On-the-Fly Performance-Aware Human Resource Allocation in the Business Process Management Systems Environment using Naïve Bayes

Arif Wibisono, Amna Shifia Nisafani[1]
Information Systems Department, Institut Teknologi Sepuluh Nopember,
Jalan Raya ITS Sukolilo, Surabaya, Indonesia
{wibisono,amna}@is.its.ac.id

Hyerim Bae
Industrial Engineering Department, Pusan National University,
Busandaehak-ro 63 beon-gil, Geumjeong-gu, Busan, Republic of Korea
hrbae@pusan.ac.kr

You-Jin Park
School of Business Administration, Chung-Ang University
84 Heukseok-dong, Dongjak-gu, Seoul, Republic of Korea
eugenepark@cau.ac.kr

Abstract. Traditionally, resource allocation problem has been considered as one of the important issues in business process management to maintain the acceptable level of each activity completion time which can reduce the total completion time. Especially, the complexity of managing resources increases when the resource type is human because performance of each human resource might fluctuate over time due to various unpredicted factors. Hence, upfront planning of the resource allocation might be unsuitable in this matter. Therefore, this study proposes an on-the-fly resource allocation using Naïve Bayes to manage human resources more efficiently. The term on-the-fly here indicates that the resource allocation planning will be frequently updated and executed during the execution time by considering recent human resource performances. In this paper, we will show the proposed approach exceeds other resource allocation approaches in terms of total completion time.

Keywords: on-the-fly resource allocation, machine learning, dispatching rules, resource-based priority rules

1 Introduction

Recent business competitiveness requires every company to remain efficient in order to survive in highly complex business environments. Therefore, more and more

[1] Corresponding author

softwares related to business process management systems (BPMS) are deployed in many companies. Here, BPMS focuses on the planning, execution, control, monitoring, and evaluation of business process (BP) execution to obtain some important efficiency extents. For this reason, several scheduling approaches were introduced to help BPMS to better organize the resources involved in the BP execution [1, 2, 3, 4, 5, 6, 7, 8, 9]

In general, there are two types of resources in BP execution: human and machine [8]. When machine is prevalent in many manufacturing process, human is dominant in numerous organizational processes such as order-to-cash, quote-to-order, procure-to-pay, issue-to-resolution, and application to approval [10]. In terms of resource allocation, machine-related processes are easier to maintain due to the lower variability of the machine performances. In contrast, human performances are oscillating continuously due to the differences of knowledge and physical/emotional conditions. Thus, it is almost impossible to ask human to work in a regular pace during his/her daily work time. To illustrate, a worker might work industriously in his/her first three hours and then the performance systematically declines around the lunch time. After doing lunch, the performance increases; however, it is not as the same pace as in the early morning, and it is getting steady from 3 PM to the end of the work time.

The issue of business process scheduling has received considerable attention among researchers. Zhao and Stohr developed method to reduce the amount of rework in claim handling system [7]. Bae *et al.* [1] proposed a methodology using mixed integer programing (MIP) for BP execution plan by taking into consideration business process semantics and alternative path in the business process management structure. Eder *et al.* [2] built personal schedule to forecast future incoming jobs in which organizations can decrease both the turnaround time and the rate of time-constraint violation.

One of the limitations with the most recent papers in business process scheduling is that they focus on the upfront planning (see Section 2.1 Literature Reviews). Here, upfront planning means that the resource allocation planning has been completely established before the execution takes place. The upfront planning might be unable to fully accommodate resource performance dynamics during execution. Hence, it might fail to allocate right resource(s) in doing an incoming job.

This study aims to develop an on-the-fly performance-aware resource allocation by incorporating Naïve Bayes in the proposed algorithm. To this point, the resource allocation planning will be rigorously updated and (then) executed given the evident of each human resource performances. By doing so, we can enhance the resource performance prediction, thus making a better resource allocation. For this reason, we expect a better human resource allocation in terms of the completion time. Most of the work of this paper including the algorithm basis and the real-world study case is an extension of Nisafani *et al.* [8]. The results of this study prove to be very useful for any process efficiency oriented managers, especially those who are responsible for managing human-intensive processes. The reminder sections of this paper are organized as follows: Section 2 defines the literature reviews, Section 3 explains our proposed approach, Section 4 provides the experiments of our model, and finally, Section 5 presents the conclusions of the research.

