International Review on Computers and Software (I.RE.CO.S.), Vol. 11, N. 4
ISSN 1828-6003 *April 2016*

Hidden Markov Model for Process Mining of Parallel Business Processes

R. Sarno, Kelly R. Sungkono

Abstract – *One of all the works on process mining is the process discovery which produces a representation of a parallel business process. This representation is called process model and it consists of sequence and parallel control-flow patterns. The parallel control-flow patterns contain XOR, AND, and OR relations. Hidden Markov Model is rarely used to represent a process model since XOR, AND and OR relations are not visible. In Hidden Markov Model, the control-flow patterns are represented by probabilities of state transitions. This research proposes an algorithm consisting in a process discovery based on Hidden Markov Model. This algorithm contains equations and rules: the equations are used to differentiate XOR, AND, and OR relations, while the rules are used to establish the process model utilizing detected control-flow patterns. The experiment results show that the proposed algorithm obtain the right control-flow patterns in the process model. The paper demonstrates that the fitness of process models obtained by the proposed algorithm are relatively higher respect to those obtained by Heuristics Miner and Time-based Heuristics Miner algorithms. This paper also shows that the validity of process models obtained by the proposed algorithm are better than those obtained by other algorithms.* ***Copyright © 2016 Praise Worthy Prize S.r.l. - All rights reserved.***

***Keywords**: Fitness, Hidden Markov Model, Parallel Business Process, Process Mining, Validity*

Nomenclature

$\sum_{k=1}^{N} a_{ij} = 1$	Sum of probabilities of transitioning from state i to other states is equal to 1
$\sum_{k=1}^{N} b_j(k) = 1$	Sum of probabilities of observations depend on state j is equal to 1
$\sum_{i=1}^{N} \pi_i = 1$	Sum of probabilities of states as initial states is equal to 1
π	A vector of initial states probabilities
π_i	Probability of state i as an initial state
a	Number of activities in event log
a_{ii}	Probability of transitioning from state i to itself state
a_{ij}	Probability of transitioning from state *i* to state *j*
$a_{ij} \neq 0$	Probability of transitioning from state i to state j is not equal to zero
a_t	Number of accurated traces (number of traces which are appropiate with process model)
average a_{ij}	Average of probabilities of transitioning from state *i* to other states
average $b_i(k)$	Average of probabilities of observations k depend on state i
avgPP	Average of PP as a threshold for differentiating parallel control-flow patterns
A	A matrix of probabilities of state transitions
AND	AND relation
$b_i(k) \neq 0$	Probability of observation k depends on state i is not equal to zero
$b_j(k)$	Probability of observation *k* depends on state *j*
B	A matrix of probabilities of observation dependencies toward states
$Dm(x)$	The percentage of traces of event log x depicted in process models (Depiction Measure of event log x)
ea	Number of excessing activities at the endpoint of the trace
HMM	Hidden Markov Model
ma	Number of missing activities from the trace
minPP	Minimal value of PP as a threshold for differentiating parallel control-flow patterns
M	Number of observations depend on state *i*
$n(b_i(k))$	Number of observations k depend on state i
N	Number of states
O	Observations of Hidden Markov Model
OR	OR relation
$PP(S_i)$	Positive value of state i as input of equation avgpp and minpp
q_t	State at time = *t*

Copyright © 2016 Praise Worthy Prize S.r.l. - All rights reserved

DOI: 10.15866/irecos.v11i4.8700

Relation Measure (S_i)	Relation measure of state S_i as a parameter to differentiate the parallel control-flow patterns
S	States of Hidden Markov Model
S_t	State at this time
S_{t+1}	State at a later time
$S_t(O)$	Observations depend on state at this time
$S_{t+1}(O)$	Observations depend on state at a later time
$SDm(x)$	The percentage of activities in traces of event log x depicted in process models (Specific Depiction Measure of event log x)
t	In equations $Dm(x)$, it means number of traces but in other equations, it means time
XOR	XOR relation

I. Introduction

A business process is examined by a technique called process mining. The business process is formed by activities with multiple purposes [1]. Process mining contains several required works. Process discovery is one of these required works and it analyzes the business process by extracting current activities of organization in event logs [2]. The goal of process discovery is to depict the most effective activities of the organization in a process model. Process model constitutes a guidance to verify and analyze the performance of the present business process, which can change overtime in a large scale of applications [3].

It can also be a guidance to solve the complexity of issues in activities. These issues occur in any fields, e.g. business [4], environment [5], [6], smartphone [7], and fraud [8], [9]. Not all the business process activities are executed sequentially. Some activities can be executed parallelly. A parallel business process consists of activities which are executed sequentially and parallelly. A process model has sequence and parallel control-flow patterns to present the parallel business process.

The control-flow patterns are the forms which express the ordering of activities [10]. The sequence control-flow pattern is used if the next order of an activity is only one activity, whereas the parallel control-flow pattern is used if the next order of an activity is composed by more than one activity. The parallel control-flow patterns are XOR relation, OR relation, and AND relation [11], [12]. XOR relation selects exactly one activity (event) to be executed. OR relation selects one or multiple activities to be executed. AND relation permits some activities to be executed parallelly. Besides sequence and parallel, there are non-free choice relations. They occur when an activity is executed after a certain activity which gets a parallel control-flows patterns. [13] is concerned with determining non-free choice relation using decision mining.

There are several representations of process model, e.g. Petri Net, Business Process Model and Notations (BPMN), YAWL, and Hidden Markov Model. None of all the existing process discoveries, e.g. Alpha, Alpha++ [10], and Heuristics Miner [14] algorithms, use Hidden Markov Model to depict its process model.

This is because the parallel control-flow patterns of Hidden Markov Model are not visible. Hidden Markov Model represents its control-flow patterns by probabilities of its state transitions.

The sequence control-flow pattern appears if the state has transition with one state, and the parallel control-flow pattern appears if the state has transition with more than one state. With only relying on the probabilities of state transitions, it is difficult to differentiate XOR, AND and OR relations as the parallel control-flows pattern.

Therefore, this research concludes that the parallel control-flows pattern of Hidden Markov Model is categorized as invisible. Several researches have exploited Hidden Markov Model for solving their problems, e.g. [15]-[20].

[15], [16] have exploited Hidden Markov Model in process mining. However, no research utilized Hidden Markov Model for discovering parallel business process.

This research proposes an algorithm to obtain a process model of parallel business process based on Hidden Markov Model. The proposed algorithm utilizes Baum-Welch method and double time-stamped event log. This research proposes new equations of the proposed algorithm to differentiate XOR, AND, and OR relations, which is a tough task for Alpha, Alpha++ and Heuristics Miner algorithms. This research also proposes rules to establish the process model by determining activity relations and utilizing sequence and parallel control-flow patterns which are results of the proposed equations.

