Abstract—Workflow management becomes increasingly important in today’s information-oriented society. An important research area of workflow management is performance analysis that is driven by the need for improved efficiency of business processes. The study of the performance of a workflow process, as the focus of this paper, requires the estimation of the duration of tasks, which is often unpredictable and nondeterministic. Current research in this field has focused on workflow stochastic Petri nets (WF-SPNs)—which are a class of workflow nets with exponential distributed execution times assigned to transitions. In this paper, in order to deal with this uncertainty, we use fuzzy estimators constructed from statistical data to describe time, and we present an analytical method to proceed with the performance evaluation of workflow stochastic PNs based on block reduction. A comparison example is provided to show the benefits of the proposed method.

Index Terms—Fuzzy estimators, performance evaluation, workflow management, workflow nets, workflow reduction, workflow stochastic Petri nets (WF-SPNs).

I. INTRODUCTION

WORKFLOW is the automation of a business process, in whole or part, during which documents, information, or tasks are passed between one participant and another for action, according to a set of procedural rules [1]. Workflow management is concerned with the coordination, control, and communication of work, toward the goal of streamlined business processes, thereby eliminating unnecessary tasks, reducing costs, and/or improving operational efficiency. Since workflow nets [2] were established by van der Aalst as a proper way of designing and modeling workflow processes, they are widely used because of their clear semantics, graphical nature, expressive power, and strong analysis techniques.

The study of the performance of a workflow process is an important research area of workflow management, which is critical to the successful implementation of a workflow system. The main objectives of performance analysis are to evaluate existing or planned systems, to compare alternative configurations, and/or to find an optimal configuration of a system [3]. Existing research in this area has focused on timed Petri nets (PNs) [4], [5] and stochastic PNs (SPNs) [10] in which a deterministic and stochastic firing time is assigned to each transition, respectively. Ordinary workflow nets have been extended in various ways to include time [6]–[9]. Some studies consider durations to either places or transitions, others consider maximum and minimum timing constraints to either places or transitions, and others consider waiting times to tokens, but in general, a deterministic or stochastic quantitative analysis is performed based on extended with time models.

More recently, the integration of PNs and fuzzy logic has emerged as a promising approach in order to obtain a more realistic model of uncertainty [12]–[15]. However, although there is extensive work in the formalization of fuzzy PNs, there is a lack of research dealing with fuzzy performance analysis in the workflow domain.

In this paper, we are motivated by the work of Lin et al. [11], where an analytical performance analysis framework based on block reduction was developed. According to block reduction, the original model is successively reduced to simpler models. This allows us to perform performance calculations until the model is reduced to a workflow net consisting only of one transition and two places (source and sink place). The execution time of the last transition represents the workflow throughput time of the original model. Moreover, Lin et al. used statistical estimators as durations of tasks to conduct approximate performance analysis for workflow SPNs (WF-SPNs). We have chosen this framework (from now on, we will call it the classical approach) for the following reasons.

1) An analytical method to proceed with the performance analysis of workflow systems provides an accurate estimation with less computational cost compared to simulation approaches.
2) The block reduction approach does not lose any information required for calculating the workflow throughput time.
3) The throughput time is calculated easily using well-defined formulas.

Although the classical approach is precise and computationally efficient, only the estimated mean execution times are considered in the proposed performance formulas. We initiate the use of nonasymptotic fuzzy estimators based on the confidence intervals introduced by Papadopoulos and Sfiris in [21] (submitted for publication) in order to estimate the execution durations of tasks and propose a fuzzy performance evaluation methodology of WF-SPNs by means of block reduction. More specifically, we provide four well-defined formulas which can be used for direct construction of the fuzzy workflow throughput time for the four control structures: sequential, parallel,
selective, and iterative structures. The total fuzzy workflow throughput time can be calculated using the defined formulas and block reduction of workflow nets.

Integrating the fuzzy methodology into WF-SPNs provides us a robust representation of uncertain durations of tasks; thus, the reformulated framework enhances the capability of classical performance analysis when statistical data about execution times of tasks are available.

The remainder of this paper is organized as follows. In Section II, we review the existing literature on block reduction and performance analysis and discuss issues concerning the model applicability. In Section III, we introduce basic notions and definitions from PN theory and their application to workflow management that will be used herein. Section IV presents the way to construct nonasymptotic fuzzy estimators based on statistical data. In Section V, we present the calculation steps of the proposed methodology, and we prove the fuzzy formulas for the calculation of the workflow throughput time. Finally, in Section VI, we developed an application example, which helps in the presentation of the capabilities of the proposed approach. Conclusions follow in Section VII.

II. LITERATURE REVIEW

With the general growth of workflow management systems, it is hardly surprising that workflow languages have attracted considerable attention in recent years in an attempt to specify workflow processes. Van der Aalst et al. [25] systematically investigated the use of building blocks in the form of basic and complex workflow patterns. The basic workflow patterns include the AND-join, AND-split, OR-join, OR-split, loop, and sequence building blocks and can be found in almost all workflow language specifications. While some research has focused on complex patterns [25]–[27], authors agree that the implementation of complex patterns remains uncertain by most of the current workflow management systems. An in-depth comparison of the workflow patterns supported by a number of workflow products can be found in [25].

