Mining Collaboration Business Process Containing Invisible Task by Using Modified Alpha

Abdullah Faqih Septiyanto
Department of Informatics Engineering
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
faqih.206025@mhs.its.ac.id

Riyanarto Sarno
Department of Informatics Engineering
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
riyanarto@if.its.ac.id

Kelly R. Sungkono
Department of Informatics Engineering
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
kelly@its.ac.id

***Abstract*—Business processes are experiencing increasingly complex developments; therefore, an extensive business process must cover all existing process flows. Applied business process collaboration between organizations can complete a complex business process. The additional information which shows the collaboration of activities is called messages. Process discovery is currently focused on a series of activities in a single process model, so the process discovery cannot depict the messages in the business process collaboration. In addition, there are several problems in describing the condition of activities, e.g., an Invisible Task. The Invisible Task is a condition of additional tasks that appear not in the event logs but in the process models. The Invisible Task must be described in the process model; therefore, it can be analyzed further. Several conditions which need the Invisible Task are redo, switch, and skip conditions. In this research, the proposed method is to obtain information about the event log of all activities of the business process collaboration and discover any Invisible Task to describe in the process model. The proposed method, named the Modified Alpha algorithm, builds several rules for adding messages and the Invisible Task in the event log before executing the Alpha algorithm. The results of this study indicate that the Modified Alpha algorithm can describe the collaboration process model. Based on the comparison results, the Modified Alpha algorithm gets the best results than other algorithms, namely Alpha# and Inductive Miner. Modified Alpha received 1.00, 1.00, 1.00, 0.82 for the fitness, precision, simplicity, and generalization. Alpha# Miner earned 0.74 and 0.70 for the simplicity and generalization, and Inductive Miner gained 0.55 simplicity value and 0.72 generalization value. Alpha# Miner and Inductive Miner got 0.00 for the fitness and the precision.**

Keywords—Alpha Miner, Collaboration Business Process, Invisible Task, Process Discovery, Process Mining

I. INTRODUCTION

Business processes are activities within the company dependent on activities by the sequence of activities that occur within the company [1]. In extensive business processes, there can be a collaboration between interrelated sub-processes to complete the process flow and use automation company performance along with the increasing use of business process management technology [2]. This collaboration occurs when there is a relationship between processes that run independently and provide information to each other to be reprocessed in each existing sub-process. Every company must have a standard business process to achieve specific goals [3]. Standard Operating Procedure (SOP) is a standard determined to detect irregularities that occur in existing business processes [4]. The SOP can be used as a reference to assess whether all employees can complete tasks according to the procedure. The existence of an information system in the company will generate event logs based on the activities undertaken, which can provide knowledge of business process activities that have just been running or have been running in the past [5]. Results of the event log can be analyzed and extract business processes using process mining [6].

Process mining combines machine learning studies with data mining, both of which are done to analyze the business processes used by an organization. Process discovery is one of the tasks of process mining to generate a process model using an event log [7] to analyze business processes which allows getting information about control flow, organization, and time [8]. There are algorithms to analyze event logs in the process discovery, such as Alpha Miner, which consists of Alpha, Alpha+, and Alpha++, or the Inductive Miner algorithm. Each of these algorithms has its advantages and disadvantages in the event log process discovery [7].

In previous research, the Alpha++ algorithm was modified to be able to detect event logs with various conditions; these algorithms include new Time-based Alpha++ algorithm [7] and a hybrid of non-free-choice and Invisible Task discover by Modified Alpha++ [9]. In [7], a new rule based on the double-timestamp event log is added to determine the pattern of time intervals and can define all sequences and parallel relationships. Furthermore, in [9], the overlapping pattern that occurs is added with the Invisible Task automatically to the event log. Both studies were successful in being able to cover business processes with Modified Alpha++ algorithm.