In addition to discovering business process, the evaluation of discovered process model is also important. Validity and fitness are quality measurements of discovered process models [14]. This research compares discovered process models obtained by the proposed algorithm with others algorithm. The comparison is based on the validity and the fitness.

This research is constructed as follows: Section II presents Hidden Markov Model for Activity Relation Determination. This section also reviews equations to evaluate the validity and the fitness of a discovered process model. Section III describes the proposed algorithm. Section IV reports the steps and final outputs of the experiment process. The last section, Section V, presents conclusions of the research.

II. Research Method

This section contains an explanation of Hidden Markov Model for Activity Relation Determination as the basis of the proposed algorithm. It also contains the equations to calculate the validity and the fitness, as the quality measurements of discovered process model.

Hidden Markov Model for Activity Relation Determination utilizes a collection of event logs having a double time-stamped. The double time-stamped are a start time and an end time of activities in a event log.

They are used to classify observations of each state.

II.1. Hidden Markov Model of Activity Relation Determination

Hidden Markov Model is a combination of two probability distribution processes wherein one of them is hidden. A hidden process can only be determined through another process that produces a sequence of observations [21]. Each observation depends on the states in hidden process. Hidden Markov Model can also be interpreted as Markov Model wherein its proper states are not directly observed [15].

Mathematical model of Hidden Markov Model is HMM = (S, O, A, B, π). S declares a restricted group of hidden states and O declares a restricted group of observations.

A contains many probabilities of the state transitions. Each state transitions to one or many states, either itself or others. The verb in this sentence is missing. Please correct it. However, the total of probabilities of transitioning from a state to other states must be 1. A is defined in Eq. (1):

$$a_{ij} = P\left(q_{t+1} = S_j \mid q_t = S_i\right), 1 \le i, j \le N, \sum_{i=1}^{N} a_{ij} = 1 \quad (1)$$

B contains many probabilities of the observations which depend on certain states. Each observation can depend on one or many states. However, the total of probabilities of a state with its observations must be 1.

B is defined in Eq. (2):

$$b_j\left(k\right) = P\left(O_k \mid q_t = S_i\right), 1 \le i \le N, 1 \le k \le M, \\ \sum_{k=1}^{N} b_j\left(k\right) = 1 \quad (2)$$

Hidden Markov Model has one or many initial states. The probabilities of these initial states are stored in a vector symbolized by π.

The total of probabilities of the initial states must be 1. π is defined in Eq. (3):

$$\pi_i = P\left(q_1 = S_i\right), 1 \le i \le N, \sum_{i=1}^{N} \pi_i = 1 \quad (3)$$

This research uses the Hidden Markov Model, which has Petri Net model as its reference. Petri Net model connects some places with some transitions forming a two-way graph. Each place has one or more transitions which occur if their places are traversed [10]. Each transition contains an activity of the event log.

This research determines the places as states and activities with one additional activity at the endpoint of a trace as observations in the Hidden Markov Model.

The concept expressed above is very confusing because of a bad english construction. Please try to express it more clearly.

The trace means a thread of activities in event log. The additional activity is symbolized by *e*. Petri Net defines that the transitions depend on the same place if only one of the transitions occurs in each trace. These transitions are called choice transitions. Hidden Markov Model adapts the place as a state and adapts the transitions as the obervations. Differently by Petri Net, the observations of Hidden Markov Model depend on the same state if they are indicated as choice observations or they have *time-overlap* with others.

The observations are indicated as choice observations if they have the same order of the execution in their traces. If an observation has more than one order of the execution, then the biggest order is used.

The observations have *time-overlap* with others if they have the same start time or they have the same end time or the time span of observations is met.

Since there is no prior knowledge of the state transition probabilities, an initial guess of transition probability for each state in Hidden Markov Model is determined by 1. The initial guess of observation probabilities and the initial state probability distribution are mined from reference traces. Reference traces are obtained from stored event logs. This research uses 20% of traces in the event log as the reference traces.

The common overview about Hidden Markov Model for activity relation determination is presented in Fig. 1.

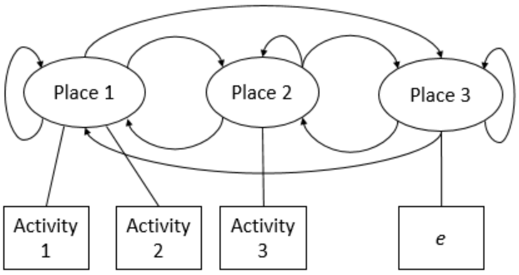


Fig. 1. Hidden Markov Model for Activity Relation Determination

II.2. The Validity and the Fitness of Discovered Process Models

The quality of process discovery can be assessed by evaluating their discovered outputs, i.e. process models.

The evaluation of the process discovery refers to compare discovered process models from different algorithms. The final decision determines the best algorithm among all the algorithms [14].

Process mining reports have not yet defined the agreed algorithm to evaluate the process models. However, the validity and the fitness are already defined as the domain to evaluate process models. The fitness can be obtained by calculating the percentage of recognized traces included in the discovered process model. The high value of fitness of discovered process model shows that many traces are depicted, whereas the low value of fitness of discovered process model shows that few traces are depicted [14].

The validity can be obtained by calculating correctness statements included in the discovered process model [14]. The statements are activities and their relations. A process model is valid if it has all the correctness statements, and vice versa.

Reference [14] shows that there are some equations for calculating the fitness of process model. This research calls the equations as Depiction Measure (*Dm*) and Specific Depiction Measure (*SDm*). Dm is the percentage of traces depicted in process models and SDm is the percentage of the activities in traces depicted in the process models. *Dm* is obtained by dividing the accurate traces with the total of traces and SDm is obtained by dividing the accurated activities with the total of activities in traces. The *Dm* is defined in equation (4) while the SDm is defined in Eq. (5):

$$Dm(x) = \frac{at}{t} \tag{4}$$

$$SDm(x) = \frac{1}{2}\frac{(a-ma)}{a} + \frac{1}{2}\frac{(a-ea)}{a} \tag{5}$$

In reference [14], the discovered process model is categorized as a valid process model if every statement in this process model is the same with a reference process model containing valid statements.

To determine the validity of discovered process model, this research uses Causal Nets as comparison tool. Causal Nets represent the activities with a set of input binding as leading activites and a set of output binding as following activites [22]. Causal Nets are chosen because they determine the complete statements of process models by defining the leading and the following activities for each activity in the process models. This research generates the Causal Nets of the reference process model from event log without noise.

III. Proposed Method

This section explains the details of the proposed algorithm. It has 3 steps: predicting the probabilities of Hidden Markov Model, determining the sequence relation, the parallel relation and the loop condition, and establishing the process model.

III.1. Predicting the Probabilities of Hidden Markov Model

Baum-Welch method is employed to predict the probabilities of Hidden Markov Model. Baum-Welch method is a combination of *forward* method and *backward* method to determine new Hidden Markov Model (*An*, *Bn*, *πn*).

