An Approach to Business Process Model Structuredness Analysis: Errors Detection and Cost-Saving Estimation

Dmytro Orlovskyi [1] and Andrii Kopp [1]

[1] *National Technical University "Kharkiv Polytechnic Institute", Kyrpychova str. 2, Kharkiv, 61002, Ukraine*

Abstract

This paper considers business process model structuredness issues, which are mostly related to inaccurate usage of gateways. According to related work in the process model structuredness domain, split gateways ought to match respective join gateways of the same type, while the existing mismatch measure allows evaluating model structuredness only by degrees of split and join gateways. Thus, the current measure of process model structuredness is not accurate enough and process model shortcomings may remain undetected, which may affect negatively model understandability, maintainability, and increase the error probability of business process models. Hence, error fixing costs may grow exponentially during later stages of the information system lifecycle. Therefore, we have proposed an improved gateway mismatch measure and a model to detect design issues and suggest changes necessary to achieve a sufficient level of business process model structuredness. The software tool for business process model structuredness analysis was developed to perform experiments with a large set of business process models of different industries. Analysis of obtained results, including sample business process models, detected design issues, and estimated efforts and cost-saving benefits are outlined. Conclusions were made, and future work was formulated.

Keywords

Business Process Model, Structuredness, Error Detection, Cost Estimation, Software Tool

1. Introduction: Related Work and Problem Statement

Business process modeling is the major Business Process Management (BPM) capability, which is used to describe organizational activities as the graphical diagrams. In general BPM is considered as the discipline, which combines IT (Information Technology) and management practices, and helps organizations to discover, analyze, re-design, implement, and monitor business processes [1]. BPM capabilities are used by organizations to provide products and services of high quality and find ways to improve organizational activities. Business processes are scenarios of activities, which execution is driven by events and decisions. Business processes are also considered as "chains of events, activities and decisions" according to [2]. Business process models are descriptions of business processes, which help to design and analyze of information systems, and used as the mechanism of communication among the stakeholders [3]. Thus, quality of business process models is vital for the successful implementation of BPM projects or, at least, for the requirements gathering for information systems development or upgrade.

This paper is structured in a following way: subsections 1.1 outlines related work in business process modeling and model quality measurement areas, while subsection 1.2 describes problem statement and research relevance. Section 2 demonstrates proposed approach to business process model structuredness analysis, which includes modeling errors detection, suggestion of necessary changes, and cost-saving estimation as possible benefit from business process model improvement. Section 3 demonstrates developed software tool used to perform experiments, outlines obtained results, their analysis, and

---

ICTERI-2021, Vol II: Workshops, September 28 – October 2, 2021, Kherson, Ukraine

EMAIL: orlovskyi.dm@gmail.com (A. 1); kopp93@gmail.com (A. 2)
ORCID: 0000-0002-8261-2988 (A. 1); 0000-0002-3189-5623 (A. 2)

CEUR Workshop Proceedings (CEUR-WS.org)

discussion. Section 4 contains conclusions that were made and future research directions that were formulated.

1.1. Related Work

There are numerous studies in the business process model quality field: SEQUAL Framework, The Guidelines of Modeling (GoM), Quality Framework for conceptual modeling, Seven Process Modeling Guidelines (7PMG), process model quality metrics and others [4]. Thus, quality of business process models could be considered as the degree to which a process model fulfills requirements of process modeling guidelines with respect to the ISO 9001 standards, which declare quality as the “degree to which a set of inherent characteristics fulfills requirements” [5].

Multiple business process modeling notations are used nowadays, among which the most widely used are EPC (Event-driven Process Chain) and BPMN (Business Process Model and Notation) – graphical modeling languages used to describe business processes. However, BPMN has replaced EPC as the most used business process modeling standard due to increasing popularity of BPMN in recent years [6].

Generic BPMN elements are events (start, end, and intermediate), tasks, connectors (or gateways), and sequence flows (see Fig. 1) [7].

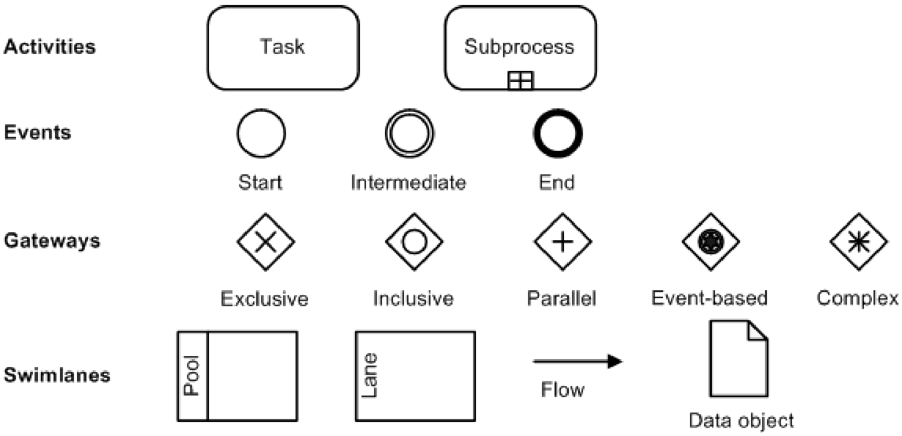

Figure 1: Generic BPMN modeling elements

In BPMN process models events and activities are connected through sequence flows. Gateways are special elements that indicate whether the process goes in parallel (AND), exclusively (XOR), or optionally (OR) [8].

Research of connector interplay includes the concept of business process model structuredness, which means that each split-connector (gateway) matches a corresponding join-connector of the same type (exclusive – XOR, parallel – AND, or inclusive – OR) [9, 10]. Since the connector mismatch might be the source of error, there were proposed the corresponding measure [9]:

$$MM_l = \left| \sum_{c \in S_l} d(c) - \sum_{c \in J_l} d(c) \right|, \qquad (1)$$

where:

- l is the connector type, $l \in \{or, xor, and\}$;
- $S_l \subseteq C_l$ is the subset of split-connectors of type l;
- $J_l \subseteq C_l$ is the subset of join-connectors of type l;
- c is the element of connectors set $c \in C_l$;
- $d(c)$ is the degree of element c.

