RESEARCH ARTICLE

Overload Control in a Token Player for a Fuzzy Workflow Management System

Controle de sobrecarga em *Token Player* para um Sistema de Gerenciamento de Fluxo de Trabalho *Fuzzy*

Felipe Nedopetalski[1]*, Joslaine Cristina Jeske de Freitas[1]**

Abstract: The underlying proposal of this work is to control overload of a Token Player in a Fuzzy Workflow Management System. In order to accomplish this, possibility theory is used to measure how much the Token Player can be overloaded. The model used in this work is built using Colored Petri net and the simulation is made using CPN Tools. Finally, is possible to control overload of the Token Player in a Fuzzy Workflow Management System, nevertheless more time is spent to achieve the end of activities.
Keywords: CPN Tools — Workflow Management Systems — Fuzzy — Overload Control — Petri Net

Resumo: A principal proposta deste trabalho é controlar a sobrecarga de um *Token Player* em um sistema de gerenciamento de fluxo de trabalho *fuzzy*. Para atingir este objetivo, a teoria das possibilidades foi utilizada para calcular quando o *Token Player* pode ser sobrecarregado. O modelo utilizado neste trabalho foi construido utilizando redes de Petri colorida e a simulação foi feita utilizando *CPN Tools*. Como resultado, é possível controlar a sobrecarga do *Token Player* em um sistema de gerenciamento de fluxo de trabalho *fuzzy*, no entanto, mais tempo é gasto para atingir o fim das atividades.
Palavras-Chave: CPN Tools — Sistema de Gerenciamento de Fluxo de Trabalho — *Fuzzy* — Controle de Sobrecarga — Redes de Petri

[1] *Department of Computer Science, Universidade Federal de Jataí (UFJ), Brazil*
*Corresponding author: felipenedopetalski@discente.ufg.br
**Corresponding author: joslaine@ufg.br

1. Introduction

Workflow Management Systems is certainly not a new topic. It can be considered as generic packages for Business Process Management (BPM) [1, 2, 3, 4]. These systems can provide a better management and efficiency in organizational processes, once resources are allocated automatically and business processes are managed according to miscellaneous activities.

Workflow is generally assigned with business processes. For Eshuis [3] workflow is an operational business process. For Workflow Management Coalition [5] workflow is the automation of business processes in which documents, informations or activities are past from one participant to another according to a set of rules. Although workflows are strongly related with business process, tools and principles of workflows can be applied in many activities where the coordination of activities are necessary [6]. Systems that can manipulate workflows can be used since for DNA sequences [7] until scientific search for proof of relative theory [8].

Normally workflows are found in big factories for resource control of automatic processes where machines are used almost everywhere. Calculate the resource spent by a machine is a pretty simple task. However when tasks are made by humans, it is not that simple to measure the resource spent in that activity, but it is not impossible also, specially if the person is overloaded due short deadlines or low productivity [9].

Working with workflows for process automation remains to be one of the best options to improve service quality saving money and time. Ougaabal in [10] proposes a finer granularity simulation considering the definitions and particularities of each type of resource used. In [11], Wang and Tian show that efficient workflows for emergency department in hospitals helps improve the timeliness of emergency care services. Is presented a stochastic timed Petri net based on healthcare workflow and a technique for resource modeling looking for ensure timely care service and incur no excessive cost from unnecessary resource acquisition.

Cabanillas, Resinas and del-Río-Ortega in [12] identified and defined seven design-time analysis operations on how

resources are involved in process activities. As a result, they implemented a system called CRISTAL that give automatic answer to different questions related to the management of resources in business process. In [13], Cabanillas et al. introduced a graphical notation for the assignment of human resources to business process activities.

Olivieri in [14] improve workflow and resource usage in construction schedules using critical path method through location-based management systems. Mazandarani in [15] used fractional membership functions to represent the time uncertainty and the resource availability. These fractional membership functions can not only model uncertainty, but also control the amount of data included in the access functions.

Havur et al. in [16] developed a formal technique to derive an optimal schedule for work items that have dependencies and resource conflicts. They applied the solution presented in the paper in an industry scenario to evaluate its effectiveness. Ouyang et al. in [17] developed with her partners a conceptual data model of resources that takes into account the various resource classes and their interactions. They used a real-life healthcare scenario to validate the conceptual resource model.

Although Workflow Management systems still a relevant topic nowadays, resource control is not explored as it should due the simplicity treating about machines or the difficulty treating about human resource, specially when the resource is overloaded. This work propose a way to control overload resource based in a Token Player for a Fuzzy Workflow Management System using the possibility measure to calculate how much the resource can be overloaded. The model used is based in the model proposed by Aalst and Hee [1] and modified by Freitas, Julia and Vallete [18] in which the problem was found.

2. Theoretical framework

2.1 Workflow net
A Petri net that models a workflow process is called a Workflow net [1]. A Workflow net satisfies the following properties [19]:

- 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*.