According to Leymann and Roller [20], three types of workflows are widely considered: 1) ad hoc workflows, which consist of both automatic and human processes and the tasks involved are nonrepetitive in nature; 2) administrative workflows, which are composed of mostly repetitive tasks; and 3) production workflows, which are composed of highly structured tasks. Much of the research on block reduction and performance evaluation has focused on production workflows. This is not a coincidence since the execution order of tasks in ad hoc and administrative workflows may depend upon the situation at run time, and therefore, it is not possible to develop an exact model and make accurate time estimations at design time. On the other hand, the exact structure of production workflows allows us to make effective structural and performance analysis optimizations. As Allen [30] put it, “the key goal of a production workflow is to automate as many activities as practical and to optimize productivity.”

Earlier research in workflow block reduction emphasized the use of well-structured workflow nets [26], [28]. Reijers [26] referred to a synthesis method where, instead of workflow patterns, workflow control structures were used. He argued that the difference between a workflow pattern and a workflow control structure is that each control structure is a well-structured workflow net. Thus, four workflow control structures were recognized: sequential, parallel, selective, and iterative structures. Based on this principle, Li et al. [28] suggested a Petri-nets-based approach for verifying the correctness of workflow models amenable to block reduction.

Existing literature on performance evaluation of PN and workflow nets can be classified into two general approaches: 1) the simulation approach [15], [16] and 2) the analytical approach [5], [17]–[19]. In the analytical approaches, Tian et al. [19] proposed a fuzzy estimate PN model for temporal workflow analysis using triangular fuzzy numbers. They argued that the fuzzy execution times of tasks can be assessed either empirically from expert opinions or they can be deduced by a data mining process from statistical data and developed fuzzy performance formulas for workflow control structures based on expert opinions. Other authors [5], [17], [18] have taken a different approach by using methods based on Markov chains theory to aim at the development of efficient performance metrics. In the simulation approaches, Dehnert et al. [16] presented a modeling methodology based on colored SPNs and showed how it can be used for performance evaluation of workflows. The authors generally agree that analytical methods are preferable over simulation ones because of the computational correct results that they yield; however, there is clearly a great deal of more research that needs to be done to extend existing studies in more ways.

III. BACKGROUND

In this section, we describe and present the basic notions and fundamental principles of PNs, SPNs, and their application to workflow management.

A PN is a bipartite-directed graph which has place nodes, transition nodes, and directed arcs connecting places with transitions. Formally, a PN is a tuple \((P,T,F)\) with the following.

1. \(P = \{p_1, p_2, \ldots, p_n\}\) is a finite set of places.
2. \(T = \{t_1, t_2, \ldots, t_m\}\) is a finite set of transitions \((P \cap T = \emptyset)\).
3. \(F \subseteq (P \times T) \cup (T \times P)\) is a set of arcs, known as flow relation.

We denote by \(\cdot t\) the set of all input places of transition \(t\), which we call the preset of \(t\). We denote by \(t\cdot\) the set of all output places of transition \(t\), which we call the postset of \(t\). The same notation applies for places as well.

A marked PN is a four-tuple \((P,T,F,M_0)\), where \((P,T,F)\) is a PN according to the previous definition and \(M_0 : P \rightarrow N\) (the set of non-negative integers) is the initial marking of the places in the net.

A PN \(PN = (P,T,F)\) is called a workflow net if and only if the following conditions hold.

1. There is one source place \(i \in P\) such that \(i = \emptyset\).
2. There is one sink place \(o \in P\) such that \(o = \emptyset\).

If an extra transition \(t_e\) is added to \(PN\) which connects place \(o\) with \(i\), i.e., \(t_e = \{o\}\) and \(t_e. = \{i\}\), then the resulting PN is
strongly connected. That is, every node \( x \in P \cup T \) is on a path from \( i \) to \( o \) [22].

An SPN is a PN in which each transition has an exponentially distributed firing time. Formally, a marked SPN is a five-tuple \( \langle P, T, F, M_0, \Lambda \rangle \), where \( \Lambda : T \rightarrow R^+ \) (the set of non-negative real numbers) and \( \lambda_k \in \Lambda \) is the firing rate of transition \( t_k \in T \).

A WF-SPN is an SPN which fulfills the same conditions, as mentioned earlier, with a workflow net.

From a statistical point of view, it is well known that the exponential distribution [23] has the form
\[
    f_x(x) = \begin{cases} 
        \lambda e^{-\lambda x}, & 0 \leq x \leq \infty \\
        0, & x < 0 
    \end{cases}
\]
where \( \lambda > 0 \). The expected value and the variance of the exponential distribution are
\[
    E(X) = \frac{1}{\lambda}, \quad \text{Var}(X) = \frac{1}{\lambda^2}.
\]

Let \( X \) be a random variable and \( X_1, X_2, \ldots, X_n \) be a random sample that is exponentially distributed. If \( x_1, x_2, \ldots, x_n \) are sample values, then the likelihood function for \( \lambda \) is
\[
    L(\lambda) = \prod_{i=1}^{n} \lambda \exp(-\lambda x_i) = \lambda^n \exp(-\lambda n \bar{x}).
\]

Thus, \( \hat{\lambda} = 1/\bar{x} \), where \( \bar{x} = (1/n) \sum_{i=1}^{n} x_i \) is the sample mean and \( \hat{\lambda} \) is the maximum likelihood estimate for \( \lambda \).

Now, let us give some well-known definitions and notations from the theory of fuzzy sets which will be used in the following to define a fuzzy timed workflow net (FTWN) model.