The gateway mismatch coefficient is the sum of mismatch MM_l (1) for each gateway type l [9]:

$$MM = MM_{or} + MM_{xor} + MM_{and}. \quad (2)$$

There is limitation related to the fact that BPMN notation offers multiple end events. However, it is recommended to use start and end events consistently and events that in fact represent the same state should be merged in a single event. Hence, this limitation is turned into the additional highlight for the model designer to check whether multiple events represent the same event or distinguished events [11].

If connectors mismatch is high (i.e. when the sum of connector pairs that do not match with each other [10] is high), there is a higher probability that a process may contain errors, such as deadlocks (when the process execution is blocked [12, 13]) or lacks of synchronization (when multiple concurrent process paths cannot be synchronized [12, 13]).

However, we have noticed that the idea of structuredness (matching of each split-connector to a corresponding join-connector of the same type) is not directly declared by the known mismatch measure MM (2), since the only numbers of flows are controlled through the degrees of gateways, but not the numbers of gateways. While the recent research results in the field of connector interplay confirm that every split ought to match a respective join gateway of the same type [14, 15].

It is well-known that properly structured business process models can help to avoid errors in early stages of information system design [16]. Obviously, such errors and related fixing efforts should be somehow estimated. The first remarkable attempt to compute the cost of business processes using the BPMN notation is presented in [17]. However, estimation of the cost of BPMN-driven business process modeling, based on the COCOMO (The Constructive Cost Model) model, used for decades in the software engineering domain, is proposed in [18]. Also from the software engineering domain we know that error fixing efforts grow exponentially during the project lifecycle and may cost 100 times more in the maintenance stage in compare to the design stage [19].

1.2. Problem Statement

Examples of deadlock and lack of synchronization errors are shown in Fig. 2, where the initiated process instances (100 instances) are compared to the resolved instances (0 instances for the deadlock and 200 for the lack of synchronization), which should be of the same number as the initiated.

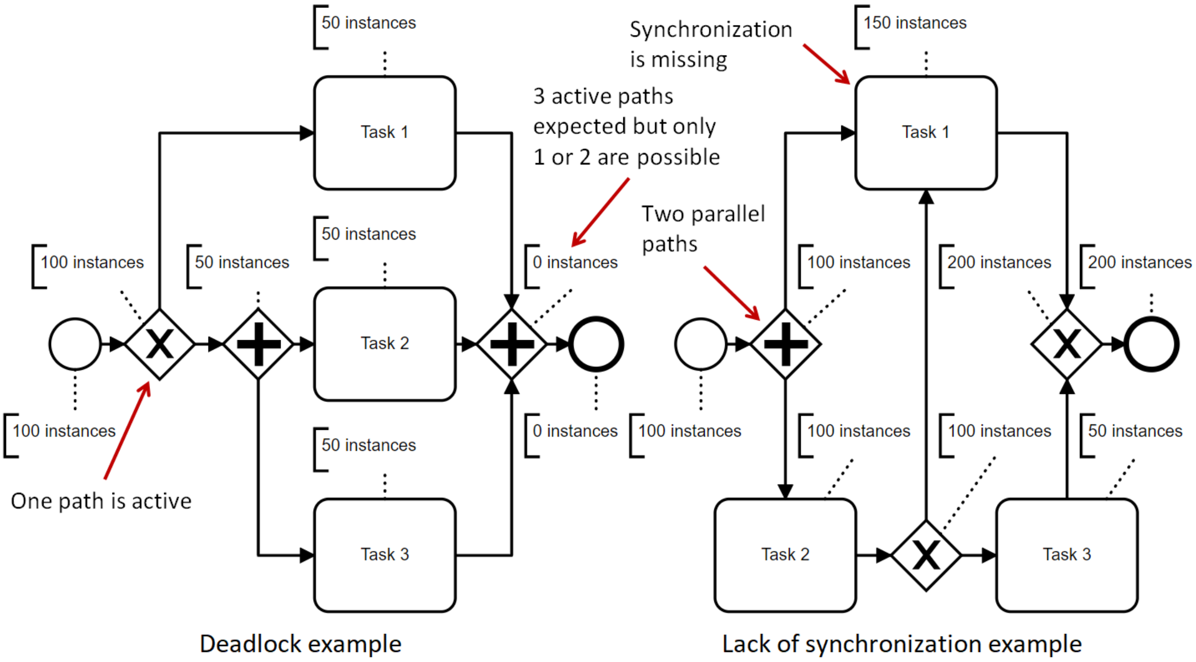

Figure 2: Examples of business process errors caused by mismatched connectors

As it is demonstrated in Fig. 2, the gateway mismatch phenomenon is indeed the source of occurred deadlock and lack of synchronization issues:

- Deadlock: the XOR-gateways mismatch is $MM_{xor} = 3$ (1) and the AND-gateways mismatch is $MM_{and} = 1$ (1), so the total gateway mismatch coefficient is $MM = 4$ (2).
- Lack of synchronization: the AND-gateways mismatch $MM_{and} = 3$ (1) and the XOR-gateways mismatch is $MM_{xor} = 0$ (1), so the total gateway mismatch coefficient is $MM = 3$ (2).

Thus, considered examples (Fig. 2) demonstrate how the gateway mismatch coefficient can be used to detect connector interplay issues, which cause deadlocks and synchronization errors in business process models executed by BPMN engines or "hard-coded" in custom information systems. Hence, some business process tasks may be never reached, while other tasks may be executed more times than necessary. In other words, organizational performance may decrease, while expenses may increase.

Hence, detection of connector interplay errors becomes the relevant research subject. Moreover, structuredness errors increase the cost of implementation and subsequent execution of business processes. Therefore, this study is aimed at early detection of structuredness errors in business process models and cost-saving estimation of errors fixing in different stages of business process lifecycle.

2. Proposed Approach to Business Process Model Structuredness Analysis

2.1. Improved Gateway Mismatch Measure

In order to consider not only correspondence between branched and merged control flows, but also direct correspondence between split and join gateways according to the structuredness definition [10], let us propose the following modification of the MM_l measure:

$$MM'_l = \max\left\{\left|\sum_{c\in S_l} d(c) - \sum_{c\in J_l} d(c)\right|, \left|\left|S_l\right| - \left|J_l\right|\right|\right\}. \tag{3}$$

This modification is expected to consider direct correspondence between split and join gateways, as well as correspondence of control flows. Improved measure should take into account not only basic gateways, which are exclusive (XOR), parallel (AND), and inclusive (OR) logical connectors, but also special BPMN 2.0 gateways (see Fig. 1), such as event-based gateways (exclusive or parallel) that check which event has been occurred instead of checking which condition has been met ($event$) and complex gateways that use textual descriptions to explain complex logic of business process flows ($complex$).