The formal definition of a Workflow net is as follows [1]:

Definition 1 *(Workflow net) A Petri net PN = (P,T,F) is a Workflow net if and only if:*

- *There is one source place $i \in P$ such that $\bullet i = \phi$;*

- *There is one sink place $o \in P$ such that $o\bullet = \phi$;*

- *Every node $x \in P \cup T$ is on a path from i to o.*

According to Aalst and Hee [1], tasks are modeled by transitions. However, it can also be represented by a specific place of an ordinary Petri net [20]. It needs to have an input transition which shows beginning of the activity and an output transition which shows the end of the activity.

2.2 Fuzzy Resource Allocation Mechanism
The concept of fuzzy set has been introduced by Zadeh [21] to represent the gradual nature of human knowledge. Regularly a fuzzy set is represented by a trapezoid $A = [a1, a2, a3, a4]$. There are some particular cases to represent a fuzzy set. The triangular form, where $a2 = a3$. The imprecise form, where $a1 = a2$ and $a3 = a4$. And the precise form, where $a1 = a2 = a3 = a4$. A fuzzy continuous allocation mechanism can be defined by the marked fuzzy hybrid Petri net model [18] $C_{FCR} = \langle A_{FCR}, T_{FCR}, Pre_{FCR}, Pos_{FCR}, M_{FCR} \rangle$ with:

- $A_{FCR} = \bigcup_{\alpha=1}^{N_{FCR}} A_\alpha \cup \{R_{FC}\}$ where R_{FC} represents the fuzzy continuous resource place, A_α an activity place and N_{FCR} the number of activities which are connected to the fuzzy continuous resource place R_{FC}. A_{FCR} is the interval of availability of the fuzzy continuous resource.

- $T_{FCR} = \bigcup_{\alpha=1}^{N_{FCR}} T_{in_\alpha} \cup \bigcup_{\alpha=1}^{N_{FCR}} T_{out_\alpha}$ where T_{in_α} represents the discrete input transition of the activity A_α and T_{out_α} represents the discrete output transition of the activity A_α. T_{FCR} is the transition function for a Petri net with a fuzzy continuous resource.

- $Pre_{FCR} : A_{FCR} \times T_{FCR} \to F$ the input incidence application such as:

$$Pre_{FCR}(R_{FC}, T_{in_\alpha}) = [w1, w2, w3, w4]$$
$$\text{with } w2 = w3 \text{ and } Pre_{FCR}(A_\alpha, T_{out_\alpha}) = [1,1,1,1]$$
$$\text{where } (w1, w2, w3) \in \Re \text{ and } w4 \in \Re^+$$

(other combinations of place/transition are equal to zero) with F the set of fuzzy numbers of the triangular form. Pre_{FCR} is the Pre function of a Petri net with a fuzzy continuous resource.

- $Pos_{FCR} : A_{FCR} \times T_{FCR} \to F$ the output incidence application such as:

$$Pos_{FCR}(R_{FC}, T_{out_\alpha}) = [w1, w2, w3, w4]$$
$$\text{with } w2 = w3 \text{ and } Pos_{FCR}(A_\alpha, T_{in_\alpha}) = [1,1,1,1]$$
$$\text{where } (w1, w2, w3) \in \Re \text{ and } w4 \in \Re^+$$

(other combinations of place/transition are equal to zero). Pos_{FCR} is the Pos function of a Petri net with a fuzzy continuous resource.

- $M_{FCR} : R_{FC} \to F$ the initial marking application such as:

$$M_{FCR}(R_{FC}) = [m1, m2, m3, m4] \text{ with}$$
$$(m1, m2, m3) \in \Re \text{ and } m4 \in \Re^+$$
$$\text{and}$$
$$M_{FCR}(A_{(FCR)_\alpha}) = 0$$

the fuzzy availability (in percentage) of the fuzzy continuous resource. M_{FCR} is the initial marking function of a Petri net with a fuzzy continuous resource.

2.3 Possibility Theory

Considering one set of reference X, is attributed for each value defined in X a value that belongs to the set $[0, 1]$ and is evaluated which is the possibility of the point occur. In particular, is possible define one possibility distribution associated with one side f from a fuzzy set F [22], [23] such that:

$$\forall x \in X, \quad \Pi_f(x) = \mu_F(x)$$

The possibility measure and occurrence necessity of an event over a set of reference X is defined as:

Definition 2 *(Possibility and necessity measure) The possibility* $\Pi(S)$ *and the necessity* $N(S)$ *measures of occurrence in an event S over a set of reference X, when is considered the possibility distribution defined as* $\Pi_f : X \to [0, 1]$*, are defined as:*

$$\Pi(S) = sup_{x \in S} \Pi_f(x)$$

and

$$N(S) = inf_{x \notin S}(1 - \Pi_f(x)) = 1 - \Pi(\bar{S})$$

If $\Pi(S) = 0$, is impossible for S to happen; if $\Pi(S) = 1$, is possible for S to happen, however depends the value of $N(S)$. If $N(S) = 1$, is certain that S will occur. In particular, exists the duality relationship between the two possibility and necessities modalities, in which postulates that one event is necessary when the opposite is impossible.

