Scientia Iranica, Vol. 8, No. 4, pp 303–311

SCIENTIA
IRANICA

Research Note

Stochastic Activity Networks: A New Definition and Some Properties

A. Movaghar[1]

Stochastic Activity Networks (SANs) are a stochastic generalization of Petri nets which have been defined for the modeling and analysis of distributed real-time systems. In this paper, a new definiton for these models is presented. This definition is based on a unified view of the system in three settings: nondeterministic, probabilistic and stochastic. Some general properties of these models are also investigated.

INTRODUCTION

Stochastic Activity Networks (SANs) are a stochastic generalization of Petri nets which have been defined for the modeling and analysis of distributed real-time systems [1,2]. These models are more powerful and flexible than most other stochastic extensions of Petri nets including some notable models such as stochastic Petri nets [3] and generalized stochastic Petri nets [4]. In this paper, a new definition for SANs is proposed. This definition is based on a unified view of the system in three settings: nondeterministic, probabilistic and stochastic. Two important aspects of concurrency are distinguished, namely, nondeterminacy and parallelism. The former is concerned with conditions where the completion of an activity may result in different system behaviors. The latter, on the other hand, refers to cases where there are several activities competing for completion and there is an uncertainty as to which one of these activities may complete first. In a nondeterministic setting, nondeterminacy and parallelism are represented in a nondeterministic manner. In a probabilistic setting, nondeterminacy is specified probabilistically but parallelism is treated nondeterministically. In a stochastic setting, both nondeterminacy and parallelism are modeled probabilistically.

Using the above framework, a new and systematic definition for SANs is given. The purpose of such a systematic definition is twofold. First, it allows for a better and more formal definition per se. Second, it allows for the use of the model for the analysis of both functional and performance aspects of the system. An important characteristic of distributed real-time systems is that their functional and performance aspects are usually intertwined and may not be separated. As a result, an appropriate class of models for such systems must include both of these aspects in a unified manner [5,6] like the systematic definition presented here for SANs.

This paper is organized as follows. In the next section, nondeterministic models are defined for concurrent systems. Then, these models are extended to some probabilistic and stochastic models, respectively. Some general properties of these models are investigated. Finally, the main results of the paper are summarized.

NONDETERMINISTIC MODELS

Activity networks [1] are nondeterministic models which have been developed for representing concurrent systems. These models are closely related to Petri nets [7] with the following extensions. The transitions in Petri nets are replaced by the primitives called "activities." There are two types of activities: "instantaneous" activities and "timed" activities. The former describes events which occur instantaneously, the latter represents processes which usually take some time to complete. Instantaneous activities model nondeterminacy while timed activities represent parallelism. A similar approach has also been used in [4] for representing nondeterminacy and parallelism. Other primitives which distinguish activity networks from Petri nets are "gates." Gates model complex interactions among activities and, thus, increase modeling flexibility. The following presents an updated

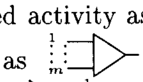
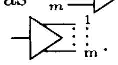
1. *Department of Computer Engineering, Sharif University of Technology, Tehran, I.R. Iran.*

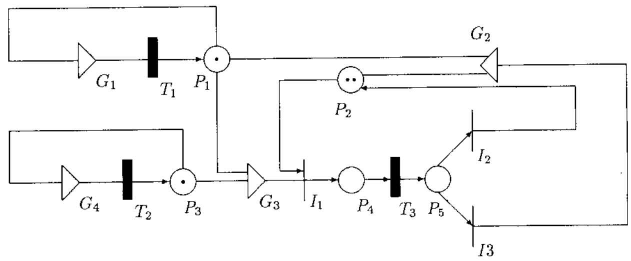
definition of activity networks which is slightly different from that appeared in [1]. The latter definition includes extra primitives, called "cases", for modeling nondeterminacy which, with the current definition, can equivalently be replaced by some instantaneous activities. Throughout this paper, $\mathcal{N}$ denotes the set of natural numbers and $\mathcal{R}_+$ represents the set of non-negative real numbers.

Definition 1

An activity network is a 7-tuple $(P, IA, TA, IG, OG, IR, OR)$ where:

