Analysis of Realizability Conditions for Web Service Choreographies [⋆]

Raman Kazhamiakin and Marco Pistore

DIT, University of Trento
via Sommarive 14, 38050, Trento, Italy
[raman,pistore]@dit.unitn.it

Abstract. Web service choreography languages allow for the description of multipart collaborations from a global point of view, specifying the information exchanged by the participants in order to accomplish a common business goal. An important issue, emerging from the choreography modelling, is the protocol realizability, i.e., the possibility to extract the local specifications of the participants, so that their interactions preserve certain crucial properties of the global description.
In this paper, we present a formal framework for the definition of both the global protocols and the local specifications. The key feature of the approach is that it allows for arbitrary communication models (synchronous/asynchronous, with/without buffers) in the composition of the local specifications. We introduce a hierarchy of realizability notions that allows for capturing various properties of the global specifications, and associate specific communication models to each of them. We also present an approach, based on the analysis of the communication models, that allows to associate a particular level of realizability to the global protocol specification.

1 Introduction

Web service technology facilitates the development of complex distributed systems that span across the enterprise boundaries. It enables the specification, deployment, and enactment of heterogeneous software components accessible on the web via standardized protocols. One of the fundamental ideas underlying the Web service technology is the possibility to provide composite business applications by integrating existing services. To make this possibility realistic, it is necessary to address the problem of the specification and design of such compositions, taking into account the various aspects of a composition description.

One of the key aspects of this description is the ability to represent the stateful and coordinated behavior of the composite system. A wide range of standards and languages has been proposed for capturing these aspects [1–3]. Among them, the *choreography* specification languages, like e.g. Web Services

[⋆] This work is partially funded by the MIUR-FIRB project RBNE0195K5, "KLASE", by the MIUR-PRIN 2004 project "STRAP", and by the EU-IST project FP6-016004 "SENSORIA".

Choreography Description Language (WS-CDL, [3]), are particularly relevant for the design phase of the composition, as they allow for the representation of the global observable behavior of the distributed business application.

The choreography languages open up the possibility of applying a range of formal techniques for the analysis of Web service compositions. Particularly relevant is the problem of *realizability* of the choreography specifications, that is, the possibility to automatically extract from the choreography the behavioral skeletons of the participants so that the concrete implementations, built on the basis of these skeletons, are guaranteed to satisfy the choreography specification.

This problem is made difficult by several crucial factors. First, the behavior of the application strongly depends on the way the services are exchanging the information, that is the *communication model* of the composition. The hypothesis of synchronous interactions, widely used for the analysis of service compositions, is not satisfied by many existing real-world applications. Moreover, the underlying interaction mechanism is not always known a priori, thus making problematic the usage of a particular model. Second, the strictness to which the application should satisfy the ordering constraints on messages and/or internal activities may differ from one scenario to another. In order to address this diversity, the realizability model has to be flexible, and should allow for an analysis which is parametric with respect to the set of requirements.

In this paper we address the problem of analyzing the realizability of choreographic protocols. We present a hierarchy of realizability notions that allows for capturing a variety of the choreography properties, thus providing a basis for a more flexible analysis. The presented approach is based on our previous work [4], where we give a formal model and analysis framework for the Web service compositions defined as composition of *local* participants specification. The key feature of that model is the definition of a parametric communication mechanism. More precisely, it is possible to specify different communication models for the interaction among the participants by changing the number of queues, their alphabets, and ordering rules. Using this formalism, it is possible to determine the communication model that is *adequate* for a given composition, that is the one that allows for the most complete description of the composition behavior.

In this work, we extend the approach of [4] and introduce a formalism for the *global* model that allows for the description of the compositions from a choreographic point of view. Furthermore, we express the problem of choreography realizability in terms of the composition of its local projections, and show how the hierarchy of the realizability notions is related to the hierarchy of communication models. Using this relation, we present an algorithm that allows to determine the appropriate level of the realizability, and the corresponding required conditions.

The paper is structured as follows. Section 2 introduces the realizability problem using variants of a simple example. Sections 3 and 4 define the formal models for the description of the underlying systems from the global perspective and as a composition of interacting local services respectively. In Sect. 5 the realizability problem is formalized and described, and various properties of the global protocols are discussed in terms of the composition of local models. Section 6 presents

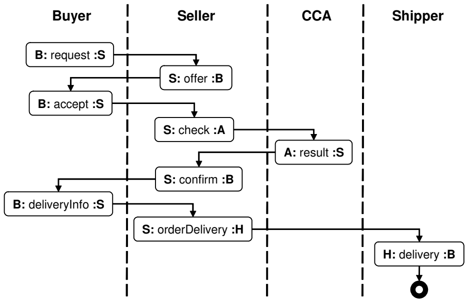

Fig. 1. RFQ case study: nominal case

the algorithm that permits to reason on the realizability properties of the global protocols, which is based on the analysis of the communication models of the compositions. Conclusions and related works are discussed in Sect. 7.

2 Choreography Models

We illustrate the problem of modelling and analyzing Web service choreographies by means of an example. In this example we model a composition of Web services from the global point of view. We incrementally present several variants of the scenario, illustrating the approach presented in this paper. The global representation of the composition is modelled using the WS-CDL language that allows to describe the observable behavior of the composite protocol.

In our example we describe a simple business collaboration protocol for purchasing goods among a buyer, a seller, a credit card agency (CCA), and a shipper. The nominal case, defined as a UML activity diagram, is presented in Fig. 1. The protocol is defined as follows. First, the Buyer asks the Seller for a particular good, sending a request. The quote offer is prepared and sent back to the Buyer that accepts it (interaction `accept`). Given the payment details, the Seller asks the CCA to verify the information, and after receiving a positive `result` message, sends a confirmation to the Buyer. The Buyer sends the address information to the Seller, which forwards this data to the Shipper (`orderDelivery` message). Finally, the Shipper sends the delivery confirmation to the Buyer (`delivery`).

