Proceedings of the 2017 Winter Simulation Conference
W. K. V. Chan, A. D'Ambrogio, G. Zacharewicz, N. Mustafee, G. Wainer, and E. Page, eds.

A TOOL FOR MINING DISCRETE EVENT SIMULATION MODEL

Yan Wang

University of Bordeaux - Bat A31
Laboratory IMS UMR CNRS 5218
351 Cours de la Libération, 33405 Talence cedex, FRANCE

Grégory Zacharewicz

University of Bordeaux - Bat A31
Laboratory IMS UMR CNRS 5218
351 Cours de la Libération, 33405 Talence cedex, FRANCE

Mamadou Kaba Traoré

University of Blaise Pascal
Laboratory LIMOS UMR CNRS 6158
1 rue de la Chebarde, 63178 Aubiere cedex, FRANCE

David Chen

University of Bordeaux - Bat A31
Laboratory IMS UMR CNRS 5218
351 Cours de la Libération, 33405 Talence cedex, FRANCE

ABSTRACT

Mining a discrete event simulation model from data has always been a big challenge, which is related to the problem of system inference in systems theory. D2FD (Data to Fuzzy-DEVS model) method can be used to discover a discrete event simulation model. This method not only provides a way of data mining but also integrates process mining with the modularity, frequency, timing aspects and event data. This paper presents a mature tool applying D2FD method. This tool is implemented as an available and dedicated plug-in within the open-source process mining toolkit ProM. The simulation tool SimStudio is embedded in this plug-in and it can simulate Fuzzy-DEVS atomic and coupled model. Two case studies of real life processes, taken from Rabobank Group ICT and Dutch Employee Insurance Agency, are analyzed to evaluate this tool.

1 INTRODUCTION

Systems theory (Simon 1991) is related to the study which uses the mathematical models to describe systems. The knowledge of the system can be organized in a 4-level hierarchy (Klir 2013) as shown in Figure 1. The source level identifies a portion of the real world where we are going to measure and observe; the data level is a data base of measurements and observations made for the source system; the generative level uses formulas or other means to constitute knowledge; the structure level describes the component systems that are interconnected together. The structure level which is at the top of these levels can provide more knowledge than the source level which is at the bottom of these levels. Zeigler (Zeigler et al. 2000) proposes three systems problem according to this 4-level hierarchy: in system analysis, we have an existing system and we are trying to review or create source; in system design, the system doesn't exist yet and we are investigating the alternative structures for a new system; in system inference, the system exists and we are trying to generate the structure. Among them, system inference is a big challenge of system modeling.

Thanks to D2FD method, it provides a complete method to deal with system inference and mine a discrete event simulation model. This method contains three parts. In the first part, it provides a new method to transform the event data to the event logs. The event data is observed from real world and the event logs is recorded based on XES (Extensible Event Stream) standard (Van der Alast 2011). In the second part, a method of transforming event logs into transition systems (Robert 1976) is included. This method is coming from process mining (Van der Alast 2011). In the third part, a new method of transforming transition system into Fuzzy-DEVS (Discrete Event System Specification) is proposed. Fuzzy-DEVS model (Kwon et al. 1996) not only provides a general framework to represent dynamic systems, but also represents more information including the modularity, frequency and timing aspects.

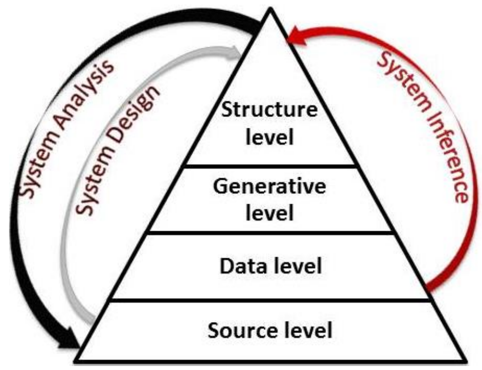

Figure 1: System problems and hierarchy of knowledge.

The implementation of D2FD method is the plugin based on the Process Mining Framework (ProM). ProM (Van der Alast 2011) is an open-source framework for collecting tools and applications of process mining. This plugin is practical and available (Wang 2016). It is also integrated with the tool SimStudio (Traoré 2008) for the simulation of Fuzzy-DEVS model. This paper presents a mature tool for mining discrete event simulation model and applies it in two real case studies. These two case studies come from Rabobank Group ICT (BPI Challenge 2014) and Dutch Employee Insurance Agency (BPI Challenge 2016). The reason of choosing these two challenges as case studies is because the source we choose comes from data instead of event logs. The data in two case studies are used to mine the Fuzzy-DEVS models. The simulation results are extended with the results of Fuzzy-DEVS coupled model using fuzzy cluster (Kaufman et al. 2009).

This paper is organized as follows. Section 2 introduces some related studies. Section 3 gives a general view of D2FD method. Section 4 presents the results of two case studies. At last, the paper is concluded in Section 5.

2 RELATED STUDIES

The event log is usually the starting point of process mining. Jans et al. (2014) proposes a new method to audit data from bank in reality by auditing relevant information from the event log. Van der Aalst (2015) proposes a conceptual approach to extract event data from databases. It is recognized as a formal disciplines before transforming event data into event logs. However, there is no practical example for this transformation. Process mining is able to mine knowledge from event logs and build models. The typical methods of process mining are α-algorithm and Two Phase Approach (Van der Alast 2011). α-algorithm can discover concurrency but unable to consider about the frequencies. Two Phase Approach has the shortcoming of the representational bias which cannot represent the timing aspects. In the practical field of process mining, Thaler et al. (2014) take the data of BPI Challenge 2014. They propose an integrated solution to make a detailed analysis of data. Heidari et al. (2016) take the data of BPI Challenge 2016. They propose a new methodology with three major phases before analysis and present the analysis of the result on the click data. These studies make a complete analysis for data but the underlying relationships between data are not identified. In other words, the modularity is not taken into account.

