Modeling a Fuzzy Resource Allocation Mechanism based on Workflow Nets

Joslaine Cristina Jeske de Freitas[1], Stéphane Julia[1] and Leiliane Pereira de Rezende[1,2]
[1]*Computing Faculty, Federal University of Uberlândia - UFU, Uberlândia - MG, Brazil*
[2]*DMIA Department, Institut Supérieur de l'Aéronautique et de l'Espace - ISAE, Toulouse, France*

Keywords: Petri Net, Workflow Net, Resource Allocation, Possibility Theory, CPN Tools.

Abstract: Colored Petri Nets (CPN) arose from the need to model very large and complex systems, which are found in real industrial applications. The idea behind CPN is to unite the ability to represent synchronization and competition for resources of Petri nets with the expressive power of programming languages, data types and diverse abstraction levels. This work proposes the use of Hierarchical Colored Petri Nets and CPN Tools to model a Workflow net with fuzzy sets delimited by possibility distributions associated with the Petri net models that represent human type resource allocation mechanisms. Additionally, the duration of activities that appear on the routes (control structure) of the Workflow process, will be represented by fuzzy time intervals. Besides, firing rules based on a joint possibility distribution will be used in order to express in a more realistic way the resource allocation mechanisms when human behavior is considered in Workflow activities.

1 INTRODUCTION

The purpose of Workflow Management Systems is to execute Workflow processes. Workflow processes represent the sequence of activities that have to be executed within an organization to treat specific cases and to reach a well-defined goal. Of all notations used for the modeling of Workflow processes, Petri nets are very suitable (Aalst and van Hee, 2004), as they represent basic routings of Business Processes. Moreover, Petri nets can be used for specifying the real time characteristics of Workflow Management Systems (in the time Petri net case) as well as complex resource allocation mechanisms. As a matter of fact, late deliveries in an organization are generally due to resources overload.

Many papers have already considered the Petri net theory as an efficient tool for the modeling and analysis of Workflow Management Systems. In (Aalst and van Hee, 2004), Workflow nets, which are acyclic Petri net models used to represent the Workflow process, are defined. Workflow nets have been identified and widely used as a solid model of Workflow processes by several authors (Aalst, 1997), (Ling and Schmidt, 2000), (Kotb and Badreddin, 2005), (van Hee et al., 2006), (Martos-Salgado and Rosa-Velardo, 2011), (Wang et al., 2009), (Wang and Li, 2013). In (Ling and Schmidt, 2000), an extension of Workflow nets is presented. This model is called time Workflow net and associates time intervals with the transitions of the corresponding Petri net model. In (Kotb and Badreddin, 2005), an extended Workflow Petri net model is defined. Such a model allows for the treatment of critical resources which have to be used for specific activities in real time. In (Wang et al., 2009), a resource-oriented Workflow net (ROWN) based on a two-transition task model was introduced for resource-constrained Workflow modeling and analysis. Considering the possibility of task failure during execution, in (Wang and Li, 2013), a three-transition task model for specifying a task start, end and failure was proposed.

The majority of existing models put their focus on the process aspect and do not consider important characteristics of the Workflow Management System. In (Aalst, 1997) and (van Hee et al., 2006) for example, the resource allocation mechanisms are represented only in an informal way. In (Ling and Schmidt, 2000), (Kotb and Badreddin, 2005) and (Wang and Li, 2013) resource allocation mechanisms are represented by simple tokens in places as is generally the case in production systems (Lee and DiCesare, 1994). But a simple token in a place will not represent in a

---

The third author received CAPES Scholarship - Proc. n°. 99999.001925/2015-06.

Freitas, J., Julia, S. and Rezende, L.
Modeling a Fuzzy Resource Allocation Mechanism based on Workflow Nets.
In *Proceedings of the 18th International Conference on Enterprise Information Systems (ICEIS 2016) - Volume 2*, pages 559-566
ISBN: 978-989-758-187-8

realistic way human employees who can treat simultaneously different cases in a single day, as is usually the case in most Business processes.

Colored Petri Nets (CPN) can be used graphical modeling language for managing business processes (Serral et al., 2014). Business processes often appear in dynamic environments, for this reason, context adaptation has recently emerged as a new challenge to explicitly address fitness between Business process modeling and its execution environment. Initially, CPNs were supported by Design/CPN. Later, Design/CPN was replaced by CPN Tools.