This research proposed a Modified Alpha algorithm to mining collaboration business processes that contain Invisible Task. The existing process discovery algorithms, namely the Alpha, Alpha+, and Alpha++ algorithms, cannot mine a collaboration business process and handle Invisible Task, so the Petri Net generated from the process discovery of the three methods is not following the SOP. The researcher chooses the Alpha algorithm to be modified because this research only focuses on producing the appropriate Petri Net from the collaboration business process and can detect the location of the Invisible Task. In contrast, the concentration of the Alpha+ [10] algorithm is used to handle short loops, and the Alpha++ [11] algorithm is used to manage non-free-choice. The Waste Handling business process generates event logs for the discovery process. Two sub-processes collaborate, namely Flow Waste (FW) and Flow Document (FD). All activities of those sub-processes are stored in the system. The storing activities provide much information about their existence and the exchange of information between the two sub-processes. This research can pull an event log considering that information by the sequence of activities based on the SOP. Based on the event log, Invisible Task was found using the Invisible Task search algorithm and made modifications to the

978-1-6654-4059-2/21/$31.00 ©2021 IEEE

event log; therefore, the Alpha algorithm could discover Invisible Task. Invisible Task clarifies some process conditions in the process model as additional tasks [12]. Results of process discovery are calculated using fitness, precision, simplicity, and generalization for comparison with other process discovery algorithms.

II. Research Method

A. Collaboration Business Process

Collaboration or collaborative business process is a business process that is mutually integrated between organizations [13]. Until now, collaborative business processes are often used and become increasingly complex [14] – hence there are many sub-processes in the main business processes. In business process collaboration, there is a flow of message delivery between sub-processes [15]. The message contains the information needed to be processed in each organization; forasmuch as it can trigger other organizations to carry out their business processes based on the delivery of the message.

B. Fitness, Precision, Simplicity, and Generalization

The calculation of fitness, precision, simplicity, and generalization is done in assessing the results of the process discovery from the application of nowadays process discovery algorithms. Fitness calculation to count all traces are caught in a model, precision calculation to count all traces of the model arrest based on event log [9]. Simplicity calculation to count by comparing the process tree size of a process model with the activity in the event log [16]. Generalization calculation to count frequency of the appearance of nodes in the process tree based on the event log. In this research, equations to calculate fitness as in Eq. 1, precision as in Eq. 2, simplicity (Q_s) as in Eq. 3, and generalization (Q_g) as in Eq. 4.

$$Fitness = \frac{n_{ct}}{n_t} \quad (1)$$

where:

n_{ct} : trace in model process

n_t : trace of transformation SOP

$$Precision = \frac{n_{ctm}}{n_{tm}} \quad (2)$$

where:

n_{ctm} : trace of transformation SOP

n_{tm} : $trace\ in\ model\ process$

$$Q_s = 1 - \frac{\#da + \#ma}{\#npt + \#ec} \quad (3)$$

where:

$\#da$: total events in the event log displayed redundantly in the process tree

$\#ma$: total activities in the event log not displayed in the process tree and total of leaf nodes not either in the event log.

$\#npt$: total leaf nodes in the process tree.

$\#ec$: total events in the event log.

$$Q_g = 1 - \frac{\sum_i^{nt} (\sqrt{\#exc})^{-1}}{\#nt} \quad (4)$$

where:

$\#nt$: total node operators in the process tree.

nt : node operator enforced in the event log.

$\#exc$: total operator nodes enforced in the trace event log in the event log.

III. Proposed Method

The proposed method in this study is to describe the flow of modification of the Alpha algorithm therefore it can handle Invisible Task in business process collaboration. The proposed method is depicted in Fig. 1.

