

Covering Places and Transitions in Open Nets

Christian Stahl¹ and Karsten Wolf²

¹ Humboldt-Universität zu Berlin, Institut für Informatik
Unter den Linden 6, 10099 Berlin, Germany
`stahl@informatik.hu-berlin.de`

² Universität Rostock, Institut für Informatik
18051 Rostock, Germany
`karsten.wolf@uni-rostock.de`

Abstract. We present a finite representation of all services M where the composition with a given service N is deadlock-free, and a given set of activities of N can be *covered* (i.e. is not dead). Our representation is an extension of the existing notion of an operating guideline which only cared about deadlock freedom. We further present an algorithm to decide whether a service M matches with the extended operating guideline of N .

Key words: process modeling and analysis, SOA, Petri nets, operating guidelines

1 Introduction

One of the objectives of service-oriented computing (SOC) [1] is the modular structuring and loose coupling of interorganisational business processes. In this aspect, SOC meets the area of modeling and analysing workflows [2]. While SOC aims at composing complex business activities from more elementary ones (services), workflow modeling is (among others) concerned with the study of well-designed workflows and business processes. Central to the wellformedness of workflows is the concept of *soundness*. This property basically states that every process instance will terminate in a well-defined final state while there are no useless (dead) activities. In the intersection of SOC and workflow modeling, we are thus interested in mechanisms for service composition (and related tasks such as discovery) which assure soundness in the overall system (e.g. a service orchestration).

Current approaches for matching and discovering services are incapable of asserting soundness in service discovery scenarios. Some approaches propose to compute and publish a public view P' of a provided service P [3, 4]. Then, a service requester R can check its composition $R \oplus P'$ to decide proper interaction. However, public view approaches do not explicitly state whether soundness of $P' \oplus R$ implies soundness of $P \oplus R$. Thus, existing public view approaches cannot be applied to obtain a globally sound system.

Other approaches suggest to compute an *operating guideline* OG_P for a given service P which represents all correctly interacting partners of P [5]. Then, a matching procedure between R and OG_P can be used for deciding whether $P \oplus R$ would interact correctly. Here, correctness refers to deadlock freedom so far.

Deadlock freedom is a necessary but insufficient condition for soundness. In this paper, we extend the operating guideline approach by asserting—in addition to deadlock freedom—the absence of dead activities in the composed system. This is another necessary condition for soundness. The only remaining gap between the new approach and soundness is the possible existence of livelocks. For acyclic services, our approach already establishes soundness in the composed system since acyclic services cannot contain livelocks.

Another motivating scenario for our approach is inspired by [6]. In this article, all partners of a given service, which *enforce* or *exclude* certain behavioral patterns such as occurrences of activities, are characterized. This approach can be used, among others, for

- filtering of service registries for services that fit specific specifications (“enforce book”: I want to get a book selling service; “exclude credit card”: I do not want to pay by credit card),
- validating services by checking whether there exist partners that access certain features

Sometimes, enforcing some behavior is too strict. Consider an application for a credit with an online bank service. Of course, the user (service requester) wishes to have the activity “credit approved” executed in the service. However, there is hardly an online bank service where “credit approved” can be enforced by the user (which would mean that the user can always obtain a credit by just following a suitable communication pattern). There will rather be an internal decision based on which a credit is either approved or denied. In typical service models, the decision appears to the user as a nondeterministic choice. Thus, we need a weaker criterion that rules out at least all those services where “credit approved” is completely impossible. That is, R should match with P if and only if it is at least *possible* to execute activity “credit approved” in the composition of the online bank service and the requester.

Formally, we want to compute a finite representation of the (generally infinite) set of all those partners R of a given service P where the composition $P \oplus R$ of both services is deadlock-free, and a certain set X of activities is not dead. For establishing soundness, this set X would be the set of all activities of P . In the online banking example, X would consist only of activity “credit approved”. We achieve this goal by extending the existing operating guideline approach with deadlock-free interaction.

The paper is structured as follows. In Sect. 2 we recall open nets and operating guidelines. Next, in Sect. 3, we extend our notion of partners R for P to those partners R' where a certain set of places and transitions in $P \oplus R'$ is covered (i.e. each place can be marked and each transition is not dead). We show how to calculate a finite representation of all these partners by extending our notion of

an operating guideline with a global constraint. Section 4 presents related work and finally conclusions are drawn in Sect. 5.

2 Preliminaries

2.1 Open Nets

We assume the usual definition of a (place/transition) Petri net $N = (P, T, F)$ (see [7], for instance) and use standard notation to denote the preset and postset of a place or a transition: $\bullet x = \{y \mid (y, x) \in F\}$ and $x^\bullet = \{y \mid (x, y) \in F\}$.

Definition 1 (Open net). *An open net $N = (P, T, F, I, O, m_0, \Omega)$ consists of a Petri net (P, T, F) together with*

- an interface defined as a set $I \subseteq P$ of input places such that $\bullet p = \emptyset$ for any $p \in I$ and a set $O \subseteq P$ of output places such that $p^\bullet = \emptyset$ for any $p \in O$ and $I \cap O = \emptyset$,
- a distinguished initial marking m_0 , and
- a set Ω of final markings such that no transition of N is enabled at any $m \in \Omega$.

We further require that $m \in \Omega \cup \{m_0\}$ implies $m(p) = 0$ for all $p \in I \cup O$; that is, in the initial and the final markings the interface places are not marked.

We use indices to distinguish the constituents of different open nets (e. g. I_j refers to the set of input places of open net N_j).

The behavior of an open net is defined using the standard Petri net semantics [7]; that is, a transition is enabled if each place of its preset holds a token. An enabled transition t can fire in a marking m by consuming tokens from the preset places and producing tokens on the postset places, yielding a marking m' . The firing of t is denoted by $m \xrightarrow{t} m'$ (a t -step), the successively firing of a sequence of transitions is denoted by $m \xrightarrow{*} m'$.

