Decomposition Using Refined Process Structure Tree (RPST) and Control Flow Complexity Metrics

Indra Gita Anugrah[1], Riyanarto Sarno[2], Ratih Nur Esti Anggraini[3]
[1, 2, 3] Department of Informatics, Faculty of Information Technology
Institut Teknologi Sepuluh Nopember
Surabaya
e-mail : indra14@mhs.if.its.ac.id[1], riyanarto@if.its.ac.id[2], ratih_nea@if.its.ac.id[3]

Abstract* — Process mining is a technique that aims to gain knowledge of the event log. The amount of data in the event log is very influential in the Process mining, because it contains millions of activities that shape the behavior of a company. The three main capabilities possessed by mining process is a discovery, conformance, and enhancement. This paper, we present an approach to decompose business processes using Refine Process Structure Tree (RPST). By breaking down a whole into sub models Business Processes (fragments) to the smallest part (atomic) can facilitate the analysis process and can easily be rebuilt. To measure the level of complexity in the model fragment and atomic models we use complexity Control flow metrics. Control flow complexity metrics have two main approaches that are count based measurement and execution path based measurement path. Count based measurement used to describe a static character, while an execution path based measurement used to describe the dynamic character of each model fragment or atomic models (bond fragment).

Keywords* — Process mining; decompose business process; refined process structure tree; control flow complexity metrcs.

I. Introduction

Nowadays a variety of approaches to address the challenges involved in Business Process Management (BPM) is mostly done. By using the event log we could see the events in a business process. Event log contains records of events derived from the activities of a company from the implementation of an information system.

Event logs can be used to look at the ability of the mining process. The first discovery is the formation of the actual model business processes by utilizing the data from the event logs. The second is the process of comparing the conformance of existing models in the event log with other business process model. Conformance checking can be used to check that there are real events in the event log. While the third enhancement is a process improvement models by studying the logs event data that has been generated by the system, where conformance checking measures the similarity between the model and reality [2].

Rapid data growth is causing problems in the mining process. Existing process mining techniques have problems in reading behavior on a large event log. One approach is by using the technique of decomposition, the decomposition process is a method used to break up a business model of the overall process into sub model called fragment. Decomposition approach can generally be made by combining the discovery process with the conformance checking process [4].

An idea to build reusable business process fragment with the decomposition of the overall business process models that have a complex structure fragment into a model that aims to be easy to understand, easy to maintain and quick process reuse. Fragments can be reused in accordance with the requirements of business processes as desired. Several approaches to decompose one business process using business process linear and not linear business process. In the linear process of decomposition is easily done using the activity element [8]. An atomic activity which has its own business goal is considered as business fragment. Furthermore, activities of succession that satisfy a specific business need can be regrouped and considered as business process fragments [9].

This paper will discuss the decomposition of a business process model into multiple models fragments consisting of the very small (atomic) called bond fragments using RPST were then analyzed using flow control complexity metrics to obtain fragments or variants most suitable bond fragment. In the second section will discuss the literature study, in section 3 will discuss the methods used and study cases, then section 4 is the conclusion of research and section 5 is the future work.

II. Literature Review

A. Petri Nets

Petri net is a graphical tool as a formal description of a flow of activities in complex systems (using a logical representation of the block diagram or trees). Petri net represents a natural interaction logic between the parts or activity in a system. A situation that can be modeled with Petri Net is synchronization, sequentiality, concurrency and conflict. Classical Petri Net is a graph that connected with two types of nodes, called places and transitions. Each node is connected with arcs. Places represented by circles and transitions to a square [1].

978-1-5090-0096-8/15/$31.00 © 2015 IEEE

Definition 1 *(Petri Net)*
A Petri net is a triple *PN = (P, T, F)* , where P is a finite set of places.

