory analyses. First, we check collinearity, then we determine which variables are included in the regression model. Furthermore, we excluded 29 EPCs from the analysis that were not syntactically correct. *Collinearity* describes the phenomenon that at least one of the independent variables can be represented as a linear combination of other variables. The absence of collinearity is not a hard criterion for the applicability of logistic regression, but it is desirable. Since the tolerance test indicated collinearity problems, we had to dropped all count metrics apart from S_N since they were highly correlated. This resulted in a reduced variable set with no collinearity problems. Furthermore, we calculated *univariate models* with and without a constant in order to check whether all inputs were significantly different from zero. As a conclusion from this analysis we drop the constant and the control flow complexity CFC for the multivariate analysis. First, the constant is not significantly different from zero (Wald statistic of 0.872 and 0.117) in the separability and the sequentiality model which suggests that it is not necessary. Second, the CFC metric is not significantly different from zero (Wald statistic of 0.531 and 0.382) in both models with and without constant. All other metrics stay in the set of input variables from the multivariate logistic regression model.

The final model was calculated in nine steps and it includes seven variables. It is interesting to note that again the hypothetical impact direction of the included metrics is confirmed. All variables have an excellent *Wald statistic* value of better than 0.001 indicating that they are significantly different from zero. Furthermore, the *Hosmer & Lemeshow test* is greater than 0.05 which is also a good value. Finally, the *Nagelkerke* R^2 has an excellent value of 0.901 indicating a high degree of explanation. Based on the regression results we can derive a classification function $p(EPC)$ for EPCs. It predicts that the EPC has errors if the result is greater than 0.5. Otherwise it predicts that there are no errors in the EPC. It is calculated by the help of the metrics coefficient of connectivity CNC, connector mismatch MM, cyclicity CYC, separability Π, structuredness Φ, connector heterogeneity CH, and the diameter. It is

$$p(EPC) = \frac{e^{logit(EPC)}}{1 + e^{logit(EPC)}}$$

with

$$\begin{aligned} logit(EPC) = & +4.008\ CNC \\ & +0.094\ MM \\ & +3.409\ CYC \\ & -2.338\ \Pi \\ & -9.957\ \Phi \\ & +3.003\ CH \\ & +0.064\ diameter \end{aligned}$$

It is easy to calculate an error prediction for an EPC based on this function. For the sample this function yields the following classification:

- 1724 EPCs are correctly predicted to have no errors,
- 155 EPCs are correctly predicted to have errors,
- 58 EPCs are predicted to have no errors, but actually had, and
- 37 EPCs are predicted to have errors, but actually had none.

Altogether 1879 EPCs have the correct prediction. The overall percentage is 95.2%, that is 6% better than the naive model that always predicts no error (89.2%). Furthermore, there are 213 EPCs with errors in the reduced sample. 155 of them are correctly predicted, i.e. 72.7%. Finally, the prediction function gives a clue about the relative importance of the different metrics. Structuredness appears to be the most important parameter since its absolute value is three times as high as the second. Then, the coefficient of connectivity, cyclicity, separability, and connector heterogeneity seem to be of comparable importance. Finally, connector mismatch and the diameter might be of minor importance.

In the following section we analyze how good the regression function is able to forecast errors in a sample of EPCs that was not included in the estimation.

4.3 Logistic Regression Validation

In this section we utilize the estimated function to predict errors in EPCs from a holdout sample. This step is of paramount importance for establishing the criterion validity of the measurements, i.e. their pragmatic value (cf. e.g. [23]). For testing the performance of the prediction function we gathered a sample from popular German EPC business process modeling textbooks. The sample includes 112 models from the following books in alphabetical order:

- *Becker and Schütte* [24]. This book discusses information systems in the retail sector with a special focus on conceptual modeling. In particular it covers 65 EPC models that we include in the holdout sample.
- *Scheer* [25]. This textbook is an introduction to the ARIS framework and uses reference models for production companies to illustrate it. From this book we includes 27 EPC reference models in the holdout sample.
- *Seidlmeier* [26]. This book is an introduction to the ARIS framework. It includes 10 EPCs that we include in the holdout sample.
- *Staud* [27]. This book focuses on business process modeling and in particular EPCs. We included 13 EPCs from this book in the holdout sample.