In order to assign a reasonable meaning to *final* markings, we restrict our approach to such open nets where a marking in Ω does not enable any transition.

As an example, consider the open net N_c depicted in Fig. 1(a). The initial marking is $m_{0_{N_c}} = [p_0]$ and the set of final markings is defined by $\Omega_{N_c} = \{[p_7]\}$. N_c has three input and four output places that are depicted on the dashed frame: $I_{N_c} = \{\text{req_c}, \text{cc_y}, \text{cc_n}\}$ and $O_{N_c} = \{\text{r_low}, \text{r_high}, \text{rej}, \text{acc}\}$. The open net models a credit approval process of an online banking service. After the customer has requested a credit (transition t_1), the bank decides whether the risk is high or low (transitions t_2 and t_3). Then, the customer has to decide whether he accepts a credit control or not (transitions $t_4 - t_7$). Based on this information the bank distinguishes three cases: If the risk is high and the customer does not accept a credit control, then the credit request is rejected (transition t_8). If there is only low risk and the customer accepts a credit control, then the request is accepted (transition t_{11}). In the third case, that is, if the risk is high and the customer accepts a credit control or the risk is low but the customer does not accept a

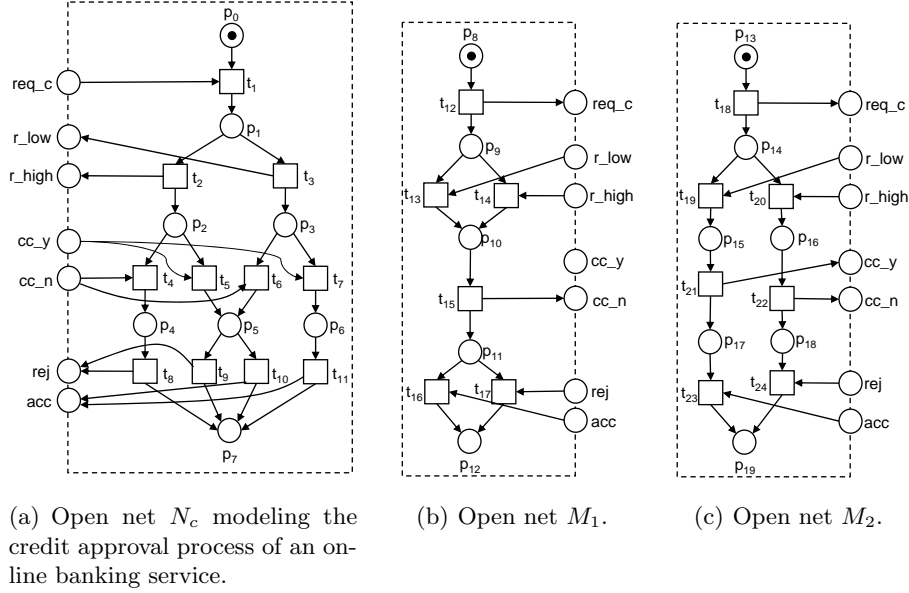


Fig. 1. The running example process N_c and two strategies M_1 and M_2 .

credit control, the request is examined by an employee of the bank which is modeled by a nondeterministic choice (transitions t_9 and t_{10}).

The $inner_N$ of an open net N defines the Petri net that results from removing the interface places and the adjacent arcs from N . Obviously, $inner_N$ and N coincide if N has an empty interface. The inner of N_c , $inner_{N_c}$, is the net inside the dashed frame in Fig. 1(a).

As a correctness criterion for an open net N we require the absence of deadlocks in N .

Definition 2 (Deadlock). *Let N be an open net. A deadlock is a nonfinal marking in N that does not enable a transition. If N does not have deadlocks, it is called deadlock-free.*

Two open nets M and N are *composable* if all constituents (except for the interfaces) are pairwise disjoint. This can be achieved easily by renaming. For the interfaces, we require that the input places of M are the output places of N and vice versa (i.e. $I_M = O_N$ and $O_M = I_N$). For markings $m_M \in M, m_N \in N$, their composition $m = m_M \oplus m_N$ is defined by $(m_M \oplus m_N)(p) = m_M(p) + m_N(p)$ (assuming $m_M(p) = 0$ for $p \notin P_M$ and $m_N(p) = 0$ for $p \notin P_N$). These considerations lead to the following definition of composition.

Definition 3 (Composition of open nets). *Let M, N be composable open nets. Then, the composition of M and N is the open net $M \oplus N$ defined as follows:*

- $P = P_M \cup P_N$,
- $T = T_M \cup T_N$,
- $F = F_M \cup F_N$,
- $I = O = \emptyset$,
- $m_0 = m_{0_M} \oplus m_{0_N}$, and
- $\Omega = \{m_M \oplus m_N \mid m_M \in \Omega_M, m_N \in \Omega_N\}$.

Consider the two open nets M_1 and M_2 depicted in Fig. 1(b) and Fig. 1(c), respectively and assume $m_{0_{M_1}} = [p_8]$, $\Omega_{M_1} = \{[p_{12}]\}$, $m_{0_{M_2}} = [p_{13}]$, and $\Omega_{M_2} = \{[p_{19}]\}$. Then, N_c and M_1 as well as N_c and M_2 are composable. Notice that place `cc.y` becomes internal in the composition $N_c \oplus M_1$, but it is never marked.

Clearly, we are mostly interested in composing open nets such that the composition is deadlock-free. To this end, we define the notion of a strategy.

Definition 4 (Strategy). *An open net M is a strategy for an open net N if $M \oplus N$ is deadlock-free. $\text{Strat}(N)$ denotes the set of all strategies for N .*

Both, $M_1 \oplus N_c$ and $M_2 \oplus N_c$, are deadlock-free and thus, M_1 and M_2 are strategies for N_c .

