



Measuring Scalable Business Process Model Complexity Based on Basic Control Structure

Muhammad Ainul Yaqin^{1,2*}

Riyanarto Sarno²

Siti Rochimah²

¹*Department of Informatics, Universitas Islam Negeri Maulana Malik Ibrahim Malang, Indonesia*

²*Department of Informatics, Institut Teknologi Sepuluh Nopember Surabaya, Indonesia*

* Corresponding author's Email: yaqinov@ti.uin-malang.ac.id

Abstract: The business process model is a representation of business activities illustrated through diagram notations. This model is composed of repeated specific patterns called basic control structure. Each basic control structure has a level of complexity. The metrics for formulating existing complexity are very diverse, but can only define complexity partially and are less sensitive to small changes in the structure of the business process model. In this paper, we propose a formula of complexity metric that can indicate small changes in structure, type of branching logic, number of branches, loops, and depth. We call it the Yaqin complexity formula. To get the Yaqin complexity formula, we carried out several activities. These activities are identifying the metrics involved using the Goal-Question-Metric (GQM) method, then formulating the metric complexity, the next activity is testing the formula with several business process models and analyzing the test results, and then validating the Yaqin complexity formula using the Weyuker's process framework. The Yaqin complexity formula, which involves seven parameters, is proven to be more comprehensive than other complexity formulas that involve less than seven parameters. The Yaqin complexity formula also proved to be more sensitive to other complexity formulas, where 7 out of 8 cases affected the Yaqin complexity metric. The validation results state that the Yaqin complexity formula meets 8 of 9 Weyuker's properties. Thus we have succeeded in formulating the Yaqin complexity, which is more comprehensive in involving parameters and more sensitive in detecting small changes in the structure of the business process model.

Keywords: Complexity metrics formula, Business process model, Workflow pattern.

1 Introduction

Business processes are defined as a set of structured and interrelated activities that produce specific products or services [1]. Some business activities are composed in such a way as a business process model. The business process model is a representation of functions related to business activities such as input, control, output, and resources. According to the Business Process Model Notation (BPMN), the business process model consists of several components; event, activity, and control flow [2]. The components are simply composed to form a basic pattern. These patterns are called Workflow Patterns (WP). The WP were proposed by van der Aalst, ter Hofstede, Kiepuszewski, and Barros [3]. They defined eight groups of workflow patterns and

detailed them into 43 workflow patterns. Nick Russel, ter Hofstede, van der Aalst, and Nataliya Mulyar completed it by providing a systematic description and formal description in Colored Petri Net (CPN) notation [4].

The business process model is composed of several scalable Basic Control Structure (BCS) [5]. BCS is part of the WP. When BCS are composed to form a business process model, it has complexity. The complexity of the business process model describes the state of the problem that requires resolution [6]. Complexity can be measured in various ways and metrics. The proposed metrics also have many kinds. Jan Mendling, in 2008 in his book, explained about 28 metrics used to measure the size of the business process model [7]. He used the Event-driven Process Chain (EPC) notation. Muketha, Ghani, Selamat, and Atan in 2010 conducted a survey

of business process complexity metrics and how to validate them. Twenty-five metrics for business process complexity were identified [8]. Polancic and Cegnar, in 2017, surveyed the complexity metrics of the business process model. The survey produced 66 metrics of complexity that had been proposed [9]. In 2012 Carlo Corti created software to measure business process model metrics with BPMN notation [10].

Measurement of the business process model complexity has been carried out by several researchers, namely Cardoso in 2008, proposed the Control Flow Complexity metric (CFC) [11], which is an adaptation of Cyclomatic Complexity proposed by McCabe in 1976 [12]. Gruhn and Laue in 2006, proposed Cognitive Weight (CW) [13], which is a modification of the Cognitive Functional Size (CFS) proposed by Shao and Wang in 2003 [5]. Vanderfeesten, Cardoso, and Mendling in 2007 proposed Cross-Connectivity (CC) based on Cognitive Complexity as a metric to measure complexity [14]. The most comprehensive and sensitive complexity metric than the one proposed above is the Cognitive Activity Depth Arc Control flow (CADAC) proposed by Emriye Coskun in 2014 [15]. Yaqin also proposed a metric measurement of complexity called Scale in 2017 involving CFC parameters and the number of components of the business process model [16]. The complexity metrics they propose have weaknesses; they can only calculate partial complexity metrics and are less sensitive to small changes that occur in the measured business process model. The papers that we use as references in this study are more than three years old because, in the field of measuring the business process model complexity metric, we do not find new complexity metric findings, be it metric atomic or composite metric. The complexity metric that we make the primary comparison is CADAC, the most comprehensive and sensitive formula among the complexity metrics ever proposed.

In this paper, we examine and discuss how to measure the complexity metric of business process models based on their BCS more comprehensively and are more sensitive to small changes in the structure of business process models. The purpose of this research is to determine the formula for the complexity of business process models that are more comprehensive and sensitive based on the BCS that forms it.

The proposed contribution is the business process model complexity metric that is more comprehensive and sensitive to small changes that occur in the business process model. The more comprehensive complexity metrics are defined as a metric that

includes more measurement parameters to get the metric value of the business process model complexity. The more sensitive complexity metrics are defined as metrics that can affect the complexity of the business process model for structural changes in the business process model. We have defined the structural change into two parts. They are significant change and small change. Significant changes are changes that involve adding components and changes of logic to the business process model. In contrast, small changes are changes that occur without the addition of components and changes of logic to the structure of the business process model. The more comprehensive and sensitive complexity metric is essential to improve accuracy in estimating application development costs, increasing accuracy in application configuration that serves the business process.

This research is part of a broader scope of the development of enterprise architecture growth metrics research. One of the stages in the research is the development of the business process model complexity metric. The formula for complexity metric is useful as one of the components needed to simulate the growth of an enterprise architecture that is ultimately used to determine the configuration of information systems based on the business architecture complexity. The business process model complexity metric is also useful for getting the number of resources needed to run a business process [17].

This paper is organized as follows, background, problem formulation, related research, and contributions are presented in section 1. Section 2 explains the theories related to this research, namely about BCS, BCS metrics, and the complexity of business process models. Section 3 describes the methods and stages used to propose the complexity metric for business processes models. Then in section 4, we explain the experimental results and their discussion. Section 5 describes the conclusions of this study.

2 Theory

2.1 Basic control structure

WP are abstractions of real forms of workflow that are repeated in specific non-arbitrary contexts. The WP was proposed by van der Aalst, ter Hofstede, Kiepuszewski, and Barros, in 2003 [3]. The WP uses a Petri-net notation. WP are defined in 8 groups of patterns and broken down into 43 patterns. Then updated by Russell, ter Hofstede, van Der Aalst, Mulyar in 2006 [4]. The update is an improvement by

providing a systematic and formal description in the form of colored Petri-net. In 2004, Stephen A. White specified a WP in BPMN notation [18].

On the other hand, Shao and Wang in 2003 identified BCS as part of the WP based on cognitive patterns [5]. Each cognitive pattern has a weight. This weight is a measure of the complexity of the pattern. In 2006, Gruhn and Laue added cognitive patterns, namely multiple-choice, multiple instances, cancel activity, and cancel case [13]. In this paper, the BCS is defined as in Definition 1.

