



## A New Similarity Method based on Weighted Graph Models for Matching Parallel Business Process Models

Yohanes Setiawan<sup>1</sup>

Kelly Rossa Sungkono<sup>1</sup>

Riyanarto Sarno<sup>1\*</sup>

<sup>1</sup>*Department of Informatics Engineering, Institut Teknologi Sepuluh Nopember, Indonesia*

\* Corresponding author's Email: [riyanarto@if.its.ac.id](mailto:riyanarto@if.its.ac.id)

---

**Abstract:** In business process, similarity is important for comparing between business process models. The existing similarity methods, such as Graph-based Matching Method (GMA), Weighted Graph Edit Distance (WGED), Weighted Node Adjacent Relation Similarity (WNARS), Tree Declarative Pattern Edit Distance (TPED) and Cosine-Tree Declarative Pattern (Cosine-TDP) can distinguish between AND, OR, and XOR relationships. However, they have drawbacks in detecting same relationships with different event logs. This paper proposes a new similarity method based on weighted graph models called weighted graph-based parallel process model matching (WGPPM) for computing the behaviour of parallel activity relationships. The proposed method utilizes the frequency of activity relationships as the weight in the graph model to measure the similarity between processes containing parallel relationships. WGPPM is compared with GMA, WGED, WNARS, TPED, and Cosine-TDP. The result shows that WGPPM is able to compute similarity between same parallel relationship with different event logs.

**Keywords:** Business process modelling, Graph approach, Graph matching, Graph model, Similarity measurement.

---

### 1. Introduction

Nowadays, the application of business process modeling is very widely done and needed for many aspects to provide some business services [1]. The purpose of business process analysis is to improve the process in fulfilling some aspects, i.e. optimal, reusable, flexible, and incorporate with information technology [2], include the use of event log of a workflow process to be mined [3]. A business process model or process model is a conceptual model which depends on the business procedure in a company [4] that can be applied for many different applications, such as information related to processing in an organization and the requirements of the system [5]. Fauzan [6] states that business process models can also be developed for several other purposes, such as increasing profit by applying a model based on standard operational procedures. However, there are some issues in building business process models, for example difficulties related to parallel relationships. The methods proposed by [7–9], which were used in

this study, can successfully form parallel activity relationships, i.e. XOR, OR, and AND.

A business process model can be compared with another process model to obtain their similarity, which has important applications in business process management [10]. Zhu [11] proposed Structural Approach Matching for Business Process Models Search. This method builds a business process model that captures the information arrangement of the business process, after which the similarity value can be derived by process matching, based on edit distance of process models on both of functional and structural similarity, while structure weights are specified to contribute to the conclusion of similarity value. The matching method of [12] identifies the correspondence between the activities in two business process models to calculate the similarity. They proposed the optimization method utilizing Markov logic for choosing the best linkages among results of some similarity matching methods. [13] presented a business process model using web services that covers their functional properties, while a semantic matching algorithm determines executed

web services for choice processes. The Grammar-based Index for Matching Business Process Models was developed by [14] for indexing. They introduced a novel process evaluation process for matching business process models. In [15] used a clustering method to calculate the structural and the behavioral similarity between enterprise resource planning (ERP) models.

However, most of researchers are not paying attention with the parallel activity relationships in the similarity measure. Then, [7] proposes the Graph-based Matching Approach (GMA) for calculating similarity and matching Parallel Business Process Models. A graph-based parallel business process model contains nodes and edges that can be matched to calculate the similarity between graph-based parallel business process models. However, Graph-based Matching Approach shows that AND-XOR relations are similar than AND-OR relations. Consequently, GMA cannot distinguish same relationship but from different event log. After that, [16] proposed Weighted Graph Edit Distance (WGED) and Weighted Node Adjacent Relation Similarity (WNARS) as behavioral similarity of process model in an event log. However, WGED cannot distinguish between AND, OR, and XOR since WGED cannot detect graph with same nodes but different relationships. Also, WNARS depends on minimum and maximum value of the weight in the graph which is cannot represent the behavior of the relationship. Furthermore, [17] proposes Tree Declarative Pattern Edit Distance (TPED), Cosine-Tree Declarative Pattern (Cosine-TDP) for identifying AND and OR with AND and XOR. This method uses Linear Temporal Logic (LTL) which has many symbols to use. Even though the method can detect AND, OR and XOR, the range difference is too small such that AND-XOR is similar with AND-OR.

