An Interaction-Based Approach to Semantic Alignment

Manuel Atencia[a,b,*], Marco Schorlemmer[c,d]

[a] INRIA, 655 avenue de l'Europe, Montbonnot, 38334 Saint Ismier Cedex, France
[b] University of Grenoble, Grenoble, France
[c] Artificial Intelligence Research Institute, IIIA–CSIC, Campus de la UAB, 08193 Bellaterra, Catalonia, Spain
[d] Universitat Autònoma de Barcelona, Bellaterra, Catalonia, Spain

Abstract

We tackle the problem of semantic heterogeneity in the context of agent communication and argue that solutions based solely on ontologies and ontology matching do not capture adequately the richness of semantics as it arises in dynamic and open multiagent systems.

Current solutions to the semantic heterogeneity problem in distributed systems usually do not address the contextual nuances of the interaction underlying an agent communication. The meaning an agent attaches to its utterances is, in our view, very relative to the particular dialogue in which it may be engaged, and that the interaction model specifying its dialogical structure and its unfolding should not be left out of the semantic alignment mechanism.

In this article we provide the formal foundation of a novel, interaction-based approach to semantic alignment, drawing from a mathematical construct inspired from category theory that we call the communication product. In addition, we describe a simple alignment protocol which, combined with a probabilistic matching mechanism, endows an agent with the capacity of bootstrapping —by repeated successful interaction— the basic semantic relationship between its local vocabulary and that of another agent.

We have also implemented the alignment technique based on this approach and prove its viability by means of an abstract experimentation and a thorough statistical analysis.

Keywords: semantic alignment, agent interaction context, interaction model, communication product, alignment protocol, matching criteria

1. Introduction

The Semantic Web was envisioned, at the turn of the century, as an extension of the Web "in which information is given well-defined meaning, better enabling computers and people to work in cooperation" [1]. A key concept playing a crucial role in this vision is that of *ontologies*: documents or files that formally define vocabularies of terms and the relations amongst these terms. An important effort has gone into providing formal foundations for ontology engineering, deployment, and maintenance, both from the logical and the computational point of view [2]. As a result, much research has focussed on the computationally tractable, logic-based representation formalisms that could provide a well-defined, model-theoretic semantics to carry out inferences and drawing conclusions on top of the standard web infrastructure, and thus supporting this improved "work in cooperation" effectively and efficiently [3, 4].

This view of cooperation on the Web takes "well-defined meaning" via ontologies as a prerequisite for successful interaction. By adopting this stance, meaningful communication between, for instance, separately engineered software agents in a multiagent system relies on an a priori commitment to a shared conceptualisation of the application domain as to guarantee a shared understanding of the terms being communicated [5, 6]. Ontologies may indeed be useful for specifying such a shared conceptualisation when dealing with stable domains and closed communities of agents. But often it is impossible to reach global semantic agreements because the cost of being precise about semantics and guaranteeing this precision at a global level soon increases very quickly when the number of agents grows [7, 8].

*Corresponding author. Telephone: +33 476 61 53 55; fax: +33 476 61 52 07.
Email addresses: `manuel.atencia@inria.fr` (Manuel Atencia), `marco@iiia.csic.es` (Marco Schorlemmer)

1.1. *Semantic Heterogeneity in the Context of Multiagent Interaction*

As a result, most state-of-the-art approaches that tackle semantic heterogeneity not only seek to agree on shared global ontologies, but also attempt to find semantic correspondences between varying terminologies through ontology matching [9, 10]. We argue, however, that by computing semantic correspondences of separate terminologies focusing on ontologies and ontology matching, the problem is only partially addressed. Already back in the mid 1980s, Winograd and Flores recalled that, according to modern hermeneutics, language is listened to in a background, and that interpretation is not independent of the interpreter [11]. Meaning, they stressed, is always re-created in the context of the intentions, purposes, expectations, and commitments the interpreter attaches to a particular utterance. Consequently, meaning is ultimately interaction-dependent and relative to an implicit background that cannot be fully de-contextualised.

Despite that, most state-of-the-art ontology matching systems compute semantic alignments generally prior to and independently of interaction. Moreover, most matching techniques follow a classical functional approach according to which two or more ontologies are taken as input and a semantic alignment of ontological entities is generated as output, as part of a larger ontology-alignment life cycle [10]. Nonetheless, this approach has several drawbacks. On the one hand, it limits the dynamism and openness of the interaction, as only agents with previously matched ontologies —even if only partially— may jointly participate in an interaction. On the other hand, it keeps matching outside the context of interaction. Semantic correspondences are established by ontology matchers in an interaction-independent fashion, for instance by exploiting the internal structure of ontologies and ontological entities [12, 13] or by resorting to external sources where semantic relations were determined prior to interaction and independently from it (e.g. when using upper-level ontologies [14], lexical databases [15], or background knowledge available in the Semantic Web itself [16]).

Although recent approaches aim at approximating an ontology alignment in the context of open and dynamic multiagent systems [17] by using argumentation [18, 19] and targeting only certain ontologies fragments [20] or taking the task at hand into account [21] —thus allowing for increased openness and dynamism — such ontology matching techniques still fall back on a classical approach: when a mismatch occurs, semantic heterogeneity is solved by some sort of functional ontology matcher, albeit using only relevant fragments of the ontologies and subject to further argumentation or negotiation of the alignment. Furthermore, although done at run-time and task-oriented, matching is still carried out outside the context given by the interaction.

1.2. *Taking Interaction as Ontologically Prior to Meaning*

In this paper we investigate how software agents can establish semantic relationships between their respective terminologies on the grounds of their communication in the context of a specific interaction. We do that *not* by taking well-defined meaning as a prerequisite of successful interaction, but instead by taking successful interaction as a prerequisite for shared understanding.

Take, for example, two agents engaged in a simple bargaining dialogue. At the initial state of the dialogue the buyer may ask the seller how much a certain good costs, to which the seller may answer providing a price. In the context of a bargaining interaction, at this particular state of the dialogue, the price uttered by the seller is not an actual offer but just a starting point for the bargaining dialogue to unfold. The answer given by the seller does not have the same meaning in the context of this kind of dialogue and at this particular dialogue state as if the buyer would have asked for the price in a hotel store, for instance, where articles have a fixed price. At this state of the bargaining dialogue the seller does not expect the buyer to utter an acceptance of the price, and in certain cases not even to utter a refusal to continue bargaining. He or she waits for a counteroffer from the buyer. After this interaction state, however, the prices uttered by the buyer or seller *do* stand for genuine offers that are subject to be accepted or not.

Hence, the meaning of terms uttered by an agent ultimately arises when it actually *makes use* of these terms in the context of an interaction. We shall assume, thus, that agents are part of a regulatory environment and follow interaction models, or protocols, that regulate the utterances allowed or expected at particular dialogue states. One means to specify such regulatory environments for multiagent systems, for instance, are electronic institutions, which fix the performative structures of dialogues and the shared ontology of the content language used in the utterances effectuated by agents, and which take the form of illocutionary speech acts [22].

In the case that agents do *not* share the understanding of the content language, we argue that the regulatory environment in which the agents unfold their dialogue may provide the grounds for establishing the semantic relationships between varying local terminologies. Take again our simple bargaining dialogue and put yourself in the role of the seller. If the buyer, after the utterance of the starting price, had answered with an expression that you did not understand, you would have to guess among the alternatives for possible answers from the buyer regarding your own view of the interaction, the view of a seller, assuming that the buyer considers the dialogue to be in the same state. That is, you might take the foreign expression uttered by the buyer as a counteroffer, hence establishing a semantic relationship between the expression you did not know and the expression you were expecting. You were listening to an utterance made by the buyer in the background of a particular dialogue state. If the buyer now walks away and does not continue bargaining you realise that he or she considered to have reached a final state in the bargaining dialogue —maybe because he or she uttered a refusal to accept the price you said and to continue bargaining— while you considered the dialogue not to have finished yet. It is this unsuccessful interaction between you, the seller, and the buyer which indicates that the semantic alignment of the foreign expression you did not understand with the expression of a counteroffer was not correct. If the dialogue, however, had reached a final state for both participants in the bargaining interaction, it would be evidence for a correct semantic alignment. This is what turns the successful interaction into a prerequisite for shared understanding.

It is the assumption that agents repeatedly engage in dialogues following a fixed performative structure and that they share a notion of success of the dialogue (for example when reaching a final state) which allows agents to discover the semantic relationship of their vocabularies. Semantics is in this view closely tied not only to the illocutions allowed to be uttered at any particular state of the interaction, but also to the notion of success: two agents have understood each other if each one considers that the dialogue has been completed successfully (such situations should strengthen the semantic alignment choices made during interaction).

1.3. Interaction-Situated Semantic Alignment

In this paper we have set out to investigate to what extent two agents are capable of aligning their respective local vocabularies without accessing ontologies of foreign agents — assuming that the only way an agent has access to the vocabulary of another agent's ontology is by being aware of its utterance in the course of an interaction— nor resorting to any shared ontology or external source that may guide them in establishing semantic relationships. Instead we rely entirely on the context provided by the interaction protocol and the concrete interaction states at which a term is uttered or listened to.

For this we define a means of interaction by which agents follow their own interaction protocol and also an *alignment protocol* in parallel. As a start we have focussed on two-agent protocols as represented by finite-state machines, because this formalism underlies most dialogue representation frameworks such as the aforesaid electronic institutions. The alignment protocol acts as a meta-protocol through which the actual communication is carried out: any utterance that would have been a speech act at the object-level communication becomes ineffective and has an effective counterpart at the meta-level. Additionally, we endow agents with a matching mechanism that they use to perform the actual matching. Matching elements are strengthened as many interactions are completed, and this strengthening is based on statistical reasoning. Eventually, expressions uttered and listened to by agents are deemed semantically related if they trigger compatible interaction state transitions. As content language we have initially constrained ourselves to a propositional language.

As notion of success we have initially explored a very obvious one, namely that of reaching a final state in the dialogue. This, of course, can be made more complex, for example by paying attention to commitments taken during the course of a dialogue and their posterior fulfilment or not. This, in turn, enriches the semantics implicit in the performative structure of a dialogue.

The initial ideas and a preliminary formalisation of the approach set forth in this paper were published in [23] and a first set experimentation results were presented in [24]. In this paper, however, we present the comprehensive description and experimental validation of our approach. For this we have taken a web-based reservation scenario, described in Section 2, as running example with which to illustrate the key insights of our alignment technique, which we put forth in Section 3. Section 4 presents the complete formalisation, whereas Section 5 describes the alignment dynamics, discussing several alternative matching criteria not tackled in our initial work.

The theoretical model is not left by itself, and we have carried out an implementation of the model, showing empirically in Section 6 its effectiveness in establishing the semantic alignment that arises in the context of an interaction. For this we have generated interaction protocols of varying complexity in terms of interaction states and interaction state transitions, and we have let agents repeatedly interact according to the dialogical structure specified in these protocols to see how they are capable of bootstrapping a basic alignment of their vocabularies —of different size and complexity— and hence to improve their success rate while interacting. This confirms and adds to our initial results. We conclude in Section 8 with some references to related ideas and a discussion on future research directions.

2. A Running Example: Online Travel Reservation

Imagine a travel agency that offers facilities to make reservations of flights and hotels. Consider also that this travel agency is up to date with the new Semantic Web technologies and delegates to a software agent the task of making reservations. This software agent is thus programmed to interact with customers —whether they are human or software agents— and satisfy all their requests. We will particularly study a scenario where two software agents, one as travel agent and the other as customer, participate in a travel reservation interaction.

2.1. Interaction Models for the Travel Reservation Scenario

Agent interactions can be specified by means of finite state automata, which is the formalism that we will be using in this paper. This is the way, for instance, in which particular *scenes* (bounded scopes of interactions) are specified for electronic institutions [22]. Figure 1 depicts the message-passing behaviour of a customer and a travel agent in a travel reservation scenario. Transitions between states are labelled with *illocution* schemata containing variables written in uppercase letters. Illocutions are tuples the components of which are an illocutionary particle, the identifier of the sender together with the role it is playing, the identifier of the receiver together with the role it is playing, and the content of the message uttered. The latter is expressed in some language whose vocabulary is defined in a particular ontology. During an interaction, the variables in illocution schemata are bound to the values of the uttered illocutions. Variables get their values in those illocutions in which they occur preceded by a question mark (?), and these values are subsequently used in those illocutions in which the corresponding variable occurs preceded by an exclamation mark (!). We call this kind of automaton an *interaction model*. Certainly, there exist other elements to be considered when specifying agent interactions (e.g. time stamps), but this simplified model is adequate for the purpose of this work.

According to Figure 1, at the initial state s_{00}, the customer is supposed to send a message to the travel agent requesting either a flight (illocution i_1) or an accommodation (i_2). Each choice triggers a different interaction. Assume that the customer asks the travel agent to book a flight, in our terms, the customer sends the following illocution:

$$i_1 = \langle \mathit{request}, (a : \mathit{customer}), (b : \mathit{travel\_agent}), \texttt{flight} \rangle$$

where a and b identify the customer and travel agent, respectively.

A flight trip may be a return trip or a single trip, and this is something the customer must be specific about first (via illocutions i_3 and i_4). Once it is done, the customer is supposed to send some required information about the desired flight. This information depends on the previous choice. Indeed, if the customer asks for a single flight, she only needs to provide information about the departure, but if she asks for a return flight then she needs to give information about the return too. This information is of a varying nature: origin and destination (i_5, i_6), dates (i_7, i_9), times (i_8, i_{10}), and number of passengers (i_{11}).

In ontological terms, the travel agent is supposed to build a concept description, e.g., in a description logic, of the flight the customer is looking for with this information; something like

$$\begin{aligned}
\mathsf{Flight} \quad \sqcap \quad & \mathsf{Return} \sqcap \\
& \mathsf{origin} : \text{Lyon} \sqcap \\
& \mathsf{destination} : \text{Barcelona} \sqcap \\
& \mathsf{departing} : \text{2011-06-10} \sqcap \\
& \mathsf{outboundTime} : \text{18:00} \sqcap \\
& \mathsf{returning} : \text{2011-06-14} \sqcap \\
& \mathsf{inboundTime} : \text{20:00} \sqcap \\
& \mathsf{numberOfPassengers} : 2
\end{aligned}$$

(if variable X in i_5 is grounded with 'Lyon', Y in i_6 with 'Barcelona', etc.).