Given this global description, a straightforward step is to extract from it the local specifications of the participants. These specifications may be further detailed and serve as the basis for the composition implementation. In this example, the local specifications can be easily obtained by *projection* of the global model onto a particular participant. Moreover, the composition of these projections will behave exactly as the global specification, and therefore, the global model is *realizable*.

This is, however, not always the case. Consider a modification of the nominal case represented in Fig. 2(a). Here, after the `orderDelivery` interaction

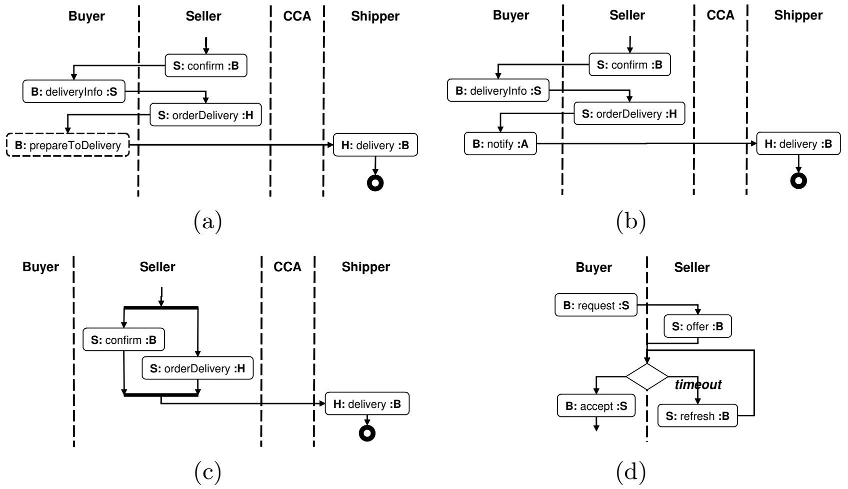

Fig. 2. RFQ case study: implementation variants

has been performed, the internal activity `prepareToDelivery` is invoked in the Buyer role. While the global model requires that the delivery is performed after this activity, the composition of projections is not able to guarantee this. Indeed, this internal activity of the Buyer is independent from the emission of delivery notification of the Shipper, and therefore the realizability is violated. In order to resolve this problem, either the specification should be modified and an additional synchronization interaction should be added, or the ordering requirement should be relaxed, allowing for interleaving internal activities and interactions.

A different modification of the nominal case is presented in Fig. 2(b). Here, after the `orderDelivery` interaction, the CCA is notified about the order by the `notify` message. In this case, also the ordering of interactions is violated, since the notification of CCA is independent from the emission of the delivery notification. We remark however that, while the order of actions specified in the global protocol is not respected, the states of the participants and the state of the protocol is not affected. That is, this scenario satisfies the property that, even if the actions are reordered, the outcome of the execution is the same.

In the implementation presented in Fig. 2(c) this property does not hold. In this scenario the addressing information of the Buyer is not required, and the confirmation and the order delivery activities may be invoked in parallel. This may lead to the following problem in the composition. The local specification of the Buyer requires that the delivery notification arrives after the confirmation from the Seller. Due to the fact that the Seller and the Shipper are independent, this order may be violated, and the final state of the Buyer depends on the implementation of the underlying communication system. Indeed, if the messages may be stored in queues, and, moreover, the messages from different partners are managed independently, then the Buyer process is not blocked, the messages are simply consumed in a different order. As a result, under certain assumptions on the middleware implementation and the usage of local variables, the global protocol may still be considered correct regardless the message ordering problems.

Contrary to the above examples, the acceptable conditions for the scenario presented in Fig. 2(d) are much more difficult to define. Here, if the Buyer does not provide an acceptance within a given period of time, the Seller refreshes the good information, sending a corresponding message `refresh`. The first problem is that the refresh message may be sent simultaneously with the acceptance message, even if the good is not in the stock anymore. The `refresh` message is ignored and the state of the protocol is incorrect. Second, if the time required for the offer processing by the Buyer exceeds the refreshment timeout, then the Seller will send more and more messages, and the queue of the Buyer will grow unboundedly. Finally, when the Buyer sends an acceptance message after several offer refresh messages, it is not clear which one is accepted due to possible reorderings and intersections of messages leading again to an incorrect state.

This example shows the complexity of defining realizability in a way that makes it possible to deal with scenarios like the ones illustrated above.

3 Global Model

In this section we introduce the formal model for representing the choreography specification. This global model defines the dynamic aspects of the service compositions from a global point of view by defining the involved participants and their interactions. The formalism follows the approach of [5, 6] for modelling the global interaction protocols of Web service compositions.

The model is based on the notion of *roles* and *actions*. A role represents the behavior of a particular participant of the composed system. During the protocol execution, the i^{th} role can be in one of its possible *states* $\mathcal{S}_i$ (one of which is marked as an *initial state* s_{0i}) and can evolve to new states as a result of performing some actions.

We model message communications as *interactions* defined on a set of service operations (or message types) M. The signature of the interaction a_o has the form (r_s, r_d, μ), where r_s and r_d are the roles of the sender and receiver respectively, μ is the service operation. The set of interactions is denoted as $\mathcal{A}_O$.

We also define *internal actions* $\mathcal{A}_\tau$, which are used to represent evolutions of the system that do not involve interactions between services. In particular, such an action may represent the internal decision branching of a particular participant (or even a group of participants[1]), or a modification of variables being performed by one or more partners in parallel. An internal action a_τ has the form $(\mathcal{R}_\tau, \tau)$, where $\mathcal{R}_\tau \subseteq \mathcal{R}$ denotes a subset of roles that perform an action, and τ is used to denote the internal action itself.

[1] The possibility of a group of participants to perform an internal action is used in [3] to model simultaneous evaluation of the branching condition by a group of roles.