This work proposes the use of Hierarchical Colored Petri Nets and CPN Tools to model a Workflow net with fuzzy sets delimited by possibility distributions associated with the Petri net models that represent human type resource allocation mechanisms. Additionally, the duration of activities that appear on the routes (control structure) of the Workflow process, will be represented by fuzzy time intervals. Besides, firing rules based on a joint possibility distribution will be used in order to express in a more realistic way the resource allocation mechanisms when human behavior is considered in Workflow activities.

The remainder of this paper is set out as follows. Section 2 shows the concepts the Workflow net. Section 3 presents concepts of CPN. Section 4 introduces the concepts of fuzzy sets and possibility measures. Section 5 presents resource allocation mechanisms. Section 6 presents a fuzzy time constraint propagation mechanism. Section 7 defines firing rules that consider fuzzy time constraints as well as fuzzy resource allocation mechanisms. Section 8 shows the CPN Model for “Handle Complaint Process”. Finally, section 9 concludes the paper and provides references for additional works.

2 WORKFLOW NET

A Petri net that models a Workflow process is called a Workflow net (Aalst and van Hee, 2004). A Workflow net satisfies the following properties (Aalst, 1998):

- It has only one source place, named *Start* and only one sink place, named *End*. These are special places such that the place *Start* has only outgoing arcs and the place *End* has only incoming arcs.
- A token in *Start* represents a case that needs to be handled and a token in *End* represents a case that has been handled.
- Every task t (transition) and condition p (place) should be on a path from place *Start* to place *End*.

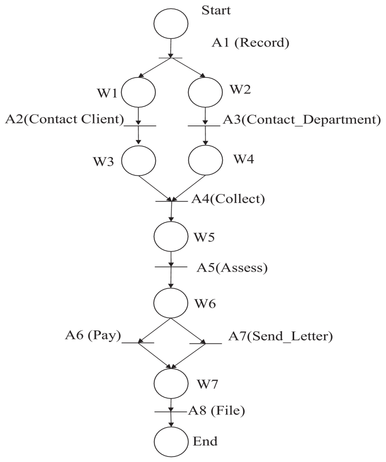


Figure 1: “Handle Complaint Process”.

Figure 1 illustrates the Workflow process which takes care of the processing of claims related to car damage (Aalst and van Hee, 2004). An incoming complaint is first recorded. Then the client who has complained along with the department affected by the complaint are contacted. The client is approached for more information. The department is informed of the complaint and may be asked for its initial reaction. These two tasks may be performed in parallel, i.e. simultaneously or in any order. After this, data is gathered and a decision is made. Depending upon the decision, either a compensation payment is made or a letter is sent. Finally, the complaint is filed.

3 CONCEPTS OF CPN

CPN is a graphical modeling language (Jensen and Kristensen, 2009), which combines the strengths of Petri nets (Wolfang Reisig, 2013) and of functional programming languages (Milner et al., 1997). The formalism of Petri nets is well suited for describing concurrent and synchronizing actions in distributed systems. Programming languages can be used to define data types and to manipulate data. An introduction to the practical use of CPN can be found in (Kristensen et al., 2004).

The CPN are designed to reduce the size of the model, allowing individualization of tokens thigh the use of colors assigned to them. Different processes or resources can then be represented in the same network. Colors do not mean just colors or patterns. They can represent complex data types (Jensen and Kristensen, 2009).

CPN models are formal, in the sense that the CPN modeling language has a mathematical definition of its syntax and semantics. This means that they can be used to verify system properties, i.e., prove that certain desired properties are fulfilled or that certain undesired properties are guaranteed to be absent.

Large and complex models can be built using hierarchical CPN in which modules, which are called pages in the CPN terminology, are related to each other in a well-defined way. Without the hierarchical structuring mechanism, it would be difficult to create understandable CPN models of real-world systems (Jensen and Kristensen, 2009).

4 FUZZY SETS AND POSSIBILITY MEASURES

The notion of fuzzy set was introduced by (Zadeh, 1965) in order to represent the gradual nature of human knowledge. For example, the size of a man could be considered by the majority of a population as small, normal, tall, etc. A certain degree of belief can be attached to each possible interpretation of symbolic information and can simply be formalized by a fuzzy set F of a reference set X that can be defined by a membership function $\mu_F(x) \in [0,1]$. In particular, for a given element $x \in X$, $\mu_F(x) = 0$ denote that x is not a member of the set F, $\mu_F(x) = 1$ denotes that x is definitely a member of the set F, and intermediate values denote the fact that x is more or less an element of F. Normally, a fuzzy set is represented by a trapezoid $A = [a1, a2, a3, a4]$ where the smallest subset corresponding to the membership value equal to 1 is called the core, and the largest subset corresponding to the membership value greater than 0 is called the support.