Fuzzy-DEVS model, as the extension of DEVS model, is able to represent complex systems. Bisgambiglia et al. (2010) present fuzzy modeling in a simple way to define complex system DEVS.

They use fuzzy inference systems (FIS) with DEVS formalism in order to perform the control or the learning on systems. A case study is used to support this approach. Some other studies (Youcef and Maamar 2014; Santucci and Capocchi 2014) propose to use fuzzy reasoning rules to indicate a Fuzzy-DEVS model and apply the corresponding method on real case studies. The shortcoming of these studies is that the Fuzzy-DEVS models are not mined from real data.

3 D2FD METHOD

The D2FD method has three major stages, as presented in Figure 2: (1) from event data to event logs; (2) from event logs to transition system; (3) from transition system to Fuzzy-DEVS model. Here we introduce the general structure of D2FD method.

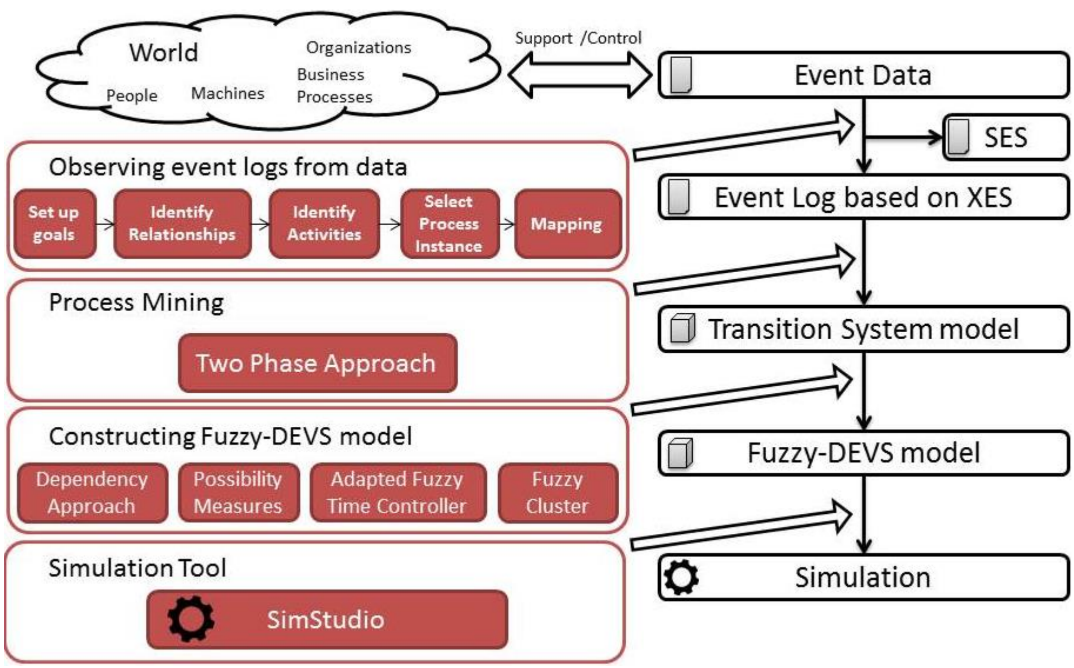

Figure 2: Structure of D2FD method.

3.1 From Event Data to Event Logs

The data is observed from the world in the data level of Figure 1. However, there are some criteria for D2FD method. The event data are the starting point and they are selected through the twelve guidelines from Van der Aalst (2015) and four more guidelines as below.

- All the event data are required to be recorded in the type of csv or excel documents, including start, middle and end documents;
- Activity names (reference names) should be simple, precise and clear;
- Every activity is structured. They can be grouped to attribute, time, case and instance;
- Events are first ordered by instances and then ordered by time (from early to late).

When the event data is observed, five steps are proposed to transform event data to event logs as shown in Figure 2. First, we need to set up the goal by the interview of business people. The goal can be the business problem which we are going to solve. Second, System Entity Structure (SES) (Zeigler et al. 2007) defines a ontological framework in the systems theory. We construct SES from event data in order to discover the relationships between the activities and refine the event data. Third, we identify the activities as well as the modularity. The activities can be identified as public activity and private activity.

If some activities have a strong relationship with some activities in other documents, we can define their children activities as private activities. Conversely, public activities. Fourth, process instance relates to the object that you will follow throughout the process. As we get the hierarchy of SES structure, we select the interesting level according to the goals. This interesting level is related the one of the attributes and contains several activities. Fifth, the activities of interesting attribute, time, case and instance are transformed into the parameters of the event logs.

3.2 From Event Logs to Transition System

The method of transforming from event logs to transition system is coming from Two Phase Approach. Let S^T be the state and T be the transition. In this method, there are different methods in different dimensions to capture states, which leads to different kinds of transition systems. For example, past or future; set abstraction or multi-set abstraction; k-tail method. In this paper, the state is represented by the multi-sets of activities. The transition is discovered between these multi-sets.