3.1 Global Protocol

We model choreography behavior as a *global protocol.* This behavior is defined by the *global transition relation* $\mathcal{T}$, which describes how the states of the participants can evolve on the basis of external and internal actions. We represent a *global state* of the choreography as a vector $\bar{s} = \langle s_1, \ldots, s_n \rangle$, where s_i is a local state of the role r_i. We denote a vector $\bar{s}$ with component s_i updated to s'_i as $\bar{s}[s'_i/s_i]$.

Definition 1 (Global Protocol). *A* global protocol *representing the collaboration of n roles is a tuple $P = \langle \mathcal{R}, \mathcal{S}, \bar{s}_0, \mathcal{A}, \mathcal{T} \rangle$, where*

- *$\mathcal{R}$ is a set of n roles;*
- *$\mathcal{S} \subseteq \mathcal{S}_i \times \cdots \times \mathcal{S}_n$ is a set of global states, and $\bar{s}_0 \in \mathcal{S}$ is an initial state;*
- *$\mathcal{A} = \mathcal{A}_\tau \cup \mathcal{A}_O$ is a set of actions;*
- *$\mathcal{T} \subseteq \mathcal{S} \times \mathcal{A} \times \mathcal{S}$ is a global transition relation. A transition $(\bar{s}, a, \bar{s}') \in \mathcal{T}$ if*
  - *$a = (r_a, r_b, \mu)$ and $\bar{s}' = \bar{s}[s'_a/s_a, s'_b/s_b]$, or*
  - *$a = (\mathcal{R}_\tau, \tau)$ and $\bar{s}'$ agrees with $\bar{s}$ except for the state of the roles in $\mathcal{R}_\tau$.*

3.2 Behavior of the Global Model

We define the behavior of the global protocol using the notions of a *run* and a *conversation.* A run describes the evolution of the system by specifying, which actions were performed, and which global states were visited. The conversation, on the contrary, records only the sequences of interactions being performed during the evolution of the system.

More formally, the protocol behavior is defined as follows. We say that an action $a \in \mathcal{A}$ is *fireable* in a state $\bar{s}$, denoted as $\bar{s} \xrightarrow{a} \bar{s}'$, if there is a transition $(\bar{s}, a, \bar{s}') \in \mathcal{T}$. Let $\pi = \bar{s}_1, a_1, \bar{s}_2, a_2, \ldots$ be a (possibly infinite) sequence of states and actions. We say that the sequence is fireable from $\bar{s}_1$, written as $\bar{s}_1 \xrightarrow{\pi}^{*}$, if $\forall k \geq 1,\ \bar{s}_k \xrightarrow{a_k} \bar{s}_{k+1}$. Let us also denote as $\rho(\pi) = \mu_1, \mu_2, \ldots$ a sequence of all the interactions appeared on the sequence π.

Definition 2. *Let $P = \langle \mathcal{R}, \mathcal{S}, \bar{s}_0, \mathcal{A}, \mathcal{T} \rangle$ be a global protocol. A* run *of the protocol is a sequence $\pi = \bar{s}_0, a_0, \bar{s}_1, a_1, \ldots$ such that $\bar{s}_0 \xrightarrow{\pi}^{*}$. The* behavior *of the protocol is the set of all the runs of the protocol:* $\mathrm{B} = \{\pi \mid \bar{s}_0 \xrightarrow{\pi}^{*}\}$.
A conversation *of the protocol $\rho(\pi)$ is a sequence of interactions performed on some run π. The set of all the conversations, denoted as Ω, is called the* conversation set *of the protocol:* $\Omega = \{\rho(\pi) \mid \bar{s}_0 \xrightarrow{\pi}^{*}\}$.

4 Local Model

In the local model, the Web service composition is defined by a set of local protocols that separately describe the behavior of each participant of the composition. During their executions, the participants exchange messages with other participants through a certain communication medium, thus forming the dynamic behavior of the composed system. This bottom-up representation of the composition, inspired by the specification languages like BPEL [1], relies on the notions of a *local protocol*, which describes the behavior of a particular actor of the composition, and a *communication model*, which characterizes the interaction mechanisms of the composition instance.

4.1 Composition of Local Protocols

The local behavior of the participant is given in terms of a local protocol that specifies the states and actions performed by the participant. We distinguish *input actions* $\mathcal{I}$, which represent the reception of message, denoted as $\overleftarrow{\mu}$; *output actions* $\mathcal{O}$, which represent messages sent to other participants, denoted as $\overrightarrow{\mu}$; and *internal actions*, which characterize non-observable operations of the participant, denoted as τ.

Definition 3 (Local protocol). *A* local protocol *is a tuple* $\langle \mathcal{S}, s_0, \mathcal{A}, \mathcal{T} \rangle$*, where*

- $\mathcal{S}$ *and* $s_0 \in \mathcal{S}$ *are the set of role states and the initial state respectively;*
- $\mathcal{A} = \mathcal{I} \cup \mathcal{O} \cup \{\tau\}$ *is a set of role actions;*
- $\mathcal{T} \subseteq \mathcal{S} \times \mathcal{A} \times \mathcal{S}$ *is a local transition relation.*

The behavior of the composition of local protocols depends on the communication model [4] adopted and used for the description of the message exchanges between partners. Such a communication model is given by a set of communication channels (or queues), and is characterized by the number of the queues, message ordering, bounds etc. The composition of the local projections is therefore parametric with regards to the communication model, and may exhibit different behaviors.

More formally, let us model the interactions with set of $m > 0$ queues with disjoint alphabets $\mathrm{M}_j \subseteq \mathrm{M}$. A queue q_j may be declared as *bounded*, with the corresponding capacity $0 < b_j < \infty$, or *unbounded*, in which case $b_j = \infty$.

