Computer Science Journal of Moldova, vol.**29**, no.3(87), 2021

Uncertainty modelling of dynamically reconfigurable systems based on rewriting stochastic reward nets with z-fuzzy parameters

Victor Moraru, Emilian Guţuleac, Sergiu Zaporojan

Abstract

This paper presents the descriptive compositional approach for uncertainty modelling and performance evaluation of dynamic reconfigurable discrete event systems (ReDES) using rewriting stochastic reward nets (ReSRN) with Z-fuzzy parameters (FReSRN) that can modify in run-time their own structure by the rewriting of the rules. The expected Z-fuzzy values of the transition and rewriting rule firing rates are calculated based on credibility theory, the FReSRN model is degenerated to a conventional ReSRN model. A case study for performance modelling and analysis of particular ReDES is given in order to show the effectiveness of the proposed method.

Keywords: Discrete event systems, Petri Nets, dynamic reconfiguration, fuzzy parameters, model composition, performance modelling, rewriting rules, stochastic reward networks.

1 Introduction

Currently, dynamic reconfiguration of computational processes is used in a multitude of applications of dynamically reconfigurable and variably interconnected discrete event systems (ReDES) with structures that change dynamically during their operating time, adapting to the changing requirements and environment. This type of systems includes, for example: computing systems and networks; mobile robot systems; mobile dynamic Ad-hock computer networks; reconfigurable manufacturing systems and many applications with new technological solutions, based on cloud computing and systems with Internet of Things applications, etc.

As a formal tool, the use of different extensions of Petri Nets (PNs) [1], such as generalized stochastic PN (GSPN) [2] and stochastic reward nets (SRN) [3], in the study of discrete event systems (DES), attracts many researchers. SRNs are a generalization of GSPNs that often result in more compact models and specify output measures as reward-based functions for the performance evaluation of complex DES.

Although, PNs (of low or high level), underlying GSPN or SRN, are a powerful and expressive formal tool. They are unable to specify/verify, in a natural way, advanced systems having dynamic structures. To meet the new needs of designers in the analysis of behavioural qualitative properties of ReDES, some researchers have introduced reconfigurability in several classes of PNs [4]-[6]. This type of PNs focuses only on the off-line analysis of qualitative properties without taking into account the quantitative ones. To overcome this problem, researchers are enriching GSPNs with reconfigurability [7], [8].

When analysing the performance of a DES or ReDES with GSPN or SRN, the quantitative parameters and, therefore, the state probabilities of this type of model are generally considered accurate and perfectly known. However, while analysing the performance of a ReDES, the known information about values of component's failure parameters, attack rates, etc. is, in general, not perfect. The uncertainty of real values of the quantitative parameters can have two origins [9], [10]. The first source of uncertainty comes from the randomness character of the information that has a natural stochastic variability. The second source of epistemic uncertainty is related to the imprecise and incomplete character of information because there is no knowledge about real values of system quantitative parameters, which change dynamically their state. Therefore, in order to make our modelling approach describing the behaviour of a ReDES more accurate, realistic and versatile, it is necessary to take into account the probabilistic and fuzzy aspects, which are based on SRN in couple with the fuzzy set theory.

To the best of our knowledge, there is no work in the literature that treats the dynamic reconfigurability in SRNs with fuzzy Z-parameters

[10] for modelling dynamic run-time reconfiguration and performance evaluation measures of modern ReDES.

In order to describe more accurately the expected behaviour uncertainty and run-time component reconfiguration of ReDES, this paper proposes a new approach for modelling and performance measures evaluation that introduces run-time reconfigurability in SRN and Z-fuzzy parameters, which extends the work presented in [7] and [9]. Thus, combining these two paradigms, a new class of descriptive-compositional SRN with Z-fuzzy firing rates of timed transitions and rewriting rules is defined, that is called FReSRN. In this context a numerical example is examined to demonstrate the applicability and utility of the FReSRN approach proposed in this paper for performance modelling of ReDES.

Next, due to the space restrictions we will only give a brief overview of this topic and refer the reader to papers [3], [7], [9]-[12].

2 Elements of Z-fuzzy numbers and credibility theory

L. Zadeh proposed in [10] the concept of fuzzy Z-numbers, which also allows us to take into account the inaccuracy of our knowledge of the membership function using a joint approach from the standpoint of probability theory and theory of possibility. The Z-numbers have more capability to describe the uncertain information. A Z-number is an ordered pair of fuzzy numbers and is denoted as $Z = \left(\widetilde{A},\ \widetilde{R}\right)$. The first component $\widetilde{A}$ plays the role of a fuzzy restriction on the values. It is a real-valued uncertain variable X. The second one, $\widetilde{R}$, is the measure of the reliability for the first component.

Computing with Z-numbers can be realized by directly using Zadeh expansion principle, which requires very cumbersome calculations and is extremely difficult when solving complex applied problems. In [11], it was proposed a method of converting Z-numbers to generalized fuzzy numbers. Assume $Z = \left(\widetilde{A},\ \widetilde{R}\right)$ is a Z-number. The left of Z is the part of restriction, and the right of Z is the part of reliability. Let

$\widetilde{A} = \{\langle x,\ \mu_{\widetilde{A}}(x)\rangle | x \in [0,\ 1]\}$ and $\widetilde{R} = \{\langle x,\ \mu_{\widetilde{R}}(x)\rangle | x \in [0,\ 1]\}$, where $\mu_{\widetilde{A}}(x)$ is a *trapezoidal* membership function and $\mu_{\widetilde{R}}(x)$ is a *triangular* membership function.

Let $\widetilde{A} = (a_1, a_2, a_3, a_4; 1)$ is the trapezoidal fuzzy number and α is the weight of $\widetilde{R}$ in $Z = \big(\widetilde{A},\ \widetilde{R}\big)$, $\alpha{=}\big(\int x \cdot \mu_{\widetilde{R}}(x)\, dx\big) \,/\, \big(\int \mu_{\widetilde{R}}(x)\, dx\big)$. Then $\widetilde{Z}^{\alpha}{=}(a_1, a_2, a_3, a_4; \alpha)$ and $\widetilde{Z}'{=}(a_1 \cdot\sqrt{\alpha}, a_2 \cdot\sqrt{\alpha}, a_3 \cdot\sqrt{\alpha}, a_4 \cdot\sqrt{\alpha}; 1)$ is a *regular* fuzzy number.