3.3 From Transition System to Fuzzy-DEVS Model

Fuzzy-DEVS model is based on the Fuzzy-DEVS formalism (Kwon, Park, Jung and Kim 1996) which applies fuzzy sets theory into the functions of DEVS formalism. It consists of Fuzzy-DEVS atomic model and Fuzzy-DEVS coupled model. A fuzzy atomic model $\tilde{M}$ is characterized by: X is the set of input values; Y is the set of output values; S is the set of states; $\tilde{\delta}_{int}$ is the fuzzy internal transition function; $\tilde{\delta}_{ext}$ is the fuzzy external transition function; $\tilde{\lambda}$ is the fuzzy output function; $\tilde{ta}$ is the fuzzy time advance function. A coupled model DN is characterized by: D is the DEVS components set; EIC is the external input coupling; EOC is the external output coupling; IC is the internal coupling; SELECT is the tie-breaking selector. The method of transforming from transition system to Fuzzy-DEVS model is based on the previous work (Wang et al. 2015). In Fuzzy-DEVS atomic model, an improved region-based approach is defined to specify state. Let TS = (S^T, A, T) be a transition system and $R \subseteq S^T$ be a subset of states. P_a is a period time for each activity $a \in A$. R is a region if for each activity $a \in A$ and one of the following conditions holds:

- All transition $(s_1^T, a, s_2^T) \in T$ enter R, i.e. $s_1^T \notin R$ and $s_2^T \in R$;
- All transition $(s_1^T, a, s_2^T) \in T$ exit R, i.e. $s_1^T \in R$ and $s_2^T \notin R$;
- All transition $(s_1^T, a, s_2^T) \in T$ do not cross R, i.e. $s_1^T, s_2^T \in R$ or, $s_1^T, s_2^T \notin R$;
- For all the transitions $a_1 \in T_1$, $a_2 \in T_2$, …, $a_n \in T_n$ enter R, $P_{a1} \approx P_{a2} \approx \ldots \approx P_{an}$.

Let pa be the private activity and ua be the public activity. According to Fuzzy-DEVS formalism in chapter 3, the transformation follows the rules:

$$R \rightarrow S \tag{1}$$

Where the state of DEVS atomic model $s \in S$.

$$ua \rightarrow x \cup y \tag{2}$$

Where the input value $x \in X$ and the output value $y \in Y$.

$$\widetilde{ta} = \begin{cases} 0 & \exists s_0 \in S \\ & T^F \\ Infinite & \neg\exists S = S_1^I \end{cases} \tag{3}$$

Where s_0 is the initial state, T^F is the result coming from Adapted Fuzzy Time Controller, S_1^I is the input states of all internal transition.

$$\begin{gathered} T \to \tilde{\delta}_{\text{int}} \\ (s_1^T, ua, s_2^T) \to (s_1, s_2, \mu_{\text{int}}) \end{gathered} \tag{4}$$

Where s1 ∈ R1 and s2 ∈ R2, μ_{int} is the result coming from dependency method.

$$\tilde{\lambda} : (y, \mu_{\text{int}}) \tag{5}$$

$$\begin{gathered} T \to \tilde{\delta}_{ext} \\ (s_1^T, pa, s_2^T) \to (s_1, e, x, s_2, \mu_{ext}) \end{gathered} \tag{6}$$

$$\mu(s_i \to s_j) = \begin{cases} \dfrac{F(s_i \to s_j) - F(s_j \to s_i)}{F(s_i \to s_j) + F(s_j \to s_i) + 1} \exists i \neq j \\ \dfrac{F(s_i \to s_j)}{F(s_i \to s_j) + 1} \exists i = j \end{cases} \tag{7}$$

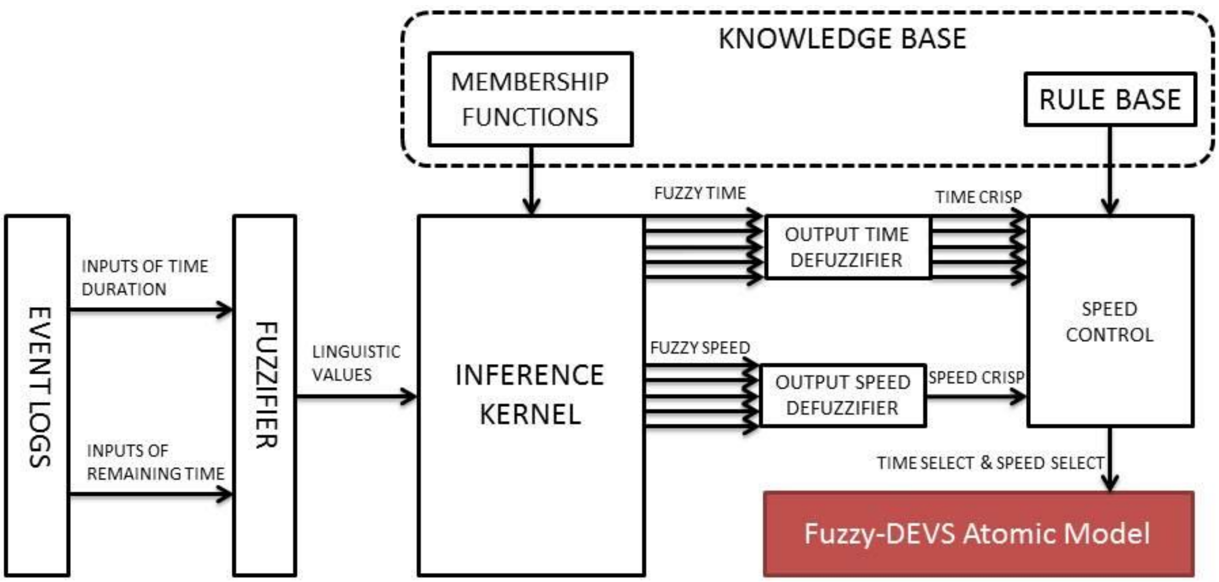

Figure 3: Structure of AFTC.