Definition 4. *A* communication model *for the composition is a tuple* $\Delta = \langle \mathcal{L}_B, \mathcal{L}_M, \mathcal{L}_O \rangle$*, where* $\mathcal{L}_B = \langle b_1, \ldots, b_m \rangle$*, is a vector of queue bounds,* $\mathcal{L}_M : \mathrm{M} \to [1 \ldots m]$ *is a function that associates an operation* μ *with a queue* i*, and* $\mathcal{L}_O : [1 \ldots m] \to \{\top, \bot\}$ *is a function that declares the queue as either ordered or unordered. The alphabet* M_i *of queue* i *is defined as* $\mathrm{M}_i = \{\mu \mid \mathcal{L}_M(\mu) = i\}$*.*

Let M^* be a set of sequences (or strings) of elements from M. Let also $\mathbb{N}^{\mathrm{M}}$ be a set of multisets of M, i.e. sets of mappings from M to the set $\mathbb{N}$ of natural numbers. Given two elements w and w', we write $w.w'$ to denote string concatenation, if $w, w' \in \mathrm{M}^*$, and multiset union, if $w, w' \in \mathbb{N}^{\mathrm{M}}$.

We define a queue content as a vector $C = \langle w_1, \ldots, w_m \rangle$, where $w_j \in \mathrm{M}_j^*$ and j^{th} queue is ordered, or $w_j \in \mathbb{N}^{\mathrm{M}}$ and j^{th} queue is unordered. We extend the operator . to the queue content as follows: $C.\mu = \langle w_1', \ldots, w_m' \rangle$, where $w_j' = w_j.\mu$ if $\alpha \in \mathrm{M}_j$, and $w_j' = w_j$ otherwise. We write $|C| \leq \mathcal{L}_B$ to specify that $|q_i| \leq b_i$.

We define the composition of the local protocols as a *composition transition system* (CTS). The definition of the CTS is parametric with respect to the communication model.

Definition 5 (CTS [4]). *A* composition transition system *representing the composition of n local protocols under a model $\Delta = \langle \mathcal{L}_B, \mathcal{L}_M, \mathcal{L}_O \rangle$ is a transition system $\Sigma_\Delta = \langle \mathit{\Gamma}, \gamma_0, \mathcal{A}, \mathcal{T} \rangle$, where*

- *$\mathit{\Gamma}$ is a set of* configurations *of the form $\gamma = \langle \bar{s}, C \rangle$, and $\gamma_0 = \langle \bar{s}_0, \langle \epsilon, \ldots, \epsilon \rangle \rangle$ is an initial configuration;*
- *$\mathcal{A} = \bigcup_i \mathcal{A}^i$ is a set of actions;*
- *$\mathcal{T} \subseteq \mathit{\Gamma} \times \mathcal{A} \times \mathit{\Gamma}$ is the global transition relation.*
  A transition $(\langle \bar{s}, C \rangle, a, \langle \bar{s}', C' \rangle)$ is in $\mathcal{T}$, if for some $1 \leq i \leq n$, $\bar{s}' = \bar{s}[s'_i/s_i]$, and $(s_i, a, s'_i) \in \mathcal{T}^i$, and one of the following holds:
  - *$a = \overrightarrow{\mu} \ \wedge \ C' = C.\mu \ \wedge \ |C'| \leq \mathcal{L}_B$;*
  - *$a = \overleftarrow{\mu} \ \wedge \ C = \mu.C'$;*
  - *$a = \tau \ \wedge \ C' = C$.*

In the following, we say that the channel of a composition have a *bounded growth* if, for each queue q_i, either a finite bound $b_i < \infty$ is declared, or there is some constant K_i such that the queue contains at most K_i messages in all reachable states. The composition is *bounded* if it has bounded growth.

We say that the composition is *complete* if all the terminating configurations $\langle \bar{s}, C \rangle$ have empty queue content: $C = \langle \epsilon, \ldots, \epsilon \rangle$. We remark that systems that are not complete lose messages: indeed, at the end of the computation there are unconsumed messages in queues.

4.2 Behavior of the Local Model

While the emission and the reception of the messages are indistinguishable in the global protocol, this is not the case for the composition of projections. Depending on the communication applied, the messages may be interleaved, reordered, and even ignored. Moreover, the structure of the composition configuration is more complex than that of the global protocol, and depends on the communication model. This is reflected in the way behaviors and conversations are defined. In particular, since the reception of a message does not necessarily follow the emission (the message may be lost), we will define a conversation set in two ways: we denote as $\overrightarrow{\rho}(\pi) = \mu_1, \mu_2, \ldots$ the sequence of all the messages emitted on the sequence π, and as $\overleftarrow{\rho}(\pi)$ the sequence of received messages.

An action $a \in \mathcal{A}$ is *fireable* in γ, denoted as $\gamma \xrightarrow{a} \gamma'$, if there is a transition $(\gamma, a, \gamma') \in \mathcal{T}$. Let $\omega = \langle \bar{s}_1, C_1 \rangle, a_1, \langle \bar{s}_2, C_2 \rangle, a_2, \ldots$ be a (possibly infinite) sequence of configurations and actions. We say that the sequence is fireable from $\langle \bar{s}_1, C_1 \rangle$, written as $\langle \bar{s}_1, C_1 \rangle \xrightarrow{\omega}^*$, if $\forall k \geq 1$, $\langle \bar{s}_k, C_k \rangle \xrightarrow{a_k} \langle \bar{s}_{k+1}, C_{k+1} \rangle$. We denote as $\pi(\omega) = \bar{s}_1, a_1, \bar{s}_2, a_2, \ldots$ a corresponding sequence of states and actions.