According to [12], the average credibility value $\bar{z}'{=}E\big[z'\big]$ of trapezoidal fuzzy variable is determined by the relation $\bar{z}'{=}E\big[z'\big]{=}(\sqrt{\alpha}) \cdot (a_1 +\ a_2 +\ a_3 +\ a_4)\,/4$. This expression will further be used to determine the credible parameters of a FReSRN model.

3 Rewriting SRN with Z-fuzzy Parameters

Definition and behaviour of FReSRN. Let IN_+ and IR^+ be the sets of non-negative natural and, respectively, non-negative real numbers.

The definition of an FReSRN is derived according to [7] and inherits most of the SRN characteristics [2]. Thus, the *FReSRN*, denoted $R\Gamma$, is defined as a 12-tuple system such that $R\Gamma = \langle\ P,\ E,\ A_{rcs},\ Pri,\ G^E,\ G^R,\ K^p,\ M_0,\ \Lambda,\ \omega,\ \rho,\ Lib\ \rangle$, where: P is a finite set of places; $E{=}T \cup R$ is a finite set of events, $T \cap R{=}\varnothing$, $P \cap E{=}\varnothing$, where T is a finite set of transitions and R is a finite set of rewriting rules about the run-time structural change (reconfiguration) of $R\Gamma$. The set E is partitioned into $E{=}E_0 \cup E_\tau$, $E_0 \cap E_\tau{=}\varnothing$ so that: E_τ is a set of timed events and E_0 is a set of immediate events; $A_{rcs} = {<}Pre,\ Post,\ Inh{>}$ is a set of *forward*, *backward*, and *inhibition* functions, that describe, respectively, the arcs with marking-dependent weight cardinalities; *Pri* defines the dynamic marking-dependent priority function for the *firing* of each *enabled* $e{\in}E$. The firing of an enabled event with higher priority potentially disables each event $e{\in}E$ with the lower priority. By default, $Pri(\ E_0\ ){>}Pri(\ E_\tau\ )$; $G^E{:}E{\times}IN_+^{|P|}{\to}\{\ True,\ False\}$ is the set of *guard functions* associated with each event $e{\in}E$; and $G^R{:}R{\times}IN_+^{|P|}{\to}\{\ True,\ False\}$ is the set of *guard functions* associated with all *rewriting rules* $r{\in}R$; $K^p{:}\ P \times IN_+^{|P|}{\to}IN_+ \cup \{\infty\}$ is the capacity bound of each place

$p_i \in P$, which can contain an *integer* number of *tokens*. By default, K_i^p is equal to $+\infty$; M_0 is the initial marking; $\widetilde{\Lambda}: E_\tau \times IN_+^{|P|} \rightarrow IR^+$ is the function that determines the Z-fuzzy firing rate $0 < \widetilde{\lambda}(e, M) < +\infty$ (the parameters of exponential-negative law) of the timed event $e \in E_\tau$, that is enabled by the current marking $M; \omega: E_0 \times IN_+^{|P|} \rightarrow IR^+$ is the Z-fuzzy weight function $0 \leq \omega(e, M) < +\infty$ which determines the firing probability $q(t, M)$ of the immediate event $e \in E_0$, enabled by the current marking M and therein describes a probabilistic selector; $\widetilde{\rho}: P \cup E \rightarrow IR^+$ is the function that determines the Z-fuzzy function reward rates (real numbers) assigned to each current marking M and to each firing event $e \in E$; $Lsp \subset Lib$ is the set of $R\Gamma_\nu$, $\nu=1, 2, \cdots, n_\nu$ are *subnet* patterns and/or parameters of the class library involved in structural reconfiguration of the current $R\Gamma$ configuration by firing of an enabled rewriting rule $r \in R$.

Figure 1 summarizes the graphical representation of all $R\Gamma$ primitive elements.

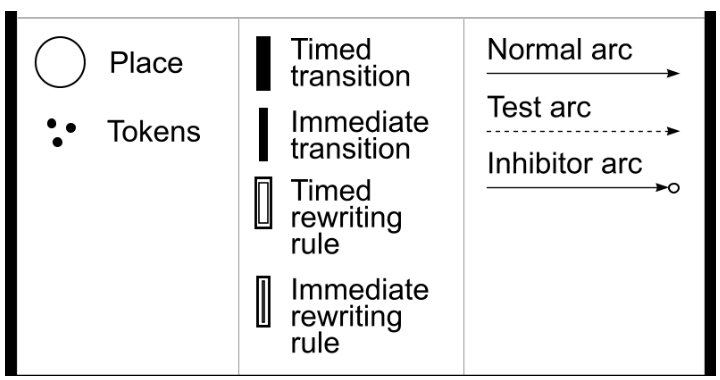

Figure 1. The structural primitives of $R\Gamma$ elements

Let ${}^AW = \{{}^\bullet W,\ W^\bullet,\ {}^\circ W\}$ be the weights of the respective types of arcs in A_{rcs}. We note by $Atr_\Gamma = \{{}^AW,\ G^E,\ G^R,\ K^p,\ \Lambda,\ \omega,\ \rho\ ,\ Pri\}$ the set of *quantitative attributes* of the currently activated (sub)nets of type $R\Gamma$. Also, let $RN = < \Gamma,\ M >$ be the current configuration of

$R\Gamma$, where $\Gamma{=}R\Gamma\backslash M$ and M is the current marking of $R\Gamma$.

A dynamic reconfiguration of RN by the firing of enabled $r{\in}R$ is a map r: $\{RN_L,\ Atr_L\}{\rhd}\{RN_W,\ Atr_W\ \}$, where $RN_L{\in}Lsp_{RN}$, $Atr_L{\in}Lsp_{Atr}$ is the left-hand side and $RN_W{\in}Lsp_{RN}$, $Atr_W{\in}Lsp_{Atr}$ is the right-hand side of the *rewriting operator* $\rhd$ assigned to rewriting rule r, respectively. The operator $\rhd$ represents a binary rewriting operation which produces a *structural change* and/or *change of attributes* in RN by replacing (rewriting) the fixed current subnet $\{RN_L,\ Atr_L\}{\subseteq}RN$ (RN_L are dissolved with $P_L{\subseteq}P$, $E_L{\subseteq}E$ and subset of arcs $A_L{\subseteq}A$ and/or Atr_L are deleted) and a new $\{RN_W,\ Atr_W\ \}{\in}Lsp$ subnet (with $P_w{\subseteq}P$, $E_w{\subseteq}E$ and set of arcs A_W and/or Atr_W are added), belonging to the new modified resulting underlying net $RN^{'}{=}(RN\backslash RN_L)\cup RN_W$, with $P^{'}{=}(P\backslash P_L)\cup P_W$ and $E^{'}{=}(E\backslash E_L)\cup E_W$, $A^{'}{=}(A-A_L)+A_W$, where the meaning of operations $\backslash$ (resp. $\cup$) is removing (resp. adding) of RN_L from RN_W (resp. to RN).