Once a concept description is built, the travel agent is supposed to collect all instances that satisfy it. Instances may differ, for example, over the airline that operates the flight. Let $[a_1, \ldots, a_n]$ be the resulting list of instances (we can think of a_k as a URI which identifies a flight). The travel agent informs the customer of the result of the search through

$$i_{12} = \langle \mathit{inform}, (b : \mathit{travel\_agent}), (a : \mathit{customer}), \\ \texttt{results}([a_1, \ldots, a_n]) \rangle$$

One of the suggested flights may be to the customer's liking. If so, the customer is supposed to inform the travel agent of her choice (i_{13}). If not, she will request a new search (i_{14}). The remainder of the interaction involves informing of passenger details, either accepting or rejecting reservation terms, and committing to pay a reservation price, among others. The states s_{18} and s_{22}

are final states. If agents reach s_{18} then the reservation is unfinished, whereas if they reach s_{22}, it is completed.

Imagine, however, that the customer and the travel agent have different perspectives of the interaction and follow different interaction models. More specifically, assume that the customer follows the one depicted in Figure 1, while the travel agent the one in Figure 2. These interaction models differ in both transitions and ontologies. For instance, while the customer agent uses terms such as `return`, `single` or `accommodation`, the travel agent uses `roundTrip`, `oneWay` and `hotel`, instead. In what follows, we present the insights of our approach to semantic alignment.

3. I-SSA Insights

We consider a scenario in which two or more agents participate in an interaction following distinct interaction models. Agents may misunderstand each other because they do not share the same ontology, or expect to receive or send messages in different order. Our approach, called interaction-situated semantic alignment (I-SSA), looks at the semantics of messages exchanged during an interaction entirely from an interaction-specific point of view.

3.1. I-SSA Principles

The I-SSA approach is founded upon a number of principles. Here we give an informal account of these principles. A formalisation is provided in Section 4.

Principle 1. *Whether to match a foreign term with a local one depends on the particular interaction state where the former is received.*

This principle stresses the fact that, when an agent receives a message, this is received in a particular interaction state, and, regardless of the size of the agent's vocabulary, the foreign message is to be matched with one of the local messages that the agent expects to receive at that state. Imagine that the travel agent receives $\langle \mathit{request}, (a : \mathit{customer}), (b : \mathit{travel\_agent}), \texttt{flight} \rangle$ at the initial state t_{00}. According to her interaction model, the travel agent can only receive two messages at t_{00}: `flight` and `hotel`. As a consequence of Principle 1, travel agent's decision comes down to these two options.

Principle 2. *Whether to match a foreign term with a local one depends on the illocutionary force with which the former is uttered.*

Messages arise along with particles that inform of the illocutionary force of their utterance. These illocutionary particles are typically realised in terms of speech act verbs such as "request", "inform" or "commit", among others. Principle 2 states that, when matching two terms, their illocutionary particles must be the same. For example, if it happens that the travel agent receives $\langle \mathit{inform}, (a : \mathit{customer}), (b : \mathit{travel\_agent}), \texttt{choice} \rangle$ at t_{13}, according to Principle 2, `choice` cannot be matched with `newSearch`, since the latter can only arise within an illocution with *request* as illocutionary particle.

These first two principles state compatibility between illocutions and rule step-by-step matching decisions. However, only if agents succeed to interact, matching decisions prove to be valid, and matched terms to be semantically related. Principle 3 synthesises the reverse of this statement.

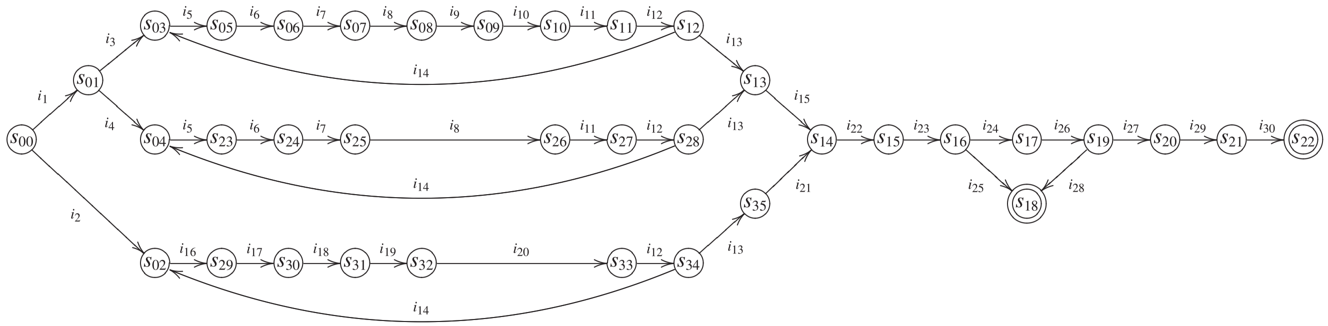
Principle 3. *Two terms are semantically unrelated if whenever they are matched agents interact unsuccessfully.*

The above principle highly depends on when an interaction is qualified as "successful". One possibility is to consider an interaction to be successful as long as agents eventually jointly reach final states. The following is thereby a more specific version of Principle 3.

Principle 3*. *Two terms are semantically unrelated if whenever they are matched agents do not eventually jointly reach final states.*

Let us imagine that the travel agent receives $\langle \mathit{request}, (a : \mathit{customer}), (b : \mathit{travel\_agent}), \texttt{flight} \rangle$ at the initial state t_{00}. Neither Principle 1 nor Principle 2 are helpful for the travel agent to decide whether to match `flight` with her local message `flight` or `hotel`. Only the subsequent interaction unfolding will show which matching decision leads agents to jointly reach final states.

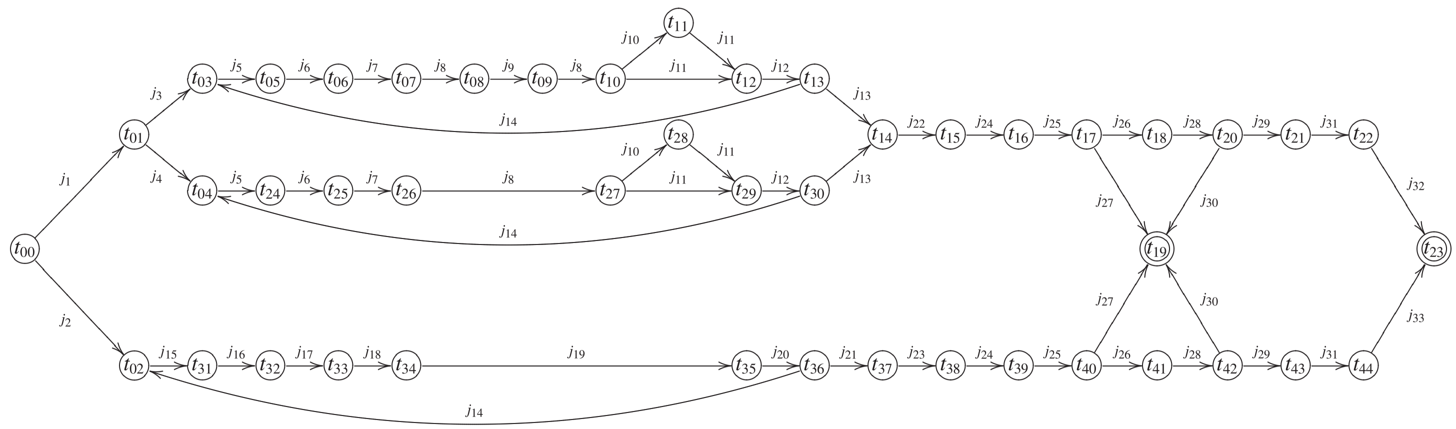
These three principles are the basis for an interaction-situated semantic alignment. For the purpose of this work, Principle 3* is satisfactory, since it establishes a minimal requirement for an interaction to be successful. Other more sophisticated versions of Principle 3 can be proposed, though, and any new extension will yield a different semantic alignment.



$i_1 = \langle \mathit{request}, (?A : \mathit{customer}), (?B : \mathit{travel\_agent}), \texttt{flight} \rangle$
$i_2 = \langle \mathit{request}, (?A : \mathit{customer}), (?B : \mathit{travel\_agent}), \texttt{accommodation} \rangle$
$i_3 = \langle \mathit{inform}, (!A : \mathit{customer}), (!B : \mathit{travel\_agent}), \texttt{return} \rangle$
$i_4 = \langle \mathit{inform}, (!A : \mathit{customer}), (!B : \mathit{travel\_agent}), \texttt{single} \rangle$
$i_5 = \langle \mathit{inform}, (!A : \mathit{customer}), (!B : \mathit{travel\_agent}), \texttt{origin}(?X) \rangle$
$i_6 = \langle \mathit{inform}, (!A : \mathit{customer}), (!B : \mathit{travel\_agent}), \texttt{destination}(?Y) \rangle$
$i_7 = \langle \mathit{inform}, (!A : \mathit{customer}), (!B : \mathit{travel\_agent}), \texttt{departing}(?V) \rangle$
$i_8 = \langle \mathit{inform}, (!A : \mathit{customer}), (!B : \mathit{travel\_agent}), \texttt{outboundTime}(?OT) \rangle$
$i_9 = \langle \mathit{inform}, (!A : \mathit{customer}), (!B : \mathit{travel\_agent}), \texttt{returning}(?W) \rangle$
$i_{10} = \langle \mathit{inform}, (!A : \mathit{customer}), (!B : \mathit{travel\_agent}), \texttt{inboundTime}(?IT) \rangle$
$i_{11} = \langle \mathit{inform}, (!A : \mathit{customer}), (!B : \mathit{travel\_agent}), \texttt{numberOfPassengers}(?NP) \rangle$
$i_{12} = \langle \mathit{inform}, (!B : \mathit{travel\_agent}), (!A : \mathit{customer}), \texttt{results}(?R) \rangle$
$i_{13} = \langle \mathit{inform}, (!A : \mathit{customer}), (!B : \mathit{travel\_agent}), \texttt{choice}(?C) \rangle$
$i_{14} = \langle \mathit{request}, (!A : \mathit{customer}), (!B : \mathit{travel\_agent}), \texttt{search} \rangle$
$i_{15} = \langle \mathit{inform}, (!A : \mathit{customer}), (!B : \mathit{travel\_agent}), \texttt{passengerDetails}(?PD) \rangle$

$i_{16} = \langle \mathit{inform}, (!A : \mathit{customer}), (!B : \mathit{travel\_agent}), \texttt{hotelBookingsIn}(?X) \rangle$
$i_{17} = \langle \mathit{inform}, (!A : \mathit{customer}), (!B : \mathit{travel\_agent}), \texttt{signIn}(?V) \rangle$
$i_{18} = \langle \mathit{inform}, (!A : \mathit{customer}), (!B : \mathit{travel\_agent}), \texttt{signOut}(?W) \rangle$
$i_{19} = \langle \mathit{inform}, (!A : \mathit{customer}), (!B : \mathit{travel\_agent}), \texttt{numberOfRooms}(?NR) \rangle$
$i_{20} = \langle \mathit{inform}, (!A : \mathit{customer}), (!B : \mathit{travel\_agent}), \texttt{numberOfGuests}(?NG) \rangle$
$i_{21} = \langle \mathit{inform}, (!A : \mathit{customer}), (!B : \mathit{travel\_agent}), \texttt{guestDetails}(?GD) \rangle$
$i_{22} = \langle \mathit{inform}, (!A : \mathit{customer}), (!B : \mathit{travel\_agent}), \texttt{customerDetails}(?CD) \rangle$
$i_{23} = \langle \mathit{inform}, (!B : \mathit{travel\_agent}), (!A : \mathit{customer}), \texttt{reservationTerms}(?RT) \rangle$
$i_{24} = \langle \mathit{accept}, (!A : \mathit{customer}), (!B : \mathit{travel\_agent}), \texttt{reservationTerms}(!RT) \rangle$
$i_{25} = \langle \mathit{reject}, (!A : \mathit{customer}), (!B : \mathit{travel\_agent}), \texttt{reservationTerms}(!RT) \rangle$
$i_{26} = \langle \mathit{inform}, (!B : \mathit{travel\_agent}), (!A : \mathit{customer}), \texttt{totalAmountToPay}(?TA) \rangle$
$i_{27} = \langle \mathit{commit}, (!A : \mathit{customer}), (!B : \mathit{travel\_agent}), \texttt{totalAmountToPay}(!TA) \rangle$
$i_{28} = \langle \mathit{reject}, (!A : \mathit{customer}), (!B : \mathit{travel\_agent}), \texttt{totalAmountToPay}(!TA) \rangle$
$i_{29} = \langle \mathit{inform}, (!A : \mathit{customer}), (!B : \mathit{travel\_agent}), \texttt{paymentInfo}(?PI) \rangle$
$i_{30} = \langle \mathit{inform}, (!B : \mathit{travel\_agent}), (!A : \mathit{customer}), \texttt{reservationSummary}(?RS) \rangle$

Figure 1: Interaction model for the customer agent



$j_1 = \langle request, (?A : customer), (?B : travel\_agent), \texttt{flight}\rangle$

$j_2 = \langle request, (?A : customer), (?B : travel\_agent), \texttt{hotel}\rangle$

$j_3 = \langle inform, (!A : customer), (!B : travel\_agent), \texttt{roundTrip}\rangle$

$j_4 = \langle inform, (!A : customer), (!B : travel\_agent), \texttt{oneWay}\rangle$

$j_5 = \langle inform, (!A : customer), (!B : travel\_agent), \texttt{from}(?X)\rangle$

$j_6 = \langle inform, (!A : customer), (!B : travel\_agent), \texttt{to}(?Y)\rangle$

$j_7 = \langle inform, (!A : customer), (!B : travel\_agent), \texttt{leavingDate}(?V)\rangle$

$j_8 = \langle inform, (!A : customer), (!B : travel\_agent), \texttt{time}(?T)\rangle$

$j_9 = \langle inform, (!A : customer), (!B : travel\_agent), \texttt{returnDate}(?W)\rangle$

$j_{10} = \langle inform, (!A : customer), (!B : travel\_agent), \texttt{flexibleOnDates}(?FD)\rangle$

$j_{11} = \langle inform, (!A : customer), (!B : travel\_agent), \texttt{passengers}(?P)\rangle$

$j_{12} = \langle inform, (!B : travel\_agent), (!A : customer), \texttt{flightOutcome}(?FO)\rangle$

$j_{13} = \langle inform, (!A : customer), (!B : travel\_agent), \texttt{myFlight}(?F)\rangle$

$j_{14} = \langle request, (!A : customer), (!B : travel\_agent), \texttt{newSearch}\rangle$