2.2 Operating Guidelines

In the following we recapitulate our concept of an *operating guideline* [8, 5]. With the help of operating guidelines we are able to represent the set of all strategies M for an open net N in a compact way. Technically, an operating guideline is a special annotated automaton. An annotated automaton A^Φ consists of a finite deterministic automaton A and a function Φ that assigns to each state q of A a Boolean formula $\Phi(q)$. A^Φ represents a set $\text{Strat}(A^\Phi)$ of open nets. For each element of $N \in \text{Strat}(A^\Phi)$, we say that N *matches* with A^Φ . We continue by first defining the notions of annotated automata and matching in general and then introducing operating guidelines.

Definition 5 (Annotated automaton). *$A^\Phi = [Q, C, \delta, q_0, \Phi]$ is an annotated automaton iff Q is a nonempty finite set of states, C is a set of labels, $\delta \subseteq Q \times C \times Q$ is a transition relation such that every state $q \in Q$ is reachable from q_0 via transitive applications of δ , $q_0 \in Q$ is the initial state, and Φ is an annotation function, where, for all $q \in Q$, $\Phi(q)$ is a Boolean formula over literals in C .*

We use annotated automata to represent a set of *open nets*. Therefore, we take an annotated automaton A^Φ with Boolean formulae over literals in $C = I \cup O$ and a special literal *final* and define when a service described in terms of an open net M with the interface $I \cup O$ matches with A^Φ . Intuitively, M matches with A^Φ if (1) its *behavior* is simulated by A^Φ and (2) if a marking m of M is simulated by a state q of A^Φ , then the arcs leaving m — interpreted as an assignment assigning *true* to the corresponding literals of the formula $\Phi(q)$ — satisfy $\Phi(q)$. For more details, we refer to [9, 5].

In order to simplify presentation, we assume that each transition of an open net is connected to at most one interface place. This assumption does, however, not restrict generality as every open net can be transformed into an equivalent one that obeys this restriction [5].

Definition 6 (Matching with A^Φ). *Let M be an open net that obeys the assumption stated above and let Y be the set of all reachable markings of the Petri net $M^* = \text{inner}_M$. Let $A^\Phi = (Q, C, \delta, q_0, \Phi)$ be an annotated automaton with $C = I_M \cup O_M \cup \{\text{final}\}$. Then M matches with A^Φ iff there is a relation $\rho \subseteq Y \times Q$ inductively defined as follows:*

1. $(m_{0_M}, q_0) \in \rho$;
2. If t is an internal transition of M (i. e., t is not connected to any interface place), $m, m' \in Y$, and $m \xrightarrow{t} m'$, then $(m, q) \in \rho$ implies $(m', q) \in \rho$;
3. If t is a receiving transition of M with $c \in I_M$, $c \in \bullet t$, $m, m' \in Y$, and $(m + [c]) \xrightarrow{t} m'$, then $(m, q) \in \rho$ implies $(m', q') \in \rho$ for some q' with $(q, c, q') \in \delta$;
4. If t is a sending transition of M with $c \in O_M$, $c \in t^\bullet$, $m, m' \in Y$, and $m \xrightarrow{t} (m' + [c])$, then $(m, q) \in \rho$ implies $(m', q') \in \rho$ for some q' with $(q, c, q') \in \delta$;
5. For all $m \in Y$, at least one of the following properties holds:
 - An internal transition t is enabled at m ; or,
 - for all q such that $(m, q) \in \rho$, $\Phi(q)$ evaluates to true for the following assignment β :
 - $\beta(c) = \text{true}$ if $c \in O_M$ and there is a transition t with $c \in t^\bullet$ that is enabled at m ;
 - $\beta(c) = \text{true}$ if $c \in I_M$ and there is a transition t with $c \in \bullet t$ that is enabled at $m + [c]$;
 - $\beta(c) = \text{true}$ if $c = \text{final}$ and $m \in \Omega_M$;
 - $\beta(c) = \text{false}$, otherwise.

Let $\text{Match}(A^\Phi)$ denote the set of all M such that M matches A^Φ .

In the formal definition, ρ represents the informally described (weak) simulation relation. The assignment used for evaluating an annotation represents transitions t of M that leave the considered marking m of M^* .

An operating guideline OG_N of an open net N is a special annotated automaton, such that an open net M matches with OG_N if and only if M is a strategy for N .

Definition 7 (Operating guideline). *An annotated automaton is an operating guideline OG_N of an open net N iff $\text{Strat}(N) = \text{Match}(OG_N)$.*

Figure 2 depicts the operating guideline OG_{N_c} for the credit approval process N_c (see Fig. 1(a)). It consists of 16 nodes and 31 edges and was calculated by our tool Fiona [10]. In the initial state q_0 , the annotation is `!cc.y ∨ !cc.n ∨ !req.c` reflecting the possible choices of a strategy M for N_c . More precisely, M must

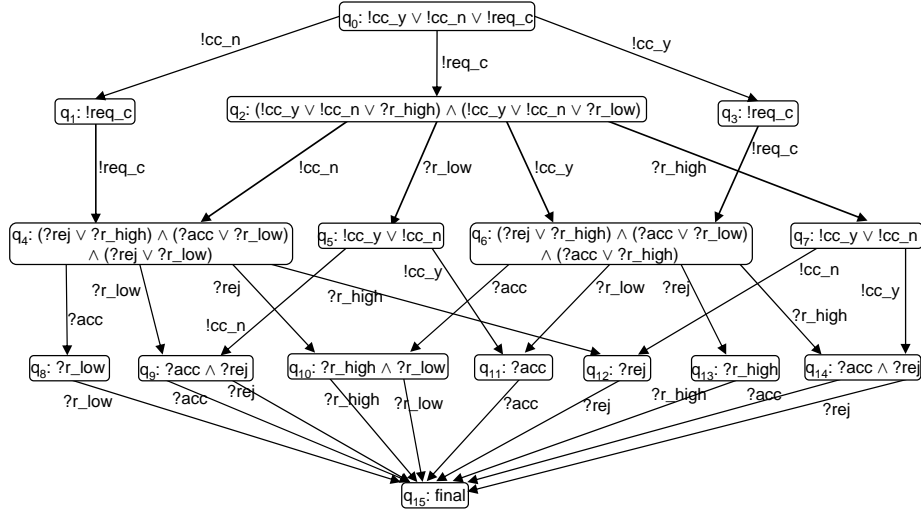


Fig. 2. The operating guideline OG_{N_c} for the credit approval process N_c depicted in Fig. 1(a). For better readability, we add a leading “!” (“?”) to a literal x in the graphics of an OG_N if x is an output (input) place of a strategy M for N .

be able to send at least one (expressed by the disjunction) of the three messages cc_y , cc_n , and req_c in its initial state. In contrast, annotation $?acc \wedge ?rej$ in state q_{14} reflects the fact that M being in marking m with $(m, q_{14}) \in \rho$ must be able to receive message acc and message rej . The two open nets M_1 and M_2 fulfil the requirements of Def. 6 and thus match with OG_{N_c} .