The example below (see Fig. 3) shows weakness of the original mismatch measure in compare to its modified version. Domain-specific examples of fault BPMN models are given in section 3.2.

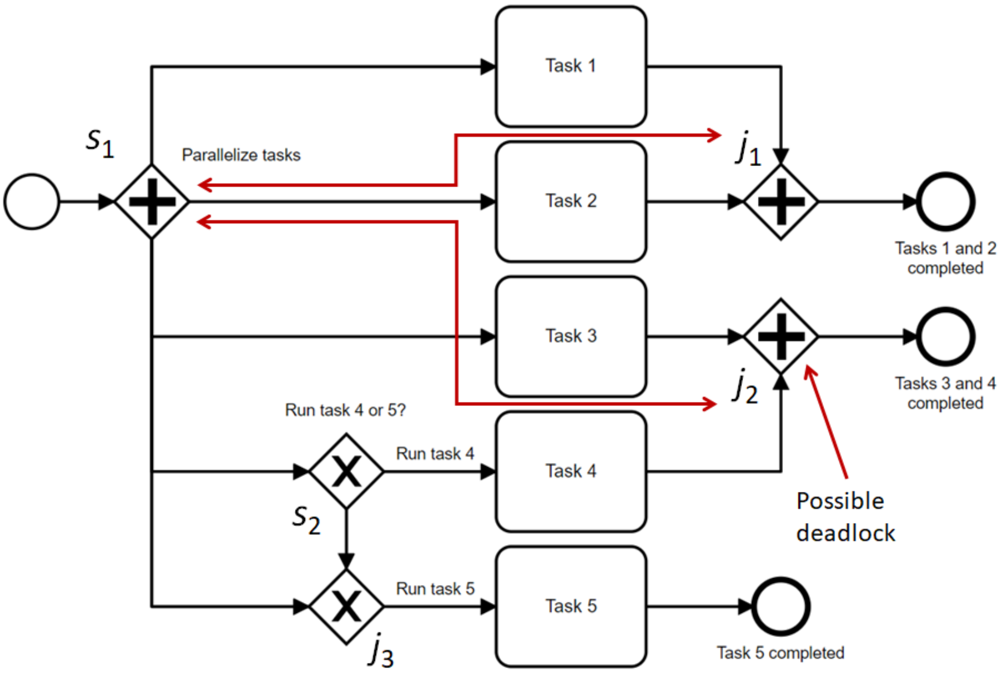

Figure 3: Sample business process model with gateway mismatch

It is clear that there is the gateway mismatch shown in Fig. 3 since there is single AND-split and two AND-joins, even though the sum of degrees of AND-split and AND-join gateways is equal to 6 (i.e. 1 incoming and 5 outgoing flows for s_1, and 2 incoming and 1 outgoing flows for both j_1 and j_2 respectively). But it is interesting, that the original measure MM_{and} (1) does not show any mismatch:

$$MM_{and} = \left| \sum_{c \in S_{and}} d(c) - \sum_{c \in J_{and}} d(c) \right| = \left|(1+5) - [(1+2)+(1+2)]\right| = \left|6-6\right| = 0. \tag{4}$$

As for the proposed modification MM'_{and} (3), we have obtained the following measure for the demonstrated model (see Fig. 3):

$$\begin{aligned} MM'_{and} &= \max\left\{ \left| \sum_{c \in S_{and}} d(c) - \sum_{c \in J_{and}} d(c) \right|, \left\| S_{and} \right| - \left| J_{and} \right\| \right\} = \\ &= \max\{\left|(1+5) - [(1+2)+(1+2)]\right|, \left|1-2\right|\} = \max\{\left|6-6\right|, \left|1-2\right|\} = 1. \end{aligned} \tag{5}$$

Using the example of (4) and (5) we have demonstrated how implicit faults could be detected using proposed modification of the connectors mismatch measure. As it is shown in Fig. 3, the deadlock is possible in j_2, since when the s_2 condition is set to run the “Task 5” path, there will be only one active path that triggers j_2, whereas two active paths must trigger j_2 in order to resolve the synchronization of parallel paths. Hence, the business process instance will be never terminated at the “Tasks 3 and 4 completed” event. But when the original mismatch coefficient MM_{and} was used (4), it did not signalize about such severe connector interplay error as the deadlock. Only the modified mismatch measure MM'_{and} (5) noticed that such defect is presented in the business process model (Fig. 3).

It is also necessary to detect gateways that are neither splits nor joins because of their incorrect degrees (multiple incoming and outgoing flows at the same time, one incoming and one outgoing flow at the same time, missing incoming or outgoing flows).

Finally, modified gateway mismatch measure may be the following:

$$MM' = \sum_{l \in G} MM'_l + \sum_{l \in G} (|C_l| - |S_l| - |J_l|), \tag{6}$$

where $G = \{or, xor, and, event, complex\}$ is the set of gateway types.

The physical meaning of gateway mismatch measures, both the original MM (2) and the modified MM' (6), is the number of changes (added or removed control flows or gateways) required to bring a business process model into the well-structured condition.

Therefore, a business process model that requires one action in a BPMN editor at the design stage, may lead to necessity of literally a hundred times more actions (e.g. information system components re-design) at the maintenance stage of a poor-structured workflow [16, 19].