- P is a finite set of *places*;
- IA is a finite set of *instantaneous activities*;
- TA is a finite set of *timed activities*;
- IG is a finite set of *input gates*. Each input gate has a finite number of *inputs*. To each $G \in IG$, with m inputs, a function $f_G : \mathcal{N}^m \longrightarrow \mathcal{N}^m$ is associated called the *function* of G, and a predicate $g_G : \mathcal{N}^m \longrightarrow \{true, false\}$, called the *enabling predicate* of G;
- OG is a finite set of *output gates*. Each output gate has a finite number of *outputs*. To each $G \in OG$, with m outputs, a function $f_G : \mathcal{N}^m \longrightarrow \mathcal{N}^m$ is associated called the *function* of G;
- $IR \subseteq P \times \{1, \ldots, |P|\} \times IG \times (IA \cup TA)$ is the *input relation*. IR satisfies the following conditions:
  - For any $(P_1, i, G, a) \in IR$ such that G has m inputs, $i \leq m$,
  - For any $G \in IG$ with m inputs and $i \in \mathcal{N}$, $i \leq m$, there exist $a \in (IA \cup TA)$ and $P_1 \in P$ such that $(P_1, i, G, a) \in IR$,
  - For any $(P_1, i, G_1, a), (P_1, j, G_2, a) \in IR$, $i = j$ and $G_1 = G_2$,
- $OR \subseteq (IA \cup TA) \times OG \times \{1, \ldots, |P|\} \times P$ is the *output relation*. OR satisfies the following conditions:
  - For any $(a, G, i, P_1) \in OR$ such that G has m outputs, $i \leq m$,
  - For any $G \in OG$ with m outputs and $i \in \mathcal{N}$, $i \leq m$, there exist $a \in (IA \cup TA)$ and $P_1 \in P$ such that $(a, G, i, P_1) \in OR$,
  - For any $(a, G_1, i, P_1), (a, G_2, j, P_1) \in OR$, $i = j$ and $G_1 = G_2$.

Graphically, an activity network is represented as follows. A place is depicted as ○, an instantaneous activity is represented as | and a timed activity as ▮. An input gate with m inputs is shown as  and an output gate with m outputs as . As an example, consider the graphical representation of an activity network shown in Figure 1. P_1, P_2, P_3, P_4 and P_5 are places; T_1, T_2, and T_3 are timed activities and I_1, I_2 and I_3 are instantaneous activities. G_1, G_3 and G_4 are input gates and G_2 is an output gate. The enabling predicates and functions of these gates are indicated in a table called "Gate Table." Note that the function of G_1 and G_4 are not shown in this table because they are identitiy functions. A directed line from a place to an activity represents a special input gate with a single input and an enabling predicate g and a function f such that $g(x) = true$, iff $x \geq 1$ and $f(x) = x - 1$ (e.g., directed lines from P_2 to I_1 and from P_5 to I_3). A directed line from an activity to a place represents a special output gate with a single output and a function f such that $f(x) = x + 1$ (e.g., the directed lines from T_1 to P_1 and from I_1 to P_4). These special gates are referred to as "standard" gates. Consider an activity network as in Definition 1. Suppose $(P_k, k, G, a) \in IR$. Then, in a graphical representation, place P_k is linked to the k-th input of an input gate G whose output is connected to activity a. P_k is said to be an *input place* of a and G is referred to as an *input gate* of a. For example, let IR be the input relation of the model of Figure 1. Then, $(P_1, 1, G_3, I_1), (P_3, 2, G_3, I_1) \in IR$, P_1 and P_3 are input places of I_1 and G_3 is an input gate of I_1. Similarly, suppose $(a, G, k, P_k) \in OR$. Then, in a graphical representation, activity a is linked to the input of an output gate G whose k-th output is connected to place P_k. G is said to be an *output gate* of a and P_k is referred to as an *output place* of a. For example, let OR be the output relation of the model of Figure 1. Then, $(I_3, G_2, 1, P_2), (I_3, G_2, 2, P_1) \in OR$,



Gate	Enabling Predicate	Function
G_1	$x < 3$	
G_2		$x_1 := x_1 + 1$; **if** $x_2 < 2$ **then** $x_2 := x_2 + 2$; **else if** $x_2 := 2$; **then** $x_2 := x_2 + 1$
G_3	$x_1 \geq 2$, $x_2 \geq 1$	$x_1 := x_1 - 2$; $x_2 := x_2 - 1$
G_4	$x < 5$	

Gate Table

Figure 1. Graphical representation of activity network with a marking.

G_2 is an output gate of I_3, and P_1 and P_2 are output places of I_3.

Like Petri nets, there is a notion of "marking" for activity networks.

Definition 2

Consider an activity network as in Definition 1. A marking is a function,

$$\mu : P \longrightarrow \mathcal{N}.$$

It is often convenient to characterize a marking μ as a vector, that is, $\mu = (\mu_1, \dots, \mu_n)$, where $\mu_i = \mu(P_i)$, $i = 1, \dots, n$ and $P = \{P_1, \dots, P_n\}$. In a graphical representation, a marking is characterized by *tokens* (dots) inside places. The number of tokens in a place represents the marking of that place (e.g., marking $(1, 2, 1, 0, 0)$ in Figure 1). An activity is "enabled" in a marking if the enabling predicates of its input gates are true in that marking. More formally, the following definition can be presented.

Definition 3

Consider an activity network as in Definition 1. $a \in (IA \cup TA)$ is *enabled* in a marking μ if for any input gate G of a with m inputs and an enabling predicate g_G,

$$g_G(\mu_1, \dots, \mu_m) = true,$$

where $\mu_k = \mu(P_k)$, for some $P_k \in P$ such that $(P_k, k, G, a) \in IR$, $k = 1, \dots, m$.

