Discovering Optimized Process Model using Rule Discovery Hybrid Particle Swarm Optimization

Yutika Amelia Effendi, Riyanarto Sarno
Department of Informatics
Faculty of Information Technology, Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
e-mail: yutika.effendi@gmail.com, riyanarto@if.its.ac.id

Abstract—This paper presents a bio-inspired hybrid method which concentrate on the optimal or a near-optimal business process model from an event log. The discovery of Hybrid Particle Swarm Optimization (Hybrid PSO) algorithm comes from the combination of Particle Swarm Optimization (PSO) algorithm and Simulated Annealing (SA) method. This paper presents a method which combines Rule discovery task and Hybrid PSO. The proposed method can discover not only classification rules that produce the most optimal business process model from event logs, but also can optimize the quality of process model. To be formulated into an optimization problem, we use rule discovery task to get the high accuracy, comprehensibility and generalization performance. After we get the results from rule discovery task, we use Hybrid PSO to resolve the problem. In this proposed method, we use continuous data as data set and fitness function as evaluation criteria of quality of discovered business process model. As final results, we prove that the proposed method has the best results in terms of average fitness and number of iterations, compared with classical PSO algorithm and original hybrid PSO algorithm.

Keywords—Particle Swarm Optimization; rule discovery; Hybrid PSO; Simulated Annealing; process mining; optimization; average fitness; business process

I. INTRODUCTION

A process which analyzes business processes based on event logs within an organization is defined as process mining. Event log contains the specific information, such as case identification number/ case ID, activities, date execution, timestamp, activity resources, and activity cost [1]. Using process mining techniques, the organization's business processes are capable of being discovered [2], improved [3] and optimized [4]. An optimization problem is the problem of finding the most optimal of causality relations from all activities in an event log. Based on this problem, the optimal discovered business process should be represented well according to the event log [5].

There are many algorithms to solve optimization problem which use birds, insects and animals that has successfully survived and find the best way to migration and foraging in their colony, such as PSO (Particle Swarm Optimization), BCO (Bee Colony Optimization), and ACO (Ant Colony Optimization) [6]. All algorithms are usually related with many optimization problems. Because the purpose of research is to provide the most optimal, then the best choice is bio-inspired meta-heuristics method which can generate the results in a short time. The bio-inspired meta-heuristics method only process the half of the search space. Meanwhile, we use simulated annealing (SA) for finding a good solution to an optimization problem. SA is a method which uses probabilistic technique. To predict global optimization in a large search space, it is suitable to use SA method. SA may be preferable for problems which prefers finding a global optimum rather than finding a local optimum at a time.

In this paper, we present how the PSO algorithm meta-heuristic [7] can be adjusted to identify the most optimal solution in business process mining. To be formulated into an optimization problem, we combine the Hybrid PSO with Rule discovery task to get the high accuracy, comprehensibility and generalization performance. After we get the results from rule discovery task, we use hybrid particle swarm optimization to resolve the problem. There are three main methods in PSO, namely modeling the entities of PSO, processing the problems, and running the algorithm for discovering the business process model from the event log. We use SA method to overcome the lack of PSO algorithm, because SA is suitable for finding the best solution in term of the current processed solution. In this paper, we follow Michigan approach for our rule discovery hybrid particle swarm optimization where an individual produces one prediction rule. If we do one iteration, we will get one discovered rule. Therefore, the we have to run the algorithm several times to get a set of rules. So, if we want to predict N different classes, we have to run the algorithm at least N times [8].

Next step, we need to determine the evaluation criteria of a discovered business process model, specifically the quality criteria. It relates with completeness and preciseness. In this paper, fitness function is calculated because it contains the evaluation criteria of quality of business process model [9]. We organize this paper into several sections. In related work section, we review works that relate to our proposed method. In Section III, we introduce our proposed method. The algorithm and the experimental results are explained in Section IV and V. Last, we conclude this paper with conclusions and future work in Section VI.