Where the elapsed time e: $0 \le e \le \tilde{ta}$, μ_{ext} is the result coming from the dependency method. The transition system is first divided into regions then transformed into state. The public activities can be the sets of the input or the output values. If the state is the initial state, the fuzzy time is set as 0. If the state has no internal transition, the fuzzy time is set infinite. Otherwise, the fuzzy time is calculated based on Adapted Fuzzy Time Controller (AFTC) as shown in Figure 3. Time duration and remaining time are the inputs. They first convert into five fuzzy sets by membership function and then defuzzify into five time crisp value and one speed crisp value. They activate the final fuzzy time based on the rule base of speed. If the transition T contains the public activity, it is transformed into fuzzy internal transition. If the transition T contains the private activity, it is transformed into fuzzy external transition. Both the fuzzy internal transition and fuzzy external transition have the membership function μ. It is measured by Dependency Method. The dependency method is based on Equation 7. The fuzzy output function has the same membership function as fuzzy internal transition.

$$E\widetilde{IC} \subseteq \{((N, ip_N), (d, ip_d), (\mu_{EIC}, e)) | \mathrm{ip_N} \in IPorts, d \in D, ip_d \in IPorts_d\} \tag{8}$$

$$E\widetilde{OC} \subseteq \{((d, op_d), (N, op_N), (\mu_{EOC}, e)) | \mathrm{op_N} \in OPorts, d \in D, op_d \in OPorts_d\} \tag{9}$$

$$I\widetilde{C} \subseteq \{((a, op_a), (b, ip_b), (\mu_{IC}, e)) | \mathrm{a, b} \in D, op_a \in Oports_a, ip_b \in IPorts_b\} \tag{10}$$

In Fuzzy-DEVS coupled model, fuzzy cluster is proposed for integration (Wang et al. 2017). Three functions are integrated with membership coefficients. In Equation 8, 9 and 10, μ is the membership coefficient calculated based on Dependency Method and e is the elapsed time. While the time elapse, the membership coefficients may change.

4 CASE STUDIES

4.1 BPI Challenge 2014

The first case study is from BPI Challenge (2014). It covers two parts of an IT Service Management (ITSM) at Rabobank Group ICT. These parts are Change Management and Incident Management from the ITIL framework. One of the goals is to design a predictive model to support Incident Management with less impact of workload at the Service Desk and/or IT operations. There are four case files in CSV. The "incident.csv" and "incident activity.csv" corresponds to the goal. Table 1 shows the attributes of the two files. The gray parts are the associated parts. The relationship of these two files is that "incident activity.csv" is the aspect of "incident.csv" so we focus on "incident activity.csv". In "incident activity.csv", we create a new attribute which is called "DataStampStart" which is taken from the "Open Time" as the first time of each incident and from "Date Stamp" as the next time of each incident. The SES structure of "incident activity.csv" can be mined as presented in Figure 4. All the activities are identified as public activities.

Table 1: The attributes of "incident.csv" and "incident activity.csv".

Incident	Incident activity
CI Name (aff)	Incident ID
CI Type (aff)	DateStamp
CI Subtype (aff)	IncidentActivityNumber
Service Comp (aff)	IncidentActivity Type
Incident ID	Interaction ID
Status	Assignment Group
…	KM number

In Figure 4, the interesting level is the activities level. We propose to select "Assignment", "Communication with customer", "Communication with vendor", "External vendor assignment", "External vendor reassignment" and "Resolved" as the interesting activities. The attributes of "DataStampStart" and "DataStamp" are selected as start timestamp and end timestamp. "Incident ID" is selected as trace.

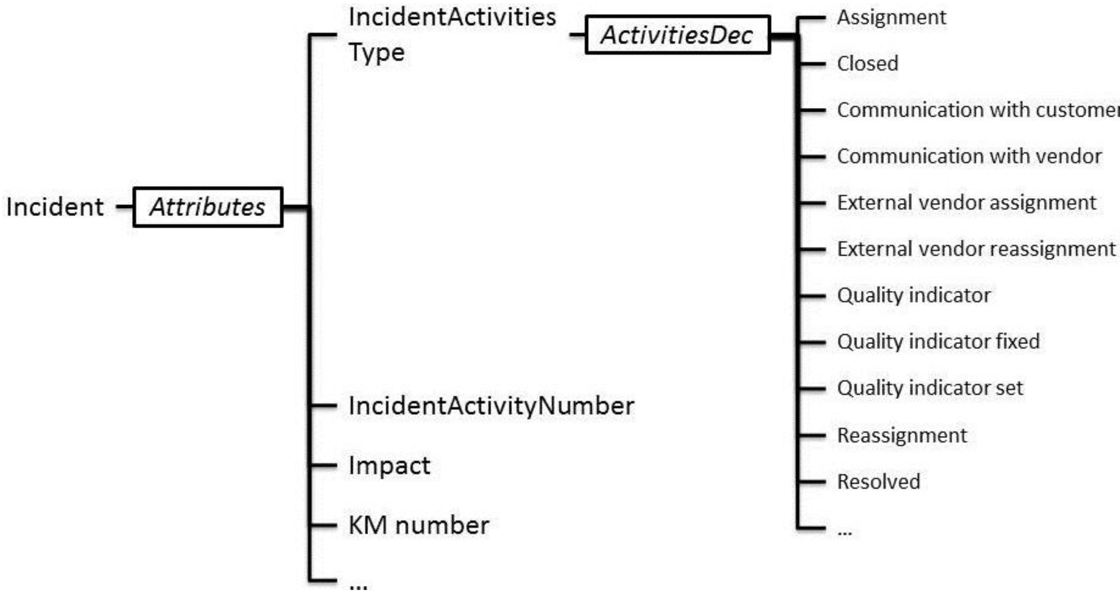

Figure 4: SES structure of “incident activity.csv”.

