Simulating dialogues with finite state transition systems

Maria Adela Grando [1]

Rovira i Virgili University, Research Group on Mathematical Linguistics
Pl. Imperial Tárraco 1, 43005 Tarragona, Spain
mariaadela.grando@estudiants.urv.es

Abstract

In this paper we contribute to Dialogue Theory introducing two formal frameworks inspired in Formal Language Theory, for the simulation of dialogues as finite state transition systems. In the first framework agents are provided with a share memory limited to an stack of locutions. An stack proves to be adequate for simulating goal-oriented dialogues. The stack can save during the dialogue all the unachieved goals. The top of the stack corresponds to the last unachieved goal which is removed from the stack when it is solved. But this framework can not simulate all the dialogues based on the notion of social semantics. The principle behind a system based on social semantics is that when speakers utter locutions they state publicly their knowledge and they publicly acquire commitments. The truth of an speaker's speech acts in general can not be verified, but at least an conversant's speech consistency can be assessed inspecting the social commitments he acquired in the dialogue. In order to provide the framework with the expressive power required to simulate systems based on the notion of social semantic we extended the stack of locutions to an string of symb ols over some finite alphabet.

Keywords: Dialogue Theory, Protocol of communication, Formal Language Theory, Linguistics

[1] Supported by research grant "Programa Nacional para la Formación del Profesorado Universitario" from the Ministry of Education, Culture and Sports of Spain

Resumen

En este trabajo contribuimos a la Teoría de Diálogos definiendo dos marcos formales inspirados en la Teoría de Lenguajes Formales, para la simulación de diálogos como sistemas de transiciones de estados finitos. En el primer marco formal los agentes son provisto con una memoria compartida que se limita a una pila que almacena locuciones. Esta pila prueba ser adecuada para simular diálogos guiados por metas. La pila puede almacenar durante el diálogo las metas no conseguidas. El tope de la pila corresponde a la última meta a lograr, la cual es removida de la pila cuando es alcanzada. Pero este marco formal no puede simular todos los diálogos basados en la noción de de semántica social. El principio subyacente en todo sistema basado en la semántica social es que cuando los participantes en el diálogo hablan, están dando a conocer públicamente su conocimiento y adquiriendo compromisos. La verdad de lo expresado por un hablante en general no puede ser verificado, pero al menos la consistencia de su discurso puede ser corroborada a través de los compromisos sociales que ha adquirido. Con el propósito de proveer al marco definido con el poder expresivo requerido para simular diálogos basados en la noción de semántica social, extendemos la pila de locuciones a una cadena de símbolos sobre un alfabeto finito.

Palabras clave: Teoría de Diálogo, Protocolo de comunicación, Teoría de Lenguajes Formales, Lingüística

Resum

En aquest article es fa una contribució a la Teoria del Diàleg mitjanant la introducció de dos models formals inspirats en la Teoria de Llenguatges Formals, per la simulació de diàlegs com a sistemes de transició d'estats finits. En el primer d'aquests models, es descriuen agents equipats amb una memória compartida limitada a una pila de locucions. Es demostra que una pila és suficient per simular diàlegs orientats a un objectiu. La pila pot emmagatzemar durant el diàleg tots els objectius que no han estat assolits. La part superior correspon al darrer d'aquests objectius que no s'han aconseguit, i que s'esborra tan aviat aquest es resol. Però aquest model no pot simular tots els diàlegs basats en la noció de semàntica social. El principi subjacent a un sistema basat en la semàntica social és que quan els parlants generen locucions afirmen públicament el seu coneixement i adquireixen compromisos públicament. En realitat, un acte de parla no pot ser verificat, però al menys se'n pot fer una valoració examinant els compromisos socials que s'han adquirit durant el diàleg. Per tal de formalitzar un model amb el poder expressiu necessari per simular sistemes basats en la noció de la semàntica social s'ha estès la pila de locucions fins a una cadena de símbols sobre un alfabet finit.

Mots clau: Teoria del diàleg, Protocol de comunicació, Teoria de Llenguatges Formals, Lingüística

Table of Contents

1. Introduction

The conjecture that the background of human dialogues is not formalizable (Dahlabäck 1992) (Shneiderman 1980) has produced a reformulation of goals in Dialogue Theory. Currently one of the main goals of Dialogue Theory is the definition of effective human to machine and machine to machine goal-oriented dialogue protocols.

Dialogue protocols specify patterns of communication between speakers in a society and they are used to state their social norms. Protocols have be defined by means of finite automata (Vasconcelos 2004, Esteva 2001), hight level petri nets (Holvoet 1998) (Moldt 1997) (Purvis 1996), diagrams provided by the Unified Modeling Language (Woods 1999) (Parunak 2000) (Wei 2001) (Koningand 2001), logic (Woods 1999) (Genereth 1994), and process descriptions (Robertson 2004) (Walton 2004). A society protocol is examined in-advance by an speaker in order to decide if he joins or not the corresponding society. The protocol also acts as a guide for the speaker to follow once they are operating within the society. For instance a 2-party protocol takes place when we talk by telephone respecting social norms of the type: the speakers take turns to talk, overlapping is possible but undesirable, any party can decide to finish the conversation in any moment, it is desirable but not mandatory that the party that wants to finish the dialogue let the other party know it before finishing it, any party can assume that the other party finished the dialogue without notification after a prudential time without dialogue, etcetera.

The aim of this paper is to contribute to Dialogue Theory from the perspective of Formal Language Theory, introducing two formal frameworks for the definition of dialogue protocols as finite state transition systems. Each framework we propose defines a generic, abstract and well defined formalism in which protocols can be specified, compared and evaluated in the same notation in a precise way. According to Linguistic studies the frameworks that we introduce show some features characteristic of human-like talk. Our frameworks can be studied from the perspective of Formal Language Theory, analyzing for instance the expressive power, computational (time and space) complexity and the decidability of the problems of: determining if a dialogue instance satisfies a given protocol, specifying the set of dialogue instances (language) generated by a protocol, determining if the language generated by a protocol is finite, infinite or empty, etcetera.