Definition 1 Basic Control Structure

Business process model (BPmodel) consists of Basic Control Structure (BCS), where

$$BCS = \{seq, AND, OR, XOR, cyc \in WP\}$$

BCS is defined as *seq* iff $t_0 \xrightarrow{next} t_k$, where $t \in Ac$

BCS is defined as *AND* iff t_0 has branch t_{01}, \dots, t_{0s} , where $t \in Ac$

BCS is defined as *OR* iff t_0 has branch t_{01}, \dots, t_{0s} , where $t \in Ac$

BCS is defined as *XOR* iff t_0 has branch t_{01}, \dots, t_{0s} , where $t \in Ac$

BCS is defined as *cyc* iff t_k has loop branch, where $t \in Ac$

2.2 Metrics for workflow patterns

The business process model metric consists of atomic and composite metrics. Atomic metrics are stand-alone metrics and are not formed by other metrics in a business process model, which in this paper is defined as in Definition 2. While composite metrics are metrics formed by other metrics in a business process model in this paper is defined as in Definition 3.

Definition 2 Atomic metrics

The atomic metrics used in this paper are the arcs size (*As*), the starts size (*Ss*), the ends size (*Es*), the intermediates size (*Is*), the activities size (*Acs*), the depth (*D*), the branches size (*Bs*), and diameters (*Dm*), that are part of BPmodel The atomic metric is defined as :

$$M_{Atomic} = \{As, Ss, Es, Is, Acs, D, Bs, Dm \in BPmodel\}$$

Definition 3 Composite metrics

The composite metrics used in this paper are the number of nodes (*Ns*), number of branching types (*Bt*), and the complexity of the loop (*C_{cyc}*) that are part of the *BPmodel*. Composite metrics are defined as :

$$M_{Composite} = \{Ns, Bt, C_{cyc} \in BPmodel\}$$

where

$$Ns = Ss + Es + Is + Acs + Bt \quad (1)$$

where

$$Bt = ANDs + XORs + ORs \quad (2)$$

where

ANDs : number of AND branching

XORs : number of XOR branching

ORs : number of OR branching

2.3 The complexity of business process models

Every BCS has a complexity value. Edmond defines complexity as the nature of language expression, which makes it difficult to formulate its overall behavior, even when given almost complete information about the atomic components and the relationships between them [19]. The complexity of the business process model is measured through metrics in diagrammatic representation or formal language. Measurement of business process models complexity can be approached based on software engineering theory, cognitive science, and graph theory. According to Fenton and James Bieman, the definition and validation of new metrics must go through three stages, namely the definition of metrics, theoretical validation of metrics, and empirical validation of metrics. The procedure for defining metrics goes through three stages, namely the identification of the measured entity, the identification of the attributes of the measured entity, and the definition of the metric [20]. The complexity of the business process model is measured by several aspects, namely activities, control flow, data flow, and resources [21].

The cognitive weight (CW) of software is the level of difficulty required to understand software that is modeled with several BCS [5]. The CW value for BCS used in this paper is presented in Table 1.

3 Methods

3.1 Identification of business process model complexity metrics

Business process model metrics are selected and identified using the Goal-Question-Metric (GQM) method. GQM is a framework that uses a top-down approach [22]. The stage of GQM consists of: (1) Specific goals stated; (2) Asking questions whose answers help achieve the goal; (3) Provide a measurement scheme.

The Goal measurement of the business process model complexity is defined as presented in Table 2.

Table 1. Cognitive weight on basic control flow

Workflow Pattern	Weight	Source
Sequence	1	[13]
Exclusive Choice (XOR)	3	
Parallel Split (AND)	4	
Multiple Choice (OR)	7	
Looping (Cyclic)	3	[5]
Depth	14	[15]

Table 2. Goal measurement

Analysis	:	Business process model
Issue	:	To obtain quantitative value for and complexity
Focus of measurement	of:	The elements and structures that exist in the business process model
Point of view	:	User, developer, owner
Context of measurement	of:	A business process model that is built from its workflow pattern

Table 3. Identification of the scale metrics and complexity of the business process model

Q.1	What parameters are used to measure the business process model complexity?
M.1.1	The business process model is represented by graphs using BPMN
M.1.2	The number of nodes
M.1.3	The number of arcs
M.1.4	The number of branches and types
M.1.5	The number of branches in each branch
M.1.6	The number of loops
M.1.7	The average depth of each node

3.2 Proposed formula

The Yaqin complexity formula is more comprehensive and more sensitive to small changes in the structure of the business process model. Our proposed formula accommodates aspects of activities (Acs), branch type (Bt), cyclic (Cyc), depth (D), and cognitive weight (CW). Ns is the sum of the event (Ss, Es, Is), activity (Acs), and branch type (Bt). The branch type aspect is represented by the modified control flow complexity, which consists of branch type AND, XOR, and OR. This modification was inspired by the determination of split-join logic in the process mining [23-26] as defined in Definition 4, Definition 5, Definition 6.

Definition 4 AND branch complexity

Let the business process model in Fig. 2 (b). Possible paths are ABDEC; ABEDC; ADEBC; ADBEC; AEBDC; AEDBC. Three branches produce six possible paths, so $6 = 3!$. Weights are given based on Table 1 $CW_{AND} = 4$, so

$$C_{AND} = 4 \cdot \sum_i (n!)_i \quad (3)$$

where n : number of branches in each branch and i : index of AND branches in the business process model.

Definition 5 XOR branch complexity

Let the business process model in Fig. 7 (a). Possible paths are ABC, ADC, AEC. Three branches produce three possible paths. Weights are given in Table 1 $CW_{XOR} = 3$, so

$$C_{XOR} = 3 \cdot \sum_i (n)_i \quad (4)$$

Definition 6 OR branch complexity

Let the business process model in Fig. 7 (b). Possible paths are ABC; ADC; AEC; ABDC; ADEC; ABEC; ABDEC; ABEDC; ADEBC; ADBEC; AEBDC; AEDBC. From 3 branches produces 12 possible paths, so $12 = \frac{3!}{(3-2)!}$. Weights are given based on Table 1 $CW_{OR} = 7$ so

$$C_{OR} = 7 \cdot \sum_i \left(\sum_{k=1}^{n-1} \frac{n!}{(n-k)!} \right)_i \quad (5)$$

where n : number of branches in each branch, k : the number of branches that may be passed at a branch, and i : index of OR branches in the business process model

The complexity of cyclic is also taken into account in the formulation of the complexity of business process models. The cyclic complexity is defined as in Definition 7.

Definition 7 Cyclic complexity

The business process model is said to be cyclically complete if the number of activities involved in a loop (Acs_{cyc}) equals as the diameter (Dm), so cyclical complexity is defined as the ratio of the number of activities involved in cyclic to its diameter. The intended diameter, as defined by Jan Mendling [7]. Weight given is based on Table 1 $CW_{cyc} = 3$ so

$$C_{cyc} = 3 \cdot \frac{Acs_{cyc}}{Dm} \quad (6)$$

The depth complexity is defined as the average depth of each activity in the business process model. The complexity of depth is defined in Definition 8.

Definition 8 Depth complexity

The complexity of depth is measured through the average depth of each activity formulated in Eq. (7).

$$D_{avg} = \frac{\sum_j^{Acs} D_j}{Acs} \quad (7)$$

where

j : number of activities

D : depth of each activities

Weight given is based on Table 1 $CW_D = 14$ so

$$C_D = 14 \cdot Davg \tag{8}$$

Finally, the Yaqin complexity metric of the business process model is defined as the sum of the number of vertices, the number of arcs, the sum of the AND, XOR, OR gateway complexities, the sum of cyclic complexities, and the depth complexity as defined in Definition 9.