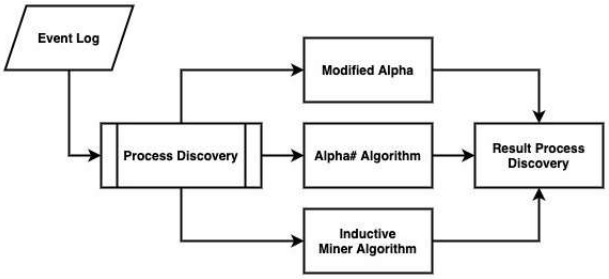

Fig. 1. Proposed Method

TABLE I. List Abbreviations of Activities Names

Initial	Description
CW	Select Waste
PWI	Place Rubbish in Recycle Bin
PWS	Place Rubbish in Safety Pack
PWY	Place Rubbish in Yellow Basket
PWB	Place Rubbish in Brown Basket
PWP	Place Rubbish in Purple Basket
PWR	Place Rubbish in Red Basket
SWR	Send Rubbish to the Recycle Place
SW	Discard Rubbish
SWI	Send Rubbish to Incinerator
ERW	Entry Recycled Rubbish Advent Log
CWT	Clean Rubbish
ENW	Entry Non-Recycled Rubbish Advent Log
BW	Incinerate Rubbish
PA	Pack Cinders
CLW	Cleavage Rubbish
SOW	Soak Rubbish
EAT	Entry Cinders of Rubbish Log
CHW	Slash Rubbish
DW	Arid Rubbish
SAT	Save Cinder in TPS Non-Recycled Rubbish
PW	Pack Rubbish
ELA	Entry Log of Cinders in TPA
WW	Weight Rubbish
CAD	Make Cinders Delivery Schedule Offer
GAN	Earn Approval for Non-Recycled Rubbish Schedule
ERP	Entry Recycled Rubbish Processed Log
SWW	Save Rubbish in TPS Recycled Rubbish
TAP	Take Cinders by P3
ERT	Entry Log of Recycled Rubbish in TPA
CMA	Make Minutes of Cinders Shipments
CWD	Make Rubbish Delivery Schedule Offer
SAP	Send Cinders by P3
GAW	Earn Approval for Rubbish Schedule
TWI	Take Rubbish by Industry
CMW	Make Minutes of Rubbish Shipments
SWY	Send Rubbish to Industry
IT1-5	Invisible Task 1-5
MS1-12	Messages ID 1-12

A. Event Log Retrieval Rules

In the case of collaboration business process requires a separate Structured Query Language (SQL) to get a complete event log to describe the process for archiving a better process model, an event log should record all activities carried out in the information system database sequentially according to the SOP. SOP of Waste Handling Process is shown in Fig. 2, the transformation of SOP depicted in Fig. 3, and a list abbreviations of activities names is depicted in TABLE I. Retrieval of event logs using default SQL from database which is commonly used to retrieve event logs on non-collaborative business processes in previous studies. The results of the event log using default SQL there is no information indicating that in the event log there is a collaboration between organizations. Therefore, the results cannot describe the overall information from the business process collaboration. The results of using the default SQL query are depicted in TABLE II.

The collaboration business process of Waste Handling produces a lot of information. From TABLE II, each activity cannot show the organization's name, and the CaseID column cannot provide the correct value for the collaboration business process. Then SQL query is modified to get a complete event log by the business process collaboration. Modified SQL queries by Modified Alpha depicted in TABLE IV.

TABLE II. PART OF EVENT LOG FROM RESULT DEFAULT SQL QUERY

CaseID	Activity	Timestamp	Resource
1	CW	10/30/2020 10:53:55	AW
1	PWS	10/30/2020 10:57:21	AW
1	SWI	10/30/2020 11:02:05	AW
2	ENW	10/30/2020 11:06:22	AD
1	BW	10/30/2020 11:10:08	AW
1	PA	10/30/2020 11:14:43	AW
2	EAT	10/30/2020 11:18:12	AD
1	SAT	10/30/2020 11:22:07	AW
2	ELA	10/30/2020 11:27:22	AD
2	CAD	10/30/2020 11:31:26	AD
2	GAN	10/30/2020 11:37:44	AD
1	TAP	10/30/2020 11:42:01	AW
2	CMA	10/30/2020 11:47:42	AD
1	SWY	10/30/2020 11:53:55	AW

The modified SQL queries by Modified Alpha have undergone many changes compared to the default SQL queries. The modified SQL queries retrieve information about the message and name of the sub-process for each activity. Therefore, it can show a more detailed information event log and depict in TABLE III. The event log contains Process Name, CaseID, Activity, Timestamp, Resource, and Message To. Process Name column shows the sub-process name of the activity, CaseID column shows the Case ID for each existing case. Activity column shows the name of the activity that was executed, Timestamp column shows the start time of the activity, Resource column shows the name of the operator who carried out the activity, and Message To shows activity name of receiving message.