$j_{15} = \langle inform, (!A : customer), (!B : travel\_agent), \texttt{city}(?X)\rangle$

$j_{16} = \langle inform, (!A : customer), (!B : travel\_agent), \texttt{register}(?V)\rangle$

$j_{17} = \langle inform, (!A : customer), (!B : travel\_agent), \texttt{nights}(?N)\rangle$

$j_{18} = \langle inform, (!A : customer), (!B : travel\_agent), \texttt{rooms}(?R)\rangle$

$j_{19} = \langle inform, (!A : customer), (!B : travel\_agent), \texttt{lodgers}(?L)\rangle$

$j_{20} = \langle inform, (!B : travel\_agent), (!A : customer), \texttt{hotelOutcome}(?HO)\rangle$

$j_{21} = \langle inform, (!A : customer), (!B : travel\_agent), \texttt{myHotel}(?H)\rangle$

$j_{22} = \langle inform, (!A : customer), (!B : travel\_agent), \texttt{passengerData}(?PD)\rangle$

$j_{23} = \langle inform, (!A : customer), (!B : travel\_agent), \texttt{lodgerData}(?LD)\rangle$

$j_{24} = \langle inform, (!A : customer), (!B : travel\_agent), \texttt{contactInfo}(?CI)\rangle$

$j_{25} = \langle inform, (!B : travel\_agent), (!A : customer), \texttt{rulesAndRestrictions}(?RR)\rangle$

$j_{26} = \langle accept, (!A : customer), (!B : travel\_agent), \texttt{rulesAndRestrictions}(!RR)\rangle$

$j_{27} = \langle reject, (!A : customer), (!B : travel\_agent), \texttt{rulesAndRestrictions}(!RR)\rangle$

$j_{28} = \langle inform, (!B : travel\_agent), (!A : customer), \texttt{totalPrice}(?TP)\rangle$

$j_{29} = \langle commit, (!A : customer), (!B : travel\_agent), \texttt{totalPrice}(!TP)\rangle$

$j_{30} = \langle reject, (!A : customer), (!B : travel\_agent), \texttt{totalPrice}(!TP)\rangle$

$j_{31} = \langle inform, (!A : customer), (!B : travel\_agent), \texttt{payingData}(?PD)\rangle$

$j_{32} = \langle inform, (!B : travel\_agent), (!A : customer), \texttt{flightSummary}(?FS)\rangle$

$j_{33} = \langle inform, (!B : travel\_agent), (!A : customer), \texttt{hotelSummary}(?HS)\rangle$

Figure 2: Interaction model for the travel agent

3.2. Global Interaction

If the customer and travel agent interact by message passing, another interaction unfolds. This contains more detail than the ones specified in Figure 1 and Figure 2 which only capture a partial view of the actual *global interaction*, namely, the view from the perspective of the customer and the travel agent, respectively. A global interaction model matches all messages occurring in compatible illocutions of agent interaction models, where compatibility is based on Principle 1 and Principle 2.

Actually, neither agent needs to be aware of the model followed by the other for the interaction to unfold correctly in its totality. In general, two (or more) agents are capable to interact following separate interaction models if their states are assumed to be projections of states of a global interaction —which, in general, is not known to each of the agents— and each state transition that separate agents follow when an illocution is uttered has a corresponding state transition in the global interaction. In order for this to happen, an alignment protocol is proposed in Section 5.1. Nonetheless, the global interaction model itself is helpful from a theoretical point of view as it allows us to define the I-SSA semantic alignment. In Section 4.2 we give a formal account of the global interaction model through the idea of a product of interaction models, which we call the communication product.

3.3. What is Shared?

Even though we do not assume agents to share any ontology, agents must agree on the following in order for I-SSA to be effective.

A common language of illocutionary particles. These are typically realised in terms of speech act verbs, the number of which can be taken as reasonably low. Wierzbicka's dictionary is a remarkable effort to define a semantic dictionary of English speech act verbs. It contains definitions of 250 speech act verbs [25]. KQML contains no more than 35 performatives [26].

A family of roles. Senders and recipients of messages must be identified. This is usually done by means of agent identifiers and roles, and, for this reason, agents must share a collection of roles.

A content language. Although agents' ontologies may be different, we assume that agents agree on a language with which the content of illocutions is expressed. This language is generally as expressive as first-order logic, but we shall treat messages as propositions, that is, as grounded atomic sentences, leaving the generalisation to first-order sentences for future work.

An alignment protocol. This protocol will help agents resolving semantic mismatches. It makes use of a minimal set of terms the semantics of which is also assumed to be agreed by all interacting agents.

4. I-SSA Formalisation

We model a multiagent system as a set MAS of agents. Each agent in MAS has a unique identifier and may take one (or more) roles in the context of an interaction. Let *Role* be the set of roles and *Id* the set of agent identifiers. We write $(id : r)$, with $r \in Role$ and $id \in Id$, for the agent in MAS with identifier *id* playing the role r.

Each agent is able to communicate by sending messages from a set M, which is local to the agent. We assume that a set $\mathfrak{I}_P$ of *illocutionary particles* is shared by all agents.

Definition 1. *Given a non-empty set M of messages, the set of* illocutions generated by M*, denoted by* $\mathfrak{I}(M)$*, is the set of tuples* $\langle \iota, (id : r), (id' : r'), m \rangle$ *with* $\iota \in \mathfrak{I}_P$*,* $m \in M$*, and* $(id : r), (id' : r')$ *agents such that* $id \neq id'$*.*

If $\varphi = \langle \iota, (id : r), (id' : r'), m \rangle$ *is an illocution then* $(id : r)$ *is the* sender *of* φ *and* $(id' : r')$ *is the* receiver *of* φ*. In addition,* $\langle \iota, (id : r), (id' : r') \rangle$ *and* m *are called the* head *and* content *of* φ*, respectively.*

4.1. Interaction Models

We model an interaction model as a (partial) deterministic finite-state machine whose transitions are labelled either with illocutions, or with special transitions such as, e.g., timeouts, or null transitions (λ-transitions), which prompt state changes without message passing.

Definition 2. *An* interaction model *is a tuple* $\mathrm{IM} = \langle Q, q^0, F, M, C, \delta \rangle$ *where:*

- Q *is a finite set of* states*,*
- $q^0 \in Q$ *is a distinguished element of* Q *called the* initial state*,*
- F *is a non-empty subset of* Q *whose elements are called* final states*,*
- M *is a finite non-empty set of* messages*,*
- C *is a finite set of* special transitions*, and*
- δ *is a partial function from* $Q \times (\mathfrak{I}(M) \cup C)$ *to* Q *called the* transition function*.*

Remark. Although not explicitly stated in Definition 2, for theoretical reasons we take for granted that every interaction model contains a special transition ε such that $\delta(q,\varepsilon) = q$ for all $q \in Q$.

Given an interaction model $\mathrm{IM} = \langle Q, q^0, F, M, C, \delta\rangle$, we denote by $\mathfrak{I}_{\mathrm{IM}}$ (or simply $\mathfrak{I}$) the subset of $\mathfrak{I}(M)$ made up of all the illocutions present in IM, i.e., all the illocutions that appear in elements of the domain of δ. IM is associated with an automaton, $Aut(\mathrm{IM}) = \langle Q, q^0, F, \Sigma, \delta\rangle$, where $\Sigma = \mathfrak{I} \cup C$.

Example. *As hinted before, all messages will be treated as grounded atomic sentences. If we replace illocutions in Figure 1 with the ones bellow (not all are included), we obtain an interaction model as in Definition 2 and similarly for Figure 2. From here on these will be the automata under consideration.*

$$
\begin{aligned}
i_1 &= \langle \textit{request}, (a : \textit{customer}), (b : \textit{travel\_agent}), \texttt{flight}\rangle\\
i_2 &= \langle \textit{request}, (a : \textit{customer}), (b : \textit{travel\_agent}), \texttt{accommodation}\rangle\\
i_3 &= \langle \textit{inform}, (a : \textit{customer}), (b : \textit{travel\_agent}), \texttt{return}\rangle\\
i_4 &= \langle \textit{inform}, (a : \textit{customer}), (b : \textit{travel\_agent}), \texttt{single}\rangle\\
i_5 &= \langle \textit{inform}, (a : \textit{customer}), (b : \textit{travel\_agent}), \texttt{origin}\rangle\\
i_6 &= \langle \textit{inform}, (a : \textit{customer}), (b : \textit{travel\_agent}), \texttt{destination}\rangle\\
i_7 &= \langle \textit{inform}, (a : \textit{customer}), (b : \textit{travel\_agent}), \texttt{departing}\rangle\\
i_8 &= \langle \textit{inform}, (a : \textit{customer}), (b : \textit{travel\_agent}), \texttt{outboundTime}\rangle\\
i_9 &= \langle \textit{inform}, (a : \textit{customer}), (b : \textit{travel\_agent}), \texttt{returning}\rangle\\
i_{10} &= \langle \textit{inform}, (a : \textit{customer}), (b : \textit{travel\_agent}), \texttt{inboundTime}\rangle\\
i_{11} &= \langle \textit{inform}, (a : \textit{customer}), (b : \textit{travel\_agent}),\\
&\qquad \texttt{numberOfPassengers}\rangle\\
i_{12} &= \langle \textit{inform}, (b : \textit{travel\_agent}), (a : \textit{customer}), \texttt{results}\rangle
\end{aligned}
$$

4.2. The Communication Product

We shall use the algebraic product of two interaction models in order to capture all possible interactions between agents. In general, a product of two objects is the natural algebraic construction that represents all possible behaviours of the combination of these two objects. The *communication product* (CP) defined below, thus, captures the global interaction with respect to the message-passing behaviour of agents of two interaction models. It is not an unconstrained product, as it considers compatibility of illocutions in terms of illocutionary particles, senders, and receivers. In category theory a constrained product is called a pullback [27]. Theorem 1 states that the communication product is a pullback in the natural category of interaction models.

Definition 3. *Let* $\mathrm{IM}_i = \langle Q_i, q_i^0, F_i, M_i, C_i, \delta_i\rangle$ $(i = 1, 2)$ *be two interaction models. The* communication product *of* IM_1 *and* IM_2*, denoted by* $\mathrm{IM}_1 \otimes \mathrm{IM}_2$*, is the interaction model* $\langle Q, q^0, F, M, C, \delta\rangle$ *where:*

- *Q is the Cartesian product of Q_1 and Q_2, that is, the states in Q are all possible ordered pairs $\langle q_1, q_2\rangle$ with $q_1 \in Q_1$ and $q_2 \in Q_2$,*
- *the initial state q^0 is the pair $\langle q_1^0, q_2^0\rangle$,*
- *F is the Cartesian product of F_1 and F_2,*
- *M is the Cartesian product of M_1 and M_2,*
- *C is the Cartesian product of C_1 and C_2,*
- *δ is defined as follows: $\langle q_1', q_2'\rangle = \delta(\langle q_1, q_2\rangle, \sigma)$ if*
  - *$\sigma = \langle \iota, (id : r), (id' : r'), \langle m_1, m_2\rangle\rangle$ and $q_i' = \delta_i(q_i, \langle \iota, (id : r), (id' : r'), m_i\rangle)$ or*
  - *$\sigma = (c_1, c_2)$ and $q_i' = \delta_i(q_i, c_i)$ for $i = 1, 2$.*

Remark. Notice that, according to the definition of δ, $\varepsilon = \langle \varepsilon_1, \varepsilon_2\rangle$ is such that $\delta(\langle q_1, q_2\rangle, \varepsilon) = \langle q_1, q_2\rangle$ for all $\langle q_1, q_2\rangle \in Q_1 \times Q_2$. Additionally, special transitions of IM_i are paired with ε_j of IM_j $(i \neq j)$. In this way, we capture the idea that, although the global interaction state changes, this may not be the case for one of the interaction models.

Example. *The communication product of the interaction models for the roles of customer and travel agent is partially depicted in Figure 3. For instance, there exists an arc from state $\langle s_{00}, t_{00}\rangle$ to state $\langle s_{02}, t_{02}\rangle$ labelled with*

$$
\begin{aligned}
k_4 = \langle &\textit{request}, (a : \textit{customer}), (b : \textit{travel\_agent}),\\
&\langle \texttt{accommodation}, \texttt{hotel}\rangle\rangle
\end{aligned}
$$

This is due to the fact that (i) there is an arc from the state s_{00} to the state s_{02} labelled with

$$
\begin{aligned}
i_2 = \langle &\textit{request}, (a : \textit{customer}), (b : \textit{travel\_agent}),\\
&\texttt{accommodation}\rangle
\end{aligned}
$$

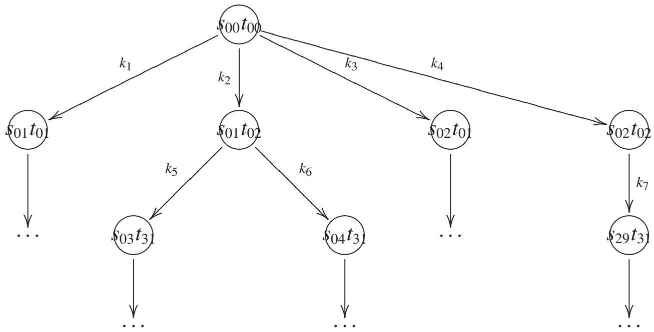
in the customer's interaction model, (ii) there exists an arc from t_{00} to t_{02} labelled with

$$
j_2 = \langle \textit{request}, (a : \textit{customer}), (b : \textit{travel\_agent}), \texttt{hotel}\rangle
$$

in the travel agent's interaction model, and (iii) the two illocution heads match up. Although not shown, this path leads to a final state. Notice also that the term `accommodation` *is paired with* `flight` *in illocution k_3. This path, though, does not lead to a final state.*

4.3. A Categorical Characterisation of the Communication Product

As hinted at the beginning of Section 4.2, there exists a natural categorical characterisation of the communication product. In what follows we define the category of interaction models and prove that the communication product is a pullback in this category.