We introduce in section 2 a framework that we call Stack Finite State Transition ($StackFST$) system. Framework $StackFST$ allows to specify dialogue protocols where the number of agents is fixed during design time, agents are provided with fix sets of private believes and fix sets of computable decision procedures, the interaction model is described by a finite state transition system whose transitions are conditionals and labeled with locutions, and the share knowledge is saved in an stack of locutions. The states of the automata correspond to possible stages of the conversation and the transitions to dialogue moves. For a labeled transition to be triggered the precondition associated to the locution that labels it has to be satisfied. And the preconditions of the locutions are formulas over some logic which are evaluated according to the agent knowledge and the locution saved in the top of the memory stack. If a transition is triggered a new state is reached and some of the following operations can be performed over the stack of interchanged locutions: replace the locutio n in the top for a new locution, or remove the locution in the top.

In section 2.1 we provide an example of argumentation-based dialogue protocol specified in framework $StackFST$. Argumentation Theory is a subfield of Dialogue Theory based on the interchange of arguments and contraarguments between speakers with the purpose of arriving to conclusions even in presence of incomplete or inconsistent information. According to (Parsons 2003) the principle behind social semantics in dialogues is that when speakers utter locutions they state publicly their knowledge. The truth of an speaker's speech acts can not necessarily be verified, but at least an speaker's consistency can be assessed inspecting the commitments he acquired during the dialogue. After presenting the example we show that when the agents are limited to reasoning over an stack of locutions they can loose information corresponding to the social commitments acquired by the participants in the conversation, what can lead to wrong reasonings and conclusions.

The previous analysis justifies the introduction in section 3 of an extension of framework *StackFST* that we call *Conversational Finite State Transition (ConvFST)* system. A preliminary version of this framework can be found in (Grando 2007). In framework *ConvFST* the stack is replaced by a common string that every active agent can access to. With this string it is possible to simulate social semantics and also to simulate what in Pragmatics is called dialogue context: a common repository of agents communication actions that constitute their interface to the other agents (observable behavior). In section 3.1 we exemplify the use of this framework with the specification of an information-seeking dialogue protocol.

Finally in section 4 we present some conclusions and some proposals of future work.

2. Framework *StackFST*

In this section we introduce a framework that we call *Stack Finite State Transition (StackFST)* systems to formally specify dialogue protocols between a fix number of $n \geq 1$ agents. The agents share a common knowledge base that corresponds to an stack of locutions. This means that each time the agents inspect the stack looking for a saved locution in position $s \geq 0$, they loose the locutions that were uttered from the position $s+1$ to the top of the memory stack. Each agent A_i, $1 \leq i \leq n$, is provided with a private knowledge base K_i containing their beliefs. These knowledge bases are fixed during the dialogue, it means that agents do not learn through the dialogue. The dialogue moves are described by a finite state transition system where the states correspond to stages in the conversation and transitions are conditional and labeled by locutions from a locution set. For a labeled transition to be triggered the precondition associated to the locution that labels it has to be satisfied. We assume that for every locution from the set of interchanged locutions there is a decidable procedure to check if its associated precondition is satisfied. And the preconditions of the locutions are formulas over some logic which are evaluated according to the agent knowledge and the top of the stack corresponding to the share knowledge base. If a transition is triggered a new state is reached and the top of the stack can be removed or changed for a new locution.

Formally:

Definition 1 *A dialogue protocol $W \in StackFST$ of degree $n \geq 1$ is a tuple:*

$$W = \langle K_{id_1}, ..., K_{id_n}, \Sigma, Q, LS_{\mathcal{L}}, \Gamma, \delta, q_0, SK_0, F \rangle$$

where:

- Σ *is a finite set of symbols;*
- $K_{id_i} \in \Sigma$*, for all* $1 \leq i \leq n$*, are the initial believes of agent* A_{id_i}*;*
- Q *is a finite set of states;*
- $q_0 \in Q$ *is the initial state,*
- $F \subseteq Q$*, are the final states, and*
- $LS_{\mathcal{L}}$ *is a finite set of locutions. Considering that SK is the content of the stack and* $top(SK)$ *is the last added element or top of SK, then:*
  $$LS_{\mathcal{L}} = \left\{ \begin{array}{l} \rho_{id_i}(\phi^{(m)}) \mid m \geq 0 \wedge 1 \leq i \leq n \wedge prec(\rho_{id_i}(\phi^{(m)})) \textit{ is a wff in logic } \mathcal{L} \\ \textit{with free variables ranging over } \{\phi^1, ..., \phi^m, K_{id_i}, top(SK)\} \end{array} \right\}$$
  The locution $\rho_{id_i}(\phi^{(m)}) \in LS_{\mathcal{L}}$ *with constants* ρ, id_i *and terms* $\phi^{(1)}, ..., \phi^{(m)}$ *has associated a well formulated formula* $prec(\rho_{id_i}(\phi^{(m)}))$ *in logic* $\mathcal{L}$ *called precondition, such that checking the satisfaction of* $prec(\rho_{id_i}(v^{(m)}))$ *is decidable for all parameter values* $v^{(m)}$*.*
- $\Gamma = LS_{\mathcal{L}}$
- $SK_0 \in LS_{\mathcal{L}}$ *is the initial stack symbol;*
- δ *is a finite transition relation* $(Q \times LS_{\mathcal{L}} \times LS_{\mathcal{L}}) \rightarrow 2^{Q \times LS_{\mathcal{L}}^*}$*.*