The definition of a fuzzy set is as follows. Let \( X \) be a set of real numbers \( R \) (or "crisp") set. Every function \( \alpha \) assigns a "membership degree" of \( \lambda \) to each element in \( X \).

Fuzzy sets are generalizations of the classical sets, but in contrast to the classical sets, a continuum of membership degrees is allowed. In this paper, we will consider \( X \) to be the set of real numbers \( R \).

If \( A \) is a fuzzy set, then by \( \alpha \)-cuts, we mean the sets \( A_\alpha = \{ x \in R : A(x) \geq \alpha \} \). It is known that the \( \alpha \)-cuts determine the fuzzy set \( A \). We say that \( A \) is a fuzzy number if the following conditions hold.

1) \( A \) is normal, i.e., there exists \( x \in R \) such that \( A(x) = 1 \).
2) \( A \) is a convex fuzzy set, i.e., for every \( t \in [0,1] \) and \( x_1, x_2 \in R \), we have
   \[
   A( (1-t)x_1 + tx_2 ) \geq \min \{ A(x_1), A(x_2) \}.
   \]
3) \( A \) is upper semicontinuous.
4) The support of \( A \)
   \[
   \text{supp}A = \bigcup_{\alpha \in (0,1]} A = \{ x : A(x) > 0 \}
   \]
is compact [24].

In several practical applications appear operations and elementary functions which involve fuzzy numbers. Zadeh’s extension principle [24] is used for the purpose of extending the classical operations between real numbers and their fuzzy counterparts. Based on 1), 3), and 4) and the extension principle, \( \forall u \) and for \( v \in A \) and \( 0 \leq \alpha \leq 1 \), the \( \alpha \)-set operations are the following.

1) \( [u + v]_\alpha = [u]_\alpha + [v]_\alpha = \{ \alpha + b : \alpha \in [u]_\alpha, b \in [v]_\alpha \} \).
2) \( [u - v]_\alpha = [u]_\alpha - [v]_\alpha = \{ \alpha - b : \alpha \in [u]_\alpha, b \in [v]_\alpha \} \).
3) \( [uv]_\alpha = [u]_\alpha [v]_\alpha = \{ ab : \alpha \in [u]_\alpha, b \in [v]_\alpha \} \).
4) \( [u/v]_\alpha = [u]_\alpha / [v]_\alpha = \{ ab : \alpha \in [u]_\alpha, b \in [v]_\alpha \} \).

A fuzzy timed PN (FTPN) [13] is a tuple \( \langle P, T, F, M \rangle \) with the following.

1) \( P = [p_1, p_2, \ldots, p_n] \) is a finite nonempty set of places.
2) \( T = [t_1, t_2, \ldots, t_m] \) is a finite nonempty set of transitions.
3) \( F \subseteq (P \times T) \cup (T \times P) \) is a set of arcs (flow relations).
4) \( W : F \rightarrow N \) is a mapping to assign weight to each arc, and \( M : T \rightarrow A \) is a mapping to assign fuzzy firing time to each transition.

We define an FWTN as a FTPN which also satisfies the following conditions.

1) There is one source place \( i \in P \) such that \( i = \emptyset \).
2) There is one sink place \( o \in P \) such that \( o = \emptyset \).
3) Every node \( x \in P \cup T \) is on a path from \( i \) to \( o \).

IV. NONASYMPTOTIC FUZZY ESTIMATORS

In this section, we present our method of constructing nonasymptotic fuzzy estimators [21] (submitted for publication). In previous studies to construct fuzzy estimators from statistical data, Wagner [32] describes a method of constructing \( \alpha \)-cuts on the basis of observed data, and Buckley [33] uses fuzzy numbers for parameters in probability density (mass) functions that have been estimated from random samples. Here, instead of considering the confidence intervals as \( \alpha \)-cuts, we construct fuzzy estimators in a more natural way using all the \( \alpha \)-cuts and doing an appropriate transform, such that, on the one hand, we ensure compact support for these estimators and, on the other hand, we have an analytical form of them.

Let \( X_1, X_2, \ldots, X_n \) be a random sample, and let \( x_1, x_2, \ldots, x_n \) be sample values assumed by the sample. Let also \( \beta \in (0,1) \). If the sample size is large enough, then

\[
M(x) = \begin{cases} 
\frac{2}{1-\beta} \Phi \left( \frac{x-\bar{x}}{\sigma/\sqrt{n}} \right), & \text{if } \bar{x} - \sigma/\sqrt{n} \Phi^{-1} \left( 1 - \frac{\beta}{2} \right) \leq x \leq \bar{x} \\
\frac{2}{1-\beta} \Phi \left( \frac{x-\bar{x}}{\sigma/\sqrt{n}} \right) + \frac{2}{1-\beta}, & \text{if } \bar{x} \leq x \leq \bar{x} + \sigma/\sqrt{n} \Phi^{-1} \left( 1 - \frac{\beta}{2} \right)
\end{cases}
\]

is a fuzzy number, the base of which is exactly the \( 1 - \beta \) confidence interval for \( \mu \), and the \( \alpha \)-cuts of this fuzzy number are the closed intervals
\[
\alpha M = \left[ \bar{x} - K_{\gamma}(a) \frac{\sigma}{\sqrt{n}}, \bar{x} + K_{\gamma}(a) \frac{\sigma}{\sqrt{n}} \right]
\]
which are exactly the \((1 - \alpha)(1 - \beta)\) confidence intervals for \(\mu\), where
\[
g(\alpha) = \left(\frac{1}{2} - \frac{\beta}{2}\right) \alpha + \frac{\beta}{2}, \quad \left(\alpha : [0, 1] \rightarrow \left[\frac{\beta}{2}, 0.5\right]\right)
\]
\(K_{g(\alpha)} = \Phi^{-1}(1 - g(\alpha))\), and \(\Phi\) denotes the standard normal distribution function. The graph of this fuzzy number is shown in Fig. 1.