In the case which two data a and b feature by two fuzzy sets A and B, the possibility measure for $a \leq b$ is defined as:

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

The necessity measure is defined as:

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

2.4 Token Player

According to Cardoso and Vallete in [24], Petri nets can be considered as a system of production rules. Each transition is considered as a rule of the transformation in the space. From a given space (mark), an applicable rule must be found to pass to next state. One way to implement the dynamic behavior in Petri nets in a grand domain variety is the concept of Token Player [25].

For Cardoso and Vallete [19], an inference engine specialized, frequently called Token Player, is able to play Petri nets, this means be able to shift tokens in a way to respect the firing transitions rules. The firing of external transitions must be synchronized with the event associated. This corresponds to the reception of update messages of knowledge, or the requisitions of the system.

On the other hand, from a given mark, all the internal transitions enabled must be fired before considering the next external event. It correspond to internal decisions and can be translated eventually through commands and answers to external decisions. The stable state of a marking net for which only transitions associated with external events are enabled. The Token Player must be in a waiting state for the external environment.

When a message is received, is necessary to find the transition that is associated with the message and fire it. From the new mark is necessary to fire all the internal transitions enabled to reach the new stable state. Firing all the internal transitions, the actions associated are executed, which provokes operations through the data and message submissions to the system external environment.

One of the particularities of the Token Player presented in [18] is the fact that in the cases where the resources are represented using fuzzy intervals, exists the possibility of overloading one resource if deadline violation happens. The measurement of the sensitization possibility of the transition $t(\Pi_t)$ in normal work situations is 1. When this value is below 1, can be noticeable that the resource can work above normal capacity. When the Player is initiated, is chosen which is the possibility value (Π_t) that is accepted to fire a transition. However, when the maximum board of a visibility interval is achieved, exists the choice of extend the visibility interval to avoid the death of the token, or overload the corresponding resource, decreasing the value of (Π_t) specified to authorise the fire of the transition.

2.5 CPN Tools

CPN Tools [26] is an application to edit, simulate and analyse state space properties of colored Petri nets models [27]. It contains an environment of simulation that can follow step-by-step a firing sequence and is possible to use simulation of Monte Carlo as well as replications. CPN Tools offers the possibility to integrate processes with complex data in only one model due the fact that the integration of processes with complex data is essential to modeling complex processes.
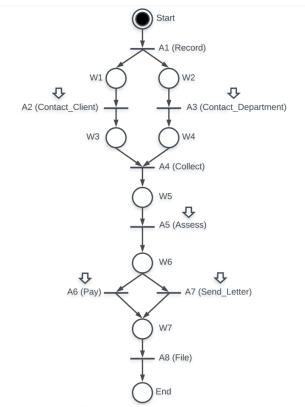
Several works has being developed using CPN Tools for modeling and model validation: [28, 29, 30, 31, 32, 33]. Due this, CPN Tools was chosen to be used in this work for modeling and simulate what is proposed.
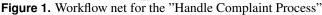
3. Model

The Workflow net model that this work is based on, is represented in Figure 1. According to Aalst and Hee [1], the

following basic constructions for task routing must be considered:

- sequential: the simplest form of task execution, where one task is executed after another, having dependency between each other;

- parallel: more than one task can be executed simultaneously, or in any order;

- conditional: when there is a choice between two tasks;

- iterative: when it is necessary to execute the same task, or a set of tasks, multiple times.

Figure 1. Workflow net for the "Handle Complaint Process"

Considering the handle complaint process represented in Figure 1, the tasks *Contact_Client* and *Contact_Department* are an example of parallel routing. Tasks *Collect* and *Assess* are an example of sequential routing. The tasks *Pay* and *Send_Letter* are an example of conditional routing.

Aalst and Hee [1] defined the notion of actuation as well, being an external condition that guide the execution of an enabled task. According to them [1], there is four distinct types of actuation: user, message, time and automatic. However, only in the user actuation, identified by the symbol ⇓, where one task is added by one resource, there is the necessity to allocate resource for task treatment. In the other types of actuations is not obliged the allocation of associated resource. In Figure 1, tasks *Record*, *Contact_Client*,

Contact_Department, *Assess*, *Send_Letter* and *Pay* are user tasks. The other three, *Record*, *Collect* and *File* are treated automatically.

The implementation of this model with fuzzy resource mechanism allocation made by Freitas, Julia and Rezende [34] in CPN Tools is represented by Figure 2. In this model, the resource spent in each activity by the Token Player is measured by a fuzzy set simulating the human behavior and it is represented by Figures 3 and 4.

3.1 Problem

In Freitas, Julia and Vallete [18] work, the Workflow net used could simulate human behavior while overloaded. However there is no control on how much the resource could be overloaded. This cause a super overload in the resource every time is needed.