In this new $RN^{'}$ net, obtained by the execution of the enabled $r{\in}R\,(M)$, the places and the events belonging to $RN^{'}$ are *fused* [7]. So, the current state configuration of an RN net is $\gamma{=}(\Gamma, M)$, i.e. the current structure configuration $R\Gamma$ of the net together with a current marking M. Also, the pair $\gamma_0{=}(\Gamma_0, M_0)$ is the initial configuration.

Enabling rule of events $e{\in}E$ by current marking is the same as for GSPN and SRN [2], [3]. Let the $E\,(M){=}T\,(M)\cup R\,(M)$, $T\,(M)\cap R\,(M){=}\varnothing$, be the set of enabled events in a current marking M, where $T\,(M)$ and $R\,(M)$ are the sets of enabled transitions and enabled rewriting rules, respectively.

In the following, we will present only the *firing rule* of a $r{\in}R\,(M)$. The event $e_j{\in}E\,(M)$ fires if *no other event* $e_k{\in}E\,(M)$ with higher priority has been enabled.

Hence, for event e_j, ***if*** $\big((e_j{=}\ t_j)\vee(e_j{=}\ r_j)\wedge\big(g^R\,(r_j, M){=}False\big)\big)$, ***then*** the firing of $t_j{\in}T\,(M)$ or of $r_j{\in}R\,(M)$ changes only the current marking: $\big(\Gamma, M\big)\xrightarrow{\ e_j\ }\big(\Gamma, M^{'}\big)\Leftrightarrow(\Gamma\ {=}\Gamma)$ and $M\xrightarrow{\ e_j\ }M^{'}$ in Γ. Also, for every $e_j{\in}E$, *if* $\big((e_j{=}\ r_j\ )\wedge\big(g^R\,(r_j, M){=}True\big)\big)$, ***then*** the event e_j occurs at firing of the rewriting rule r_j and it changes the configuration and marking of the current net, such that: $(\Gamma, M)\xrightarrow{r_j}\big(\Gamma^{'}, M^{'}\big)$

$\Leftrightarrow \left(\Gamma = \Gamma'\right)$ and $M \xrightarrow{e_j} M'$ in Γ'.

The reachability state graph (RG) of current configuration $\gamma=<\Gamma, M>$ from initial configuration $\gamma_0=<\Gamma_0, M_0>$ is the *labelled directed graph* whose nodes are states and whose arcs, which are labelled with events or rewriting rules of RN, are of two kinds:

(i) $(R\Gamma, M) \xrightarrow{e_j} \left(R\Gamma, M'\right)$ if $((e_j = r_j) \vee (e_j = r_j)) \wedge \left(g^R\left(r_j, M\right) = False\right)$;

(ii) $(R\Gamma, M) \xrightarrow{r_j} \left(R\Gamma', M'\right)$ if $(e_j = r_j) \wedge \left(g^R\left(r_j, M\right) = True\right)$.

Based on this RG, we can build the embedded continuous time Markov chain ($ECTMC$) with fuzzy credible parameters and evaluate the specified performance measures.

4 Descriptive composition of FReSRN

Enhancing PNs with compositionality properties is an essential feature that is required today for modelling of large-sized ReDES. It allows constructing complex PN models of ReDES by combining smaller entities of sub-models based on its constitutive descriptive expressions (DE), which are a compact symbolic representation of these PN sub-models and their relationships [7].

Details concerning the using of DE approach for compositional construction of PN underlying the GSPN models can be found in [7], as they require a great deal of space. In this section, we present only the notion of *dexel* (***d****escriptive* ***ex****pression* ***el****ement*), denoted as *bDE*, and some compositional operations that allow constructing in the analytical form the DE, mapped in PN models, underlying those of FReSRN, that use the approach proposed in [7] as follows:

$$bDE = {}_{g_j}^{\Pi_j}|_{e_j}^{\alpha_j} m_{0i}^{k_i} y_i^{\beta_i} \ \left[W_i^+, W_i^-\right]_{g_k}^{\Pi_k} |_{e_k}^{\alpha_k},$$

where $y \in \{p, \bar{p}\}$ is the place-symbol that determines respectively the type of arc ({ *normal*, *inhibitor*}) with the incident weight $W_i^- \in \{\Pr e\left(e_k, \ p_i\right), \ \ Inh\left(e_k, \ p_i\right)\}$ arc, normal or inhibitor, before the event

$|_{e_k}$, and is the weight $W_i^+ \in \{Post\,(e_j,\ p_i)\}$ of the normal arc that gets out of the event $|_{e_j}$ and enters the place p_i. The translation of bPN and it derivatives for $e_j{=}t_j$ and $e_k{=}t_k$ are shown in Figure 2. The attributes of the place p_i are, respectively, as follows: m_{0i} – the initial marking; k_i – the place capacity; the place label β_i shows the type of conditions. The attributes of the events e_j and e_k are, respectively, as follows: g_j and g_k – the guard function; Π_j and Π_k – the priority function; α_j and α_k – labels showing the type of action.