Considering that a sequence of locutions $\rho_i(v^{(m)})\beta \in (LS_{\mathcal{L}})^+$ with parameter values $v^{(m)}$ is being processed, for any $q \in Q$, $\rho_i(v^{(m)}) \in LS_{\mathcal{L}}$, $\beta \in (LS_{\mathcal{L}})^*$, $\tau_j(y^{(k)}) \in LS_{\mathcal{L}}$ the top of the stack with parameter values $y^{(k)}$, the interpretation of

$$\delta(q, \rho_i(v^{(m)}), \tau_j(y^{(k)})) = \{(p_1, \gamma_1(s_1^{(t_1)})), ..., (p_m, \gamma_m(s_m^{(t_m)}))\}$$

is that if the dialogue protocol W is in state q with current locution $\rho_i(v^{(m)}) \in LS_{\mathcal{L}}$, with the locution $\tau_j(y^{(k)})$ at the top of the stack, $\delta(q, \rho_i(\phi^{(m)}), \tau_j(y^{(k)})) =$

$\{(p_1,\gamma_1(s_1^{(t_1)})),...,(p_m,\gamma_m(s_m^{(t_m)}))\}$, and $prec(\rho_i(v^{(m)}))$ is satisfied, then W can for any $1 \leq x \leq m$ replace $\tau_j(y^{(k)})$ with locution $\gamma_x(s_x^{(t_x)})$ with values $s_x^{(t_x)}$, take the first locution in β and enter state p_x.

To formally describe the configuration of a dialogue protocol W at a given instant we define what we call an *instantaneous description*. An instantaneous description records the state and content of the stack at a given instant, along with the sequence of locutions that is being processed:

Definition 2 *An instantaneous description for a dialogue protocol is a tuple* (q,α,γ) *where* $q \in Q$ *is the current state,* $\alpha \in (LS_{\mathcal{L}})^*$ *is the remaining sequence of locutions,* $\gamma \in \Gamma^*$ *is the current stack.*

Definition 3 *The relation* $\vdash$ *satisfies* $(q,\rho_{id}(v^{(m)})\alpha,SK) \vdash (p,\alpha,newSK)$ *iff* $(p,newSK) \in \delta(q,\rho_{id}(v^{(m)}),SK)$.

We use $\vdash^*$ to denote the reflexive and transitive closure of relation $\vdash$, and we use $\vdash^+$ to denote the transitive closure of $\vdash$.

Definition 4 *The language of dialogues generated by a system* $W \in StackFST$ *specified as* $W = \langle K_{id_1},...,K_{id_n},\Sigma,Q,LS_{\mathcal{L}},\Gamma,\delta,q_0,SK_0,F\rangle$ *is defined as:*

$$L_{Dg}(W) = \{\alpha \in (LS_{\mathcal{L}})^* \mid (q_0,\alpha,SK_0) \Rightarrow_{\Sigma}^* (q_f,\lambda,SK) \wedge q_f \in F\}.$$

Definition 5 *The share knowledge generated by a system* $W \in StackFST$ *specified as* $W = \langle K_{id_1},...,K_{id_n},\Sigma,Q,LS_{\mathcal{L}},\Gamma,\delta,q_0,SK_0,F\rangle$ *is defined as:*

$$L_{SK}(W) = \left\{ \begin{array}{l} ShareK \in \Gamma^* \mid (q_0,\alpha,SK_0) \Rightarrow_{\Sigma}^* (k,\beta,ShareK) \Rightarrow_{\Sigma}^* (q_f,\lambda,SK)\wedge \\ q_f \in F \end{array} \right\}.$$

2.1 An example of dialogue protocol in framework *StackFST*

Bellow we provide an example of dialogue protocol specified in framework *StackFST* and corresponding to a goal-oriented argumentation-based two-party information-seeking protocol, in the sense of (Walton 1995):

Example 1 *Let us consider an information-seeking protocol* $\mathcal{ISP}$ *where the Information-Seeker, IS, does not know the truth of a proposition p and asks an Information-Provider, IP, about p. If the agent IP knows if p is true or false, it will inform the IS and provide, upon request, the reasons that justify the value of truth of p. The agent IS can accept or challenge the provided reasons. It can also happen that the agent IP does not know the value of truth of p. In this case then* *the dialogue finishes with the agent IS still not knowing the truth value of p. Agent IS is provided with an argumentation system* $(\Sigma_{IS}, \models_{IS})$ *where* Σ_{IS} *is a finite set of axioms in* $\mathcal{PL}$*, Propositional Logic that includes the symbol* $\mathcal{U}$*to denote uncertainty. The infix binary predicate* $\models_{IS}$ *corresponds to the inference in that logic, where the first parameter is the set of axioms and the second parameter is the formula whose validity has to be deduced from the first parameter. For instance if* $\Sigma_{IS} = \{p, p \to q\}$ *then from the agent knowledge base* Σ_{IS} *it can be deduced with* $\models_{IS}$ *that the formula q is satisfied, what we denote as* $\Sigma_{IS} \models_{IS} q$*. Agent IP is provided with the argumentation system* $(\Sigma_{IP}, \models_{IP})$ *interpreted in the same way as for agent IS.*

Formally $\mathcal{ISP} = \langle K_{IS}, K_{IP}, \mathcal{PL}, Q, LS_{\mathcal{PL}}, LS_{\mathcal{PL}}, \delta, q_0, SK_0, F\rangle$ *where:*