978-1-5090-5864-8/17/$31.00 ©2017 IEEE

II. Related Work

In the related work, we review works that relate to our proposed method in business process mining.

Many algorithms can analyze the business process mining, such as α, α+, α++ and heuristics miner algorithm. They can discover all relations from all activities of the event log as a result. The two main steps of the algorithms, namely using an event log to get the input activity and mining the event log to produce a workflow net. Based on the information in the event log which activity as predecessor or follows another activity can be captured by the algorithms, such as follows relations, causal relations, parallel relations and unrelated relations [1]. Recently, the new publication in business process mining area is about an optimization problem. Generally, its method is used to produce the most optimal business process model [6].

In [9], authors use genetic-based algorithm to solve the problem of business process. Causal matrix is a representation of discovered business process model after we map a genetic individual. Using the causal matrix, we can get the dependency relations from all activities of the event log. Genetic individual has the quality criteria. It also relates with completeness and preciseness. In this paper, a fitness function is calculated because it contains the evaluation criteria of quality of a genetic individual. Genetic-based algorithm uses the mutation and crossover to provide the best genetic individual. Sequences activities, choice, parallel relations, short loops, tasks which are invisible and duplicate, non-free choice construct, and noisy logs are able to discovered by using genetic-based algorithm.

In [10], a SA-based process mining method is proposed by the authors. Causal matrix is an extended model of a discovered process model [9]. In this paper, we apply SA method twice. Each applied time try to identify different types of control structures. Basic structures, duplicate and invisible tasks are identified in the first processing, while non-free choice construct is identified in the second processing using the SA method. To determine the quality criteria of a discovered business process, this paper also uses a fitness function which deals with completeness and preciseness [9]. This method, the α algorithm, and genetic algorithm were analyzed [11]. Based on experimental results of this paper, the SA method is able to identify both duplicate and invisible tasks, unlike the α algorithm, and producing the same results as genetic-based algorithm, but in a short time.

The combination of SA method and PSO algorithm is presented in [10] which is an improved version of SA-based process mining method. The authors use SA method to improve the result which give impact to the quality of a discovered business process model. Using principal of SA method which uses probabilistic technique, so if the initial solution has a worse quality than the chosen solution, then the new local optimal is the chosen solution. There is a possibility for accepting a worse solution, not the initial solution. This paper is also explained well. If the initial solution has a worse fitness values than the worse solution, then we choose the worse solution to be the local optimal.

Authors in [8] propose PSO algorithm to discover rules. The standard form of rule is IF-THEN which are chosen in the discovery process. To be formulated into an optimization problem, we use rule discovery task to get the high accuracy, comprehensibility and generalization performance. After we get the results from rule discovery task, we use Hybrid PSO to resolve the problem. According to the experimental results of this paper, this method can be used for categorical data set and continuous data set.

III. Rule Discovery Hybrid PSO Model

In section III, we explain the components of PSO and components of SA included in the rule discovery hybrid PSO model.

A. *Components of PSO Algorithm*

In this section, we explain about PSO algorithm and the main step of PSO to solve the problem of business process model.

1) *PSO Algorithm*

Eberhart and Kennedy introduced the PSO algorithm in 1995. PSO is one of the optimization algorithms based on evolutionary computation technique. The inspiration of this algorithm comes from social behavior simulation of a bird colony [6]. Birds colony in nature that has successfully found the best way to foraging becomes the inspiration in developing PSO algorithm. When foraging or searching for food, the birds colony without being centraly coordinated collaborate with other bird to find a food source. During the foraging, the birds colony can find an area which contains their food source and they can efficiently combine not only the information regarding the other bird's current position, but also the information regarding the former position. If the neighborhood birds find a rich food area, the birds emit the loudest sound to communicate with other birds [12].