B. Refined Process Structure Tree (RPST) and SESE

Workflow graph used as a model control flow of business process and study the problem of decomposition workflow. for example, found the structure of the workflow graph more precisely find the decomposition of workflow graph into a hierarchy of sub-workflows by using a single control and a Single Entry Exit which is an important stage of decomposition which aims to be the result of decomposition into unique, modular, which means modular local changes occurs on workflow graph will only change the local on this graph workflow refined process called tree structure, based on the expansion of sequential programs by Trajan and Valdes [ACM POPL '80, 1980, pp. 95-105]. There are two independent characteristics of the process tree structure:

1. Describe simple characteristics to justify the choice of decomposition and building characteristics that allow us to calculate the decomposition in linear time.
2. The latter is based on the tree of triconnected components (also known as SPQR tree) from a biconnected graph [5].

C. Definition and Measurement of Control Flow Complexity

Control flow behavior of a process built as splits. Splits are determined to follow the path that might control of a process. Joins have a different role, the type of synchronization has the specific point in the process. In the study conducted by Jorge Cordosa et al [10] presented the evaluation of the complexity of branching based XOR-split, OR-split and AND-split.

D. Control Flow Complexity Metrics for Petri Net

To measure the level of complexity of the process required complexity metrics. Based on research that conducted by Chengying Mao et al. The research given the complexity of the two types of metrics that count-based metric and metric-based execution path [7].

1. *Count Based Measurement*

Count-based measurement is a naïve approach to calculate the level of complexity of construction in the program, process or network. In the study Chengying Mao et al. Count-based measurement used to measure the complexity of the structure of the business processes are represented by Petri Net. But in this study to count based measurement we will only use the analysis of number of places, number of transitions, and average degree of places.

a. *Number of Places*

On the number of places we calculate the total amount of places of Petri net based business processes that reflect the exchange of data in business processes. Can be calculated using the formula:

$$N_p = | P | \quad (1)$$

P is the set of places petri net.

b. *Number of Transitions*

On the number of transitions we count the total number of petri net transitions. Transition is an operation or control logic. Can be calculated using the formula:

$$N_T = | T | \quad (2)$$

T is a set of petri net transitions.

c. *Average Degree of Place*

Based on the perspective of the network. Information from the node reflects the degree of interaction between nodes. Two types of nodes in a Petri net business processes, so the interaction of place and transition considered. Average Degree of Place (ADP) is calculated using the formula:

$$ADP = \frac{\sum_i deg(pi)}{|P|} = \frac{\sum_i [indeg(pi) + outdeg(pi)]}{|P|} \quad (3)$$

pi ∈ P is the place to i in petri net and deg (pi) is the degree of the nodes in accordance with place pi in the network and can be divided into two parts, namely indeg (pi) and outdeg (pi).

d. *Average Degree of Transition*

As well as ADP. *Average Degree of Transition (ADP)* can be calculated by the formula:

$$ADT = \frac{\sum_i deg(ti)}{|T|} = \frac{\sum_i [indeg(ti) + outdeg(ti)]}{|T|} \quad (4)$$

ti ∈ T is a transition to i in petri net and deg (Ti) is the degree of the nodes in accordance with transition ti in the network and can be divided into two parts, namely indeg (ti) and outdeg (ti).

2. *Execution Path Based Measurement*

Count based measurement approach is an approach that reflects the static feature of the composition of the bond fragment in a model fragment so that it takes a different approach that describes the dynamic character. Execution path based on metrics introduced by adopting the knowledge of cognitive informatics.

In an execution path based on metrics, all the places and transitions of nodes are given different weights tailored to the complexity of the branching occurs. Therefore, an effective way is to measure the complexity of a node places or transition from the perspective of cognitive informatics.

Definition 2 *(Parallel Execution Relation)*
Suppose seq 1 and seq 2 represent two sequences following each other in the node transition Petri net denoted seq seq 1 || seq 2

Definition 3 *(Execution Path)* Sequence is composed of nodes and transition nodes place called the execution path. It is possible path may contain subsequences serial and parallel subsequences.