$k_1 = \langle request, (a : customer), (b : travel\_agent), \langle \texttt{flight}, \texttt{flight} \rangle \rangle$
$k_2 = \langle request, (a : customer), (b : travel\_agent), \langle \texttt{flight}, \texttt{hotel} \rangle \rangle$
$k_3 = \langle request, (a : customer), (b : travel\_agent), \langle \texttt{accommodation}, \texttt{flight} \rangle \rangle$
$k_4 = \langle request, (a : customer), (b : travel\_agent), \langle \texttt{accommodation}, \texttt{hotel} \rangle \rangle$
$k_5 = \langle inform, (a : customer), (b : travel\_agent), \langle \texttt{return}, \texttt{city} \rangle \rangle$
$k_6 = \langle inform, (a : customer), (b : travel\_agent), \langle \texttt{single}, \texttt{city} \rangle \rangle$
$k_7 = \langle inform, (a : customer), (b : travel\_agent), \langle \texttt{hotelBookingsIn}, \texttt{city} \rangle \rangle$

Figure 3: Partial description of the communication product in the travel reservation scenario

Definition 4. *Let* $\mathrm{IM}_i = \langle Q_i, q_i^0, F_i, M_i, C_i, \delta_i \rangle$ $(i = 1, 2)$ *be two interaction models. A* morphism of interaction models $f : \mathrm{IM}_1 \to \mathrm{IM}_2$ *is a pair of functions* $f = \langle g, h \rangle$, *where* $g : Q_1 \to Q_2$ *and* $h : \Sigma_1 \to \Sigma_2$*, such that:*

- $g(q_1^0) = q_2^0$ *and* $g(F_1) \subseteq F_2$,
- $h(\Im_1) \subseteq \Im_2$, $h(C_1) \subseteq C_2$ *and* $h(\varepsilon_1) = \varepsilon_2$,
- $g(\delta_1(q_1, \sigma_1)) = \delta_2(g(q_1), h(\sigma_1))$ *for all* $q_1 \in Q_1$ *and* $\sigma_1 \in \Sigma_1$.

From here on, if $f = \langle g, h \rangle$ is a morphism of interaction models, we use f both applying on states and transitions, providing that no confusion arises. Hence $f(q)$ and $f(\sigma)$ replace $g(q)$ and $h(\sigma)$, respectively.

Definition 5. *The category of interaction models* **IM** *has interaction models as objects, and morphisms of interaction models as arrows. Both composition law and identity are defined in the natural way.*

Theorem 1. *The communication product is a pullback in the category* **IM**.

Proof. Let IM_* be the interaction model with q_* and m_* as the only state and message, respectively, and transition function δ_* defined as follows:

$$\delta_*(q_*, \langle \iota, (id : r), (id' : r'), m_* \rangle) = q_*$$

for all $\langle \iota, (id : r), (id' : r'), m_* \rangle \in \Im(\{m_*\})$.

Let $f_i : \mathrm{IM}_i \to \mathrm{IM}_*$ (with $i = 1, 2$) defined by $f_i(q_i) = q_*$ for $q_i \in Q_i$, $f_i(\langle \iota, (id : r), (id' : r'), m_i \rangle) = \langle \iota, (id : r), (id' : r'), m_* \rangle$, for $\langle \iota, (id : r), (id' : r'), m_i \rangle \in \Im_i$ while $f_i(c_i) = \varepsilon_*$ for each $c_i \in C_i$. It is straightforward to prove that f_i $(i = 1, 2)$ is a morphism of interaction models. In the remainder of the proof we show that $\mathrm{IM}_1 \otimes \mathrm{IM}_2$ is a pullback of the arrows f_1 and f_2 (see Figure 4).

Let $\theta_i : \mathrm{IM}_1 \otimes \mathrm{IM}_2 \to \mathrm{IM}_i$ $(i = 1, 2)$ be defined as the projection on states, that is, $\theta_i(\langle q_1, q_2 \rangle) = q_i$, $\theta_i(\langle \iota, (id : r), (id' : r'), \langle m_1, m_2 \rangle \rangle) = \langle \iota, (id : r), (id' : r'), m_i \rangle$, while $\theta_i(\langle c_1, c_2 \rangle) = c_i$. θ_i is a morphism of interaction models and $f_1\theta_1 = f_2\theta_2$. Now, let us assume that there exist two morphisms ϑ_1 and ϑ_2, $\vartheta_i : \mathrm{IM} \to \mathrm{IM}_i$ $(i = 1, 2)$, such that $f_1\vartheta_1 = f_2\vartheta_2$. We must prove that there exists a unique morphism $\xi : \mathrm{IM} \to \mathrm{IM}_1 \otimes \mathrm{IM}_2$ such that $\vartheta_i = \theta_i\xi$. First of all, let us define $\xi(q) = \langle \vartheta_1(q), \vartheta_2(q) \rangle$ on the states of IM. Secondly, given an illocution φ of IM, the fact that $f_1\vartheta_1 = f_2\vartheta_2$ ensures that $\vartheta_1(\varphi)$ and $\vartheta_2(\varphi)$ have the same illocution head. Then we can write $\vartheta_i(\varphi) = \langle \iota, (id : r), (id' : r'), m_i \rangle$. Accordingly, we define $\xi(\varphi) = \langle \iota, (id : r), (id' : r'), \langle m_1, m_2 \rangle \rangle$, and $\xi(c) = \langle \vartheta_1(c), \vartheta_2(c) \rangle$ for each special transition c of IM. It is straightforward to prove that ξ is a morphism of interaction models, and also that ξ is the unique morphism in such conditions.

Q.E.D.

$$
\begin{tikzcd}
\mathrm{IM} \arrow[dr, dashed, "\xi"] \arrow[drr, "\vartheta_2"] \arrow[ddr, "\vartheta_1"'] & & \\
& \mathrm{IM}_1 \otimes \mathrm{IM}_2 \arrow[r, "\theta_2"] \arrow[d, "\theta_1"] & \mathrm{IM}_2 \arrow[d, "f_2"] \\
& \mathrm{IM}_1 \arrow[r, "f_1"] & \mathrm{IM}_*
\end{tikzcd}
$$

Figure 4: Pullback diagram

4.4. Semantic Alignment through the Communication Product

Being a model of all compatible agent interactions of varying interaction models, the communication product is the place to look at if we want to define the I-SSA semantic alignment. From a theoretical viewpoint, in order to establish semantic relations among messages, we examine the language generated by the communication product. This formally synthesises Principle 3* explained in Section 3.1. Thus messages of different interaction models are semantically related if they are paired in illocutions whose utterance makes the interaction reach a final state (that is, makes the interaction successfull) according to the global interaction determined by the communication product. This is formally given below. We use '$\sqsubseteq$' to denote subsumption of messages, while '$\sqcup$' to denote disjunction. Semantic equivalence between messages, denoted with '$\equiv$', arises when they subsume each other. We also pair messages with natural numbers to keep syntactically identical messages separate, since they may not be semantically equivalent.

Definition 6. *Let* $\mathrm{IM}_i = \langle Q_i, q_i^0, F_i, M_i, C_i, \delta_i \rangle$ *(with* $i = 1, 2$*) be two interaction models. Let* $m \in M_1$ *and* $m^1, \ldots, m^n \in M_2$*. We write*

$$\langle 1, m \rangle \sqsubseteq \langle 2, m^1 \rangle \sqcup \cdots \sqcup \langle 2, m^n \rangle$$

if for all strings x accepted by the product $\mathrm{IM}_1 \otimes \mathrm{IM}_2$*, if* $\langle \iota, (id : r), (id' : r'), \langle m, m' \rangle \rangle$ *appears in x then* $m' = m^k$ *for some* $k \in \{1, \ldots, n\}$*. If such* $m^1, \ldots, m^n \in M_2$ *do not exist, we simply write*

$$\langle 1, m \rangle \sqsubseteq \bot$$

We define $\langle 2, m \rangle \sqsubseteq \langle 1, m^1 \rangle \sqcup \cdots \sqcup \langle 1, m^n \rangle$ *analogously.*

It is possible to establish relations amongst messages with respect to a specific illocution particle.

Definition 7. *Let* $\mathrm{IM}_i = \langle Q_i, q_i^0, F_i, M_i, C_i, \delta_i \rangle$ *(with* $i = 1, 2$*) be two interaction models. Let* $m \in M_1$ *and* $m^1, \ldots, m^n \in M_2$*. Let* $\iota_0 \in \mathfrak{I}_{\mathrm{P}}$*. We write*

$$\langle 1, m \rangle \sqsubseteq_{\iota_0} \langle 2, m^1 \rangle \sqcup \cdots \sqcup \langle 2, m^n \rangle$$

if for all strings x accepted by the product $\mathrm{IM}_1 \otimes \mathrm{IM}_2$*, if* $\langle \iota_0, (id : r), (id : r'), \langle m, m' \rangle \rangle$ *appears in x then* $m' = m^k$ *for some* $k \in \{1, \ldots, n\}$*. If such* $m^1, \ldots, m^n \in M_2$ *do not exist, we simply write*

$$\langle 1, m \rangle \sqsubseteq_{\iota_0} \bot$$

We define $\langle 2, m \rangle \sqsubseteq_{\iota_0} \langle 1, m^1 \rangle \sqcup \cdots \sqcup \langle 1, m^n \rangle$ *analogously.*

The *semantic alignment* is made up of all these expressions (Definition 6 and Definition 7). It represents the formal synthesis of I-SSA principles.

Example. *The semantic relations among the customer's and travel agent's messages are enumerated in Figure 5. Only the semantic alignment that corresponds to Definition 6 is shown, as the other one is analogous.*

4.5. Interaction- vs Non-Interaction-Situated Semantic Alignment

The characteristics of I-SSA become more apparent if we compare it with matching techniques that are not interaction-situated. Far from making a thorough numerical analysis, we have set I-SSA against three state-of-the-art ontology matchers, namely, COMA++ [28], Falcon-OA [29] and OLA [30], in order to highlight more qualitative differences. For this, we have designed two ontologies that conform to more complex versions of the interaction models presented in this paper and launched the three matchers. For a fully description of this scenario we refer the reader to [31].

As an example, the three matchers failed to match `accommodation` and `hotel`. OLA and COMA++ (with a node strategy) returned the equivalence of classes Accommodation $\equiv$ Account with confidence values 0.38 and 0.23, respectively, whereas Falcon-OA returned no relation involving any of these terms. In the extended version of the travel agent's interaction model, the travel agent suggests the customer to create an account before informing the customer of the rules and restrictions. Thus, the above equivalence could have never be returned by I-SSA as `accommodation` and `account` come along with distinct illocutionary particles (*inform* and *suggest*). Moreover, they cannot be uttered in concurrent interaction states.

$$
\begin{aligned}
\langle a, \texttt{flight}\rangle &\equiv \langle b, \texttt{flight}\rangle\\
\langle a, \texttt{accommodation}\rangle &\equiv \langle b, \texttt{hotel}\rangle\\
\langle a, \texttt{return}\rangle &\equiv \langle b, \texttt{roundTrip}\rangle\\
\langle a, \texttt{single}\rangle &\equiv \langle b, \texttt{oneWay}\rangle\\
\langle a, \texttt{origin}\rangle &\equiv \langle b, \texttt{from}\rangle\\
\langle a, \texttt{destination}\rangle &\equiv \langle b, \texttt{to}\rangle\\
\langle a, \texttt{departing}\rangle &\equiv \langle b, \texttt{leavingDate}\rangle\\
\langle a, \texttt{returning}\rangle &\equiv \langle b, \texttt{returnDate}\rangle\\
\langle a, \texttt{outboundTime}\rangle &\sqsubseteq \langle b, \texttt{time}\rangle\\
\langle a, \texttt{inboundTime}\rangle &\sqsubseteq \langle b, \texttt{time}\rangle\\
\bot &\sqsupseteq \langle b, \texttt{flexibleOnDates}\rangle\\
\langle a, \texttt{numberOfPassengers}\rangle &\equiv \langle b, \texttt{passengers}\rangle\\
\langle a, \texttt{hotelBookingsIn}\rangle &\equiv \langle b, \texttt{city}\rangle\\
\langle a, \texttt{signIn}\rangle &\equiv \langle b, \texttt{register}\rangle\\
\langle a, \texttt{signOut}\rangle &\equiv \langle b, \texttt{nights}\rangle\\
\langle a, \texttt{numberOfRooms}\rangle &\equiv \langle b, \texttt{rooms}\rangle\\
\langle a, \texttt{numberOfGuests}\rangle &\equiv \langle b, \texttt{lodgers}\rangle\\
\langle a, \texttt{result}\rangle &\sqsubseteq \langle b, \texttt{flightOutcome}\rangle \sqcup\\
& \langle b, \texttt{hotelOutcome}\rangle\\
\langle a, \texttt{choice}\rangle &\sqsubseteq \langle b, \texttt{myFlight}\rangle \sqcup\\
& \langle b, \texttt{myHotel}\rangle\\
\langle a, \texttt{search}\rangle &\equiv \langle b, \texttt{newSearch}\rangle\\
\langle a, \texttt{passengerDetails}\rangle &\equiv \langle b, \texttt{passengerData}\rangle\\
\langle a, \texttt{guestDetails}\rangle &\equiv \langle b, \texttt{lodgerData}\rangle\\
\langle a, \texttt{customerDetails}\rangle &\equiv \langle b, \texttt{contactInfo}\rangle\\
\langle a, \texttt{reservationTerms}\rangle &\equiv \langle b, \texttt{rulesAndRestrictions}\rangle\\
\langle a, \texttt{totalAmountToPay}\rangle &\equiv \langle b, \texttt{totalPrice}\rangle\\
\langle a, \texttt{paymentInfo}\rangle &\equiv \langle b, \texttt{payingData}\rangle\\
\langle a, \texttt{reservation\_summary}\rangle &\sqsubseteq \langle b, \texttt{flightSummary}\rangle \sqcup\\
& \langle b, \texttt{hotelSummary}\rangle
\end{aligned}
$$

Figure 5: Semantic alignment in the travel reservation scenario

Particularly, OLA resorts to WordNet [32] in order to discover semantic relations. Nonetheless, there is no apparent relationship between the synsets of the words 'hotel' and 'accommodation'. They actually become related within this specific interaction.

5. I-SSA Dynamics

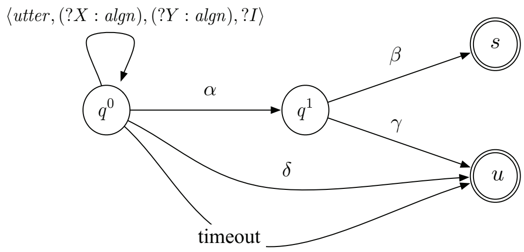
As mentioned before, interaction models specify the space of interactions that are allowed, and their communication product captures the entire space of actual interactions when combining particular ones. The semantic relations defined above are those justified by the entire space of actual interactions. This product, though, may not be accessible to agents. This is the case when interaction models are not completely open for inspection, because, for example, they are based on commercially confidential information, so agents are only aware of their local ones. Furthermore, interaction models can be of a size that computing the product becomes a high time and memory consuming task.