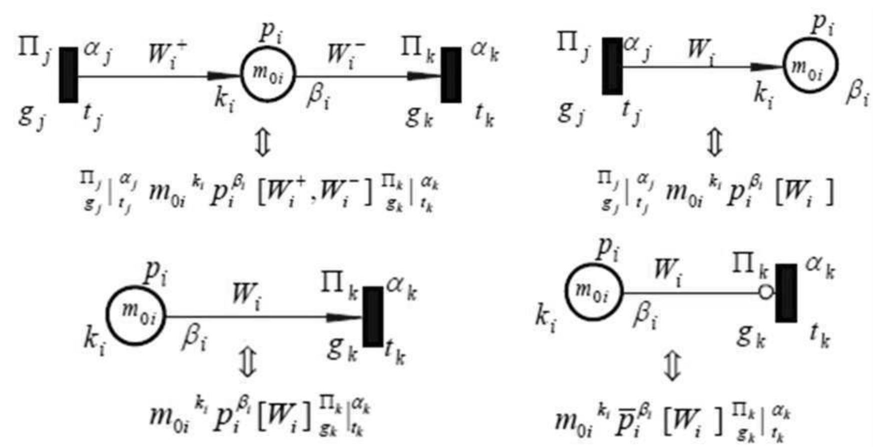

Figure 2. Translation in *bPN* of *bDE* and its derivatives

If the current marking m_i^0 of place p_i has zero tokens, we can omit $m_i^0 = 0$ in *bDE*. By default, if the label's action is not mentioned, then it matches the name of the event. From a *bDE* we can build more complex *DE* of PN components by using composition operations. By default, if $W_i^+ = W_i^- = 1$, we present *bDE* and its derivatives as follows: $|_{e_j} m_i^0 p_i\ |_{e_k}$, $|_{e_j} m_i^0 p_i$, $m_i^0 p_i |_{e_k}$ or $m_i^0 \bar{p}_i |_{e_k}$.

Any DE of PN is $DE ::= bDE/DE_j * DE_k/ \circ DE$, where $*$ represents any *binary* composition operation and $\circ$ – any *unary* operation. The composition operations are reflected at the DE components of PN models by fusion of places, fusion of transitions with the same name (label) or sharing the subnets. For example, when merging the place p of two different subnets, DE_i and DE_j, with the respective current markings, m_i^0 and m_j^0, respectively, the number of tokens in this resulting place p is added, i.e. the resulting marking is $\left(m_i^0 + m_j^0\right)p$.

Descriptive compositional operations: 1) **Inhibition** unary operation, represented by *inhibitory operator* "$^-$"(place-symbol with over-bar),describes the inhibitor arc in PN models with a weight function $W_i = Inh\,(p_i, e_j)$; 2) **And-join** binary operation, represented by the operators " $\bullet$" or " $\overleftrightarrow{\wedge}$", describes the rendezvous (synchronization by e_k) of two or more conditions represented, respectively, by symbol-places $p_i \in {}^\bullet e_k$, $i = \overline{1,\ n}$, i.e., it indicates that all preceding conditions of occurrence actions must have been completed; 3) **And-split** binary operation, represented by the operator "$\diamond$", determines the causal relations between activity e_k and its post-conditions: after completion of the preceding action of e_k, concomitantly several other post-conditions can occur in parallel ("*message sending*"); 4) **Sequential** binary operation, represented by the *operator* "|", determines the logic of an interaction between two local states p_i (pre-condition) and p_k (post-condition) by e_j action that are in precedence and succeeding (*causality-consequence*) relations relative to this action; 5) **Competing Parallelism** binary operation is represented by the "$\overleftrightarrow{\nabla}$" *operator*, and it can be applied over two PN_A with $DE_A = A$ and PN_B with $DE_B = B$ or internally into the resulting PN_R with $DE_R = A \overleftrightarrow{\nabla} B$, between the places of a single PN_R which are the symbol-places or symbol-events with the same name, respectively fused. The fused places or events will inherit the arcs in A and B.

Further details on definition and using compositional operations can be found in [7].

5 Performance modelling of a particular CS using FReSRN

We will illustrate in this section the application of the FReSRN to a case study based on a particular data center computing system (CS).

In this case study, we consider a CS of a data center including *three servers*: Sv_1, Sv_2, Sv_3, and *two buffers* in which two types of jobs are processed: in the buffer Buf_1 with the capacity L_1 of waiting places, the jobs of class C_1 arrive and are stored; in the buffer Buf_2 with the capacity L_2 of waiting places, the jobs of class C_2 arrive and are stored. In both buffers, the FIFO "*first come, first served*" discipline is implemented. The server Sv_1 processes C_1 class jobs, while the server Sv_2 is dedicated to C_2 class jobs processing. In the initial configuration γ_0, in order to save power consumption, the server Sv_3 doesn't work and stays in *standby mode*. The Sv_3 is immediately activated and joins the server Sv_1 (resp. Sv_2) when the number of jobs stored in Buf_1 (resp. Buf_2) is equal to the threshold n_1 (resp. n_2). In this case, the CS has a three-servers configuration, and the transition from γ_0 configuration to another γ_1 or γ_2 is performed depending on the current number of jobs waiting in those buffers, Buf_1 or Buf_2, respectively.

The behaviour of the CS that has the initial γ_0 configuration is described by the $R\Gamma_0$ model presented in Figure 3 in which $p_{i\_j}{:=}\,p_{i,j}$, $t_{i\_k}{:=}t_{i,k}$, and $r_{i\_l}{:=}r_{i,l}$, i=1, 2, 3 .

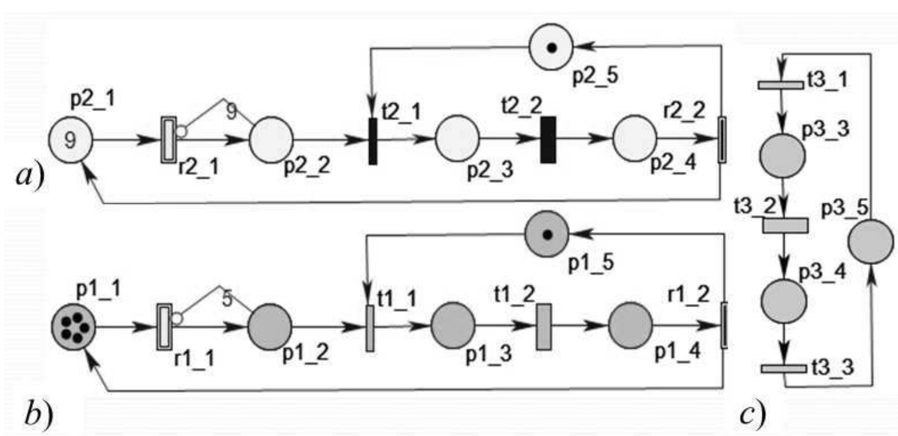

Figure 3. The model $R\Gamma_1^0$ of CS in γ_0 that describes the active behaviour *a)* of Sv_1; *b)* of Sv_2; *c)* the passive behaviour of Sv_3

In the $R\Gamma_0$, the inhibitory arc $Inh\,(p_{1,2},\ r_{1,1})$=5 represents the capacity L_1=5 of Buf_1, and respectively, $Inh\,(p_{2,2},\ r_{2,1})$=9 represents the capacity L_2=9 of Buf_2.