Definition 4 *(Execution Path Complexity)* Given execution path Pt, control complexity of Pt can be defined as the sum of the complexity of all the places and transitions in the execution path

$$C(Pt) = \sum_{i} C(pt) + \sum_{i} C(tj) \qquad (5)$$

calculation of dynamic complexity can be increased with the complexity of the average execution path (AEPC).

$$\text{AEPC} = \sum_{i=1}^{k} prob\,(Pti).\ \text{C}(Pti) \qquad (6)$$

III. The Proposed Method

The proposed method combines the use Refined Process decomposition Tree Structure and Flow Control Complexity Metrics to produce reusable business process. In the first phase is to decompose the overall model in this case study using a model of Claims in an insurance company that produces sub model called the fragment and a fragment obtained from a smallest part called bond fragment. In the next phase will be analyzed to determine the level of complexity of the model fragment or bond fragment.

A. Refined Process Structure Tree (RPST)

1. *Fragment*

Based on the basic notations of graph theory. A graph is 2 pieces of nodes connected by an edge and Workflow graph based on the two terminal graph (TTG). G = (V, E, F) where G is a graph, V is a set of nodes, E is a collection of edges and M is a mapping of each pair of nodes with each edge to edge directed or undirected multi-graph

Definition 5 *(Boundary node, entry, exit, fragment)* G is a TTG and F is a subset of edges as Gf is a connectedness sub graph of G. A node v ∈ Vf is a boundary node of F, A boundary node v is an entry of F when no incoming edges of v in F. F is called a fragment of G if it has two boundary nodes as entry and exit.

2. *Objective fragments and the Refined Process Structure Tree*

Find a decomposition of a workflow graph into fragments. Form a decomposition (or a parse) of a TTG. G = (V, E, F) is a hierarchy of the fragment.

Definition 6 *(Refined process structure tree (RPST))* Fragment G is objective if there is no overlap with other fragment, a collection of objective fragment called RPST decomposition of G, corresponding decomposition tree is called Refined Process Structure Tree (RPST).

3. *Characteristic of RPST*
   1. *Triconnected component*

   Triconnected component of which is bound and polygon (triconnected graph), a bound is a graph that contains them two and two edge nodes, while a polygon is a graph that contains at least three nodes.

   2. *Sequence and bond fragment*

   Definition 7 *(Maximal sequence)*
   1. If $F_0 \in \mathcal{F}(v_0, v_1)$ and $F_1 \in \mathcal{F}(v_1, v_2)$ such that $F_0 \cup F_1 = F \in \mathcal{F}(v_0, v_2)$, we say that F_0 and F_1 are in sequence (likewise: F_1 and F_0 are in sequence) and that F is a sequence.
   2. F is a maximal sequence if there is no fragment F_2 such that F and F_2 are in sequence.

   Definition 8 *(Bond fragments)*
   1. A bond fragment if it is the union of at least two branches from $\mathcal{D}(u,v) \cup \mathcal{U}(u,v) \cup \mathcal{D}(u,v)$,
   2. A directed bond fragment if it is the union of at least two branches from $\mathcal{D}(u,v) \cup \mathcal{U}(u,v)$.

B. Claims in a insurance company : A Study Case

In this case study will be conducted on the claims in an insurance company [6] where in one case consists of some activity, starting from the registry claim activity to archive claim activity. In the above case study we tried to decompose the model into sub-models within the so-called fragment and fragment consists of a bond fragment that is the basic activity called atomic models using the Refined Process Tree Structure (RPST).

C. Fragment and Bound Model from Decomposition using Refined Process Structure Tree (RPST)

Case study will be conducted in this study is the claims in an insurance company in Fig. 1, starting from the registry claim activity to archive claim activity. In those case study we tried to decompose the model into sub-models within the so-called fragment and fragment consisting of a bond fragment which is a basic activity which is the atomic models. Decomposition using Process Refined Structure Tree (RPST) starts with changing the entire model into a multi undirected graph and change the transition graph form the edge that connects every places.