V. FUZZY WORKFLOW THROUGHPUT CALCULATION

This section describes and analyzes our proposed methodology. The workflow throughput time is the average time required for a single case to be completed and includes service time and sojourn time [26]. Before describing the performance evaluation, we need time information about the tasks. We collect such information through execution logs by applying a process mining technique for the timed workflow net model. The procedure starts by putting a token in place \(\{i\}\). Then, every time a transition in the timed workflow net is fired, service and sojourn times are computed, and a new timestamp is assigned to the token. Thus, by computing the difference between the timestamps, we collect the execution times for each task.

The workflows which we are concerned are the production workflows, and the PN model considered for performance evaluation are WF-SPNs. As a statistical model, WF-SPNs have their own assumptions that have to be verified and met. The probability distribution assumed in this model is the exponential distribution. This means that the probability of execution of the next task is independent of the occurrence time of the past execution.

The assumption of independence implies that sampling of task executions has been performed as described before applying our performance evaluation methodology. Having knowledge about the execution times of the tasks is needed in order to assess that the observations are independent and representative of the population from which they come. To assess the exponentiality of the data, one can use formal methods (chi square, Anderson–Darling, Kolmogorov–Smirnov, etc.), which usually require lengthy calculations, or one of the informal methods presented in [35] based on the properties of the exponential distribution.

We use a fuzzy estimator constructed from statistical data for the parameter \(\lambda\) of the exponential distribution. In that way, we obtain workflow FTPNs on top of WF-SPNs. In the following are the calculation steps of our methodology.

Step 1) Construction of a fuzzy estimator from sample data. Suppose that we have an exponentially distributed data set of observations concerning the execution time of some task. Then, the mean execution time of the task can be defined, as shown in Section IV, either by the \(\alpha\)-cuts of the nonasymptotic fuzzy estimator or by its analytical form. Using (2), the \(\alpha\)-cuts of the fuzzy expectation value are
\[
aE[X] = \left[\mu - K_{g(\alpha)} \frac{\sigma}{\sqrt{n}}, \mu + K_{g(\alpha)} \frac{\sigma}{\sqrt{n}}\right].
\]

Step 2) Calculation of fuzzy parameter \(\lambda\) of the exponential distribution. Since
\[
\lambda = \frac{1}{E[X]}
\]
then according to [34, Proposition 5.1], the set representation of the fuzzy \(\lambda\) is
\[
[\lambda_{a,l}, \lambda_{a,r}] = \left[\frac{1}{E[X]_l}, \frac{1}{E[X]_r}\right] = \left[\mu + K_{g(\alpha)} \frac{\sigma}{\sqrt{n}}, \mu - K_{g(\alpha)} \frac{\sigma}{\sqrt{n}}\right]^{-1}
\]
\[
\Leftrightarrow \quad [\lambda_{a,l}, \lambda_{a,r}] = \left[\frac{1}{\mu + K_{g(\alpha)} \frac{\sigma}{\sqrt{n}}}, \frac{1}{\mu - K_{g(\alpha)} \frac{\sigma}{\sqrt{n}}}\right]^{-1}
\]
where \(l\) denotes the left side and \(r\) denotes the right side of an \(\alpha\)-cut interval.

Step 3) By applying steps 1) and 2) on the exponentially distributed data set of observations for each task, we obtain the corresponding fuzzy parameter \(\lambda\) for all tasks similarly.

We will now provide the fuzzy performance formulas for computing the overall fuzzy throughput time in WF-SPNs. The execution times of tasks are now fuzzy numbers estimated from samples. We prove the fuzzy throughput time for sequential, parallel, selective, and iterative structures. First, a sequential and a parallel structure of two tasks are examined in order to show our approach. Then, we generalize the results to more tasks and examine other structures.

Proposition 5.1: If there are two transitions \(t_1, t_2\) in a row, as shown in Fig. 2(a), then \(t_1\) and \(t_2\) can be replaced by \(t_{12}\) [Fig. 2(b)], and the throughput time is
\[
\frac{1}{\lambda} = \frac{1}{\lambda_1} + \frac{1}{\lambda_2}
\]
where \(\lambda, \lambda_1,\) and \(\lambda_2\) are crisp numbers, representing the parameters of the corresponding exponential distributions. Let \(M, M_1,\) and \(M_2\) be nonasymptotic fuzzy estimators that describe the mean execution time of transitions. Then, according to (3) and (4), the operation (6) becomes an operation among fuzzy numbers.
Since \( g(\alpha) = ((1/2) - (\beta/2))\alpha + (\beta/2) \) (as that in Section IV), we get

\[
\alpha = \frac{2}{1 - \beta} \Phi \left( \frac{x - (\bar{t}_1 + \bar{t}_2)}{\sqrt{\frac{\sigma_1}{\bar{n}_1} + \frac{\sigma_2}{\bar{n}_2}}} \right) + \frac{2 - \beta}{1 - \beta}.
\]