TABLE III. PART OF EVENT LOG CONTAINING MESSAGES

PN	CaseID	Activity	Timestamp	R	MT
FW	1	MS1	10/31/2020 08:56:51	AW	ENW
FW	1	MS2	10/31/2020 09:19:33	AW	EAW
FW	1	MS3	10/31/2020 09:35:31	AW	ELA
FD	1	MS4	10/31/2020 10:16:09	AD	TAP
FW	1	MS5	10/31/2020 10:28:38	AW	CMA
FD	1	MS6	10/31/2020 10:42:22	AD	SAT

where:
PN : Process Name
R : Resource
MT : Message To

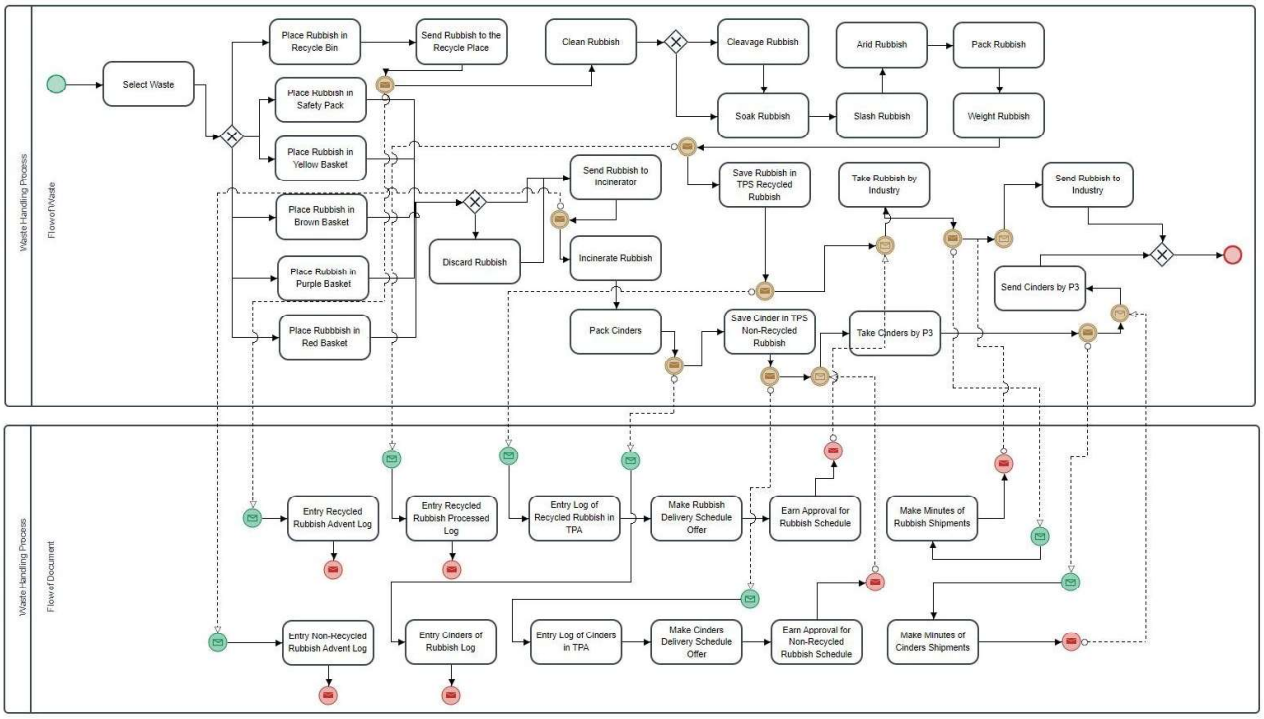

Fig. 2. SOP of Waste Handling Process

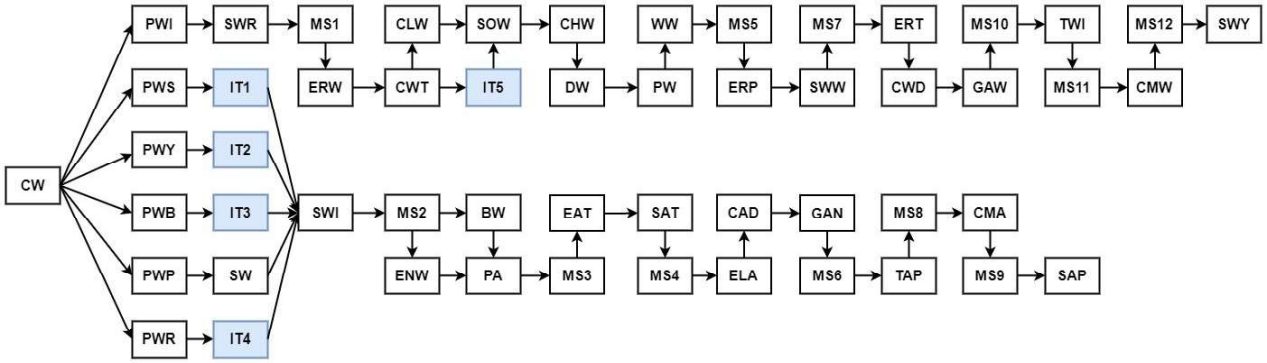

Fig. 3. Model Process Transformation of SOP

TABLE IV. PSEUDOCODE FOR GETTING EVENT LOG FROM DATABASE IN COLLABORATION BUSSINESS PROCESS QUERY

	Pseudocode Using for Database Process Maker
1	GET IF LAN_NAME AS PN IS NULL
2	GET PN FROM bpmn_lane Which is related with bpmn_bound AND APV
3	IF DFD = MSGTIME
4	GET PN, CID, ACT, MSGTIME
5	IF ORG IS NULL
6	GET ORG FROM users Which is related with APV
7	IF DFD = MSGTIME LIMIT 1
8	ELSE ORG
9	GET TAS_TITLE AS TT FROM task Which is related with app_history
0	IF HISTORY_DATE AS HD > MSGTIME LIMIT 1 AS MT
10	Which is related with bpmn_event AND APV
11	IF EVN_TYPE = 'INTERMEDIATE'
12	MSGTIME = DEL_DELEGATE_DATE
13	ELSE
14	MSGTIME = APP_FINISH_DATE
15	Which is related is task AND bpmn_bound AND bpmn_lane AND users
16	GROUP BY TIME
17	MERGE ALL
18	GET PN, CID, TAS_TITLE AS ACT, HISTORY_DATE AS TIME, ORG
19	SET NULL AS MT
20	FROM bpmn_lane Which is related with bpmn_bound AND task Which is related with app_history AND users
21	GROUP BY CID AND DEL_INDEX
22	SORT BY TIME

where:

APV : App Cache View
DFD : Delete Finish Date
MSGTIME : Message App Throw Date
CID : CaseID
ACT : Activity
TIME : Timestamp
ORG : Resource

B. Process Discovery

At this stage, the researcher modifies the Alpha algorithm to detect the presence of an Invisible Task. Invisible Task detection algorithm based on Alpha base logic. There are two steps to detect Invisible Task: (1) Draw Patterns Based Event Log and (2) Modify Event Log Based Check Invisible Task. Part of the event log take which contains two different cases and two traces is depicted in TABLE V. Invisible Task detection is done by drawing a pattern based on the event log depicted in Fig. 4. The pattern drawing process is carried out to determine the relationship between the activity and the Invisible Task based on the event log. Next, modify the event log based on the location where an Invisible Task is detected.