- K_{IS}, K_{IP} *are set of formulas in* $\mathcal{PL}$,
- $Q = \{k_0, k_1, k_2, k_3, k_4\}$,
- $F = \{k_2, k_3\}$,
- $LS_{\mathcal{PL}} = \left\{ \begin{array}{l} ask_i(p), challenge_i(p), accept_i(p), assert_i(p), assert_i(T) \mid \\ p \text{ is a formula in } \mathcal{PL} \wedge \\ T \text{ is a set of formulas in } \mathcal{PL} \wedge i \in \{IS, IP\} \end{array} \right\}$,
  with the following preconditions and postconditions:
  - *Agent IS asks* $question_{IS}(p)$ *at the beginning of the dialogue iff the stack S is empty and he can not deduce p from his argumentation system and he places the locution on the top of S.*
  - *Agent IS utters* $accept_{IS}(p)$ *iff* $top(S) = assert_{IP}(p)$ *and he can not deduce* $(\neg p)$ *from his argumentation system, and he pops the top of S.*
  - *Agent IS utters* $challenge_{IS}(p)$ *iff* $top(S) = assert_{IP}(p)$ *and he can not deduce p from his argumentation system, and he replace the top of S for* $challenge_{IS}(p)$.
  - *Agent IP utters* $assert_{IP}(p)$ *iff* $top(S) = question_{IS}(p)$ *and p is not an axiom in his argumentation system and he can deduce p from his argumentation system. The agent replaces the top of S for* $assert_{IP}(p)$.
  - *Agent IP utters* $assert_{IP}(\mathcal{U})$ *iff* $top(S) = question_{IS}(p)$ *or* $top(IS) = challenge_{IS}(p)$ *and p is an axiom in his argumentation system or he can no deduce p. The agent replaces the top of S for* $assert_{IP}(\mathcal{U})$.

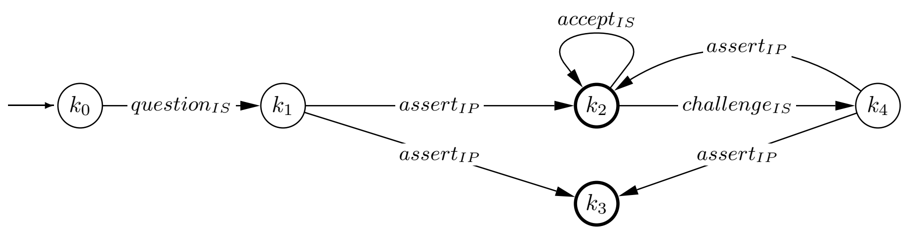

Figure 1: Transition graph for locutions interchange in $\mathcal{ISP}$

 - *Agent IP utters* $assert_{IP}(\{t_1,...,t_n\})$ *iff* $top(S) = challenge_{IS}(p)$ *and* p *is not an axiom in his argumentation system and he can deduce* p *from* $t_1 \wedge ... \wedge t_n$*. The agent replaces the top of* S *for* $assert_{IP}(t_1)...assert_{IP}(t_n)$*.*

- *From the posconditions of* $LS_{\mathcal{PL}}$ *we define function* δ*:*
  - $\delta(k_0, question_{IS}(p), SK_0) = \{(k_1, question_{IS}(p))\}$,
  - $\delta(k_2, accept_{IS}(p), assert_{IP}(p)) = \{(k_2, \lambda)\}$,
  - $\delta(k_2, challenge_{IS}(p), assert_{IP}(p)) = \{(k_4, challenge_{IS}(p))\}$,
  - $\delta(k_1, assert_{IP}(p), question_{IS}(p)) = \{(k_2, assert_{IP}(p))\}$, $p \neq \mathcal{U}$
  - $\delta(k_1, assert_{IP}(\mathcal{U}), question_{IS}(p)) = \{(k_3, assert_{IP}(\mathcal{U}))\}$,
  - $\delta(k_1, assert_{IP}(\mathcal{U}), challenge_{IS}(p)) = \{(k_3, assert_{IP}(\mathcal{U}))\}$,
  - $\delta(k_4, assert_{IP}(\{t_1,...,t_n\}), challenge_{IS}(p)) =$ $\{(k_2, assert_{IP}(t_1)...assert_{IP}(t_n))\}$.

Considering $\Sigma_{IS} = \{\neg p\}$ and $\Sigma_{IP} = (\{p, \neg r, p \rightarrow r, \neg r \rightarrow p\}$, the dialogue instance

$$question_{IS}(r)assert_{IP}(r)challenge_{IS}(r)assert_{IP}(\{p \rightarrow r, p\})$$
$$accept_{IS}(p \rightarrow r)challenge_{IS}(p)assert_{IP}(\{\neg r, \neg r \rightarrow p\})$$
$$accept_{IS}(\neg r)accept_{IS}(\neg r \rightarrow p)$$

corresponds to an information-seeking dialogue specified by dialogue protocol $\mathcal{ISP}$, where S is the stack of uttered locutions.

Formally:
$(k_0, question_{IS}(r)\alpha_1, SK_0) \Rightarrow$

$(k_1, assert_{IP}(r)\alpha_2, question_{IS}(r)) \Rightarrow$
$(k_2, challenge_{IS}(r)\alpha_3, assert_{IP}(r)) \Rightarrow$
$(k_4, assert_{IP}(\{p \rightarrow r, p\})\alpha_4, challenge_{IS}(r)) \Rightarrow$
$(k_2, accept_{IS}(p \rightarrow r)\alpha_5, assert_{IP}(p \rightarrow r), assert_{IP}(p)) \Rightarrow$
$(k_2, challenge_{IS}(p)\alpha_6, assert_{IP}(p)) \Rightarrow$
$(k_4, assert_{IP}(\{\neg r, \neg r \rightarrow p\})\alpha_7, challenge_{IS}(p)) \Rightarrow$
$(k_2, accept_{IS}(\neg r)\alpha_8, assert_{IP}(\neg r) assert_{IP}(\neg r \rightarrow p)) \Rightarrow$
$(k_2, accept_{IS}(\neg r \rightarrow p)\alpha_9, assert_{IP}(\neg r \rightarrow p)) \Rightarrow$
(k_2, λ, λ), with $k_2 \in F$.

Informally:

1 $S = SK_0$ and agent *IS* utters $question_{IS}(r)$.

2 $S = [question_{IS}(r)]$ and agent *IP* answers $assert_{IP}(r)$, because he can support r with arguments $\{p, p \rightarrow r\}$.

3 $S = [assert_{IP}(r)]$ and agent *IS* utters $challenge_{IS}(r)$ because he can not prove r from $\{\neg p\}$.

4 $S = [challenge_{IS}(r)]$ and agent *IP* answers $assert_{IP}(\{p \rightarrow r, p\})$. .