All EPCs of the holdout sample were checked for errors first with *xoEPC* and afterwards with the ProM plug-in. Altogether there are 25 of the 113 models that have errors, i.e. 21.43%. Based on the metrics generated by *xoEPC* we can easily apply the prediction function. The result of this calculation is summarized in the classification table in Figure 8. It can be seen that 102 of the 113 EPCs are classified correctly, i.e. 86 models without errors are predicted to have none and 16 with errors are predicted to have at least one. Altogether 90.27% of the 113 EPCs were predicted correctly. Please note that there is a difference in the interpretation of this classification result and the one in Section 4.2. During the estimation of the logistic regression the sample is known and used to tune the

Classification Table

Observed		Predicted hasErrors 0	1	Percentage Correct
hasErrors	0	86	2	97,73%
	1	9	16	64,00%
Overall Percentage				90,27%

The cut value is ,500
113 cases included

Fig. 8. Classification table for EPCs from the holdout sample

coefficients. Here, we use this function to classify an independent sample. Based on the De Moivre-Laplace theorem, we are able to calculate a confidence interval for the accuracy of the prediction function. With a confidence value of 95% it yields an accuracy interval from 81.15% to 96.77%, i.e. the prediction can be expected to be correct in at least 81.15% of the cases with a 95% confidence. This result strongly supports the validity of the function for predicting error probability.

4.4 Implications of the Findings

In this section we have conducted several statistical analyses related to the hypotheses on a connection between metrics and error probability. The results strongly confirm the hypotheses since the mean difference between error and non-error models, the correlation coefficients, and the regression coefficients confirm the hypothetical impact direction of all metrics except the density metric (see Table 4). This metric appears to be more closely related to the inverse of size than the relative number of arcs of an EPC.

These results have strong implications for the quality of business process modeling:

1. The connection of the metrics with error probability provides a theoretical and empirical basis for defining process modeling principles and guidelines. The analysis reveals that in particular structured models are less error prone.
2. The established connection builds a foundation for a measurement-based management approach for the process of business process modeling. Different design alternatives can be discussed more objectively on the metric values.
3. The design of future business process modeling tools can benefit from these findings by providing online feedback to the modeler when a certain metric passes an error threshold.
4. It has also some implications on the level of the process modeling language. Considering that the connector heterogeneity has an impact on error probability it might be a good idea to restrict modeling to the two connector types AND and XOR, and use OR-connectors only in structured blocks. Furthermore, there was a strong correlation between the number of start and end events with error probability. This fact suggests to restrict the use of multiple

Table 4. Hypothetical and empirical connection between metrics and errors

	Hypothetical connection	$\mu_e - \mu_n$	Correlation	Regression coefficient	Direction
S_N	+	24.93	0.38		confirmed
S_E	+	13.11	0.38		confirmed
S_F	+	2.86	0.19		confirmed
S_C	+	8.96	0.43		confirmed
S_A	+	27.64	0.38		confirmed
$diam$	+	7.62	0.30	0.064	confirmed
Δ	+	-0.06	-0.37		not confirmed
CNC	+	0.11	0.28	4.008	confirmed
$\overline{d_C}$	+	0.76	0.23		confirmed
$\widehat{d_C}$	+	2.33	0.33		confirmed
Sep. Π	-	-0.24	-0.29	-2.338	confirmed
Seq. Ξ	-	-0.31	-0.35		confirmed
Strct. Φ	-	-0.20	-0.36	-9.957	confirmed
Depth Λ	+	0.85	0.34		confirmed
MM	+	7.18	0.42	0.094	confirmed
CH	+	0.54	0.46	3.003	confirmed
CFC	+	1680.99	0.39		confirmed
CYC	+	0.06	0.30	3.409	confirmed
TS	+	4.97	0.38		confirmed

starts and ends. Modelers seem to loose track of the allowed combinations of these elements quite fast. In the reduced set of EPCs there are several EPCs for which no combination of start events guarantees a proper execution.

5. The results have implications for the teaching of business process modeling. On the one hand, the large number of errors suggests that practitioners frequently have problems to understand the behavioral implications of their design. On the other hand, the metrics are a good starting point to teach patterns that are unlikely to result in errors.