Definition 9 Yaqin complexity metric

The Yaqin complexity metric of the business process model (YC) is defined based on Eq. (1), Eq. (2), Eq. (3), Eq. (4), Eq. (5), Eq. (6), and Eq. (8), so

$$YC = Ns + As + C_{AND} + C_{XOR} + C_{OR} + C_{cyc} + C_D \tag{9}$$

3.3 Experiment design

To test the Yaqin complexity formula, we use a dataset consisting of several business process models.

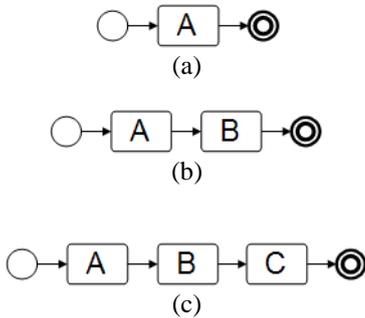


Figure. 1 Sequence business process models

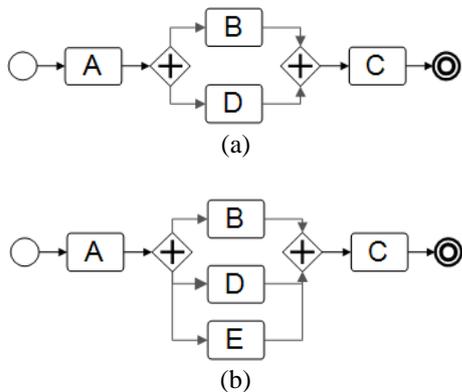


Figure. 2 Business process models with AND branches

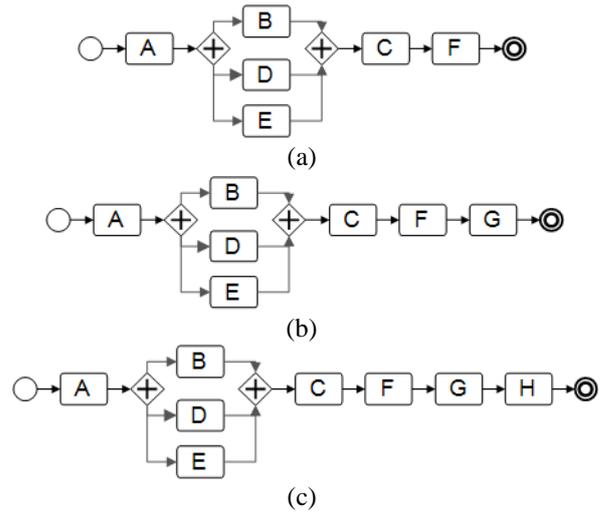


Figure. 3 Business process models with the addition of activities in sequence

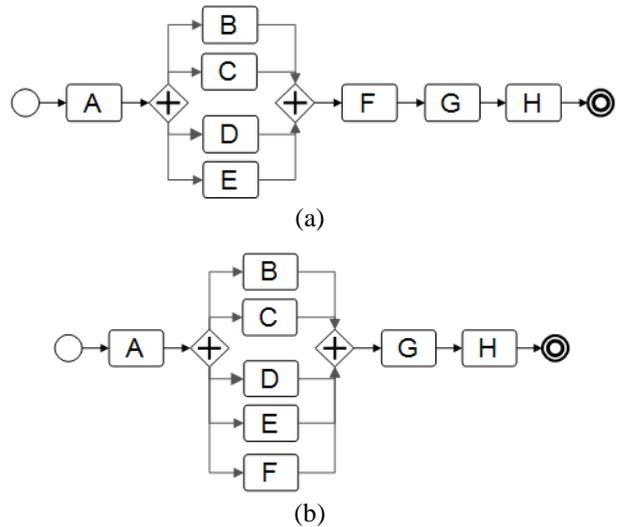


Figure. 4 Business process models with variations in the number of branches

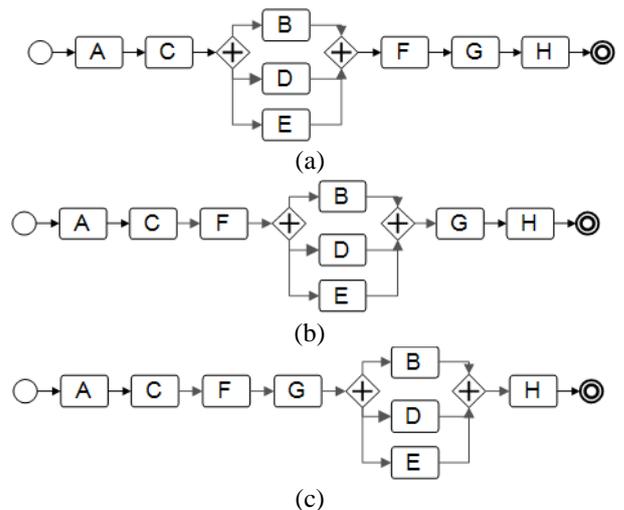


Figure. 5 Business process models with variations of branching position

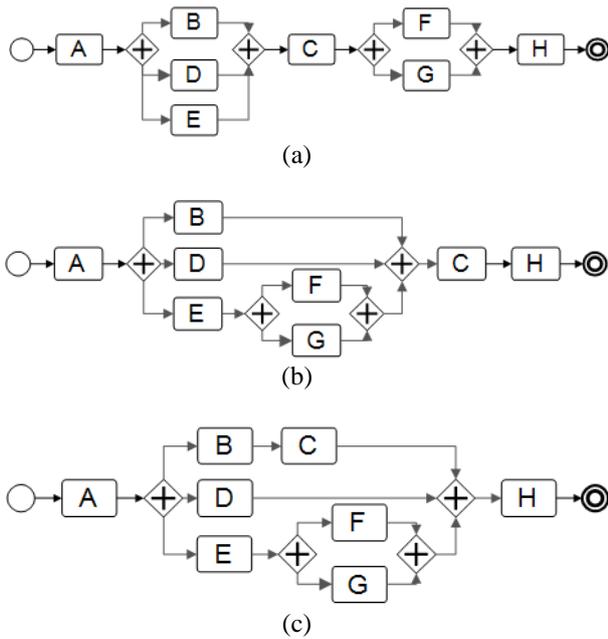


Figure. 6 Business process models with variations of branching depth

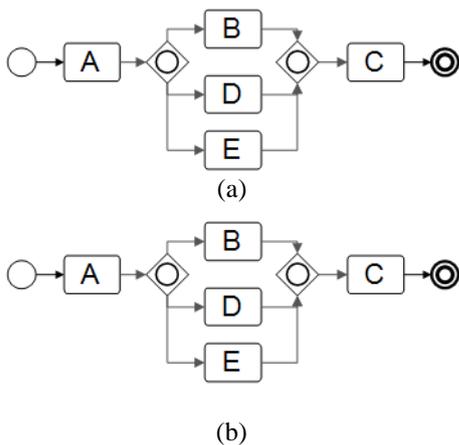


Figure. 7 Business process models with variations in branching types

The business process models presented are required to test the complexity metric measurements, as shown in Table 4.