If the effort of an activity is between 0 and 1, being 0 no activity and 1 maximum effort, and if it is overloaded it becomes between 0 and 2, being 0 no activity and 2 maximum overload, the system behavior always tries to use between 0 and 2 due better performance results.

As shown by Wickens [35] and verified by Nakatumba and Aalst in [36], the quantity of stress applied in employees during working time is key to work efficiency. Having control of how much resources can be overloaded is essential to not apply an exaggerated stress in employees, loosing performance after some time.

To solve this problem and control the resource overload with humans, a Token Player with restriction was added to permit manage the maximum overload allowed for human resources.

3.2 Implementation of resource control

Table 1. Table of functions used to control overload resource ⋆ This value can be between 0 and 1.

fun mR (rr:RESOURCEF) = (intToReal(1)/(intToReal(#4(rr))- intToReal(#3(rr))));
fun mC (RBA1 : RESOURCEF) = (intToReal(1)/intToReal(#2(RBA1))- intToReal(#1(RBA1)));
fun x (rr : RESOURCEF, RBA1 : RESOURCEF) = (((mC(RBA1)*intToReal(#1(RBA1)))-(mR(rr)* intToReal(#3(rr))) +intToReal(1))/ (mC(RBA1)-mR(RBA1)));
fun y (rr: RESOURCEF, RBA1 : RESOURCEF) = if (#3(rr) = #4(rr)) then true else if ((mR(rr)*x(rr,RBA1))-(mR(rr)* intToReal(#3(rr)))+intToReal(1)) $>= 1.0^\star$ then true else false;

The implementation of resource control in the "Handle Complaint Process" [1] was made in CPN Tools. Figures 2, 3 and 4 represent the implementation of the process in CPN
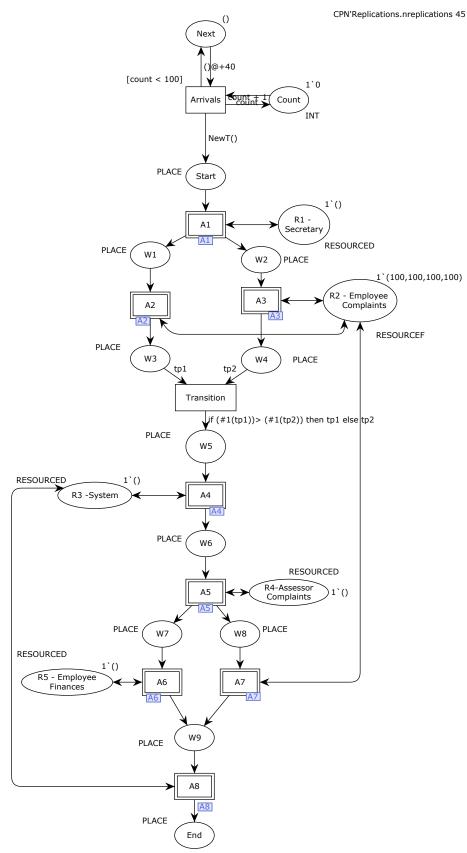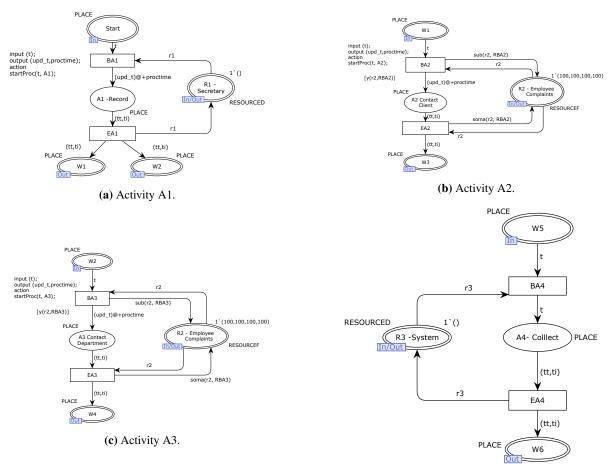
CPN'Replications.nreplications 45

Figure 2. Workflow net model made in CPN Tools with fuzzy resource mechanism allocation.

(a) Activity A1.

(b) Activity A2.

(c) Activity A3.

(d) Activity A4.

Figure 3. Activities A1, A2, A3 and A4 from Treatment Process of Complaints - Discrete and *Fuzzy* Resource.

Tools. Table 1 represent the functions created for the resource control.