5 Related Work

There are basically two main streams of research related to our work in the conceptual modeling area: top-down quality frameworks and bottom-up metrics that relate to quality aspects. For related work on Petri net verification refer to [28] and on EPCs to [16].

One prominent *top-down quality framework* is the SEQUAL framework [1, 4]. It builds on semiotic theory and defines several quality aspects based on relationships between a model, a body of knowledge, a domain, a modeling language, and the activities of learning, taking action, and modeling. Its usefulness was confirmed in an experiment [29]. The Guidelines of Modeling (GoM) [2] define an alternative quality framework that is inspired by general accounting principles. The guidelines include the six principles of correctness, clarity, relevance, comparability, economic efficiency, and systematic design. This framework was operationalized for EPCs and also tested in experiments [2]. Furthermore, there are authors (e.g. [5]) advocating a specification of a quality framework for conceptual modeling in compliance with the ISO 9126 standard [30] for software quality. A respective adaptation to business process modeling is reported in [31]. Our research complements these approaches regarding semantical correctness. While the frameworks tend to be rather abstract, we find strong support for operational recommendations like using structured building blocks and limiting the number of nodes in a single process model.

Much work has been done related to *bottom-up metrics that relate to quality aspects* of process models, stemming from different research and partially isolated from each other (see [32–40, 10] or for an overview [16]). Several of these contributions are theoretic without empirical validation. Most authors doing experiments focus on the relationship between metrics and quality aspects: *Canfora et al.* study the connection between mainly count metrics for e.g. activities or routing elements and maintainability of software process models [38]; *Cardoso* validates the correlation between control flow complexity and perceived complexity [41]; and *Mendling et al.* use metrics to predict control flow errors such as deadlocks in process models [8, 10]. The results of this research confirm the negative connection between size and quality aspects. Beyond that, it extends this stream of research with a validation of an error prediction function that was derived by the help of an extensive sample of process models from practice.

Finally, there are some further surveys that investigate the maturity [42], usability [43], and understandability of business process modeling languages [44]. They also relate to quality aspects of process models, but not directly to the connection of errors and metrics.

6 Conclusions and Future Work

With this paper, we addressed the shortage of empirical insight into business process modeling and its quality parameters in practice. In particular, we used a collection of 2003 EPC business process models from practice, and determined for each of the models whether they have errors or not. Furthermore, we calculated an extensive set of metrics for each model. Based on this data, we were able to show that several metrics have a strong statistical connection with the occurrence of errors, and that most of the metrics increase or decrease error probability as expected. Using a logistic regression model, we could even derive a prediction function that accurately classifies models as having errors or not based on metrics.

These findings clearly demonstrate that errors do not occur by chance in business process models, and that certain characteristics like structuredness are desirable for avoiding errors. In future research we aim to conduct further experiments to test the connection between the metrics and other quality aspects of modeling like understandability and maintainability. As a result from these experiments, we expect to define new business process modeling guidelines which are metrics-based, which can be easily translated into operations, and which lead to high quality business process models.