This new model has the maximum probabilities given by the observation sequences. According to [23], Baum-Welch method needs Hidden Markov Model (A, B, π) as the initial model and observation sequences. The initial model of Baum-Welch method is formed according to explanation in Section II.1.

This research gives a simplified example. There are some traces, i.e. ABD and ACD, which have been observed. The ABD appears 4 times and the ACD appears 6 times. Considering these traces, the research discovered four events, such as A, B, C, D. These events are the observations in the Hidden Markov Model.

Under Section II.1, the choice observations or the *time-overlap* observations are observed. The choice observations are event B and event C because they are equally executed as the second event. The time-overlap observations are not found because no events have *time-overlap* with others. Considering the choice observations and an additional activity in the end (*e*), Hidden Markov Model of this simplified example has four states.

They are Place 1 (P1), Place 2 (P2), Place 3 (P3), and Place 4 (P4). P1 has event A as its observation. P2 has event B and event C as its observations. P3 has event D as its observation. P4 has *e* as its observation. By following the steps of Baum-Welch method in [19], a new Hidden Markov Model of simplified example (*An*, *Bn*, *πn*) is displayed in Table I until Table III.

TABLE I
THE INITIAL STATE PROBABILITY
OF NEW HIDDEN MARKOV MODEL

πn	
State	Probability
P1	1

TABLE II
THE STATE TRANSITION PROBABILITY MATRIX
OF NEW HIDDEN MARKOV MODEL

An		
State at this time	State at a later time	Probability
P1	P2	1
P2	P3	1
P3	P4	1

TABLE III
THE OBSERVATION PROBABILITY MATRIX
OF NEW HIDDEN MARKOV MODEL

Bn		
State	Observation	Probability
P1	A	1
P2	B	0.4
P2	C	0.6
P3	D	1
P4	*e*	1

III.2. Determining the Sequence Relation, Parallel Relation and Loop Condition

There are two types of relations in the process model: sequence relations and parallel relations. Sequence relations appear when the states have dependency relations with one state, while parallel relations appear when the states have dependency relations with more than one state.

There are conditions wherein the states have no parallel relations even though they have dependency relations with more than one state.

The conditions are when the states have dependency relations with themself or when the states have dependency relations with their previous state.

This condition is called loop conditions. In many researches like [14], [24], [25], the loop condition is divided in 2 parts: length one loop and length two loop. Length one loop appears when the state has dependency relation with itself and length two loop appears when the state has dependency relation with its previous state. The loop conditions are classified as sequence relations in process model.

The loop conditions can be considered as noise if they appear low in event logs [14]. Similarly to what expressed above there are KK activity as length one loop in five traces. These traces in the event log out of the nine hundred traces are indicated as noise because they appear low in this event log. Their appearance is 0.6% of all the traces. Before classifying sequence relations and parallel relations, this research proposes the equations to filter dependency relations from noise traces and to determine specific parallel relations.

The dependency relations are collected by obtaining the state probabilities.

The proposed equations are PP, minPP, and avgPP. PP (Positive Probability) is a positive value of each state as the input for the calculation of minPP and avgPP. PP is obtained by multiplying the positive state probabilities and their observation probabilities.

MinPP and avgPP are thresholds for determining specific parallel relations. MinPP is a minimal value from all of the PP (Positive Probability). AvgPP is an average value from all of the PP (Positive Probability).

The dependency relations are allowed if the state probabilities are more than minPP. The proposed equations are described in the following:

$$PP(S_i) = \underset{1\le j\le N}{average}\ a_{ij} \times \underset{1\le k\le M}{average}\ b_i(k) \qquad a_{ij} \neq 0, b_i(k) \neq 0 \tag{6}$$

$$avgPP = \underset{1\le i\le N}{average}\, PP(S_i) \tag{7}$$

$$min\,PP = \underset{1\le i\le N}{min}\, PP(S_i) \tag{8}$$

The classification of sequence relations and parallel relations relies on the number of observations for every state. A state has a sequence relation if it has one observation or has loop conditions. A state has a parallel relation if it has more than one observation.

To determine the specific parallel relation, a formula, called relation measure is proposed. The relation measure is used as the parallel determination value for each state and it can be obtained by dividing the state probability with the state probability of another state.

XOR relation occurs if the relation measure is less than or equal to minPP. AND relation occurs if the relation measure is more than or equal to avgPP. OR relation occurs if the relation measure is between minPP and avgPP. The following equations are obtained:

$$\text{Relation Measure}(S_i) = \frac{a_{ii}}{a_{ij}} \times \frac{1}{n(b_i(k)) - 1}, \qquad 1 \le j \le N, a_{ij} \neq 0 \tag{9}$$

$$\text{if Relation Measure}(S_i) \le min\,PP \text{ then } XOR \tag{10}$$

$$\text{if } min\,PP < \text{Relation Measure}(S_i) < avgPP \text{ then OR} \tag{11}$$

$$\text{if } avgPP \ \le \text{Relation Measure}(S_i) \text{ then AND} \tag{12}$$

According to the example in Section III.1, this research determines sequence relations and parallel relations based on the observation probability matrix in Table III. Considering Eq. (6), the Positive Probability (PP) of P2 is calculated as follows:

$$\begin{aligned} P2 &= An(P2P3) \times \text{avg}(An(P2B), An(P2C)) \\ &= 1 \times \text{avg}(0.4, 0.6) \\ &= 0.12 \end{aligned}$$

By performing the same equation, the PP of P1, P3 and P4 are 1. Considering equation (7) and equation (8), minPP is 0.12 and avgPP is 0.71.

Because all the state probabilities in Table II are more than minPP, the dependency relations of all the states are allowed. State P1, P3 and P4 are grouped as sequence relations and State P2 is grouped as a parallel relation. Considering Eq. (9) until Eq. (12), the Relation Measure of $P2$ is calculated as shown below:

$$\begin{aligned} P2 &= (An(P2P2)/An(P2P3)) \times (1/(2\text{-}1)) \\ &= 0/1 \times 1 \\ &= 0 \end{aligned}$$

Because the Relation Measure of $P2$ is less than $minPP$, so $P2$ is classified as XOR relation.

III.3. Establishing the Process Model

The dependency relations among activities are determined to establish the process model.

The information of dependency relations contain the start activities, the end activities, and the relation between them. The determination of the dependency relations is relied on the probabilities of state transitions.

The start activities are obtained by the observations which rely on states at this time.

The end activities are obtained by the observations which rely on states at a later time. Finally, the relation is obtained by the following rules, described in Table IV. In accordance with the same example in Section III.1 and Section III.2, there are two dependency relations: P1 to P2 and P2 to P3. This research uses the first, second and sixth step of rules as described in Table IV.