This research is motivated by similarity in behaviour between the parallel activity relationships, i.e. XOR, OR, and AND. A weighted graph model is a graph with a weight in the form of a numerical value as a property of its edges, which influences the matching of parallel business process models having XOR, OR, and AND. The matching algorithm modifies algorithms forming by [18]. Summarizing, the major contributions of this research are:

- 1) Weighted Graph Model as a novel parallel business process model, which reflects the behaviour of parallel relationships, i.e. XOR, OR and AND, by using the frequency of activity relations as a weight in the form of a numeric value. The weight added as a property of the edges of the graph-based process model

influences the measurement of the similarity between two process models.

- 2) Weighted Graph-based Parallel Process Model Matching (WGPPM) as a new matching algorithm which modifies from [18] to calculate the similarity of the parallel relationships between weighted graph models. The algorithm was applied to prove the truth of the parallel relationship behaviour. The result was compared with that of an existing matching algorithm, i.e. Graph-based Matching [7], Weighted Graph Edit Distance (WGED) [16], Weighted Node Adjacent Relation Similarity (WNARS) [16], Tree Declarative Pattern Edit Distance (TPED) [17], Cosine-Tree Declarative Pattern (Cosine-TDP) [17] to prove this statement.

## 2. Preliminaries

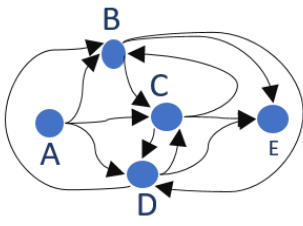
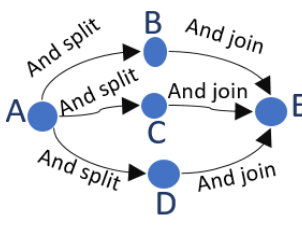
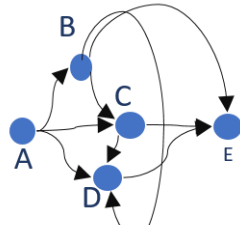
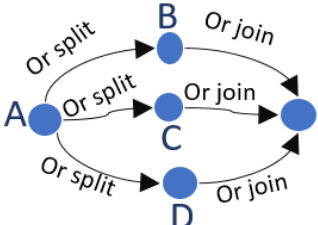
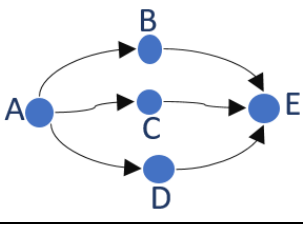
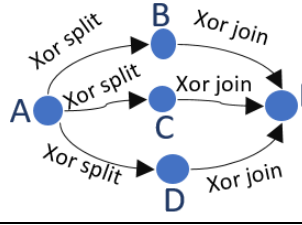
### 2.1 Parallel business process model

Business process models according to event logs help business process analysts to describe and evaluate business processes [19], and also to conduct performance analyses [20]. In a business process model, each activity is related to other activities. The relationship between activities consist of parallel and sequential, then the business process models which contains the parallel activity relationships called parallel business process models [8].

There are kinds of gateways representing parallel activity relationships, i.e. XOR, OR and AND. From [19], if only one out of multiple activities in a process model is selected for execution, then the parallel relationship is XOR, an OR relationship occurs when more than one tasks are determined in a process, and an AND relationship occurs if the business process model executes all activities with different order in each process. Because OR relationships have high tractability in the execution of activities, many process discovery algorithms have difficulty in defining this relationship. OR relationships are associated with an XOR relationship or an AND relationship even though both XOR relationships and AND relationships do not resemble the function of OR relationships [19].

Table 1 gives an example of the behavior of parallel gateway relationships XOR, OR, and AND from graph model and control flow-pattern. There are five activities, i.e. A, B, C, D, and E. Utilizing these activities, The OR relationship has some AND relationships that an XOR relationship does not have, e.g. BC. Of course, the OR relationship has all the relationships of the XOR relationship. Thus, it can be said that the OR relationship is more similar to the

Table 1. Types of parallel relationships

Relation	Graph Model	Control-Flow Pattern
AND		
OR		
XOR		

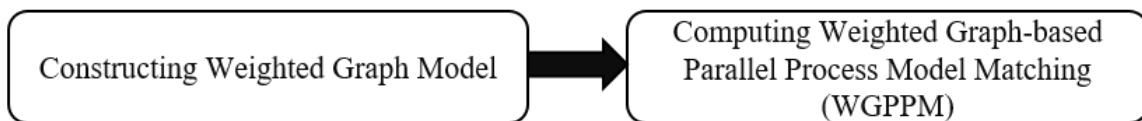


Figure. 1 The flow of the proposed method

AND relationship. Interestingly, [7] proved that the XOR relationship is more similar to the AND relationship according to their behaviour. This research attempted to prove this similarity based on the behaviour of relationships.