5 $S = [assert_{IP}(p \rightarrow r), assert_{IP}(p)]$ and agent *IS* replies with $accept_{IS}(p \rightarrow r)$ because he can not deduce $\neg(p \rightarrow r)$ from $\{\neg p\}$.

6 $S = [assert_{IP}(p)]$ and agent *IS* utters $challenge_{IS}(p)$ because he can deduce $\neg p$.

7 $S = [challenge_{IS}(p)]$ and agent *IP* says $assert_{IP}(\{\neg r, \neg r \rightarrow p\})$.

8 $S = [assert_{IP}(\neg r), assert_{IP}(\neg r \rightarrow p)]$ and agent *IS* utters $accept_{IS}(\neg r)$ because he can not deduce r from $\{\neg p\}$.

9 $S = [assert_{IP}(\neg r \rightarrow p)]$ and agent *IS* utters $accept_{IS}(\neg r \rightarrow p)$ because he can not deduce $\neg(\neg r \rightarrow p)$ from $\{\neg p\}$.

10 $S = []$ and the dialogue finishes successfully and agent *IS* accepts the propositions from *IP* that justify r.

In (Parsons 2003) is explained the role of the concept of social semantics in the argumentation-based dialogue protocols: an agent X can construct their arguments from their private knowledge bases Σ_X but also form the set of commitments $CS(Y)$ from agent Y. The principle behind social semantics is that when agents utter illocutions they state publicly their knowledge, which is saved in a commitment store. The truth of an agent's speech acts can not be verifiable (Wooldridge 2000), but at least an agent's consistency can be assessed inspecting the content of the commitment store.

If in the previous example we consider now the following:

- Agents IS and IP are respectively provided with argumentation systems $(\Sigma_{IS} \cup CS(IP), \models_{IS})$ and $(\Sigma_{IP} \cup CS(IS), \models_{IP})$ with $CS(IS)$ and $CS(IP)$ the commitment stores of agent IS and IP respectively.

- Instead of a memory stack S of locutions they use an string SK as memory which is initially empty. The agents use part of SK as an stack S' and the rest as commitment stores $CS(A)$ and $CS(B)$. Initially S' is an empty stack, $CS(A)$ and $CS(B)$ are empty sets.

- When an agent asserts a formula p, accepts a formula p or asserts a set of formulas S, those formulas become part of his commitment store.

- We use the same set of locutions $LS_{\mathcal{PL}}$ but we add to the postconditions the effect that the action of uttering locutions has over the commitment stores:

  – Agent IS asks $question_{IS}(p)$ at the beginning of the dialogue iff the stack S' is empty and he can not deduce p from his argumentation system and he places the locution on the top of S'.

  – Agent IS utters $accept_{IS}(p)$ iff $top(S') = assert_{IP}(p)$ and he can not deduce $(\neg p)$ from his argumentation system, and he pops the top of S' and adds p to $CS(IS)$.

  – Agent IS utters $challenge_{IS}(p)$ iff $top(S') = assert_{IP}(p)$ and he can not deduce p from his argumentation system, and he replace the top of S' for $challenge_{IS}(p)$.

  – Agent IP utters $assert_{IP}(p)$ iff $top(S') = question_{IS}(p)$ and p is not an axiom in his argumentation system and he can deduce p from his argumentation system. The agent replaces the top of S' for $assert_{IP}(p)$ and adds p to $CS(IP)$.

  – Agent IP utters $assert_{IP}(\mathcal{U})$ iff $top(S') = question_{IS}(p)$ or $top(S') = challenge_{IS}(p)$ and p is an axiom in his argumentation system or he can no deduce p. The agent replaces the top of S' for $assert_{IP}(\mathcal{U})$.

– Agent IP utters $assert_{IP}(\{t_1,...,t_n\})$ iff $top(S') = challenge_{IS}(p)$ and p is not an axiom in his argumentation system and p can be deduced from $t_1 \wedge ... \wedge t_n$. The agent replaces the top of S' for $assert_{IP}(t_1)...$ $assert_{IP}(t_n)$ and adds $t_1,...,t_n$ to $CS(IP)$.

Under the new conditions considered above and considering the same knowledge bases $\Sigma_{IS} = \{\neg p\}$ and $\Sigma_{IP} = (\{p, \neg r, p \to r, \neg r \to p\}$ as in the example presented in section 2.1 we can obtain the following dialogue instance:

$question_{IS}(r)assert_{IP}(r)challenge_{IS}(r)assert_{IP}(\{p \to r, p\})$
$accept_{IS}(p \to r)challenge_{IS}(p)assert_{IP}(\{\neg r, \neg r \to p\})$
$challenge_{IS}(\neg r)accert_{IP}(\mathcal{U})$.

Bellow we describe how we obtain the dialogue instance from above:

1 $S' = SK_0$ and $CS(IS) = CS(IP) = \emptyset$. Agent IS utters $question_{IS}(r)$.

2 $S' = [question_{IS}(r)]$ and $CS(IS) = CS(IP) = \emptyset$. Agent IP answers $assert_{IP}(r)$, because he can prove r from $\{p, p \to r\}$

3 $S' = [assert_{IP}(r)], CS(IS) = \emptyset$ and $CS(IP) = \{r\}$. Agent IS utters $challenge_{IS}(r)$, because he can not prove r from $\{\neg p\} \cup \emptyset$.

4 $S' = [challenge_{IS}(r)], CS(IS) = \emptyset$ and $CS(IP) = \{r\}$. Agent IP answers $assert_{IP}(\{p \to r, p\})$.

5 $S' = [assert_{IP}(p \to r), assert_{IP}(p)]$, $CS(IS) = \emptyset$ and $CS(IP) = \{r, p \to r, p\}$. Agent IS replies with $accept_{IS}(p \to r)$ because he can not deduce $\neg(p \to r)$ from $\{\neg p\} \cup \{r, p \to r, p\}$.