A fuzzy set F can be delimited by a possibility distribution Π_f, such as: $\forall x \in X, \Pi_f(x) = \mu_F(x)$ (Dubois and Prade, 1988), (Cardoso et al., 1999). Given a possibility distribution $\Pi_a(x)$, the measure of possibility $\Pi(S)$ and necessity $N(S)$ that a data a belongs to a crisp set S of X is defined by $\Pi(S) = sup_{x\in S}\Pi_a(x)$ and $N(S) = inf_{x\notin S}(1-\Pi_a(x)) = 1 - \Pi(\bar{S})$. If $\Pi(S) = 0$, it is impossible that a belongs to S. If $\Pi(S) = 1$, it is possible that a belongs to S, but it also depends on the value of $N(S)$. If $N(S) = 1$, it is certain that (the larger the value of $N(S)$, the more the proposition is believed in). In particular, there exists a duality relationship between the modalities of the possible and the necessary which postulates that an event is necessary when its contrary is impossible. Some practical examples of possibility and necessity measures are presented in (Dubois and Prade, 1988).

Given two data a and b characterized by two fuzzy sets A and B the measure of possibility and necessity of having $a \le b$ are defined as:

$$\Pi(a \le b) = sup_{x\le y}(min(\Pi_a(x), min(\Pi_b(y))) = max([A,+\infty[\cap]-\infty,B]) \quad (1)$$

and

$$N(a \le b) = 1 - sup_{x\le y}(min(\Pi_a(x), min(\Pi_b(y))). \quad (2)$$

Given a normalized possibility distribution π_a, (Dubois and Prade, 1989) defines the following fuzzy sets of the time point that are:

- possibly after a: $\mu_{[A,+\infty[}(x) = sup_{x\in X}\pi_a(s)$
- necessarily after a: $\mu_{]A,+\infty[}(x) = inf_{x\in X}(1-\pi_a(s))$
- possibly before a: $\mu_{]-\infty,A]}(x) = sup_{x\in X}\pi_a(s)$
- necessarily before a: $\mu_{]-\infty,A[}(x) = inf_{x\in X}(1-\pi_a(s))$

A visibility time interval $[a,b]$ is a period of time between two dates a and b. In the case where a and b are fuzzy dates A and B (delimited by π_a and π_b) respectively, the interval $[a,b]$ is represented by the following pair of fuzzy sets:

- $[A,B]$, the conjunctive set of time instants that represents the set of dates possibly after A and possibly before B;
- $]A,B[$, the conjunctive set of time instants that represents the set of dates necessarily after A and necessarily before B.

The joint possibility admits as upper bound in (Dubois and Prade, 1988):

$$\forall x \in X \quad \forall y \in Y \quad \pi(x,y) = min(\pi_X(x), \pi_Y(y)) \quad (3)$$

when the reference sets are non-interactive (the value of x in X has no influence on the value of y in Y, and vice versa).

5 RESOURCE ALLOCATION MECHANISM

Resources in Workflow Management Systems are non-preemptive (Aalst and van Hee, 2004): once a resource has been allocated to a specific activity, it cannot be free before ending the corresponding activity. As already mentioned, there exists different kinds of resources in Workflow processes. Some of which are of the discrete type and can be represented by a simple token. For example, a printer used to treat a specific class of documents will be represented as a non-preemptive resource and could be allocated to a single document at a same time. On the contrary, some other resources cannot be represented by simple tokens. This is generally the case with human type resources. As a matter of fact, it is not unusual for an employee who works in an administration to treat several cases simultaneously. For example, in an insurance company, a single employee can normally treat several documents during a working day and not necessarily in a pure sequential order. In this case, a simple token could not model human behavior in a proper manner. Fuzzy allocation mechanisms were presented in (Jeske et al., 2009).

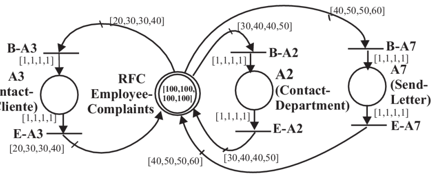


Figure 2: Fuzzy Continuous Resource.