Then the business process models complexity are calculated with Eq. (9) to be analyzed. the complexity of the business process model is also calculated by formula McCabe’s cyclomatic complexity [12], Jorge Cardoso’s control flow complexity [11], The cognitive complexity formula proposed by Shao and Wang [5] that uses the definitions shown in Table 1, the Coskun’s CADAC complexity formula [15], and the Scale formula [16]. The McCabe’s cyclomatic complexity is

$$CyclomaticComplexity = N_s + A_s \quad (10)$$

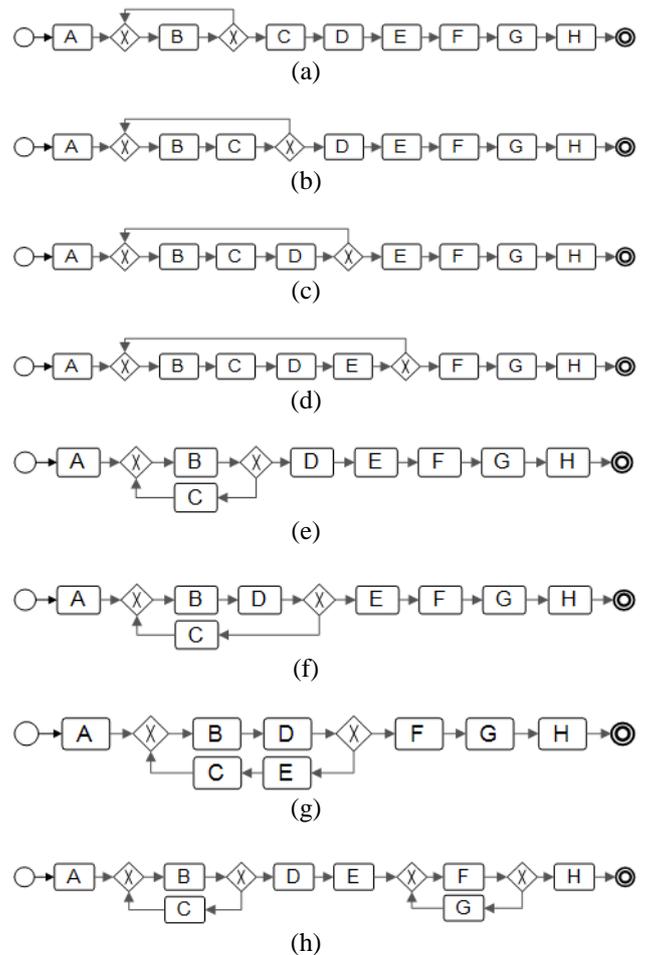


Figure. 8 Business process models with looping variations

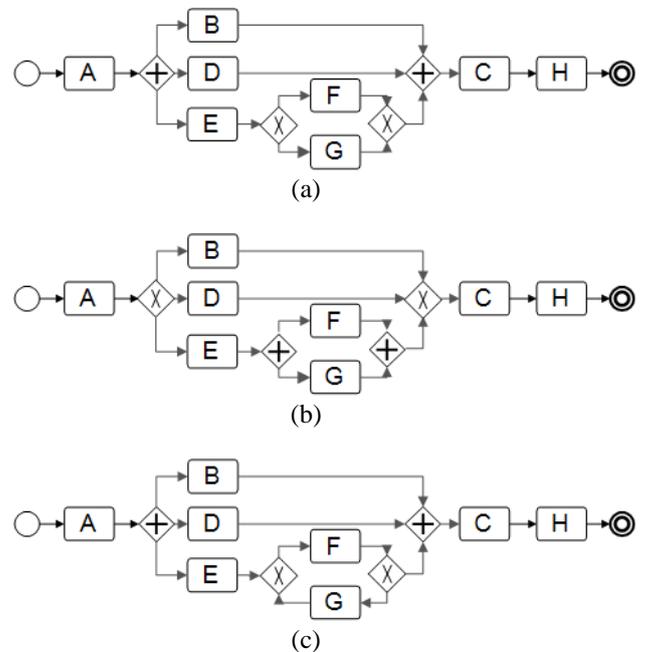


Figure. 9 Business process models with a combination of branching logic

Table 4. Purpose of business process models

Figure	Purpose
Fig. 1	Sequence business process models with different activities numbers. These business process models are intended to show the effect of adding sequential activity on the overall business process model complexity.
Fig. 2	Business process models have AND branches with a different number of branches. These business process models are intended to show the effect of increasing the number of branches on the business process model complexity
Fig. 3	Business process models have AND branches with a different number of sequence activities. These business process models are intended to show the effect of adding sequential activity on the business process model complexity
Fig. 4	Business process models that have AND branches with a different number of branches. These business process models are intended to show the effect of increasing the number of branches on the business process model complexity
Fig. 5	Business process models that have AND branches with different branch location variations. These business process models are intended to show the effect of varying branching locations on the business process complexity
Fig. 6	Business process models that have AND branches with different depth variations. These business process models are intended to show the effect of different depths on the business process complexity
Fig. 7	Business process models with variations in branching XOR and OR. These business process models are intended to show the effect of each gateway on the business process model complexity
Fig. 8	Business process models with looping variations. These business process models are intended to show the effect of various variations on the business process model complexity
Fig. 9	Business process models with a combination of logic and looping gates. These business process models are intended to show the effect of a combination of logic gates and looping on the business process model complexity

The Jorge Cardoso's control flow complexity is

$$CFC = \sum CFC_{XOR} + \sum CFC_{OR} + \sum CFC_{AND} \quad (11)$$

where

$$CFC_{XOR} = fanout$$

$$CFC_{OR} = 2^{fanout} - 1$$

$$CFC_{AND} = 1$$

The cognitive complexity formula is

$$CW = \{\sum_j \sum_i CW(j, i)\} \quad (12)$$

The CADAC complexity formula is

$$CADAC = (Acs \cdot 1) + (D_{max} \cdot 14) + (XORs \cdot 2) + (ANDs \cdot 4) + (ORs \cdot 7) + (Max(fanin \cdot fanout)^2 \cdot 4) + (As \cdot 1) \quad (13)$$

The Scale is

$$Scale = (Ns + As) \cdot CFC \quad (14)$$

The results of calculations with various formulas were analyzed by comparing them. Comparison results will prove that the Yaqin complexity formula is more comprehensive and more sensitive to small changes in the structure of the business process model.

3.4 Theoretical validation of business process models complexity metrics

After the complexity metric is defined, the next step is to validate the complexity metric. Theoretical validation applied to the complexity metric is Weyuker's properties [27]. Then the results of the validation are compared with the results of the validation of the Cyclomatic Complexity, CFC, CW, CADAC, and Scale formula.

4 Result and discussion

4.1 Comprehensiveness of the yaqin complexity formula

The comprehensiveness of complexity formula is defined as the number of parameters involved in the formulation: the more parameters involved, the more comprehensive the formula. The Yaqin complexity formula includes more parameters than other complexity formulas. The Yaqin complexity formula consists of seven parameters namely *Ns*, *As*, *ANDs*, *ORs*, *XORs*, *cyc*, and *D*. While CADAC consists of six parameters, Cyclomatic Complexity consists of two parameters, CFC consists of three parameters, CW consists of four parameters, and the Scale consists of five parameters. The comparison shows that the Yaqin complexity formula is more comprehensive than other formulas. The summary of the comprehensiveness is shown in Table 5.

4.2 Experiment results

Metrics consist of two groups, namely atomic metrics and composite metrics. Atomic metrics are stand-alone metrics and are not composed of other metrics. Composite metrics are metrics composed of

Table 5. Parameters of the complexity formula

Complexity formula	<i>Ns</i>	<i>Acs</i>	<i>As</i>	<i>ANDs</i>	<i>ORs</i>	<i>XORs</i>	<i>cyc</i>	<i>D</i>
<i>YC</i>	✓		✓	✓	✓	✓	✓	✓
<i>CADAC</i>		✓	✓	✓	✓	✓		✓
Cyclomatic Complexity	✓		✓					
<i>CFC</i>				✓	✓	✓		
<i>CW</i>				✓	✓	✓	✓	
<i>Scale</i>	✓		✓	✓	✓	✓		

several other metrics that form a formula.