6 $S' = [assert_{IP}(p)]$, $CS(IS) = \{p \to r\}$ and $CS(IP) = \{r, p \to r, p\}$. Agent IS utters $challenge_{IS}(p)$ because he can deduce $\neg p$.

7 $S' = [challenge_{IS}(p)], CS(IS) = \{p \to r\}$ and $CS(IS) = \{r, p \to r, p\}$. Agent IP replies with $assert_{IP}(\{\neg r, \neg r \to p\})$.

8 $S' = [assert_{IP}(\neg r), assert_{IP}(\neg r \to p)], CS(IS) = \{p \to r\}$ and $CS(IP) = \{r, p \to r, p, \neg r, \neg r \to p\}$. Agent IS utters $challenge_{IS}(\neg r)$ because he can deduce r from $\{\neg p\} \cup \{r, p \to r, p, \neg r, \neg r \to p\}$.

9 $S' = [challenge_{IS}(\neg r), assert_{IP}(\neg r \to p)]$, $CS(IS) = \{p \to r\}$ and $CS(IP) = \{r, p \to r, p, \neg r, \neg r \to p\}$. Agent IP replies $assert_{IP}(\mathcal{U})$ because $\neg r$ is an axiom for him.

10 The dialogue finishes unsuccessfully, agent *IP* can not prove to agent *IS* that r is true, because due to the consideration of a commitment store $CS(IP)$ agent A can detect an inconsistency in the knowledge base of agent *IP*: agent *IP* first *asserts* r and then he *asserts* $\neg r$ during the dialogue. Considering only an stack this inconsistency in the knowledge base of agent *IP* could not be detected by agent *IS*, as the dialogue instance from previous example shows.

The analyzes presented in this section justifies the extension of framework $StackFST$ into a more expressive one that we call *Conversational Finite State Transition* ($ConvFST$) systems, where the stack is replaced but an string that can be freely accessed and modified by all the agents in the dialogue.

3. Framework $ConvFST$

Definition 6 *A dialogue protocol* $W \in ConvFST$ *of degree* $n \geq 1$ *is a tuple:*

$$W = \langle K_{id_1}, ..., K_{id_n}, \Sigma, Q, LS_{\mathcal{L}}, \Gamma, \delta, q_0, SK_0, F \rangle$$

where:

- Σ, K_{id_i} *for all* $1 \leq i \leq n$, Q, q_0 *and* F *are interpreted as in framework* $StackFST$.
- $LS_{\mathcal{L}}$ *is a finite set of locutions.*
  $$LS_{\mathcal{L}} = \left\{ \begin{array}{l} \rho_{id_i}(\phi^{(m)}) \mid m \geq 0 \land 1 \leq i \leq n \land prec(\rho_{id_i}(\phi^{(m)})) \textit{ is a wff in logic } \mathcal{L} \\ \textit{with free variables ranging over } \{\phi^1, ..., \phi^m, K_{id_i}, SK\} \end{array} \right\}$$
  The locution $\rho_{id_i}(\phi^{(m)}) \in LS_{\mathcal{L}}$ *with constants* ρ, id_i *and terms* $\phi^{(1)}, ..., \phi^{(m)}$ *has associated a well formulated formula* $prec(\rho_{id_i}(\phi^{(m)}))$ *in logic* $\mathcal{L}$ *called precondition.*
- Γ *is a finite set of symbols;*
- $SK_0 \in \Gamma^*$ *is the string corresponding to the initial share knowledge;*
- δ *is a finite transition relation* $(Q \times LS_{\mathcal{L}} \times \Gamma^*) \to 2^{Q \times \Gamma^*}$.

Considering that a sequence of locutions $\rho_{id}(v^{(m)})\beta \in (LS_{\mathcal{L}})^+$ is being processed, for any $q \in Q$, $\rho_{id}(v^{(m)}) \in LS_{\mathcal{L}}$, with parameter values $v^{(m)}$, $\beta \in (LS_{\mathcal{L}})^*$, $SK \in \Gamma^*$, the interpretation of

$$\delta(q, \rho_{id}(v^{(m)}), SK) = \{(p_1, newSK_1), ..., (p_m, newSK_m)\}$$

is that if the dialogue protocol W is in state q with current locution $\rho_{id}(v^{(m)}) \in LS_{\mathcal{L}}$, with share knowledge SK, $\delta(q, \rho_{id}(\phi^{(m)}), SK) = \{(p_1, newSK_1), ..., (p_m, newSK_m)\}$, and the formula $prec(\rho_{id}(v^{(m)}))$ is satisfied, then W can for any $1 \leq j \leq m$ replace SK with $newSK_j$, take the first locution in β and enter state p_j.

To formally describe the configuration of a dialogue protocol W at a given instant we define what we call an *instantaneous description*. An instantaneous description records the state and content of the stack at a given instant, along with the sequence of locutions that is being processed:

Definition 7 *An instantaneous description for a dialogue protocol is a tuple (q, α, SK) where $q \in Q$ is the current state, $\alpha \in (LS_{\mathcal{L}})^*$ is the remaining sequence of locutions, $SK \in \Gamma^*$ is the current share knowledge.*

The relation $\vdash$, the language of dialogues and the share knowledge generated by a system $W \in ConvFST$ is defined as in framework $StackFST$.

3.1 An example of dialogue protocol in framework *ConvFST*

We consider a dialogue protocol from framework $ConvFST$ corresponding to an argumentation-based two-party information seeking protocol, in the sense of (Walton 1995), which we call $\mathcal{ISP}_2$.