TABLE V. PART OF EVENT LOG

CaseID	Activity
2	CW
2	PWR
2	SWR
2	MS1
2	ERW
2	CLW
2	CVW
2	SW
3	CW
3	PWR
3	SWR
3	MS1
3	ERW
3	CLW
3	SW

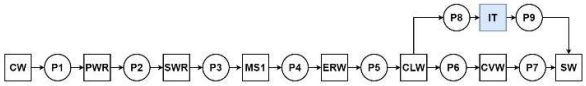

Fig. 4. Patterns Based Event Log

TABLE VI. DETAIL EVENT LOG

PN	CL	Cases	Traces	Activities	Messages	IT
FW	Yes	9	7	25	8	5
FD	Yes	9	7	12	4	0

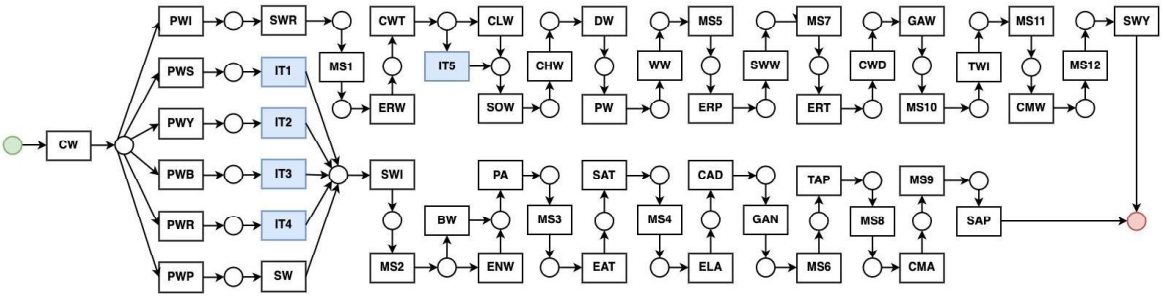

Fig. 5. Petri Net from Modified Alpha Algorithm

1) Draw Patterns Based Event Log

The connector between activities is initialized with P1...Pn. In this research, there were 9 relations between activities used. The pattern result obtained from all the steps carried out gets the result depicted in Fig. 4. At this step, it is known that an Invisible Task is detected in the relationship between CLW and SW. Where CLW, which usually passes CVW before SW, in certain cases can go directly to SW without passing CVW. Therefore, the location of the invisible task is detected on P8 and P9.

2) Modify Event Log Based Check Invisible Task

At this stage, the event log is modified by adding a new activity, namely Invisible Task. based on the results of Fig. 4, detected the location of the invisible task to produce a pattern based on the event log depicted in TABLE V. Invisible Task is added between CLW and SW activities with the given time based on the period of the two activities. Modifications are made to each location where an Invisible Task is detected in the event log.

IV. Experimental Result

A. Material Data

In this research, the proposed method is applied to obtain event logs from collaboration business process activities with messages between FW and FD with adding new activities based on detection of Invisible Task. Details of the obtained event log results are depicted in TABLE VI.

From TABLE VI, in the FW sub-process, there are 25 activities, 8 messages, and 5 Invisible Task, therefore the total activities contained in the FW sub-process are 38. In the FD sub-process, there is no Invisible Task so there are 12 activities, 4 messages, the total activity in the FD sub-process is 16. The CL column in TABLE VI shows the two sub-processes have collaborated.

B. Evaluation

After the Invisible Task is added into the event log, the discovery process is implemented. In all event log data, five Invisible Task were detected. The five locations were added with new activities called Invisible Task 1 to Invisible Task 5. The process discovery uses the Modified Alpha algorithm, Alpha# algorithm, and Inductive Miner algorithm. The process discovery result from the Modified Alpha algorithm is depicted in Fig. 5. The result of the Alpha# algorithm is depicted in Fig. 6, and the result from Inductive Miner is depicted in Fig. 7.