This section presents the results of the business process model complexity metric calculations, which are the case studies shown in Fig. 1 to Fig. 9 As example we show the calculation of the Yaqin complexity using Fig. 6 (b).

step 1: calculate *Ns* using Eq. (1).

let

$$\begin{aligned} Ss &= 1 & Acs &= 8 \\ Es &= 1 & Bt &= 4 \\ Is &= 0 \end{aligned}$$

where $Bt = ANDs + ORs + XORs$ such as shown in Eq. (2).

let

$$\begin{aligned} ANDs &= 4 \\ ORs &= 0 \\ XORs &= 0 \end{aligned}$$

then

$$\begin{aligned} Ns &= 1 + 1 + 0 + 8 + 4 \\ Ns &= 14 \end{aligned}$$

step 2: calculate *As*

$$As = 16$$

step 3: calculate C_{AND} using Eq. (3), C_{XOR} using Eq. (4), C_{OR} using Eq. (5), and C_{cyc} using Eq. (6).

let

$$\begin{aligned} n_1 &= 3 & n_2 &= 2 \\ C_{AND} &= 4 \cdot (3! + 2!) \\ C_{AND} &= 32 \end{aligned}$$

while in Fig. 6 (b) XOR branch, OR branch, and loop are not found, so $C_{XOR} = 0$, $C_{OR} = 0$, $C_{cyc} = 0$.

step 4: calculate D_{Avg} using Eq. (7).

Table 6. Metric measurement results of business process models

Business Process Models	<i>Ns</i>	<i>ACS</i>	<i>As</i>	C_{AND}	C_{XOR}	C_{OR}	<i>Dm</i>	C_{Cyc}	D_{Avg}	D_{max}
Fig. 1 (a)	3	1	2	0	0	0	3	0	1.000	1
Fig. 1 (b)	4	2	3	0	0	0	4	0	1.000	1
Fig. 1 (c)	5	3	4	0	0	0	5	0	1.000	1
Fig. 2 (a)	8	4	8	8	0	0	6	0	1.500	2
Fig. 2 (b)	9	5	10	24	0	0	7	0	1.600	2
Fig. 3 (a)	10	6	11	24	0	0	8	0	1.500	2
Fig. 3 (b)	11	7	12	24	0	0	9	0	1.429	2
Fig. 3 (c)	12	8	13	24	0	0	10	0	1.375	2
Fig. 4 (a)	12	8	14	96	0	0	10	0	1.500	2
Fig. 4 (b)	12	8	15	480	0	0	10	0	1.625	2
Fig. 5 (a)	12	8	13	24	0	0	10	0	1.375	2
Fig. 5 (b)	12	8	13	24	0	0	10	0	1.375	2
Fig. 5 (c)	12	8	13	24	0	0	10	0	1.375	2
Fig. 6 (a)	14	8	16	32	0	0	10	0	1.625	2
Fig. 6 (b)	14	8	16	32	0	0	10	0	1.875	3
Fig. 6 (c)	14	8	16	32	0	0	10	0	2.000	3
Fig. 7 (a)	9	5	10	0	9	0	5	0	1.600	2
Fig. 7 (b)	9	5	10	0	0	42	7	0	1.600	2
Fig. 8 (a)	12	8	12	0	0	0	11	0.273	1.125	2
Fig. 8 (b)	12	8	12	0	0	0	12	0.500	1.250	2
Fig. 8 (c)	12	8	12	0	0	0	13	0.692	1.375	2
Fig. 8 (d)	12	8	12	0	0	0	14	0.857	1.500	2
Fig. 8 (e)	12	8	12	0	0	0	11	0.545	1.250	2
Fig. 8 (f)	12	8	12	0	0	0	12	0.750	1.375	2
Fig. 8 (g)	12	8	12	0	0	0	12	1.000	1.500	2
Fig. 8 (h)	14	8	15	0	0	0	12	1.000	1.500	2
Fig. 9 (a)	14	8	16	24	6	0	9	0	1.875	2
Fig. 9 (b)	14	8	16	8	9	0	8	0	1.875	2
Fig. 9 (c)	14	8	16	24	0	0	9	0.667	1.875	2

Table 7. Comparison of complexity metrics among business process models

Business Process Models	YC	CAD AC	Cyclomatic Complexity	CFC	CW	Scale
Fig. 1 (a)	19.000	21	1	1	1	5
Fig. 1 (b)	21.000	23	1	1	1	7
Fig. 1 (c)	23.000	25	1	1	1	9
Fig. 2 (a)	45.000	48	2	1	4	16
Fig. 2 (b)	65.400	51	3	1	4	19
Fig. 3 (a)	66.000	53	3	1	4	21
Fig. 3 (b)	67.000	55	3	1	4	23
Fig. 3 (c)	68.250	57	3	1	4	25
Fig. 4 (a)	143.000	58	4	1	4	26
Fig. 4 (b)	529.750	59	5	1	4	27
Fig. 5 (a)	68.250	57	3	1	4	25
Fig. 5 (b)	68.250	57	3	1	4	25
Fig. 5 (c)	68.250	57	3	1	4	25
Fig. 6 (a)	84.750	64	4	2	8	60
Fig. 6 (b)	88.250	78	4	2	8	60
Fig. 6 (c)	90.000	78	4	2	8	60
Fig. 7 (a)	50.400	49	3	3	3	57
Fig. 7 (b)	83.400	54	3	7	7	133
Fig. 8 (a)	40.023	52	2	2	3	48
Fig. 8 (b)	42.000	52	2	2	3	48
Fig. 8 (c)	43.942	52	2	2	3	48
Fig. 8 (d)	45.857	52	2	2	3	48
Fig. 8 (e)	42.045	52	2	2	3	48
Fig. 8 (f)	44.000	52	2	2	3	48
Fig. 8 (g)	46.000	52	2	2	3	48
Fig. 8 (h)	51.000	55	3	4	6	116
Fig. 9 (a)	86.250	62	4	4	7	120
Fig. 9 (b)	73.250	62	4	4	7	120
Fig. 9 (c)	80.917	60	4	4	7	120

let

$$\begin{aligned}
 D_A &= 1 & D_E &= 2 \\
 D_B &= 2 & D_F &= 1 \\
 D_C &= 2 & D_G &= 3 \\
 D_D &= 1 & D_H &= 3
 \end{aligned}$$

$$D_{Avg} = \frac{15}{8} = 1.875$$

so

$$C_D = 14 \cdot 1.875 = 26.25$$

step 5: calculate YC using Eq. (9).

based on result of previous step, then

$$YC = 14 + 16 + 32 + 0 + 0 + 0 + 26.25$$

$$YC = 88.25$$

The results of the business process model metric calculations are presented in Table 6. Then the metrics are used to calculate complexity metrics with CADAC as in Eq. (13), Cyclomatic Complexity as in Eq. (10), CFC as in Eq. (11), CW as in Eq. (12), and Scale as in Eq. (14). The results of calculating the complexity metrics are presented in Table 7.

4.3 Discussion of experimental results

In this section, we show the effect of several cases on the Yaqin complexity formula, namely.