2 Literature Reviews

2.1 Resource Allocation in the Business Process Management Systems Environment

Most available methods of the resource allocation in business process have been focusing on the upfront resource allocations planning [11]. Wu *et al.* [6] predicted the future resource behavior using workload dynamics. Huang *et al.* [3] proposed a resource allocation algorithm that utilizes Markov decision process and solved using reinforcement learning. Ha *et al.* [12] introduced process execution rule to fairly distribute workload for each involved resource (which called as agent). Huang *et al.* [4] suggested a method of resource behavior measurement using four crucial factors to improve BP execution namely availability, corporation, preference, and competence.

Most of the studies reviewed so far; however, suffers from the fact that upfront resource planning can be unsuitable to capture the resource performance dynamics (especially whenever the resource type is human). Most of the studies assume that resource performances are uniformly distributed during time horizon, in fact they are not. Here, the forecast accuracy of the resource performances declines heavily as the forecast horizon increases [13]. Hence, it is necessary to consider a run-time oriented and performance-aware resource allocation in business process for improving the system performance. There is a limited study that starts investigating effective methods for on-the-fly performance-aware resource allocation. Nisafani *et al.* [8] simulated the on-the-fly performance-aware resource allocation on a real-world semi-automatic business process and recommended resource allocation algorithm to use Bayesian network (BN) as a model. The BN incorporates several factors in BP execution such as workload, inter-arrival time, daytime, and working hours [8]. The result showed that the existing resource allocation exceeded four resource-based priority rules namely index-ordered, shortest-idled, longest-idled, and random allocation in terms of average completion time, average waiting time, and average cycle time [8].

The weakness of Nisafani *et al.*'s approach is that the employed heuristic-based BN introduces a BN structure which quality has never been measured statistically. At this point, the BN designs the relationship dependencies among factors involved in BP execution such as resource performance, queue, and inter-arrival time. In addition, there is no exact formula to model BN among aforementioned factors thus heuristic approach is used. That's why expert is assumed has a comprehensive understanding about the intertwined factors. Unfortunately, the understanding might be mistaken. For instance, an individual with a highly imposed workload does not basically demonstrate performance decline. In contrast, an individual even though is imposed with long queues does not essentially increase his/her performance. As a result, a statistical measurement is required to appraise the BN structure. Surely, a better BN structure will introduce a better prediction.

It is known that measuring BN structure does not automatically produce a good BN structure, rather, additional time consuming algorithm (such as K2 Algorithm) should be performed later to construct a BN structure. To increase the prediction accuracy in the long run, a periodic invocation of the BN constructing algorithm is also necessary and, consequently, finding a method that averts extra computing time to establish a new BN structure or reestablish existing BN structure as well as the expert's erroneous BN causal relationship making is indispensable. Thus, we employ Naïve Bayes Assumption (NBA) to formulate our proposed approach. In the NBA, each factor is considered independently to others except to one factor (which we called as "target"). Further, even though in the long run, we find that two or more factors are dependent each other, the NBA still demonstrate a good conjecture [14]. By obtaining this characteristic, we can discourage any unnecessary algorithm and erroneous expert judgment in making BN structure while still maintaining prediction accuracy.

2.2 Bayesian Network and Naïve Bayes

A Bayesian Network (BN) which consists of a set of nodes and links is a causal representation model and is useful to model uncertainty [15]. A BN assumes a form of Directed Acyclic Graph (DAG) in which every node within BN (we called as BN variables) denotes random variables and every link within BN characterizes probabilistic dependences of BN variables [16]. These relationships are then measured by associating a conditional probability table with each BN variable. Usually, let $G=(V,E)$ be a DAG with a node set V and a link set E, and let $X=(X_v)_{v\in V}$ be a set of random variables indexed by v.