### 2.2 Methods of matching business process models

There are several techniques to calculate the matching for similarity between business process models based on graph-based matching have been introduced in the literature. Graph-based matching assumes that a business process model is a labeled graph, where the nodes correspond to tasks, events or gateways that are related to the edges, which capture the control-flow between them. It exploits the syntactic and or semantic meaning of the labels of the nodes to determine a mapping between the nodes and compares the structure of the graph [21]. The comparison of two business process models for recognizing linked activities that have alike parts is referred to [22] as business process model matching.

There is software available that can automate this process, which is called matching.

Many different equations are used in business process model matching, as mentioned in [18,23,24]. This research developed a new equation for matching two process models. The matching process obtains a tier that consists of linked activities. Linkages are bidirectional as a result of the matching process, which is individualistic of the model sequences [23].

When calculating the similarity or dissimilarity between two business process models it is reasonable that the distance  $dis(a, b)$  should be positive or zero, which is shown in Property 1. The distance between nodes  $a$  and  $b$  must be the same as the distance between nodes  $b$  and  $a$ . which is shown in Property 2. Property 3 is required because the distance between two business process models should only be zero if and only if the models are really identical.

**Property 1**  $dis(a, b) \geq 0$

**Property 2**  $dis(a, b) = dis(b, a)$

**Property 3**  $dis(a, b) = 0 \Leftrightarrow a \equiv b$

### 2.3 Weighted tree declarative pattern matching approach

The Weighted Tree Declarative Pattern Matching Approach (WTDPMA) is proposed with two methods of similarity, such as TPED and Cosine-TDP [17]. All of the similarity method is using Weighted-Linear Temporal Logic (W-LTL) as the business process model. The TPED is identifying structural similarity, while the Cosine-TDP is identifying behavioral similarity. WTDPMA proves that AND and OR is more similar than AND and XOR.

Basic methodology for achieving similarity in WTDPMA is following these steps:

1. Building the W-LTL Declarative Pattern which is each relation is identified by a weight. The weight represents the occurrences of activities.
2. Creating a Weighted-Tree Declarative Pattern (W-TDP) Model. The W-TDP model is generated by all discovered patterns in W-LTL.
3. Computing the similarity between AND and OR, AND and XOR, and XOR and OR using TPED and Cosine-TDP based on the model that has been created.

For more specifics steps of the method, readers should read the journal reference [17].

### 2.4 Graph-based matching approach

**Definition 1.** A graph-based process model (*GPM*) is a 3-tuple.  $GPM = (Nd, Eg, \theta)$  where *Nd* is a set of node *S*, *Eg* is a set of edges and  $\theta : Nd$  is a function of relation between *Nd* and *Eg*.

Transformation of a business process model into a graph is defined as a graph-based process model, which consists of nodes and edges, where the nodes define the activity and the edges define their relationships, as stated in Definition 1. In graph-based approaches, processes are treated as directed graphs and conventional graph or pattern matching algorithms are applied.

A graph-based process model consists of labeled nodes of different types (e.g. activities, gateways, events) and directed edges connecting them [12]. When two graph models have been constructed, pattern matching is done by the Graph-based Matching Approach proposed by [6] using a brute force algorithm and Phonetic Text Procedures in Neo4j. This Graph-based Matching Approach (GMA) is using Linear Temporal Logic (LTL), which is a pattern with the combination of atomic propositions, temporal capital operators, and logical operators. GMA calculates the similarity between two parallel

process models having XOR, OR, and AND relations. The general methodology of the GMA are following these steps:

1. Read the data in.csv format from LTL pattern
2. Convert the data to strings for inputting, using commas to divide them into three sub-strings. The obtained sub-strings are 'node from' strings, 'relation' strings, and 'node to' strings.
3. Repeat for  $n \times m$  times, where *n* is the amount number of relations in the event log and *m* is the amount number of relations in the LTL pattern). Each iteration compares the 'node from', 'relation', and 'node to' strings between both process models.
4. Choose the highest matching percentage for a better matching result.

## 3. Proposed method

The flow of the proposed method can be seen in Fig. 1, where the weighted graph-based process model represents the graph-based process model that was used in this research. In the first step, a graph-based parallel model is constructed from a given event log. Then, the weight as a property of the edges is calculated. After that, the similarity of the model is calculated using the above equation. The proposed weighted graph and method for calculating the similarity between two parallel business process models will be described in this section.