This is because all the states appear once and no dependency relations occur between the same states. P4 contains an additional activity (*e*) so the relation between P3 and P4 is ignored.

The start activities and the end activities of the dependency relations between P1 and P2 are Event A and Event (B, C). The dependency relation is XOR relation because P1 is sequence relation and P2 is XOR relation. The start activities and the end activities of the dependency relations between P1 and P2 are Event (B, C) and Event D. The dependency relation is XOR relation because P3 is sequence relation and P2 is XOR relation. The process model relied on dependency relations is presented in Fig. 2.

IV. Results and Analysis

This experiment uses the double time-stamped event log. Each event log has attributes, such as the case id, the activities, and the time of activity executions. These time are the start time and the finish time.

The business process in this experiment contains 11 activities described in Table V.

TABLE IV
RULES OF DETERMINING ACTIVITY RELATIONS

Number	Rules
1	The start activities are the observations of the state at this time and the end activities are the observations of the state at a later time $(S_{t+1}(O))$.
2	The dependency relation is not included if S_t is same as S_{t+1} while their relation is a specific parallel relation or the end activity is an additional activity.
3	The dependency relation is a sequence relation if S_t is same as S_{t+1} *while their relation* is the sequence relation.
4	The dependency relation is a specific parallel relation if one of the relations of S_t and S_{t+1} is a specific parallel relation.
5	The dependency relation is a sequence relation if the relations of S_t and S_{t+1} are the same specific parallel relations or they are sequence relation.
6	The dependency relation is a specific parallel relation of S_{t+1} if the relation of S_t and the relation of S_{t+1} are different specific parallel relations.
7	If S_t appears more than once, the dependency relation is a relation of S_{t+1} which appears once and the start activities are an observations of S_t which has a closest observation probability with the state probability of S_t and S_{t+1}.
8	If S_{t+1} appears more than once, it merges that the start activities from different S_t and the dependency relation are a specific parallel relation of S_t

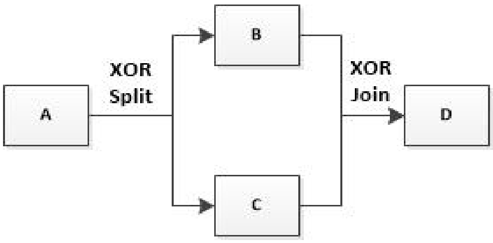


Fig. 2. The process model

TABLE V
THIS RESEARCH ACTIVITY NAMES

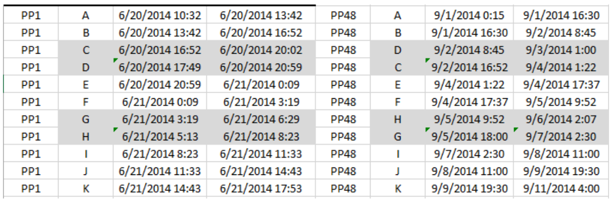
The real name of activities	The name of activities in this research
Getting good receive	A
Bale opening and blending	B
Opossing spike	C
Air current blowing	D
Striking cotton	E
Carding	F
Drawing frame	G
Roving frame	H
Combing	I
Ring framing	J
Cone winding	K

The opposing spike and air current blowing activities are not mutually dependent, so they can be executed parallelly. Thereafter, one or all the drawing frame and roving frame activities are executed.

IV.1. Experiment Data

The event logs as experiment data in Section IV contain 50 traces in four conditions. These conditions are event log without noise, event log with 10% noise, event log with 30% noise and event log with 50% noise.

This section focuses on the event log without noise. The piece of event log used in this experiment is presented in Fig. 3.



PP1	A	6/20/2014 10:32	6/20/2014 13:42	PP48	A	9/1/2014 0:15	9/1/2014 16:30
PP1	B	6/20/2014 13:42	6/20/2014 16:52	PP48	B	9/1/2014 16:30	9/2/2014 8:45
PP1	C	6/20/2014 16:52	6/20/2014 20:02	PP48	D	9/2/2014 8:45	9/3/2014 1:00
PP1	D	6/20/2014 17:49	6/20/2014 20:59	PP48	C	9/2/2014 16:52	9/4/2014 1:22
PP1	E	6/20/2014 20:59	6/21/2014 0:09	PP48	E	9/4/2014 1:22	9/4/2014 17:37
PP1	F	6/21/2014 0:09	6/21/2014 3:19	PP48	F	9/4/2014 17:37	9/5/2014 9:52
PP1	G	6/21/2014 3:19	6/21/2014 6:29	PP48	H	9/5/2014 9:52	9/6/2014 2:07
PP1	H	6/21/2014 5:13	6/21/2014 8:23	PP48	G	9/5/2014 18:00	9/7/2014 2:30
PP1	I	6/21/2014 8:23	6/21/2014 11:33	PP48	I	9/7/2014 2:30	9/8/2014 11:00
PP1	J	6/21/2014 11:33	6/21/2014 14:43	PP48	J	9/8/2014 11:00	9/9/2014 19:30
PP1	K	6/21/2014 14:43	6/21/2014 17:53	PP48	K	9/9/2014 19:30	9/11/2014 4:00

Fig. 3. The piece of event log

IV.2. Process Discovery using Modified Hidden Markov Model

This section describes the steps of process discovery using the proposed algorithm with the event log presented in Section IV.1.

IV.2.1. Predicting the Probabilities of Hidden Markov Model

This research predicts the probabilities of Hidden Markov Model. The observations are made by all the

Copyright © 2016 Praise Worthy Prize S.r.l. - All rights reserved

activities described in the beginning of Section IV. Two *time-overlap* paired events and no event indicated as choice observation were founded after observing 20% traces. The two paired events are event C with event D and event G with event H. Ten states are determined by considering the discoverably paired events.

They are Place 1 (P1) until Place 10 (P10). The observations of each place are described in Table VI. Table VII until Table IX show the Hidden Markov Model using Baum-Welch method.