Case 1 Adding elements to the business process model

The Yaqin complexity metric increases with the increasing number of elements in the business process model. Fig. 1 shows the increase in the Acs and As in the sequential business process model. Table 6 shows that Ns and As are different so that it affects the business process model complexity. The difference occurs because the Yaqin complexity formula involves Ns and As . Fig. 2 shows the addition of Acs and n in a business process model with AND branching. Table 6 shows that Ns , As , C_{AND} , and D_{Avg} are different. The addition of Acs and n affects D_{Avg} and C_{AND} , which also affects the complexity of the business process model. The difference occurs because the Yaqin complexity formula also involves C_{AND} and D . Fig. 3 shows the addition of Acs in sequence to a business process model that has AND branches. Table 6 shows that Ns , As , and D_{Avg} are different, but the C_{AND} is the same. The addition of Acs in sequence to a business process model that has AND branches affects the complexity of the business process model. The results of calculating the business process model complexity in Table 7 show that the Yaqin complexity metric is different in Fig.1, Fig. 2, and Fig. 3. This case proves that the addition of Acs affects Ns and As , and in some cases, also affects D_{Avg} and C_{AND} , so it affects the business process model complexity. Table 7 shows that CADAC and Scale are different, so it can be said that this case affects the CADAC and Scale. This case does not affect Cyclomatic Complexity as evidenced by Fig. 1 and Fig. 3, each having the same complexity metric and only in Fig. 2, which has a different complexity metric. This case also does not affect CFC, where Fig. 1, Fig. 2, and Fig. 3 have the same CFC. This case also does not affect CW, wherein Fig. 1, the CW is the same, and in Fig. 2, and Fig. 3, the CW is the same too. Thus it can be concluded that in this case, the Yaqin complexity metric is more sensitive than Cyclomatic Complexity, CFC, and CW, but equally sensitive to CADAC and Scale.

Case 2 Add branches to a branch

The Yaqin complexity metric increases with the number of branches in a branch. Fig. 3 (c) and Fig. 4 show the addition of branches in an AND branch with the same Ns . This difference affects the complexity of

the business process model because C_{AND} involves the number of branches in its formula in Eq. (3). In this case, the AND branch is used as an example. The XOR and OR branches also have the same effect according to the formulas in Eq. (4) and Eq. (5). The calculation results of the business process model complexity in Table 7 show that the Yaqin complexity metric in Fig. 3 (c) and Fig. 4 are different. This case proves that increasing n in a branch affects C_{AND} and D_{Avg} , thus also affecting the Yaqin complexity metric of the business process model. Table 6 also shows that CADAC, Scale, and Cyclomatic Complexity differ due to differences in As , so it can be said that this case affects the CADAC, Scale, and Cyclomatic Complexity. Adding branches to a branch will add As . In CADAC and Scale, branching logic will affect the complexity metrics, but the Cyclomatic Complexity does not affect. This case does not affect CFC and CW because they do not involve the number of branches in its complexity calculation. The results of the CFC and Cognitive Complexity calculations, in this case, are the same. Thus it can be concluded that the Yaqin complexity metrics more sensitive than CFC and CW, but it is equally sensitive to CADAC, Scale, and Cyclomatic Complexity.

Case 3 Shifting the location of the branches

The Yaqin complexity metric is not affected by shifting branching locations. In Fig. 3 (c) and Fig. 5, each shows an AND branching with three branches with different branching locations. Fig. 3 (c) and Fig. 5 do not show differences in Ns , As , C_{AND} , and D_{Avg} have shown in Table 6. The calculation of complexity in Table 7 also does not show differences. This case proves that the branching location shift does not affect the complexity of the business process model. In this case, Table 7 shows that CADAC, Cognitive Complexity, CFC, CW, and Scale are also not affected by shifting branching locations. Thus, the branching location shift does not affect the complexity of the business process model.

Case 4 Depth changing in the business process model

The Yaqin complexity metric changes when D changes in the business process model. Fig. 6 shows business process models with Acs equal to various depths. Table 6 shows that D_{max} for Fig. 6 (a) and Fig. 6 (b) is different, as shown in Fig. 6 (d) and Fig. 6 (e). While Fig. 6 (b), Fig. 6 (c), and Fig. 6 (d) D_{max} are the same. Fig. 6 (a) and Fig. 6 (b) show business

process models with 2 AND branches that differ in D . Table 7 shows that the results of the Yaqin complexity metric calculation in Fig. 6 (a) and Fig. 6 (b), are different even though Ns , As , and C_{AND} are the same. D_{max} causes different Yaqin complexity metrics. The Yaqin complexity metric in Fig. 6 (b) and Fig. 6 (c) also differs even though Ns , As , C_{AND} , and D_{max} are the same. D_{Avg} is causing different Yaqin complexity metrics. Table 7 shows that CADAC in Fig. 6 (a) and Fig. 6 (b) are different, whereas CADAC in Fig. 6 (b) and Fig. 6 (c) are the same. These cases show that CADAC only accommodates D_{max} changes. Cyclomatic Complexity, CFC, CW, and Scale do not provide differences in depth in the business process model. Thus, the Yaqin complexity metric is the most sensitive in accommodating depth to the business process model. CADAC is less sensitive in providing depth to business process models. Whereas Cyclomatic Complexity, CFC, CW, and Scale are not sensitive in accommodating depth to business process models.

Case 5 Changes in branching logic in the business process model

The Yaqin complexity metric changes when the branching logic changes in the business process model. This change in branching logic is shown in Fig. 2 (b) and Fig. 7. Fig. 2 (b) shows a business process model with two AND branching logic. In contrast, Fig. 7 (a) shows a business process model with XOR branching logic, and Fig. 7 (b) shows a business process model with OR branching logic. All three business process models have the same Ns , As , and D_{Avg} , but differ in terms of C_{AND} , C_{XOR} , and C_{OR} shown in Table 6. The Yaqin complexity metric shows the differences in all three, as shown in Table 7. Table 7 also shows that CADAC, CFC, CW, and Scale differ in the three business process models, while in Cyclomatic Complexity, the same. Cyclomatic complexity only pays attention to Ns and As without paying attention to the logic of branching. Thus it can be concluded that the Yaqin complexity metric is more sensitive than Cyclomatic Complexity and equally sensitive to CADAC, CFC, CW, and Scale.

Case 6 Number of nodes involved in a loop, and the location of the nodes in that loop

The Yaqin complexity metric is affected by Acs involved in a loop, and the location of activities in that loop. Fig. 8 represents this case with the same Ns , As and one loop with a variety of loop patterns, except for Fig. 8 (h), which contains two loops. Fig. 8 (b) and Fig. 8 (e) show a business process model

that involves two activities in one loop with different locations that produce different Yaqin complexity metrics. Likewise, Fig. 8 (c) and Fig. 8 (f) include three activities in a loop with different positions. Fig. 8 (d) and Fig. 8 (g) involve four activities in one loop with different locations. In Table 6, the business process models differ in their cyclical complexity involving diameters as defined in Definition 7. Table 7 also shows that CADAC, Cyclomatic Complexity, CFC, Cognitive Complexity, and Scale are all the same except in Fig. 8 (h) because CADAC, Cyclomatic Complexity, CFC, CW, and Scale do not accommodate loops. It can be concluded that the Yaqin complexity metric is more sensitive than CADAC, Cyclomatic Complexity, CFC, CW, and Scale.

Case 7 Number of the loop in the business process model

The Yaqin complexity metric is affected by the number of loops in the business process model. Fig. 8 (d) and Fig. 8 (g) show a business process model with one loop involving four activities, while Fig. 8 (h) shows a business process model with two loops involving four activities. Table 6 shows As differences in the business process models. As the number of loops increases, As also increases. The Yaqin complexity metrics for the business process models are shown in Table 7. Table 7 also shows that CADAC, Cyclomatic Complexity, CFC, CW, and Scale are all different because of the difference in As. Thus it can be concluded that the metric complexity of Yaqin is as sensitive as CADAC, Cyclomatic Complexity, CFC, CW, and Scale.