An example of fuzzy continuous resource is given in Figure 2. For example, this Figure shows that 30% ± 10% of the resource availability $R2$ is necessary to realize the activity $A3$ (Contact-Client).

The behavior of a fuzzy continuous resource allocation model can be defined through the concepts of "enabled transition" and "fundamental equation".

In an ordinary Petri net, a transition t is enabled if and only if for all the input places p of the transition, $M(p) \geq Pre(p,t)$, which means that the number of tokens in each input place is greater or equal to the weight associated to the arcs which connect the input places to the transition t. With a fuzzy continuous resource allocation mechanism, considering a transition t, the marking of an input place p and the weights associated to the arc which connects this place to the transition t are defined through different fuzzy sets. In this case, a transition t is enabled if and only if (for all the input places of the transition t):

$$\Pi_t = \Pi(Pre_{FCR}(p,t) \leq M_{FCR}(p)) > 0. \tag{4}$$

For an ordinary Petri net, once a transition is enabled by a marking M, it can be fired and a new marking M' is obtained according to the fundamental equation:

$$M'(p) = M(p) - Pre(p,t) + Pos(p,t). \tag{5}$$

With a fuzzy continuous resource allocation model, the marking evolution is defined through the following fundamental equation:

$$M'_{FCR}(p) = M_{FCR}(p) \ominus Pre_{FCR}(p,t) \boxplus Pos_{FCR}(p,t) \tag{6}$$

The operation "$\ominus$" corresponds to the fuzzy subtraction. The operation "$\boxplus$", when considering the sum of two fuzzy sets, is different from the one given in fuzzy logic and is defined as:

$$[a1,a2,a3,a4] \boxplus [b1,b2,b3,b4] = [a1+b1,a2+b2,a3+b3,a4+b4].$$

This difference is due to the fact that the fuzzy operation "$\oplus$" does not maintain the marking of the fuzzy continuous resource allocation model invariant (the p-invariant property of the Petri net theory (Murata, 1989)). As a matter of fact, after realizing different activities, the resource's availability must go back to 100 %, even in the fuzzy case. To a certain extent, from the point of view of fuzzy continuous resource allocation mechanisms, the operation "$\boxplus$" can be seen as a kind of defuzzyfication operation.

6 FUZZY TIME CONSTRAINT PROPAGATION MECHANISM

As the actual time required by an activity in a Workflow Management System is non-deterministic and not easily predicted, a fuzzy time interval can be assigned to every Workflow activity.

The static definition of a fuzzy time Workflow net is based on fuzzy static intervals $[a1,a2,a3,a4]s$ which represent the permanency duration (sojourn time) of a token in places. Before duration $a1$ the token is in the non-available state. After $a1$ and before $a4$, the token is in the available state for the firing of a transition. After $a4$, the token is again in the non-available state and cannot enable any transition: it therefore becomes a dead token. In a real time system case, the "death" of a token has to be seen as a time constraint that is not respected. A transition cannot be fired with dead tokens as this would correspond to an illegal action or behavior: a constraint violation. The dynamic evolution depends on the time situation of the tokens (date intervals associated with the tokens).

In a Workflow Management System, a visibility interval depends on a global clock associated to the entire net which calculates the passage of time from date $= 0$, which corresponds to the start of the systems operation. In particular, the existing waiting times between sequential activities can be represented by visibility intervals whose minimum and maximum fuzzy boundaries will depend on the earliest and latest delivery dates of the considered case. Through correct knowledge of the beginning date of the process and the maximum duration of a case, it is possible to calculate estimated visibility intervals associated with each token in each waiting place using constraint propagation techniques very similar to the ones used in scheduling problems based on activity-on-arc graphs without circuits (Gondran et al., 1984).

Figure 3 shows the fuzzy static intervals (intervals of fuzzy durations) associated to the activity places