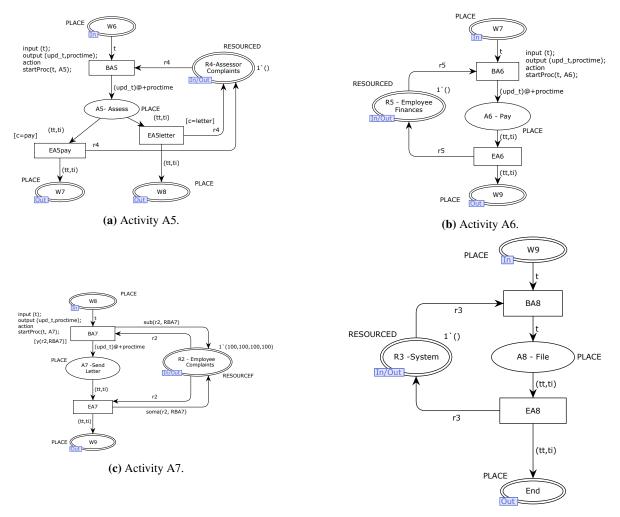Here is the meaning of some words used in Table 1:

- mR = Measure Resource;

- rr and RBA1 = names of variables;

- RESOURCEF = type of variable;

- mC = Measure Cost;

Function *mR* is responsible for measure the correspondent value of the resource. This function calculates the difference between the fourth and the third values of the fuzzy set. If the corresponding values of the resource are $[10, 20, 20, 30]$, the final result would be $\frac{1}{10}$.

Function *mC* is responsible for measure the correspondent value of the activity cost. This function does the same calculation as the mR function, however now it calculates the difference between the second and the first values of the fuzzy set of the activity cost. If the cost values of the fuzzy set to do an activity are $[30, 45, 45, 60]$, the final result would be $\frac{1}{15}$.

Function *x* is responsible for verify if the resource available is enough to complete the activity. This function calculate the difference between the cost and the resource available based on the activity that the resource is doing. If the set of possible costs of an activity are $[30, 45, 45, 60]$ and the resource available are $[10, 20, 20, 30]$, the activity would be finished but the resource would be overloaded due the final set would be $[-50, -25, -25, 0]$.

Functions *mR, mC* and *x* are responsible for giving values for function *y*. Function *y* is responsible for set the condition of how much the resource can be overloaded. The first line of function *y*, represented in Table 1, verifies if the third and the fourth values of the resource fuzzy set are equal due first iteration of simulation, the resource set is $[100, 100, 100, 100]$. This way, the activity can be finished. After the first iteration of the simulation, the resource available may be lower due activity costs, so the function verifies if the difference between the cost and the resource available is greater or equal to some value between 0 and 1. If the value of comparison is 0, the resource can be overloaded at maximum. If the value of comparison is 1, the resource can not be overloaded. Any value between 0 and 1, represent the inverted percentage of

(a) Activity A5.

(b) Activity A6.

(c) Activity A7.

(d) Activity A8.

Figure 4. Activities A5, A6, A7 and A8 from Treatment Process of Complaints - Discrete and *Fuzzy* Resource.

overload. For example, if the value is 0.8, the resource can be 20% overloaded. If the value is 0.3, the resource can be 70% overloaded.

As can be seen in Figures 3 and 4, the function *y* is called as a *guard* ($y(r2, RBA2)$) in activities A2, A3 and A7, represented by sub-figures 3b, 3c and 4c, meaning that it is a condition. The first parameter (*r2*) is the human resource responsible for doing the activity and the second parameter (*RBA2*) is the cost related of the activity. If the function returns *true*, the activity continues, otherwise not. Only activities A2, A3 and A7 received the guard function due these activities are made by the same human resource. This way, the overload of the resource can be measured more precisely.

More information about other functions from the model can be found in [18].

4. Simulation

The simulation was done in CPN Tools as well as replications. In total, 45 replications was done for each scenario (0%, 50% and 100% of overload control). CPN Tools uses Monte Carlo method of simulation by default. The confidence interval for statistical values for all simulations and replications was 95%.

After all modifications in the Workflow net, the Simulation Tool from CPN Tools was utilized to simulate the net. The tool can simulate each step of the Petri net, however is possible to set a fix number of steps to take. In this case, 2100 steps was enough to simulate 100 cases.

The function CPN'Replications.45replications (auxiliary text represented by the text in the top-right corner of Figure 2) was used to automatically run 45 simulations. The 0% of overload control scenario correspond to the case in which human resources can be fully overloaded due no control is made. The 50% of overload control scenario correspond to the case in which human resources can be overloaded at the maximum of 50%. The 100% overload control scenario correspond to the case where human resources can not be overloaded.

Tables 2, 3 and 4 presents the statistical values for the simulations and replications. The maximum duration for this

case, by default, is 105 units of time. Due this, all values presented in Tables 2, 3 and 4 are units of time. The simulations with no overload control is represented by Table 2, 50% of overload control is represented by Table 3 and with total overload control is represented by Table 4. The complete log values can be found in Annexes A1, A2 and A3.

Table 2. Table of statistical values for no overload control
*Confidence Interval

Name	Average	CI*	Min	Max
Min_id	57.42	+-0.59	52	61
Max_id	92.93	+-0.58	89	97
Average_id	75.21	+-0.32	72.71	77.61

Table 2 presents the minimum values between all scenarios due no control of resource overload is done. This means that resource can be used until it is over. The highest maximum in this scenario was 97 units of time, meaning that all the tasks was made under the limit time in this scenario.