Case 8 Branching logic type exchange

The Yaqin complexity metric is affected by the exchange of branching types in the business process model. Fig. 6 (b) and Fig. 9 show business process models with the same structure with variations in the location of branching types. Fig. 6 (b) shows a business process model with two AND branches. Fig. 9 (a) and Fig. 9 (b) show business process models with AND and XOR branching variations. Fig. 9 (c) shows a business process model with a loop. The structure of the business process models is the same, but the branching logic is different. Table 6 shows that N_s , A_s , and D_{Avg} for Fig. 6 (b) and Fig. 9 are the same. C_{AND} and C_{XOR} for Fig. 6 (b) and Fig. 9 differ because the number of branches in each branch is different, so the Yaqin complexity metric shown in Table 7 is also different. Table 7 shows that CADAC is not affected by the branching type exchange described in Fig. 9 (a) and Fig. 9 (b) because CADAC does not involve the number of branches in its

Table 8. Comparison of the complexity metrics effect among cases

Case	YC	CADAC	Cyclomatic Complexity	CFC	CW	Scale
1	✓	✓	✗	✗	✗	✓
2	✓	✓	✓	✗	✗	✓
3	✗	✗	✗	✗	✗	✗
4	✓	✓	✗	✗	✗	✗
5	✓	✓	✗	✓	✓	✓
6	✓	✗	✗	✗	✗	✗
7	✓	✓	✓	✓	✓	✓
8	✓	✗	✗	✗	✗	✗

formula. Still, in Fig. 9 (c), CADAC is affected by loops in the business process model. Meanwhile, Cyclomatic Complexity is not affected at all in this case, because Cyclomatic Complexity only involves N_s and A_s . CFC, CW, and Scale in Fig. 6 (b), and Fig. 9 are different because the combination of branching types is also different. Thus it can be concluded that the order of sensitivity of the most sensitive is the metric complexity of Yaqin. The next order is CADAC, then CFC. CW and Scale together are almost insensitive. Cyclomatic complexity is not sensitive.

The analysis shows that the Yaqin complexity proved to be more sensitive to small changes in the business process model, because the different composition of elements can produce different complexity metric values. The third case produces the same complexity metric because the Yaqin complexity formula only accommodates changes in structure and does not accommodate changes in behavior in the business process model. The summary of the cases above can be seen in Table 8.

4.4 Theoretical validation

The Yaqin complexity metric is then validated with Weyuker's properties. Weyuker's properties have nine properties that are used to test the validity of the Yaqin complexity metric. The nine properties are:

Property 1: Non-coarseness : $(\exists P)(\exists Q); (|P| \neq |Q|)$. Complexity metrics can distinguish two different process models, so they produce different metrics. The Yaqin complexity metric can distinguish two different business process models; this can be seen in Table 7, where each business process model discussed has a different complexity metric. Thus property 1 has been fulfilled by Yaqin complexity metrics.

Property 2: Granularity: Complexity metrics must be non-negative. Table 7 shows Yaqin complexity

metric values, which are always positive, and if we look at the Eq. (7), the complexity metric values never to be negative. Thus property 2 has been fulfilled by Yaqin complexity metrics.

Property 3: Non-uniqueness : $(P \neq Q); (|P| = |Q|)$. In some cases, two different business process models can have the same complexity metrics. The Yaqin complexity also fulfills this property. This property is proven in Fig. 3 (c), Fig. 5 (a), (b), and (c), where they have different business process model structures but have the same complexity metric.

Property 4: Design Implication : $(\exists P)(\exists Q); (P \equiv Q \ \& \ |P| \neq |Q|)$. Complexity metrics must be able to distinguish between two business process models that have the same function but have different implementation details. The Yaqin complexity metric can distinguish business process models that have different structures but have the same functions as those in property 1. Thus the Yaqin complexity metric fulfills this property.

Property 5: Monotonicity : $(\forall P)(\forall Q); (|P| \leq |P; Q| \ \& \ |Q| \leq |P; Q|)$. Complexity metrics must be able to show the unification of several new elements in a business process model. The Yaqin complexity metric can indicate the unification of new elements in a business process model. The business process model in Fig. 1 (a) and Fig. 1 (b) connected sequentially to produce a complexity metric of 23. This figure has more significant complexity than the complexity of Fig. 1 (a) of 19 and Fig. 1 (b) of 21. Thus the Yaqin complexity metric fulfills property 5.

Property 6: Non-equivalence of Interaction:
 6a. $(\exists P)(\exists Q)(\exists R); (|P| = |Q| \ \& \ |P; R| \neq |Q; R| \ \& \ * \in \{-, \oplus, \circ\})$.
 6b. $(\exists P)(\exists Q)(\exists R); (|P| = |Q| \ \& \ |R; P| \neq |R; Q| \ \& \ * \in \{-, \oplus, \circ\})$.

Complexity metrics must be able to distinguish two identical processes. When there are three business process models P, Q, and R with the same metric value, then the business process model P is connected with R or Q is connected with R, then the two metric values should not be the same. The simple cases used for validation on this property are Fig. 5 (a) as the business process model P, Fig. 5 (b) as the business process model Q, and Fig. 1 (a) as the business process model R. When the business process model P is connected with the business process model R, and the business process model Q is connected with business process models R, both of which produce the same complexity metric. Thus the Yaqin complexity metric does not fulfill property 6.

Table 9. Weyuker's properties comparison

Properties	YC	CADAC	Cyclomatic Complexity	CFC	CW	Scale
1	✓	✓	✓	✓	✓	✓
2	✓	✓	✗	✗	✓	✓
3	✓	✓	✓	✓	✓	✓
4	✓	✓	✓	✓	✓	✓
5	✓	✓	✓	✓	✓	✓
6	✗	✗	✗	✗	✗	✗
7	✓	✗	✗	✓	✓	✗
8	✓	✓	✓	✓	✓	✓
9	✓	✓	✗	✓	✓	✓

Property 7: Significance of Permutation: $(\exists P)(\exists Q); (|P| \neq |Q|) \leftarrow$ If Q is formed by the sequence of permutations of activity P. Workflow patterns have been identified using a combination formula and permutation so that the logic difference in the branching and the number of branches in each branching significantly affect this complexity metric. The prove of this property can be seen in Fig. 6 (b) and Fig. 9 and the results of the metric calculations in Table 7. Thus the Yaqin complexity metric fulfills this property 7.

Property 8: No change in renaming. Changing names or labels on the business process model elements does not change complexity metrics. The Yaqin complexity metric bases the calculation on the structure of the business process model so that changes to the label do not affect the value of the complexity metric. Thus the Yaqin complexity metric fulfills property 8.

Property 9: Interaction Complexity : $(\exists P)(\exists Q); (|P| + |Q| < |P; Q|)$. The combined complexity metrics of several business process models must be greater than the sum of the complexity metrics of each. This property has already been discussed in property 5. Thus the Yaqin complexity metric fulfills property 9.

Based on this analysis, the Yaqin complexity metric meets 8 of 9 Weyuker's properties. As a comparison, we also present the Weyuker's properties evaluation of CADAC [15], Cyclomatic Complexity [28], CFC [29], CW [28], and Scale. The summary can be seen in Table 9.

5 Conclusions

Based on the previously elaborated discussion, we conclude that:
 1. The Yaqin complexity metric has advantages in detecting differences in the number of branches, differences in the structure of nested branches, and loops in the business process model.

2. The Yaqin complexity formula proved to be more comprehensive by involving seven parameters than the other complexity formulas that involve fewer parameters.
3. The Yaqin complexity metric proved to be more sensitive by affecting 7 of the 8 cases provided than the other complexity metric.
4. The Yaqin complexity metric has been validated with Weyuker's properties, which state that 8 out of 9 properties have been met.