### 3.1 Weighted graph model

**Definition 2.** A weighted graph model *WGPM* is a 4-tuple.  $WGPM = (Nd, Eg, w, \theta)$  where *Nd* is a set of nodes *S*, *Eg* is a set of edges,  $w \in \mathbb{R}$  is weight, and  $\theta : Nd$  is a function of the relation between *Nd* and *Eg*.

---

Algorithm 1: Algorithm of Weighted Graph Model

---

**Input: Activities in Event log**

*iteration* = 3 (AND, OR, XOR)

For *iteration* = 1 : 3

1. Construct the Graph Model as shown in Table 1
2. Compute the weight for every relation based on Eq. (2)
3. Create Control-Flow Pattern as shown in Table 1 by deleting relations (**only** for AND and XOR)

End For

**Output: Weighted Graph Model AND, OR, XOR**

---

**Definition 3.** The relation between nodes  $A$  to  $B$ , denoted  $Relation_{A \rightarrow B}$ , is a set of connected edges  $Eg$  where  $Eg, A, B \neq \emptyset$

**Definition 4.** The number of relations of node  $A$  to node  $B$ , denoted  $count(Relation_{A \rightarrow B})$ , is the total case of relations of  $A$  to  $B$  where  $A, B \neq \emptyset$

**Definition 5.** Let  $w_{A \rightarrow B} \in \mathbb{R}$  be the weight of  $Relation_{A \rightarrow B}$ . The real weight value is computed by Eq. (1):

$$w_{A \rightarrow B} = \frac{count(Relation_{A \rightarrow B})}{count(Relation_A)} \quad (1)$$

A weighted graph model is a graph in which the edges have a weight for its in the form of a numerical value. Specifically, the weighted graph model represents the behaviour of a parallel relation-ship by calculating the frequency of relationships in the event log as stated in Definition 2. Definition 3 defines the relation between nodes in a parallel process model, while Definition 4 gives  $count(Relation_{A \rightarrow B})$  as the total frequency of relationships between nodes  $A$  to  $B$  and  $count(Relation_A)$  as the total frequency of the relationships between node  $A$  and  $B$  in the event log. Then, the weight in the weighted graph model is calculated that will help to calculate the similarity between parallel business process models as stated in Definition 5, which is provided by  $count(Relation_A)$  which computes all outgoing relations of  $A$  in the graph-based model before the process model is built.

The parallel process model used in this paper was built as a weighted graph by adding a weight to the edges of the graph. The process model contains XOR, OR, and AND relationships. Examples of XOR, OR, and AND relations from an event log were used to build the weighted graph in the parallel process model. Weighted graph model is built is shown in Algorithm 1.

### 3.2 Weighted graph-based parallel process model matching

A new method is proposed here for calculating the similarity between two weighted graph models, which differs from other similarity algorithms. The similarity equation for matching between parallel process model  $a$  and  $b$  is using the similarity measure  $sim(a, b)$  which is shown in Eq. (2). The distance  $dis(a, b)$  is formulated as absolute difference between weight of both parallel relationship in Eq. (3).

$$dis(a, b) = |(w_{A \rightarrow B})_a - (w_{A \rightarrow B})_b| \quad (2)$$

---

#### Algorithm 2: WGPPM (Similarity)

---

**Input: Weighted Graph Model  $a$  and  $b$**

Iteration:

For  $iteration = 1$  : number of relations

1. Compute the distance with Eq. (2)
2. Compute the similarity with Eq. (3)

End For

After iteration:

1. Average the similarity from number of relations
2. Compute the range transformation with Eq. (4)

**Output: Similarity between  $a$  and  $b$**

---

$$sim(a, b) = \frac{1}{1 + dis(a, b)} \quad (3)$$

The distance between two process models is in range between zero and one. When the distance is calculated as zero, both parallel process models are exactly the same ( $sim(a, b) = 1$ ). Otherwise, both of business process model is totally different when the distance is calculated as one ( $sim(a, b) = 0.5$ ). Therefore, we have  $sim(a, b) \in [0.5, 1]$ . Without loss of meaning,  $sim(a, b) \in [0.5, 1]$  can be transformed into  $sim(a, b) \in [0, 1]$  for well-reading of the degree of similarity, with  $sim(a, b) = 1$  and  $sim(a, b) = 0$  means similar and not similar, respectively. The change of range similarity is using range transformation which is computed with Eq. (4).

$$sim(a, b)_{transformed} = 2(sim(a, b) - 0.5) \quad (4)$$

The detailed steps of the similarity measure for WGPPM are shown in Algorithm 2.