3 Covering Open Net Nodes

The notion of soundness guarantees (among others) the absence of dead transitions in a workflow net. In this section, this idea is adapted to open nets. For an open net N and a set $X \subseteq P_N \cup T_N$ of open net nodes, we will characterize those strategies M for N such that X is *covered* in the composition $M \oplus N$. Here, to cover a place p means that p can be marked in some reachable marking while to cover a transition t means that t is not dead. Such a strategy M is then called a *Cover $_X$ -strategy* for N . Clearly, if X contains all transitions of N , our coverage notion for open nets coincides with soundness, except for the fact that the composition may contain livelocks.

The motivation for dealing with *Cover $_X$ -strategies* is to figure out if some functionality of a service (i.e. some communication patterns), for example a credit approval, can in principle be used by other services. We further show how to calculate a finite representation of all *Cover $_X$ -strategies* for N by extending operating guidelines with a global constraint.

3.1 Deciding the Coverage of Open Net Nodes

In this section, we show how a strategy M for N can be discovered as a $Cover_X$ -strategy by just considering the operating guideline of N . In order to define our notion of $Cover_X$ -strategies, we need to define what it means to cover an open net node.

Definition 8 (Cover a place/transition). *Let $N = (P, T, F, I, O, m_0, \Omega)$ be a deadlock-free open net with empty interface ($I = O = \emptyset$), and let $X \subseteq P \cup T$, $p \in P$, and $t \in T$. N covers X iff for all $p \in X \cap P$ (for all $t \in X \cap T$) there exists a run of N that includes a marking m with $m(p) \geq 1$ (a t -step).*

Notice that if N covers two nodes, there is not necessarily a run in which both nodes are covered. In the example, transitions $t_1 - t_4$, t_6 , and $t_8 - t_{10}$ are covered in $M_1 \oplus N_c$ and transitions $t_1 - t_3$, t_5 , t_7 , and $t_9 - t_{11}$ are covered in $M_2 \oplus N_c$.

The following definition canonically extends strategies to strategies that cover a set X of open net nodes.

Definition 9 ($Cover_X$ -strategy). *Let M be a strategy for an open net N , and let $X \subseteq P_N \cup T_N$. M is a $Cover_X$ -strategy for N iff X is covered in $M \oplus N$. With $Strat_{Cover_X}(N)$ we denote the set of all $Cover_X$ -strategies for N .*

For N_c let $X = \{\text{acc}\}$ be given. That means, we are interested whether a credit approval is possible. Then, M_1 and M_2 are $Cover_X$ -strategies for N_c . Let $X = \{t_5, t_6\}$, that is, we are interested whether it is possible that a credit request has to be examined by an employee if the customer is not fixed in his credit control decision. Then M_1 is a $Cover_X$ -strategy for N_c , but M_2 is not (because transitions t_5, t_6 cannot be enabled in $M_2 \oplus N$).

By definition, every $Cover_X$ -strategy for N is also a strategy for N . Obviously, covering open net nodes restricts the set of strategies for N . Thus, we conclude $Strat_{Cover_X}(N) \subseteq Strat(N)$.

In the remainder of this section, we will define some notions and prove some properties of operating guidelines. Based on these properties, we can prove a criterion to decide whether an open net M is a $Cover_X$ -strategy for N . We start with the definition of the most permissive strategy for N . This strategy has the least restrictions of all strategies. Thus, the state space of its inner corresponds exactly to the transition system of the underlying automaton of OG_N .

Definition 10 (Most permissive strategy). *Let $OG_N = (Q, C, \delta, q_0, \Phi)$. The most permissive strategy for N is the open net $MP_N = (P, T, F, I, O, m_0, \Omega)$ whose behavior corresponds exactly to the transition system (Q, C, δ, q_0) with*

- $P = Q \cup C$,
- $T = \{t_{q_1, c, q_2} \mid (q_1, c, q_2) \in \delta, \text{ with } q_1, q_2 \in Q, c \in C\}$,
- $F = \{(q_1, t_{q_1, c, q_2}), (t_{q_1, c, q_2}, q_2) \mid (q_1, c, q_2) \in \delta\} \cup \left\{ \begin{array}{l} (c, t_{q_1, c, q_2}), \text{ if } c \in I; \\ (t_{q_1, c, q_2}, c), \text{ if } c \in O. \end{array} \right.$
- $I = O_N$,

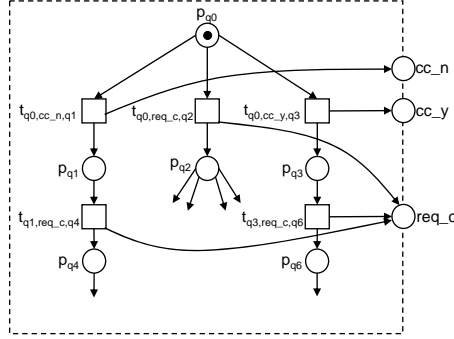


Fig. 3. The initial part of the most permissive strategy MP_{N_c} for N_c which has been constructed according to Def. 10.

- $O = I_N$,
- $m_0 = q_0$, and
- $\Omega = \{q \mid c \text{ is in } \Phi(q) \text{ with } c = \text{final}\}$.