References

1. Lindland, O.I., Sindre, G., Sølvberg, A.: Understanding quality in conceptual modeling. IEEE Software **11** (1994) 42–49
2. Becker, J., Rosemann, M., Uthmann, C.: Guidelines of Business Process Modeling. In van der Aalst, W., Desel, J., Oberweis, A., eds.: Business Process Management. Models, Techniques, and Empirical Studies. Springer, Berlin et al. (2000) 30–49
3. Hoppenbrouwers, S.S., Proper, H.E., van der Weide, T.: A Fundamental View on the Process of Conceptual Modeling. In: Conceptual Modeling – ER 2005 – 24 International Conference on Conceptual Modeling. Volume 3716 of Lecture Notes in Computer Science. (2005) 128–143
4. Krogstie, J., Sindre, G., Jørgensen, H.: Process models representing knowledge for action: a revised quality framework. European Journal of Information Systems **15** (2006) 91–102
5. Moody, D.: Theoretical and practical issues in evaluating the quality of conceptual models: current state and future directions. Data & Knowledge Engineering **55** (2005) 243–276
6. Davies, I., Green, P., Rosemann, M., Indulska, M., Gallo, S.: How do practitioners use conceptual modeling in practice? Data & Knowledge Engineering **58** (2006) 358–380
7. Mendling, J., Moser, M., Neumann, G., Verbeek, H., Dongen, B., van der Aalst, W.: Faulty EPCs in the SAP Reference Model. In S. Dustdar, J.F., Sheth, A., eds.: Proceedings of BPM 2006. Volume 4102 of Lecture Notes in Computer Science., Vienna, Austria, Springer-Verlag (2006) 451457
8. Mendling, J., Moser, M., Neumann, G., Verbeek, H., Dongen, B., van der Aalst, W.: Detection and Prediction of Errors in EPCs of the SAP Reference Model. *Data & Knowledge Engineering*, 2007. accepted for publication.
9. Simon, H.: Sciences of the Artificial. 3rd edn. The MIT Press (1996)
10. Mendling, J., Neumann, G.: Error metrics for business process models. Technical Report JM-2006-12-03, Vienna Univ. of Econ. and Business Administration (2006)
11. Keller, G., Nüttgens, M., Scheer, A.W.: Semantische Prozessmodellierung auf der Grundlage "Ereignisgesteuerter Prozessketten (EPK)". Heft 89, Institut für Wirtschaftsinformatik, Saarbrücken, Germany (1992)
12. OMG, ed.: Business Process Modeling Notation (BPMN) Specification. Final Adopted Specification, dtc/06-02-01, Object Management Group (2006)
13. van der Aalst, W., ter Hofstede, A.: YAWL: Yet Another Workflow Language. Information Systems **30** (2005) 245–275
14. Mendling, J., van der Aalst, W.: Formalization and Verification of EPCs with OR-Joins Based on State and Context. In: Proceedings of the 19th Conference on Advanced Information Systems Engineering (CAiSE 2007). Proceedings. (2007)
15. van Dongen, B., Medeiros, A., Verbeek, H., Weijters, A., van der Aalst, W.: The ProM framework: A New Era in Process Mining Tool Support. In Ciardo, G., Darondeau, P., eds.: Application and Theory of Petri Nets 2005. Volume 3536 of LNCS. Springer-Verlag, Berlin (2005) 444–454
16. Mendling, J.: Detection and Prediction of Errors in EPC Business Process Models. PhD thesis, Vienna University of Economics and Business Administration (2007)