Based TTG (Two Terminal Graph) fragment formed with respect to the Definition 5 where the fragment is sub graph of G which has a boundary, a single entry and single exit. To facilitate the decomposition process determine the maximal sequence as in Definition 7. Next break form the fragment into component such triconnected is a polygon and bonds. Polygon fragment is a sub graph that contains three nodes that overlap with maximal sequences. Bonds is the smallest fragment consisting of two nodes and two edges are interconnected in Fig. 2. Some fragment obtained polygon models and bounds fragment where T is fragment while B is a bond which is the smallest part of the fragment. Results of the decomposition process are presented using the tree structure as in Fig. 3.

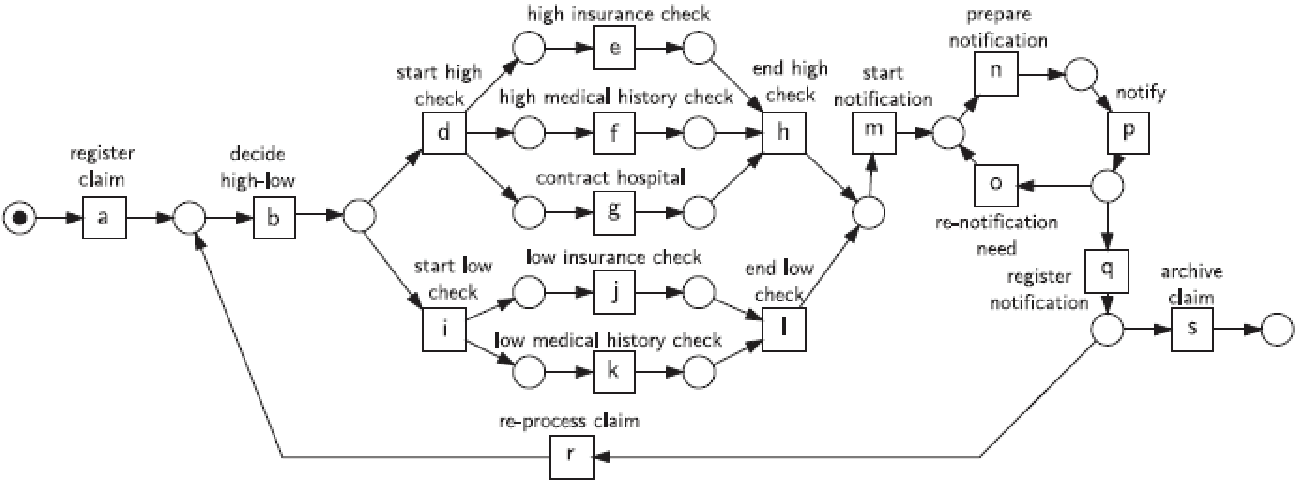

Fig 1. Claims in an insurance company for a case study [6]