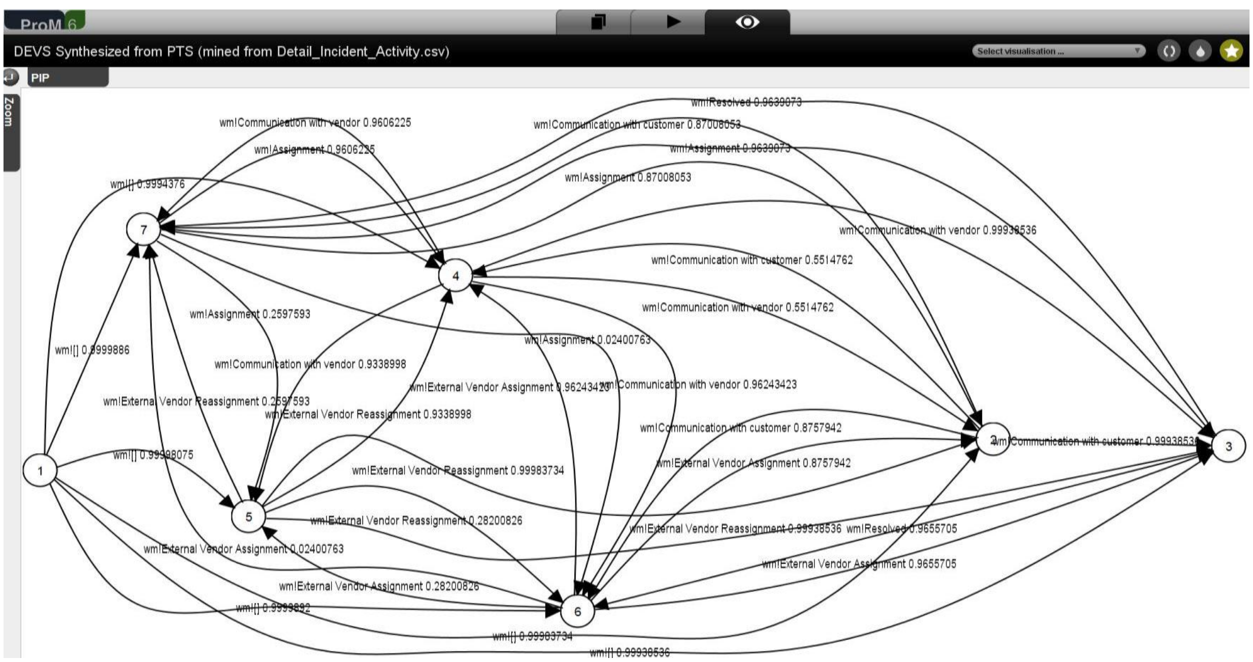

Figure 5: Fuzzy-DEVS model mined from “incident activity.csv”.

```
fuzzy time controller starts
 Time of [[],[]] is 0.0 hours
fuzzy time controller starts
VS: 1.0 S: 0.0 M: 0.0 B: 0.0 VB: 0.0
VL: 0.99 L: 0.01 A: 0.0 H: 0.0 VH: 0.0
fuzzyspeed: 3347302.149042358
Selection of speed: very fast
 Time of [[Assignment],[Assignment+]] is 784.4613222222222 hours
```

Figure 6: Part of fuzzy time results from “incident activity.csv” by using AFTC.

By executing the plugin for mining discrete event simulation model, we get the Fuzzy-DEVS model as shown in Figure 5. The state starts from the initial state 1 until it reaches the end state 3. All the transitions are internal transitions represented as classical arrow. The graphical notation of the internal transition is combined with output port, "!", output function and membership function. There is an output port op_a which is called "wm" used for the output functions. Every state has a fuzzy time. The fuzzy time is calculated by AFTC as shown in Figure 6. Figure 7 presents the simulation results. The number before colon represents the time series of hours. The activities after colon are output function. The time between two time series corresponds to the fuzzy time of the state. When the elapsed time is equal to the fuzzy time, the internal transition with the maximum membership function executes and sends the output function to the output port "wm". For example, the first output is "[]" which has a fuzzy time of 0 in Figure 6. So the time series is 1. The internal transition of "[]" to "Assignment" has the biggest membership functions. After the fuzzy time of "Assignment" 784 hours, it sends out the output function "Assignment" to the port "wm" at the time series of 785. The process of the activities shows the critical workload and the time series show the performance.

```
1 : []
Internal transition: [] & [Assignment] ---- 0.9999886
Internal transition: [] & [External Vendor Reassignment] ---- 0.97727275
Internal transition: [] & [Communication with vendor] ---- 0.9994376
Internal transition: [] & [External Vendor Assignment] ---- 0.9997704
Internal transition: [] & [Resolved] ---- 0.99938536
Internal transition: [] & [Communication with customer] ---- 0.99983734
785 : [Assignment]
Internal transition: [Assignment] & [Communication with customer] ---- 0.87008053
Internal transition: [Assignment] & [External Vendor Reassignment] ---- 0.9990174
Internal transition: [Assignment] & [External Vendor Assignment] ---- 0.90621066
Internal transition: [Assignment] & [Resolved] ---- 0.9639073
Internal transition: [Assignment] & [Communication with vendor] ---- 0.9606225
1439 : [External Vendor Reassignment]
```

Figure 7: Part of simulation results from "incident activity.csv" by using the simulation tool SimStudio.

4.2 BPI Challenge 2016

The second case study is from BPI Challenge (2016). It covers the information of the customers and the records of the telephone calls in the category of question and complaints from an Dutch Employee Insurance Agency. The workbook message is contacted by customers through a digital channel (Here the Dutch language is translated into English). Two main goals can be captured: how the channels are being used; when customers move from one contact channel to next. Based on the goals, we locate on two files "Question.csv" and "Werkmap-message.csv". The corresponding SES structure of "Question.csv" is shown in Figure 8. WN represents this agency and it has several departments as aspects. "Internet Helpdesk" are the most interesting department based on goals. All the activities in this department are listed as aspects without children aspects. All the activities are identified as public activities. The corresponding SES structure of "Werkmap-message.csv" is quite simple. The department of "Internet Helpdesk" only has the activity of "Workbook: message". This activity has variables of "Channel 1" and "Channel 2". As "Workbook: message" has a strong relationship with four activities in Figure 8, the children variables "Channel 1" and "Channel 2" are identified as private activities. The final selected activities of "Question.csv" are: "Register and login", "Registration general", "Registration disturbance", "Other general", "Other disturbance", "Application benefits: general", "Application benefits: disturbance", "Workbook: general", "Workbook: application", "Workbook: taken", "Workbook: disturbance", "Workbook". The final selected activities of "Werkmap-message.csv" are: "Channel 1", "Channel 2".