Definition 6. *Let $\Sigma_\Delta = \langle \mathit{\Gamma}, \gamma_0, \mathcal{A}, \mathcal{T} \rangle$ be a composition of local protocols. Given a sequence $\omega = \gamma_0, a_0, \gamma_1, a_1$ such that $\gamma_0 \xrightarrow{\omega}^*$, a* run *of the composition is a sequence $\pi(\omega)$. The* behavior *of the composition is the set of all the runs of the composition:* $\mathrm{B} = \{\pi(\omega) \mid \gamma_0 \xrightarrow{\omega}^*\}$*. An* output conversation *(respectively* input conversation*) of the composition is a sequence of messages emitted (resp. received) on some $\pi \in \mathrm{B}$. The set of all the output (resp. input) conversations is called the* output conversation set $\overrightarrow{\Omega}$ *(resp.* input conversation set $\overleftarrow{\Omega}$*).*

4.3 Communication Models

The behavior of the composition depends on the communication model. Indeed, the certain ordering should be satisfied for the message to be consumed from the queue, the queue bound should restrict the emission of new messages, etc. One of the key problems for the analysis of the behavior of the protocol is to determine the relations between the behaviors exhibited by different implementations of the communication medium. This requires the introduction of certain relations between communication models, namely *simulation* relations.

Definition 7. *We say that a configuration* $\gamma_2 = \langle \bar{s}_2, C_2 \rangle$ *of* Σ_{Δ_2} simulates *a configuration* $\gamma_1 = \langle \bar{s}_1, C_1 \rangle$ *of* Σ_{Δ_1}*, written as* $\gamma_1 \preceq \gamma_2$*, iff*

- $\bar{s}_1 = \bar{s}_2$,
- $\forall\ a,\ \forall\ \gamma_1'$*, if* $\gamma_1 \xrightarrow{a} \gamma_1'$*, then* $\exists\ \gamma_2'$*, s.t.* $\gamma_2 \xrightarrow{a} \gamma_2'$*, and* $\gamma_1' \preceq \gamma_2'$.

We write $\Sigma_{\Delta_1} \preceq \Sigma_{\Delta_2}$ *to denote that* $\gamma_{01} \preceq \gamma_{02}$.
We also write $\Sigma_{\Delta_1} \approx \Sigma_{\Delta_2}$ *when* $\Sigma_{\Delta_1} \preceq \Sigma_{\Delta_2}\ \wedge\ \Sigma_{\Delta_2} \preceq \Sigma_{\Delta_1}$.

Proposition 1. $\Sigma_{\Delta_1} \approx \Sigma_{\Delta_2}$ *iff* $\mathrm{B}(\Sigma_{\Delta_1}) = \mathrm{B}(\Sigma_{\Delta_2})$.

When the simulation relation among two communication models Δ_1 and Δ_2 holds for any set of local protocols, we say that Δ_2 is *more general* than Δ_1.

Definition 8. *Communication model* Δ_2 simulates *model* Δ_1*, written as* $\Delta_1 \sqsubseteq \Delta_2$*, if for any composition of STSs,* $\Sigma_{\Delta_1} \preceq \Sigma_{\Delta_2}$.

Being reflexive and transitive, this relation forms a partial order on the set of communication models. Below we will show that there is a "most general" model, that is the model Δ_{MG}, such that for any other model Δ holds $\Delta \sqsubseteq \Delta_{MG}$.

The relation among communication models relies on the structure of the queues. The models differ in two dimensions. First, the relation depends on the queue bounds: the bigger a bound is, the more transitions are enabled. Second, it depends on the distribution of the message alphabets: if the alphabet of each ordered queue in one model is a subset of the alphabet of some ordered queue in another model, then the first model is more general than the other. The following theorem defines a relation between the models with different queue structures.

Theorem 1. *Consider two communication models* $\Delta_1 = \langle \mathcal{L}_{1B}, \mathcal{L}_{1M}, \mathcal{L}_{1O} \rangle$ *and* $\Delta_2 = \langle \mathcal{L}_{2B}, \mathcal{L}_{2M}, \mathcal{L}_{2O} \rangle$*. If for each queue* q_{2i} *holds that*

- *if the queue* q_{2i} *is ordered, then there exists an ordered queue* q_{1j}*, s.t.* $\mathrm{M}_{2i} \subseteq \mathrm{M}_{1j}$*, and*
- $b_{2i} \geq \sum_{\mathrm{M}_{2i} \cap \mathrm{M}_{1j} \neq \emptyset} b_{1j}$,

then $\Delta_1 \sqsubseteq \Delta_2$.

Let us define the *most general* model, that is the model that allows for the largest set of behaviors. In order to respect the assumptions presented above, this model has to allow for potentially unbounded queues, non-blocking emissions, and arbitrary, unordered access to the content of any queue.

Definition 9. *The* Most General Communication Model *(MG-model) is a communication model $\Delta_{MG} = \langle \mathcal{L}_B, \mathcal{L}_M, \mathcal{L}_O \rangle$, with 1 unordered queue, $b = \infty$, and $\mathcal{L}_M(\alpha_i) = 1$.*

It is easy to see that such a model is indeed a generalization of any other communication model w.r.t. the behavior of any composition of STSs.

Proposition 2 (from [4]). *For any communication model Δ, $\Delta \sqsubseteq \Delta_{MG}$.*

Whenever a composition under a certain model Δ simulates the most general composition, we say that this model is *adequate* for the description of the composition scenario.

Definition 10 (from [4]). *A communication model Δ is said to be* adequate *for the given composition scenario if $\Sigma_\Delta \approx \Sigma_{\Delta_{MG}}$.*

5 Protocol Realizability

A natural question that comes with the specification of the global choreography protocol is the *realizability* of the specification. It consists in deciding whether there is a way to extract the local implementations of the participating roles such that, when composed together, they satisfy the protocol specifications. We refer to these local implementations as *projections.* Sets of such projections will be used as the implementation candidates for the composition.

Intuitively, a behavior of a projection is constructed from those transitions of the global protocol, in which the participant is involved. The internal action $(\mathcal{R}_\tau, \tau)$ is projected onto an internal action of each role $r_i \in \mathcal{R}_\tau$. The interaction action (r_s, r_d, μ) is projected onto the input action $\overleftarrow{\mu}$ of the role r_d, and on the output action $\overrightarrow{\mu}$ of the role r_s.