Naïve Bayes is a subset of BN. It has a simple structure with one target node as the parent node of all other nodes and it has a restriction that other structure to occur. [17]. The benefit of employing Naïve Bayes is that it reduces the complex calculation efforts compared to general BN due to its simple structure (see Fig. 1). That is, it is possible to avoid a more complex calculation because Naïve Bayes supposes that each node is independent to other nodes except to the target node. This independent assumption might problematic [17], however Langley *et al.* [18] have found that Naïve Bayes has surpassed other complex algorithms for a problem with a highly large datasets, especially when factors (in which each factor is represented as an attribute in data mining terminology) are independent of each other.

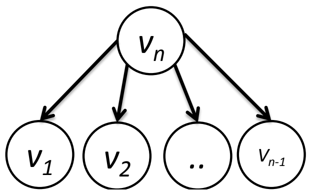

Fig. 1. A simple Naïve Bayes structure

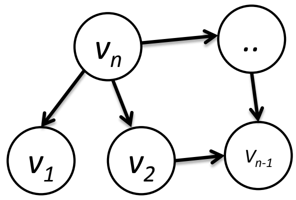

Fig. 2. A simple Bayesian Network structure

3 Proposed Approach

3.1 Process structure

Definition 1. (Process Structure)
The definition of process structure is adapted from [5]. A process structure is a directed graph $P= (A, L, F)$ consisting of sets of node A, sets of arcs L and the labeling function F

- $A=\{a_\mathrm{i} \mid i = 1, . . . , N\}$ is the set of activities, where a_i is the i^{th} activity and N is the total number of activities in P.
- $F \subseteq \{(f_\mathrm{s}, f_m)\}$ is the set of labeling function, where fs is the split function and f_m is the merge function
- $L \subseteq \{(a_{i-}, a_{j+}) \mid a_{i-}, a_{i+} \in A$ and $i+ \neq i-\}$ is the set of links where an element (a_{i-}, a_{i+}) represents a_{i-} immediately precedes, a_{j+}.
- For a split activity a_j, such that $|SA_i| > 1$, where $SA_i=\{a_{j+} |(a_i, a_{i+}) \in L\}$, $f(a_i)$ = 'AND' if all a_{i+} ' s should be executed; otherwise $f(a_{i+})$ = 'OR'.
- For a merger activity a_j such that $|MA_i| > 1$, where $MA = \{a_{j-}|(a_{i-}, a_j) \in L\}$,
- $f(a_{j-})$='AND' if all a_i should be executed; otherwise $f(a_{j-})$ = 'OR'.
- For a merger activity a_j such that $|P| > 1$, where $P = \{a_i|(a_i, a_j) \in L\}$, $f(a_j)$='AND' if all a_i should be executed; otherwise $f(a_i)$ = 'OR'.

3.2 Naïve Bayes in the Proposed Approach

We denote the Naïve Bayes incorporated in our algorithm as Naïve Bayes Model (NBM). The NBM consists of five nodes: Human Performance, Activity, Queue, Inter-arrival and Day Time (see Fig. 2). Each node represents factors involved in BP Execution. There are two types of node: target node and child node.

The target node is the factor to characterize human performance while the four other factors are the child nodes. Here, the human performance is something to predict given information from all child nodes. The detail description of each node can be seen in Table 1.