It is necessary to provide agents with a mechanism to discover the above semantic relationships while interactions unfold —in the kind of manner as intuitively described for our example above— assuming that for all agents participating in the interaction, the state they perceive stems from the actual global state (in other words, their locally managed states are projections of the actual global state), and this occurs throughout the entire interaction.

5.1. The Alignment Protocol

Let us consider a scenario where two agents A_1 and A_2, identified with id_1 and id_2, try to interact following (possibly distinct) interaction models IM_1 and IM_2, respectively. Let us assume that no other agents will take part in the interaction according to IM_1 and IM_2.

With agents knowing that they follow different interaction models and that semantic mismatches are likely to occur, communication requires to be done at another level. For this reason we define an *alignment protocol* that acts as a meta-protocol that links agents' interaction models. The alignment protocol (henceforth AP) is depicted in Figure 6.



$$
\begin{aligned}
\alpha &= \langle \mathit{inform}, (?X : \mathit{algn}), (?Y : \mathit{algn}), \texttt{final\_state}\rangle\\
\beta &= \langle \mathit{confirm}, (!Y : \mathit{algn}), (!X : \mathit{algn}), \texttt{final\_state}\rangle\\
\gamma &= \langle \mathit{deny}, (!Y : \mathit{algn}), (!X : \mathit{algn}), \texttt{final\_state}\rangle\\
\delta &= \langle \mathit{inform}, (?X : \mathit{algn}), (?Y : \mathit{algn}), \texttt{failure}\rangle
\end{aligned}
$$

Figure 6: The alignment protocol

There are four states: the initial state q^0, an intermediate state q^1, and two final states by name of letters s and u. These last ones are the initial letters of the words *successful* and *unsuccessful*: if the meta-level state s is reached, whatever path is followed, the object-level interaction is considered successful, otherwise unsuccessful. In this sense, we distinguish for the moment only between two kinds of interactions.

Regarding transitions, they are all listed below the figure except one that has a special status. Note that agents can only adopt one role, namely, the role of "aligner" (*algn*). There exist two sorts of messages: `failure` and `final_state`. The former can be tagged with the illocutionary particle *inform*, and the latter with *inform*, *confirm* and *deny*.

The following illocution scheme links agents' interaction models with the alignment protocol:

$$\langle utter, (?X : algn), (?Y : algn), ?I\rangle \qquad (1)$$

Above X and Y are identifier variables, while I is an illocution variable. Thus (1) can be seen as a meta-illocution, since its content is also an illocution. It is grounded with illocutions of the form $\langle utter, (id_i : algn), (id_j : algn), \varphi\rangle$, where $\varphi = \langle \iota, (id_i : r), (id_j : r'), v)\rangle$ is an illocution of agent A_i's interaction model IM_i. The sender and receiver of φ must be equal to the instantiations of X and Y, respectively. Further, let us stress that φ has to come from the interaction model associated with X's instantiation. Consequently, the choice of "utter" as illocutionary particle seems natural, as it expresses the sender's attitude with respect to its own interaction model: if A_j receives $\langle utter, (id_i : algn), (id_j : algn), \varphi\rangle$, A_j can safely assume that A_i has decided to utter φ according to IM_i. In the following section the dynamics of the alignment protocol is thoroughly explained.

5.1.1. Alignment Protocol Dynamics

Each agent is guided by both the alignment protocol and its interaction model, whilst effective communication is done through the former. When agents initiate an interaction, both of them are in state q^0 with respect to AP. Also A_i is in state q_i^0 with respect to IM_i ($i = 1, 2$). In order to cover all cases, let us assume that A_i is in an arbitrary state $q_i \in Q_i$. There are several possibilities:

- AP.1 A_i decides to utter $\varphi = \langle \iota, (id_i : r), (id_j : r'), v)\rangle$ in the IM_i context, with $\varphi \in \delta_i(q_i, \cdot)$.[1] The communication act is carried out via AP: agent A_i sends the meta-illocution $\langle utter, (id_i : algn), (id_j : algn), \varphi\rangle$ to A_j. The state remains the same in the AP context, whereas q_i turns to $q_i' = \delta_i(q_i, \varphi)$ in the IM_i context.
- AP.2 A_i prompts a state change by a special transition $c_i \in C_i$ in the IM_i context. Thus q_i turns to $q_i' = \delta_i(q_i, c_i)$. This action is not reflected in AP since it does not involve any communication act.
- AP.3 A_i receives $\langle utter, (id_j : algn), (id_i : algn), \varphi\rangle$ in the meta-level AP, with $\varphi = \langle \iota, (id_j : r), (id_i : r'), v)\rangle$. Recall that from A_i's point of view v is a foreign message, and, for this reason, it is considered semantically different from all local ones. The key is that v is to be mapped with one of those messages A_i expects to receive at state q_i in the IM_i context (Principle 1 stated in Section 3.1). Moreover, we can make a selection and consider those messages contained in illocutions whose head is equal to that of φ (Principle 2 along with agent identification). In this way, A_i is to choose an element from the following set:

  $$D = \{w \mid \langle \iota, (id_j : r), (id_i : r'), w\rangle \in dom(\delta_i(q_i, \cdot))\}$$

  There are two possibilites: either D is empty or not.

  (a) As long as D is not empty, A_i can select an element w of D by making use of the matching mechanism explained in Section 5.1.2. So q_i becomes $q_i' = \delta_i(q_i, \psi)$, where $\psi = \langle \iota, (id_j : r), (id_i : r'), w)\rangle$.

  (b) If D is empty, v cannot be matched. The interaction is considered unsuccessful. A_i is to send a failure message to A_j by uttering $\langle inform, (id_i : algn), (id_j : algn), \texttt{failure}\rangle$ which matches with the illocution scheme δ. Thus q^0 becomes u in the AP context.
- AP.4 If q_i is a final state and A_i considers the interaction to be finished, she can send the illocution $\langle inform, (id_i : algn), (id_j : algn), \texttt{final\_state}\rangle$ to A_j, which matches with the illocution scheme α. Thus q^0 turns to q^1, and A_j is supposed to ground β or γ, either confirming or denying the completion of the interaction, respectively. Grounding β makes agents to reach the final state s, and the interaction is considered successful; γ, however, leads to an unsuccessful interaction.
- AP.5 Finally, we have to take into account the possibility of a deadlock. This is the case when, for example, successive mappings have led the agents to states where both of them only await messages. To avoid deadlocks, the special transition *timeout* is linked to the initial state q^0 in AP. If a specific period of time is exceeded, this transition leads agents to finish the interaction, which is unsuccessful.

[1] $\delta_i(q_i, \cdot)$ is the function defined from $\Sigma_i = \mathfrak{I}_i \cup C_i$ to Q_i in the natural way.

5.1.2. The Matching Mechanism

As mentioned above, a matching mechanism is called whenever a message is received. In a nutshell, it is based on three assertions:

- every foreign message is associated with a categorical variable ranging over local messages, and a variable assignment represents a matching element;
- the matching mechanism computes frequency distributions of all these variables on the basis of past successful interactions;
- matching decisions are determined by virtue of these distributions.

Past Information: Histories and Frequency Distributions

Whenever an interaction is successfully performed, agents record relevant information that will be helpful in future interactions. This information is revealed in terms of histories that gather all past matching decisions. These histories increasingly enlarge the population on which a statistical reasoning for forthcoming matching decisions will be based. Below we explain both statistical updating and matching decisions in detail.

Agents build histories while interacting with the help of the alignment protocol. Specifically, a *history* is a sequence of the form:

$$h = q_i^0, \sigma_i^1, q_i^1, \ldots, q_i^{k-1}, \sigma_i^k, q_i^k, \ldots, q_i^{n-1}, \sigma_i^n, q_i^n$$

computed recursively as follows:

- q_i^0 is the initial state of IM_i, and
- $[\varphi, q_i']$ is queued in h if A_i is in case AP.1,
- $[c_i, q_i']$ is queued in h if A_i is in case AP.2,
- $[\langle \iota, (id_j : r), (id_i : r'), [v/w] \rangle, q_i']$ is queued in h if A_i is in case AP.3.a,
- q_i^n is a final state of IM_i.

Notice that unsuccessful interactions are not considered. It is not so easy to find out which particular matching is responsible for a failure, or if we should blame one agent or another for a wrong matching decision. For the purpose of this work we only consider successful interactions and leave the study of unsuccessful interactions for future research.

In order to make the notation clearer, we will dispense with subscripts. So we have two agents A and B, identified with a and b, and associated with interaction models IM_a and IM_b.

Let $\mathcal{H} = \{h^k\}_{k=1}^n$ be the sequence of all past successful histories reported by agent A so far. Notice that it may happen $h^k = h^l$ for $1 \leq k, l \leq n$ and $k \neq l$. If this is the case, as far as agent A is concerned, there is no other distinction between h^k and h^l but time occurrence. Now, from all information contained in these histories, we will particularly pay attention to those pairs of the form

$$p = \langle q, \langle \iota, (b : r), (a : r'), [v/w] \rangle \rangle \tag{2}$$

where $\langle \iota, (b : r), (a : r'), [v/w] \rangle$ comes straight after the state q in (at least) one history of $\mathcal{H}$. The reader should think of p as follows: at some point in the past and having received the illocution $\langle \iota, (b : r), (a : r'), v \rangle$ at state q, agent A decided to match message v with the local message w.

Forthcoming matching decisions will be based on successful past matching decisions, represented by pairs as p in (2). Henceforth we will refer to these pairs with the abbreviation *pmd* (past matching decision), or *pmd on v* if we want to specify the matched message.

Assume that agent A received v in the past. Let us consider the multiset (or bag) $\mathcal{P}_v$ of all pmd on v that appear in $\mathcal{H}$ (indeed, there may be more than one occurrence of the same pmd). $\mathcal{P}_v = \langle P_v, \pi_v \rangle$ where P_v is the underlying set of elements and $\pi_v : P_v \to \mathbb{N}$ is the multiplicity function. For the task at hand, message v will be treated as a statistical variable $V : P_v \to M_a$, where M_a is the set of A's local messages and V is defined in the natural way. If v turned out to be matched with $w \in M_a$ (in other words, w is a member of the range of V), the frequency associated with w is:

$$F(V = w) = \frac{\sum_{V(p)=w} \pi_v(p)}{\sum \pi_v(p)}$$

where summations range by default over $p \in P_v$. There is another attribute of the elements of $\mathcal{P}_v$ which is worth studying. If $I_{\mathrm{P}} : P_v \to \mathfrak{I}_{\mathrm{P}}$ is defined in the natural way,

$$F(V = w | I_{\mathrm{P}} = \iota) = \frac{\sum_{V(p)=w, I_{\mathrm{P}}(p)=\iota} \pi_v(p)}{\sum_{I_{\mathrm{P}}(p)=\iota} \pi_v(p)}$$

We will make use of the symbol $\mathcal{F}_v$ when referring to this frequency distribution. If $\mathcal{H} = \{h^k\}_{k=1}^n$ is the resulting history recording of $n \geq 1$ interactions with agent B, then $\mathcal{H}$ generates a family of frequency distributions $\mathcal{F} = \{\mathcal{F}_v\}_{v \in \Omega}$, where Ω is the set of all B's messages received by A so far. At whatever time a new interaction is successfully completed, $\mathcal{F}$ has to be updated.

Matching Criteria

In this section we explain the reasoning followed by agents when facing matching decisions. Imagine that A receives $\varphi = \langle \iota, (b : r), (a : r'), v_0 \rangle$ from B at state $q \in Q_a$. Let us consider the set D defined in case AP.3.a:

$$D = \{w \mid \langle \iota, (b : r), (a : r'), w \rangle \in dom(\delta_a(q, \cdot))\}$$

In principle, v_0 could be matched with any $w \in D$, but this can be refined. Let us distinguish between two cases: A has information about past successful interactions with B that involve v_0 or not.

In the case of no information, we let agent A to choose a message $w_0 \in D$ randomly. More specifically, if $D = \{w_1, \ldots, w_n\}$, $w_0 \in D$ is chosen with probability $p = \frac{1}{n}$. If agent A has information about former successful interactions, this will become available in terms of frequency distributions, $\mathcal{F} = \{\mathcal{F}_v\}$, as we have already explained. If it so happens that $\mathcal{F}_{v_0} \in \mathcal{F}$ then agent A can benefit from this information when making a matching decision on v_0. One first idea is to choose a local message w_0 such that

$$F(V_0 = w | V_0 \in D) \leq F(V_0 = w_0 | V_0 \in D)$$

for every $w \in D$. This leads us to the following criterion.

First Matching Criterion (maximal frequency criterion)

if $\mathcal{F}_{v_0} \in \mathcal{F}$ **then**
  choose $w_0 \in D$ such that
  $F(V_0 = w | V_0 \in D) \leq F(V_0 = w_0 | V_0 \in D)$
  for all $w \in D$
else
  choose $w_0 \in D$ with probability $p = \frac{1}{n}$
end if

This criterion highly depends on how rich the frequency distributions are. If there is not much information about past interactions, though, it does not make sense to fully rely on a matching element with maximal frequency. As an alternative, we propose to take the probability distribution $\{(w_i, p_i)\}_{i=1}^n$ where $p_i = F(V_0 = w_i | V_0 \in D)$, and choose w_i with probability p_i.

Second Matching Criterion (probability criterion)

if $\mathcal{F}_{v_0} \in \mathcal{F}$ **then**
  choose $w_0 \in D$ with probability
  $p = F(V_0 = w_0 | V_0 \in D)$
else
  choose $w_0 \in D$ with probability $p = \frac{1}{n}$
end if

Neither the maximal frequency criterion nor the probability-based one allow to discover new matching elements. In order to overcome this we put forward a last criterion which consists in contaminating the previous distribution with a discrete uniform distribution (the contamination parameter $s \in (0, 1)$ is usually a number close to 1).

Third Matching Criterion (contaminated probability criterion)

Require: $s \in (0, 1)$, $s \approx 1$
if $\mathcal{F}_{v_0} \in \mathcal{F}$ **then**
  choose $w_0 \in D$ with probability
  $p = s \cdot F(V_0 = w_0 | V_0 \in D) + (1 - s) \cdot \frac{1}{n}$
else
  choose $w_0 \in D$ with probability $p = \frac{1}{n}$
end if

The three matching criteria described above can be further refined by truncating with the event $\{I_P = \iota\}$ where ι is the illocutionary particle of φ.