An activity is *disabled* in a marking if it is not enabled in that marking. A marking is *stable* if no instantaneous activity is enabled in that marking. A marking is *unstable* if it is not stable. For example, $(1, 2, 1, 0, 0)$ is a stable marking in Figure 1. In this marking, only T_1 and T_2 are enabled. On the other hand, $(2, 2, 1, 0, 0)$ is an unstable marking for the model of Figure 2. In this marking, T_1, T_2 and I_1 are the only activities which are enabled.

An activity network with a marking is a dynamic system. A marking changes only if an activity completes. In a stable marking, only one of the enabled timed activities is allowed to complete. When there are more than one enabled timed activity, the choice of which activity to complete first is done nondeterministically. In an unstable marking, only one of the enabled instantaneous activities may complete (i.e., enabled instantaneous activities have priority over enabled timed activities for completion). When there is more than one enabled instantaneous activity, the choice of which activity to complete first is also done nondeterministically. When an activity completes, it may change the marking of its input and output places. This change is governed by the functions of its input gates and output gates and is done in two steps as follows. First, the marking of the input places may change due to the functions of the input gates, resulting in an intermediary marking. Next, in this latter marking, the marking of the output places may also change due to the functions of the output gates, resulting in a final marking after the completion of that activity. More specifically, consider an activity network as in Definition 1. Suppose an activity a completes in a marking μ. The next marking μ' is determined in two steps as follows. First, an intermediary marking μ'' is obtained from μ by the functions of input gates of a. μ' is then determined from μ'' by the functions of output gates of a. More formally, μ'' and μ' are defined as follows:

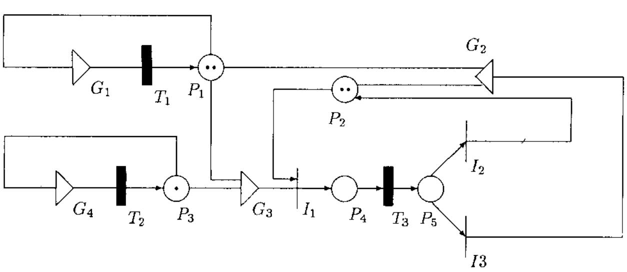


Figure 2. Marking of the model after T_1 completes in the model of Figure 1.

- For any $P_1 \in P$ which is not an input or output place of a,

$$\mu''(P_1) = \mu'(P_1) = \mu(P_1),$$

- For any input gate G of a with m inputs and a function f_G,

$$f_G(\mu_1, \dots, \mu_m) = (\mu''_1, \dots, \mu''_m),$$

where $\mu_k = \mu(P_k)$ and $\mu''_k = \mu''(P_k)$ such that $(P_k, k, G, a) \in IR$, $k = 1, \dots, m$,

- For any output gate G of a with m outputs and a function f_G,

$$f_G(\mu''_1, \dots, \mu''_m) = (\mu'_1, \dots, \mu'_m),$$

where $\mu''_k = \mu''(P_k)$ and $\mu'_k = \mu'(P_k)$ such that $(a, G, k, P_k) \in OR$, $k = 1, \dots, m$.

The above items summarize the behavior of an activity network. As an example, consider the model of Figure 1 with a marking $(1, 2, 1, 0, 0)$. In this marking, T_1 and T_2 are enabled. Any of these activities may complete. Suppose T_1 completes first, then P_1 gains a token and the marking changes to an unstable marking $(2, 2, 1, 0, 0)$, as shown in Figure 2. In this marking, I_1 is the only enabled instantaneous activity.

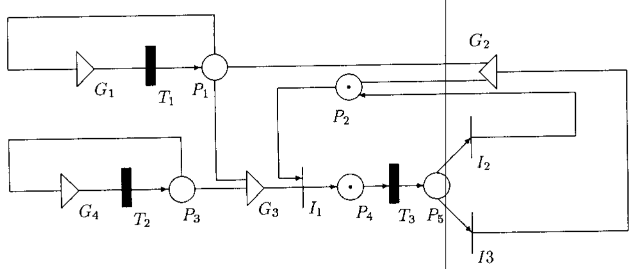


Figure 3. Marking of the model after I_1 completes in the model of Figure 2

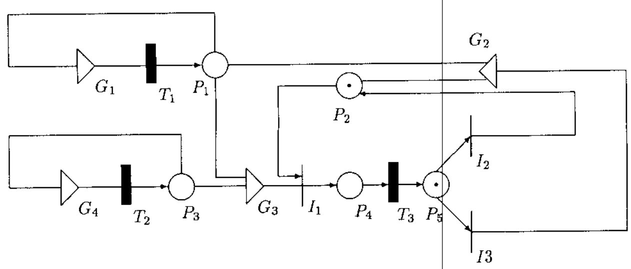


Figure 4. Marking of the model after T_3 completes in the model of Figure 3.

