Stochastic Activity Networks Templates

Leonardo Montecchi                  Paolo Lollini
leonardo@ic.unicamp.br             lollini@unifi.it

Andrea Bondavalli
bondavalli@unifi.it

Resilient Computing Lab
University of Firenze
Technical Report RCL180401
v1.0
(29 April 2018)

Abstract

This technical report defines Stochastic Activity Networks Templates (SAN-T), a formalism based on Stochastic Activity Networks (SAN), with the addition of variability aspects. Its purpose is to define library of model templates, as in [1]. In Section 1 we recall the definition of Stochastic Activity Networks (SAN), while in Section 2 we introduce SAN Templates (SAN-T). The \texttt{concretize()} function to generate a SAN concrete model from a SAN-T is defined in Section 3. Finally, the notion of compatibility between a template specification, and its implementation with SAN-T is defined in Section 4.

1 Stochastic Activity Networks

A formal definition of Stochastic Activity Networks (SANs) was given by Sanders and Meyer in [2]. We recall here the basic definitions provided in that paper.

An Activity Network (AN) is defined as an eight-tuple [2]

\begin{equation}
AN = (P, A, I, O, \gamma, \tau, \iota, o),
\end{equation}

where: $P$ is a finite set of places; $A$ is a finite set of activities; $I$ is a finite set of input gates; $O$ is a finite set of output gates; $\gamma: A \rightarrow \mathbb{N}^+$ specifies the number of cases for each activity; $\tau: A \rightarrow \{\text{timed, instantaneous}\}$ specifies the type of each activity; $\iota: I \rightarrow A$ maps input gates to activities; $o: O \rightarrow \{(a, c) \mid a \in A \text{ and } c \in \{1, 2, \ldots, \gamma(a)\}\}$ maps output gates to cases of activities.

We also recall that, if $S$ is a set of places ($S \subseteq P$), a \textit{marking} of $S$ is a mapping $\mu: S \rightarrow \mathbb{N}$. The value $\mu(p)$ is the marking of place $p$, i.e., the number
of tokens it holds. Similarly, the set of possible markings of $S$ is the set of functions $M_S = \{ \mu : S \rightarrow \mathbb{N} \}$.

An input gate is defined to be a triple, $(G, e, f)$, where $G \subseteq P$ is the set of input places associated with the gate, $e : M_G \rightarrow \{0, 1\}$ is the enabling predicate of the gate, and $f : M_G \rightarrow M_G$ is the input function of the gate. An output gate is a pair, $(G, f)$, where $G \subseteq P$ is the set of output places associated with the gate, and $f : M_G \rightarrow M_G$ is the output function of the gate.

Given an activity network that is stabilizing in some specified initial marking $\mu_0 \in M_P$, a Stochastic Activity Network (SAN) is formed by adjoining functions assignments $C, F, G$, where for each activity $a$, $C_a \in C$ is a function specifying the probability distribution of its cases; $F_a \in F$ is a function specifying the probability distribution of its firing delay time; and $G_a \in G$ is a function that describes its reactivation markings [2].

Formally:

$$SAN = (AN, \mu_0, C, F, G) = ((P, A, I, O, \gamma, \tau, \iota, o), \mu_0, C, F, G)_2.$$ (2)

2 SAN Templates

We base on the previous definition to define Stochastic Activity Networks Templates (SAN-T), i.e., SANs in which some elements are not fully defined, but instead they depend on some parameters.

The purpose of defining SAN-Ts is to be able to specify parametric models based on the SAN formalism, to be employed as template-level formalism to define implementations of atomic templates. From a SAN-T model different variants of a base SAN skeleton can be generated, based on the values assigned to the parameters in $\Delta$. Place templates are mapped to a finite number of “normal” places, based on their multiplicity functions; similarly, activity templates are mapped to ordinary activities with a finite number of cases.