5.2. Semantic Alignment through the Alignment Protocol

The alignment protocol described in Section 5.1 helps agents to interact successfully. The more interactions are completed, the more messages become related. In what follows, we firstly pin down these semantic relationships in a logical fashion, and then expound the link with the semantic alignment deduced from the communication product by means of Theorem 2.

Assume that agent A_i $(i = 1, 2)$ has generated a family $\mathcal{F}$ of frequency distributions, and let $v \in M_j$ $(j \neq i)$ for which $\mathcal{F}_v \in \mathcal{F}$. If $w \in M_i$ is such that $t = F(V = w) \neq 0$, we write

$$\langle j, v \rangle \sqsubseteq \langle i, w \rangle [t]$$

Additionally, when $t = F(V = w | I_P = \iota) \neq 0$, we write $\langle j, v \rangle \sqsubseteq_\iota \langle i, w \rangle [t]$. In both cases, $t \in (0, 1]$ can be seen as a *confidence degree* of the subsumption.

The way the alignment protocol is designed ensures the following —which has a clear counterpart regarding illocutionary particles.

Theorem 2. *Let us assume that $\langle j, v \rangle \sqsubseteq \langle i, w^1 \rangle \sqcup \ldots \sqcup \langle i, w^n \rangle$ belongs to the semantic alignment drawn from the communication product of the interaction models* IM_1 *and* IM_2*. If $\langle j, v \rangle \sqsubseteq \langle i, w \rangle [t]$ is computed through the alignment protocol, $w = w^k$ for some $k = 1, \ldots, n$.*

The communication product then represents a boundary of the semantic alignment that agents can reach through the alignment protocol.

Example. *With the help of the alignment protocol, the travel agent is able to compute, among others, the following semantic relations:*

$$\begin{aligned}\langle a, \texttt{flight}\rangle &\sqsubseteq \langle b, \texttt{flight}\rangle\,[1.0]\\ \langle a, \texttt{accommodation}\rangle &\sqsubseteq \langle b, \texttt{hotel}\rangle\,[1.0]\\ \langle a, \texttt{choice}\rangle &\sqsubseteq \langle b, \texttt{myFlight}\rangle\,[r_1]\\ \langle a, \texttt{choice}\rangle &\sqsubseteq \langle b, \texttt{myHotel}\rangle\,[r_2]\end{aligned}$$

with $r_1 + r_2 = 1$. *The customer can compute, among others, the following:*

$$\begin{aligned}\langle b, \texttt{flightOutcome}\rangle &\sqsubseteq \langle a, \texttt{result}\rangle\,[1.0]\\ \langle b, \texttt{hotelOutcome}\rangle &\sqsubseteq \langle a, \texttt{result}\rangle\,[1.0]\\ \langle b, \texttt{flightSummary}\rangle &\sqsubseteq \langle a, \texttt{reservationSummary}\rangle\,[1.0]\\ \langle b, \texttt{hotelSummary}\rangle &\sqsubseteq \langle a, \texttt{reservationSummary}\rangle\,[1.0]\end{aligned}$$

6. Experimentation

Section 4 includes a formalisation of I-SSA whereas Section 5 comprises the description of an alignment protocol agents can follow to put I-SSA into practice. Here we present and analyse experimental results.

We set out to answer two Research Questions:

1. Is there a gain in communication accuracy —measured in the number of successful interactions (interactions reaching a final state)— by repeated semantic alignment through a meta-level alignment protocol and use of a matching mechanism?
2. If so, how many repeated interactions between two agents are needed in order to get sufficiently good alignments —measured in the probability of a successful interaction?

The experimentation design opens the section followed by a presentation of its execution and evaluation. A thorough statistical analysis completes it.

6.1. Experimentation Design

In this section the experiment design is explained. The alignment protocol is implemented in SICStus Prolog Release 4.0.7 [33] and random operations are executed with the SICStus Prolog random library.

In order to overcome the lack of sufficiently complex examples on which to run our implementation, we have proceeded as follows. First, an abstract alphabet made up of arbitrary illocutions and special transitions is generated. Second, a regular expression is built upon this alphabet and a previously fixed number of Kleene star, concatenation and alternation operators. Finally, the regular expression is compiled into an automaton —not necessarily minimal— making use of the FSA utilities toolbox [34]. This process is illustrated in Figure 7. Table 1 shows the variables considered in this process and the range of values they may take.

Name	Variable	Range
No. of illocutions	N_{ill}	$\mathbb{N}$
No. of illocutionary particles	N_{ip}	$\mathbb{N}$
No. of roles	N_{role}	$\mathbb{N}$
No. of messages	N_{msg}	$\mathbb{N}$
No. of special transitions	N_{spt}	$\mathbb{N}_0$
No. of Kleene star operators	N_{star}	$\mathbb{N}_0$
No. of concatenation operators	N_{con}	$\mathbb{N}_0$
No. of alternation operators	N_{alt}	$\mathbb{N}_0$

Table 1: Relevant variables when generating abstract interaction models ($\mathbb{N}$ stands for the set of all positive integers, and $\mathbb{N}_0$ for the set of non-negative integers, i.e., $\mathbb{N}_0 = \mathbb{N} \cup \{0\}$).

In order to execute the simulations we first had to give values to the variables above. Nonetheless, there was no need to choose bounds for the number of illocutions N_{ill} as we can always find lower and upper bounds in terms of the rest of variables. It is easy to prove that N_{ill} has this lower bound:

$$N_{ill} \geq \max\{N_{ip}, N_{msg}, \left\lceil \frac{N_{role}}{2} \right\rceil + 1\} \tag{3}$$

Indeed we must have more illocutions than illocutionary particles, otherwise some of them would be discarded, ditto messages and roles (recall that each illocution has two roles, namely, the sender's and receiver's roles).

Before presenting an upper bound for N_{ill}, we need to explain how regular expressions are built. It is straightforward to check that any expression of n binary operators —concatenations and alternations— has less than

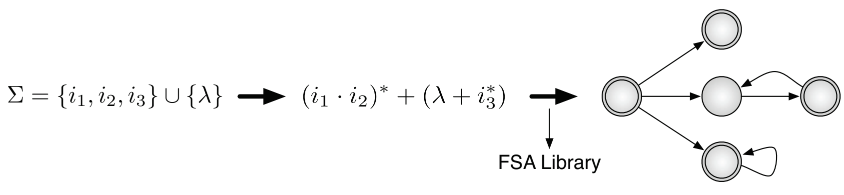


Figure 7: Process of generating abstract interaction models

$n + 1$ distinct alphabet symbols (the number of Kleene star operators is not relevant in this regard). In our case, these symbols may be illocutions or special transitions. If n_{con} and n_{alt} are the number of concatenation and alternation operators included in a regular expression r, respectively, then there exist $n_{con}+n_{alt}+1$ placeholders in r to be filled with alphabet symbols. These correspond to leaves in a tree representation of the regular expression (see Figure 8). In our implementation, operators are randomly chosen, and placeholders are randomly filled with either illocutions or special transitions.

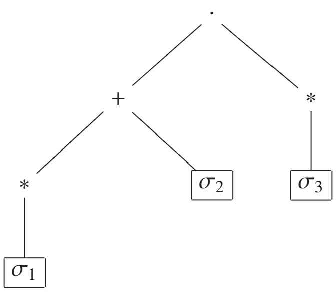


Figure 8: Tree representation of $r = (\sigma_1^* + \sigma_2) \cdot \sigma_3^*$

Let us represent by N_{leaf} the number of leaves of a regular expression built with our implementation. Since $N_{leaf} = N_{con} + N_{alt} + 1$ we have:

$$N_{ill} + N_{spt} \leq N_{con} + N_{alt} + 1 \tag{4}$$

while putting together (3) and (4):

$$\max\{N_{ip}, N_{msg}, \left[\frac{N_{role}}{2}\right] + 1\} \leq N_{ill} + N_{spt} \leq N_{con} + N_{alt} + 1$$

Thus, N_{ill} is lower and upper bounded if we give values for the variables $N_{ip}, N_{msg}, N_{role}, N_{spt}, N_{con}$ and N_{alt}.

Table 2 summarises the ranges of the variables taken into account in our simulations. We have chosen upper bounds that cover interaction models that have been actually deployed in several regulated environments (see, e.g., [35, 36, 37, 38]). Notice that N_{ill} is not included this time in Table 2.

6.2. Execution and Evaluation

Recall that in our model agents consider all foreign messages semantically different *a priori*, even if they match syntactically local ones. It justifies our decision to let agents follow the same interaction model, since agents will deal with this situation as if they conform to disparate models.

Two experiments were performed. In Experiment 1, we simulated two agents interacting via the alignment protocol and taking decisions in accord with the first, second and third matching criteria (with contamination factor $s = 0.1$), and also without applying the matching mechanism (no update of frequency distributions). Matching criteria were conditional to $\{I_P = \iota\}$.

Name	**Variable**	**Range**
No. of illocutionary particles	N_{ip}	[1..15]
No. of roles	N_{role}	[1..15]
No. of messages	N_{msg}	[1..100]
No. of special transitions	N_{spt}	[0..5]
No. of Kleene star operators	N_{star}	[0..100]
No. of concatenation operators	N_{con}	[0..100]
No. of alternation operators	N_{alt}	[0..100]

Table 2: Ranges of relevant variables when generating interaction models

We ran our implementation in series of $N = 2^n$ interactions, $n \in [1..10]$ (so the maximum number of consecutive interactions was 1024). Each of the series was completed 50 times, each time counting the number of failures and finally calculating the average $F = F(N)$. In order to compare the performance of the different matching criteria, we computed the ratio of failures to interactions, that is, $R = \frac{F}{N}$.

Experiment 1 was performed on the basis of five interaction models of varying complexity. Table 3 presents the parameter choices.

	imodel1	*imodel2*	*imodel3*	*imodel4*	*imodel5*
N_{ill}	15	20	30	50	100
N_{ip}	1	1	2	1	4
N_{role}	1	2	3	1	5
N_{msg}	5	10	15	40	80
N_{spt}	0	0	2	0	2
N_{star}	2	5	10	15	20
N_{con}	10	15	20	30	50
N_{alt}	15	10	25	25	80

Table 3: Generated interaction models

The results of Experiment 1 are shown in Figure 9. All the matching criteria did better than the case of no update of frequency distributions. This gives support to a positive answer to Research Question 1 stated at the beginning of this section. Furthermore, the ratio of failures to interactions approaches 0.0 in all interaction models but *imodel1*. It is not surprising as in this case $N_{ip} = 1$. However, the amount of illocutionary particles is greater in *imodel5* where even the alignment protocol itself without updating frequency distributions guaranteed a low number of failures. We will take up this issue again in Section 6.3. Regarding the comparison of matching criteria, both the maximal frequency crite-

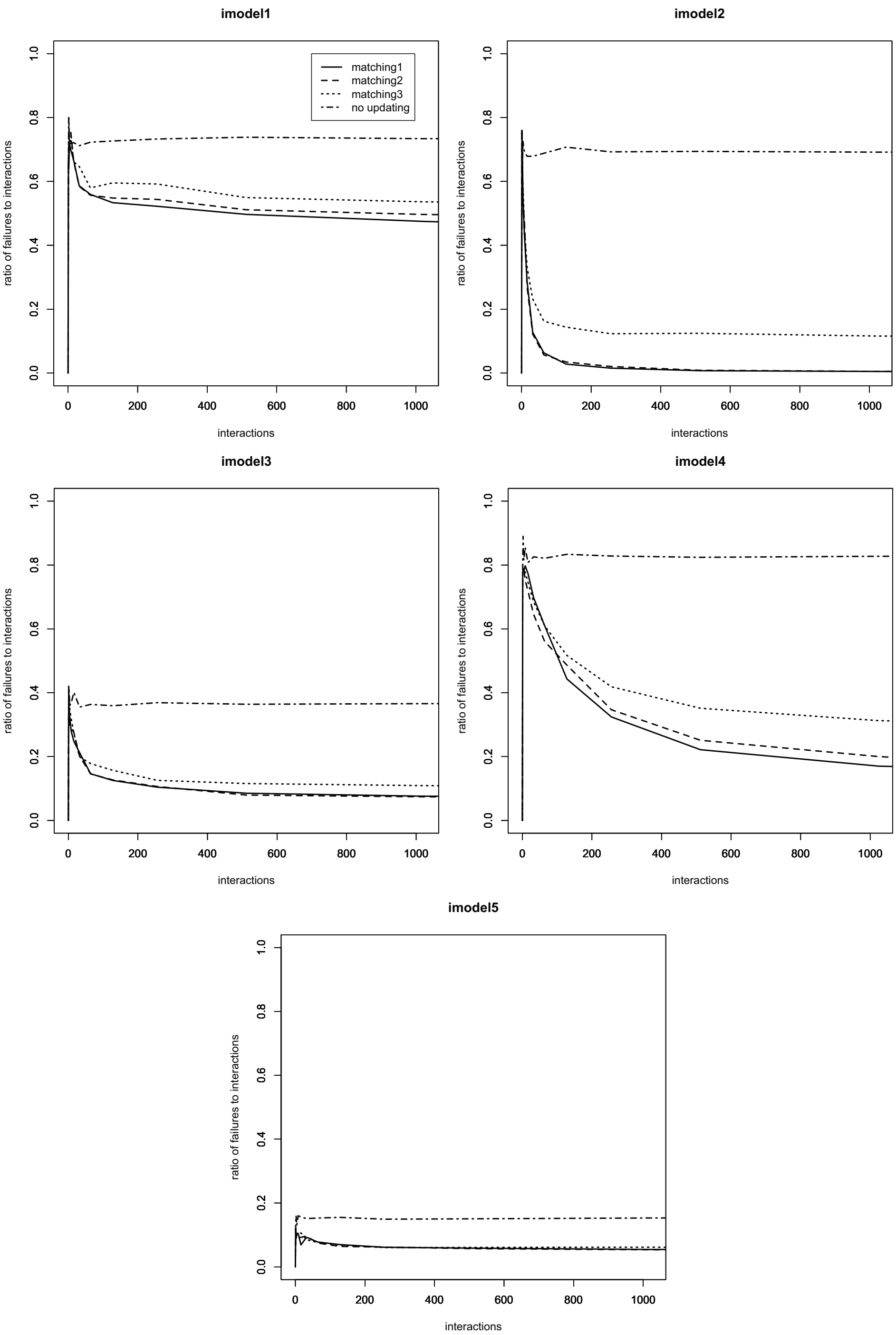


Figure 9: Results of Experiment 1

rion and the probability criterion performed better than the contaminated probability criterion as expected. The first did slightly better than the second, specially after a number of interactions.