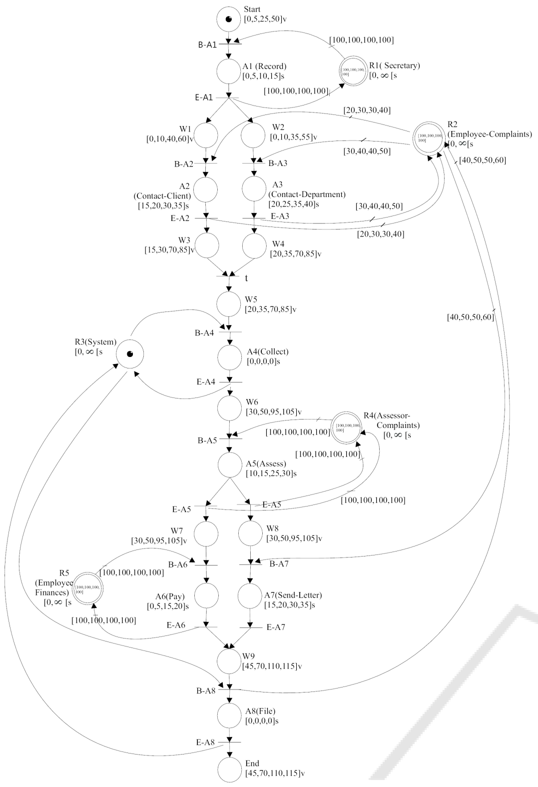


Figure 3: Visibility intervals of the “Handle Complaint Process”.

of the process and the fuzzy visibility intervals (intervals of fuzzy dates) associated with the waiting places (condition places of the Workflow net). It is important to note that there is no time restriction on resources - static interval defined for each resource is $[0,\infty[s$. The minimal fuzzy bounds of the estimated visibility intervals attached to the waiting places are calculated applying a forward constraint propagation technique applied to the different kinds of routings associated with the “Handle Complaint Process”, and the maximum fuzzy bounds of the estimated visibility intervals are calculated by applying a backward constraint propagation techniques to the different kinds of routings considering the latest delivery dates of the case.

If the token appears in a place p at date δ and if its visibility interval is given by $[a1,a2,a3,a4]$, then this token could be used for the firing of a transition at the earliest date $a1$ and at the latest date $a4$. The global state of the Workflow net will be then defined by the current marking of the net and by the time measured by the clock through the different visibility intervals. When a transition t is fired at a date which belongs to its enabling interval, a new marking will be calculated, the tokens which will not be used for the firing of the transition will continue with their visibility interval, and new estimated visibility intervals will be associated to the tokens produced by the firing of t.

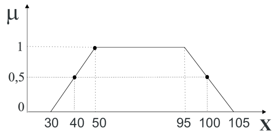


Figure 4: Possibilistic Distribution of W6.

For example, if a token is produced in place $W6$ at date $\delta = 50$, considering the possibilistic distribution shown in Figure 4, the firing possibility measure of transition $B\text{-}A5$ will be equal to $\mu = 1$ and the activity associated to place $A5$ will be initiated normally and its visibility interval will be $[50,50,50,50]$ (firing of $B\text{-}A5$) $\oplus$ $[10,15,25,30]s$ (static interval associated to $A5$) $= [60,65,75,80]v$. If the token in place $W6$ is produced earlier at date $\delta = 40$, for example, the firing possibility of transition $B\text{-}A5$ will be $\mu = 0.5$ (see Figure 4) and its visibility interval will be $[40,40,40,40]$ (firing of $B\text{-}A5$) $\oplus$ $[10,15,25,30]s$ (static interval associated to $A5$) $= [50,55,65,70]v$. However the firing could eventually be delayed until reaching a date corresponding to a possibility equal to $\mu = 1$. Finally, if the token in place $W6$ is produced at date $\delta = 100$, the firing possibility of transition $B\text{-}A5$ will be equal to $\delta = 0.5$ (see Figure 4) but with a different meaning. This situation will correspond to a case where some of the previous activities on the process were delayed and its visibility interval will be $[100,100,100,100]$ (firing of $B\text{-}A5$) $\oplus$ $[10,15,25,30]s$ (static interval associated to $A5$) $= [110,115,125,130]v$. It will be important then to immediately fire the transition $B\text{-}A5$ corresponding to the beginning of the next activity and to inform the responsible resource for executing this activity of the delay. Eventually, some of the next activities of this process will be executed with a high rank priority and the firing possibility of some of the last transitions in the process will reach a possibility $\mu = 1$ again, ensuring that the process deadline is respected. The complete definition of fuzzy time constraint propagation mechanism can be found in (Jeske de Freitas and Julia, 2015).