Definition 8 *We define $\mathcal{ISP}_2$ as the tuple:*

$$\mathcal{ISP}_2 = \langle K_{IS}, K_{IP}, \Sigma, Q, LS_{\mathcal{FOL}}, \Gamma, \delta, q_0, SK_0, F\rangle$$

where:

- $\Sigma = 2^{\mathcal{PL}}$, *with $\mathcal{PL}$ denoting propositional logic,*
- $K_{IS} \in 2^{\mathcal{PL}}$ *and* $K_{IP} \in 2^{\mathcal{PL}}$ *are consistent sets of axioms in propositional logic, so it is not possible that* $(\exists \alpha \in K_i : K_i \models_i \alpha \wedge K_i \models \neg\alpha)$, $i \in \{IS, IP\}$. *Predicate* $\models: 2^{\mathcal{PL}} \times \mathcal{PL} \to Boolean$ *is interpreted as logical deduction in propositional logic. Therefore* $K_i \models \varphi$ *indicates that formula* $\varphi \in \mathcal{PL}$ *can be deduced from set of axioms* $K_i \in \mathcal{PL}$ *using logical deduction in propositional logic.*
- $Q = \{q_0, q_1, q_2, q_3, q_4, q_5\}$,
- $SK_0 = \lambda$,

- $\Gamma = \{\phi, \phi?, \phi! \mid \phi \in \mathcal{PL}\}$,
- *We denote* $\mathcal{FOL}$ *the first order logic, then*
$LS_{\mathcal{FOL}} = \left\{ \begin{array}{l} ask_i(\varphi), claim_i(\varphi), retract_i(), why_i(\varphi), argue_i(\varphi), concede_i(\varphi), \\ unknown_i(\varphi) \mid i \in \{IS, IP\} \wedge \varphi \in \mathcal{PL} \end{array} \right\}$.

  Then the semantic of the locutions in $LS_{\mathcal{FOL}}$ *is the following, considering* $i \in \{IS, IP\}$ *and SK the string of share knowledge:*

$$\begin{array}{rcl} prec(ask_i(\varphi)) & = & SK = \lambda \wedge \neg(K_i \models \varphi) \wedge \neg(K_i \models \neg\varphi), \\ prec(claim_i(\varphi)) & = & SK = \psi?\alpha \wedge [\exists\varphi : (K_i \models \varphi) \wedge (\varphi = \psi \vee \varphi = \neg\psi)], \\ prec(unknown_i(\varphi)) & = & SK = \varphi?\alpha \wedge \neg(K_i \models \varphi) \wedge \neg(K_i \models \neg\varphi), \\ prec(retract_i()) & = & SK = \varphi?\alpha \wedge \neg[\exists\alpha : \neg(\alpha \leftrightarrow \varphi) \wedge (K_i \models (\alpha \rightarrow \varphi) \wedge \alpha)], \\ prec(why_i(\alpha_2)) & = & SK = \psi\alpha \wedge \\ & & \left[\begin{array}{l} \exists\alpha_1, \alpha_2 : (\psi \leftrightarrow \alpha_1 \wedge \alpha_2) \\ \wedge(K_i \models \alpha_1) \wedge \neg(K_i \models \alpha_2) \wedge \\ \neg \left[\begin{array}{l} \exists\alpha_3, \alpha_4 : (\alpha_2 \leftrightarrow \alpha_3 \wedge \alpha_4) \wedge (K_i \models \alpha_4) \wedge \\ (\alpha_4 \leftrightarrow false) \end{array} \right] \end{array} \right], \\ prec(argue_i(\varphi)) & = & SK = \psi?\alpha \wedge [\exists\varphi : (K_i \models (\varphi \rightarrow \psi) \wedge \varphi)], \textit{ and} \\ prec(concede_i(\varphi)) & = & SK = \varphi\alpha \wedge (K_i \models \varphi) \end{array}$$

- $\delta(q_0, ask_{IS}(\varphi), SK_0) = \{(q_1, \varphi?)\}$,
$\delta(q_1, claim_{IP}(\varphi), \psi?\,T) = \{(q_2, \varphi\,\psi?\,T)\}$,
$\delta(q_1, unknown_{IP}(), \psi?\,T) = \{(q_5, \psi?\,T)\}$,
$\delta(q_2, why_{IS}(\varphi), \psi\,T) = \{(q_4, \varphi?\,\psi\,T)\}$,
$\delta(q_4, argue_{IP}(\varphi), \psi?\,T) = \{(q_2, \varphi\,\psi?\,T)\}$,
$\delta(q_2, concede_{IS}(\varphi), \varphi\,T) = \{(q_3, \varphi!\,\varphi\,T)\}$,
$\delta(q_4, retract_{IP}(), \psi?\,T) = \{(q_5, \psi?\,T)\}$,
- $F = \{q_3, q_5\}$.

The interchange of locutions in $\mathcal{ISP}_2$ can be represented by the graph in figure 2.

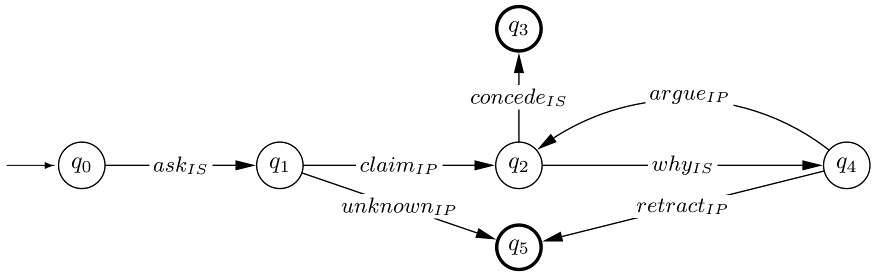

Figure 2: Transition graph for locutions interchange in $IS\mathcal{P}_2$

4. Conclusions

This work provides evidence of the importance of addressing topics of study, in this case *Dialogue Theory*, that provide an appropriate field of interdisciplinary research where theories and results from different areas of study, like *Formal Language Theory*, *Linguistic* and *Artificial Intelligence* can be reinterpreted and connected.