Accordingly, I_1 will complete next. Then, P_2 and P_3 lose a token each, P_1 loses two tokens and P_4 gains a token and the new marking becomes $(0,1,0,1,0)$, as shown in Figure 3. In this marking, T_1, T_2 and T_3 are enabled. Suppose T_3 will complete next. Then, a token is removed from P_4 and is added to P_5, and the marking changes to an unstable marking $(0,1,0,0,1)$ as shown in Figure 4. In this marking, two instantaneous activities, namely, I_2 and I_3, are enabled. Any of these activities may complete. Suppose I_3 is the one to complete first. Then, a token is removed from P_5, two tokens are added to P_1 and a token to P_2 and the new marking becomes $(2,2,0,0,0,)$ as in Figure 5.

Following the above discussion, a marking μ' is said to be *reachable* from a marking μ under a string of activities $a_1,\ldots,a_n$, if the successive completion of $a_1,\ldots,a_n$ changes the marking of the network from μ to μ'. μ' is said to be *reachable* from μ, if μ' is reachable from μ under a string of activities or $\mu' = \mu$. For example, $(2,2,0,0,0)$ is reachable from $(1,2,1,0,0)$ under $T_1I_1T_3I_3$ and $(2,2,0,0,0)$ is reachable from $(0,1,0,1,0)$ and $(2,2,0,0,0)$.

The behavior of an activity network is concerned with the manner in which various stable markings are reached from each other due to the completion of timed activities. In order to study this behavior more formally, the notion of an activity system is used [2].

Definition 4

An *activity system* is a 4-tuple $(Q, A, \rightarrow, Q_0)$ where:

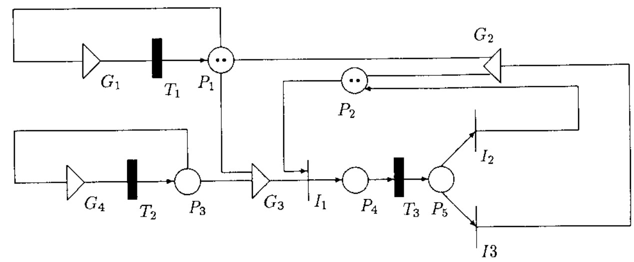


Figure 5. Marking of the model after I_3 completes in the model of Figure 4.

- Q is a set of *states*,
- A is the *activity* alphabet,
- $\rightarrow \subseteq Q \times A \times Q$ is the *transition relation*,
- Q_0 is the set of *initial states*.

A state, q', is *immediately reachable* from state q under activity a, denoted as $q \xrightarrow{a} q'$, if $(q,a,q') \in \rightarrow$.

Next, a notion of equivalence is defined for activity systems based on the concept of bisimulation [8,9]. Let B and B' be some arbitrary sets and $\gamma \subseteq B \times B'$. For $B'' \subseteq B$, let:

$$\gamma(B'') = \{b' \mid b' \in B' \text{ and, for some } b \in B'',\ (b,b') \in \gamma\}.$$

Definition 5

Let $S = (Q, A, \rightarrow, Q_0)$ and $S' = (Q', A', \rightarrow', Q'_0)$ be two activity systems with the same activity alphabet (i.e., $A = A'$). S and S' are said to be *equivalent* if there exists a symmetric binary relation γ on $Q \cup Q'$ such that:

- $Q = \gamma(Q')$ and $Q' = \gamma(Q)$,
- $Q_0 = \gamma(Q'_0)$ and $Q'_0 = \gamma(Q_0)$,
- For any $q_1, q_2 \in Q$, $q'_1 \in Q'$ and $a \in A$ such that $(q_1, q'_1) \in \gamma$ and $q_1 \xrightarrow{a} q_2$, there exists $q'_2 \in Q'$ such that $(q_2, q'_2) \in \gamma$ and $q'_1 \xrightarrow{a}{}' q'_2$; also, for any $q'_1, q'_2 \in Q'$, $q_1 \in Q$ and $a \in A$ such that $(q'_1, q_1) \in \gamma$ and $q'_1 \xrightarrow{a}{}' q'_2$, there exists $q_2 \in Q$ such that $(q'_2, q_2) \in \gamma$ and $q_1 \xrightarrow{a} q_2$.

The above mentioned γ is said to be a *bisimulation* between S and S'. S and S' are *isomorphic* if γ is a bijection.

Now, it is possible to formalize the notion of the behavior of an activity network as follows.

Definition 6

Let (K, μ_0) denote an activity network K with an initial marking μ_0 where K is defined as in Definition 1. (K, μ_0) is said to *realize* an activity system $S = (Q, A, \rightarrow, Q_0)$ where:

- Q is the set of all stable markings of K which are reachable from μ_0 and a state Δ if, in K, an infinite sequence of instantaneous activities can be completed in a marking reachable from μ_0,
- $A = TA$,
- For any $\mu, \mu' \in Q$ and $a \in A$, $\mu \xrightarrow{a} \mu'$ iff, in K, μ' is reachable from μ under a string of activities ax, where x is a (possibly an empty) string of instantaneous activities; $\mu \xrightarrow{a} \Delta$ iff, in K, a sequence of activities ay can be completed in μ, where y is an infinite sequence of instantaneous activities,
- Q_0 is the set of all stable markings of K which are reachable from μ_0 under a (possibly an empty) string of instantaneous activities and a state Δ if, in K, an infinite sequence of instantaneous activities can be completed in μ_0.