Meanings of places and events at configuration γ_0 :

- **Places**. $p_{1,1}$ or $p_{2,1}$ – the number of tokens at these places respectively describe the potential number of the type C_1 jobs or those of the type C_2 that should be processed by Sv_1 or Sv_2, respectively; $p_{1,2}$ and $p_{2,2}$ – the buffers Buf_1 and Buf_2. The number of tokens in these places models the number of jobs in the CS waiting in the respective buffer to be processed; a token in $p_{2,3},\ p_{3,3}$ indicates that a job of respective type is being processed by one of the servers $Sv_1,\ Sv_2$ or Sv_3; symbol-places $p_{1,4},\ p_{2,4}$, $p_{3,4}$ indicate that the respective server $Sv_1,\ Sv_2$ or Sv_3 has finished processing jobs of the respective type; symbol-places $p_{1,5}$, $p_{2,5}$ or $p_{3,5}$ indicate that the respective server is in passive state.

- ***Transitions*** $t_{2,1},\ t_{3,1}$ are the *immediate* transitions, the firing of which leads to the activation of the respective server, Sv_2 or Sv_3, to start processing a task of the respective type from its buffer. The firing of immediate transition $t_{3,3}$ models the download of the processed job by the Sv_3 server; the firing time of timed transitions $t_{1,2}, t_{2,2},\ t_{3,2}$ represents the processing time of a job of that respective type.

- ***Rewriting rules***. $r_{1,1}$ (resp. $r_{2,1}$) – *timed rewriting rule*, the firing of which describes the job arrival on the server Sv_1 (resp. Sv_2) if $g^R_{1,1}\,(M)$:=”$False$” (resp. $g^R_{2,1}\,(M)$:=”$False$”) or reconfiguration from the initial configuration γ_0 to the configuration γ_1 (resp. γ_0 to the configuration γ_2) if $g^R_{1,1}\,(M)$:=”$True$” (resp. $g^R_{1,2}\,(M)$:=”$True$”); $r_{1,2}$ (resp. $r_{2,2}$) – the immediate rewriting rule, the firing of which describes the download of the processed job by the server Sv_1 (resp. Sv_2) and its passing into the passive state, if $g^R_{1,2}\,(M)$:=”$False$” (resp. $g^R_{2,2}\,(M)$:=”$False$”) or reconfiguration from the configuration γ_1 (resp. γ_2) to the configuration γ_0, if $g^R_{1,2}(M)$:="$False$" (resp. $g^R_{2,2}(M)$:="$True$").

The descriptive expression DE^0 of the model $R\Gamma_0$ describing the behaviour of CS in γ_0 configuration showed in the Figure 3 is: $DE^0{=}DE^0_i \overset{\leftrightarrow}{\triangledown} DE^0_3$, where $DE^0_i{=}DE^0_{i,1} \overset{\leftrightarrow}{\triangledown} DE^0_{i,2} \overset{\leftrightarrow}{\triangledown} DE^0_{i,3}$, and $DE^0_{i,1}{=} \left(m^0_{i,1}p_{i,1} \bullet p_{i,2}\left[w^0_{i,1}\right]\right)|_{r_{i,1}}p_{i,2}$, $DE^0_{i,2}{=}\ (p_{i,2} \bullet 1p_{i,5})\ |_{t_{i,1}}p_{i,3}|_{t_{i,2}}p_{i,4}$, $DE^0_{i,3}{=} p_{i,4}|_{r_{1,2}}\ (p_{i,1} \diamond p_{i,5})$, $i{=}1,2$ describes the behaviour of Sv_i (see the Figure 3a and Figure 3b), respectively. $DE^0_3{=}\ p_{3,5}|_{t_{3,1}}p_{3,3}|_{t_{3,2}}p_{3,4}|_{t_{3,3}}p_{3,5}$ describes Sv_3 in the passive state (Figure 3b).

The reconfigurations $\gamma_0{\Rightarrow}\gamma_j$ are described by $r^0_{j,1}$: $\{\ g^{R_0}_{j,1},\ g^{R_0}_{j,2}\ \}\triangleright \{DE^j_3\ ,\ g^{R_j}_{j,1},\ g^{R_j}_{j,2}\ \}$, $j{=}1,\ 2$; $DE^j_3{=}DE^0_3 \overset{\leftrightarrow}{\triangledown} 1p_{3,5} \overset{\leftrightarrow}{\triangledown} Post\,(t_{3,3},\ p_{j,1}) \overset{\leftrightarrow}{\triangledown} Pre\ (t_{3,3},\ p_{j,1})$, $g^{R_0}_{j,1}{:=}((M\ (p_{j,2}){=}n_j)\ \&\ (M\ (p_{j,3}){=}1)\ \&\ (M\ (p_{3,3}){=}0))$, $g^{R_0}_{j,2}{:=}False$, $j{=}1,\ 2$.

Also, $r^1_{j,2}$: $\{DE^j_3,\ g^{R_j}_{j,1},\ g^{R_j}_{j,2}\ \}\triangleright\{DE^0_3,\ g^{R_0}_{j,1},\ g^{R_0}_{j,2}\ \}$, describes the reconfigurations $\gamma_j{\Rightarrow}\gamma_0$, $j{=}1,\ 2$, where: $g^{R_j}_{j,1}{:=}False, j{=}1,\ 2$.

In Figure 4, there is presented the model $R\Gamma^1_1{\subset}R\Gamma_1$ given by DE^1_1:

$$DE^1_1{=}(2p_{1,1} \bullet 1p_{1,2}\,[5])\,|_{r_{1,1}}p_{1,2} \overset{\leftrightarrow}{\triangledown}$$

$$(p_{1,2} \bullet p_{1,5})\,|_{t_{1,1}}1p_{1,3}|_{t_{1,2}}p_{1,4}|_{r_{1,1}}\,(p_{1,1} \diamond p_{1,5})\,,$$

which describes the active behaviour of Sv_1 with Sv_3 in γ_1 for processing only jobs of the type C_1. Due to the space constraint, the model $R\Gamma^2_2$ that describes the active behaviour of Sv_2 with Sv_3 in γ_2, being similar to $R\Gamma^1_2{\subset}R\Gamma_2$, is not presented.