TABLE VI
THE OBSERVATION OF HIDDEN MARKOV MODEL

State	Observation	State	Observation
P1	A	P6	G,H
P2	B	P7	I
P3	C,D	P8	J
P4	E	P9	K
P5	F	P10	*e*

TABLE VII
THE INITIAL STATE PROBABILITY OF HIDDEN MARKOV MODEL

πn	
State	Probability
P1	1

TABLE VIII
THE STATE TRANSITION PROBABILITY MATRIX
OF HIDDEN MARKOV MODEL

An					
State at this time	State at a later time	Probability	State at this time	State at a later time	Probability
P1	P2	1	P6	P6	0.32
P2	P3	1	P6	P7	0.68
P3	P3	0.5	P7	P8	1
P3	P4	0.5	P8	P8	0.38
P4	P5	1	P8	P9	0.62
P5	P6	1	P9	P10	1

TABLE IX
THE OBSERVATION PROBABILITY MATRIX
OF HIDDEN MARKOV MODEL

Bn					
State	Observation	Probability	State	Observation	Probability
P1	A	1	P6	G	0.5
P2	B	1	P6	H	0.5
P3	C	0.5	P7	I	1
P3	D	0.5	P8	J	1
P4	E	1	P9	K	1
P5	F	1	P10	*e*	1

IV.2.2. Determining the Sequence and Parallel Relation

The avgPP and minPP are calculated by using Eq. (6), Eq. (7) and Eq. (8). The results of avgPP and minPP are 0.75 and 0.25. All the state transitions are allowed because all the state probabilities in Table VI are more than minPP. Under Section III.2, the classified states in sequence relations are P1, P2, P4, P5, P7, P8, P9, P10 and the classified states in parallel relations are P3 and P6. Using Eq. (9), the Relation Measure of P3 is 1 and the Relation Measure of P6 is 0.5. Because the Relation Measure of P3 is more than avgPP and the Relation Measure of P6 is between avgPP and minPP, P3 is classified as AND relation and P6 is classified as XOR relation.

IV.2.3. Establishing the Process Model

Under Section III.3, the dependency relations among activities are determined.

By using the rules in Table IV, the dependency relations are described in Table X.

The discovered process model based on the relations in Table VII is displayed in Fig. 4.

This research also discovers event logs which contain 10% noise, 30% noise and 50% noise with the proposed algorithm. Based on the experiments, the discovered process models from that event logs are the same as the discovered process model from event log without noise displayed in Fig. 4.

IV.3. Process Discovery Using Another Algorithm

In addition to the proposed algorithm, this research discovers the process model using Modified Time-Based Heuristics Miner algorithm [26] and Original Heuristics Miner algorithm [24], [25] with event log from Section IV.1. The discovered process models using Original Heuristics Miner algorithm from event log as described in Section IV.1 are displayed in Figs. 5, 6, 7 and 8.

The discovered process model using Modified Time-Based Heuristics Miner algorithm from event log as described in Section IV.1 are displayed in Figs. 9, 10, 11 and 12.

TABLE X
DEPENDENCY RELATION

State (Before)	State (After)	Start Activities	End Activities	Relation	Rule Numbers
P1	P2	A	B	Seq	1,5
P2	P3	B	C,D	AND	1,4
P3	P4	C,D	E	AND	1,4
P4	P5	E	F	Seq	1,5
P5	P6	F	G,H	OR	1,4
P6	P7	G,H	I	OR	1,4
P7	P8	I	J	Seq	1,5
P8	P8	J	J	Seq	1,3
P8	P9	J	K	Seq	1,5
P9	P10	K	*e*	-	1,2

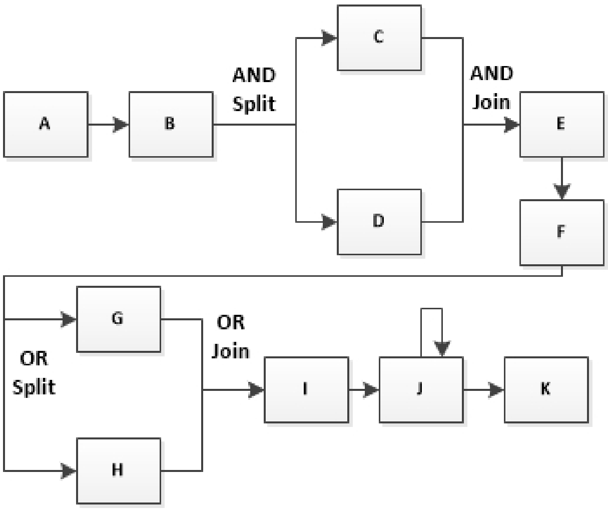


Fig. 4. Discovered Process Model using proposed algorithm

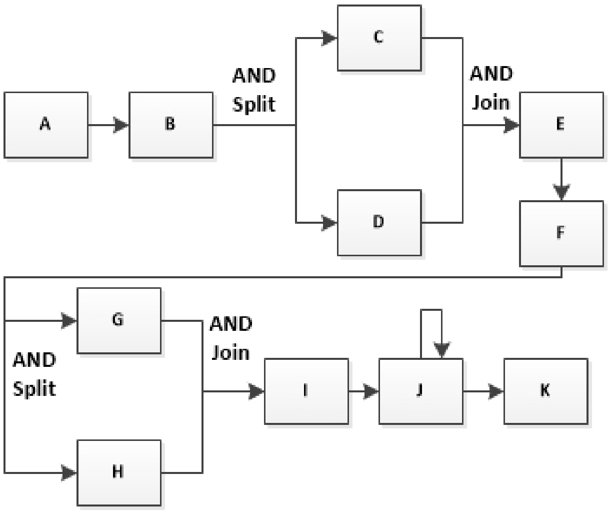


Fig. 5. Discovered Process Model using Original Heuristics Miner algorithm

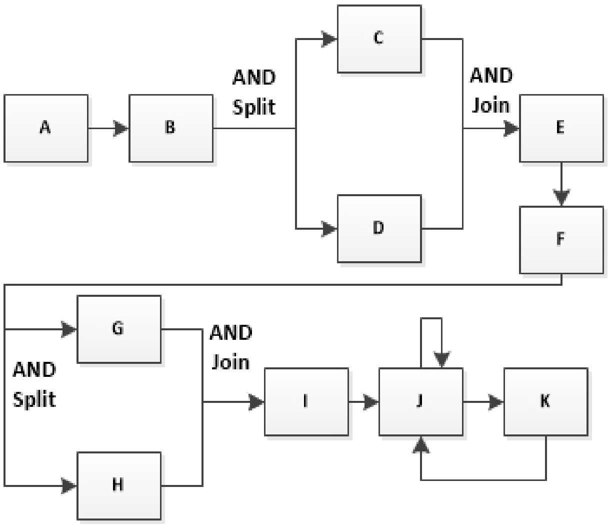


Fig. 6. Discovered 10% Noise Process Model using Original Heuristics Miner algorithm

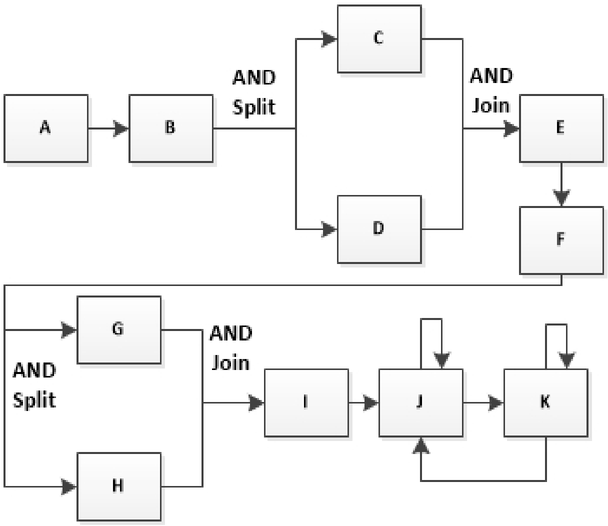


Fig. 7. Discovered 30% Noise Process Model using Original Heuristics Miner algorithm