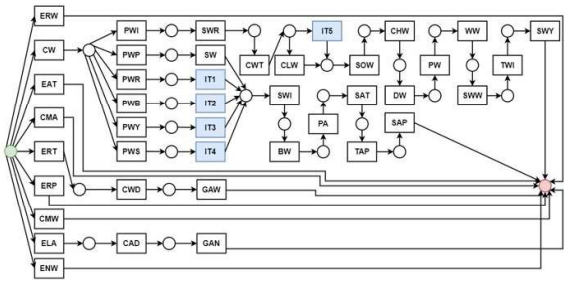

Fig. 6. Petri Net from Alpha# Algorithm

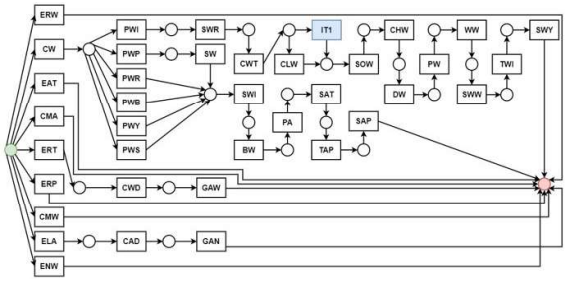

Fig. 7. Petri Net from Inductive Miner

From Fig. 5, the result of the proposed method shows that the Petri net can describe the collaboration business process based on all traces in the event log and describe the existence of all Invisible Task. In Fig. 6, the Alpha# algorithm can define all Invisible Task but cannot describe the collaboration business processes. Each first activity in the sub-process FD is described as a new trace, and all messages cannot be described as an activity. In Fig. 7, the Inductive Miner algorithm same as the results of the Alpha# algorithm but cannot describe all Invisible Task in the entire business process.

Finally, the comparative results of the process findings are depicted in TABLE VII. Modified Alpha received the highest values of fitness, precision, simplicity, and generalization than others. Modified Alpha got the fitness, precision, simplicity value of 1 and the generalization value of 0.82 because all traces in the process model match the process transformation model from the SOP. Alpha# and Inductive Miner produced 0.00 for the fitness and the precision because there are no traces in the process model of process discovery that match the SOP. Alpha# earned 0.02 fewer generalization values than Inductive Miner, whereas the simplicity of Alpha# scored 0.19 higher than Inductive Miner. It can be concluded that Modified Alpha can produce the best results among the three tested methods.

TABLE VII. COMPARISON RESULTS OF PROCESS DISCOVERY

Algorithm	Fitness	Precision	Simplicity	Generalization
Modified Alpha	1.00	1.00	1.00	0.82
Alpha#	0	0	0.74	0.70
Inductive Miner	0	0	0.55	0.72

V. CONCLUSION

Through a proposed algorithm called Modified Alpha, this research focuses on how to describe the process model in business process collaboration. The event log is obtained by modified queries to get all the activities along with the messages generated from the business process collaboration. Invisible Task is a problem that must be handled in the process model, so Modified Alpha detects the location of the Invisible Task in the collaboration process, adds new activities based on the area, and forms the process model by Alpha algorithm.

The research results show that queries of Modified Alpha can get all activities along with messages from business process collaboration. Modified Alpha can detect the location of Invisible Task. The evaluation results show that Modified Alpha provides the best process model results with the fitness, precision, simplicity values are 1, and the generalization is 0.82.

Future work in this research is to create own algorithm to produce a model process that can describe the existing sub-processes based on the process name contained in the event log and handle noise in the event log used. Because in research, the resulting process model is still in one unit and has not dealt with any noise in the event log.

ACKNOWLEDGMENT

This research was funded by the Indonesian Ministry of Education and Culture under Penelitian Terapan Unggulan Perguruan Tinggi (PTUPT) Program managed by Institut Teknologi Sepuluh Nopember (ITS) and under Riset Inovatif-Produktif (RISPRO) Invitation Program managed by Lembaga Pengelola Dana Pendidikan (LPDP).

REFERENCES

[1] K. N. Aisyah, K. R. Sungkono, and R. Sarno, "A new similarity method based on weighted-linear temporal logic tree and weighted directed acyclic graph for graph-based business process models," *International Journal of Intelligent Engineering and Systems*, vol. 13, no. 5, pp. 356–367, 2020, doi: 10.22266/ijies2020.1031.32.