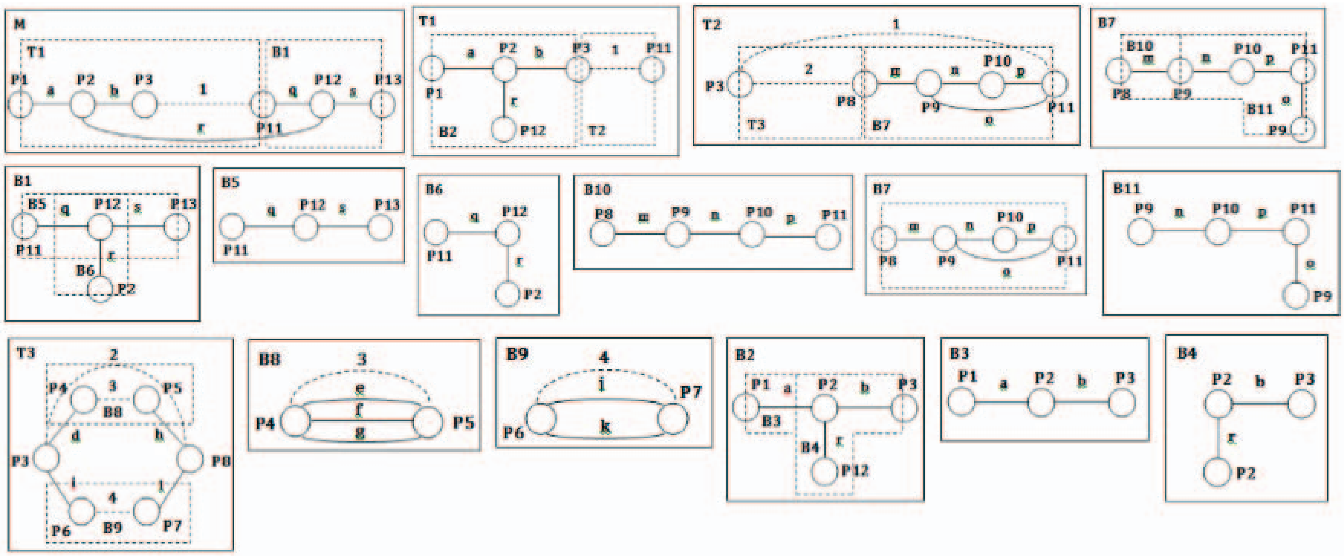

Fig 2. Fragment decomposition overall model from fig. 1. T is fragment and B is bond fragment

D. Control Flow Complexity Metrics for variant Fragment

As noted in section 2, to measure the level of complexity of the process required complexity Metrics. The method used to measure the complexity of the fragment is Count Based Measurement and Execution Based Measurement path. This method was chosen because the Control Flow Complexity Metrics can measure the level of complexity of Petri Net in modeling business processes as a whole, while the fragment is a sub model of the overall model is based Petri Net. But first we calculate complexity of the overall model.

Before we analyze using the Control Flow Complexity Metrics first we specify the model fragment that will be analyzed. Where a model has variants bond fragment which is the smallest part which forms a model of fragments. One model fragment formed from decomposition using Process Refined Structure Tree (RPST) as in fig. 4, where on the model fragment T2, T3 fragment generated models consists of several sections smallest B8 and B9. The next step is to change the model fragment into a graph trending and changing edges into transitions, the same thing is done on the B8 and B9 bond fragment thus obtained as in fig. 6.

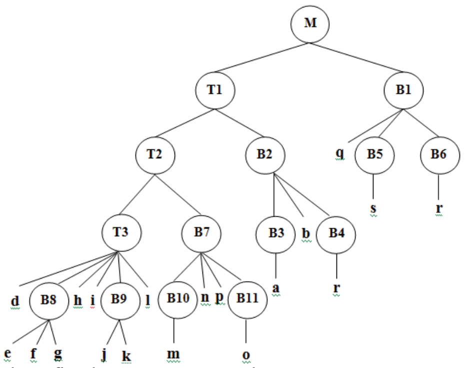

Fig 3. The refined process structured tree

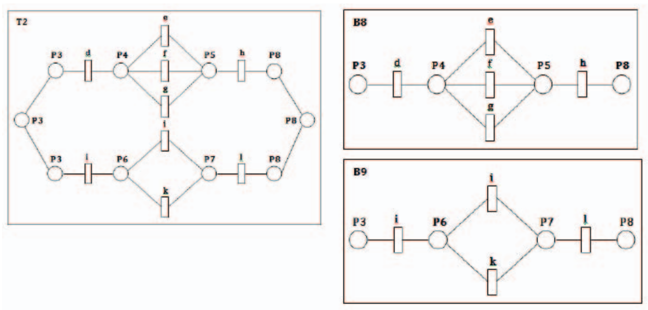

Fig 4. Model fragment T2, bond fragments B8 and B9.

TABLE 1. COMPLEXITY WEIGHT FROM FIGURE 1.

No.	Basic Structure	Formula	Weight
1.	xor split (decide high - low)	$CFC_{XOR-split}(a) = fan-out(a)$	2
2.	or split (start high check)	$CFC_{OR-split}(a) = 2^{fan-out(a)}$	2^3-1
3.	xor split (start low check)	$CFC_{XOR-split}(a) = fan-out(a)$	2
4.	xor split (notify)	$CFC_{XOR-split}(a) = fan-out(a)$	2
5.	xor split (register notification)	$CFC_{XOR-split}(a) = fan-out(a)$	2

Here are the steps to measure the level of complexity:

- CFC_{abs} is the largest value of the complexity of a process $CFC_{abs} = 2 + 7 + 2 + 2 + 2 = 15$
- So that CFC_{rel} is the average greater from overall architectural complexity of a process.