In Experiment 2 we simulated two agents interacting as in the former so as to compute an alignment, again in series of $N = 2^n$ interactions, $n \in [1..10]$. Only the contaminated probability criterion was applied. Later this alignment was used by agents to interact 50 times with no update of frequency distributions and applying the maximal frequency criterion. This time we recorded the ratio of successes to interactions, i.e., $R = \frac{S}{50}$.

Figure 10 shows the results of Experiment 2 with the same five interaction models. In all the cases the ratio R approaches 1.0. In fact, no more than 256 interactions were needed to obtain a semantic alignment that ensured a probability close to 0.8 to interact successfully. This answers Research Question 2.

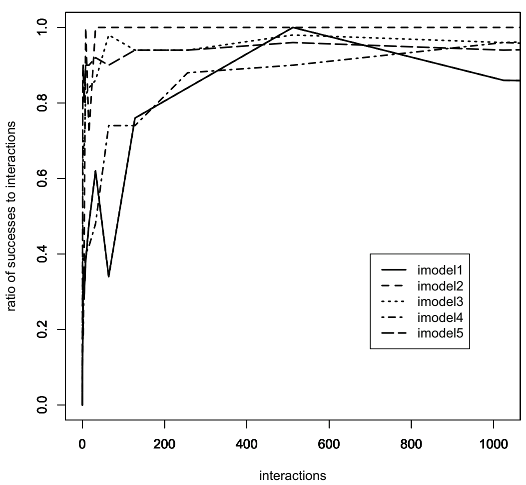


Figure 10: Results of Experiment 2

6.3. Statistical Analysis

When we look at the experimental results presented in Section 6.2, two natural questions arise: do all parameters influence the final result?, and, if so, what is their influence? Which values do better for the parameters?

In order to answer these questions, Experiment 1 was executed on the basis of a factorial generation of interaction models. The matching criterion applied was the contaminated probability criterion ($s = 0.1$). We performed a statistical analysis of the resulting experimental data by combining analysis of variance (ANOVA) with post-hoc comparisons using the so-called Tukey test [39]. The first is useful to discover whether there was a significant relation between the independent variables —parameters in the simulation— and the dependent variable —ratio of failures to interactions. The second is helpful to find out which values did better for each of the independent variables.

For ANOVA test results to be reliable, a number of conditions must be satisfied. One refers to independence in the sample. This led us to modify the input parameters, since restrictions (3) and (4) explained in Section 6.1 violate the required independence. One possible first step is to discard the number of special transitions N_{spt}. This is not a great loss, since we are more interested in studying the effect of the illocution components and the structure of the interaction model. In this way, $N_{leaf} = N_{ill}$, so that an ANOVA test can be run for each specific value of N_{ill}. The following were selected:

$$N_{ill} = 8, 16, 32, 64, 128$$

The number of alternation operators was not considered, since, as already seen, $N_{alt} = N_{ill} - N_{con} - 1$, so any statement about N_{con} has a counterpart statement about N_{alt}. We also replaced variables N_{ip} and N_{role} with a unifying variable N_{head} which accounts for the number of illocution heads.

Once a particular value of N_{ill} is selected, an upper bound for both N_{head} and N_{msg} is laid down: $1 \leq N_{head}, N_{msg} \leq N_{ill}$. Nevertheless, since an interaction model in which there are no repeated illocution heads is not interesting for the task at hand (agents would always be able to distinguish the correct message between all incoming ones, and, hence, they would not fail at all), $\frac{1}{2}N_{ill}$ is a much more effective upper bound for N_{head}. Interaction models with only one message were not taken into account either. Concerning operators, we decided to generate regular expressions with at least one operator of each type. Thus, $1 \leq N_{con} \leq N_{ill} - 2$. Kleene star operators, though, were left to be in any case lower than $\frac{1}{2}N_{ill}$. Again, we ran our implementation in series of $N = 2^n$ interactions, but this time with $n \in [1..8]$. Table 4 shows the selected values in the particular case of $N_{ill} = 32$.

Name	**Selected Values**
No. of heads	1,2,4,8,16
No. of messages	2,4,8,16,32
No. of Kleene star operators	1,2,4,8,16
No. of concatenation operators	1,2,4,8,16,30
No. of interactions	1,2,4,8,16,...,256,512

Table 4: Selected values for $N_{ill} = 32$

Before executing the ANOVA tests we verified that the resulting data had a normal distribution through a

Quantile-Quantile test, which is another precondition for the ANOVA results to be reliable. We ran five ANOVA tests —one for each value of N_{ill}— with the software environment R [40].

The ANOVA results demonstrate that all the independent variables were statistically significant with a p-value much lower than 0.05, which is the standard threshold for statistical significance tests. Therefore the ANOVA tests confirmed (or did not refute)

Hypothesis 1. *The following factors affect the number of failures:*

1. *the variety of illocution heads,*
2. *the amount of local messages,*
3. *the structure of interaction models, and*
4. *the number of interactions.*

One also expects the hypothesis below to be confirmed.

Hypothesis 2. *The following imply a lower number of failures:*

1. *a higher number of illocution heads,*
2. *a lower number of messages, and*
3. *a higher number of interactions.*

For this reason, we ran post-hoc comparisons using the Tukey test. Figure 11 shows the results of the tests in the case of $N_{ill} = 32$. For the rest of values we obtained similar results. Hence, Hypothesis 2 was confirmed too. With regard to the operators of regular expressions, the following was confirmed:

Hypothesis 3. *A lower number of concatenations implies a lower number of failures. Alternatively, a higher number of alternations implies a lower number of failures.*

Recall that our approach highly depends on the criterion followed when it comes to classify an interaction as successful or unsuccessful (see I-SSA third principle in Section 3.1). In this paper, in order for an interaction to be qualified as successful, agents must jointly reach final states. As a general rule, the more alternation operators a regular expression has, the more paths leading to final states in the interaction model, and, thus, the more chances for the agents to interact successfully. This gives an explanation to Hypothesis 3. However, different notions of success —as a consequence of, for example, more expressiveness in interaction models— would result in different hypotheses.

With respect to Kleene star operators, the Tukey tests provided confusing results. Neither a higher nor a lower number of star operators implied a lower number of failures. This fact made us think that this might not be the proper parameter to study. A second option is to look into the complexity of an interaction model by measuring its *star height* [41]. Actually, the three operators considered in this work are all involved in the star height measure. However, a preliminary experimentation suggested that this is not the right way either. It seems to be necessary to work on a specific notion of complexity appropriate for the case at hand. This task has not been addressed for this paper and has been left for future work.

7. Related Work

Other approaches share with ours the insight that semantics is fundamentally interaction-specific.

Besana and Robertson attach probabilistic values to meanings of terms that are determined by earlier and similar interactions [42]. These values are then used to predict the set of possible meanings of a message. As with our approach, meaning is defined relative to a particular interaction, but Besana and Robertson aim to reduce the search space of possible a priori mappings between ontological entities (computed in a classical sense), namely by assessing those ones with highest probability in the context of an interaction. Instead of finite-state machines the formalism adopted to model agent interactions is LCC (Lightweight Coordination Calculus) [43]. Besana and Robertson do not provide any formal grounding for their prediction reasoning. In contrast, our alignment protocol and matching criteria are based on the communication product which realises I-SSA principles.

Bravo and Velázquez discover pragmatic similarity relations among messages in different agent interaction protocols [44]. Like ours this approach is based on the analysis of transition functions in finite automata. Two input messages (transitions) are equivalent when their respective initial and final states are equal. This requires that the set of states is the same in all agent interaction protocols. Rather than separate transitions, we look at histories which allows us to capture interdependence of messages. Bravo and Velázquez do not consider any content language since in their framework messages are actually performatives. In addition, no other semantic relation but equivalence is studied, while subsumption of messages is also defined in I-SSA.

Although Rovatsos et al. do not address the problem of semantic heterogeneity, their approach has a number commonalities with I-SSA [45]. Rovatsos et al. propose a semantics for agent communication languages (ACLs)

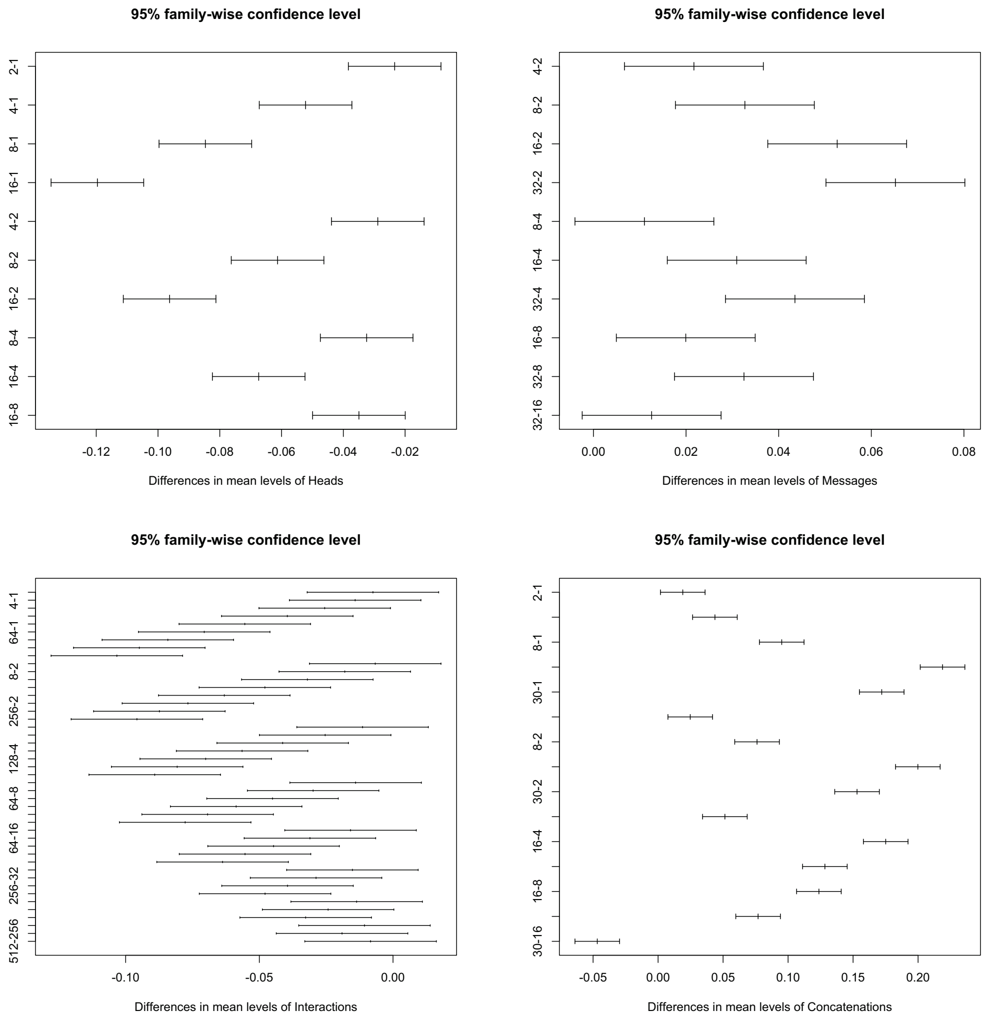


Figure 11: Results of the Tukey tests

in the context of open systems in which the meaning of a message is defined in terms of its consequences, namely, those messages and actions that are likely to follow it. It is claimed that an agent strives to reduce the uncertainty about others' communicate behaviour (entropy), and at the same time to increase her own autonomy (utility). Indeed this can be seen as an alternative to the I-SSA Principle 3*. Furthermore, like ours, this model relies on a statistical analysis of observed communication.

Our approach is reminiscent to the research of Steels in which he explores how a group of distributed agents adapt to form an ontology and a shared lexicon in an emergent, bottom-up manner, with no central control authority and only local interactions [46, 47]. This sort of self-organised emergence of shared meaning is ultimately grounded on the physical interaction of agents with the environment. In our approach, though, we have addressed the case in which agents are already endowed with a top-down engineered ontology (it can even be the same one), which they do not adapt or refine, but for which they want to find the semantic relationships with separate ontologies of other agents on the grounds of their communication according to a specific interaction model.

As with the work of Steels, our view of meaning and its role in multiagent interaction is, to a certain extent, related to the idea of a language game as put forth by Wittgenstein [48]. Interaction protocols can be seen as the game rules that constrained the moves —the words uttered— that are allowed at each state of the game. The meaning an agent attaches to a term, then, is the state transition it thinks is the result from the term's utterance in a particular speech act, according to the agent's view of the interaction and of the current interaction state. As with a language game, the guesses of what the meanings of the words are may be wrong, which will eventually lead to a breakdown of the communication: the interaction has not progressed in the direction foreseen by the interaction models of each agent. Agents can be aware of such a breakdown if they are capable to communicate to each other about the interactions themselves [49].

8. Concluding Remarks and Further Work

In this paper we have laid the formal foundations for a novel approach to tackle the problem of semantic heterogeneity in multiagent communication. We did not take the predominant stance that shared semantics is a prerequisite for successful interaction, but instead attempted to establish semantic similarity on the grounds of successful interaction itself. For this we have looked at the semantics of messages from an interaction-based viewpoint, as it arises in the context of a dialogue that unfolds according to previously specified interaction models. In our approach messages are deemed semantically related if they trigger compatible interaction-state transitions, where compatibility means that the interaction progresses in the same direction for each agent — albeit their interaction views (i.e., their own interaction models) may be more constrained than the interaction that is actually happening.

An advantage of this approach is that it takes meaning into account that is very interaction-specific and which cannot be derived neither from a local ontology nor from sources that are external to the interaction. In this sense we see our approach as a complement to current state-of-the-art matching techniques as it may provide valuable information for pruning the search space or disambiguating the results of candidate semantic alignments computed with today's ontology-matching technology.

The viability of the I-SSA approach has been evinced through our abstract experimentation and statistical study. Through the combination of analyses of variance and Tukey tests we have been able to identify which factors —number of illocution heads, messages and interactions— have an influence on the total amount of failed interactions, and which values do better for each of the independent variables.

The actual applicability, however, will depend largely on how each potential application domain conforms to the underlying assumptions of I-SSA, namely that (i) agents are part of a regulatory environment, (ii) they may engage repeatedly in the same sort of dialogue, and (iii) they are able to communicate each other some shared notion of dialogue success. The first two hold for most implementations of the electronic institution paradigm, which range from auctioning [35] and electronic markets [36] to public-policy management [37] and online dispute resolution [38]. As it stands, though, I-SSA has to go beyond current representational limitations for it to be readily applicable in these domains.