Table 3. Table of statistical values for 50% of overload control
*Confidence Interval

Name	Average	CI*	Min	Max
Min_id	80.67	+-3.96	59	110
Max_id	2464.13	+-123.73	1626	3362
Average_id	706.31	+-27.97	559.27	945.89

Table 3 presents the statistical values for the 50% of overload control scenario, meaning that the resource can be overloaded for 50% above of capacity. The average of the minimum values is 80.67 meaning that in most cases the minimum values respected the limit time. However, is not possible to see the same pattern in maximum values, in which the average time spent was 2464.13 units. The lowest value of the maximum time spent did not achieved the time limit of 105 units of time neither the average values for this scenario.

Table 4. Table of statistical values for 100% of overload control
*Confidence Interval

Name	Average	CI*	Min	Max
Min_id	78.58	+-4.26	59	115
Max_id	2606.47	+-156.16	1421	3868
Average_id	707.33	+-30.47	521.43	1009.65

Table 4 presents the 100% of overload control scenario, meaning the resource can not be overloaded any time. Due this, in this scenario we had the worse performance overall. The average between minimum values (78.58) is lower than the previous scenario (50%), however the highest minimum, and the highest maximum are greater than other scenarios. The maximum average of 1009.65 units of time is much higher compared with the previous scenario (945.89) as well.

5. Conclusion

As showed in this paper, the control of overload in a Token Player during an activity can be done using the possibility theory and fuzzy logic. The results obtained in simulations showed that in the scenario with no overload control (Table 2), all the time limits were respected. The same is not applied for the other two simulations made (Table 3 and Table 4). Futhermore, the results showed that has an extremely difference between the simulation with no overload control (Table 2) and with total overload control (Table 4), meaning that control the resource overload delays the activities.

This can be seen in the comparison between simulations in Table 3 and Table 4, where the maximum values were higher in Table 4 (115 and 3868 lowest maximum and highest maximum respectively) than in Table 3 (110 and 3362 lowest maximum and highest maximum respectively), meaning that higher the overload control, higher is the time spent doing the activity.

As a future work, the use of possibility theory with time restriction can be utilized to equilibrate time and resource spent using the Token Player for simulations for better control of Workflow in Workflow Management Systems. Another possible future work is to use Workflow nets from industry to verify if results are similar to what we propose here.

Author contributions

Felipe Nedopetalski contributed researching and writing the paper. Joslaine Cristina Jeske de Freitas contributed guiding Author 1 on what and how he could prepare the paper.

References

[1]   AALST, W. van der; HEE, K. van. **Workflow Management: Models, Methods, and Systems**. 1. ed. Cambridge, MA, USA: MIT Press, 2004.

[2]   ALONSO, G. et al. Functionality and Limitations of Current Workflow Management Systems. **IEEE Expert**, Washington, v. 12, n. 5, p. 105–111, set-out. 1997.

[3]   ESHUIS, H. **Semantics and Verification of UML Activity Diagrams for Workflow Modelling**. Tese (Doutorado em Ciência da Computação) — University of Twente, Enschede, 2002.

[4]   HOLLINGSWORTH, D.; HAMPSHIRE, U. Workflow management coalition: The workflow reference model. **Document Number TC00-1003**, Citeseer, Hampshire, v. 19, n. 16, p. 224, 1995.

[5]   WFMC. Glossary. **Workflow Management Coalition**: Terminology and glosssary. 3. ed. Hampshire: Workflow Management Coalition, 1999. 65 p. Document Number WFMC-TC-1011. Disponível em: ⟨http://www.wfmc.org/standards/docs/TC-1011_term_glossary_v3.pdf⟩.

[6] PLESMUMS, C. Introduction to Workflow. In: FIS-CHER, LAYNA. **Workflow Handbook**. Lighthouse Point: Future Strategies, 2002. p. 19–38. Published in associaton with the Workflow Management Coalition.

[7] MEIDANIS, J.; VOSSEN, G.; WESKE, M. Using workflow management in dna sequencing. In: IFCIS - International Conference on Cooperative Information Systems, 1., 1996, Brussels. **Proceedings of the [...]**. Brussels: IEEE, 1996. p. 114–123.

[8] GIL, Y. et al. Artificial intelligence and grids: Workflow planning and beyond. **IEEE Intelligent Systems**, IEEE, Washington, v. 19, n. 1, p. 26–33, 2004.

[9] PIKA, A. et al. Mining Resource Profiles from Event Logs. **ACM Trans. Manage. Inf. Syst.**, Association for Computing Machinery, New York, v. 8, n. 1, p. 1–31, março de 2017. Disponível em: ⟨https://doi.org/10.1145/3041218⟩.