In the principal of PSO algorithm, to identify the global optimum, birds become particles which explore the search space, characterized by a position of particle and a velocity of particle. In (1), we calculate the position of a particle x_i^{t+1} at the time $t+1$.

$$x_i^{t+1} = x_i^t + V_i^{t+1} \quad (1)$$

where the symbols mean:

x_i^t : the position of particle at the time

V_i^{t+1} : the new velocity of particle at the time

Three components that affected the velocity of the particle, such as the previous velocity of particle (V_i^t) as well as the position of particle (x_i^t), the best local optimum solution (x_i^l) and last, the best global optimum solution (x_j^g). In (2), we calculate the new velocity of the particle:

$$V_i^{t+1} = x * V_i^t + y * (x_i^l - x_i^t) + z * (x_j^g - x_i^t) \quad (2)$$

where the symbols mean:

$x\ y\ z$: three random numbers

2) Main Step of PSO Algorithm to Solve the Problem of Business Process Model

The main step of PSO algorithm need to be adapted and defined correctly so that the algorithm can solve the problem of business process model. There are some components to be defined, such as particle, position of particle, velocity of particle, substraction, additional, and multiplication operators [6]. In this paper, we give description a particle as an agent. Discovered business process model is illustrated as an agent which has an important role. Meanwhile the velocity of particle can improve the result of process model. In this proposed method, the causal matrix represents a business process model which is defined in (3):

$$CM = \{(In\ (a_1), Out\ (a_1)), \ldots, (In\ (a_n), Out\ (a_n))\} \quad (3)$$

where the symbols mean:
a_i : activity in the event log
In (a_i) : activity which directly precede activity a_i
Out (a_i) : activity which directly follow activity a_i
n : the total number from all activities of the event log

An XOR-split or XOR-join relation is generated from the activities in $In\ (a_i)/Out\ (a_i)$ which has the same subset, while an AND-split or AND-join relation is generated from the activities in $In\ (a_i)/Out\ (a_i)$ which has the different subset [9]. The velocity of particle has the same representation as the position of the particle. In process mining, we need to identify the dependency measure. In this proposed method, the calculation of the dependency measure (DM) is obtained based on dependency relations between two activities and the frequency of the short loops.

B. *Component of Simulated Annealing*

In this proposed method, we combine PSO algorithm with SA to improve the results which give impact to the discovered business process model. The combination of PSO and SA is called hybrid particle swarm optimization. At first, the inspiration of SA comes from annealing process in metallurgy [13] which gives intention to decrease the defect of material as well as to increase the dimensions of the material. Using principal of SA method which uses probabilistic technique, so if the initial solution has a worse quality than the chosen solution, then the new local optimal is the chosen solution. There is a possibility for accepting a worse solution, not the initial solution. This paper is also explained well. If the inital solution has a worse fitness values than the worse solution, then we choose the worse solution to be the local optimal.

In our proposed method, we use a business process model to represent a solution involved in SA. We need to obtain the neighborhood of a solution which can be formed by following a set of rules [9].

- Remove and add all activities from the set of solution of the In sets and Out sets
- Decrease and increase the number of the set solution of the In sets and Out sets

Using (4), we can determine whether worse solution can be accepted or not [13]:

$$P = e^{(Fitness(X') - Fitness(X))/T} \quad (4)$$

where the symbols mean:
X : the initial solution
X' : the worse solution
$Fitness\ (X), Fitness\ (X')$: the fitness function [9]
$X\ and\ X' \in [-1,1]$
T : the temperature at the time.

So, based on (4) we can conclude that the worse solution can be accepted if a random number is higher than the value of P, vice versa.

C. *Rule Discovery Task*

In this paper, we follow Michigan approach for our rule discovery hybrid particle swarm optimization where an individual produces one prediction rule. If we do one iteration, we will get one discovered rule. Therefore, the we have to run the algorithm several times to get a set of rules. So, if we want to predict N different classes, we have to run the algorithm at least N times [8].