Finally, since \( 0 \leq \alpha \leq 1 \), (7) is true when

\[
(\bar{t}_1 + \bar{t}_2) - \left( \frac{\sqrt{\bar{n}_2}\sigma_1 + \sqrt{\bar{n}_1}\sigma_2}{\sqrt{\bar{n}_1\bar{n}_2}} \right) \Phi^{-1} \left( 1 - \frac{\beta}{2} \right) \leq x \leq (\bar{t}_1 + \bar{t}_2).
\]

The procedure is analogous for the other side of the \( \alpha \)-cut interval. Therefore, the workflow fuzzy throughput time has the analytical form shown near the bottom of the page.

**Proposition 5.2:** Let \( X_1, X_2, \ldots, X_n \) be mutually independent exponentially distributed random variables, and let \( Y_i = (Y_{i,1}, Y_{i,2}, \ldots, Y_{i,n}) \) be a random sample that is large enough from each random variable \( X_i \). If \( Y_{i} = t_{i,1}, Y_{i} = t_{i,2}, \ldots, Y_{i} = t_{i,n} \) are sample values, the expression shown at the bottom of the page is a fuzzy number representing the workflow fuzzy throughput time of \( k \) sequential tasks, the \( \alpha \)-cuts of which are the closed intervals

\[
M_a = \left[ \sum_{i=1}^{k} \bar{t}_i - K_{g(a)} \left( \sum_{i=1}^{k} \frac{\sigma_i}{\bar{n}_i} \right), \sum_{i=1}^{k} \bar{t}_i + K_{g(a)} \left( \sum_{i=1}^{k} \frac{\sigma_i}{\bar{n}_i} \right) \right]
\]

for each \( a \in [0, 1] \).

**Proposition 5.3:** If there is a structure of two parallel transitions, then the throughput time for WF-SPNs is (a special case of [11, Th. 2])

\[
1 = \lambda_1 + \lambda_2 - \lambda_1 \lambda_2
\]

\[
M(x) = \begin{cases} 
\frac{2}{1 - \beta} \Phi \left( \frac{x - (\bar{t}_1 + \bar{t}_2)}{\sqrt{\frac{\sigma_1}{\bar{n}_1} + \frac{\sigma_2}{\bar{n}_2}}} \right) + \frac{2 - \beta}{1 - \beta}, & \text{if } (\bar{t}_1 + \bar{t}_2) - \left( \frac{\sqrt{\bar{n}_2}\sigma_1 + \sqrt{\bar{n}_1}\sigma_2}{\sqrt{\bar{n}_1\bar{n}_2}} \right) \Phi^{-1} \left( 1 - \frac{\beta}{2} \right) \leq x \leq (\bar{t}_1 + \bar{t}_2) \\
\frac{2}{1 - \beta} \Phi \left( \frac{(\bar{t}_1 + \bar{t}_2) - x}{\sqrt{\frac{\sigma_1}{\bar{n}_1} + \frac{\sigma_2}{\bar{n}_2}}} \right) + \frac{2 - \beta}{1 - \beta}, & \text{if } (\bar{t}_1 + \bar{t}_2) \leq x \leq (\bar{t}_1 + \bar{t}_2) + \left( \frac{\sqrt{\bar{n}_2}\sigma_1 + \sqrt{\bar{n}_1}\sigma_2}{\sqrt{\bar{n}_1\bar{n}_2}} \right) \Phi^{-1} \left( 1 - \frac{\beta}{2} \right)
\end{cases}
\]

\[
M(x) = \begin{cases} 
\frac{2}{1 - \beta} \Phi \left( \frac{x - \sum_{i=1}^{k} \bar{t}_i}{\left( \sum_{i=1}^{k} \frac{\sigma_i}{\bar{n}_i} \right)} \right) + \frac{2 - \beta}{1 - \beta}, & \text{if } \sum_{i=1}^{k} \bar{t}_i - \left( \sum_{i=1}^{k} \frac{\sigma_i}{\bar{n}_i} \right) \Phi^{-1} \left( 1 - \frac{\beta}{2} \right) \leq x \leq \sum_{i=1}^{k} \bar{t}_i \\
\frac{2}{1 - \beta} \Phi \left( \frac{\sum_{i=1}^{k} \bar{t}_i - x}{\left( \sum_{i=1}^{k} \frac{\sigma_i}{\bar{n}_i} \right)} \right) + \frac{2 - \beta}{1 - \beta}, & \text{if } \sum_{i=1}^{k} \bar{t}_i \leq x \leq \sum_{i=1}^{k} \bar{t}_i + \left( \sum_{i=1}^{k} \frac{\sigma_i}{\bar{n}_i} \right) \Phi^{-1} \left( 1 - \frac{\beta}{2} \right)
\end{cases}
\]
where $\lambda$, $\lambda_1$, and $\lambda_2$ are the parameters of the corresponding exponential distributions. Fig. 3 shows a parallel structure of two tasks.