[10] OUGAABAL, K. et al. Distinguishing resource type in BPMN workflows at simulation phase. In: SPRING SIMULATION CONFERENCE (SPRINGSIM), 2019, Tucson. **Proceedings of the [...]**. Tucson: IEEE, 2019. p. 1–12. Disponível em: ⟨https://ieeexplore.ieee.org/document/8732881/⟩.

[11] WANG, J.; TIAN, J.; SUN, R. Emergency healthcare resource requirement analysis: A stochastic timed Petri net approach. In: INTERNATIONAL CONFERENCE ON NETWORKING, SENSING AND CONTROL (ICNSC), 15., 2018, Zhuhai. **Proceedings of the [...]**. Zhuhai: IEEE, 2018. p. 1–6. Disponível em: ⟨https://doi.org/10.1109/ICNSC.2018.8361301⟩.

[12] CABANILLAS, C. et al. Specification and automated design-time analysis of the business process human resource perspective. **Information Systems**, Elsevier, Hoboken, v. 52, p. 55–82, abril de 2015.

[13] CABANILLAS, C. et al. Ralph: A graphical notation for resource assignments in business processes. In: ZDRAVKOVIC, J.; KIRIKOVA, M.; JOHANNESSON, P. (Ed.). **Advanced Information Systems Engineering**. Cham: Springer International Publishing, 2015. p. 53–68.

[14] OLIVIERI, H.; SEPPÄNEN, O.; GRANJA, A. D. Improving workflow and resource usage in construction schedules through location-based management system (LBMS). **Construction Management and Economics**, Routledge, Abingdon, v. 36, n. 2, p. 109–124, 2018. Disponível em: ⟨https://doi.org/10.1080/01446193.2017.1410561⟩.

[15] MAZANDARANI, M.; LI, X. Fractional Fuzzy Inference System: The New Generation of Fuzzy Inference Systems. **IEEE Access**, IEEE, New York, v. 8, p. 126066–126082, julho de 2020. Disponível em: ⟨https://doi.org/10.1109/ACCESS.2020.3008064⟩.

[16] HAVUR, G. et al. Resource allocation with dependencies in business process management systems. In: BUSINESS PROCESS MANAGEMENT FORUM, 2016, Rio de Janeiro. **Proceedings of the [...]**. Cham: Springer, 2016. p. 3–19.

[17] OUYANG, C. et al. Modelling complex resource requirements in business process management systems. In: AUSTRALASIAN CONFERENCES ON INFORMATION SYSTEMS (ACIS), 21., 2010, Brisbane. **Proceedings of the [...]**. Auckland: AIS Library, 2010. p. 1–11.

[18] JESKE, J. C.; JULIA, S.; VALETTE, R. Fuzzy continuous resource allocation mechanisms in workflow management systems. In: BRAZILIAN SYMPOSIUM ON SOFTWARE ENGINEERING, XXIII., 2009, Fortaleza. **Proceedings of the [...]**. New York: IEEE, 2009. p. 236–251.

[19] AALST, W. van der. The Application of Petri Nets to Workflow Management. **Journal of Circuits, Systems, and Computers**, Singapore, v. 8, n. 1, p. 21–66, 1998.

[20] DAVID, R.; ALLA, H. **Discrete, Continuous, and Hybrid Petri Nets**. 2. ed. Cham: Springer Publishing Company, Incorporated, 2010.

[21] ZADEH, L. A. Fuzzy Sets. **Information and Control**, Amsterdam, v. 8, n. 3, p. 338–353, junho de 1965.

[22] DUBOIS, D.; PRADE, H. **Possibility theory**. 1. ed. New York: Plenum Press, 1988.

[23] CARDOSO, J.; VALETTE, R.; DUBOIS, D. Possibilistic Petri nets. **IEEE Transactions on Systems, Man, and Cybernetics, Part B**, New York, v. 29, n. 5, p. 573–582, 1999.

[24] CARDOSO, J.; VALETTE, R. **Redes de Petri**. 1. ed. Florianópolis: Editora da UFSC, 1997. (Série Didática).

[25] VALETTE, R.; ATABAKHCHE, H. Petri Nets for Sequence Constraint Propagation in Knowledge Based Approaches. In: ROZENBERG, GRZEGORZ AND GENRICH, H. AND VOß, KLAUS. **Concurrency and Nets: Advances in Petri Nets**. Berlin: Springer-Verlag, 1987. p. 555–569.

[26] AIS group. **CPN Tools**. Eindhoven: Eindhoven University of Technology, 2020. Disponível em: ⟨http://cpntools.org/⟩. Acesso em: 11 jul. 2020.

[27] JENSEN, K.; KRISTENSEN, L. M. **Coloured Petri nets: modelling and validation of concurrent systems**. 2009. ed. Berlin: Springer Science & Business Media, 2009.

[28] AALST, W. M. van der. Business process simulation survival guide. In: BROCKE, JAN VOM AND ROSEMANN, MICHAEL. **Handbook on Business Process Management 1**. Cham: Springer, 2015. p. 337–370.