1) Rule Presentation

In Fig.1 we represent three main structures of a particle in the range [0,1]. Based on Fig.1, decision attributes and each part has same number of *n* elements. If decision attributes and each part have n elements, then *3n* is the size of a particle. Next step is if we want to form a rule based on a particle, we need to translate three parts into the original information, namely existence of attributes, between attributes and their values needs operators, and in the data set should use original values of attributes [8].

Array-of-Attribute-Existence	Array-of-Operator	Array-of-Attribute

Fig. 1. Three Main Structures of a Particle.

First main structure of a particle is *array-of-attribute-existence.* The *j*-th attribute appears in rule antecedent, if the *j*-th element value is greater than 0. The *j*-th attribut does not appear in rule antecedent, if the *j*-th element value is smaller than 0. In data mining, there are several types of attributes, such as categorical type, continuous type. Therefore rule discovery hybrid PSO must find a way to solve the categorical type and continuous type. Second main structure of a particle is *Array-of-Operator*. In this step, the types of attributes have different presentation. For continuous attribute, the operator will be either '≥' or '<', it depends on the *j*-th elements; greater than 0 or less than 0. For categorical (nominal or integer) attribute, the operator will be either '=' or '!=', it depends on the *j*-th elements; greater than 0 or less than 0. And third main structure of a particle is *Array-of-Attribute*. The calculation of *Array-of-Attribute* is more complex than *array-*

of-attribute-existence and *array-of-operator*. This happens because the different types of the attributes have different presentation [8].

- Equation (5) is used to solve the integer type

$$V_{org}[n] = ceil\,(v_n * (V_n max - V_n min) + V_n min) \quad (5)$$

- Equation (6) is used to solve the real type

$$V_{org}[n] = v_n * (V_n max - V_n min) + V_n min \quad (6)$$

- Equation (7) is used to solve the nominal type

$$V_{org}[n] = ValArr_n\,(ceil\,(v_n * Count_n)) \quad (7)$$

where the symbols mean:
$V_{org}[n]$: the value for the n-th attribute of particle,
v_n : the n-th value in the particle
$V_n max$: the maximum value of the n-th attribute
$V_n min$: the minimum value of the n-th attribute
$ValArr_n$: the array which is used to save all different nominal values of the n-th attribute
$Count_n$: the total number of different nominal values of the n-th attribute.
$ceil()$: ceiling function

Equation (5) and (6) use real or integer as the type of the n-th attribute, while (7) uses nominal as the type of the n-th attribute.

D. Rule Discovery Hybrid PSO Evaluation (Fitness Function Design)

We use standard form of rule as IF X THEN Y. Next step, we need to determine the evaluation criteria of a discovered business process model, specifically the quality criteria. The quality criteria relates with completeness and preciseness. In this paper, fitness function is calculated because it contains the evaluation criteria of quality of business process model. There are two terms of fitness function which should be known. The fitness will be high if all activities correctly describes in the event log. Otherwise, the fitness will be low if many activities do not correctly describe in the event log.

In this paper, the number of traces which are correctly parsed from event log is related with fitness function. This term can not be applied in noisy situation because process model can not correctly parse all traces in the event log. The fitness of a process model can be 100% or 1 in a noise-free situation, this is because all traces in the event log from a process model can be parsed. The value of fitness ranges from 0 to 1. Fitness 1 means all traces in the event log from a process model can be parsed, while fitness 0 means no trace in the event log from process model can be parsed by discovered business process model. We can calculate the fitness function using (8) [9].

$$Fitness = 40\% * \frac{APA}{NOAAEL} + \quad (8)$$

$$60\% * \frac{APCLT}{NOTAEL}$$

where:
$NOAAEL : number\ of\ activities\ at\ event\ log$
$NOTAEL : number\ of\ traces\ at\ event\ log$
$APA : all\ Parsed\ Activities$
$APCLT : all\ properly\ completed\ log\ traces$