A Stochastic Activity Network Template (SAN-T) is a tuple

$$SAN-T = (\Delta, \tilde{P}, \tilde{A}, \tilde{I}, \tilde{O}, \tilde{\gamma}, \tilde{\tau}, \tilde{\iota}, \tilde{o}, \tilde{\mu}_0, \tilde{C}, \tilde{F}, \tilde{G}),$$ (3)

where $\Delta$ is a set of parameters, and elements marked with a tilde accent, $\tilde{\cdot}$, are modified versions of elements existing in plain SANs, modified to take parameters into account. In more details:

- $\Delta$ is a sorted set of parameters of the template.
- $\tilde{P}$ is a finite set of place templates. A place template can be seen as a placeholder for multiple places that, in a regular SAN model, would be strongly related to each other. It is the case, for example, of places connected to different cases of the same activity. Based on parameters’ values, a template place will be expanded to a precise set of places.

Formally, a place template is defined by a pair $(\tau, k)$, where $\tau$ is the name of the place, and $k \in \text{TERM}(O \cup \Delta)_{\text{set}\{\text{Int}\}}$ is its multiplicity. Evaluating the
term $k$ with respect to an assignment $\xi$ identifies a set of integer indices $K \subset \mathbb{N}$. Such indices identify the set of places that, with the given setting of parameters, the template place is expanded to. Normal places (i.e., those always expanding to a single place of ordinary SANs) are those for which $Val_\xi(k) = \{1_{\text{Int}}\}$ for any assignment of parameters $\xi$.

- $\tilde{A}$ is a finite set of activity templates.
- $\tilde{I}$ is a finite set of input gate templates.
- $\tilde{O}$ is a finite set of output gate templates.
- $\tilde{\gamma}: \tilde{A} \to \text{TERM}(O \cup \Delta)_{\text{Int}}$ specifies the number of cases for each activity template. For each activity template $\tilde{a} \in \tilde{A}$, evaluating $\tilde{\gamma}(\tilde{a})$ with respect to an assignment $\xi$ returns an integer number. If $Val_\xi(\tilde{\gamma}(\tilde{a})) = m_{\text{Int}}$ for any assignment $\xi$, then the activity is a regular activity having a fixed number of cases $m$.
- $\tilde{\tau}: \tilde{A} \to \{\text{timed}, \text{instantaneous}\}$ specifies the type of each activity template.
- $\tilde{i}: \tilde{I} \to \tilde{A}$ maps input gate templates to activity templates.
- $\tilde{o}: \tilde{O} \to \tilde{A}$ maps output gate templates to activity templates.

In order to correctly define input and output gate templates, the concept of marking needs to be extended, making it applicable to place templates. The idea is to let the marking function anticipate that the place template will be mapped to a set of places, and thus allow the marking for each of them to be specified, through an index value. Formally, if $\tilde{S} \subseteq \tilde{P}$ is a set of place templates, a marking of $\tilde{S}$ is a mapping $\tilde{\mu}: \tilde{S} \times \mathbb{N} \to \mathbb{N}$. For example, $\tilde{\mu}(\tilde{p}, 2) = 10$, with $\tilde{p} \in \tilde{S}$, means that the place generated from $\tilde{p}$ having index 2 contains 10 tokens. Similarly, the set of possible markings of $\tilde{S}$ is the set of functions $\tilde{M}_\tilde{S} = \{\tilde{\mu} | \tilde{\mu}: \tilde{S} \times \mathbb{N} \to \mathbb{N}\}$.

An input gate template is also defined as a triple $(\tilde{G}, \tilde{e}, \tilde{f})$, where $\tilde{G} \subseteq \tilde{P}$ is the set of input places associated with the gate, $\tilde{e}: \tilde{M}_{\tilde{G}} \to \text{TERM}(O \cup \Delta)_{\text{Bool}}$ is the enabling predicate of the gate, and $\tilde{f}: \tilde{M}_{\tilde{G}} \times \Xi \to \tilde{M}_{\tilde{G}}$ is the input function of the gate, where $\Xi$ is the set of all possible assignments. An input gate template will always result in a single input gate in the concrete SAN model. The enabling predicate and the input function of the gate are instead parametric, i.e., in they depend on all parameters of the template.