By executing the plugin for mining a discrete event simulation model, we get the get the Fuzzy-DEVS model from "Question.csv" and the Fuzzy-DEVS model from "Werkmap-message.csv" in Figures 9 and 10. In Figure 9, the state starts from the initial state 1 until it reaches the expired state. All the transitions are internal transitions represented as classical arrow with the graphical notation. Every state has a fuzzy time from AFTC. Part of the fuzzy time in the first atomic model is shown in Figure 11. There is an output port op_a which is called "wm" used for the output functions. In Figure 10, the initial state is state 1. All the transitions are external transitions represented as diamond arrow. The graphical notation of the external transition is combined with the input port, "?", external event and membership function. Event state has an infinite time. There is an input port ip_b called "wm" which is the same as the output port op_a. It is used for the external events to execute these external transitions.

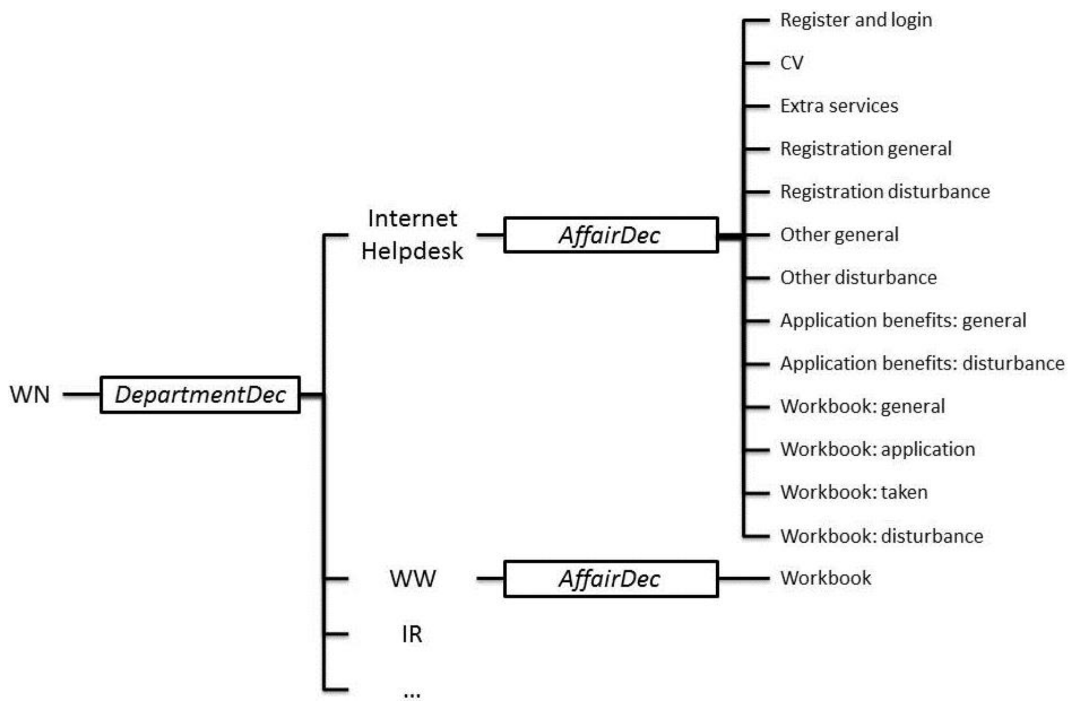

Figure 8: SES structure of "Question.csv".

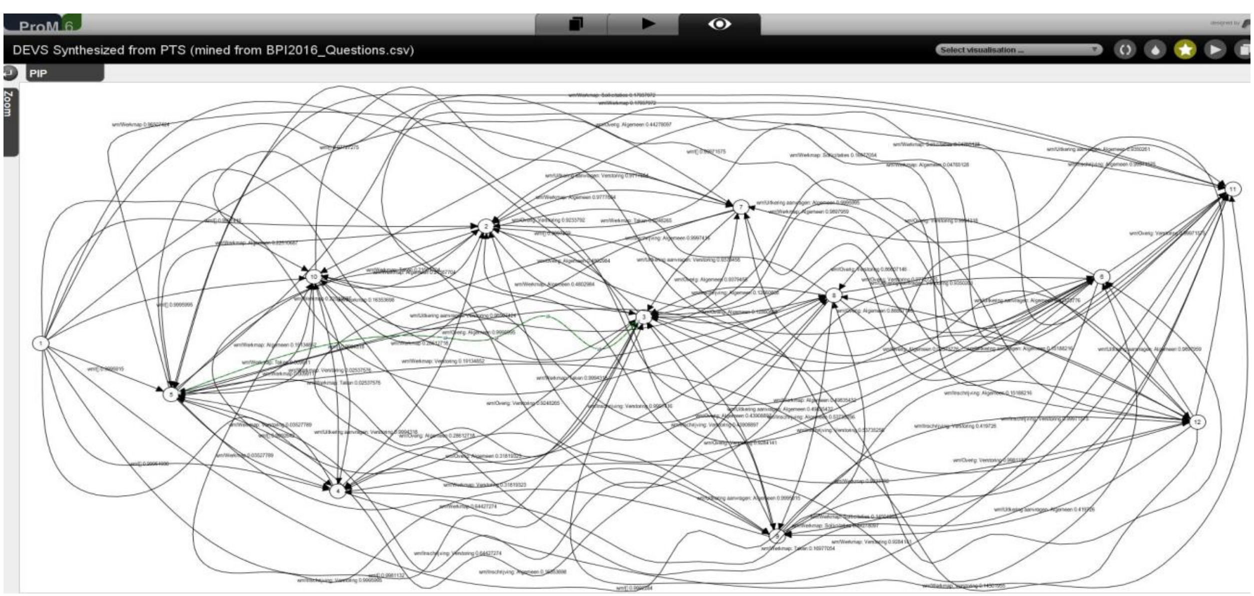

Figure 9: Fuzzy-DEVS atomic model mined from "Question.csv".