2.2. Detection of Business Process Modeling Errors

After the mismatch was detected, it is required to define weak spots in the business process model. These faults could be found using the following model:

$$\begin{gathered} W = \sum_{l \in G} \max\left\{ \left| \sum_{c \in S_l} d(c) \cdot (1 - x_l^1) - \sum_{c \in J_l} d(c) \cdot (1 - x_l^2) \right|, \right. \\ \left. \left| |S_l| \cdot (1 - x_l^3) - |J_l| \cdot (1 - x_l^4) \right| \right\} + \sum_{l \in G} \left[(|C_l| - |S_l| - |J_l|) \cdot (1 - x_l^5) \right] \to \min_{x_l^i}, \\ x_l^i \in \{0,1\},\ i = \overline{1,5},\ l \in G, \end{gathered} \tag{7}$$

where:

- x_l^1 indicates that splitted paths of type l do not match joined paths of type l – changes of flows outgoing from split connectors are suggested, $l \in G$;
- x_l^2 indicates that joined paths of type l do not match splitted paths of type l – changes of flows outgoing from join connectors are suggested, $l \in G$;
- x_l^3 indicates that split connectors of type l do not match join connectors of type l – changes of split connectors number are suggested, $l \in G$;
- x_l^4 indicates that join connectors of type l do not match split connectors of type l – changes of join connectors number are suggested, $l \in G$;
- x_l^5 indicates that connectors of type l are neither splits nor joins, hence, re-arrangement measures are suggested, $l \in G$.

This is a 0-1 discrete optimization problem with binary variables. These variables could be found using the following equations, since it is required to eliminate deviations between split and join paths and connectors in order to achieve the minimum (ideally zero) gateway mismatch:

$$x_l^1 = \begin{cases} 1, \sum_{c \in S_l} d(c) \neq \sum_{c \in J_l} d(c) \text{ and } \sum_{c \in S_l} d(c) > 0, \\ 0, \text{ else}, \end{cases}, l \in G, \tag{8}$$

$$x_l^2 = \begin{cases} 1, \sum_{c \in S_l} d(c) \neq \sum_{c \in J_l} d(c) \text{ and } \sum_{c \in J_l} d(c) > 0, \\ 0, \text{ else}, \end{cases}, l \in G, \tag{9}$$

$$x_l^3 = \begin{cases} 1, |S_l| \neq |J_l| \text{ and } |S_l| > 0 \\ 0, \text{ else} \end{cases}, x_l^4 = \begin{cases} 1, |S_l| \neq |J_l| \text{ and } |J_l| > 0 \\ 0, \text{ else} \end{cases}, l \in G, \tag{10}$$

$$x_l^5 = \begin{cases} 1, |C_l| - |S_l| - |J_l| > 0 \\ 0, \text{ else} \end{cases}, l \in G. \tag{11}$$

Considering the fixed size of optimization problem, the direct search may be used to find x_l^i (8) – (11), $i = \overline{1,5}$, $l \in G$ (see Fig. 4).

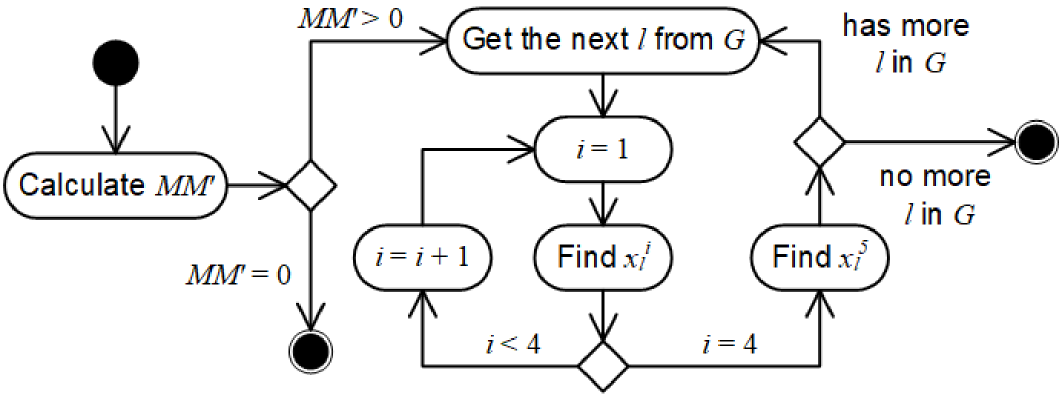

Figure 4: Direct search of optimization variables

2.3. Estimation of Costs to Fix Detected Errors in Different Project Stages

Using calculated values x_l^i, $i = \overline{1,5}$, $l \in G$ we can estimate relative efforts and costs to fix errors, which could be prevented by eliminating detected design shortcomings. Project Effort (PE) estimation equation was inspired by the COCOMO model, which was already used by authors of [18] to estimate cost of business process modeling.

However, original PE equation was modified in order to take into account growth of costs to fix defects during the information system lifecycle stages [19]:

$$PE_s = \lambda_s \cdot H \cdot a \cdot \left(10^{-3} \cdot \sum_{l \in G} \sum_{i=1}^{5} x_l^i \right)^b, s = \overline{1,4}, \tag{12}$$

where:

- λ_s is the coefficient of efforts and costs that depends on the stage in which defects were detected: $\lambda_1 = 1$ for design stage, $\lambda_2 = 6.5$ for implementation stage, $\lambda_3 = 15$ for testing stage, and $\lambda_4 = 100$ for maintenance stage [18];
- a and b are COCOMO parameters (2.4 and 1.05 for "easy" projects, 3 and 1.12 for "medium" projects, and 3.6 and 1.2 for "complex" projects respectively);
- H is the hours per person-month to get estimations in person-hours, $H = 152$;
- the sum of obtained values x_l^i, $i = \overline{1,5}$, $l \in G$ considered as required units of work to bring a business process model to the well-structured condition according, which corresponds to assumptions made in [16];
- since the original COCOMO model requires KLOC (kilo lines of code) to be used, the sum of required business process modeling actions is multiplied by 10^{-3}.

Following section 3 demonstrates evaluation of connector interplay using the improved gateway mismatch measure (6), detection of design flaws, caused by gateway mismatch, using the optimization model (7), and cost-saving estimation (12).

3. Results and Discussion

3.1. Software Tool Development and Experimental Usage

In order to validate the proposed connectors mismatch measure MM' we have used the set of BPMN business process models from the open GitHub repository provided by Camunda [20]. In this public repository are presented different versions of the products dispatch, insurance recourse, credit scoring, and self-service restaurant business process models designed during Camunda training sessions for BPMN modeling. In total we have used 6137 process models that do not contain syntax errors but may contain design flaws [20].