The respective models $R\Gamma_0, R\Gamma_1$ and $R\Gamma_2$ are *bounded*, *live* and *reversible* according to [1], [2] and [3].

Next, we will present the evaluation of some quantitative measures [2], [3], [7] only for the case when the jobs of the class C_1 are processed in reconfigurations $\gamma_0{\Rightarrow}\gamma_1{\Rightarrow}\gamma_0$, because it is similar to the one for processing the jobs of the class C_2 in reconfigurations $\gamma_0{\Rightarrow}\gamma_2{\Rightarrow}\gamma_0$. We also mention that in the configurations γ_0 and γ_1 (resp. γ_0 and γ_2) the server Sv_2 (resp. Sv_1) works independently of the other servers.

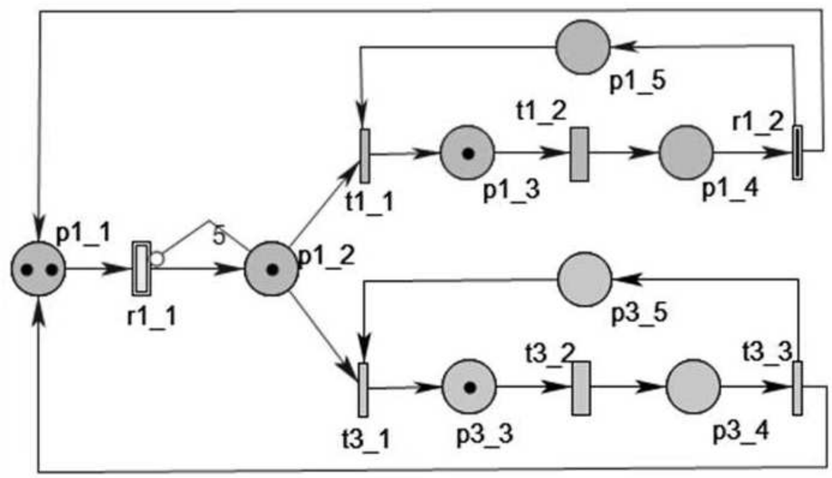

Figure 4. The model $R\Gamma_1^1$ of CS in γ_1 that describes the active behaviour of Sv_1 with Sv_3

The reachability graph RG1 of the model $R\Gamma_1^1$, which is presented in Figure 3a, shows the state space in which the server Sv_1 can be when processing jobs of the class C_1 in the configuration γ_0 and reconfiguration $\gamma_0 \Rightarrow \gamma_1$ for the capacity L_1=5 and the threshold n_1=2 at the firing of the rewriting rule $r_{1,1}^0$. The reachability graph RG1, in the form of a list with tangible markings M_i^0 of the model $R\Gamma_1^0$ shown in Figure 3a, is as follows:

$$M_0^0=(5p_{1,1}p_{1,5})\xrightarrow{r_{1,1}^0 t_{1,1}} M_1^0;$$

$$M_1^0=(4p_{1,1}p_{1,3})\xrightarrow{r_{1,1}^0}>M_2^0, \quad \xrightarrow{t_{1,2}r_{1,2}^0}>M_0^0;$$

$$M_2^0=(3p_{1,1}p_{1,2}p_{1,3})\xrightarrow{r_{1,1}^0}>M_3^0, \quad \xrightarrow{t_{1,2}r_{1,2}^0 t_{1,1}}>M_1^0;$$

$$M_3^0=(2p_{1,1}2p_{1,2}p_{1,3})\xrightarrow{t_{1,2}r_{1,2}^0 t_{1,1}}>M_2^1,$$

$$\xrightarrow{r^0_{1,1}} M^1_0 \Rightarrow to\ reconfiguration\ \gamma_1.$$

Thus, as soon as the number of waiting jobs of the class C_1 is equal to the threshold n_1=2, the server Sv_3 is immediately activated. After that, the server Sv_3 joins the server Sv_1 to process the jobs from the buffer Buf_1. As a result, the model $R\Gamma^1_1$ is obtained, described by DE^1_1, in the configuration γ_1 (see Figure 4).

The reachability graph RG2 of the model $R\Gamma^1_1$, (Figure 4) shows the state space in which the servers Sv_1 and Sv_3 can be in the γ_1 configuration and reconfiguration $\gamma_1 \Rightarrow \gamma_0$ at the firing of the $r^1_{1,2}$. The RG2, in the form of a list with tangible markings M^1_i of $R\Gamma^1_1$, is as follows:

$$M^1_0 = (2p_{1,1}p_{1,2}p_{1,3}p_{3,3}) \xrightarrow{r^1_{1,1}} M^1_1, \quad \xrightarrow{t_{1,2}r^1_{1,2}t_{1,1}} M^1_3, \quad \xrightarrow{t_{3,2}t_{3,3}t_{3,1}} M^1_3;$$

$$M^1_1 = (p_{1,1}2p_{1,2}p_{1,3}p_{3,3}) \xrightarrow{r^1_{1,1}} M^1_2, \quad \xrightarrow{t_{1,2}r^1_{1,2}t_{1,1}} M^1_0, \quad \xrightarrow{t_{3,2}t_{3,3}t_{3,1}} M^1_0;$$

$$M^1_2 = (3p_{1,2}p_{1,3}p_{3,3}) \xrightarrow{t_{1,2}r^1_{1,2}t_{1,1}} M^1_1, \quad \xrightarrow{t_{3,2}t_{3,3}t_{3,1}} M^1_1;$$

$$M^1_3 = (3p_{1,1}p_{1,3}p_{3,3}) \xrightarrow{r^1_{1,1}} M^1_0, \quad \xrightarrow{t_{1,2}r^1_{1,2}} M^1_4, \quad \xrightarrow{t_{3,2}t_{3,3}} M^1_5;$$

$$M^1_4 = (4p_{1,1}p_{1,5}p_{3,3}) \xrightarrow{r^1_{1,1}t_{1,1}} M^1_3, \quad \xrightarrow{t_{3,2}t_{3,3}} M^1_6;$$

$$M^1_5 = (4p_{1,1}p_{1,3}p_{3,5}) \xrightarrow{r^1_{1,1}t_{3,1}} M^1_3, \quad \xrightarrow{t_{1,2}r^1_{1,2}} M^0_0 \Rightarrow to\ reconfig.\ \gamma_0;$$

$$M^1_6 = (5p_{1,1}p_{1,5}p_{3,5}) \xrightarrow{r^1_{1,1}t_{3,1}} M^1_4, \quad \xrightarrow{r^1_{1,1}t_{1,1}} M^0_5.$$