The resulting open net MP is a state machine. Figure 3 illustrates the construction of the most permissive strategy MP_{N_c} of the operating guideline OG_{N_c} depicted in Fig. 2. As the whole open net would be too big, we depict only the first few nodes.

By the help of the following corollary, we prove that the most permissive strategy MP for N is indeed a strategy for N .

Corollary 1. *The most permissive strategy MP for N is a strategy for N .*

For the proof of this corollary, we rely on a fact about operating guidelines as constructed in [8]. As we cannot repeat the whole approach of [8], we just state this fact without proof.

Proposition 1 ([8]). *For every operating guideline $OG_N = (Q, C, \delta, q_0, \Phi)$ (of some service N) and all $q \in Q$, the formula $\Phi(q)$*

1. *uses only literals c where there is some $q' \in Q$ with $(q, c, q') \in \delta$, and*
2. *is satisfied for the assignment assigning true to all literals in $\Phi(q)$.*

Proof (of Corollary 1). Let $OG_N = (Q, C, \delta, q_0, \Phi)$. We construct open net MP as described in Def. 10. Let m_{q_0} be the initial marking of MP . By induction, it can be shown that, for all $q \in Q$, m_q is reached by Def. 6, with $(m_q, q) \in \rho$.

As there is a transition for each $(q, c, q') \in \delta$, we can derive from Prop. 1 that all annotations evaluate to *true* when MP is evaluated according to Def. 6. Consequently, MP matches with OG_N and hence MP is a strategy for N . \square

The next definition establishes a connection between markings of an open net N and the inner of a strategy $M \in \text{Strat}(N)$. If $inner_M$ is in a marking m , then $K(m)$ (the knowledge that $inner_M$ has about N) is the set of markings of N that N might be in while $inner_M$ is in marking m .

Definition 11 (Knowledge). Let M be a strategy for an open net N . Let $Mark_{M^*}$ and $Mark_N$ denote the set of all reachable markings of $inner_M$ and N , respectively. Let further m_M denote a marking of M and m_{M^*} denote its restriction to places in $inner_M$. The knowledge $K : Mark_{M^*} \rightarrow \mathcal{P}(Mark_N)$ that $inner_{MP}$ has about the possible markings of N in marking m_{M^*} is defined by $K(m_{M^*}) = \{m_N \mid (m_M \oplus m_N) \text{ is reachable from } (m_{0_M} \oplus m_{0_N})\}$.

For the most permissive strategy MP_{N_c} for N_c (see Fig. 3), we have the following knowledge values:

$$\begin{aligned} K([p_{q0}]) &= \{[p_0]\}, \\ K([p_{q1}]) &= \{[p_0, cc_n]\}, \\ K([p_{q2}]) &= \{[p_0, req_c], [p_1], [p_2, r_high], [p_3, r_low]\}, \\ K([p_{q3}]) &= \{[p_0, cc_y]\}, \\ K([p_{q4}]) &= \{[p_0, cc_n, req_c], [p_1, cc_n], [p_2, cc_n, r_high], [p_3, cc_n, r_low], \\ &\quad [p_4, r_high], [p_5, r_low], [p_7, r_high, rej], [p_7, r_low, acc], [p_7, r_low, rej]\} \end{aligned}$$

The simulation relation ρ used in Def. 6 actually establishes a relation between the knowledge values of the involved states. As the following proposition states, $(m, q) \in \rho$ implies that $K(m) \supseteq K(m_q)$ where m_q is the marking in the most permissive partner that corresponds to state q of an operating guideline.

Proposition 2 ([5]). Let M be a strategy for N and MP be the most permissive strategy for N . Let m_q denote the marking in $inner_{MP}$ that corresponds to state $q \in Q$ in OG_N (i.e. $(m_q, q) \in \rho_{MP}$). Let m be reachable in $inner_M$. Then $K(m) = \bigcup_{q:(m,q) \in \rho} K(m_q)$.

The matching relation ρ relates a marking m of $inner_M$ to a (possible) set of states q of OG_N . Therefore, the knowledge that $inner_M$ has about the possible markings of N in m is equivalent to the union of the knowledge values of all markings m_q of $inner_{MP}$ with $(m_q, q) \in \rho_{MP}$.

The notion of knowledge can be applied to the operating guideline OG_N of N . As every marking m_q in $inner_{MP}$ corresponds to a state q of OG_N , the knowledge OG_N has about N in q is equivalent to the knowledge $inner_{MP}$ has about N in m_q .

Definition 12 (Knowledge in OG). For an open net N let MP be the most permissive strategy for N and $OG_N = (Q, C, \delta, q_0, \Phi)$. Let $Mark_N$ denote the set of markings of N and m_q be a marking of $inner_{MP}$. The knowledge $K : Q \rightarrow \mathcal{P}(Mark_N)$ that OG_N has about the possible markings of N in state $q \in Q$ is defined by $K(q) = K(m_q)$.

The following theorem presents a way to decide, on the basis of an operating guideline, whether a strategy M for N is also a $Cover_X$ -strategy for N .

Theorem 1 (Place/Transition coverability). Let M be a strategy for open net N . A place $p \in P_N$ (a transition $t \in T_N$) is covered in $M \oplus N$ iff there is a state $q \in Q$ of OG_N , a marking m_M in $inner_M$, and a marking $m_N \in K(q)$ with $(m_M, q) \in \rho$, and $m_N(p) \geq 1$ (t is enabled in m_N).

Proof. We present the proof for the case of a covered transition only. The case of a covered place is analogous.

(\Rightarrow) Let N , OG_N , and $M \in \text{Strat}(N)$ be given and let transition t be covered in $M \oplus N$. Then, according to Def. 8, there is a run $m_{0_{M \oplus N}} \xrightarrow{t_1} \dots \xrightarrow{t_n} m_{M \oplus N} \xrightarrow{t} m'_{M \oplus N}$ in $M \oplus N$, $m'_{M \oplus N}(p) \geq 1$. Let m_M and m_N be the restrictions of marking $m_{M \oplus N}$ to places in inner_M and N , respectively. As t is a transition of N , t is enabled in m_N as well. By Def. 11, we have $m_N \in K(m_M)$. By Proposition 2, there must be a state q in OG_N where $m_N \in K(q)$ and hence the implication of this theorem holds.