Algorithm 1: Rule_Discovery_Hybrid_Particle_Swarm_Optimization
Input: EL - event log, $nPart$ – the number of particle, $T0$ – the initial temperature, mIt – the maximum iterations, $bPart$ – the number of best particle
Output: OPT_{model} – the optimal discovered process model
start
$ControlStructures$ = **Recognize_Control_Structures** (EL)
Y = **Initial_Part**($nPart, EL, ControlStructures$)
Y = **Choose_Best_Part**($Y, bPart$
$nPart = bPart$
OPT_{model}= **Initial_GBest**(Y)
n =1
while (**not Stopping_Criteria** (mIt))
do
LB = 0
for n = 1 **to** $nPart$ **do**
$p[n]$ = **Update_the_Value_of_Particle** ($p[n]$)
$T = T0$
while (**not Stopping_Criteria_of_SA** ())
do
X = **Adjust_the_Structural_Strategy**($p[n].X$)
if QF($p[n].X$) < **QF**(X) **then** $p[n].X = X$
else $p[n].X$ = **Determine_the_Probability** ($T.p[n].X, X$)
T = **Update**(T)
end while
$p[n]$ = **Filter**($p[n]$)
$p[n]$ = **Update_the_Value_of_LBest**($p[n]$)
OPT_{model} =
Update_the_Value_of_GBest ($p[n], OPT_{model}$)
end for
end while
return OPT_{model}
Rule_Discovery_Tasks (OPT_{model})
if (*array-of-Attribute* > 0)
then the n-th attribute is used in rule antecedent
else the n-th attribute is not used in rule antecedent
array-of-Attribute:
$V_{org}[n] = v_n * (V_n max - V_n min) + V_n min$
if (*array-of-Operator* > 0)
then '≥' , **else** '<' (continuous type)
if (*array-of-Operator* > 0)
then '=' , **else** '! =' (categorical type)
IF THEN rules
return OPT_{model}
end

IV. Rule Discovery Hybrid PSO Algorithm

We present the algorithm of Rule Discovery Hybrid PSO in Algorithm 1 which uses an event log (EL), the maximum iterations (mIt), the number of particles ($nPart$), the initial temperature ($T0$) required in SA method, and the number of best particle ($bPart$) as input. As an output, we get the optimal discovered process model (OPT_{model}). The algorithm does two main steps: an initialization step and an iterative step.

A. Initialization Step

Lines 4-9 in the initialization step explains that from the event log, we extract the activities, then we initialize the particles using causal matrixes which are produced according to the activities [9]. Next, we have a matrix which has the activities from the event log correlated to each column and row. The value of a matrix is 1 or 0. If value of matrix is 1, then there is a causality relation, while if value of matrix is 0, then there is no causality relation. We determine the value of matrix using dependency measure. Then, we generate the In and Out sets of each activity based on a causality matrix for each particle. At last, we get a number $bPart$ of the best particles and the result is used to further process, the iterative step.

B. Iterative Step

Second step of the algorithm is the iterative step (lines 10-38). In this step, we update the value of particles so that the most optimal of discovered business process model can be determined. During iteration, we update the position of each particle and the velocity of each particle continuously. In (1), (2), and (3) we update the position according to its best position at each iteration as well as the velocity according to the best position generated at the swarm level. In this iterative stage, we also apply the simulated annealing (SA) on each particle position. In (4), new particle position is unconditionally accepted if a random number is higher than the value of P, vice versa.

SA will process a particle position. And then a particle position will be defined to remove the incorrect dependency relations of all activities in the event log. We update the local best value of each particle as well as the global best value at the swarm level using the fitness function. If the discovered process model parses all the cases and all the traces in the event log, then the completeness is good. If the discovered process model defines additional behavior of activities which does not appear in the event log at the starting point.