7 FIRING RULES WITH FUZZY TIME AND FUZZY RESOURCE

If a transition has n input places and if each one of these places has several tokens in it, then the enabling time interval $[a1,a2,a3,a4]$ of this transition is obtained by choosing for each one of these n input places a token, the visibility interval associated with it. In this paper, there exists no time restriction on the resources (the static interval) attached to the resource places is always $[0, \infty[s$ and, as a consequence, the enabling time interval of a transition will simply be equal to the visibility interval associated with the case to be treated by the corresponding transition. For example, knowing that the visibility interval attached to the case represented by a token in place $W1$ is equal to $[0, 10, 40, 60]v$, the enabling time interval of the transition B-$A2$ will be $[0, 10, 40, 60]v$ too.

For firing a transition, it is necessary that the arrival date of the token in the input place of the transition belongs to the fuzzy visibility interval associated with the input place of the transition ($\mu > 0$) and the resource availability (equation (1)) necessary to realize the activity initiated by the firing of the transition must be greater than 0 ($\Pi(a \leq b) > 0$). To evaluate the availability of resource and time simultaneously, the joint possibility presented in equation (3) must be calculated, where $\pi_X(x)$ corresponds to the resource availability and $\pi_Y(y)$ to the time location of the corresponding activity. A complete definition of firing rules with fuzzy time and fuzzy resource can be found in (Jeske de Freitas et al., 2015).

8 CPN MODEL FOR “HANDLE COMPLAINT PROCESS”

The modeling approach is hierarchical, where each transition can be replaced by a specific module. The most abstract level of the modeling of the “Handle Complaint Process” is depicted in Figure 5.

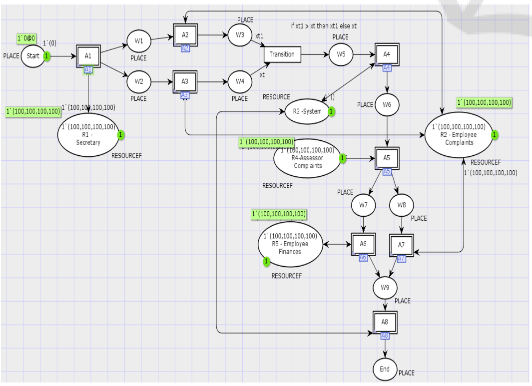


Figure 5: The most abstract level of “Handle Complaint Process’.

Observing Figure 5, note that:

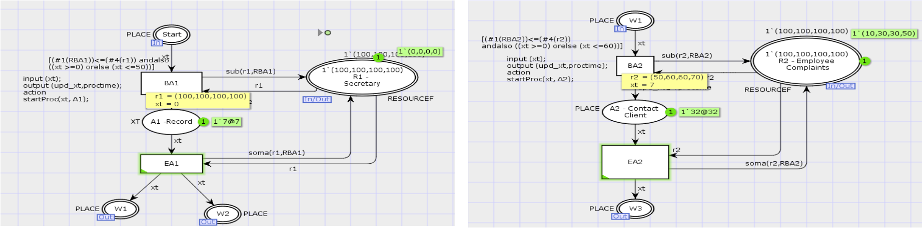
- A token in place *Start* enables the transition that defines the A1 activity for firing.
- The colset *PLACE* is a set of colors that defines the types associated with the places of the model. In addition, there are two types of resources defined in the model: RESOURCE that defines discrete resource and RESOURCEF that defines the fuzzy resources.
- Each of the transitions with double borders denominated as “abstract transition” create a link with one of the sub-networks (it corresponds to the hierarchy concept) $A1$, $A2$, $A3$, $A4$, $A5$, $A6$, $A7$ and $A8$. As already mentioned, A2 and A3 are parallel activities. At the end of these activities, a synchronization must be made. In this case, it is necessary to compare the end time of each activity and must use the higher value. In the proposed model, this condition is defined with the subscription expression “*if xt*1 > *xt then xt1 else xt*” in the output arc.

Basically, each activity is defined as an input transition bounded by the guard condition that checks if there is resource availability and if the available time is within the visible interval. If the transition is fired then *Startproc function* is called and the resource is used. The *SartProc function* takes as input the current value of time and the interval that defines the duration of the activity, and returns the value of time to be updated after the end of the activity. When the resource is used, the *sub function* is called to calculate the new value of resource availability. When the activity ends, the value of the resource is returned by *soma function*.