Table 1. Nodes in the NBM

No	Nodes/Factors	Possible States	Notes
1	Human Performance	Low, Medium, High	Human resource performance prediction **(Target Node)**
2	Queue	Low, Medium, High	Queue in front of the activity
3	Inter-arrival rate	Short, Medium, Long	Average of the systems' inter-arrival time/hours.
4	Performer	{human resource name}	
5	Activity	{activity name}	
6	Day time	Morning, Afternoon, Evening	The working shift

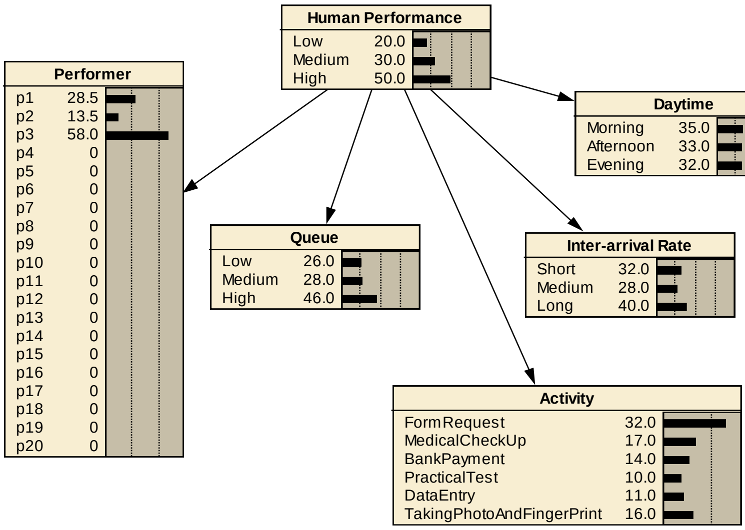

Fig. 3. Naïve Bayes Model (NBM)

To illustrate (see Fig 3), during simulation at time *t*, we need to determine the best performer (human resource) to carry out a job in the activity a_i (let say "For request"). Suppose, we observe that current situations at time *t* are: the daytime is in the *morning* (*first shift*), the inter-arrival rate is *short*, and the imposed queue is *low*. Hence, from Fig 3, we can see that we should select p_3 because it introduces probability value of 80%. The second alternative whenever p_3 is unavailable is p_1 because p_1 values 15%. Here 80% and 15% are the possibilities that p_3 and p_1 will have higher human performances.

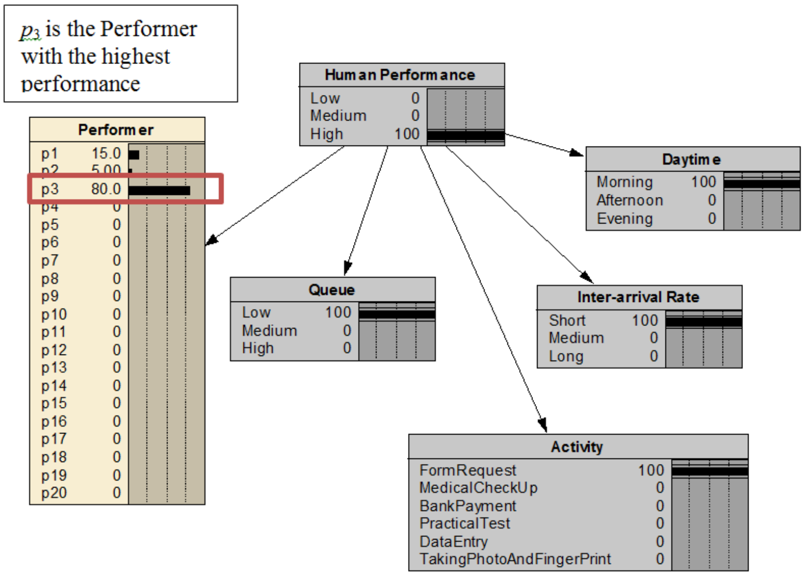

Fig. 4. Example of NBM with some evident at time = t, Selected performer will be determined by the probability of a performer to produce a higher human performance.

3.3 Naïve Bayes Selection Rule (NBSR) Algorithm

We propose *NBSR* (t, a_i) as on-the-fly performance-aware algorithm that uses NBM described in the earlier section. NBSR is an extension of BSR which is an allocation resource algorithm proposed by Nisafani *et al.* [8]. NBSR is similar with BSR algorithm except employing the NBM as the BN model. Nisafani *et al.* [8] accommodates some previously determined expert judgment factors in the BN model such as perceived workload, working pressure, technology support, performer ability, and environment condition. All of the aforementioned factors are excluded in our NBM because the judgment factors are static and might not be compatible with the randomization in the simulation software in the long run; hence, it will reduce the prediction accuracy.