## 4. Results and analysis

In the experiment an event log was taken to build a graph-based parallel business process model, after which the weighted graph was computed. This section describes the event log used and the result of the matching problem.

Table 2 shows the event log used which is taken from General Ledger Department. In real-life event log retail model, General Ledger Department is analyzing material for producing game. Then, we have general ledger parts which are containing 6 traces and 200 cases AND relationship, 3 traces and 100 cases OR relationship, and 3 traces and 100 cases XOR relationship. Table 3 shows the traces and cases in the event log.

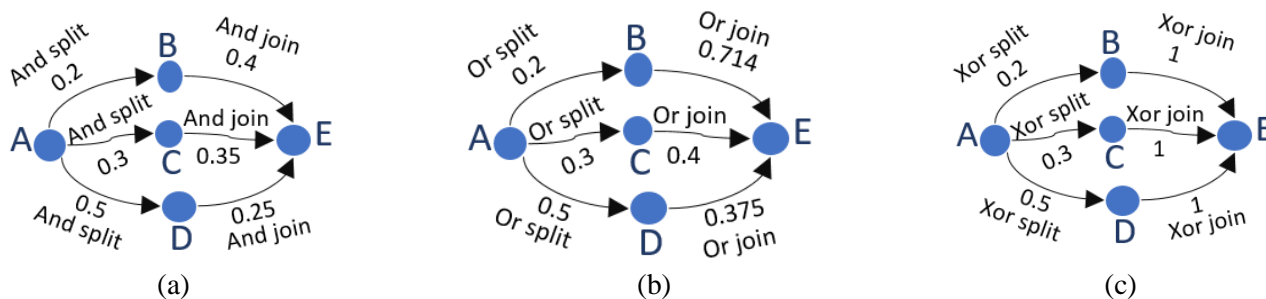


Figure. 2 Weighted graph model: (a) AND, (b) OR, and (c) XOR

Table 2. Event log

Activities	Aliases
Delivering items automatically	K
Calculating EOQ	L
Calculating supplier selection	M
Calculating ROP	N
Calculating optimal price	O

Table 3. Trace and case

(a) AND relationship		
No	Trace	Case
1	K-L-M-N-O	20
2	K-M-L-N-O	30
3	K-O-L-M-N	50
4	K-L-N-M-O	20
5	K-M-N-L-O	30
6	K-O-M-L-N	50
(b) OR relationship		
1	K-L-M-O	20
2	K-M-N-O	30
3	K-N-L-O	50
(c) XOR relationship		
1	K-L-O	20
2	K-M-O	30
3	K-N-O	50

Table 4. Trace and case from different event log

(a) AND relationship		
No	Trace	Case
1	K-L-M-N-O	10
2	K-M-L-N-O	40
3	K-O-L-M-N	60
4	K-L-N-M-O	10
5	K-M-N-L-O	30
6	K-O-M-L-N	50
(b) OR relationship		
1	K-L-M-O	10
2	K-M-N-O	40
3	K-N-L-O	50
(c) XOR relationship		
1	K-L-O	20
2	K-M-O	40
3	K-N-O	40

An example of calculating the weights for a weighted graph can be given based on the AND relationship in Table 3. When we want to calculate the weight of the between node K and L, we must compute  $count(Relation_{K \rightarrow L})$  as defined in Definition 4. In Table 3, we have K to L relations in K-L-M-N-O with  $case = 20$  and K-L-N-M-O with  $case = 20$ , such that:

$$count(Relation_{K \rightarrow L}) = 20 + 20 = 40$$

Then, we count the relations of K, which is the total frequency of the AND relationship:

$$count(Relation_K) = 200$$

After that, we have:

$$w_{K \rightarrow L} = \frac{40}{200} = 0.2$$

as the weight of the edge that represents the relationship between node K and L. The completed model AND relationship, OR relationship, and XOR relationship are shown in Fig. 2.