[2] R. Sarno and K. R. Sungkono, "A survey of graph-based algorithms for discovering business processes," *International Journal of Advances in Intelligent Informatics*, vol. 5, no. 2, pp. 137–149, 2019, doi: 10.26555/ijain.v5i2.296.

[3] A. R. Baskara, R. Sarno, and A. Solichah, "Discovering traceability between business process and software component using Latent Dirichlet Allocation," *2016 International Conference on Informatics and Computing, ICIC 2016*, no. Icic, pp. 251–256, 2017, doi: 10.1109/IAC.2016.7905724.

[4] R. Sarno and K. R. Sungkono, "Coupled Hidden Markov Model for Process Discovery of Non-Free Choice and Invisible Prime Tasks," in *Procedia Computer Science*, 2017, vol. 124, pp. 134–141, doi: 10.1016/j.procs.2017.12.139.

[5] H. Darmawan, R. Sarno, A. S. Ahmadiyah, K. R. Sungkono, and C. S. Wahyuni, "Anomaly Detection based on Control-flow Pattern of Parallel Business Processes," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 16, no. 6, p. 2809, Dec. 2018, doi: 10.12928/telkomnika.v16i6.10568.

[6] R. Sarno, K. R. Sungkono, M. Taufiqulsa'di, H. Darmawan, A. Fahmi, and K. Triyana, "Improving efficiency for discovering business processes containing invisible tasks in non-free choice," *Journal of Big Data*, vol. 8, no. 1, pp. 1–17, 2021, doi: 10.1186/s40537-021-00487-x.

[7] Y. A. Effendi and R. Sarno, "Parallel process discovery using a new Time-Based Alpha++ Miner," *IIUM Engineering Journal*, vol. 21, no. 1, pp. 126–141, 2020, doi: 10.31436/iiumej.v21i1.1173.

[8] O. Tamburis and C. Esposito, "Process mining as support to simulation modeling: A hospital-based case study," *Simulation Modelling Practice and Theory*, vol. 104, no. March, p. 102149, 2020, doi: 10.1016/j.simpat.2020.102149.

[9] L. Hakim, R. Sarno, and K. R. Sungkono, "Modified Alpha++ algorithm for discovering the hybrid of non-free choice and invisible task of business processes," *International Journal of Intelligent Engineering and Systems*, vol. 12, no. 3, pp. 31–40, 2019, doi: 10.22266/ijies2019.0630.04.

[10] A. K. A. de Medeiros, B. F. van Dongen, W. M. P. van der Aalst, and A. J. M. M. Weijters, "Process mining: Extending the a algorithm to mine short loops," 2004.

[11] L. Wen *et al.*, "Mining process models with non-free-choice constructs," *Data Mining and Knowledge Discovery*, vol. 15, no. 2, pp. 145–180, 2007, doi: 10.1007/s10618-007-0065-y.

[12] 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, no. 4, pp. 1285–1299, 2018, doi: 10.24507/ijicic.14.04.1285.

[13] B. Niehaves and R. Plattfaut, "Collaborative business process management: status quo and quo vadis," *Business Process Management Journal*, vol. 17, no. 3, pp. 384–402, 2011, doi: 10.1108/14637151111136342.

[14] B. Fleaca, "Organization and Business Environment Collaborative Model to Increase the Innovation Capacity," *International conference KNOWLEDGE-BASED ORGANIZATION*, vol. 24, no. 1, pp. 296–301, 2018, doi: 10.1515/kbo-2018-0047.

[15] F. Corradini, A. Morichetta, A. Polini, B. Re, L. Rossi, and F. Tiezzi, "Correctness checking for BPMN collaborations with sub-processes," *Journal of Systems and Software*, vol. 166, 2020, doi: 10.1016/j.jss.2020.110594.

[16] J. C. A. M. Buijs, B. F. Van Dongen, and W. M. P. Van Der Aalst, "Quality dimensions in process discovery: The importance of fitness, precision, generalization and simplicity," *International Journal of Cooperative Information Systems*, vol. 23, no. 1, 2014, doi: 10.1142/S0218843014400012.