The NBSR is to allocate the appropriate human resource to perform a process instance in the a_i at time t. *NBSR* (t, a_i) employs several parameters:

- $R_a = \{r_n | n=1, 2, ..., N\}$ is the set of human resources where r_n is the n^{th} human resource and N is the total number of resources employed in a_i
- $Q_a(t)$ is the queue before a_i at time t
- *BN* represents the utilized Naïve Bayes Model
- D_a (t) \in {morning, evening, afternoon} is daytime at time t
- $I(t)$ \in {low, medium, high} is the inter-arrival rate at time t

Fig. 5 denotes the NBSR. Here, the algorithm forcasts the performance of human resources and assigns an incoming job to human resource with the highest performance predicted from the NBM. Each resource will be recognized with non-negative and unique index, and the NBSR will select the index with the possibility to introduce a higher performance. Also, one of the algorithm component is to invoke Naïve Bayes Model (see Function *do_inference* in line 11). The *do_inference* is defined as a probability function in BN as follows:

$P(Human\_Performance = \text{“High”} | Activity = a_i, Queue = Q_a(t), Humanresource = r_n, Daytime = D_a(t), Inter\text{-}arrival = I(t)).$

By using this function, we select the human resource given activity a_i, Queue $Q_a(t)$, inter-arrival $I(t)$, human resource $r_{n,}$, and Day time $D_a(t)$.

```
1    FUNCTION SELECT RESOURCE (a_i, Q_a(t), R_a, BN, D_a (t), I(t))
2    BEGIN
3       BOOLEAN loop := TRUE ;
4       RESOURCE res ;
5       DOUBLE temp := -9999;
6           //very big negative number, indicating no human resource is selected
7       WHILE (loop = TRUE)
8       {
9               FOR (INT index :=0; index<size(R_a) ; index++)
10              {
11                  value : = do_inference(a_i, Q_a(t), R_a, BN, D_a (t), I(t))
12                  IF (temp < value && r_index IS IDLE) THEN
13                      temp := value;
14                      res = r_index ;
15                      // r_index is the resource in the R_a with index = index
16                  END IF
17              }
18              IF (res != NULL) THEN
19                  loop := FALSE;
20              END IF
21      }
22          RETURN res;
22      END
```

Fig. 5. NBSR Algorithm

4 Experiments and Results

4.1 Static Rules for Resource Allocation

Below selection rules (see Table 2) are adapted from a simulation book which can be applied in the manufacturing process for managing machine resources [19]. Since, our approach is to manage human resource, comparing these static rules with our proposed approach is relevant because their ability to measure our algorithm in accommodating the human performance dynamics.

Table 2. Static Rules for Resource Allocation [19]

No	Priority Rule	Description
1	ORDER	Select from the free resources in the preferred order
2	LIDDLE	Select the resource that has the largest idle to date
3	SIDDLE	Select the resource that has the smallest idle to date
4	RANDOM	Select randomly among all free resources

Table 3. Rules Comparison for Resource Allocation

No	Priority Rule	Description
1	ORDER	Select from the free resources in the preferred order
2	LIDDLE	Select the resource that has the largest idle to date
3	SIDDLE	Select the resource that has the smallest idle to date
4	RANDOM	Select randomly among all free resources
5	BSR	Nisafani [8]'s algorithm, select the resource using preferred priority
6	NBSR	Proposed algorithm

4.2 Experiment Results

This study uses a real world semi-automatic business process mentioned in Nisafani et al. [8]. The process is the driver lisence application process conducted in Indonesia. The business process consists of 8 activities in which 6 of them were performed by the assigned police officers (see Fig. 5). There is no officer assigned for two activities (theoritical test and practical test) since both activities are conducted by the applicants.