[29] LU, Y. et al. Formalizing Workflow with Location Constraints by Colored Petri Nets. In: IEEE INTERNATIONAL CONFERENCE ON E-BUSINESS ENGINEERING (ICEBE), 11., 2014, Guangzhou. **Proceedings of the [...]**. Washington: IEEE Computer Society, 2014. p. 226–231.

[30] CHAN, N. N. et al. Mining event logs to assist the development of executable process variants. In: ADVANCED INFORMATION SYSTEMS ENGINEERING (CAISE), 26., 2014. **Proceedings of the [...]**. Cham: Springer, 2014. p. 548–563.

[31] VENERO, M. L. F.; SILVA, F. S. C. da. Modeling and Simulating Interaction Protocols Using Nested Petri Nets. In: SEFM COLLOCATED WORKSHOPS ON SOFT-WARE ENGINEERING AND FORMAL METHODS, 11., 2013, Madrid. **Revised Selected Papers of the [...]**. Berlin: Springer-Verlag, 2014. v. 8368, p. 135–150. Disponível em: ⟨https://doi.org/10.1007/978-3-319-05032-4_11⟩.

[32] XIAO, Z.; MING, Z. A method of workflow scheduling based on colored Petri nets. **Data & Knowledge Engineering**, Elsevier, Amsterdam, v. 70, n. 2, p. 230–247, 2011.

[33] VANDERFEESTEN, I.; AALST, W. van der; REIJERS, H. A. Modelling a product based workflow system in cpn tools. In: WORKSHOP ON THE PRACTICAL USE OF COLOURED PETRI NETS AND CPN TOOLS, 6., 2005, Aarhus. **Proceedings of the [...]**. Aarhus: Department of Computer Science, Aarhus University, 2005. v. 576, p. 99–118.

[34] FREITAS., J. C. J. de; JULIA., S.; REZENDE., L. P. de. Modeling a Fuzzy Resource Allocation Mechanism based on Workflow Nets. In: INTERNATIONAL CONFERENCE ON ENTERPRISE INFORMATION SYSTEMS (ICEIS), 18., 2016, Roma. **Proceedings of the [...] - Volume 2**. Setubal: SciTePress, 2016. p. 559–566. Disponível em: ⟨https://doi.org/10.5220/0005833505590566⟩.

[35] WICKENS, C. **Engineering psychology and human performance**. 2. ed. New York: Harper-Collins, 1992.

[36] NAKATUMBA, J.; AALST, W. M. van der. Analyzing resource behavior using process mining. In: INTERNATIONAL CONFERENCE ON BUSINESS PROCESS MANAGEMENT, 7., 2009, Ulm. **Proceedings of the [...]**. Berlin, Heidelberg: Springer, 2009. p. 69–80.

Annexes

In this section, the complete statistical values from all simulations scenarios are presented. Figure A1 represents complete simulation results with no overload control. Figure A2 represents complete simulation results with 50% of overload control. Figure A3 represents complete simulation results with 100% of overload control.

Statistics							
Name	Avrg	90% Half Length	95% Half Length	99% Half Length	StD	Min	Max
Duracao							
count_iid	61.666667	0.119677	0.143626	0.192165	0.476731	61	62
max_iid	92.933333	0.482877	0.579510	0.775356	1.923538	89	97
min_iid	57.422222	0.494857	0.593887	0.794592	1.971258	52	61
sum_iid	4638.355556	19.600405	23.522813	31.472384	78.078095	4441	4812
avrg_iid	75.215536	0.263838	0.316637	0.423645	1.050997	72.709677	77.612903

Figure A1. Complete results of simulation with no overload control.

Statistics							
Name	Avrg	90% Half Length	95% Half Length	99% Half Length	StD	Min	Max
Duracao							
count_iid	51.844444	0.444436	0.533376	0.713631	1.770408	48	54
max_iid	2464.133333	103.101358	123.733874	165.549924	410.703642	1626	3362
min_iid	80.666667	3.297521	3.957417	5.294832	13.135656	59	110
sum_iid	36555.600000	1115.336259	1338.535973	1790.896225	4442.935319	28523	45403
avrg_iid	706.306576	23.308968	27.973530	37.427226	92.851134	559.274510	945.895833

Figure A2. Complete results of simulation with 50% of overload control.

Statistics							
Name	Avrg	90% Half Length	95% Half Length	99% Half Length	StD	Min	Max
Duracao							
count_iid	52.133333	0.462268	0.554777	0.742264	1.841442	48	55
max_iid	2606.466667	130.117795	156.156808	208.930237	518.323460	1421	3868
min_iid	78.577778	3.546748	4.256519	5.695016	14.128451	59	115
sum_iid	36803.333333	1204.727854	1445.816504	1934.432373	4799.026203	27636	49473
avrg_iid	707.326558	25.390291	30.471364	40.769208	101.142071	521.433962	1009.653061

Figure A3. Complete results of simulation with 100% of overload control.