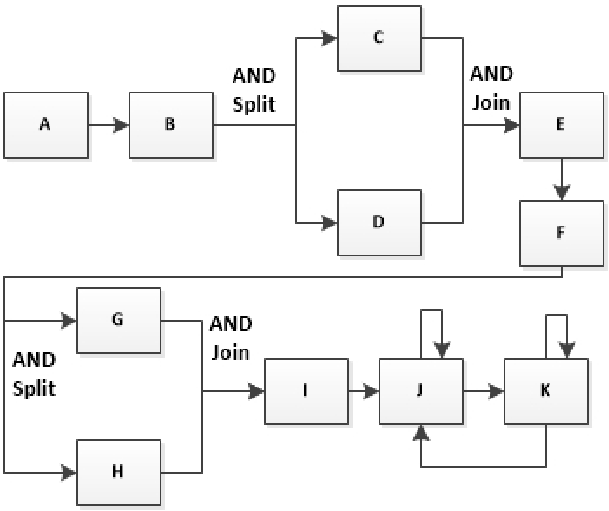


Fig. 8. Discovered 50% Noise Process Model using Original Heuristics Miner algorithm

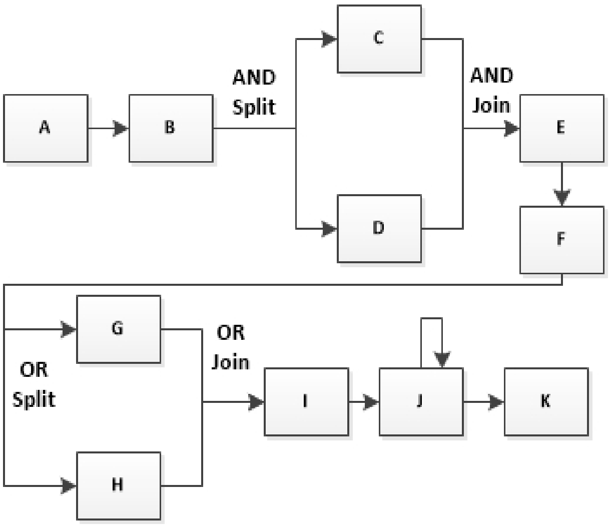


Fig. 9. Discovered No Noise Process Model using Modified Time-Based Heuristics Miner algorithm

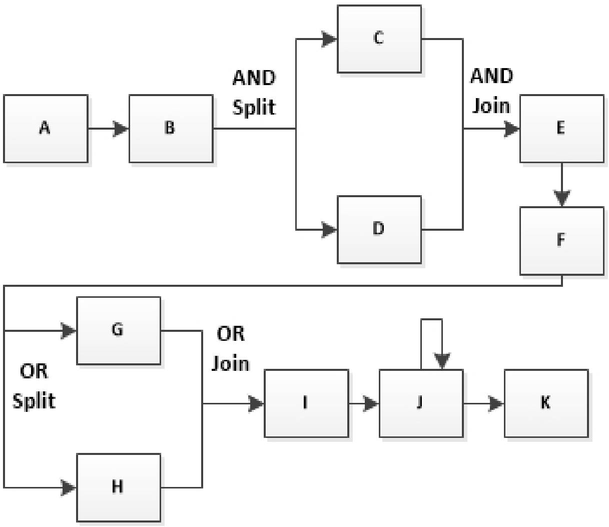


Fig. 10. Discovered 10% Noise Process Model using Modified Time-Based Heuristics Miner algorithm

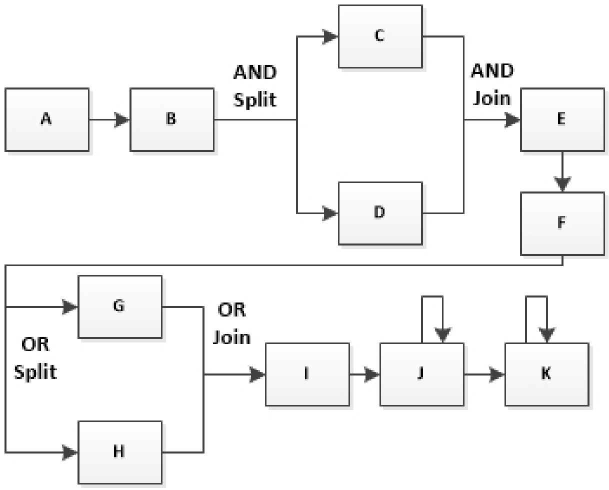


Fig. 11. Discovered 30% Noise Process Model using Modified Time-Based Heuristics Miner algorithm

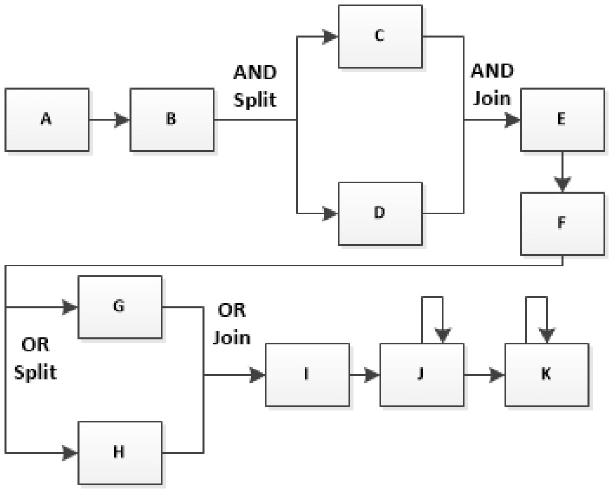


Fig. 12. Discovered 50% Noise Process Model using Modified Time-Based Heuristics Miner algorithm

IV.4. The Fitness and the Validity of Discovered Process Models

Reflecting on the fitness and the validity equations in Section II.2, the discovered process models from three algorithms can be compared.

The three algorithms are the proposed algorithm, Original Heuristics Miner algorithm, and Modified Time-Based Heuristics Miner algorithm.

Based on Depiction Measure in Eq. (4) and Spesific Depiction Measure in Eq. (5), the fitness values for each algorithm are displayed in Table XI and Table XII.

Based on Table XI and Table XII, it can be seen that the average of the difference of the discovered process models obtained by the proposed algorithm and those obtained by Modified Time-Based Heuristics Miner which gain the highest fitness, is 0.0079. Because the difference is less than 1%, it means that the fitness of the discovered process models obtained by this proposed algorithm is categorized in high value eventhough it is not the highest value.