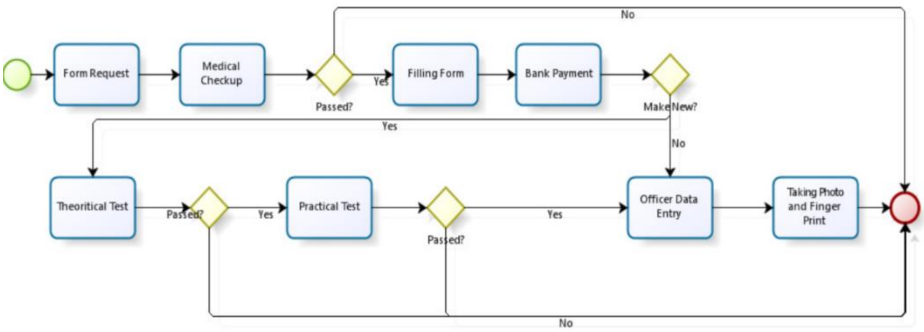

Fig. 6. Driver License Application Process

Every officer is responsible to his/her activity and officer transfer among activities is not allowed. The detail parameters of the simulated system (such as human resource processing time, interarrival time, etc) is available at Nisafani *et al.* [8]. In general the simulation running time is 13 hours per day and is consisting of three working shifts: Morning (8 AM - 12 AM), Afternoon (12 AM – 4 AM), and Evening (4 PM – 9 PM). Nisafani *et al.* [8] recorded each resource performance within three shifts, from which human performance distribution was developed. Most of the human performance distribution followed normal distribution. In addition, the replication number is 10 and the average instance numbers per replication is 1500.

We compare NBSR with the static rules described in Table 2 and BSR. The comparison is available in Table 4. In general, NBSR outperformed all static rules and BSR in terms of the mean and standard deviation. Hence, we can say that NBSR can accomodate the human performance fluctuations. However, even though NBSR demostrates a better completion time than BSR, the difference between the NBSR and BSR are very near. We suspect that the number of the human resources in each activity (three officers) and the simulation duration are responsible to the small distance of the completion time between the BSR and NBSR. A larger number of the human resources per activity as well as a longer simulation duration might help us to clearly understand how the behavior of the NBSR and BSR when the system escalates.

Table 4. Experiments Result In Terms of the Average Completion Time
(Bold Numbers indicates the Lowest Completion Time for each replication)

Replication#	**RANDOM**	**ORDER**	**SIDDLE**	**LIDDLE**	**BSR**	**NBSR**
1	1096.167	1567.904	2590.262	1808.748	922.6673	**921.3511**
2	2232.817	3127.327	2016.082	2088.21	1364.995	**1362.517**
3	1171.825	1945.549	2502.724	1656.001	**828.006**	826.453
4	983.3494	407.1013	2421.051	1953.586	919.0942	**917.6344**
5	2327.259	1139.865	1789.55	1674.413	1092.204	**1088.304**
6	1403.965	**1181.13**	3668.057	779.2716	1954.795	1948.438
7	1034.707	**1195.526**	1702.11	3018.147	1547.087	1560.595
8	638.2284	1597.009	952.9931	664.4587	912.9886	**896.3806**
9	1842.827	1664.555	**1363.279**	3165.451	1670.024	1666.54
10	925.5398	1544.924	1514.428	1987.32	1326.165	**1321.235**
Mean	1365.668	1537.089	2052.054	1879.561	1253.803	**1250.945**
Standard Deviation	547.2241	662.916	735.1494	761.2297	362.4221	**361.4053**

5 Conclusion

This study proposes an on-the-fly performance-aware resource allocation in business process management. We utilize Naïve Bayes Model in the Naïve Bayes Selection Rule (NBSR) algorithm for selecting the best performer to accomplish an incoming task. We compare our approach with four static rules and previously developed BSR. The result shows that NBSR surpasses all the aferomentioned rules. Therefore, the result indicates that Naïve Bayes approach is beneficial to model complex relationships among factors in the BP execution. Future research might accomodate resource transfer among activities and incorporate workflow resource patterns [20] in the business process management. It is necessary to conduct a longer simulation time to thoroughly observe how the NBSR works in the long run.