In order to process such data volume we have developed a software tool for business process model structuredness analysis. This is a NodeJS application for now provided only with a command line interface (CLI). However, its future growth assumes web user interface (UI) and application programming interface (API) implementation.

General system architecture of the software tool is shown in Fig. 5.

Developed software tool consists of the following main components (see Fig. 5):

- parsing component, responsible for the processing of BPMN 2.0 files, which are basically XML (eXtensible Markup Language) documents prepared with respect to the specialized schema;
- measurement component, responsible for calculation of the original MM and modified MM' measures (in order to compare received results);
- suggestion component, responsible for detection of business process modeling errors and calculation of corresponding values x_l^i, $i = \overline{1,5}$, $l \in G$; this component is also responsible for cost-saving estimation;
- main component ("index.js"), which links all the software parts together, loads business process models for parsing, and stores obtained results to the text file.

In order to compare results obtained using the original MM and modified MM' measure, the following ratio of fault models have been calculated:

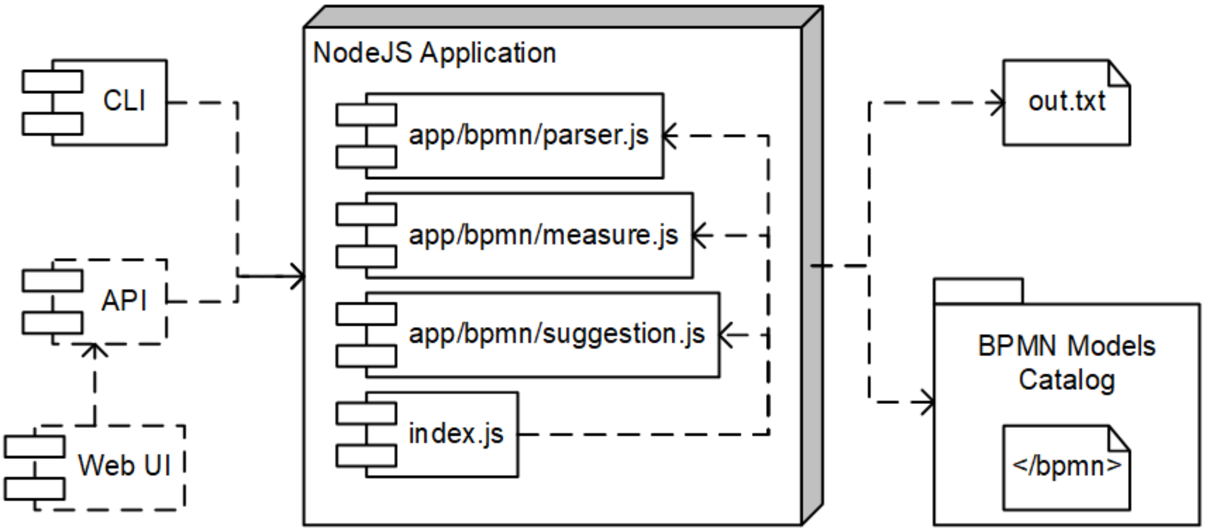

Figure 5: General system architecture of the software tool

$$\text{FaultModels} = \frac{|\{m_i \in M \mid y > 0\}|}{|M|}, \qquad (13)$$

where:

- M is the set of process models $M = \{m_1, m_2, \ldots, m_n\}$, where n is the number of models in the test set.
- y is the used measure: MM or MM'.

As it is shown in bar chart below (see Fig. 6), MM' can detect 15% more models of poor structuredness than original mismatch measure MM.

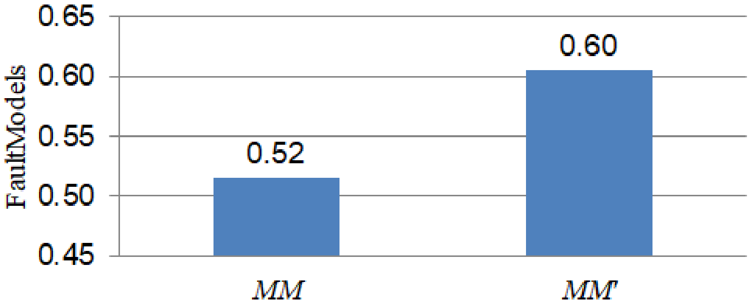

Figure 6: Comparison of the obtained results using original and improved mismatch measures

Among 6137 business process models there were detected 3712 fault models using the modified measure MM'. Whereas only 3163 fault models were detected using the original measure MM. All models detected using MM are included in the set of models detected using MM'. Obtained results (see Fig. 6) evidence better accuracy (13) of the modified measure MM' (0.60) in compare to original measure MM (0.52).

3.2. Sample Business Process Models and Detected Modeling Errors

Let us demonstrate examples of BPMN models from the Camunda GitHub repository [20]. These models demonstrate features of the following branches of business:

- “Dispatch of goods” (see sample model fragment in Fig. 7);
- “Insurance payment recourse” (see sample model fragment in Fig. 8);
- “Credit scoring” (see sample model fragment in Fig. 9);
- “Self-service restaurant” (see sample model fragment in Fig. 10).

Demonstrated design flaws, caused by gateway mismatch, have been detected using the modified mismatch measure MM' and proposed optimization model W.

First example elaborates fragment of dispatch of goods business process model (see Fig. 7). Gateway mismatch for this model is $MM' = 5$.

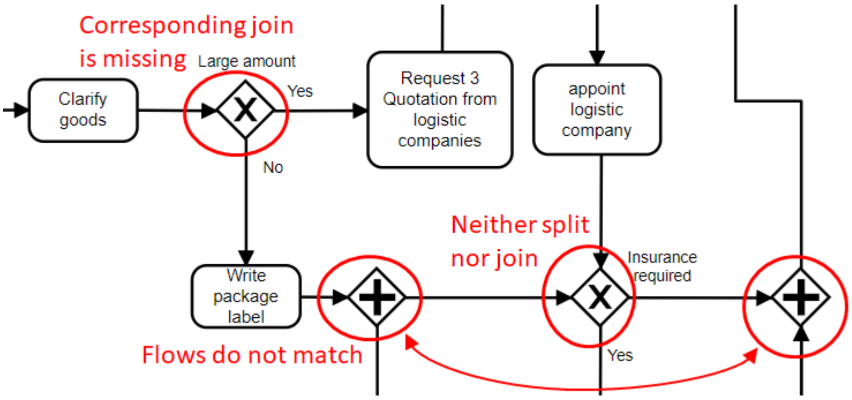

Figure 7: Fragment of dispatch of goods business process model

Second example shows fragment of insurance payment recourse business process model (see Fig. 8). Gateway mismatch for this model is $MM' = 5$.