Definition 11 (Role Projection). *Given a global protocol $\langle \mathcal{R}, \mathcal{S}, \bar{s}_0, \mathcal{A}, \mathcal{T} \rangle$, a* projection *on role r_i is a local protocol $\langle \mathcal{S}_i, s_{0i}, \mathcal{A}_i, \mathcal{T}_i \rangle$, where:*

- *$\mathcal{S}_i$ is a set of states, and s_{0i} is the initial state of r_i respectively;*
- *$\mathcal{A}_i = \mathcal{I}_i \cup \mathcal{O}_i \cup \{\tau\}$ is a set of actions of r_i;*
- *$\mathcal{T}_i \subseteq \mathcal{S}_i \times \mathcal{A}_i \times \mathcal{S}_i$ is the transition relation of r_i. Transition $(s_i, a_i, s'_i) \in \mathcal{T}_i$, if $\exists$ $(\bar{s}, a, \bar{s}') \in \mathcal{T}$, such that one of the following holds:*
  - *$a = (r_i, r_j, \mu)$, $a_i = \overrightarrow{\mu} \in \mathcal{O}_i$;*
  - *$a = (r_j, r_i, \mu)$, $a_i = \overleftarrow{\mu} \in \mathcal{I}_i$;*
  - *$a = (\mathcal{R}_\tau, \tau)$, $r_i \in \mathcal{R}_\tau$, $a_i = \tau$.*

Given the set of all role projections of a global protocol, we can compose them into a CTS according to Definition 5. In the following we denote the composition of role projections under the model Δ as Σ^p_Δ.

This composition enables the analysis of the realizability of the global protocol. In order to perform this analysis, a possibility to compare the behavior of the global protocol with the behavior of the composition of the local projections is needed. This comparison is based on the notion of the behavior expansion that re-defines the behavior of the protocol in terms of the actions of the composition.

Definition 12 (Behavior Expansion). *Given a run π of the global protocol P, the* expansion *of the run is a sequence $\overline{\pi}$ of states and composition actions obtained as follows:*

- *each interaction transition (r_i, r_j, μ) is projected onto a sequence of the corresponding send and receive transitions of r_i and r_j respectively;*
- *each internal transition $(\mathcal{R}_\tau, \tau)$ is projected onto a sequence of internal transitions of all roles in $\mathcal{R}_\tau$.*

The set of all expansions generated by $\mathrm{B}(P)$ *is denoted as* $\overline{\mathrm{B}}(P)$.

5.1 Synchronous Realizability

An intuitive candidate for realizability is the notion that requires that the composition of the projections behaves exactly as the given global specification regardless the communication model that is applied. We refer to this notion as *synchronous realizability.*

Definition 13 (Synchronous realizability). *The global protocol P is* synchronously realizable *if the behavior expansion of the protocol is equal to the behavior of the composition of the projections under Δ_{MG}:* $\overline{\mathrm{B}}(P) = \mathrm{B}(\Sigma^p_{\Delta_{MG}})$.

The notion of synchronous realizability is closely related to the *synchronizable communication model.* This is the most restricted communication model that can be defined in the local model formalism. In this model there is only one queue of capacity one.

Definition 14 (Synchronizable communications model). *The* synchronizable communication model *is the model $\Delta^1_1 = \langle \mathcal{L}_B, \mathcal{L}_M, \mathcal{L}_O \rangle$, with $\mathcal{L}_B = \langle 1 \rangle$ and $\mathcal{L}_M(\mu) = 1$ for all operations μ.*

The following result immediately follows from the above definitions.

Proposition 3. *If the protocol P is synchronously realizable, then the model Δ^1_1 is adequate for the composition of the local projections of P.*

Indeed, the synchronous realizability implies that the behavior of the composition under the MG-model is the same as that of the global protocol. Therefore, in any configuration there is at most one message to be received, and from the definition of composition follows that the model Δ^1_1 is also adequate.

It is easy to see that the protocol represented in Fig. 1 is synchronously realizable.

5.2 Strong Realizability

The restrictions imposed by the synchronous realizability are often too strong for the implemented system. Indeed, it requires that the order of internal and external actions is respected by the implementation, or that the next emission cannot start before the previous message was received, even if the acting participants are independent.

The notion of *strong realizability* relaxes these constraints. The ordering restrictions concern only the communication actions, and not the internal ones. Intuitively, a protocol is strongly realizable if the set of conversations of the protocol and of the compositions is the same, the composition is bounded, and all the emitted messages are received.

Definition 15 (Strong realizability). *Given a protocol P and the composition of the projections under Δ_{MG}, if the composition is bounded and $\Omega(P) = \overrightarrow{\Omega}(\Sigma^p_{\Delta_{MG}}) = \overleftarrow{\Omega}(\Sigma^p_{\Delta_{MG}})$, then the protocol is said to be* strongly synchronizable*.*

This notion is related to the *globally ordered communication model*.

Definition 16 (Globally ordered communications model). *The* globally ordered communication model *is the model $\Delta_{go} = \langle \mathcal{L}_B, \mathcal{L}_M, \mathcal{L}_O \rangle$, with one ordered queue, such that $\mathcal{L}_B = \langle \infty \rangle$ and $\mathcal{L}_M(\mu) = 1$ for all operations μ.*

Proposition 4. *The protocol P is strongly realizable iff Δ_{go} is adequate for the composition of the projections Σ^p_{Δ}, and the composition is complete and bounded.*

The completeness and adequacy of Δ_{go} immediately follows from strong realizability of P by the above definitions ($\overrightarrow{\Omega}(\Sigma^p_{\Delta_{MG}}) = \overleftarrow{\Omega}(\Sigma^p_{\Delta_{MG}})$). To see the converse, note that only one participant may send and only one may receive messages at a time (otherwise the reordering would be possible). A sequence of emissions of a participant is projected from the definition of the protocol, and therefore, reflects the sequence of interactions. Thus, $\Omega(P) = \overrightarrow{\Omega}(\Sigma^p_{\Delta_{MG}})$.