Since, in our approach, $\lambda_1$ and $\lambda_2$ are fuzzy estimators, (8) becomes

$$[M_{r,a}, M_{r,a}] = \frac{1}{[\lambda_{1r,a}, \lambda_{1r,a}]} + \frac{1}{[\lambda_{2r,a}, \lambda_{2r,a}]} - \frac{1}{[\lambda_{1r,a} + \lambda_{2r,a}]}.$$

So

$$M_{r,a} = \left[ t_1 + t_2 - K_2 \left( \frac{\sigma_1}{\sqrt{n_1}} + \frac{\sigma_2}{\sqrt{n_2}} \right) \right. $$

$$\left. - \left( t_1 + K_2 \frac{\sigma_1}{\sqrt{n_1}} \right) \left( t_2 + K_2 \frac{\sigma_2}{\sqrt{n_2}} \right) \right],$$

$$\left. \left( t_1 + t_2 + K_2 \left( \frac{\sigma_1}{\sqrt{n_1}} + \frac{\sigma_2}{\sqrt{n_2}} \right) \right) \right],$$

$$\left. - \left( t_1 - K_2 \frac{\sigma_1}{\sqrt{n_1}} \right) \left( t_2 - K_2 \frac{\sigma_2}{\sqrt{n_2}} \right) \right],$$

$$\left. \left( t_1 + t_2 - K_2 \left( \frac{\sigma_1}{\sqrt{n_1}} + \frac{\sigma_2}{\sqrt{n_2}} \right) \right) \right],$$

for each $\alpha \in [0, 1]$ are the $\alpha$-cuts of the fuzzy number representing the workflow fuzzy throughput time. In a similar way, it is easy to generalize to any number of parallel tasks.

![Fig. 3. (a) Illustration of a workflow net with two parallel transitions $t_1$ and $t_2$. $M_1$ and $M_2$ are fuzzy numbers representing time associated with transitions. Nonlabeled transitions have zero duration. (b) Reduced model of the workflow net shown in (a).](image1)

![Fig. 4. (a) Selective routing structure of $k$ tasks. (b) Reduced model of the workflow net shown in (a).](image2)

**Proposition 5.4:** When we have $n$ parallel tasks, then the formula for the total throughput time for WF-SPNs is

$$\frac{1}{\lambda} = \sum_{i=1}^{n} \frac{1}{\lambda_i} - \sum_{j=i+1}^{n} \frac{1}{\lambda_i + \lambda_j} + \sum_{j=i+1}^{n-1} \sum_{k=j+1}^{n} \frac{1}{\lambda_i + \lambda_j + \lambda_k} + \cdots + (-1)^{n-1} \frac{1}{\sum_{i=1}^{\infty} \lambda_i}. \quad (9)$$

Let $\bar{t} = 1/\lambda$ and $\bar{t}_i = 1/\lambda_i$, for all $1 \leq i \leq n$. Then

$$\bar{t} = \sum_{i=1}^{n} \bar{t}_i - \sum_{i<j} \bar{t}_i \bar{t}_j + \sum_{i<j<k} \bar{t}_i \bar{t}_j \bar{t}_k + \cdots + (-1)^{n-1} \frac{1}{\sum_{i=1}^{\infty} \bar{t}_i}. \quad (10)$$

Then, using [34, Prop. 5.1], from (10), we get the equation shown at the bottom of the page. Since we can estimate the $\alpha$-cuts for the fuzzy mean duration of all tasks, we can estimate the $\alpha$-cuts for the fuzzy throughput time using the equations shown near the middle of the next page.

**Proposition 5.5:** Sometimes, a choice between two or more tasks has to be made. In Fig. 4, a selective structure is illustrated, consisting of $k$ transitions from which only one can fire. Each transition can fire with probability $a_i$ and $\sum_{i=1}^{k} a_i = 1$. Let $X_1, X_2, \ldots, X_n$ be mutually independent exponentially distributed random variables, and let $Y_i = (Y_{i,1}, Y_{i,2}, \ldots, Y_{i,n})$
be a random sample that is large enough from each random variable $X_i$. If $Y_i = t_{i,1}, Y_i = t_{i,2}, \ldots, Y_i = t_{i,n}$ are sample values, then from [11, Th. 3], the throughput time of the selective structure can be calculated as

$$\frac{1}{\lambda} = \sum_{i=1}^{k} \frac{a_i}{\lambda_i}.$$ \hspace{1cm} (11)

The $\alpha$-cuts of the fuzzy number representing the fuzzy throughput time of the selective structure of $k$ tasks are

$$M_{\alpha} = \sum_{i=1}^{k} \left[ \frac{a_i}{\lambda_{i,r}} \frac{a_i}{\lambda_{i,l}} \right]_{\alpha} = \sum_{i=1}^{k} \left( \bar{t}_i - K_g(\alpha) \frac{\sigma_i}{\sqrt{n_i}} \right) a_i = \sum_{i=1}^{k} \left( \bar{t}_i + K_g(\alpha) \frac{\sigma_i}{\sqrt{n_i}} \right) a_i$$

for each $0 \leq \alpha \leq 1$.