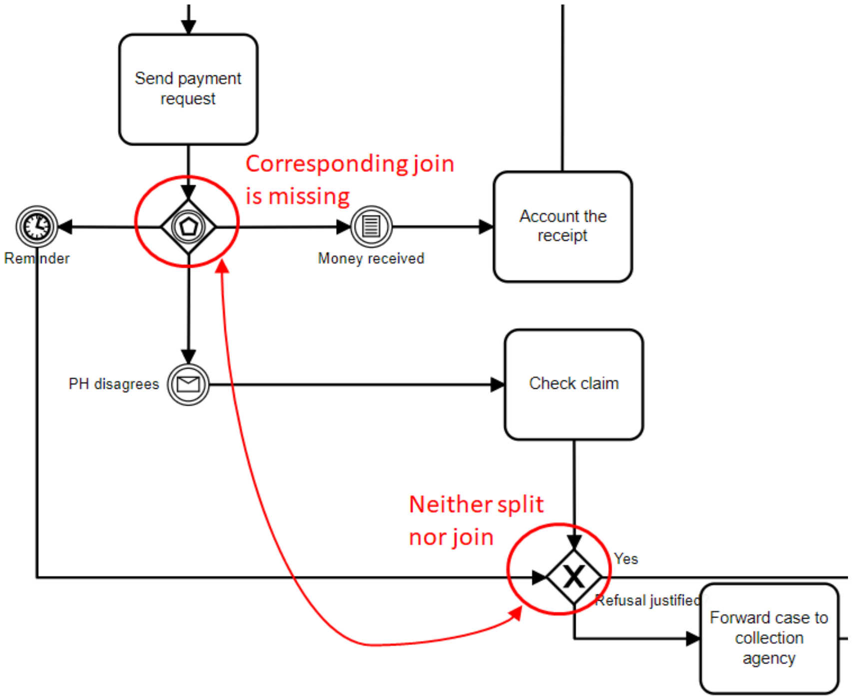

Figure 8: Fragment of insurance payment recourse business process model

Third example demonstrates fragment of credit scoring business process model (see Fig. 9). Gateway mismatch for this model is $MM' = 6$.

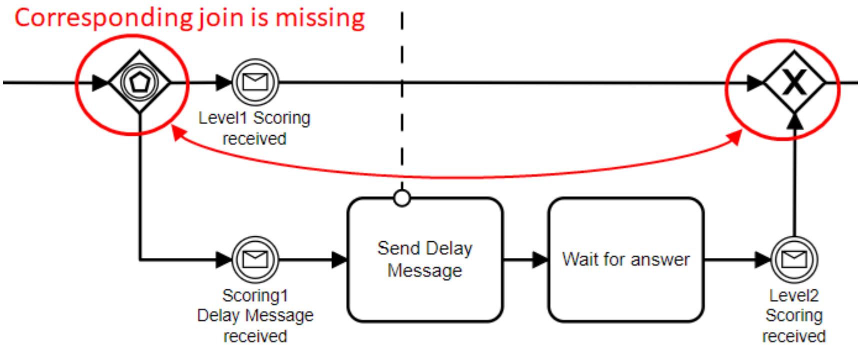

Figure 9: Fragment of credit scoring business process model

Fourth example demonstrates fragment of self-service restaurant business process model (see Fig. 10). Gateway mismatch for this model is $MM' = 4$.

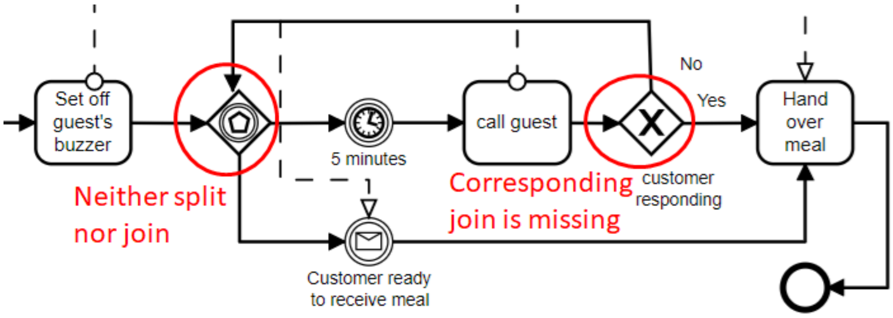

Figure 10: Fragment of self-service restaurant business process model

Full business process models, which fragments are depicted in Fig. 7 – Fig. 10, could be seen in the GitHub repository of this research project (in the "results.pdf" file), where the software tool source code is also stored [21].

In order to demonstrate numerical results obtained for considered process models (see Table 1), the following notations should be introduced:

- $D_S = \sum_{c \in S_l} d(c)$ is the sum of degrees of split connectors;
- $D_J = \sum_{c \in J_l} d(c)$ is the sum of degrees of join connectors.

Table 1
Obtained results of connector interplay analysis for considered models

Model	$l \in G$	D_S	D_J	$\lvert S_l \rvert$	$\lvert J_l \rvert$	MM'	x_l^1	x_l^2	x_l^3	x_l^4	x_l^5
Fig. 7	*xor*	3	0	1	0	5	1	0	1	0	1
	and	3	4	1	1		1	1	0	0	0
Fig. 8	*xor*	3	3	1	1	5	0	0	0	0	1
	event	4	0	1	0		1	0	1	0	0
Fig. 9	*xor*	0	3	0	1	6	0	1	0	1	0
	event	3	0	1	0		1	0	1	0	0
Fig. 10	*xor*	3	0	1	0	4	1	0	1	0	0
	event	0	0	0	0		0	0	0	0	1

Detected design flaws (see Fig. 7 – Fig. 10) negatively affect structuredness of analyzed BPMN diagrams, which became less understandable and more error prone because of possible deadlocks or lacks of synchronization [12].