So far we have taken content messages of illocutions to be grounded atomic sentences, but we need to extend the content language to cope with variables —and hence move into some degree of first-order expressiveness— if we want interaction models to capture any realistic application as those mentioned above. Potential semantic relationships would then need to be expressed between complex, structured terms instead of simple, propositional constants. How this could be done for conventional ontology matching, starting from the semantic alignment at the terminological level, has been investigated by Giunchiglia et al. applying ideas derived from the theory of abstraction combined with tree edit distance algorithms [50]. An approach such as the one put forward by I-SSA, however, would have to tackle this problem the other way around, investigating how relationships at the structural level, determined by the actual use of complex terms in the context of a particular interaction, relate to the semantic alignment at the terminological level.

Also, finite-state automata are currently not expressive enough as to capture the complexity of interaction in these domains, and richer interaction modelling formalisms, such as electronic institutions, will have to be considered. If we initially constrained ourselves to finite-state automata and propositional languages, it was because we wanted to check the viability of our approach with a simple interaction model formalism before moving to more expressive representation languages and richer specifications of interactions.

Actually, a formalism of lesser expressivity, such as a finite-state automaton, allows for sharing only a rather weak notion of dialogue success —our third assumption— as reflected in Principle 3* in Section 3.1. But other instantiations of the more general Principle 3 could have been proposed, particularly if the interaction modelling formalism allowed for expressing them. One possible extension could be to take into account commitments made by agents while interacting (such as payments or deliveries) and to check whether these commitments have been fulfilled for an interaction to be considered successful. In fact, commitments would enable agents to have checkpoints in mid-interactions, and, thus, to detect failures earlier, before reaching a final state. Therefore we expect richer interaction modelling formalisms to actually yield even more accurate alignments, as those already achieved by I-SSA in our experimentation.

Exploring the applicability of I-SSA would first need to address the above mentioned issues and its implications, and we are currently looking into them in the context of the *agreement computing* paradigm [51] —a distributed interaction-centred computational paradigm based on an explicit notion of agreement between computational entities. More specifically, we are exploring eventual semantic mismatches in the context of a two-agent negotiation protocol as the one developed for the *mWater* system [37] —a regulated environment where autonomous agents trade rights for the use of water in a closed basin— and for which the fulfilment of commitments are paramount to the understanding of the protocol. We claim that, even with shared ontologies, mismatches are susceptible of arising during interaction time, because no ontology can foresee all potential uses of terminology. In these cases an I-SSA alignment could be used to make explicit a lack of agreement at the semantic level, which would need to be included into the ontology for subsequent negotiations.

One of the main characteristics of I-SSA is that it is fully unaware of ontological information. Semantic alignment conforms solely to the agents' use of messages while interacting, though ontological information is actually implicit in this usage. Such limitation may be sensible in situations in which ontologies are not open for inspection, but nothing prevents agents from taking advantage of their own ontological information. Indeed an agent could reason about the relations between their own messages when matching a received one. Our choice not to assume agents with previously formed individual ontologies was motivated —as with the interaction modelling formalism— by our desire to focus on the viability of a purely interaction-centred approach. I-SSA was initially driven by the fact that most of the current state-of-the-art matchers put little emphasis on pragmatics. But more than a replacement or an alternative technique, we believe that I-SSA is a good complement for these matchers, and it is in our mind to work on this line in the future.

To conclude, we would like to point out that I-SSA's current matching mechanism only keeps track of past successful interactions, but unsuccessful interactions are simply discarded. Clearly, this is a great loss, since agents could also learn from past matching mistakes. The problem is that it is not straightforward to figure out which matching is responsible of a failure, or if we should blame one agent or another. Once more a probabilistic approach seems to be appropriate for this matter, attaching values to matching elements that vary as more interactions are completed, regardless of whether they are successful or unsuccessful. This should considerably improve the matching mechanism in terms of learning speed.

Acknowledgements

This work was supported under the grant 2009-SGR-1434 from the Generalitat de Catalunya, and under the projects *Agreement Technologies* (CONSOLIDER INGENIO 2010 CSD2007-00022) and CBIT (TIN2010-16306) funded by the Spanish Ministry of Science and Innovation.

The authors also would like to thank the anonymous reviewers for their valuable feedback.

References

[1] T. Berners-Lee, J. Hendler, O. Lassila, The Semantic Web, Scientific American 284 (5) (2001) 34–43.

[2] S. Staab, R. Studer (Eds.), Handbook on ontologies, International Handbooks on Information Systems, Springer, 2004.

[3] J. Hendler, F. van Harmelen, The Semantic Web: webizing knowledge representation, in: Handbook of Knowledge Representation, Foundations of Artificial Intelligence, Elsevier, 2008, Ch. 21, pp. 821–839.

[4] I. Horrocks, Ontologies and the Semantic Web, Communications of the ACM 51 (2) (2008) 58–67.

[5] M. N. Huhns, M. P. Singh, Ontologies for agents, IEEE Intelligent Systems 1 (6) (1997) 81–83.

[6] J. Hendler, Agents and the Semantic Web, IEEE Intelligent Systems 16 (2) (2001) 30–37.

[7] Y. Kalfoglou, H. Alani, M. Schorlemmer, C. Walton, On the emergent Semantic Web and overlooked issues, in: Proc. of the 3rd International Semantic Web Conference (ISWC 2004), Vol. 3298 of LNCS, Springer, 2004, pp. 576–590.

[8] D. Robertson, et al., Open Knowledge — coordinating knowledge sharing through peer-to-peer interaction, in: Proc. of the 1st International Workshop on Languages, Methodologies and Development Tools for Multi-Agent Systems (LADS 2007), Vol. 5118 of LNCS, Springer, 2008, pp. 1–18.

[9] Y. Kalfoglou, M. Schorlemmer, Ontology mapping: the state of the art, The Knowledge Engineering Review 18 (1) (2003) 1–31.

[10] J. Euzenat, P. Shvaiko, Ontology matching, Springer, 2007.

[11] T. Winograd, F. Flores, Understanding computers and cognition, Addison-Wesley, 1986.

[12] P. Bouquet, L. Serafini, S. Zanobini, Semantic coordination: A new approach and an application, in: Proc. of the 2nd International Semantic Web Conference (ISWC 2003), Vol. 2870 of LNCS, Springer, 2003, pp. 130–145.

[13] F. Giunchiglia, P. Shvaiko, Semantic matching, The Knowledge Engineering Review 18 (3) (2004) 265–280.

[14] V. Mascardi, A. Locoro, P. Rosso, Automatic ontology matching via upper ontologies: a systematic evaluation, IEEE Transactions on Knowledge and Data Engineering 22 (5) (2010) 609–623.

[15] F. Lin, K. Sandkuhl, A survey of exploiting WordNet in ontology matching, in: 2nd IFIP International Conference on Artificial Intelligence in Theory and Practice (IFIP AI 2008) at the 20th IFIP World Computer Congress (WCC 2008), Proceedings, Vol. 276 of IFIP, Springer, 2008, pp. 341–350.

[16] M. Sabou, M. d'Aquin, E. Motta, Exploring the Semantic Web as background knowledge for ontology matching, in: Journal on Data Semantics XI, Vol. 5383 of LNCS, Springer, 2008, pp. 156–190.

[17] J. van Diggelen, R. Beun, F. Dignum, R. van Eijk, J.-J. Meyer, Ontology negotiation: goals, requirements and implementation, International Journal of Agent-Oriented Software Engineering 1 (1) (2007) 63–90.

[18] L. Laera, I. Blacoe, V. Tamma, T. R. Payne, J. Euzenat, T. Bench-Capon, Argumentation over ontology correspondences in MAS, in: Proc. of the 6th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2007), IFAAMAS, 2007, pp. 1285–1292.

[19] C. Trojahn, P. Quaresma, R. Vieira, Conjunctive queries for ontology based agent communication in MAS, in: Proc. of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2008), IFAAMAS, 2008, pp. 829–836.

[20] P. Doran, V. Tamma, T. R. Payne, I. Palmisano, Dynamic selection of ontological alignments: a space reduction mechanism, in: Proc. of the 21st International Joint Conference on Artificial Intelligence (IJCAI 2009), 2009, pp. 2028–2033.

[21] F. McNeill, A. Bundy, Dynamic, automatic, first-order ontology repair by diagnosis of failed plan execution, International Journal of Semantic Web and Information Systems 3 (3) (2007) 1–35.

[22] J. L. Arcos, M. Esteva, P. Noriega, J. A. Rodríguez-Aguilar, C. Sierra, Engineering open environments with electronic institutions, Engineering Applications of Artificial Intelligence 18 (2) (2005) 191–204.

[23] M. Atencia, M. Schorlemmer, I-SSA: Interaction-Situated Semantic Alignment, in: On the Move to Meaningful Internet Systems, OTM 2008, Proceedings, Vol. 5331 of LNCS, Springer, 2008, pp. 445–455.

[24] M. Atencia, M. Schorlemmer, Reaching semantic agreements through interaction, in: 4th AIS SigPrag International Pragmatic Web Conference Track (ICPW 2009) at the 5th International Conference on Semantic Systems (i-Semantics'09), JUCS Conference Proceedings Series, Verlag der Technischen Universität Graz, 2009, pp. 726–737.

[25] A. Wierzbicka, English speech act verbs: a semantic dictionary, Academic Press, 1989.

[26] Y. Labrou, Semantics for an agent communication language, Ph.D. thesis, Computer Science and Electrical Engineering Department (CSEE), University of Maryland Graduate School, Baltimore, Maryland, USA (1996).

[27] S. Mac Lane, Categories for the working mathematician, 2nd Edition, Springer, 1998.

[28] H. H. Do, E. Rahm, COMA — a system for flexible combination of schema matching approaches, in: Proc. of the 28th International Conference on Very Large Data Bases (VLDB 2002), Morgan Kaufmann Publishers, 2002, pp. 610–621.

[29] N. Jian, W. Hu, G. Cheng, Y. Qu, Falcon-AO: aligning ontologies with Falcon, in: Integrating Ontologies Workshop at K-CAP 2005, Vol. 156 of CEUR Workshop Proceedings, CEUR-WS.org, 2005, pp. 85–91.

[30] J. Euzenat, P. Valtchev, Similarity-based ontology alignment in OWL-Lite, in: Proc. of the 16th European Conference on Artificial Intelligence (ECAI 2004), IOS Press, 2004, pp. 333–337.

[31] M. Atencia, Semantic alignment in the context of agent interaction, Vol. 46 of Monografies de l'IIIA, CSIC, 2011.

[32] G. A. Miller, WordNet: a lexical database for English, Communications of the ACM 38 (11) (1995) 39–41.

[33] M. Carlsson, SICStus Prolog user's manual. Release 4.0.7, Tech. rep., Swedish Institute of Computer Science (April 2009).

[34] G. van Noord, FSA utilities: a toolbox to manipulate finite-state automata, in: Proc. of the 1st International Workshop on Implementing Automata (WIA 1996), Vol. 1260 of LNCS, Springer, 1996, pp. 87–108.

[35] P. Noriega, The fishmarket metaphor, Vol. 8 of Monografies de l'IIIA, CSIC, 1999.

[36] G. Cuní, M. Esteva, P. Garcia, E. Puertas, C. Sierra, T. Solchaga, MASFIT: Multi-Agent Systems for FIsh Trading, in: Proc. of the 16th European Conference on Artificial Intelligence (ECAI 2004), IOS Press, 2004, pp. 710–714.

[37] A. Garrido, A. Giret, P. Noriega, mWater: a sandbox for agreement technologies, in: Proc. of the 12th International Conference of the Catalan Association for Artificial Intelligence (CCIA 2009), Vol. 202 of Frontiers in Artificial Intelligence and Applications, IOS, 2009, pp. 252–261.

[38] P. Noriega, C. López, Towards a platform for online mediation, in: Legal and Negotiation Decision Support Systems (LDSS 2009), a post-conference workshop of the 12th International Conference on ArtificiaI Intelligence and Law, Proceedings, Vol. 4 of IDT Series, 2009, pp. 69–76.

[39] P. R. Cohen, Empirical methods for artificial intelligence, The MIT Press, 1995.

[40] P. Dalgaard, Introductory Statistics with R, 2nd Edition, Statistics and Computing, Springer, 2008.

[41] L. C. Eggan, Transition graphs and the star-height of regular events, Michigan Mathematical Journal 10 (4) (1963) 385–397.

[42] P. Besana, D. Robertson, How service choreography statistics reduce the ontology mapping problem, in: Proc. of the 6th International Semantic Web Conference (ISWC 2007), Vol. 4825 of LNCS, Springer, 2007, pp. 44–57.

[43] D. Robertson, Multi-agent coordination as distributed logic programming, in: Proc. of the 20th International Conference on Logic Programming (ICLP 2004), Vol. 3132 of LNCS, Springer, 2004, pp. 416–430.

[44] M. Bravo, J. Velázquez, Discovering pragmatic similarity relations between agent interaction protocols, in: On the Move to Meaningful Internet Systems, OTM 2008, Proceedings, Vol. 5333 of LNCS, Springer, 2008, pp. 128–137.

[45] M. Rovatsos, M. Nickles, G. Weiß, Interaction is meaning: a new model for communication in open systems, in: Proc. of the 2nd International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2003), ACM, 2003, pp. 536–543.

[46] L. Steels, The origins of ontologies and communication conventions in multi-agent systems, Journal of Autonomous Agents and Multi-Agent Systems 1 (2) (1998) 169–194.

[47] L. Steels, Language games for autonomous robots, IEEE Intelligent Systems 16 (5) (2001) 16–22.

[48] L. Wittgenstein, Philosophische Untersuchungen, Macmillan, New York, 1953.

[49] M. Schorlemmer, M. Atencia, Taking interaction ontologically prior to meaning, in: Proc. of the 7th European Conference on Computing and Philosophy, International Association for Computing and Philosophy, 2009, pp. 160–161.

[50] F. Giunchiglia, M. Yatskevich, F. McNeill, P. Shvaiko, J. Pane, P. Besana, Approximate structure preserving semantic matching, in: Proc. of the 18th European Conference on Artificial Intelligence (ECAI 2008), Vol. 178 of Frontiers in Artificial Intelligence and Applications, IOS Press, 2008, pp. 743–744.

[51] C. Sierra, V. Botti, S. Ossowski, Agreement computing, Künstliche Intelligenz 25 (2011) 57–61.