TABLE XI
FITNESS VALUES USING THE PARSING MEASURE

Depiction Measure (Dm)				
Event Log	Method/algorithm	Dm	at	t
50 trace (no noise)	Modified Hidden Markov Model	1	50	50
	Original Heuristics Miner	0.5	25	50
	Modified Time-Based Heuristics Miner	1	50	50
50 trace (10% noise)	Modified Hidden Markov Model	0.9	45	50
	Original Heuristics Miner	0.42	21	50
	Modified Time-Based Heuristics Miner	0.9	45	50
50 trace (30% noise)	Modified Hidden Markov Model	0.7	35	50
	Original Heuristics Miner	0.42	21	50
	Modified Time-Based Heuristics Miner	0.74	37	50
50 trace (50% noise)	Modified Hidden Markov Model	0.5	25	50
	Original Heuristics Miner	0.38	19	50
	Modified Time-Based Heuristics Miner	0.52	26	50

TABLE XII
FITNESS VALUES USING THE CONTINUOUS PARSING MEASURE

Specific Depiction Measure (SDm)					
Event Log	Algorithm	SDm	a	ma	ea
50 trace (no noise)	Modified Hidden Markov Model	1	555	-	-
	Original Heuristics Miner	0.977	555	25	-
	Modified Time-Based Heuristics Miner	1	555	-	-
50 trace (10% noise)	Modified Hidden Markov Model	0.991	554	5	5
	Original Heuristics Miner	0.973	554	30	-
	Modified Time-Based Heuristics Miner	0.991	554	5	5
50 trace (30% noise)	Modified Hidden Markov Model	0.982	539	12	8
	Original Heuristics Miner	0.967	539	42	-
	Modified Time-Based Heuristics Miner	0.983	539	12	6
50 trace (50% noise)	Modified Hidden Markov Model	0.971	543	17	16
	Original Heuristics Miner	0.953	543	53	-
	Modified Time-Based Heuristics Miner	0.973	543	17	14

The Causal Net from event log without noise is built to obtain the validity of discovered process model using three algorithms as described in Section IV.4.

This Causal Net is named Reference Causal Net. The discovered process models using that three algorithms are valid process models if the Causal Net of this discovered process model is similar to the Reference Causal Net.

The Causal Net of discovered process models using three algorithms is built from the event log with 30% noise. All the Causal Nets are displayed in Table XIII until Table XVI. Based on the results in Table XIII until Table XVI, the only Causal Net using the proposed algorithm is similar to the Reference Causal Net.

TABLE XIII
THE REFERENCE CAUSAL NETS

INPUT SET	ACTIVITY	OUTPUT SET
{Ø}	A	{{B}}
{{B}}	B	{{C,D}}
{{B}}	C	{{E}}
{{B}}	D	{{E}}
{{C,D}}	E	{{F}}
{{E}}	F	{{G},{H},{G,H}}
{{F}}	G	{{I}}
{{F}}	H	{{I}}
{{G},{H},{G,H}}	I	{{J}}
{{I},{J}}	J	{{J},{K}}
{{J}}	K	{Ø}

TABLE XIV
THE CAUSAL NETS USING PROPOSED ALGORITHM

INPUT SET	ACTIVITY	OUTPUT SET
{Ø}	A	{{B}}
{{B}}	B	{{C,D}}
{{B}}	C	{{E}}
{{B}}	D	{{E}}
{{C,D}}	E	{{F}}
{{E}}	F	{{G},{H},{G,H}}
{{F}}	G	{{I}}
{{F}}	H	{{I}}
{{G},{H},{G,H}}	I	{{J}}
{{I},{J}}	J	{{J},{K}}
{{J}}	K	{Ø}

TABLE XV
THE CAUSAL NETS USING ORIGINAL HEURISTICS MINER

INPUT SET	ACTIVITY	OUTPUT SET
{Ø}	A	{{B}}
{{B}}	B	{{C,D}}
{{B}}	C	{{E}}
{{B}}	D	{{E}}
{{C,D}}	E	{{F}}
{{E}}	F	{{G,H}}
{{F}}	G	{{I}}
{{F}}	H	{{I}}
{{{G,H}}	I	{{J}}
{{I},{J}}	J	{{J},{K}}
{{J},{K}}	K	{{Ø},{J},{K}}

TABLE XVI
THE CAUSAL NETS USING MODIFIED TIME-BASED HEURISTICS MINER

INPUT SET	ACTIVITY	OUTPUT SET
{Ø}	A	{{B}}
{{B}}	B	{{C,D}}
{{B}}	C	{{E}}
{{B}}	D	{{E}}
{{C,D}}	E	{{F}}
{{E}}	F	{{G},{H},{G,H}}
{{F}}	G	{{I}}
{{F}}	H	{{I}}
{{G},{H},{G,H}}	I	{{J}}
{{I},{J}}	J	{{J},{K}}
{{J},{K}}	K	{{Ø},{K}}

It means that the valid process model for event log with 30% noise is the discovered process model using the proposed algorithm.

By utilizing another event log, the valid process models are shown in Table XVII.

Based on the valid process models in Table XVII, the validity of the discovered process model using the proposed algorithm, Original Heuristics Miner and Modified Time-Based Heuristics Miner are 100%, 0%, and 50% of all event logs. The validity of discovered process models using the proposed algorithm is the highest validity.

TABLE XVII
THE VALID PROCESS MODELS

Algorithms	Valid Process Models
Proposed Algorithm	Process models from all event logs
Original Heuristics Miner	-
Modified Time-Based Heuristics Miner	Process models from the event log without noise and from the event log with 10% noise

After conducting experiments as described in section IV, the advantages of the proposed algorithm are described as follows:

- The proposed algorithm differentiates XOR, AND and OR relations appropriately.
- The proposed algorithm filters noise traces when discover a process model.
- The proposed algorithm increases the validity of the discovered process models than other algorithms.
- The discovered process models of the proposed algorithm have high fitness.

V. Conclusion

This paper has proposed an algorithm based on Hidden Markov Model to discover parallel business process from event logs with noise and without noise.

The proposed algorithm consists of the following steps. First, the probabilities are estimated by Baum-Welch method. Second, the sequence and parallel relations are determined using the proposed equations.

Finally, the parallel process model are established based on the mined dependency relations by using the proposed rules. The evaluation results showed that the proposed algorithm could discover parallel process business containing XOR, AND and OR relations which are represented by the probabilities of state transitions.

The results also described that the fitness of discovered process models obtained by the proposed algorithm are relatively high as those obtained by Heuristics Miner and Time-based Heuristics Miner.

This proposed algorithm is promising because there are only few researches of process mining utilizing Hidden Markov Model.

Acknowledgements

Authors would like to thank the Institut Teknologi Sepuluh Nopember and the Ministry of Research, Technology and Higher Education of Indonesia for supporting this research.