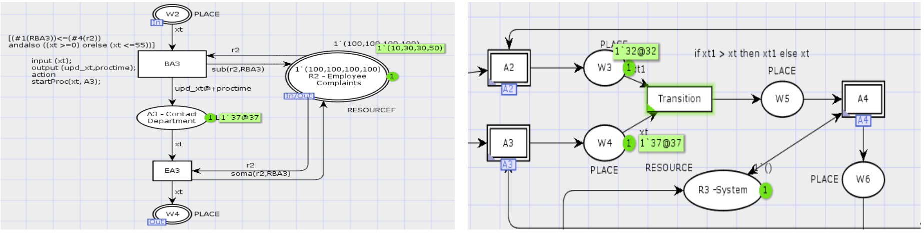
Figure 6 shows the simulation of the following scenario. An incoming complaint is first recorded (Figure 6(a)). Then the client who has complained along with the department affected by the complaint are contacted. The client is approached for further information (Figure 6(b)). The department is informed of the complaint and may be asked for its initial reaction (Figure 6(c)). These two tasks may be performed in parallel, i.e. simultaneously or in any order (Figure 6(d)). After this, data is gathered (Figure 6(e)) and a decision is made. After a decision is reached, a compensation payment is made (Figure 6(f)). Finally, the complaint is filed (Figure 6(h)).

9 CONCLUSIONS

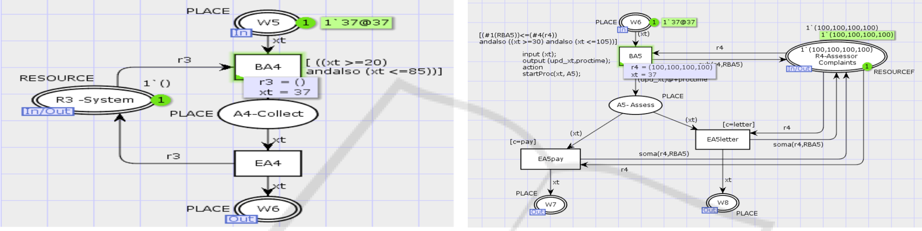
This article presented how to model fuzzy hybrid resources in Workflow nets with fuzzy time intervals associated to the activities using a CPN Tools. Besides this, through the definition as well as use of a joint possibility distribution, it was possible to define a transition firing definition. This definition takes into consideration the time constraints associated with the cases of the process as well as the availability of the resources used to execute the activities.



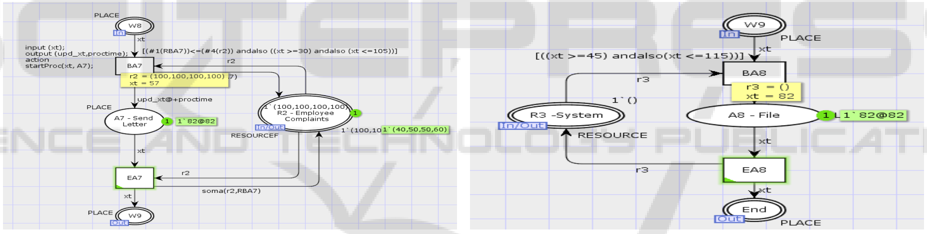
(a) Activity A1 (b) Activity A2



(c) Activity A3 (d) Sync Activity



(e) Activity A4 (f) Activity A5



(g) Activity A7 (h) Activity A8

Figure 6: Simulation of "Handle Complaint Process".

As a future work proposal, it will be interesting to build an intelligent model based on event logs to define the best option for firing a transition. For example, the event log will show the possibilities for each activity firing and may lead to a type of process quality analysis: if the activities, most of the time, are working with a possibility equal to 1, then the work resulting from the process will be of good quality. On the other hand, if a large number of the activities are associated with possibilities near to 0, then the quality of the process will be of poor quality. In addition, during the execution of process activities, the management of activities could suffer a certain influence according to the semantics associated with a low firing possibility. Finally, in the case of transitions in conflict, the information concerning the firing possibility can be used to make a decision: for example if the possibility is low because of delayed activities, we will give priority to the transition in relation to another that possesses a higher firing possibility.

ACKNOWLEDGMENT

The authors would like to thank FAPEMIG (Fundação de Amparo a Pesquisa de Minas Gerais), CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) and CNPq (Conselho Nacional de Pesquisa) for their financial support.

REFERENCES

Aalst, W. (1998). The Application of Petri Nets to Workflow Management. *Journal of Circuits, Systems, and*

Computers, 8(1):21–66.

Aalst, W. and van Hee, K. (2004). *Workflow Management: Models, Methods, and Systems*. MIT Press, Cambridge, MA, USA.