An output gate template is a pair $(\tilde{G}, \tilde{f})$, where $\tilde{G} \subseteq \tilde{P}$ is the set of output places associated with the gate, and $\tilde{f}: \tilde{M}_{\tilde{G}} \times \mathbb{N} \times \Xi \to \tilde{M}_{\tilde{G}}$ is the output function of the gate. It should be noted that the output function $\tilde{f}$ depends on the index of the case of the associated activity template ($\mathbb{N}$), as well as from the assignment of values to parameters of the SAN-T model ($\Xi$). In fact, an output gate template will be expanded to multiple concrete output gates, depending on the number of cases of the activity to which it is connected.
As in [2], the input places of an activity template \( a \) consist of the set \( IP(a) = \{ p \mid \exists (\tilde{G}, \tilde{e}, \tilde{f}) \in \tilde{\iota}^{-1}(a) \text{ such that } p \in \tilde{G} \} \).

Similarly, the output places of an activity template \( a \) consist of the set \( OP(a) = \{ p \mid \exists (\tilde{G}, \tilde{f}) \in \tilde{\sigma}^{-1}(a) \text{ such that } p \in \tilde{G} \} \).

Finally:

- \( \tilde{\mu}_0 : \Xi \to \tilde{M}_{\tilde{P}} \) is the initial marking function, which defines the initial marking based on parameters' assignment.

- \( \tilde{C} \) is the case distribution assignment, an assignment of functions to activities templates such that for any activity template \( \tilde{a} \in \tilde{A} \), function \( \tilde{C}_{\tilde{a}} \) defines the probability distribution of activity cases. Note that this also depends on parameters; thus \( \tilde{C}_{\tilde{a}} : M_{\tilde{IP}(\tilde{a}) \cup \tilde{OP}(\tilde{a})} \times \mathbb{N}^+ \times \Xi \to [0, 1] \). For the model to be well-formed, \( \tilde{C}_{\tilde{a}}(\mu, i, \xi) = 0 \) should hold for all \( i > Val(\tilde{\gamma}(\tilde{a})) \), i.e., the probability of cases that are not foreseen by the given parameters assignment \( \xi \) should be zero.

- \( \tilde{F} \) is the activity time distribution function assignment, an assignment of continuous functions to timed template activities such that for any timed activity template \( a \), function \( \tilde{F}_a : \tilde{M}_{\tilde{P}} \times \mathbb{R} \times \Xi \to [0, 1] \) defines its firing time distribution.

- \( \tilde{G} \) is the reactivation function assignment, an assignment of functions to timed activities such that for any timed activity \( a \), function \( \tilde{G}_a : \tilde{M}_{\tilde{P}} \to 2^{\tilde{M}_{\tilde{P}}} \), defines the reactivation markings, where \( 2^{\tilde{M}_{\tilde{P}}} \) denotes the power set of \( \tilde{M}_{\tilde{P}} \).

Note that the definition of SANs [2] requires that the initial marking \( \mu_0(\xi) \in \tilde{M}_{\tilde{P}} \) is a stable marking in which the network is stabilizing. However, since the actual structure of a SAN is not completely specified until a value is assigned to all the SAN-T parameters, we relax this constraint here. When parameter values are assigned to a SAN-T, well-formedness checks on the structure of the resulting model could be performed based on techniques available for ordinary SAN models (e.g., [3]).

### 3 Mapping SAN-T to SAN

In this section we define the \texttt{concretize()} function that generates an ordinary SAN model from a from a pair \((SAN-T, \xi)\), that is, from a SAN-T model and an assignment of values to its parameters. We first need to introduce some preliminary definitions.