While the protocol represented in Fig. 2(a) is not synchronously realizable (the internal activity `prepareToDelivery` may be performed before the shipper receives the `orderDelivery` message), one can see that it satisfies the strong realizability requirements. Indeed, the composition is bounded, all the messages are eventually received, and there are no concurrent message emissions/receptions, which makes the Δ_{go} adequate for the composition.

5.3 Local Realizability

Strong realizability may appear to be too restrictive for a wide class of choreography scenarios. Indeed, it does not allow for concurrent emissions and receptions of messages by independent processes. For instance, if role A interacts with B, and then C interacts with D, then the global order cannot be preserved by the composition of projections. In many cases, however, this may be irrelevant. Moreover, since the variables are local for each role, the behavior and the information of the local participant is not affected by these reorderings.

We relax the notion of strong realizability by omitting the requirement of conversation equivalence. However, the local behavior of each role should not be affected by possible reorderings of message emissions.

Definition 17 (Local Realizability). *The protocol P is* locally realizable *if the composition $\Sigma^p_{\Delta_{MG}}$ is complete, bounded, and for any run $\pi \in \mathrm{B}(\Sigma^p_{\Delta_{MG}})$, for any role r_i, $\overrightarrow{\pi_i} = \overleftarrow{\pi_i}$, where $\overrightarrow{\pi_i}$ and $\overleftarrow{\pi_i}$ are the sequences of messages sent to and received by the i^{th} participant.*

This property is especially important for monitoring, since it guarantees that the external observed order of messages is locally respected by the receiver.

Definition 18. *A* locally ordered communication model *for the composition of n local protocols is a model $\Delta_{lo} = \langle \mathcal{L}_B, \mathcal{L}_M, \mathcal{L}_O \rangle$, with n ordered queues, $b_i = \infty$, and $\forall\ \alpha$ s.t. $\overleftarrow{\alpha} \in \mathcal{I}^i$, $\mathcal{L}_M(\alpha) = q_i$.*

This communication model, exploited also in [7], requires that messages are queued on a process-by-process way. The following result immediately follows from the definition of the local realizability.

Proposition 5. *The protocol P is locally realizable iff Δ_{lo} is adequate for the composition of the projections Σ^p_{Δ}, and the composition is complete and bounded.*

One can see that the global protocol presented in Fig. 2(b) is locally realizable. While the reordering of messages is possible in the composition (thus violating the strong realizability requirements), the local order of messages is respected.

5.4 Weak Realizability

The least restrictive model of realizability further relaxes the ordering constraints, requiring only that the message ordering of the interactions among a pair of the participants is preserved. That is, each participant sending messages to its partner knows that they will be processed and managed in turn. We refer to this notion of realizability as *weak realizability.*

Definition 19 (Weak realizability). *The protocol P is* weakly realizable *if the composition $\Sigma^p_{\Delta_{MG}}$ is complete, bounded, and for any run $\pi \in \mathrm{B}(\Sigma^p_{\Delta_{MG}})$, for any pair of roles r_i and r_j, $\overrightarrow{\pi_{ji}} = \overleftarrow{\pi_{ij}}$, where $\overrightarrow{\pi_{ji}}$ and $\overleftarrow{\pi_{ij}}$ are the sequences of messages sent from r_j to r_i and received by r_i from r_j respectively.*

This definition is related to the *mutually ordered communication model.*

Definition 20 (Mutually ordered communication model). *A* mutually ordered asynchronous communication model *is a model $\Delta_{mo} = \langle \mathcal{L}_B, \mathcal{L}_M, \mathcal{L}_O \rangle$, with $n^2 - n$ ordered queues denoted as $q_{i,j}$ $(i \neq j)$ s.t. $b_{i,j} = \infty$, and $\forall\ \alpha$, $\overleftarrow{\alpha} \in \mathcal{I}^j \wedge \overrightarrow{\alpha} \in \mathcal{O}^j$ iff $\mathcal{L}_M(\alpha) = q_{i,j}$.*

In this model, a pair of queues is defined for each pair of processes, with each queue representing one direction of interaction between these processes. This model, described in [8], provides a natural representation of communicating processes since each process explicitly distinguishes each of its partners. The main feature of this model is that each pair of communicating processes preserves the order of partners' events. In other words, the order of receptions is equivalent for each pair of processes.

Proposition 6. *The protocol P is weakly realizable iff Δ_{mo} is adequate for the composition of the projections Σ^p_Δ, and the composition is complete and bounded.*

The global protocol represented in Fig. 2(c) is weakly realizable, while the one in Fig. 2(d) is not. Indeed, the composition of the local projections is incomplete and unbounded, and, moreover, it allows for reorderings of updated quotes.

6 Realizability Analysis

The notions of realizability and the respective properties suggest an analysis approach based on the analysis of communication models [4]. It is easy to see that the following hierarchy holds for the models presented above:

$$\Delta^1_1 \sqsubseteq \Delta_{go} \sqsubseteq \Delta_{lo} \sqsubseteq \Delta_{mo} \sqsubseteq \Delta_{MG}$$

Moreover, an analogous hierarchy holds also for the notions of realizability[2]. This allows for the application of the following analysis algorithm.

1. Find a minimal (w.r.t. simulation relation) adequate communication model Δ for the composition of the local projections of the protocol P.
2. Check that the composition is complete and has bounded growth. If this is not the case, the protocol is not realizable.
3. The appropriate level of realizability is determined by the corresponding communication model.

The algorithm for the adequacy check is presented in Fig. 1. We briefly describe its behavior. The reachability tree of the composition is traversed recursively, starting from the initial configuration. In each state the set of enabled transitions is compared with the set of transitions enabled in the corresponding configuration of the composition under the MG-model. If the sets are not equivalent, the current model is not adequate.