Aalst, W. M. P. v. d. (1997). Verification of workflow nets. In *Proceedings of the 18th International Conference on Application and Theory of Petri Nets*, ICATPN '97, pages 407–426, London, UK, UK. Springer-Verlag.

Cardoso, J., Valette, R., and Dubois, D. (1999). Possibilistic petri nets. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, 29(5):573–582.

Dubois, D. and Prade, H. (1988). *Possibility theory*. Plenum Press, New-York.

Dubois, D. and Prade, H. (1989). Processing fuzzy temporal knowledge. In *Systems, Man and Cybernetics*, pages 729–744. IEEE.

Gondran, M., Minoux, M., and Vajda, S. (1984). *Graphs and Algorithms*. John Wiley & Sons, Inc., New York, NY, USA.

Jensen, K. and Kristensen, L. (2009). *Coloured Petri Nets*. Springer.

Jeske, J. C., Julia, S., and Valette, R. (2009). Fuzzy continuous resource allocation mechanisms in workflow management systems. In *Software Engineering, 2009. SBES'09. XXIII Brazilian Symposium on*, pages 236–251. IEEE.

Jeske de Freitas, J. C. and Julia, S. (2015). Fuzzy time constraint propagation mechanism for workflow nets. In *12th International Conference on Information Technology - New Generations (ITNG)*, pages 367–372. International Conference on Information Technology - New Generations (ITNG).

Jeske de Freitas, J. C., Julia, S., and de Rezende, L. P. (2015). Fuzzy resource allocation mechanisms in workflow nets. In *ICEIS 2015 - Proceedings of the 17th International Conference on Enterprise Information Systems, Volume 1, Barcelona, Spain, 27-30 April, 2015*, pages 471–478.

Kotb, Y. T. and Badreddin, E. (2005). Synchronization among activities in a workflow using extended workflow petri nets. In *CEC*, pages 548–551. IEEE Computer Society.

Kristensen, L., Jørgensen, J. B., and Jensen, K. (2004). Application of coloured petri nets in system development. In *In Lecture on Concurrency and Petri Nets, Jorg Desel, Wolfgang Reisig and Grezegorz Rozenberg (Eds.), Springer, LNCS 3089*, pages 626–685. Springer-Verlag.

Lee, D. Y. and DiCesare, F. (1994). Scheduling flexible manufacturing systems using petri nets and heuristic search. *IEEE T. Robotics and Automation*, 10(2):123–132.

Ling, S. and Schmidt, H. (2000). Time Petri nets for workflow modelling and analysis. In *2000 IEEE International Conference on Systems, Man and, Cybernetics*, volume 4, pages 3039–3044. IEEE.

Martos-Salgado, M. and Rosa-Velardo, F. (2011). Dynamic soundness in resource-constrained workflow nets. In Bruni, R. and Dingel, J., editors, *FMOODS/FORTE*, volume 6722 of *Lecture Notes in Computer Science*, pages 259–273. Springer.

Milner, R., Tofte, M., and Macqueen, D. (1997). *The Definition of Standard ML*. MIT Press, Cambridge, MA, USA.

Murata, T. (1989). Petri nets: Properties, analysis and applications. *Proceedings of the IEEE*, 77(4):541–580.

Serral, E., Smedt, J. D., and Vanthienen, J. (2014). Extending cpn tools with ontologies to support the management of context-adaptive business processes. In Fournier, F. and Mendling, J., editors, *Business Process Management Workshops*, volume 202 of *Lecture Notes in Business Information Processing*, pages 198–209. Springer.

van Hee, K., Sidorova, N., and Voorhoeve, M. (2006). Resource-constrained workflow nets. *Fundam. Inf.*, 71(2,3):243–257.

Wang, J. and Li, D. (2013). Resource oriented workflow nets and workflow resource requirement analysis. *International Journal of Software Engineering and Knowledge Engineering*, 23(5):677–694.

Wang, J., Tepfenhart, W. M., and Rosca, D. (2009). Emergency response workflow resource requirements modeling and analysis. *IEEE Transactions on Systems, Man, and Cybernetics, Part C*, 39(3):270–283.

Wolfang Reisig (2013). *Understanding Petri Nets: Modeling Techniques, Analysis Methods, Case Studies*. Springer. 230 pages; ISBN 978-3-642-33277-7.

Zadeh, L. A. (1965). Fuzzy sets. *Information and control*, 8(3):338–353.