$CFC_{rel} = 15 / 5 = 3$

$$CFC_{rel}(P) = \frac{CFC_{abs}(P)}{|\{\text{XOR-splits of p}\} \cup \{\text{OR-splits of p}\} \cup \{\text{AND-splits of p}\}|}$$

Box. 1. Formula to calculation control flow complexity [10].

- To perform measurements at the level of complexity of fragment or split (Decide high-low), then the value of the CFC for T2 obtained from the basic structure split (Decide high-low) based on Table 1 is 2.

A. Count Based Measurement

1. *Number of Places*

   $N_p = | P |$

   - $N_p(T2) = 8$
   - $N_p(B8) = 4$
   - $N_p(B9) = 4$

2. *Number of Transitions*

   $N_t = | P |$

   - $N_t(T2) = 9$
   - $N_t(B8) = 5$
   - $N_t(B9) = 4$

3. *Average Degree of Places*

   $$ADP = \frac{\sum_i deg(pi)}{|P|} = \frac{\sum_i [indeg(pi) + outdeg(pi)]}{|P|}$$

   - $ADP\ (T2) = \frac{\sum_i deg(pi)}{|P|} = \frac{9+9}{8} = 2.25$
   - $ADP\ (B8) = \frac{\sum_i deg(pi)}{|P|} = \frac{5+5}{4} = 2.5$
   - $ADP\ (B9) = \frac{\sum_i deg(pi)}{|P|} = \frac{4+4}{4} = 2$

4. *Average Degree of Places*

   $$ADT = \frac{\sum_i deg(ti)}{|T|} = \frac{\sum_i [indeg(ti) + outdeg(ti)]}{|T|}$$

   - $ADT\ (T2) = \frac{\sum_i deg(ti)}{|T|} = \frac{9+9}{9} = 2$
   - $ADT\ (B8) = \frac{\sum_i deg(ti)}{|T|} = \frac{5+5}{5} = 2$
   - $ADT\ (B9) = \frac{\sum_i deg(ti)}{|T|} = \frac{4+4}{4} = 2$

B. Count Based Measurement

- *T2 :*

  Path1=P3→d→(P4→e→P5, P4→f→P5, P4→g→P5)→h→ P8

  Probability of path 1 : 0.6

  C(path1) = 3x2 + 3x1 + 2x2 + 3x2 = 19

  Path2=P3→i→(P6→j→P7, P6→k→P7)→j→ P8

  Probability of path 2 : 0.4

  C(path2) = 2x2 +2x1 + 2x2 +2x2 = 14

  AEPC = 0.6 x 16 + 0.4 x 14 = 15.2

- *B8 :*

  Path1= P3→d→P4→e→P5→h→ P8

  Probability of path 1 : 0.2

  C(path1) = 5x1 + 3x1 +1x1 = 9

  Path2= P3→d→P4→f→P5→h→ P8

  Probability of path 2 : 0.2

  C(path2) = 5x1 + 3x1 +1x1 = 9

  Path3= P3→d→P4→g→P5→h→ P8

  Probability of path 3 : 0.2

  C(path3) = 5x1 + 3x1 +1x1

= 9

AEPC = 9 x 0.2 + 9 x 0.2 + 9 x 0.2 = 5.4

- *B9 :*

  Path1= P3→i→P6→j→P7→l→ P8

  Probability of path 1 : 0.1

  C(path1) = 3x1 + 2x1 +1x1

  = 6

  Path2=P3→i→P6→k→P7→l→P8

  Probability of path 2 : 0.1

  C(path2) = 3x1 + 2x1 +1x1

  = 6

  AEPC = 6 x 0.1 + 6 x 0.1 = 1.2

TABLE 2. VALUE OF COMPLEXITY MATRICS FRAGMENT

No	Metrics	Metrics Value		
		T2	B8	B9
1	N_p	8	4	4
2	N_t	9	5	4
3	***ADP***	2.25	2.5	2
4	***ADT***	2	2	2
5	***AEPC***	15.2	5.4	1.2
6	***CFC***	2	7	2