The above definition implies a notion of equivalence for activity networks as follows.

Definition 7

Two activity networks are *equivalent* if they realize equivalent activity systems.

The following concepts help specify the modeling power of activity networks.

Definition 8

An activity system is said to be *computable* if it is isomorphic to an activity system with a computable transition relation and an enumerable set of initial states.

Definition 9

An activity network is said to be *computable* if the enabling predicates and functions of all of its input gates and the functions of all of its output gates are computable.

Theorem 1

Any computable activity system is isomorphic to an activity system realized by a computable activity network with some initial marking.

Proof

Consider the class of activity networks which only have instantaneous activities, standard gates and a special type of gates called "inhibitor" gates. An *inhibitor* gate is an input gate with an enabling predicate g and an identity function such that $g(x) = true$ iff $x = 0$. The above class of activity networks corresponds to the class of extended Petri nets [7]. It has been proven that an extended Petri net can simulate a Turing machine [7,10,11]. Using similar approach, one can show that an extended Petri net can, indeed, simulate a nondeterministic Turing machine [12]. It follows that an activity network with some instantaneous activities, standard gates and inhibitor gates is, likewise, able to simulate a nondeterministic Turing machine. This latter result will be used below to prove the theorem.

Let $S = (Q, A, \rightarrow, Q_0)$ be a computable activity system. Without loss of generality, let $Q = \mathcal{N}$. Define:

- For $a \in A$, $R_a = \{(i,j); i,j \in \mathcal{N}, i \xrightarrow{a} j\}$,
- For $a \in A$, $G_a : \mathcal{N} \longrightarrow \{true, false\}$, where $G_a(i) = true$ iff there exists $j \in \mathcal{N}$ such that $i \xrightarrow{a} j$,
- $R_{Q_0} = \{(1,i); i \in Q_0\}$.

Since S is a computable activity system, $G_a, R_a, a \in A$ and R_{Q_0} are also computable, which means that they can be simulated by some nondeterministic Turing machines. Accordingly, using the result mentioned earlier, $G_a, R_a, a \in A$ and R_{Q_0} can be simulated by some activity networks $K_{G_a}, K_{R_a}, a \in A$, and K_{Q_0}, respectively, which have only some instantaneous activities, standard gates and inhibitor gates. Now, consider an activity network K with the set of timed activities A such that for any $a \in A$, K includes an activity subnetwork as depicted in Figure 6. In Figure 6, K_{G_a} and K_{R_a} represent some activity networks which simulate G_a and R_a, respectively. P_{1a}, P_{2a}, P_{3a} and P_S are places such that when P_{1a} is empty, all activities of K_{G_a} are disabled and when P_{3a} is empty, all activities of K_{R_a} are disabled. Initially, P_{1a} has a token but P_{2a} and P_{3a} are empty. As soon as P_S acquires a marking x such that $G_a(x) = true$, K_{G_a} starts execution and after a finite number of activity completions, P_{1a} loses a token, P_{2a} gains one and the marking of P_S remains the same. When P_{3a} obtains a token, K_{R_a} also begins execution and after a finite number of activity completions, P_{3a} loses a token, P_{1a} gains one and the marking of P_S changes from x to y where $(x,y) \in R_a$. K also includes an activity subnetwork as depicted in Figure 7. In Figure 7, K_{Q_0}

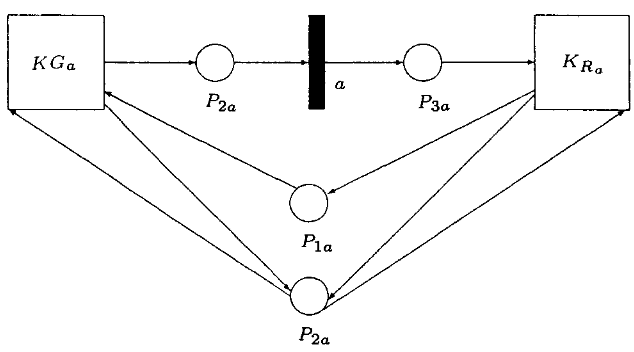


Figure 6. An activity subnetwork of K corresponding to an activity a of S.

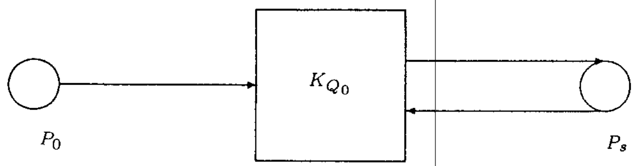


Figure 7. An activity subnetwork of K corresponding to Q_0.

represents an activity network which simulates R_{Q_0}. P_0 is a place with an initial marking 1. Initially, P_S is empty and K_{Q_0} starts execution. After a finite number of activity completions, P_0 loses a token, the marking of P_S is set to x where $(1, x) \in R_{Q_0}$, and all activities of K_{Q_0} are disabled. Suppose K includes no activity subnetwork other than those depicted in Figures 6 and 7. It can be shown that activity network K defined above, indeed, realizes an activity system which is isomorphic to S. □

The above proof indicates that as far as the behavior is concerned, any system modeled by a computable activity network may also be represented by a model which has only some standard gates and inhibitor gates. This, however, may require a large and complex model and, thus, may only be accomplished at the expense of modeling convenience, which is more formally represented in the following corollary.