In addition to some of the results above, this research also still needs some improvements, namely:

1. The Yaqin complexity formula is only tested to measure the complexity of structured business process models, so in the next research, it is necessary to examine the character of the Yaqin complexity formula in measuring the complexity of unstructured business process models.
2. Only split logic branching is measured by the Yaqin complexity formula, while the join logic is ignored. Therefore in the next research, the complexity of split and join logic branching must be considered.
3. The results of this research can also be used to measure the growth of enterprise architecture, explicitly determining the configuration of information systems based on the complexity of the business architecture.

Conflicts of Interest

The authors (Muhammad Ainul Yaqin, Riyanarto Sarno, and Siti Rochimah) declare no conflict of interest.

Author Contributions

Conceptualization, Muhammad Ainul Yaqin; methodology, Muhammad Ainul Yaqin; validation, Muhammad Ainul Yaqin; formal analysis, Muhammad Ainul Yaqin; investigation, Muhammad Ainul Yaqin; resources, Muhammad Ainul Yaqin; data curation, Muhammad Ainul Yaqin; writing—original draft preparation, Muhammad Ainul Yaqin; writing—review and editing, Riyanarto Sarno and Siti Rochimah; visualization, Muhammad Ainul Yaqin; supervision, Riyanarto Sarno and Siti Rochimah.

References

- [1] M. Dumas, M. La Rosa, J. Mendling, and H. A. Reijers, *Fundamentals of Business Process Management*, Springer, 2013.
- [2] Object Management Group (OMG). (2011) Object Management Group: Business Process

Model and Notation (BPMN) 2.0. [Online]. <https://www.omg.org/spec/BPMN/2.0/PDF>

- [3] W. M. P Van Der Aalst, A. H. M Ter Hofstede, B. Kiepuszewski, and A. P Barros, "Workflow Patterns", *Distributed and Parallel Databases*, Vol. 14, No. 1, pp. 5-51, 2003.
- [4] N. Russell, A. H. M Ter Hofstede, W. M. P Van Der Aalst, and N. Mulyar, "Workflow Control-flow Patterns: A Revised View", *BPM Center Report BPM-06-22*, *BPMcenter.org*, pp. 6-22, 2006.
- [5] J. Shao and Y. Wang, "A New Measure of Software Complexity Based on Cognitive Weights", In: *Proc. of Canadian Conf. on Electrical and Computer Engineering, 2003. IEEE CCECE 2003*, Vol. 2, pp. 133-1338, 2003.
- [6] K. B. Lassen and W. M. P. van der Aalst, "Complexity Metrics for Workflow Nets", *Information and Software Technology*, Vol. 51, No. 3, pp. 610-626, 2009.
- [7] J. Mendling, *Metrics for Process Models: Empirical Foundations of Verification, Error Prediction, and Guidelines for Correctness*, 6th ed, Springer Science & Business Media, 2008.
- [8] G. M. Muketha, A. A. A Ghani, M. H. Selamat, and R. Atan, "A Survey of Business Process Complexity Metrics", *Information Technology Journal*, Vol. 9, No. 7, pp. 1336-1344, 2010.
- [9] Gregor Polancic and Blaz Cegnar, "Complexity Metrics for Process Models-A Systematic Literature Review", *Computer Standards & Interfaces*, Vol. 51, pp. 104-117, 2017.
- [10] C. Corti, "BPMETRICS: A Software System for The Evaluation of Some Metrics for Business Process", Politecnico di Milano, Master Graduation Thesis, 2012.
- [11] J. Cardoso, "Business Process Control-flow Complexity: Metric, Evaluation, and Validation", *International Journal of Web Services Research (IJWSR)*, Vol. 5, No. 2, pp. 49-76, 2008
- [12] T. J. McCabe, "A Complexity Measure", *IEEE Transactions on Software Engineering*, Vol. 4, pp. 308-320, 1976.
- [13] V. Gruhn and R. Laue, "Adopting The Cognitive Complexity Measure for Business Process Models", In: *Proc. of 5th IEEE International Conf. on Cognitive Informatics, 2006. ICCI 2006*, pp. 236-241, 2006.
- [14] I. Vanderfeesten, J. Cardoso, J. Mendling, H. A. Reijers, and W. M. P Van Der Aalst, "Quality Metrics for Business Process Models", *BPM and Workflow Handbook*, Vol. 144, pp. 179-190, 2007.

- [15] E. Çoşkun, “A New Complexity Metric for Business Process Models”, Atilim University, Master Thesis, 2014.
- [16] M. A. Yaşın, R. Sarno, and A. C. Fauzan, “Scalability Measurement of Business Process Model using Business Processes Similarity and Complexity”, In: *Proc. of 4th International Conf. on Electrical Engineering, Computer Science and Informatics (EECSI)*, pp. 1-7, 2007.
- [17] Sholiq, R. Sarno, A. Tjahyanto, and A. D. Wulandari, “Workflow Complexity in Constructive Cost Model II”, In: *Proc. of 3rd International Conf. on Informatics and Computational Sciences, ICICOS 2019*, 2019.
- [18] S. A White, “Process Modeling Notations and Workflow Patterns”, *Workflow Handbook*, pp. 265-294, 2004.
- [19] B. Edmonds, “What is Complexity?-The Philosophy of Complexity per se with Application to Some Examples in Evolution”, in *The Evolution of Complexity*, Kluwer, Dordrecht, 1995.
- [20] N. Fenton and J. Bieman, *Software Metrics: a Rigorous and Practical Approach*, CRC press, 2014.
- [21] J. Cardoso, J. Mendling, G. Neumann, and H. A. Reijers, “A Discourse on Complexity of Process Models”, In: *Proc. of International Conf. on Business Process Management*, pp. 117-128, 2006.
- [22] H. Koziolok, “Goal, Question, Metric,” in *Dependability Metrics*, Springer, pp. 39-42, 2008.
- [23] W. M. P Van Der Aalst, *Process Mining: Discovery, Conformance and Enhancement of Business Processes*, 2nd ed, Springer, 2011.
- [24] J. Cardoso, “Business Process Quality Metrics: Log-based Complexity of Workflow Patterns”, In: *Proc. of OTM Confederated International Conf. on the Move to Meaningful Internet Systems* Springer, pp. 427-434, 2007.
- [25] R. Sarno, F. Haryadita, D. Sunaryono, and A. Munif, “Model Discovery of Parallel Business Processes using Modified Heuristic Miner”, In: *Proc. of 2015 International Conf. on Science in Information Technology (ICSITech)*, pp. 30-35, 2015.
- [26] R. Sarno, P. L. I. Sari, H. Ginardi, D. Sunaryono, and I. Mukhlash, “Decision Mining for Multi Choice Workflow Patterns”, In: *Proc. of 2013 International Conf. on Computer, Control, Informatics and Its Applications (IC3INA)*, pp. 337-342, 2013.
- [27] E. J Weyuker, “Evaluating Software Complexity Measures”, *IEEE Transactions on Software Engineering*, Vol. 14, No. 9, pp. 1357-1365, 1988.
- [28] S. Misra and A. K. Misra, “Evaluating Cognitive Complexity Measure with Weyuker Properties”, In: *Proc. of Proceedings of the Third IEEE International Conf. on Cognitive Informatics, 2004*, Vol. 3, pp. 103-108, 2004.
- [29] J. Cardoso, “Control-flow Complexity Measurement of Processes and Weyuker’s Properties,” In: *Proc. of 6th International Enformatika Conf.* Vol. 8, pp. 213-218, 2005.