(\Leftarrow) Let N be an open net and $OG_N = (Q, C, \delta, q_0, \Phi)$. Let M be a strategy for N . Since M is a strategy for N , there is a matching relation ρ of states in Q and markings in inner_M . Let m_M, q , and m_N be as assumed. Thus, $(m_M, q) \in \rho$, $m_N \in K(q)$, and t is enabled in m_N . From Proposition 2 follows $m_N \in K(m_M)$. Consequently, there is a run in $M \oplus N$ that reaches $m_M \oplus m_N$ which can be extended by an occurrence of t since activation of t in m_N implies activation of t in $m_M \oplus m_N$. Since every run in $M \oplus N$ is deadlock-free (follows from M being a strategy for N), we can conclude that the considered run is deadlock-free, too. So there exists a deadlock-free run in $M \oplus N$ where t is covered and hence the replication of this theorem holds. \square

The value of Theorem 1 is that it gives us a criterion to check whether an open net node is covered or not. A place p of N is covered by a strategy for N if there is a state q in OG_N and the knowledge in q contains a marking of N where p is marked. A transition t of N is covered by a strategy for N if there is a state q and the knowledge in q contains a marking m of N where t is enabled. That way, it is easily possible to annotate each state q of OG_N with all places and transitions which are covered in q . This can be done during the calculation of the operating guideline.

As an example, based on the knowledge values $K([p_{q_0}]) - K([p_{q_4}])$ we presented above we can derive the following sets of nodes of N_c that are covered in states $q_0 - q_4$ of OG_{N_c} :

$$\begin{aligned} q_0 &: \{p_0\} \\ q_1 &: \{p_0, cc_n\} \\ q_2 &: \{p_0 - p_3, req_c, r_high, r_low, t_1 - t_3\} \\ q_3 &: \{p_0, cc_y\} \\ q_4 &: \{p_0 - p_5, p_7, cc_n, cc_y, req_c, r_high, r_low, acc, rej, t_1 - t_4, t_6, t_8 - t_{10}\} \end{aligned}$$

3.2 A Finite Representation of all $Cover_X$ -Strategies

In this section, we introduce a notion of an operating guideline with a global constraint as a representation of all $Cover_X$ -strategies for N . We further present an algorithm for deciding when an open net M matches with such an operating guideline.

Consider again our running example N_c in Fig. 1(a). Assume we want to cover $X = \{\text{acc}\}$ in N_c , that is, we are interested in strategies in which a customer may receive an approval for his credit request. We have $[\text{acc}] \in K(\mathbf{q}_4), K(\mathbf{q}_6), K(\mathbf{q}_9), K(\mathbf{q}_{11}), K(\mathbf{q}_{14})$. So according to Theorem 1, a strategy M for N_c is a $Cover_X$ -strategy for N_c if it has at least a marking m_{acc} of $inner_M$ that matches with $\mathbf{q}_4, \mathbf{q}_6, \mathbf{q}_9, \mathbf{q}_{11}$, or \mathbf{q}_{14} . As a second example assume $X = \{\mathbf{t}_5, \mathbf{t}_6\}$, that is, we are interested in strategies in which a customer is not fixed in his credit control decision and the credit request can be examined by an employee. In that case we have $[\mathbf{t}_5] \in K(\mathbf{q}_6), K(\mathbf{q}_{14})$ and $[\mathbf{t}_6] \in K(\mathbf{q}_4), K(\mathbf{q}_9)$. So M is a $Cover_X$ -strategy for N_c if it has at least a marking m_{t5} of $inner_M$ that matches with \mathbf{q}_6 or \mathbf{q}_{14} and it has a marking m_{t6} of $inner_M$ that matches with \mathbf{q}_4 or \mathbf{q}_9 .

The examples illustrate that is in general not possible to express the constraints for covering open net nodes in the shape of local annotations in each state of the operating guideline. Consequently, the present concept of an annotated automata fails at representing all $Cover_X$ -strategies of N . To overcome this problem, we propose another representation of all $Cover_X$ -strategies of N that takes the non-locality of covering open net nodes into account. To this end, we will slightly enhance the concept of an operating guideline.

Consider again the example above. Since OG_{N_c} (see Fig. 2) represents all strategies and every $Cover_X$ -strategy for N_c is a strategy for N_c , we have to restrict OG_{N_c} to $Cover_X$ -strategies. This can be achieved by a *global constraint* specifying that, for every open net node $x \in X$ to be covered, at least one state q in OG_{N_c} with $x \in K(q)$ must be present in the matching relation between OG_{N_c} and a $Cover_X$ -strategy. This constraint can be expressed as a Boolean formula ψ_X .

In the following, we formalize annotated automata enhanced with a global constraint and define the matching relation between an open net and such an annotated automaton.

Definition 13 (Annotated automaton with global constraint). *Let $A^\Phi = (Q, C, \delta, q_0, \Phi)$ be an annotated automaton and ψ be a Boolean formula with propositions taken from the set Q . Then, $A^{\Phi, \psi} = (A^\Phi, \psi)$ is an annotated automaton with global constraint ψ .*

As an example for a global constraint to OG_{N_c} , consider $\psi = (\mathbf{q}_6 \vee \mathbf{q}_{14}) \wedge (\mathbf{q}_4 \vee \mathbf{q}_9)$. This formula is satisfied if and only if true is assigned to sufficiently many states to cover set $X = \{\mathbf{t}_5, \mathbf{t}_6\}$.

Enhancing an annotated automaton with a global constraint makes it necessary to redefine the matching relation of an open net M with an annotated automaton. M matches with an annotated automaton with global constraint $A^{\Phi, \psi}$ if it matches with the annotated automaton A^Φ , and in addition satisfies ψ .