Once the Weighted Graph Model is built, the model can be used to calculate the matching between two parallel business process models that follow Algorithm 2 using the proposed similarity method. The general steps of calculating similarity between two WGPPM are given as example as follows:

1. Determining the parallel business processes that are to be matched, e.g. AND -XOR.
2. Calculating the distance between relations by subtraction of their weight, e.g. for the AND and XOR relations between nodes K and L, the distance is computed as:

$$dis = |0.2 - 0.2| = 0$$

3. From the distance, the similarity between nodes and edges is calculated. From the distance, the similarity is obtained as:

$$sim((AND)_{K-L}, (XOR)_{K-L}) = \frac{1}{1 + 0} = 1$$

4. Step (2)-(3) will be iterated with all other relations, such as node A to node C, node A to node D, node B to node E, and so on. After the similarities between all relations have been calculated, the average of the similarity between the relations is calculated

to achieve the final similarity between the parallel business processes. In the example of AND and XOR relations, the average similarity is achieved as:

$$\text{sim}(\text{AND}, \text{XOR}) = 0.8004$$

5. Then, the final similarity is achieved through range transformation:

$$\begin{aligned} \text{sim}(\text{AND}, \text{XOR})_{\text{transformed}} &= 0.6008 \\ &\approx 0.6 \end{aligned}$$

We also compute AND-AND, OR-OR, XOR-XOR between this event log and another different event log in General Ledger Department with different spread of cases as shown in Table 4, such that results in different Weighted Graph Model. As comparing, we include result from other existing similarity methods for parallel relationship AND, OR, and XOR. Result of similarity from the existing methods and proposed method can be fully seen in Table 5. We compare the WGPPM as our proposed method with other existing methods, such as Graph-based Matching Method (GMA) [7], Weighted Graph Edit Distance (WGED) [16], Weighted Node Adjacent Relation Similarity (WNARS) [16], Tree Declarative Pattern Edit Distance (TPED) [17], and Cosine-Tree Declarative Pattern (Cosine-TDP) [17].

GMA does not have basic computation in detecting behavior since it is just calculate the string of the relations that makes AND-OR and OR-XOR is detected as same relationships. Likewise, WGED can not calculate same node with different relationships since it computes inserted/deleted nodes from different activity. Consequently, relationship between AND, OR, and XOR detected as same relationship. Then, WNARS can distinguish between AND-OR, AND-XOR, and OR-XOR through division between the maximum of weight and minimum of weight as stated in [16]. However, computing similarity based on maximum and minimum is too simple such that does not consider the behaviour of the model. TPED, Cosine-TDP, and proposed can detect that AND-OR is more similar to OR-XOR rather than AND-XOR, while GMA detects same relationship between AND-OR and OR-XOR. However, the range difference of similarity AND-OR in 0.821 and AND-XOR in 0.78 TPED is too small. If we rounds up the result similarity between AND-OR and AND-XOR, they will have same value in 0.8 such that seems similar. Also, Cosine TDP detects similarity of AND-XOR in 0.693  $\approx 0.7$  such that AND and XOR is 70% similar while the behaviour is showing higher different. Also, TPED and Cosine-TDP does not consider same relationships with different event log such that they are detected as same relationship with same event log.

WGPPM calculates the number of relation *count(...)* for measuring behaviour in parallel relationships which makes it more accurate than TPED and Cosine-TDP. Through behaviour which is recognized by Weighted Graph Model, we compute the similarity between AND, OR, and XOR properly. Therefore, higher range difference between AND-OR and AND-XOR is achieved.

Furthermore, WGPPM can compare same relationship but different event log precisely. Through WGPPM model which computes weight from number of relations, same relationship can be detected since they have same behaviour even though they have different in number of frequency from different event log. This can be useful for detecting relationships in different event logs to analyze any wrong activities in business process model. In addition, we can distinguish between same relationship with different event log without loss the identity of the same relationship, e.g. AND-AND get similarity value in 0.97 which means they are similar in 97% as their identity of same relationship between AND even they basically from different event logs. GMA, WGED, TPED, and Cosine TDP have similarity value in 1 which means they detect both event logs only have same relationship. They can not distinguish that both are different event logs. Even though WNARS can distinguish same relationship with different event log or Weighted Graph Model, the XOR-XOR relationship loss the identity of the same relationship in 0.79 similarity value because it smaller than AND-OR which is different relationship. WGPPM detects 0.91 in XOR-XOR from different event logs. Same relationship should be detected higher than different relationship since both behaviour are equal even both are from different event logs. In brief, our WGPPM can compute similarity between relationships very well with considering behaviour of relationship.

## 5. Conclusion

This paper proposes new similarity method for matching parallel business process models contains AND, OR, XOR called Weighted Graph-based Parallel Business Process Model Matching (WGPPM) through Weighted Graph Model. The first step is building the Weighted Graph Model as model from the event log. We use the activities from General Ledger Department as event log for experiment. Next, the similarity for matching between two process models is computed from the WGPPM which calculates AND-AND, OR-OR, XOR-XOR, AND-OR, AND-XOR, and OR-XOR.