#### 3.1 Preliminary Definitions

Given a place template of a SAN-T model, \( \tilde{p} = (\tau, k) \in \tilde{P} \), and an assignment of parameters \( \xi \), we denote with \( \Pi(\tilde{p}, i) = p_i \in P^\xi \) the \( i \)-th place originating from the template place \( \tilde{p} \) in the SAN-T instance. Given a marking of a SAN-T,
and an assignment function $\xi$, we denote the corresponding marking of the SAN-T instance, $\mu \in M_P$, with $\Gamma(\tilde{\mu})$. It is obtained as:

$$\mu(\Pi(\tilde{\mu}, i)) = \tilde{\mu}(\tilde{\mu}), \quad \forall \tilde{\mu} \in \tilde{P}. \quad (4)$$

Consequently, given a marking $\mu'$ of the SAN instance we denote as $\Gamma^{-1}(\mu')$ the corresponding marking of the SAN-T.

Given an input gate template $\tilde{g} = (\tilde{G}, \tilde{e}, \tilde{f}) \in \tilde{I}$, and an assignment $\xi$, we denote with $\alpha(\tilde{g})$ the corresponding input gate $g = (G, e, f) \in I$ in the SAN-T instance. It is obtained as:

$$G = \{ p_i = \Pi(\tilde{\mu}, j) \mid j \in \mathit{val}_\xi(k), \; \tilde{\mu} = (\tau, k) \in \tilde{G} \},$$

$$e(\Gamma(\mu)) = \mathit{val}_\xi(\tilde{e}(\tilde{\mu})), \quad (5)$$

$$f(\Gamma(\mu)) = \tilde{f}(\tilde{\mu}, \xi).$$

Finally, given an output gate template $\tilde{g} = (\tilde{G}, \tilde{f}) \in \tilde{O}$, and an assignment function $\xi$, we denote with $\beta(\tilde{g}, i)$ the $i$-th output gate $g_i = (G_i, f_i) \in O$ generated from it in the SAN model. It is obtained as:

$$G_i = \{ p_i = \Pi(\tilde{\mu}, j) \mid j \in \mathit{val}_\xi(k), \; \tilde{\mu} = (\tau, k) \in \tilde{G} \},$$

$$f_i(\Gamma(\mu)) = \tilde{f}(\tilde{\mu}, \xi). \quad (6)$$

### 3.2 concretize()

Given a SAN-T $S_{\Delta}$:

$$S_{\Delta} = (\Delta, \tilde{P}, \tilde{A}, \tilde{I}, \tilde{O}, \tilde{\gamma}, \tilde{i}, \tilde{o}, \tilde{\mu}_0, \tilde{C}, \tilde{F}, \tilde{G}), \quad (7)$$

and a parameter assignment function $\xi$, the corresponding SAN model $S^{\xi}$ is obtained as follows:

$$S^{\xi} = (P^{\xi}, A^{\xi}, I^{\xi}, O^{\xi}, \gamma^{\xi}, \tau^{\xi}, i^{\xi}, o^{\xi}, \mu_0^{\xi}, C^{\xi}, F^{\xi}, G^{\xi}), \quad (8)$$

where:

$$P^{\xi} = \bigcup_{(\tau, k) \in \tilde{P}} \{ \tau_i \mid i \in \mathit{val}_\xi(k) \};$$

$$A^{\xi} = \tilde{A};$$

$$\gamma^{\xi}(a) = \mathit{val}_\xi(\tilde{\gamma}(\tilde{a}));$$

$$I^{\xi} = \{ \alpha(\tilde{g}) \mid \tilde{g} \in \tilde{I} \};$$

$$O^{\xi} = \bigcup_{\tilde{g} \in \tilde{O}} \{ \beta(\tilde{g}, 1), \ldots, \beta(\tilde{g}, \mathit{val}_\xi(\tilde{\gamma}(\tilde{a}))) \mid \tilde{a} = \tilde{o}(\tilde{g}) \};$$

$$\tau^{\xi} = \tilde{\tau};$$

$$i^{\xi}(\alpha(g)) = i(g), \quad \forall g \in \tilde{I};$$

$$o^{\xi}(\beta(g, i)) = \tilde{o}(g), \quad \forall g \in \tilde{O}, \forall i \in \{1, \ldots, \mathit{val}_\xi(\tilde{\gamma}(\tilde{a})) \};$$

$$\mu_0^{\xi} = \tilde{\mu}_0(\xi).$$
Furthermore:

- $C^\xi$: for each function $\tilde{C}_a \in \tilde{C}$ in the case distribution assignment $\tilde{C}$ a corresponding function $C^\xi_a$ is included in $C^\xi$, defined as $C^\xi_a(\Gamma(\mu), k) = \tilde{C}_a(\mu, k, \xi)$, $\forall \mu \in M_\rho$, $\forall k \in \mathbb{N}^+$.  