References

[1] R. Sarno, C. A. Djeni, I. Mukhlash, D. Sunaryono, Developing A Workflow Management System for Enterprise Resource Planning, (2015) *Journal of Theoretical and Applied Information Technology*, 72 (3), pp. 412-421.

[2] R. Sarno, B. A. Sanjoyo, I. Mukhlash, H. M. Astuti, Petri Net Model of ERP Business Process Variation for Small and Medium Enterprises, (2013) *Journal of Theoretical and Applied Information Technology*, 54, pp. 412-421.

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

[4] O. T. Baruwa, M. A. Piera, Identifying FMS repetitive patterns for efficient search-based scheduling algorithm: A colored Petri Net approach, (2015) *Journal of Manufacturing Systems,* 35, pp. 120-135.
http://dx.doi.org/10.1016/j.jmsy.2014.11.009

[5] A. Sanaa, S. B. Abid, A. Boulila, C. Messaoud, M. Boussaid, N. B. Fadhel, Modelling hydrochory effects on the Tunisian island populations of Pancratium maritimum L. using colored Petri Nets, (2015) *BioSystems,* 129, pp. 19-24. http://dx.doi.org/10.1016/j.biosystems.2015.02.001

[6] J. Yuan, D. Oswald, W. Li, Autonomous tracking of chemical plumes developed in both diffusive and turbulent airflow environment using Petri Nets, (2015) *Expert Systems with Application,* 42, pp. 527-538. http://dx.doi.org/10.1016/j.eswa.2014.08.005

[7] V. R. L. Shen, H.-Y. Lai, A.-F. Lai, The implementation of a smartphone-based fall detection system using a high-level fuzzy Petri Net, (2015) *Applied Soft Computing,* 26, pp. 390-400. http://dx.doi.org/10.1016/j.asoc.2014.10.028

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

[9] Huda, S., Ahmad, T., Sarno, R., Santoso, H.A, Identification of process-based fraud patterns in credit application, *2nd International Conference on Information and Communication Technology (ICoICT)* (Year of Publication: 2014). http://dx.doi.org/10.1109/ICoICT.2014.6914045

[10] W. M. P. van der Aalst, *Process mining: discovery, conformance and enhancement of business processes* (Springer Science and Business Media, 2011). http://dx.doi.org/10.1007/978-3-642-19345-3

[11] W. M. P. van der Aalst, A. H. Ter Hofstede, B. Kiepuszewski, A. P. Barros, Workflow patterns, (2003) *Distributed and Parallel Databases*, 14 (1), pp. 5-51. http://dx.doi.org/10.1023/a:1022883727209

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

[13] Sarno, R., Sari, P.L.I., Sunaryono, D., Amaliah, B., Mukhlash, I., Mining decision to discover the relation of rules among decision points in a non-free choice construct, *Proceedings of International Conference on Information, Communication Technology and System (ICTS)* (Year of Publication: 2014). http://dx.doi.org/10.1109/ICTS.2014.7010557

[14] S. De Cnudde, J. Claes, G. Poels, Improving the Quality of the Heuristics Miner in Prom 6.2, (2014) *Expert System with Application*, 41, pp. 7678-7690. http://dx.doi.org/10.1016/j.eswa.2014.05.055

[15] A. Rozinat, M. Veloso, W. M. P. van der Aalst, Using Hidden Markov Models to Evaluate the Quality of Discovered Process Models, (2008) *BPM Center Report BPM-08-10, BPMcenter.org.*

[16] Da Silva, G.A., Ferreira, D.R., Applying Hidden Markov models to process mining, *Sistemas e Technologias de Informacao: Actas da 4a. Conferencia Iberica de Sistemas e Technologias de Informacao, AISTI/FEUP/UPF* (Page: 207-210 Year of Publication: 2009).

[17] D. Regan, S. Srivatsa, Mixed Pixel Wise Characterization Based on HMM and Hyper spectral Image Gradient Enhancement for Classification Using SVM-FSK, (2014) *International Review on Computers and Software (IRECOS)*, 9 (6), pp. 1017-1026.

[18] H. El Moubtahij, A. Halli, K. Satori, Arabic Handwriting Text Offline Recognition Using the HMM Toolkit (HTK), (2014) *International Review on Computers and Software (IRECOS),* 9 (7), pp. 1214-1219.

[19] M. Rani, A. Marimuthu, A. Kavitha, Artificial Fish Swarm Load Balancing and Job Migration Task with Overloading Detection in Cloud Computing Environments, (2014) *International Review on Computers and Software (IRECOS),* 9 (4), pp. 727-734.

[20] A. Maqqor, A. Halli, K. Satori, H. Tairi, Offline Arabic Handwriting Recognition System Based on the Combination of Multiple Semi-Continuous HMMs, (2015) *International Review on Computers and Software (IRECOS)*, 10 (7), pp. 677-683. http://dx.doi.org/10.15866/irecos.v10i7.6229

[21] Rabiner, L.R., A tutorial on Hidden Markov Models and selected applications in speech recognition, *Proceedings of the IEEE 77* (Issue: 2 Page: 257-286 Year of Publication: 1989). http://dx.doi.org/10.1109/5.18626

[22] W.M.P. van der Aalst, A. Adriansyah, B. Van Dongen, Causal Netss: A Modeling Language Tailored towards Process Discovery, In J.P.K.B. Konig, *CONCUR 2011-Concurrency Theory* (Berlin: Springer Berlin Heidelberg, 2011, 28-42) http://dx.doi.org/10.1007/978-3-642-23217-6_3

[23] L. Moss, Example of the Baum-Welch Algorithm, (2008) *Q520, Spring*.

[24] A. J. M. M. Weijters, W. M. P. van der Aalst, A. K. Alves de Medeiros, Process mining with the Heuristics-miner algorithm, (2006) *Technische Universiteit Eindhoven, Tech. Rep. WP,* 166, pp. 1-34.

[25] Weijters, A.J.M.M., Ribeiro, J.T.S., Flexible Heuristics Miner (FHM), *Proceedings of IEEE Symposium on Computational Intelligence and Data Mining* (Year of Publication: 2011). http://dx.doi.org/10.1109/cidm.2011.5949453

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

Authors' information

Informatics Departement, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia

R. Sarno received M.Sc and Ph.D in Computer Sience University of Brunswick Canada in 1988 and 1992. His research includes internet of things, enterprise computing, information management, intelligent systems and smart grids. Prof. Riyanarto has served as a faculty member in Informatics Departement in Institut Teknologi Sepuluh Nopember Surabaya. He teaches courses in systems audit, knowledge engineering, enterprise systems, and specific topics on information management.
E-mail: riyanarto@gmail.com

Kelly R. Sungkono was born in Surabaya, Indonesia. She is currently a B.Sc student of Informatics Departement, Institut Teknologi Sepuluh Nopember Surabaya, Indonesia. Her major skills and researches are in process mining. She is also interested in database management.
E-mail: kelsungkono@gmail.com