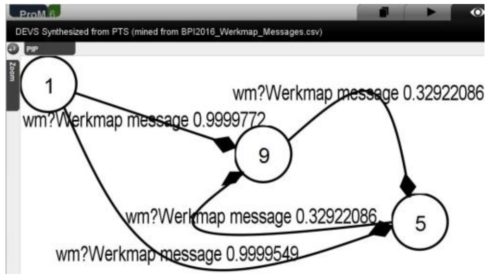

Figure 10: Fuzzy-DEVS atomic model mined from “Werkmap-message.csv”.

```
fuzzy time controller starts
 Time of [[],[]] is 0.0 seconds
fuzzy time controller starts
VS: 0.88 S: 0.0 M: 0.03 B: 0.03 VB: 0.03
VL: 0.82 L: 0.06 A: 0.03 H: 0.03 VH: 0.03
fuzzyspeed: 3234.4312499999996
Selection of speed: very fast
 Time of [[Uitkering aanvragen: Verstoring],[Uitkering aanvragen: Verstoring+]] is 1122.3 seconds
fuzzy time controller starts
VS: 0.99 S: 0.01 M: 0.0 B: 0.0 VB: 0.0
VL: 0.29 L: 0.71 A: 0.0 H: 0.0 VH: 0.0
fuzzyspeed: 3301.161844116149
Selection of speed: fast
 Time of [[Werkmap],[Werkmap+]] is 2397.69 seconds
```

Figure 11: Part of fuzzy time from “Question.csv” by using AFTC.

```
1 : (the output of Question model) []
Internal transition of Question model: [] & [Inschrijving: Algemeen] ---- 0.9878049
Internal transition of Question model: [] & [Werkmap: Sollicitaties] ---- 0.9897959
Internal transition of Question model: [] & [Uitkering aanvragen: Algemeen] ---- 0.9861111
Internal transition of Question model: [] & [Overig: Algemeen] ---- 0.99300706
Internal transition of Question model: [] & [Overig: Verstoring] ---- 0.9896907
Internal transition of Question model: [] & [Werkmap: Verstoring] ---- 0.9944444
Internal transition of Question model: [] & [Werkmap] ---- 0.9995915
Internal transition of Question model: [] & [Aanmelden en inloggen] ---- 0.975
Internal transition of Question model: [] & [Werkmap: Taken] ---- 0.98
Internal transition of Question model: [] & [Werkmap: Algemeen] ---- 0.99519235
Internal transition of Question model: [] & [Inschrijving: Verstoring] ---- 0.98507464
Internal transition of Question model: [] & [Uitkering aanvragen: Verstoring] ---- 0.9714286
40 : (the output of Question model) [Werkmap]
40 : (the state of Werkmap model) Channel 1
External transition of Werkmap model: [] & Channel 1 ---- 0.9999772
External transition of Werkmap model: [] & Channel 2 ---- 0.9999549
```

Figure 12: Part of simulation results from “Question.csv” and “Werkmap-message.csv” by SimStudio.

The simulation of this case study is based on the Fuzzy-DEVS model which the models in Figures 9 and 10 are coupled together. Part of the simulation results is shown in Figure 12. The number before colon represents the time series of minute. The activities after colon are output function in the first atomic model and state in the second atomic model. The state shows the number of the channel which is being used. The time between two time series corresponds to the fuzzy time of the state in the first atomic model. When the elapsed time is equal to the fuzzy time, the internal transition with the maximum membership function executes and sends the output function to the corresponding port “wm”. The second atomic model receives this output function and execute the external transition with the maximum membership function. The state moves to a new state. Based on fuzzy cluster, every activities in the first atomic model has a membership coefficient with external event “Workbook: message”. For example, the fuzzy time of “Werkmap” is almost 39 minutes in Figure 11. After 39 minutes and at the time series of 40, the output function “Werkmap” has a maximum membership coefficient 0.9995915 with “Workbook:

message". So "Werkmap" is sent to the second atomic model and execute the external transition. The state goes to "channel 1" with the maximum membership function of external transition 0.9999772. This proves the activity of "Werkmap" is using channel 1. The process of the simulation reveals the critical activities and handle the two goals of the second case study. The time series show the performance.

5 CONCLUSION

The D2FD method is briefly introduced which extends the knowledge of process mining. The problem of modularity, frequency, time and event data can be solved through this method. This method is implemented as a plugin based on the ProM (Wang 2016). SimStudio is integrated for its simulation. Two case studies illustrates the interoperability and feasibility of this tool. Although the data of two case studies are quite big and are not completely analyzed, the interesting simulation results by using the proposed plugin are able to solve business problem and reveal optimal business processes. This tool is able to identify the underlying variables relationships and make model visual. The identified relationships and the fuzzy cluster can make the complex and separated data connected each other. However, even if the mining process from data to Fuzzy-DEVS atomic model is automatic, Fuzzy-DEVS coupled model is still manual. We anticipate to design a more advanced tool which can mine a complete Fuzzy-DEVS model. The future work will present a detailed analysis of two case studies by applying this tool.

REFERENCES

Bisgambiglia P.A., L. Capocchi, P. Bisgambiglia and S. Garredu. 2010. "Fuzzy Inference Models for Discrete Event Systems". *Fuzzy Systems (FUZZ)*, International Conference on IEEE, 1-8.