Calculated MM' values allow to detect existence of these flaws and the optimization model W allows to detect concrete design flaws that negatively impact on structuredness of business process models. Variables x_l^i, $i=\overline{1,5}$, $l \in G$ could be used to signalize presence of such mistakes and suggest corresponding changes in order to prevent propagation of design errors to later stages of information system lifecycle, and, by that, avoid exponentially grown efforts and costs to fix errors originated from the design stage.

3.3. Estimation of Benefits from Business Process Model Improvement

Efforts and relative cost-saving benefits from business process model structuredness improvement could be estimated using the PE_S in person-hours (see Fig. 11).

Considered business process models (see Fig. 7 – Fig. 10) are of following sizes (total number BPMN elements) [21]:

- $S=14$ for dispatch of goods;
- $S=15$ for insurance recourse;
- $S=18$ for credit scoring;
- $S=34$ for self-service restaurant business process.

Since sizes of considered models are all below the threshold of 50 elements [10], we may consider them as simple enough and, therefore, take COCOMO parameters $a=2.4$ and $b=1.05$ (used for "easy" projects) [18].

Estimated cost-saving benefits calculated using these assumptions are shown in Fig. 11.

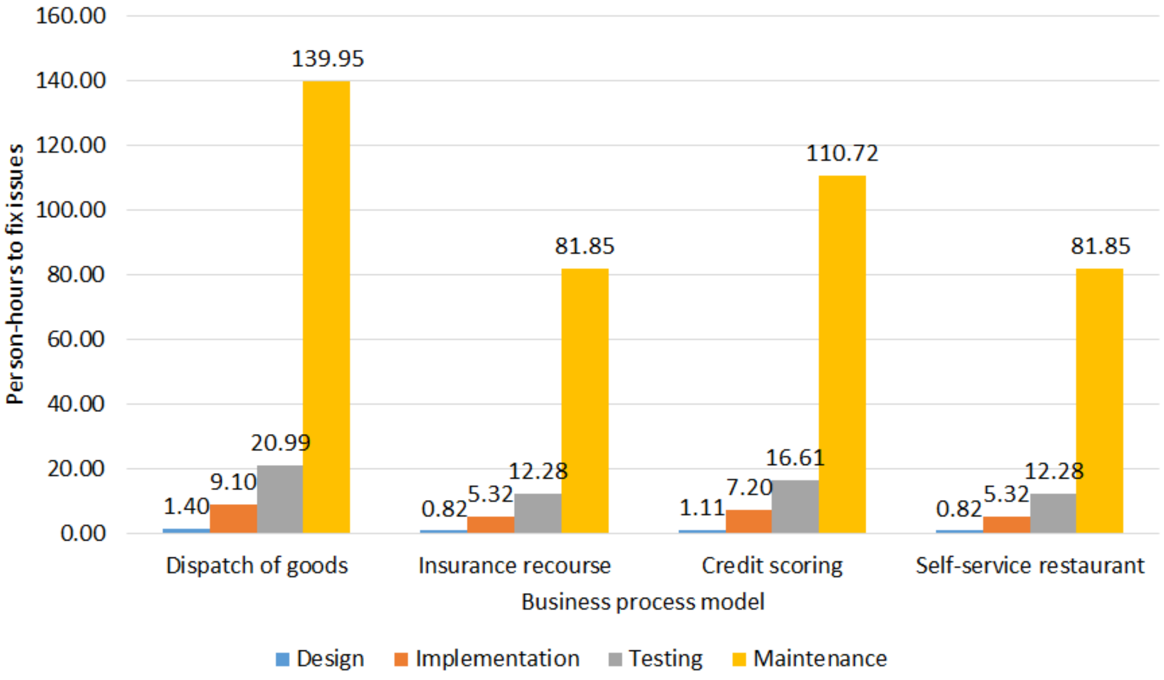

Figure 11: Estimated efforts and cost-saving benefits from business process model structuredness improvement

As it is shown in Fig. 11, relatively low efforts of detected errors fixing in BPMN diagrams (from 0.82 to 1.40 person-hours, which is absolutely adequate considering medium size and complexity of used business process models) may grow tremendously and cost 12.28 – 20.99 person-hours in the implementation stage, and 81.85 – 139.95 person-hours in maintenance stage. Thus, it is much cheaper to fix possible defects in the stage of business process modeling than it is to fix defects in already deployed business processes implemented by a certain information system. Estimated efforts to rework model should be demonstrated together with calculated measures and given suggestions by the software tool, as well as approximate costs of defects fixing in later project stages to motivate users.

3.4. Accuracy Evaluation of Suggested Business Process Model Improvements

Using obtained values x_l^i, $i=\overline{1,5}$, $l \in G$, we can define accuracy of the obtained recommendations. The main challenge is that event-based and complex gateways can be mixed with simple gateways. It means that x_l^3 and x_l^4 obtained for event-based and complex gateways $l \in \{event, complex\}$ may give irrelevant suggestions about design drawbacks in BPMN models:

$$Incorrect = x_{event}^3 + x_{event}^4 + x_{complex}^3 + x_{complex}^4,$$

$$Total = \left(\sum_{l \in G} \sum_{i=1}^{5} x_l^i \right), \tag{14}$$

$$Correct = Total - Incorrect,$$

where:

- $Incorrect$ is the number of presumably incorrect suggestions.
- $Total$ is the total number of obtained suggestions.
- $Correct$ is the total number of presumably correct suggestions calculated as the difference between the number of all suggestions and the number of presumably correct suggestions.

Therefore, accuracy could be evaluated as the following ratio of number of correct suggestions to the total number of suggestions (14):

$$Accuracy = \frac{Correct}{Total} = \frac{Correct}{Correct + Incorrect} = \frac{13}{13+2} = 0.87. \tag{15}$$

According to (15), there are 15 suggestions on structuredness improvement of analyzed business process models were obtained in total, i.e. $Total = 15$.

Whereas, among 15 suggestions, 2 could be considered as presumably incorrect ($Incorrect = 2$), since there were obtained $x_{event}^3 = 1$ values (Table 1) for the insurance payment recourse (see Fig. 8) and credit scoring (see Fig. 9) business process models after solving the optimization problem (7).

According to (14), there are 13 suggestions were obtained as those, which could be considered as presumably correct ($Correct = 13$).