Definition 14 (Matching with $A^{\Phi, \psi}$). *Let M be an open net, and let $A^{\Phi, \psi}$ be an annotated automaton A^Φ with global constraint ψ . M matches with $A^{\Phi, \psi}$ iff M matches with A^Φ using relation ρ and ψ evaluates to true in the assignment*

$\gamma_M : Q_A \rightarrow \{true, false\}$ where $\gamma_M(q) = true$ iff there is a marking m of M such that $(m, q) \in \rho$.

Finally, we are ready to construct the operating guideline with global constraint $OG_{\psi_X}(N)$ of an open net N as a representation of the set $Strat_{Cover_X}(N)$ of all $Cover_X$ -strategies for N .

Definition 15 (Global constraint for covering X). Let N be an open net and OG_N an operating guideline of N . Let $X \subseteq P_N \cup T_N$. For a place $p \in P$, let $p \sim q$ iff there is an $m \in K(q)$ where $m(p) > 0$. For a transition $t \in T$, let $t \sim q$ iff there is an $m \in K(q)$ where t is enabled. Then ψ_X is the formula

$$\bigwedge_{x:x \in X} \bigvee_{q:x \sim q} q$$

$OG_{\psi_X}(N) = (OG_N, \psi_X)$ defines an operating guideline with global constraint of N .

As a direct consequence of Theorem 1, we obtain the main result of this section, that is, $OG_{\psi_X}(N)$ represents all $Cover_X$ -strategies for N .

Theorem 2. M is a $Cover_X$ -strategy for N iff M matches with $OG_{\psi_X}(N)$.

The operating guideline representing all $Cover_X$ -strategies for N_c with $X = \{t_5, t_6\}$ is the operating guideline $OG_{\psi_X}(N_c) = (OG_{N_c}, \psi_X)$ where $\psi = (q_6 \vee q_{14}) \wedge (q_4 \vee q_9)$ as stated above. If we consider again open nets M_1 and M_2 (which are both strategies for N_c), then we get that M_1 matches with $OG_{\psi_X}(N_c)$ and it is hence a $Cover_X$ -strategy for N_c . In contrast, M_2 does not match with $OG_{\psi_X}(N_c)$, because it does not satisfy the global constraint. More precisely, there is no marking in $inner_{M_2}$ that matches with any of the nodes q_4 , q_6 , q_9 , and q_{14} .

As another example, let $X = \{t_1, \dots, t_{11}\}$, meaning all transitions of N_c should not be dead in $M \oplus N$. Then, $OG_{\psi_X}(N_c)$ has the following global constraint:

$$\begin{aligned} \psi_X = & (q_2 \vee q_4 \vee q_6) \wedge (q_2 \vee q_4 \vee q_6) \wedge (q_2 \vee q_4 \vee q_6) \wedge (q_4 \vee q_{12}) \\ & \wedge (q_6 \vee q_{14}) \wedge (q_4 \vee q_9) \wedge (q_6 \vee q_{11}) \wedge (q_4 \vee q_{12}) \\ & \wedge (q_4 \vee q_6 \vee q_9 \vee q_{14}) \wedge (q_4 \vee q_6 \vee q_9 \vee q_{14}) \wedge (q_6 \vee q_{11}) \end{aligned}$$

which is equivalent to

$$\psi_X = (q_2 \vee q_4 \vee q_6) \wedge (q_4 \vee q_{12}) \wedge (q_6 \vee q_{11}) \wedge (q_4 \vee q_6 \vee q_9 \vee q_{14}).$$

3.3 Discussion

In the following we will compare (ordinary) operating guidelines and operating guidelines with global constraint. We further discuss some complexity issues.

Comparing an operating guideline OG_N for N and an operating guideline with global constraint $OG_{\psi_X}(N)$ for N , we identify that both operating guidelines have the same underlying automaton. This is caused by the fact that each $Cover_X$ -strategy for N is also a strategy for N . Furthermore, if the most permissive strategy for N is not a $Cover_X$ -strategy for N , then the set of $Cover_X$ -strategies is empty.

Computing OG_N is proportional in time to the product of the number of states of N and an over-approximation of its most permissive strategy [9]. For $OG_{\psi_X}(N)$ the time complexity does not change, because all information necessary for annotating the states $q \in Q$ with the nodes of N and setting up the global constraint have to be computed for OG_N anyway. In order to increase efficiency, it is sufficient to annotate each state q only with open net nodes of X .

The space complexity of OG_N is proportional to the product of the number of states of N and its most permissive strategy [9]. If we compute $OG_{\psi_X}(N)$, then this complexity increases due to ψ_X . The global constraint is a conjunction of at most $|X|$ disjunctions where each disjunction may consist of at most $|Q|$ literals. Hence, the size of the global constraint is at most $O(|X| \cdot |Q|)$. The example suggests that the size of the global constraint will be much smaller in practice.

Although the time and space complexity of OG_N is high, experimental results have shown that the calculation of OG_N is feasible in practical applications both for time and space (see [5], for instance). Based on the complexity considerations for $OG_{\psi_X}(N)$ we conclude that the calculation of $OG_{\psi_X}(N)$ will be feasible in practical applications, too.

Matching an open net M with OG_N is proportional in time to the number of states in $M \oplus N$ [9]. If we match M with $OG_{\psi_X}(N)$, we additionally have to check whether the global constraint is satisfied by the assignment γ_M . This can be done in linear time w.r.t. the size of the constraint.

As the space complexity and the matching complexity for the proposed notion of operating guidelines with global constraint only marginally increase in comparison with ordinary operating guidelines, we can conclude that this novel notion is a well-suited instrument for service composition.