V. Experimental Results

In section V, our experimental results are presented using event log. We use a set of event log of handling a compensation request in an airline to evaluate the rule discovery hybrid PSO algorithm [14]. A part of the event log is shown in Fig. 2 which contains several information, such as the case identification number/ case ID, date, single time stamp, the activities, the resources, and the cost.

This experiments have performed with an initial state of 400 particles. Using large number of particles, because we have a desire to improve the performance of the rule discovery hybrid PSO. We select the top 24 ones from all particles. The top 24 ones are used in the iterative stage of proposed algorithm. Next, we apply the rule discovery, so we get the top 10 ones that meet the criteria of rule discovery. Table I illustrates the experimental results obtained using rule discovery hybrid particle swarm optimization. The average fitness increases with a high execution time. In Table II, a fitness in row 1 is 0.88997665 and the execution time is 21.458 seconds

Case ID	Date	Time Stamp	Activity	Initial Activity	Resource	Cost
1	30-12-2010	23:02	register request	A	pete	50
1	31-12-2010	10:06	examine thoroughly	B	sue	400
1	5/1/2011	15:12	check ticket	C	mike	100
1	6/1/2011	11:18	decide	D	sara	200
1	7/1/2011	14:24	reject request	E	pete	200
2	30-12-2010	11:32	register request	A	mike	50
2	30-12-2010	12:12	check ticket	C	mike	100
2	30-12-2010	14:16	examine casually	F	sean	400
2	5/1/2011	11:22	decide	D	sara	200
2	8/1/2011	12:05	pay compensation	G	ellen	200
3	30-12-2010	14:32	register request	A	pete	50
3	30-12-2010	15:06	examine casually	F	mike	400
3	30-12-2010	16:34	check ticket	C	ellen	100
3	6/1/2011	9:18	decide	D	sara	200
3	6/1/2011	12:18	reinitiate request	H	sara	200
3	6/1/2011	13:06	examine thoroughly	B	sean	400
3	8/1/2011	11:43	check ticket	C	pete	100
3	9/1/2011	9.55	decide	D	sara	200
3	15/1/2011	15:02	pay compensation	G	ellen	200
4	6/1/2011	12:06	register request	A	pete	50
4	7/1/2011	14:43	check ticket	C	mike	100
4	8/1/2011	12:02	examine thoroughly	B	sean	400
4	9/1/2011	15:44	decide	D	sara	200
4	12/1/2011	9:02	reject request	E	ellen	200

Fig. 2. Part of an event of handling an airline compensation request.

TABLE I. A Part of Experimental Result Using Original Hybrid PSO Algorithm

x	y	z	T0	Std	Fitness	Time (sec)
0.8	0.5	0.8	3	0.1102	0.8899	21.458
0.8	0.5	0.8	7	0.1215	0.8784	21.458
0.9	0.5	0.8	8	0.1306	0.8693	14.601
0.4	0.5	0.8	3	0.1506	0.8493	15.999
0.4	0.5	0.8	9	0.1457	0.8542	16.107
0.2	0.5	0.8	3	0.1381	0.8618	18.563
0.2	0.5	0.8	10	0.1164	0.8835	16.434
0.5	0.2	0.8	3	0.1737	0.8262	17.797
0.5	0.2	0.8	4	0.1359	0.8640	14.109
0.5	0.5	0.8	15	0.1278	0.8721	18.335
0.5	0.8	0.8	3	0.1073	0.8926	13.944
0.5	0.8	0.8	4	0.1202	0.8797	13.572
0.5	0.8	0.2	7	0.1123	0.8876	16.482
0.5	0.8	0.5	3	0.0931	0.9068	15.925
0.5	0.8	0.5	6	0.1226	0.8773	14.654
0.1	0.2	0.7	5	0.1166	0.8833	13.417
0.1	0.2	0.7	10	0.1177	0.8822	16.306
0.1	0.7	0.2	8	0.1193	0.8806	12.983
0.1	0.7	0.2	5	0.1395	0.8604	14.871
0.1	0.7	0.2	10	0.1203	0.8796	18.071
0.3	0.5	0.2	8	0.1079	0.8920	19.082
0.3	0.5	0.2	5	0.1426	0.8573	17.105
0.3	0.5	0.2	10	0.1569	0.8430	20.667
0.2	0.6	0.2	9	0.1107	0.8892	16.247