Table 5. Comparisons of existing methods and proposed method

Similarity	Existing Methods					Proposed (WGPPM)
	GMA [7]	WGED [16]	WNARS [16]	TPED [17]	Cosine-TDP [17]	
AND (Table 3 (a)) and AND (Table 3 (a)) - same Weighted Graph Model	1	1	1	1	1	1
OR (Table 3 (b)) and OR (Table 3 (b)) - same Weighted Graph Model	1	1	1	1	1	1
XOR (Table 3 (c)) and XOR (Table 3 (c)) - same Weighted Graph Model	1	1	1	1	1	1
AND (Table 3 (a)) and AND (Table 4 (a)) -different Weighted Graph Model	1	1	0.91	1	1	0.97
OR (Table 3 (b)) and OR (Table 4 (b)) - different Weighted Graph Model	1	1	0.95	1	1	0.97
XOR (Table 3 (c)) and XOR (Table 4 (c)) - different Weighted Graph Model	1	1	0.79	1	1	0.91
AND (Table 3 (a)) and OR (Table 3 (b))	0.33	1	0.8	0.821	0.834	0.87
OR (Table 3 (b)) and XOR (Table 3 (c))	0.33	1	0.62	0.811	0.826	0.68
AND (Table 3 (a)) and XOR (Table 3 (c))	0.5	1	0.5	0.78	0.693	0.6

The proposed similarity method (WGPPM) is compared with other existing methods, i.e. TPED, Cosine-TDP and Graph-based Matching Approach(GMA). TPED, Cosine-TDP and WGPPM is able to distinguish between AND-OR and AND-XOR while GMA and WGDE detects them as same relationship. However, WGPPM gives similarity more accurate in AND-XOR than TPED and Cosine-TDP because we compute the behavior of AND-XOR as seen in the graph model. TPED achieves 0.821 and 0.78 in AND-OR and AND-XOR, respectively, while Cosine-TDP achieves 0.834 and 0.693 in AND-OR and AND-XOR, respectively. We achieve 0.87 and 0.6 in similarity AND-OR and AND-XOR, respectively, with higher range difference. Furthermore, we compute similarity between different event logs such as AND-AND, OR-OR, and XOR-XOR. These are more accurate than WNARS detects 0.79 in XOR-XOR which is smaller than

AND-OR. We give XOR-XOR precisely in 0.91 since the same relationship should be detected higher even from different event logs. This can be applied in analyzing business process models. For future work, a similarity method for matching parallel business process models with considering activity names can be developed.

### Conflicts of Interest

The authors declare no conflict of interest

### Author Contributions

As the first author, Yohanes Setiawan contributed to the formation of the paper, including the formulation of methods, the implementation of methods and the conduct of experiments. Kelly Rossa Sungkono assisted in formulation of the proposed method and checked the paper. Prof. Riyanarto Sarno



supervised problem ideas and proposed the novelty contribution of the paper.

## Acknowledgments

Authors embed gratitude to Institut Teknologi Sepuluh Nopember that encourages academics to develop their research. Authors are also thankful to Directorate General of Higher Education, the Ministry of Education and Culture of Indonesia for supporting the research.