Each newly reached configuration is checked for boundedness (the function $isUnbounded(\gamma')$). This is performed by checking whether there exists a loop starting in a configuration with the same state, but with greater queue content. If such a loop exists, the composition is unbounded. The completeness is checked for each terminating state, i.e. a state without fireable transitions.

We remark, that whenever the inadequacy of a certain model is detected (as well as the incompleteness or the unboundedness), the sequence of configurations contained in the search stack represents a counterexample (or a witness), that describes the violation of the analyzed realizability. This counterexample may be used for further analysis of the protocol.

For the sake of simplicity we omitted the definition of variables and the data flow in the global and local models. In [9] we show how the above formalisations and analysis techniques may be extended in order to capture both the

[2] Strong realizability, however, requires an additional check that the internal actions are not interleaved with the message receptions. This check may be easily introduced in the presented algorithm.

Algorithm 1 Composition adequacy check

```
$Stack := nil$;   {Stack of configurations}
$Visited := nil$;   {Set of all visited configurations}
$IS := nil$;   {Set of incomplete configurations}
$US := nil$;   {Set of configurations, where unboundedness is detected}

$explore(\gamma_0)$;
procedure explore($\gamma$)
$push(\gamma, Stack)$;
$Fireable := out(\gamma)$; {fireable transitions}
if $Fireable \neq out_{MG}(\gamma)$ then terminate; {the model is not adequate}
if $Fireable \neq \emptyset$ then
   forall $trans \in Fireable$ do
      $\gamma' := trans.target$;
      if $\gamma' \notin Stack \cup Visited$ then {check boundedness}
         if $isUnbounded(\gamma')$ then $US := US \cup \{\gamma'\}$;
         else $explore(\gamma')$;
else if $\neg complete(\gamma)$ then $IS := IS \cup \{\gamma\}$;
$Visited := Visited \cup \{\gamma\}$;
$pop(\gamma, Stack)$;
end procedure
```

control and data in the composition representation. In particular, we show that the boundedness and adequacy results obtained on the data-less model may be propagated to the full model (under certain conditions). Moreover, using the abstraction techniques the representation may be made finite, enabling the model checking techniques as presented in [4].

7 Related Work and Conclusions

In this paper we presented a formal framework for the realizability analysis of Web service choreographies. The framework is based on the formalism suitable for modelling a composition both as a global protocol and as a set of interacting local services. For the global protocol we exploit a simple model, which is based on the notion of state transition systems, and allows for the automatic extraction of local projections that can be used as the basis for the service implementations. A composition of local services is formalized using the model of [4]. The key feature of this approach is the ability to specify for the composition a communication mechanism with an arbitrary structure. The most relevant original contribution of this paper is the definition of a hierarchy of notions of choreography realizability, which allows for capturing a wide range of properties of Web service choreographies. We also defined a correspondence between this hierarchy and the hierarchy of communication models presented in [4]. This correspondence allows us to exploit the analysis approach described in [4] for determining the level of realizability of the given global protocol specification.

The problem of the realizability of a global Web service protocol specification has been addressed in [5], while the realizability is defined as the ability of the composition of protocol projections to produce the same set of conversations. Several necessary conditions are formalized in order to enable the protocol realizability analysis. Contrary to our framework, the formalization of [5] is glued to a particular communication mechanism — the synchronous one — thus making the analysis results more restrictive. The framework presented here is also more flexible with respect to a notion of realizability.

The formalization and the analysis of the Web service choreography models are presented also in [6, 10, 11], while approaches to generate implementation templates from the choreography models are presented in [12, 13]. However, the problem of realizability is not covered in these works, and the composition formalization is based on synchronous communications assumptions, which for a large range of scenarios and systems is not realistic.

References

1. Andrews, T., Curbera, F., Dolakia, H., Goland, J., Klein, J., Leymann, F., Liu, K., Roller, D., Smith, D., Thatte, S., Trickovic, I., Weeravarana, S.: Business Process Execution Language for Web Services (version 1.1) (2003)
2. OMG: Business Process Modeling Language (BPML). (2005) [http://www.bpmi.org].
3. W3C: Web Services Choreography Description Language Version 1.0. (2005) [http://www.w3.org/TR/ws-cdl-10/].
4. Kazhamiakin, R., Pistore, M., Santuari, L.: Analysis of Communication Models in Web Service Compositions. In: Proc. WWW'06. (2006)
5. Fu, X., Bultan, T., Su, J.: Conversation protocols: a formalism for specification and verification of reactive electronic services. Theor. Comput. Sci. **328** (2004) 19–37
6. Busi, N., Gorrieri, R., Guidi, C., Lucchi, R., Zavattaro, G.: Choreography and Orchestration: A Synergic Approach for System Design. In: Proc. ICSOC'05. (2005)
7. Fu, X., Bultan, T., Su, J.: Analysis of Interacting BPEL Web Services. In: Proc. WWW'04. (2004)
8. Brand, D., Zafiropulo, P.: On communicating finite-state machines. J. ACM **30** (1983) 323–342
9. Kazhamiakin, R., Pistore, M.: Static Verification of Control and Data in Web Service Compositions. In: Proc. ICWS '06. (2006)
10. Brogi, A., Canal, C., Pimentel, E., Vallecillo, A.: Formalizing Web Services Choreographies. In: Proc. WS-FM'04. (2004)
11. Foster, H., Uchitel, S., Magee, J., Kramer, J.: Model-Based Analysis of Obligations in Web Service Choreography. In: Proc. AICT-ICIW'06. (2006)
12. Mendling, J., Hafner, M.: From Inter-Organizational Workflows to Process Execution: Generating BPEL from WS-CDL. In: Proc. OTM'05. (2005)
13. Bravetti, M., Guidi, C., Lucchi, R., Zavattaro, G.: Supporting e-commerce systems formalization with choreography languages. In: Proc. SAC '05. (2005)