Acknoledgement

This is the initial version for the Asia Pacific Business Process Management Lecture Notes in Business Information Processing Volume 219, 2015, pp 70-80 of Springer. The final version of this manuscript is available at (http://link.springer.com/chapter/10.1007/978-3-319-19509-4_6).

References

1. Bae, H., Lee, S., Moon, I. M.: Planning of business process execution in Business Process Management environments. Information Sciences 268, 357–369 (June 2014)
2. Eder, J., Pichler, H., Gruber, W., Ninaus, : Personal schedules for workflow system. Lecture Notes Computer Science, 216–231 (2003)
3. Huang, Z., van der Aalst, W. M. P., Lua, : Reinforcement learning based resource allocation in business process management. Data & Knowledge Engineering 70(1), 127–145 (January 2011)
4. Huang, Z., Lu, X., Duan, H.: Resource behaviour measure and application in business process management. Expert systems with applications 39, 6458–6468 (2012)
5. Rhee, S.-H., Bae, H., Kim, Y.: A dispatching rule for efficient workflow. Concurrent Engineering: Research and Application 12(4), 305-318 (December 2004)
6. Wu, B. Y., Chi, C.-H., Chen, Z., Gu, M., Sun, J. G.: Workflow-based resource allocation to optimize overall performance of composite services, Future Generation Computer Systems. Future Generation Computer Systems 25, 199-212 (March 2009)
7. Zhao, J. L., Stohr, E. A.: Temporal workflow management in a claim handling system. In : The international joint conference on work activities coordination and collaboration, New York, pp.187-195 (1999)
8. Nisafani, A. S., Wibisono, A., Kim, S., Bae, H.: Bayesian Selection Rule for Human-Resource Selection in Business Process Management Systems. Journal of Society for e-Business Studies 17(1) (2014)
9. Yahya, B., Wibisono, A., Bae, H., Ryu, K.: Bayesian network for finding best in BPM Environment. ICIC Express Letters 2(2), 473-479 (2011)
10. Dumas, M., La Rosa, M., Mendling, J., Reijers, H.: Fundamentals of Business Process Management. Springer, Berlin, Germany (2013)
11. Wibisono, A., Nisafani, , Bae, H., Park, Y.-J.: Early detection for supply chain disruption using bayesian network. In : Asia Pacific BPM, Brisbane, Australia (2014)
12. Ha , B.-H., Bae, J., Park, Y. T.: Development of process execution rules for workload balancing on agents. Data Knowledge Engineering 56, 64–84 (2006)
13. Armstrong, J.: Principles of Forecasting, A handbook for researcher and practitioners.

Springer, United States (2001)

14. Whitten, I., Frank, E., Hall, M.: Data Mining: Practical Machine Learning Tools and Techniques: Practical Learning Tools and Techniques, Third Edition. Morgan Kauffman, United States (2011)
15. Kao, H.-y., Huang, C.-h. H., Li, H.-l.: Supply Chain Diagnostics With Dynamic Bayesian Networks. Computer and Industrial Engineering 49(2), 339-347 (2005)
16. Gregory, C., Herskovits, E.: A Bayesian method for the induction of probabilistics network from data. Machine Learning 9(4), 309-347 (1992)
17. Cheng, J., Grainer, R.: Comparing Bayesian Network Classifiers. Morgan Kauffman Publishers (1999) 101-108
18. Langley, P., Iba, W. I., Thompson, : An analysis of Bayesian classifiers. In : Tenth National Conference of Artifical Intelligence (1992)
19. Pritsker , A., O'Reilly, J.: Simulation with Visual SLAM and AweSim. System Publishing Corporation, WestLafayette, Indiana (1999)
20. Russel, N., ter Hofstede, A., Edmond, D., van der Aalst, W.: Workflow Resource Pattern. (2004)