TABLE II. A PART OF EXPERIMENTAL RESULT USING RULE DISCOVERY HYBRID PSO ALGORITHM

x	y	z	T0	Std	Fitness	Time (sec)
0.8	0.5	0.8	3	0.1102	0.8899	21.458
0.8	0.5	0.8	7	0.1215	0.8784	21.458
0.9	0.5	0.8	8	0.1306	0.8693	14.601
0.4	0.5	0.8	3	0.1506	0.8493	15.999
0.4	0.5	0.8	9	0.1457	0.8542	16.107
0.5	0.5	0.8	15	0.1278	0.8721	18.335
0.5	0.8	0.8	3	0.1073	0.8926	13.944
0.5	0.8	0.8	4	0.1202	0.8797	13.572
0.5	0.8	0.5	3	0.0931	0.9068	15.925
0.5	0.8	0.5	6	0.1226	0.8773	14.654

The final rules discovered from the event log was listed in Table III. The results show the accuracy of our proposed method is good as well as the number of IF-THEN rule is short enough. Moreover in the rule set, we just have a few rules in the end of process, so the rules are easy to understand.

TABLE III. THE RESULTS OF LEARNING FROM EVENT LOG

Class	Rule	Particle Value		
1	IF A>= 0.5 T=3 then class 1	0.8 0.5 0.5	0.5 0.8 0.8	0.8 0.8 0.5
2	IF A>= 0.5 T=7 then class 2	0.8	0.5	0.8
3	IF A>= 0.5 T=8 then class 3	0.9	0.5	0.8
4	IF A>= 0.4 T= 3 then class 4	0.4	0.5	0.8
5	IF A>= 0.4 T= 9 then class 5	0.4	0.5	0.8

The final experiments in Table IV give comparative analysis of average fitness, maximum number of iteration, and total execution time of each algorithm. We compare the classical PSO, original Hybrid PSO and Rule discovery hybrid PSO. We can analyze that the proposed method has the best results in terms of average fitness and number of iterations, compared with classical PSO algorithm and original hybrid PSO algorithm, but still has the highest total execution time. Meanwhile the classical PSO algorithm has the shortest execution time, but still provides the lowest average fitness.

TABLE IV. COMPARATIVE ANALYSIS USING VARIOUS DIFFERENT PSO ALGORITHM

Algorithm	Average Fitness	Time (sec)	Number of iteration
Classical PSO	0.814	7.22275	24
Original Hybrid PSO	0.873	16.591275	24
Rule Discovery Hybrid PSO	0.887	16.6107	10

VI. CONCLUSIONS AND FUTURE WORK

We propose a rule discovery hybrid PSO algorithm for business process mining. In this algorithm, we have to define the particle as an agent and the velocity of the particle in order to apply PSO algorithm. An agent has an important role as a discovered business process model. Meanwhile the velocity of particle can improve the result of process model. In terms of business process mining context, the formulas of PSO algorithm must be redefined which are suitable for optimization problem in context of process mining. We use Simulated Annealing (SA) for finding a good solution to an optimization problem, which uses probabilistic technique. To be formulated into an optimization problem, we use rule discovery task to get the high accuracy, comprehensibility and generalization performance.