We derive the analytical form of $M$ as follows:

$$\sum_{i=1}^{k} \left( \bar{t}_i - K_g(a) \frac{\sigma_i}{\sqrt{n_i}} \right) a_i = x \Leftrightarrow \sum_{i=1}^{k} \bar{t}_i a_i - x \Leftrightarrow K_g(a) = \frac{\sum_{i=1}^{k} \bar{t}_i a_i - x}{\sum_{i=1}^{k} \frac{\sigma_i}{\sqrt{n_i}} a_i}.$$

$$\Leftrightarrow \Phi^{-1} \left( 1 - g(a) \right) = \frac{\sum_{i=1}^{k} \bar{t}_i a_i - x}{\sum_{i=1}^{k} \frac{\sigma_i}{\sqrt{n_i}} a_i} \Leftrightarrow \alpha = \frac{2 - \beta}{1 - \beta} - 2 \Phi \left( \frac{k}{\sum_{i=1}^{k} \frac{\sigma_i}{\sqrt{n_i}} a_i} \sqrt{\frac{n_i}{\sum_{i=1}^{k} \frac{\sigma_i}{\sqrt{n_i}} a_i}} \right).$$ \hspace{1cm} (12)

Since $0 \leq \alpha \leq 1$, then (12) is true when

$$\sum_{i=1}^{k} \bar{t}_i a_i - x \leq \frac{\sum_{i=1}^{k} \bar{t}_i a_i - x}{\sum_{i=1}^{k} \frac{\sigma_i}{\sqrt{n_i}} a_i} \leq \sum_{i=1}^{k} \bar{t}_i a_i.$$

The procedure is the same for the other side of the $\alpha$-cut interval. Therefore, the workflow fuzzy throughput time of $k$ sequential tasks is the equation shown at the bottom of the page.

\[\text{Fig. 5. (a) Iterative routing structure of the two transitions } t_1 \text{ and } t_2. \text{ (b) Reduced model of the workflow net.}\]
**Proposition 5.6:** A process also may include iteration, namely, the repeated execution of a task. Fig. 5 shows the repeated execution of transitions $t_1$ and $t_2$.

According to [11, Th. 4], the throughput time of the iterative structure in a WF-SPN model can be calculated as

$$\frac{1}{\lambda} = \frac{1}{1 - a} \left( \frac{a}{\lambda_1} + \frac{1}{\lambda_2} \right). \tag{13}$$

Since

$$M_i = \left[ \tilde{t}_i - K_{g(\alpha)} \frac{\sigma_1}{\sqrt{n_i}}, \tilde{t}_i + K_{g(\alpha)} \frac{\sigma_1}{\sqrt{n_i}} \right], \quad \text{for } 1 \leq i \leq 2 \tag{14}$$

from (13), we obtain

$$[M_l, M_r] = \frac{1}{1 - a} \left( \left[ \frac{a}{\lambda_{1,r}}, \frac{a}{\lambda_{1,l}} \right] + \left[ \frac{1}{\lambda_{2,r}}, \frac{1}{\lambda_{2,l}} \right] \right) \tag{15}$$

where $[M_l, M_r]$ are the $\alpha$-cuts of the fuzzy throughput time of the iteration structure. Then, by combining (14) and (15)

$$M_l = \frac{a}{1 - a} \left( \tilde{t}_1 - K_{g(\alpha)} \frac{\sigma_1}{\sqrt{n_1}} \right) + \frac{1}{1 - a} \left( \tilde{t}_2 - K_{g(\alpha)} \frac{\sigma_2}{\sqrt{n_2}} \right)$$

$$M_r = \frac{a}{1 - a} \left( \tilde{t}_1 + K_{g(\alpha)} \frac{\sigma_1}{\sqrt{n_1}} \right) + \frac{1}{1 - a} \left( \tilde{t}_2 + K_{g(\alpha)} \frac{\sigma_2}{\sqrt{n_2}} \right).$$

To summarize, we have identified the fuzzy formulas for the sequential, parallel, selective, and iterative structures. We are now able to compute the total fuzzy workflow throughput using block reduction of workflow nets. The following rules describe the block reduction procedure [28].

1) The beginning of the control structure is always one source place, while the ending of the control structure is always one sink place.
2) Two control structures can be chained together by merging the sink place of the one control structure with the source place of the other chained one.
3) Each control structure can be replaced by one transition.
4) Each transition can be replaced by any of the control structures. Thus, its input place will become the source place of the replacing control structure, and its output place will become the sink place of the replacing control structure.

**VI. EXAMPLE TO CLARIFY THE BLOCK REDUCTION APPROACH**

In this section, an application example for a typical production workflow system is used to demonstrate the capability of the proposed method. The application concerns the automatic responding part of an online ordering process. Fig. 6 shows a graphical representation of the developed workflow model. This is a production workflow which involves highly structured tasks executed by the online electronic system. The fuzzy performance evaluation of WF-SPNs is compared with the classical approach presented in [11].

The workflow is started when a customer makes an order through a Web browser. This is represented by a token in the source place $i$. The automatic online ordering system receives the data sent by the customer (customer name and address, ordered products, quantities and shipping details, credit card number, etc.) and performs a series of parallel and sequential operations until the order has been processed. Finally, it informs the customer via e-mail. The descriptions of the transitions are shown in Table I.