Corollary 1

Any computable activity network is equivalent to an activity network which has only standard and inhibitor gates.

Note that the class of activity networks with standard gates and inhibitor gates, in fact, corresponds to the class of generalized stochastic Petri nets [4], when the latter models are viewed in a nondeterministic setting. Thus, in a nondeterministic setting, generalized stochastic Petri nets are as powerful as computable activity networks.

PROBABILISTIC MODELS

The following models are extensions of activity networks where nondeterminacy is specified probabilistically. This is accomplished by assigning probabilities to various instantaneous activities. A similar approach has also been used in [13] in probabilistic modeling of nondeterminacy.

Definition 10

A probabilistic activity network is a 8-tuple $(P, IA, TA, IG, OG, IR, OR, C)$ where:

- $(P, IA, TA, IG, OG, IR, OR)$ is an activity network,
- $C : \mathcal{N}^n \times IA \longrightarrow [0, 1]$ is the *case probability function*, where $n = |P|$.

The behavior of the above model is similar to that of an activity network, except that when there is more than one enabled instantaneous activity in an unstable marking, the choice of which activity completes first is made probabilistically. More specifically, let L be a probabilistic activity network as in this definition. Suppose L is in an unstable marking μ. Let A' be the set of enabled instantaneous activities of L in μ. Then, $a \in A'$ completes with probability α, where:

$$\alpha = \frac{C(\mu, a)}{\sum_{a' \in A'} C(\mu, a')}.$$

The above summarizes the behavior of a probabilistic activity network. In order to study this behavior more formally, it is necessary to define the notion of a probabilistic activity system.

Definition 11

A probabilistic activity system is a 4-tuple (Q, A, h, p_0) where:

- Q is a set of *states*,
- A is the *activity* alphabet,
- $h = \{h(.|q, a); q \in Q, a \in A\}$ is the set of *transition distributions* such that for any $q \in Q$ and $a \in A$, $h(.|q, a) = 0$ or $h(.|q, a)$ is a probability distribution over Q,
- p_0 is the *initial state distribution* which is a probability distribution over Q.

For $a \in A$ and $q, q' \in Q$, q' is said to be *immediately reachable* from q under a with probability α, if $h(q'|q, a) = \alpha$.

A notion of equivalence for probabilistic activity systems is now presented.

Definition 12

Let $U = (Q, A, h, p_0)$ and $U' = (Q', A', h', p'_0)$ be two probabilistic activity systems with the same activity alphabet (i.e., $A = A'$). U and U' are said to be *equivalent* if there exists a symmetric binary relation γ on $Q \cup Q'$ such that:

- $Q = \gamma(Q')$ and $Q' = \gamma(Q)$,
- For any $q_0 \in Q$ and $q'_0 \in Q'$ such that $(q_0, q'_0) \in \gamma$,

$$\sum_{q \in \gamma(\{q'_0\})} p_0(q) = \sum_{q' \in \gamma(\{q_0\})} p'_0(q'),$$

- For any $a \in A$, $q_1, q_2 \in Q$, and $q'_1, q'_2 \in Q'$ such that $(q_1, q'_1) \in \gamma$ and $(q_2, q'_2) \in \gamma$,
$$\sum_{q \in \gamma(\{q'_2\})} h(q|q_1, a) = \sum_{q' \in \gamma(\{q_2\})} h'(q'|q'_1, a).$$

The above mentioned γ is said to be a *bisimulation* between U and U'. U and U' are *isomorphic* if γ is a bijection.

The behavior of a probabilistic activity network may now be formalized as follows.

Definition 13

Let (L, μ_0) denote a probabilistic activity network L with an initial marking μ_0 where L is defined as in Definition 10. (L, μ_0) is said to *realize* a probabilistic activity system $U = (Q, A, h, p_0)$ where:

- Q is the set of all stable markings of L which are reachable from μ_0 and a state Δ if, in L, an infinite sequence of instantaneous activities can be completed in a marking reachable from μ_0,
- $A = TA$,
- For any $\mu, \mu' \in Q$ and $a \in A$ such that a is disabled in μ, $h(.|\mu, a) = 0$,
- For any $\mu, \mu' \in Q$ and $a \in A$ such that a is enabled in μ, $h(\mu'|\mu, a)$ is the probability that, in L, μ' is the next stable marking to be reached upon completion of a in μ; $h(\Delta|\mu, a)$ is the probability that, in L, a sequence of activities ax completes in μ, where x is an infinite sequence of instantaneous activities,
- For any $\mu \in Q$, $p_0(\mu)$ is the probability that, in L, μ is reached upon completion of a (possibly an empty) string of instantaneous activities in μ_0; $p_0(\Delta)$ is the probability that, in L, an infinite sequence of instantaneous activities completes in μ_0.