## References

- [1] A. R. Baskara, R. Sarno, and A. Solichah, "Discovering traceability between business process and software component using Latent Dirichlet Allocation", In: *Proc. of International Conf. on Informatics and Computing*, pp. 251–256, 2016.
- [2] I. G. Anugrah and R. Sarno, "Business Process model similarity analysis using hybrid PLSA and WDAG methods", In: *Proc. of International Conf. on Information & Communication Technology and Systems*, pp. 231–236, 2016.
- [3] S. Garcia, A. Meinheim, E. Ribeiro, F. Junior, M. Rosano, D. Maria, V. Sato, D. Ribeiro, E. Alves, P. Santos, and E. Emilio, "Process mining techniques and applications – A systematic mapping study", *Expert Systems with Applications*, Vol. 133, pp. 260–295, 2019.
- [4] K. Shahzad, R. A. O. Muhammad, A. Nawab, A. Abid, K. Sharif, F. Ali, F. Aslam, and A. Mazhar, "A Process Model Collection and Gold Standard Correspondences for Process Model Matching", *IEEE Access*, Vol. 7, pp. 30708–30723, 2019.
- [5] E. Kuss, H. Leopold, H. van der Aa, H. Stuckenschmidt, and H. A. Reijers, "A probabilistic evaluation procedure for process model matching techniques", *Data and Knowledge Engineering*, Vol. 117, pp. 1–14.
- [6] A. C. Fauzan, R. Sarno, M. A. Yaqin, and A. Jamal, "Extracting Common Fragment Based on Behavioral Similarity Using Transition Adjacency Relations for Scalable Business process", In: *Proc. of International Conf. on Information & Communication Technology and System*, pp. 131–136, 2017.
- [7] R. Sarno, K. Sungkono, and R. Septiarakhman, "Graph-Based Approach for Modeling and Matching Parallel Business Processes", *International Information Institute*, Vol. 21, No. 5, pp. 1603–1614, 2018.
- [8] Y. A. Effendi and R. Sarno, "Modeling parallel business process using modified time-based alpha miner", *International Journal of Innovative Computing, Information and Control*, Vol. 14 pp. 1565–1582, 2018.
- [9] R. Sarno, Kartini, W. A. Wibowo, and A. Solichaha, "Time based Discovery of parallel business processes", In: *Proc. of International Conf. on Computer, Control, Informatics and Its Applications: Emerging Trends in the Era of Internet of Things*, pp. 28–33, 2015.
- [10] E. W. Pamungkas, R. Sarno, and A. Munif, "Performance Improvement of Business Process Similarity Calculation using Word Sense Disambiguation", In: *International Seminar on Science and Technology*, pp. 123–124, 2015.
- [11] J. Zhu and H. K. Pung, "Process matching: A structural approach for business process search", In: *Computation World: Future Computing, Service Computation, Adaptive, Content, Cognitive, Patterns, Computation World*, pp. 227–232, 2009.
- [12] C. Meilicke, H. Leopold, E. Kuss, H. Stuckenschmidt, and H. A. Reijers, "Overcoming individual process model matcher weaknesses using ensemble matching", *Decision Support Systems*, Vol. 100, pp. 15–26, 2017.
- [13] F. Z. Belouadha, H. Omrana, and O. Roudies, "Modeling and matching business processes using Web services", In: *Proc. of International Conf. on Multimedia Computing and Systems*, pp.617–621, 2014.
- [14] B. Mahleko, A. Wombacher, and P. Fankhauser, "A grammar-based index for matching business processes", In: *Proc. of IEEE International Conf. on Web Services*, pp. 21–32, 2005.
- [15] R. Sarno, H. Ginardi, E. W. Pamungkas, and D. Sunaryono, "Clustering of ERP business process fragments", In: *Proc. - 2013 International Conference on Computer*, In: *Control, Informatics and Its Applications: "Recent Challenges in Computer, Control and Informatics"*, pp.319–324, 2013.
- [16] C. Zhou, C. Liu, Z. Lin, and H. U. A. Duan, "A Comprehensive Process Similarity Measure Based on Models and Logs", *IEEE Access*. Vol. 7, pp. 69257-69273, 2019.
- [17] C. S. Wahyuni, K. R. Sungkono, and R. Sarno, "Novel parallel business process similarity methods based on weighted-tree declarative pattern models", *International Journal of Intelligent Engineering and Systems*, Vol. 12, pp. 236–248, 2019.
- [18] M. Becker and R. Laue, "A comparative survey of business process similarity measures", *Computers in Industry*, Vol. 63, pp. 148–167, 2012. doi:10.1016/j.compind.2011.11.003.

- [19] K. R. Sungkono and R. Sarno, “Constructing control-flow patterns containing invisible task and non-free choice based on declarative model”, *International Journal of Innovative Computing, Information and Control*, Vol. 14, pp. 1285–1299, 2018.
- [20] K. R. Sungkono, R. Sarno, and N. F. Ariyani, “Refining Business Process Ontology Model with Invisible Prime Tasks using SWRL rules”, In: *Proc. of International Conf. on Information & Communication Technology and System*, pp. 215–220, 2017.
- [21] L. Makni, N. Z. Haddar, and H. Ben-Abdallah, “Business process model matching: An approach based on semantics and structure”, In: *Proc. of International Conf. on e-Business, Proceedings; Part of 12th International Joint Conference on e-Business and Telecommunications*, pp. 64–71, 2015.
- [22] R. Sarno and K. R. Sungkono, “Coupled Hidden Markov Model for Process Discovery of Non-Free Choice and Invisible Prime Tasks”, *Procedia Computer Science*. Vol. 124, pp. 134–141, 2017.
- [23] C. Klinkmüller and I. Weber, “Analyzing control flow information to improve the effectiveness of process model matching techniques”, *Decision Support Systems*, Vol. 100, pp. 6–14, 2017.
- [24] R. Dijkman, M. Dumas, B. Van Dongen, R. Krik, and J. Mendling, “Similarity of business process models: Metrics and evaluation”, *Information Systems*, Vol. 36, pp. 498–516, 2011.