The state transition rate graph, with states $s^j_i = M^j_i$, j=0, 1, of the $ECTMC$ [2], associated with the model $R\Gamma^1_1$ in Figure 4, noted as $ECTMC1$, is shown in Figure 5. The graph $ECTMC1$ is built on the basis of RG1 and RG2 using approach presented in [2], [3]. In this graph, $\widetilde{\lambda}_{r_{1,1}} = M(p_{1,2}) \cdot \lambda_r$, $\widetilde{\lambda}_{1,2}$ and $\widetilde{\lambda}_{3,2}$ are the firing Z-fuzzy rate of the timed rewriting rule $r_{1,1}$ and those of the timed transitions $t_{1,2}$

and $t_{3,2}$, respectively. $q_{1,1}$ and $q_{3,1}$=$1-q_{1,1}$ are the firing probabilities of the immediate transitions $t_{1,1}$ and $t_{3,1}$, respectively.

The states of $ECTMC1$ are the reachability markings of the models DE_1^0 and DE_1^1, the respective configurations are γ_0 or γ_1, i.e., s_i^j=M_i^j, j=0,1. For this $ECTMC1$, we can write the Chapman-Kolmogorov linear equations system [1]-[3] the solution of which yields the steady-state probability distribution over the states $\overrightarrow{\pi}=\left(\pi_0^0, \ldots, \pi_3^0, \pi_0^1, \ldots, \pi_6^1\right)$. The MatLab software was used to this end.

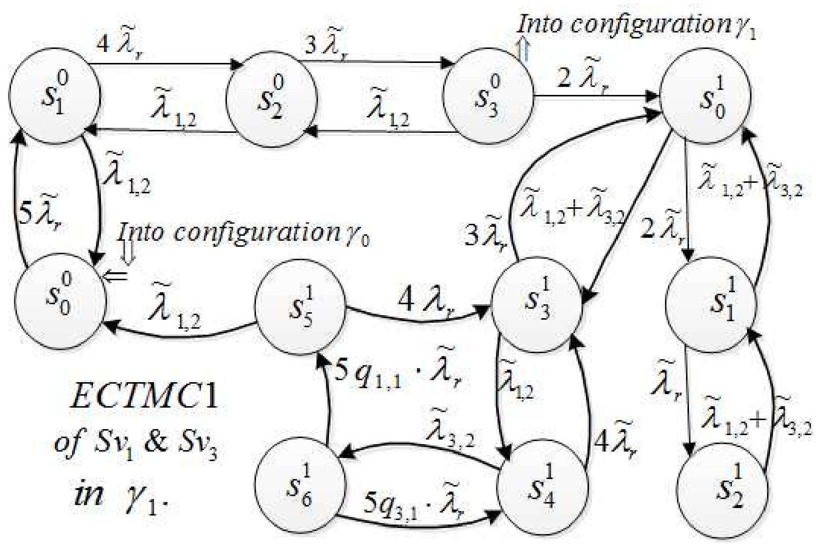

Figure 5. The $R\Gamma_1^1$ (see Figure 3) ECTMC1 state transition rate graph

Based on these probabilities $\overrightarrow{\pi}$, we can evaluate the performance measures specified according to some Z-fuzzy parameters that vary in a defined range. For example, we choose to evaluate the τ_{Buff1} average waiting time in the buffer Buf_1 of a job type C_1 depending on the Z-fuzzy firing rates: $\widetilde{\lambda}_{3,2}^Z$=$\left(\widetilde{\lambda}_{3,2}^A, \ \widetilde{\lambda}_{3,2}^R\right)$, where $\widetilde{\lambda}_{1,2}^A$=(1, 2, 2.5, 3.5; 1), $\widetilde{\lambda}_{3,2}^A$=(0.5, 1, 1.25, 1.75; 1), and $\widetilde{\lambda}_{1,2}^R$=$\widetilde{\lambda}_{3,2}^R$=(0.8, 0.9, 1; 1); $q_{1,1}$=0.6, $q_{3,1}$=0.4; $\lambda_{r_{1,1}}$=$M\left(p_{1,1}\right) \cdot \lambda_r$ varies depending on the parameter $\lambda_r \in$ [0.1, 6.0].

For Z-fuzzy parameters, we obtain: 1) the weight α=0.9 of the $\widetilde{\lambda}^R_{1,2}$=$\widetilde{\lambda}^R_{3,2}$ of $\widetilde{\lambda}^A_{1,2}$ and $\widetilde{\lambda}^A_{3,2}$, respectively; 2) the corresponding average credibility values are $\lambda^Z_{1,2}$=2.0250 and $\widetilde{\lambda}^Z_{3,2}$=1.0125.

Given the above input data and according to [2] and [3], the τ_{Buff1}=$n_{Buff1}/\lambda_{Buff1}$. Computation is performed as a function of the $\lambda_r \in [0.1,\ 6.0]$, where: τ_{Buff1} is the average sojourn time of a job in Buf_1; n_{Buff1}=$\sum_{i=0}^{3}\left(M_i^0\left(p_{1,2}\right)\cdot\pi_i^0\right)+\sum_{k=0}^{6}\left(M_k^1\left(p_{1,2}\right)\cdot\pi_k^1\right)$ is the average jobs number in Buf_1 and $\sum_{k=0}^{6}\left(M_k^1\left(p_{1,1}\right)\cdot\pi_k^1\right)$ is the throughput of $r_{1,1}$. The obtained results presented in Figure 6 show the reconfiguration from γ_0 to γ_1 impact on $\bar{\tau}_{Buff1}$.