A notion of equivalence for probabilistic activity networks may now be provided as follows.

Definition 14

Two probabilistic activity networks are *equivalent* if they realize equivalent probabilistic activity systems.

STOCHASTIC MODELS

In the previous section, nondeterminacy has been treated in a probabilistic manner. Now models which represent both nondeterminacy and parallelism probabilistically are presented. This is accomplished by assigning certain parameters to timed activities and viewing the model in a stochastic setting. The following definition is slightly different from those which appeared in [1,2]. It includes some additional extensions which allow activities to be processed at various speeds [14].

Definition 15

A *stochastic activity network* is an 11-tuple $(P, IA, TA, IG, IG, IR, OR, C, F, \Pi, \rho)$ where:

- $(P, IA, TA, IG, OG, IR, OR, C)$ is a probabilistic activity network,
- $F = \{F(.|\mu, a); \mu \in \mathcal{N}^n, a \in TA\}$ is the set of *activity time distribution functions*, where $n = |P|$ and, for any $\mu \in \mathcal{N}^n$ and $a \in TA$, $F(.|\mu, a)$ is a probability distribution function,
- $\Pi : \mathcal{N}^n \times TA \longrightarrow \{true, false\}$ is the *reactivation predicate*, where n is defined as before,
- $\rho : \mathcal{N}^n \times TA \longrightarrow \mathcal{R}_+$ is the *enabling rate function*, where n is defined as before.

The behavior of the above model is similar to that of a probabilistic activity network except that here the notion of timing is explicitly considered. When instantaneous activities are enabled they complete instantaneously. Enabled timed activities, on the other hand, require some time to complete. A timed activity becomes *active* as soon as it is enabled and remains so until it completes; otherwise, it is *inactive*. Consider a stochastic activity network M as in Definition 15. Suppose, at time t, a timed activity completes and μ is the stable marking of M immediately after t. A timed activity a is *activated* at t, if a is enabled in μ and one of the following occurs:

- a becomes inactive immediately before t;
- a completes at t;
- $\Pi(\mu, a) = true$.

Whenever the above happens, a is assigned to an *activity time* τ, where τ is a random variable with probability distribution function $F(.|\mu, a)$. When a timed activity a is enabled in a stable marking μ, it is *processed* with a rate $\rho(\mu, a)$. A timed activity *completes* whenever it is processed for its activity time. Upon completion of an activity, the next marking occurs immediately.

The above summarizes the behavior of a stochastic activity network. This behavior may be studied more formally using the following concepts.

Definition 16

Let M be a stochastic activity network. The *state process* of M is a random process $\{X(t); t \in \mathcal{R}_+\}$ where $X(t)$ denotes the stable marking of M at time t.

Definition 17

Let $X = \{X(t); t \in \mathcal{R}_+\}$ and $X' = \{X'(t); t \in \mathcal{R}_+\}$ be two random processes with the set of states Q and

Q', respectively. X and X' are said to be *stochastically equivalent* if there exists a symmetric binary relation γ on $Q \cup Q'$ such that:

- $\gamma(Q) = Q'$ and $\gamma(Q') = Q$;
- For any $t_i \in [0, \infty)$, $Q_i \subseteq Q$ and $Q'_i \subseteq Q'$, such that $Q_i = \gamma(Q'_i)$ and $Q'_i = \gamma(Q_i)$, $i = 0, \ldots, n$, $n \in \mathcal{N}$;

$$p[X(t_i) \in Q_i; i = 0, \ldots, n]$$
$$= p[X'(t_i) \in Q'_i; i = 0, \ldots, n].$$

X and X' are *stochastically isomorphic (equal)* if γ is a bijection (an equality).

Proposition 1

Let $M = (L, F, \Pi, \rho)$ and $M' = (L', F', \Pi', \rho')$ be two stochastic activity networks where L and L' are some equivalent probabilistic activity networks. Suppose L and L' realize probabilistic activity systems $U = (Q, A, h, p_0)$ and $U' = (Q', A', h', p'_0)$, respectively ($A = A'$). The state processes of M and M' will be stochastically equivalent if there exists a symmetric binary relation γ on $Q \cup Q'$ such that:

- γ is a bisimulation between U and U',
- For any $a \in A$, $q \in Q$ and $q' \in Q'$ such that $(q, q') \in \gamma$ and a is enabled in both q and q', $F(.|q, a) = F'(.|q', a)$, $G(q, a) = G(q', a)$ and $\rho(q, a) = \rho'(q', a)$.

The state behavior of a stochastic activity network is closely related to the notion of a generalized semi-Markov process as defined in [14,15].

Proposition 2

The following statements are true:

i) Any generalized semi-Markov process with a finite set of events is stochastically isomorphic to the state process of a stochastic activity network,

ii) There exists a stochastic activity network whose state process is not a generalized semi-Markov process,

iii) The state process of any stochastic activity network with state-independent activity time distribution functions and a false reactivation predicate is a generalized semi-Markov process.