Next step, we need to determine the evaluation criteria of a discovered business process model, specifically the quality criteria. The quality criteria relates with completeness and preciseness. In this paper, fitness function is calculated because it contains the evaluation criteria of the quality of discovered business process model. The proposed method, rule discovery hybrid PSO, original hybrid PSO and classical PSO are compared to analyze which method can produce a better result. The final result of rule discovery hybrid PSO is 0.887 for average fitness, 16.6107 seconds and 10 iterations. For our future work, the improvement of our proposed rule discovery hybrid PSO algorithm is very open to be developed in better way, such as reduce its higher total execution time and use more complex event logs as one of evaluation criteria.

ACKNOWLEDGMENT

Authors would like to thank the Department of Informatics, Faculty of Information Technology, Institut Teknologi Sepuluh Nopember for supporting this research.

REFERENCES

[1] R. Sarno, Y. Effendi, and F. Haryadita, “Modified Time-Based Heuristics Miner for Parallel Business Processes,” International Review on Computers and Software (IRECOS), vol. 11 (3), pp. 249-260, March 2016. http://doi.org/10.15866/irecos.v11i3.8717

[2] R. Sarno, E.W. Pamungkas, D. Sunaryono, and Sarwosri, “Business process composition based on meta models,” International Seminar on Intelligent Technology and Its Applications (ISITIA) (Year of Publication: 2015). http://doi.org/10.1109/isitia.2015.7219998

[3] R. Sarno, and K. Sungkono, “Hidden Markov Model for Process Mining of Parallel Business Processes,” International Review on Computers and Software (IRECOS), 11 (4), pp. 290-300. 2016. http://doi.org/10.15866/irecos.v11i4.8700

[4] R. Sarno, W. A. Wibowo, Kartini, and F. Haryadita, “Determining Model Using Non-Linear Heuristics Miner and Control-Flow Pattern,” TELKOMNIKA (Telecommunication Computing Electronics and Control), vol. 14 (1), 2016. http://doi.org/10.12928/telkomnika.v14i1.3257

[5] R. Sarno, R. D. Dewandono, T. Ahmad, M. F. Naufal, and F. Sinaga, “Hybrid Association Rule Learning and Process mining for Fraud Detection,” IAENG International Journal of Computer Science, 42 (2), pp. 59-72, 2015.

[6] V.R. Chifu, C.B. Pop, I. Salomie, I. Balla, and R. Paven, “Hybrid Particle Swarm Optimization Method for Process Mining,” 8th International Conference on Intelligent Computer Communication and Processing, pp.273-279. 2012. http://doi.org/10.1109/ICCP.2012.6356199

[7] J. Kennedy, and R. Eberhart, Particle Swarm Optimization, Proceedings of the International Conference on Neural Networks, Volume 4, pp. 1942 - 1948, 1995.

[8] Y. Liu, Z. Qin, Z. Shi, and J. Chen, “Rule Discovery with Particle Swarm Optimization” AWCC 2004, LNCS 3309, pp. 291–296, 2004.

[9] A. K. Alves de Medeiros, Genetic Process Mining, PhD Thesis, 2006.

[10] W. Song, S. Liu, and Q. Liu, “Business Process Mining Based on Simulated Annealing”, Proceedings of the 9th International Conference for Young Computer Scientists, pp. 725 - 730, 2008.

[11] W.M.P. van der Aalst, A.J.M.M. Weijters, and L. Maruster, Workflow Mining: Discovering Process Models from Event Logs, IEEE Transactions on Knowledge and Data Engineering, Volume 16, Issue 9, pp. 1128 1142, 2004.

[12] D. Floreano, and C. Mattiussi, Bio-Inspired Artificial Intelligence: Theories, Methods, and Technologies, The MIT Press, 2008.

[13] S. Kirkpatrick, C.D. Gelatt, and M. P. Vecchi, Optimization by simulated annealing, Science, Volume 220, Number 4598, pp. 671-680, 1983.

[14] W.M.P. van der Aalst, Process Mining: Discovery, Conformance and Enhancement of Business Processes, Springer Verlag, 2011.