17. Mendling, J., Nüttgens, M.: EPC Markup Language (EPML) - An XML-Based Interchange Format for Event-Driven Process Chains (EPC). Information Systems and e-Business Management **4** (2006) 245 – 263
18. M. Nüttgens and F.J. Rump. Syntax und Semantik Ereignisgesteuerter Prozessketten (EPK). In J. Desel and M. Weske, editor, *Proceedings of Promise 2002, Potsdam, Germany*, volume 21 of *Lecture Notes in Informatics*, pages 64–77, 2002.
19. Keller, G., Teufel, T.: SAP(R) R/3 Process Oriented Implementation: Iterative Process Prototyping. Addison-Wesley (1998)
20. Tukey, J.: Exploratory Data Analysis. Addison-Wesley (1977)
21. Judge, G., Hill, R., Griffiths, W., Lütkepohl, H., Lee, T.C.: Introduction to the theory and practice of econometrics. 2nd edn. John Wiley & Sons (1988)
22. Hosmer, D., Lemeshow, S.: Applied Logistic Regression. Wiley & Sons (2000)
23. Marczyk, G., DeMatteo, D., Festinger, D.: Essentials of Research Design and Methodology. Wiley & Sons, Inc. (2005)
24. Becker, J., Schütte, R.: Handelsinformationssysteme. Moderne Industrie (2004)
25. Scheer, A.-W.: Wirtschaftsinformatik: Referenzmodelle für industrielle Geschäftsprozesse. Springer-Verlag (1998)
26. Seidlmeier, H.: Prozessmodellierung mit ARIS. Vieweg Verlag (2002)
27. Staud, J.: Geschäftsprozessanalyse: Ereignisgesteuerte Prozessketten und Objektorientierte Geschäftsprozessmodellierung. Springer-Verlag (2006)
28. Reisig, W., Rozenberg, G., eds.: Lectures on Petri Nets I: Basic Models. Volume 1491 of LNCS., Springer-Verlag, Berlin (1998)
29. Moody, D., Sindre, G., Brasethvik, T., Sølvberg, A.: Evaluating the quality of process models: Empirical testing of a quality framework. In Spaccapietra, S., March, S.T., Kambayashi, Y., eds.: Conceptual Modeling - ER 2002, Proceedings. Volume 2503 of Lecture Notes in Computer Science., Springer (2002) 380–396
30. ISO: Information technology - software product evaluation - quality characteristics and guide lines for their use. Iso/iec is 9126 (1991)
31. Gücegliogl u, A.S., Demirörs, O.: Using software quality characteristics to measure business process quality. In van der Aalst, W., Benatallah, B., Casati, F., Curbera, F., eds.: Business Process Management, 3rd International Conference, BPM 2005, Nancy, France, September 5-8, 2005, Proceedings. Volume 3649 of Lecture Notes in Computer Science (LNCS)., Springer Verlag (2005) 374–379
32. Lee, G., Yoon, J.M.: An empirical study on the complexity metrics of petri nets. Microelectronics and Reliability **32** (1992) 323–329
33. Nissen, M.: Redesigning reengineering through measurement-driven inference. MIS Quarterly **22** (1998) 509–534
34. Morasca, S.: Measuring attributes of concurrent software specifications in petri nets. In: METRICS '99: Proceedings of the 6th International Symposium on Software Metrics, Washington, DC, USA, IEEE Computer Society (1999) 100–110
35. Reijers, H., Vanderfeesten, I.: Cohesion and coupling metrics for workflow process design. In Desel, J., Pernici, B., Weske, M., eds.: Business Process Management: Second International Conference, BPM 2004, Potsdam, Germany, June 17-18, 2004. Proceedings. Volume 3080 of Lecture Notes in Computer Science., Springer (2004) 290–305
36. Cardoso, J.: Evaluating Workflows and Web Process Complexity. In: Workflow Handbook 2005. Future Strategies Inc. (2005) 284–290
37. Balasubramanian, S., Gupta, M.: Structural metrics for goal based business process design and evaluation. Business Process Management Journal **11** (2005) 680–694

38. Canfora, G., García, F., Piattini, M., Ruiz, F., Visaggio, C.: A family of experiments to validate metrics for software process models. Journal of Systems and Software **77** (2005) 113–129
39. Aguilar, E.R., Ruiz, F., García, F., Piattini, M.: Towards a Suite of Metrics for Business Process Models in BPMN. In Manolopoulos, Y., Filipe, J., Constantopoulos, P., Cordeiro, J., eds.: ICEIS 2006 - Proceedings of the Eighth International Conference on Enterprise Information Systems: Databases and Information Systems Integration (III), Paphos, Cyprus, May 23-27, 2006. (2006) 440–443
40. Laue, R., Gruhn, V.: Complexity metrics for business process models. In Abramowicz, W., Mayr, H.C., eds.: 9th International Conference on Business Information Systems (BIS 2006). Volume 85 of Lecture Notes in Informatics. (2006) 1–12
41. Cardoso, J.: Process control-flow complexity metric: An empirical validation. In: Proceedings of IEEE International Conference on Services Computing (IEEE SCC 06), Chicago, USA, September 18-22, IEEE Computer Society (2006) 167–173
42. Rosemann, M., Recker, J., Indulska, M., Green, P.: A study of the evolution of the representational capabilities of process modeling grammars. In Dubois, E., Pohl, K., eds.: Advanced Information Systems Engineering, 18th International Conference, CAiSE 2006, Luxembourg, Luxembourg, June 5-9, 2006, Proceedings. Volume 4001 of Lecture Notes in Computer Science., Springer (2006) 447–461
43. Agarwal, R., Sinha, A.P.: Object-oriented modeling with UML: a study of developers' perceptions. Commun. ACM **46** (2003) 248–256
44. Sarshar, K., Loos, P.: Comparing the control-flow of epc and petri net from the end-user perspective. In Aalst, W., Benatallah, B., Casati, F., Curbera, F., eds.: Business Process Management, 3rd International Conference, BPM 2005, Nancy, France, September 5-8, 2005, Proceedings. LNCS 3649 (2005) 434–439