Now Markovian models are considered:

Theorem 2

Let M be a stochastic activity network as in Definition 15. The state process of M is a Markov process iff for any timed activity a which is enabled in a stable marking μ and any stable marking μ_{ac} in which a is last activated prior to being enabled in μ,

$$F(\tau|\mu_{ac}, a) = 1 - e^{-\alpha(\mu,a)\,\tau},$$

where $\alpha(\mu, a)$ is a positive real number which only depends on μ and a.

Proof

Let $M = (L, F, \Pi, \rho)$ with a corresponding probabilistic activity network L which realizes a probabilistic activity system $U = (Q, A, h, p_0)$. Denote $X = \{X(t); t \in \mathcal{R}_+\}$ as the state process of M.

If: It is noted that for any $t, \delta t \in \mathcal{R}_+$, where δt is sufficiently small and any stable markings $\mu, \mu' \in Q$,

$$P[X(t+\delta t) = \mu' | X(t) = \mu, X(t'), 0 \le t' < t]$$
$$= P[X(t+\delta t) = \mu' | X(t) = \mu]$$
$$\approx \sum_{a \in A} \alpha(\mu, a)\, \rho(\mu, a)\, h(\mu, \mu')\, \delta t.$$

Using the memoryless property of exponentially distributed random variables and the dynamic behavior of the model, it can be concluded that X is a Markov process.

Only if: Note that exponentially distributed random variables are the only random variables with memoryless property and that X is assumed to be a Markov process. The proof then follows from the definition of the dynamic behavior of the model. □

A stochastic activity network is said to be *Markovian* if its state process is a Markov process.

Corollary 2

Let M be a stochastic activity network with a set of exponential activity time distribution functions such that any activity with a state-dependent activity time distribution function has also a true reactivation predicate. Then, M is Markovian.

Corollary 3

Any discrete-space, continuous-time and time-homogeneous Markov process is stochastically isomorphic to the state process of a Markovian stochastic activity network.

CONCLUSION

In this paper, a new definition for SANs is presented. This definition is based on a unified view of the system in three settings: nondeterministic, probabilistic and stochastic. The purpose of such a systematic definition is twofold. First, it allows for a better and more formal definition per se. Second, it allows for the use of the model for the analysis of both functional and performance aspects of the system. Since the functional and performance properties of distributed real-time systems are usually intertwined, these models are appropriate for representing such complex systems. Some general properties of these models were also investigated.

REFERENCES

1. Movaghar, A. and Meyer, J.F. "Performability modeling with stochastic activity networks", *IEEE Proc. Real-Time Sys. Symp.*, Austin, TX, USA, pp 215-224 (1984).
2. Meyer, J.F., Movaghar, A. and Sanders, W.H. "Stochastic activity networks: Structure, behavior and application", *Proc. Int. Workshop on Timed Petri Nets*, Torino, Italy, pp 106-115 (1985).
3. Molloy, M.K. "Performance analysis using stochastic Petri nets", *IEEE Trans. Comput.*, **C-31**, pp 913-917 (1982).
4. Marsan, M.A., Balbo, G. and Conte, G. "A class of generalized stochastic Petri nets for performance evaluation of multiprocessors systems", *ACM Trans. Comp. Sys.*, **2**(2), pp 93-122 (1984).
5. Hillston, J. "PEPA: Performance Enhanced Process Algebra", Ph.D. Dissertation, The University of Edinburgh, UK (1993).
6. Götz, N., Herzog, U. and Rettelbach, M. "Multiprocessor and distributed system design: The integration of functional and performance analysis using stochastic process algebra", *Proc. of Performance'93*, LNCS 729 (1993).
7. Peterson, J.L., *Petri Nets Theory and the Modeling of Systems*, Prentice-Hall, Inc., Englewood Cliffs (1981).
8. Milner, R., *Communication and Concurrency*, Prentice-Hall International, UK (1989).
9. Park, D.M.R. "Concurrency and automata on infinite sequences", *Lecture Notes in Computer Science*, **104**, Springer-Verlag, New York (1981).
10. Agerwala, T. "A complete model for representing the coordination of asynchronous processes", Research Report No. 32, Computer Science Program, Johns Hopkins University, Baltimore, MD (1974).
11. Hack, M. "Decidability questions for Petri nets", Ph.D. Thesis, MIT, Cambridge, MA (1975).
12. Sipser, M. *Introduction to the Theory of Computation*, PWS Publishing Company, Boston, MA (1997).
13. Marsan, M.A., Bablo, G., Chiola, G. and Conte, G. "Generalized stochastic Petri nets revisited: Random switches and properties", *Proc. Int. Workshop on Timed Petri Nets*, Madison, WI, pp 44-62 (1987).
14. Schassberger, R. "Insensitivity of steady-state distributions of generalized semi-Markov processes with speeds", *Adv. Appl. Prob.*, **10**, pp 836-851 (1978).
15. Matthes, K. "Zur theorie der bedienungsprozesse", *Trans. 3rd Prague Conf. on Inf. Thy. Stat. Dec. Fct.*, Prague (1962).