10th International Workshop on Business Process Intelligence (BPI) Challenge. 2014. https://data.4tu.nl/repository/uuid:c3e5d162-0cfd-4bb0-bd82-af5268819c35.

12th International Workshop on Business Process Intelligence (BPI) Challenge. 2016. https://data.4tu.nl/repository/uuid:360795c8-1dd6-4a5b-a443-185001076eab.

Heidari F. and N. Assy. 2016. "Usage Analytics Using Process Mining". *12th International Workshop on Business Process Intelligence (BPI).* Key Findings for the Dutch Employee Insurance Agency.

Jans M., M.G. Alles and M.A. Vasarhelyi. 2014. *A Field Study on the Use of Process Mining of Event Logs as an Analytical Procedure in Auditing*. The Accounting Review, 89(5): 1751-1773.

Kaufman L. and P.J. Rousseeuw. 2009. *Finding Groups in Data: an Introduction to Cluster Analysis*. John Wiley & Sons.

Klir G. 2013. *Architecture of Systems Problem Solving*. Springer Science & Business Media.

Kwon Y.W., H.C. Park, S. Jung and T.G. Kim. 1996. *Fuzzy-DEVS Formalism: Concepts, Realization and Application*. Proceedings AIS. 227–234.

Robert M.K. 1976. *Formal Verification of Parallel Programs*. Communications of the ACM. 19(7), 371-384.

Santucci J.F., L. Capocchi and B.P. Zeigler. 2015. "SES Extension to Integrate Abstraction Hierarchy into DEVS Modeling and Simulation". *Proceedings of the Symposium on Theory of Modeling & Simulation: DEVS Integrative M&S Symposium*. Society for Computer Simulation International, 17-24.

Simon H.A. 1991. "The Architecture of Complexity". *Facets of Systems Science.* Springer US, 457-476.

Thaler T., S. Knoch, N. Krivograd, P. Fettke and P. Loos. 2014. "ITIL Process and Impact Analysis at Rabobank ICT". *10th International Workshop on Business Process Intelligence (BPI).*

Traoré M.K. 2008. "Simstudio: A Next Generation Modeling and Simulation Framework". *Proceedings of the 1st International Conference on Simulation Tools and Techniques for Communications, Networks and Systems & Workshops*. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

Van der Aalst W.M.P. 2011. *Process Mining: Discovery, Conformance and Enhancement of Business Processes*. Springer Science & Business Media.

Van der Aalst W.M.P. 2015. "*Extracting Event Data from Databases to Unleash Process Mining*". *BPM-Driving Innovation in a Digital World*. Springer International Publishing, 105-128.

Wang Y., G. Zacharewicz, D. Chen and M.K. Traoré. 2015. "A Proposal of Using DEVS Model for Process Mining". *Proceedings of the European Modeling and Simulation Symposium*. Bergeggi, Italy, 403-409.

Wang Y. 2016. The Subversion Server at the Technical University of Eindhoven. https://svn.win.tue.nl/repos/prom/Packages/TS2DEVS.

Wang Y., G. Zacharewicz, D. Chen and M.K. Traoré. 2017. "Use of Fuzzy Clustering for Discrete Event Simulation Model Consruction". *The 20th World Congress of the International Federation of Automatic Control*. Toulouse, France. 9-14 July.

Youcef D. and H. Maamar. 2014. *Specification of the State Lifetime in the DEVS Formalism by Fuzzy Controller*. Arxiv Preprint Arxiv:1401.5638.

Zeigler B.P., H. Praehofer and T.G. Kim. 2000. *Theory of Modeling and Simulation.* Integrating Discrete Event and Continuous Complex Dynamic Systems. Academic Press, New York.

Zeigler B.P. and P.E. Hammonds. 2007. *Modeling & Simulation-Based Data Engineering: Introducing Pragmatics into Ontologies for Net-Centric Information Exchange*. Academic Press.

AUTHOR BIOGRAPHIES

Yan Wang is an PhD student in the IMS at the University of Bordeaux (France). He graduates from Harbin Institute of Technology with a bachelor of computer science in 2012. He graduates from Harbin Institute of Technology with a master of software engineering and from University of Bordeaux with a master of enterprise engineering in 2014. His research interests include discrete event M&S, process mining, agent-based modeling and enterprise modeling (BPM and ERP). His e-mail address is yan.wang@u-bordeaux.fr.

Grégory Zacharewicz is Associate Professor HDR at the University of Bordeaux (France). His research interests include discrete event M&S, distributed simulation (HLA), model driven approaches, semantics aspects (short-lived ontology), and enterprise modeling (workflow, BPM, and ERP). He focused recently on Social-Organizational M&S and Interoperability. He is involved in several M&S conferences and journals. He has been participating in a number of French, European and transatlantic M&S projects. His email address is gregory.zacharewicz@u-bordeaux.fr.

Mamadou Kaba Traoré received his MSc (1989) and PhD (1992) from Blaise Pascal University (France) in Computer Science. He is currently heading the Web & Mobile Software Engineering MSc degree at Université Clermont Auvergne (France). He is also adjunct Professor at the African University of Science and Technology. His current research is on formal specifications, symbolic manipulation and automated code synthesis of simulation models. His email address is traore@isima.fr.

David Chen is professor at University of Bordeaux. His research interests focus on Production Management, Enterprise modelling, integration and interoperability. He has been actively involved since 1985 in multiple EU R&D programmes (FP1-FP7), and several co-operation programmes between EU and China. David CHEN is convenor of CEN TC 310/WG1 and member of ISO TC184/SC5/WG1 (Modelling and architecture). He is involved in IFIP WG5.12 and IFAC WG5.3 (Enterprise Integration and Networking) as well as IFIP WG 5.8 (Enterprise interoperability). He has published more than 160 papers in international journals and conferences. His email address is david.chen@u-bordeaux.fr.