4 Related Work

The work presented in this paper is mainly inspired by the notion of soundness for workflow nets [2]. Soundness guarantees the absence of deadlocks, livelocks, and dead transitions. In this paper, we adopt the idea of soundness to our service model open nets. Given an open net N , we are interested in all open nets M such that the composition $M \oplus N$ is deadlock-free and certain places and transitions of N are covered. Here, cover means that these places can be marked and the transitions are not dead in $M \oplus N$. So far in contrast to soundness, our approach is limited in the sense that the composed system may contain livelocks.

There is also some relation to the research fields testing and computer-aided verification. In these fields testing/checking the coverage of certain activities is also a known and important sanity check, see [11], for instance.

Besides the relation to soundness, covering open net nodes can also be seen as a behavioral constraint for services. In [6] the authors introduced two kinds of behavioral constraints: to *enforce* and to *exclude* a set of open net nodes. A strategy M for N enforces (excludes) a transition t of N if every (no) run in $M \oplus N$ includes a t -step. Covering a transition t is thus equivalent to *not exclude* t . However, cover cannot be expressed by the approach proposed in [6], because the authors model a constraint as a constraint open net C and compose C and N .

5 Conclusion

We proposed an approach to guarantee the coverage of certain activities in services. Our approach is inspired by the notion of soundness on the one hand and by the work on behavioral constraints for services on the other hand. We have shown that with the operating guideline of a service N we can decide if a partner service M is designed in a way such that a given set of activities of N can be covered in the composition of M and N . We further presented a finite representation of all partner services M by extending our notion of an operating guideline with a global constraint. The results presented in this paper have been implemented in our analysis tool FIONA³ [10]. The proposed approach can also be applied to industrial service models specified in WS-BPEL [12]. To this end the compiler BPEL2OWFN⁴ [10] can be used to translate such a WS-BPEL process into an open net. The resulting net can then be analyzed by FIONA.

The advantage of the proposed approach is that the global constraint can be calculated based on the information that is already present when calculating the operating guideline. In addition, the global constraint does only marginally increase the complexity of matching a service with the operating guideline with global constraint. Thus operating guidelines with global constraint are a well-suited instrument for service composition.

In ongoing work plan to deal with a stricter correctness criterion that also excludes livelocks. That way, we can close the gap to soundness. Furthermore, an open net N can be substituted by an open net N' w.r.t. the coverage of X if and only if every $Cover_X$ -strategy for N is also a $Cover_X$ -strategy for N' (i.e. $Strat_{Cover_X}(N') \supseteq Strat_{Cover_X}(N)$). To this end we are working on an algorithm to automatically decide substitutability w.r.t. the coverage of X .

Acknowledgements The authors wish like to thank Robert Danitz and Janine Ott for their work on the implementation of FIONA. Christian Stahl is funded by the DFG project “Substitutability of Services” (RE 834/16-1). Karsten Wolf

³ available at <http://www.service-technology.org/fiona>

⁴ available at <http://www.service-technology.org/bpel2owfn>

is supported by the DFG within grant “Operating Guidelines for Services” (WO 1466/8-1).

References

1. Papazoglou, M.P.: Web Services: Principles and Technology. Pearson - Prentice Hall, Essex (2007)
2. Aalst, W.M.P.v.d.: The Application of Petri Nets to Workflow Management. *The Journal of Circuits, Systems and Computers* **8**(1) (1998) 21–66
3. Aalst, W.M.P.v.d., Weske, M.: The P2P approach to Interorganizational Workflows. In Dittrich, K.R., Geppert, A., Norrie, M.C., eds.: Proceedings of the 13th International Conference on Advanced Information Systems Engineering (CAiSE’01). Volume 2068 of Lecture Notes in Computer Science., Interlaken, Switzerland, Springer-Verlag (2001) 140–156
4. Leymann, F., Roller, D., Schmidt, M.T.: Web services and business process management. *IBM Systems Journal* **41**(2) (2002) 198–211
5. Lohmann, N., Massuthe, P., Wolf, K.: Operating guidelines for finite-state services. In Kleijn, J., Yakovlev, A., eds.: 28th International Conference on Applications and Theory of Petri Nets and Other Models of Concurrency, ICATPN 2007, Siedlce, Poland, June 25-29, 2007, Proceedings. Volume 4546 of Lecture Notes in Computer Science., Springer-Verlag (2007) 321–341
6. Lohmann, N., Massuthe, P., Wolf, K.: Behavioral constraints for services. In Alonso, G., Dadam, P., Rosemann, M., eds.: Business Process Management, 5th International Conference, BPM 2007, Brisbane, Australia, September 24-28, 2007, Proceedings. Volume 4714 of Lecture Notes in Computer Science., Springer-Verlag (2007) 271–287
7. Reisig, W.: Petri Nets. EATCS Monographs on Theoretical Computer Science edn. Springer (1985)
8. Massuthe, P., Schmidt, K.: Operating Guidelines – an Automata-Theoretic Foundation for the Service-Oriented Architecture. In Cai, K., Ohnishi, A., Lau, M., eds.: Proceedings of the Fifth International Conference on Quality Software (QSIC 2005), Melbourne, Australia, IEEE Computer Society (2005) 452–457
9. Massuthe, P., Wolf, K.: An Algorithm for Matching Non-deterministic Services with Operating Guidelines. *International Journal of Business Process Integration and Management (IJBPIIM)* **2**(2) (2007) 81–90
10. Lohmann, N., Massuthe, P., Stahl, C., Weinberg, D.: Analyzing Interacting BPEL Processes. In Dustdar, S., Fiadeiro, J., Sheth, A., eds.: Fourth International Conference on Business Process Management (BPM 2006). Volume 4102 of Lecture Notes in Computer Science., Springer (2006) 17–32
11. Kupferman, O.: Sanity Checks in Formal Verification. In Baier, C., Hermanns, H., eds.: 17th International Conference on Concurrency Theory (CONCUR 2006), Bonn, Germany, August 27-30, 2006, Proceedings. Volume 4137 of Lecture Notes in Computer Science., Springer (2006) 37–51
12. Alves et al., A.: Web Services Business Process Execution Language Version 2.0. OASIS Standard, 11 April 2007, OASIS (2007)