From Table 2 it can be seen that the T2 and B8 have in common is having ADP value is greater than the value of ADT, which indicates that the magnitude of the strength of the interaction between nodes in model fragments or bonds fragment. Place in Petri net-based business processes indicated for data storage or justification of branching processes [7]. And from the above data it can be said that the B8 has a value that is closer than the B9 as a variant of the bond fragment because the value of T2 and T2 fragments of the same B8 ADP is greater than the value of ADT. By using the value of AEPC we can find a path that is frequently used value of T2 with the overall model is 15.2 while the value of the path that is often used in bond B8 and B9 fragment T2 model is 5.4 and 1.2. While the value of the limit value CFC show the complexity of a process that is built from some parts of the process itself as a model of T2 fragment consisting of models B8 and B9 bond fragment, so that the value of complexity that is used as a constraint to the development process is CFCrel = 2/9 = 0, 22, so that the limit value obtained is 0.22 to 2.

IV. CONCLUSION

By using Process Refined Structure Tree we can decompose the overall business processes into a model where a fragment composed of the smallest piece models called bond fragments. Control flow complexity metrics used to analyze the complexity of the model fragment and bond fragment so that we could know variants corresponding bond fragments to build a model fragment.

By using control flow complexity metrics, we know the interaction between nodes in the model fragment or bond fragment through value ADP (Average Degree of Place) and ADT (Average Degree of Transitions). To analyze the path that is often used against the entire model we can use average execution path complexity (AEPC). While CFC is used to indicate the value of the complexity of the process which is built from multiple part of process, for example, fragments T2 models constructed by bond fragments B8 and B9.

FUTURE WORK

To fix method to get a better value is the reference to rebuild (compose) Model fragment and fragment Bond into the overall Business Process Model.

REFERENCES

[1] Andrea Bobbio. System Modeling With Petri Nets. Istituto Elettrotecnico Nazionale Galileo Ferraris Strada delle Cacce 91, 10135 Torino, Italy.

[2] Wil M.P. van der Aalst. Decomposing Petri Nets for Process Mining. Architecture of Information Systems, Eindhoven University of Technology, Eindhoven, The Netherlands.

[3] Dmitry.A Zaitsev. Functional Petri Nets, Cahier N 224 mars 2005.

[4] Wil M.P. van der Aalst, Decomposing Petri Nets for Process Mining - A Generic Approach.

[5] Jussi Vanhatalo, Hagen Völzer a, Jana Koehler a, The refined process structure tree, Data & Knowledge Engineering 68 (2009) 793–818, 2009 Elsevier B.V. All rights reserved.

[6] Jorge Munoz-Gama, Josep Carmona, Wil M.P.vander Aalst, Single-Entry Single-Exit decomposed conformance checking, Information Systems, 2014 Elsevier B.V. All rights reserved.

[7] Chengying Mao, Control Flow Complexity Metrics for Petri Netbased Web Service Composition, 2010 Journal of Software, Vol. 5, No. 11, November 2010, School of Software and Communication Engineering, Jiangxi University of Finance and Economics, 330013 Nanchang, China.

[8] B. Orrins, J. Yang, and M. P. Papazoglou, "A framework for business rule driven web service composition," in Springer-Verlag Berlin Heidelberg. Springer, 2003, pp. 52–64.

[9] Maryam Radgui, Rajaa Saidi1, and Salma Mouline, A Pattern for the Decomposition of Business Processes, Special Issue of International Journal of Computer Applications (0975 – 8887) on Software Engineering, Databases and Expert Systems – SEDEXS, September 2012.

[10] Jorge Cardoso, Business Process Control-Flow Complexity, International Journal of Web Services Research, 5(2), 49-76, April-June 2008, *University of Madeira, Portugal* 2008.