In practice, all transitions (tasks) have some duration; however, in this example, only the labeled transitions have an execution time. Lin et al. [11] considered the mean execution times of tasks to be real numbers. However, in this example, the durations of tasks are unpredictable and nondeterministic depending upon various factors such as the current workload of various parts of the system and the speed of communication.
TABLE II
NONASYMPTOTIC FUZZY ESTIMATORS FOR THE MEAN EXECUTION TIMES OF TASKS

<table>
<thead>
<tr>
<th>α-cuts</th>
<th>Analytical form</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f_1$</td>
<td>${4.89 - K_{\theta}^{4.79}/\sqrt{100}, 4.89 + K_{\theta}^{4.79}/\sqrt{100}}$, M($\alpha$) = 1.9, $2/0.9 - 2/0.9$</td>
</tr>
<tr>
<td>$f_2$</td>
<td>${2.84 - K_{\theta}^{2.72}/\sqrt{100}, 2.84 + K_{\theta}^{2.72}/\sqrt{100}}$, M($\alpha$) = 1.9, $2/0.9 - 2/0.9$</td>
</tr>
<tr>
<td>$f_3$</td>
<td>${6.03 - K_{\theta}^{5.97}/\sqrt{100}, 6.03 + K_{\theta}^{5.97}/\sqrt{100}}$, M($\alpha$) = 1.9, $2/0.9 - 2/0.9$</td>
</tr>
<tr>
<td>$f_4$</td>
<td>${4.68 - K_{\theta}^{4.68}/\sqrt{100}, 4.68 + K_{\theta}^{4.68}/\sqrt{100}}$, M($\alpha$) = 1.9, $2/0.9 - 2/0.9$</td>
</tr>
<tr>
<td>$f_5$</td>
<td>${3.35 - K_{\theta}^{3.09}/\sqrt{100}, 3.35 + K_{\theta}^{3.09}/\sqrt{100}}$, M($\alpha$) = 1.9, $2/0.9 - 2/0.9$</td>
</tr>
<tr>
<td>$f_6$</td>
<td>${2.06 - K_{\theta}^{2.05}/\sqrt{100}, 2.06 + K_{\theta}^{2.05}/\sqrt{100}}$, M($\alpha$) = 1.9, $2/0.9 - 2/0.9$</td>
</tr>
<tr>
<td>$f_7$</td>
<td>${3.83 - K_{\theta}^{3.30}/\sqrt{100}, 3.83 + K_{\theta}^{3.30}/\sqrt{100}}$, M($\alpha$) = 1.9, $2/0.9 - 2/0.9$</td>
</tr>
<tr>
<td>$f_8$</td>
<td>${2.28 - K_{\theta}^{2.28}/\sqrt{100}, 2.28 + K_{\theta}^{2.28}/\sqrt{100}}$, M($\alpha$) = 1.9, $2/0.9 - 2/0.9$</td>
</tr>
<tr>
<td>$f_9$</td>
<td>${2.70 - K_{\theta}^{2.70}/\sqrt{100}, 2.70 + K_{\theta}^{2.70}/\sqrt{100}}$, M($\alpha$) = 1.9, $2/0.9 - 2/0.9$</td>
</tr>
</tbody>
</table>

The third column contains the analytical form of the fuzzy numbers. These numbers declare the fuzzy time required for the execution of each transition. The analytical form, which is presented in the third column, is a straightforward way to compute the fuzzy execution time. Conclusively, Table II concerns a concrete step of the proposed methodology.

The sample standard deviations

$\sigma_1 = 4.7878$  $\sigma_2 = 2.7217$  $\sigma_3 = 5.9663$
$\sigma_4 = 4.6815$  $\sigma_5 = 3.0906$  $\sigma_6 = 2.0468$
$\sigma_7 = 3.8341$  $\sigma_8 = 2.2767$  $\sigma_9 = 2.6991$.

Furthermore, the probability $\alpha$ (to accept an order) is 90%.

To assess the exponentiality of the data, one can use one of several methods presented in [35] based on the properties of the exponential distribution. All the constructed fuzzy estimators for the mean execution times of tasks with a confidence interval of 0.90% are shown in Table II. The first column contains the transitions, the second column contains the α-cuts, and the third column contains the analytical form of the fuzzy estimators from actual sample data taken from the aforementioned application. All values are in seconds.

The sample means of the samples taken turned out to be

$\bar{t_1} = 4.8941$  $\bar{t_2} = 2.8364$  $\bar{t_3} = 6.0325$
$\bar{t_4} = 4.5016$  $\bar{t_5} = 3.3545$  $\bar{t_6} = 2.0624$
$\bar{t_7} = 3.0056$  $\bar{t_8} = 2.8152$  $\bar{t_9} = 3.0206$

and the sample standard deviations

$\sigma_1 = 4.7878$  $\sigma_2 = 2.7217$  $\sigma_3 = 5.9663$
$\sigma_4 = 4.6815$  $\sigma_5 = 3.0906$  $\sigma_6 = 2.0468$
$\sigma_7 = 3.8341$  $\sigma_8 = 2.2767$  $\sigma_9 = 2.6991$.

Fig. 7 shows the successive steps of the workflow block reduction procedure. Model 1 in Fig. 7 is reduced from the initial model (Fig. 6), model 2 is reduced from model 1, and so on. Thus, we can successively deal with simpler workflow structures that are suitable for throughput calculation, as described in Section V. Eventually, the fuzzy execution time of
We are currently conducting research on the following issues as future research topics:

1) fuzzy estimation of task durations for discrete-time SPNs with arbitrary time distribution [35];
2) more control structures and patterns will be taken into account to deal with workflows of any complexity;
3) in-depth comparison of the results of our methodology to other works, both analytical and simulation approaches [15], [16];
4) integration of our methodology to the workflow engine of a workflow management system.

REFERENCES