- $F^\xi$: for each function $\tilde{F}_a$ in the activity time distribution assignment $\tilde{F}$ a corresponding function $F^\xi_a$ is included in $F^\xi$, defined as $F^\xi_a(\Gamma(\mu), r) = \tilde{F}_a(\mu, r, \xi)$, $\forall \mu \in M_\rho$, $\forall r \in \mathbb{R}$.  

- $G^\xi$: for each function $\tilde{G}_a \in \tilde{G}$ in the reactivation function assignment $\tilde{G}$ a corresponding function $G^\xi_a$ is added to $G^\xi$, defined as $G^\xi_a(\Gamma(\mu)) = \{ \Gamma(\tilde{\mu}) | \tilde{\mu} \in \tilde{G}_a(\tilde{\mu}) \}$ $\forall \mu \in M_\tilde{S}$.

4 Compatibility Notion for SAN-T Models

In this section we describe the constraints that need to be enforced when applying the TMDL framework using SAN-T as the template-level formalism. In particular, we define the notion of compatibility between an atomic model template specification, and the corresponding implementation with SAN-T.

Given an atomic template specification:

$$MT = (I, \Delta, O, L_T, \Psi = \mathcal{M}),$$

and its implementation as a SAN-T model:

$$\mathcal{M} = (\Delta_M, \tilde{P}, \tilde{A}, \tilde{I}, \tilde{O}, \tilde{\gamma}, \tilde{\tau}, \tilde{\iota}, \tilde{\sigma}, \tilde{\pi}, \tilde{C}, \tilde{F}, \tilde{G}),$$

the implementation $\mathcal{M}$ is compatible with the specification $(I, \Delta, O, L_T)$ if and only if the following conditions hold:

1. $\Delta = \Delta_M$.

2. $\forall (v, \Delta^v, L, k) \in I$, $\exists (\tau, k_\tau) \in \tilde{P}$ such that $Val_\xi(k) = Val_\xi(k_\tau)$ for any assignment $\xi$.

3. $\forall (v, \Delta^v, L, k) \in O$, $\exists (\tau, k_\tau) \in \tilde{P}$ such that $Val_\xi(k) = Val_\xi(k_\tau)$ for any assignment $\xi$.

Condition 1 states that the specification and the implementation must have the same set of parameters. Condition 2 states that for each meta-variable in the interfaces realized by the template specification, a template place having the same multiplicity for must exist in the SAN-T implementation. Precisely, the multiplicities of the interface variable and of the template place must be the same for any possible assignment of values to model parameters. Condition 3 states the same for observation points.

The original SAN definitions in [2], as well as SAN-T defined in Section 2 only consider integer-valued places. In case extended places of the Möbius implementation are used [4], conditions 2 e 3 must be modified to state that the sort the meta-variable and the template places must be of the same sort:
2a. $\forall (v_s, \Delta^v, L, k) \in \mathcal{I}, \exists (\tau_{s'}, k_{\tau}) \in \hat{\mathcal{P}}$ such that $s = s'$ and $Val_\xi(k) = Val_\xi(k_{\tau})$ for any assignment $\xi$.

3a. $\forall (v_s, \Delta^v, L, k) \in \mathcal{O}, \exists (\tau_{s'}, k_{\tau}) \in \hat{\mathcal{P}}$ such that $s = s'$ and $Val_\xi(k) = Val_\xi(k_{\tau})$ for any assignment $\xi$.

References