Hence, the accuracy of obtained suggestions on structuredness improvement of analyzed business process models is 0.87 (15). Calculated accuracy metric (15) could be considered as the “confidence” of given recommendations and should accompany them when demonstrated to a user. Therefore, the bigger accuracy value is, the more carefully given suggestions should be taken into account by users responsible for BPMN diagramming. Moreover, the bigger accuracy value is, the more reasonable are suggested approximate costs of defects fixing in later project stages, which makes detected errors and provided recommendations even more valuable and relevant.

4. Conclusion and Future Work

In this paper we have proposed the modified mismatch measure, which avoids drawbacks of the existing mismatch measure, and a model to define changes, which are necessary to achieve the sufficient level of BPMN business process model structuredness. In contrast with the existing measure, modified measure takes into account not only the correspondence between sequence flows, which are outgoing from split gateways and incoming to respective join gateways of the same type, but also the direct correspondence between split gateways and respective join gateways of the same type. It allows to measure structuredness of business process model more accurately and to detect design shortcomings that negatively impact on model understandability and maintainability, which lead to extra efforts and associated costs for error fixing during the later stages of the information system lifecycle. Proposed optimization model and the software tool are aimed to provide suggestions on how BPMN model structuredness could be improved by adding or removing sequence flows and gateways. Obtained results demonstrate prevalence of the improved gateway mismatch measure, as well as concrete design shortcomings detected using the optimization model. Estimated efforts and cost-saving benefits have shown that it is much cheaper to design well-structured business process models that fix defects in deployed information systems that run described business processes.

However, this study focuses on business process models in the BPMN notation only and does not cover neither structuredness analysis nor cost-saving estimation of business process models in other notations, such as IDEF0, IDEF3, EPC (Event-driven Process Chain), and DFD (Data Flow Diagram). Also this paper considers a procedure that only detects errors but does not suggest any recommendations on models improvement.

Therefore, future research in this domain includes elaboration of automatic business process model transformation methods based the obtained recommendations, as well as corresponding software components of the designed architecture. In addition to BPMN, another business process modeling notations, mentioned above, should be considered. There should be also elaborated ways to increase accuracy of obtained suggestions on business process model structuredness improvement.

5. References

[1] N. Ahrend, Opportunities and limitations of BPM initiatives in public administrations across levels and institutions, Doctoral thesis, Humboldt University, Berlin, 2014.

[2] M. Dumas, M. La Rosa, J. Mendling, H. Reijers, Introduction to Business Process Management, in: Fundamentals of Business Process Management, Springer, Heidelberg, 2013, pp. 1–33.

[3] F. Kahloun, S. A. Ghannouchi, Classification Algorithm for Assessing the Quality Criteria for Business Process Models, in: Hybrid Intelligent Systems, HIS 2017, Advances in Intelligent Systems and Computing, vol. 734, Springer, Cham, 2017, pp. 71–81.

[4] J. Pavlicek, R. Hronza, K. Jelinkova, The business process model quality metrics. In: Enterprise and Organizational Modeling and Simulation, in: EOMAS 2017, Lecture Notes in Business Information Processing, vol. 298, Springer, Cham, 2017, pp. 134–148.

[5] R. Tricker, ISO 9001:2015 for Small Business, Taylor & Francis, 2016.

[6] A. N. Khudori, T. A. Kurniawan, F. Ramdani, Quality Evaluation of EPC to BPMN Business Process Model Transformation, Journal of Information Technology and Computer Science 5(2) (2020) 207–220.

[7] N. A. Mohammadi, M. Heisel, A Framework for Systematic Refinement of Trustworthiness Requirements, Information 8(2) (2017) 46–68.

[8] H. G. Ceballos, V. Flores-Solorio, J.P. Garcia, A Probabilistic BPMN Normal Form to Model and Advise Human Activities, in: Engineering Multi-Agent Systems, EMAS 2015, Lecture Notes in Computer Science, vol. 9318, Springer, Cham, 2015, pp. 51–69.

[9] J. Mendling, Detection and prediction of errors in EPC business process models, Doctoral thesis, WU, Vienna, 2007.

[10] J. Mendling, L. Sanchez-Gonzalez, F. Garcia, M. La Rosa, Thresholds for error probability measures of business process models, Journal of Systems and Software 85(5) (2012) 1188–1197.

[11] F. Corradini, A. Ferrari, F. Fornari, S. Gnesi, A. Polini, G. O. Spagnolo, A guidelines framework for understandable BPMN models, Data & Knowledge Engineering 13 (2018) 129–154.

[12] K. Kluza, G. J. Nalepa, J. Lisiecki, Square complexity metrics for business process models, in: Advances in Business ICT, Advances in Intelligent Systems and Computing, vol. 257, Springer, Cham, 2014, pp. 89–107.

[13] A. Mroczek, A. Ligeza, A Note on BPMN Analysis. Towards a Taxonomy of Selected Potential Anomalies, in: Proceedings of the 2014 Federated Conference on Computer Science and Information Systems, vol. 2, 2014, pp. 1097–1102.

[14] F. Kahloun, S. A. Ghannouchi, Improvement of quality for business process modeling driven by guidelines, Procedia Computer Science 126 (2018) 39–48.

[15] J. Krogstie, SEQUAL Specialized for Business Process Models, in: Quality in Business Process Modeling, Springer, Cham, 2016, pp. 103–138.
[16] L. Sanchez-Gonzalez et al., Quality indicators for business process models from a gateway complexity perspective, Information and Software Technology 54(11) (2012) 1159–1174.
[17] P. B. Sampathkumaran, Computing the Cost of Business Processes, Dissertation, Munchen, 2013.
[18] F. Baumann, A. Milutinovic, D. Roller, Software engineering inspired cost estimation for process modelling, International Journal of Economics and Management Engineering 10(2) (2016) 576–586.
[19] M. Soni, Defect prevention: reducing costs and enhancing quality, iSixSigma, 2015. URL: https://www.isixsigma.com/industries/software-it/defect-prevention-reducing-costs-and-enhancing-quality/.
[20] BPMN for research, 2015. URL: https://github.com/camunda/bpmn-for-research.
[21] BPMN structuredness, 2021. URL: https://github.com/freebpmnquality/bpmn_structuredness.