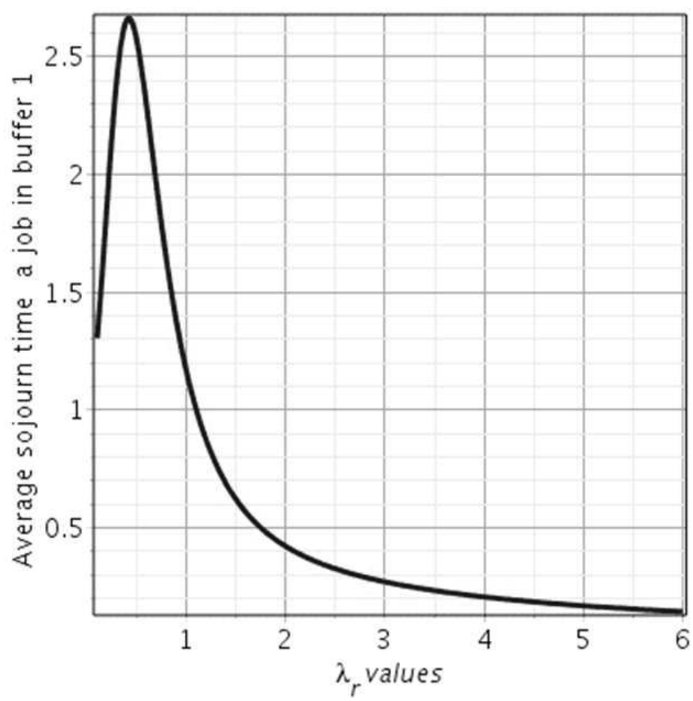

Figure 6. Average job sojourn time in Buf_1

Thus, $\bar{\tau}_{Buff1}$ increases in the configuration γ_0 up to $\bar{\tau}^{min}_{Buff1}$=

2.662317420 for λ_r=0.425, and then it decreases rapidly down to $\bar{\tau}_{Buff1}^{min}$=0.1433143438 for λ_r=6.0 upon the activation of Sv_3 in the configuration γ_1.

Since the class C_2 jobs processing in the CS configurations γ_0 and γ_1 is independent from those of the class C_1, the evaluation of their performances can be performed following the approach described in [2], [7], and [9], with the Z-fuzzy parameters respectively specified.

6 Conclusions

The paper presents the descriptive compositional approach for uncertainty modelling and performance evaluation of ReDES using ReSRN with Z-fuzzy parameters, called FReSRN, that can modify in run-time their own structure by some rewriting rules of their components. The Z-fuzzy expected values of the transition and rewriting rule firing rates are calculated based on credibility theory, and then the FReSRN model is degenerated to a conventional ReSRN model. A numerical example for performance modelling and analysis of a particular ReDES is given to show the effectiveness of the proposed method.

We will focus in our future works on developing a user friendly interface visual simulator software tool for verifying and performance evaluation of FReSRN models involving other kinds of transition time distributions laws and others firing rules.

References

[1] T. Murata, "Petri nets: properties, analysis and applications," *in Proc. of the IEEE*, vol. 77, no. 4, pp. 541–580, 1989.

[2] G. Chiola, M. A. Marsan, G. Balbo, and G. Conte, "Generalized stochastic Petri nets: A definition at the net level and its implications," *IEEE Transactions on Software Engineering*, vol. 19 no. 2, pp. 89–107, 1993.

[3] G. Ciardo, A. Blakemore, P. F. Chimento, J.K. Muppala, and K.S. Trivedi, "Automated generation and analysis of Markov reward models using stochastic reward nets," in *Linear Algebra, Markov Chains, and Queueing Models* (The IMA Volumes in Mathematics and its Applications, vol. 48), C.D. Meyer, R.J. Plemmons, Eds. New York, NY: Springer, 1993, pp. 145–191. DOI: https://doi.org/10.1007/978-1-4613-8351-2_11.

[4] M. Llorens and J. Oliver, "Structural and dynamic changes in concurrent systems: reconfigurable Petri nets," *IEEE Transactions on Computers*, vol. 53, no. 9, pp. 1147–1158, 2004.

[5] J. Li, X. Dai, Z. Meng, J. Dou, and X. Guan, "Rapid design and reconfiguration of Petri net models for reconfigurable manufacturing cells with improved net rewriting systems and activity diagrams," *Computers and Industrial Engineering*, vol. 57, no. 4, pp. 1431–1451, 2009.

[6] J. Padberg and L. Kahloul, "Overview of Reconfigurable Petri Nets," in *Graph Transformation, Specifications, and Nets* (Lecture Notes in Computer Science, vol 10800), Cham: Springer, 2018, pp. 201–222. DOI: https://doi.org/10.1007/978-3-319-75396-6_11.

[7] E. Guţuleac and M. L. Mocanu, "Descriptive Dynamic Rewriting GSPN-based Performance Modeling of Computer Systems," in *Proc. of the 15th International Conference on Control Systems and Computer Science*, 25-27 May 2005, pp. 656–661.

[8] S. Tigane, L. Kahloul, and S. Bourekkache,"Generalized stochastic Petri nets with rewritable topology," in *Proc. of International Conf. on EDiS*, 2017, pp. 1–6.

[9] F. Tuysuz and C. Kahraman, "Modeling a flexible manufacturing cell using stochastic Petri nets with fuzzy parameters," *Expert Systems with Applications*, vol. 37, pp. 3910–3920, 2010.

[10] L. A. Zadeh, "A note on Z-numbers," *Information Science*, no. 181, pp. 2923–2932, 2011.

[11] B. Kang, D. Wei, Y. Li, and Y. Deng, "A Method of Converting Z-number to Classical Fuzzy Number," *Journal of Information and Computational Science*, vol. 9. no. 3, pp. 703–709, 2012.

[12] B. Liu and Y. Liu, "Expected value of fuzzy variable and fuzzy expected value model," *IEEE Transactions on Fuzzy Systems*, vol. 10, no. 4, pp. 445–450, 2002.

Victor Moraru, Emilian Guţuleac,
Sergiu Zaporojan

Received March 20, 2021
Accepted July 30, 2021

Victor Moraru
Technical University of Moldova
168, Stefan cel Mare Bd., MD-2004, Chisinau, Republic of Moldova
Phone: + 373 60 60 96 04
E–mail: `victor.moraru@calc.utm.md`

Emilian Guţuleac
Technical University of Moldova
168, Stefan cel Mare Bd., MD-2004, Chisinau, Republic of Moldova
Phone: + 373 79 56 34 64
E–mail: `emilian.gutuleac@calc.utm.md`

Sergiu Zaporojan
Technical University of Moldova
168, Stefan cel Mare Bd., MD-2004, Chisinau, Republic of Moldova
Phone: + 373 68 22 82 12
E–mail: `sergiu.zaporojan@adm.utm.md`