Here we contribute to *Dialogue Theory* from the perspective of *Formal Language Theory* with the introduction of two frameworks for the specification of dialogue protocols: *StackFST* and *ConvFST*. Because we specify frameworks *StackFST* and *ConvFST* as finite state transition systems, they have the following properties:

- They are easy and understandable methods for describing protocols.
- They are provided with a formal semantic.
- They can be subject of a verification strategy called model checking.
- They allow to specify dialogues where speakers take turns.

In particular the framework *StackFST* proves to show the following features which are, according to *Linguistic* studies, characteristic of human-like conversations:

- modification of the knowledge base (the stack) share by the agents in the dialogue,
- backtracking (the capacity to reply to locutions uttered at any earlier step of the dialogue and not only the previous one),

- flexibility in: turn-taking rules, type of replies (single, multiple), the selection of locutions used, the type of agent private knowledge bases allowed, the complexity of agent reasoning strategies, the dialogue initiatives (system initiative, user initiative and mixed initiative).

But between others, the framework *StackFST* lacks of these features from human-like talks:

- unrestricted dynamic incorporation of conversants during the dialogue,
- capacity of the agents to learn or modify their own private knowledge bases,
- simulate social semantics (Parsons 2003),
- freedom in the selection of the type of information saved in the share knowledge base and in the way to manipulate the share knowledge base.

The extension of framework *StackFST* into the framework *ConvFST* results in the possibility of incorporation to dialogue protocols the collection of all the agent observable behaviors, what in *Linguistic* is known as dialogue context and was later considered in *Artificial Intelligence* with names like agent observable behaviors (Viroli 2002), histories of communication (Van Eijk 2003) or traces of communication (Widom 1987).

The frameworks *StackFST* and *ConvFST* that we introduce define generic, abstract and well defined formalisms in which protocols can be specified, compared and evaluated in the same notation in a precise way. For the protocol $IS\mathcal{P}_2$ from section 3.1 that is expressed in framework *ConvFST*, we prove in (Grando 2007) that it is possible to formally specify the set of dialogue instances(language) that it generates and from this language specification prove some formal properties. In (Grando 2007) we also provide an example of two information-seeking dialogue protocols expressed in different notations, that once they are specified in the same framework *ConvFST* can be compared. An open problem in the area of *Dialogue Theory* is the absence of a formal strategy to compare arbitrary dialogue protocols. In a future we are interested to study this issue to address the type of problems pointed out in the area of agent communication languages in *Artificial Intelligence* (Parsons 2003): How might one choose between two protocols?, when is one protocol preferable to another?, when do two protocols differ?, can we tell if a protocol is new (in the sense of providing a different functionality from an existing protocol rather than just having equivalent locution with different names)?, is a protocol new (in the sense of providing a different functionality from an existing protocol rather than just having equivalent locutions with different names)?, ...

We are currently analyzing from the perspective of *Formal Language Theory* some formal properties satisfied by the frameworks $StackFST$ and $ConvFST$. For instance expressive power, computational complexity, the influence that the number of speakers or other restrictions we impose have over its expressive power, decidability of the problem of determining if a dialogue instance satisfies a dialogue protocol (membership problem), etcetera. The research we are undertaking will help designers of protocols in the area of *Artificial Intelligence*, for the task of selecting the most suitable formalism for the definition of dialogues that fulfil better their requirements of expressible power and computational complexity.

5. References

Dahlbäck N. and Jönsson A. (1992). "An empirically based computationally tractable dialogue model". In *Proceedings of the 14th Annual Conference of the Cognitive Science Society (COGSCI-92)*. USA, pp. 785–790.

Esteva M., Rodriguez J. A., Sierra C., Garcia P. and Arcos J. L. (2001). "On the Formal Specification of Electronic Institutions". In *Lecture Notes in Artificial Intelligence, Agent-mediated Electronic Commerce (The European AgentLink Perspective)*, pp. 126-147.

Genereth M. R. and Ketchpel S. P. (1994). "Software Agents". In *Communications of the ACM 37 (7)*, pp. 48-53.

Grando M. A. and Parsons S. (2007). "Dialogues as extended finite transition systems". In *Internacional workshop on non-classical formal languages in linguistics (ForLing) 2007*.Budapest (Hungria). To appear.

Holvoet T. and Kielmann T. (1998). "Behavior specification of parallel active objects". In *Journal on Parallel Computing* 24(7), pp. 1107-1135.

Koningand J. L., Huget M. P., Wei J. and Wang X. (2001). "Extended modeling languages for interaction protocol design". In *Proceedings of Agent-Oriented Software Engineering (AOSE-01)*.

Moldt D. and Wienberg F. (1997). "Multi-agent-systems based on coloured petri nets". In *Proceedings of the 18th International Conference on Application and Theory of Petri Nets (ICATPN '97)* 1248. Lecture Notes in Computer Science, pp. 82-101.

Parsons S., Wooldridge M. and Amgoud L. (2003). "Properties and Complexity of Some Formal Inter-agent Dialogues". In *Journal of Logic and Computation 13(3)*, pp. 347-376.

Parunak D. and Bauer B. (2000)."Extending UML for agents". In *Proceedings of Agent-Oriented Information Systems Workshop (AOIS-00).*

Purvis M. and Craneffeld S. (1996). "Agent modelling with Petri Nets". In *Proceedings of the CESA '96 (Computational Engineering in Systems Applications) Symposium on Discrete Events and Manufacturing Systems*, pp. 602-607.

Robertson D. (2004). "A Lightweight Coordination Calculus for Agent Systems". In *Declarative Agent Languages and Technologies II*. Springer Berling/ Heidelberg, Lecture Notes in Computer Science 3476/2005, pp. 183-197.

Shneiderman B. (1980). "Natural vs. precise concise languages for human operation of computers: Research issues and experimental approaches". In *Proceedings of the 18th Annual Meeting of the Association for Computational Linguistics*. USA: Association for Computational Linguistics, pp. 139–141.

Van Eijk R. M., De Boer F. S., Van Der Hoek W. and Meyer J. Ch. (2003). "A Verification Framework for Agent Communication". In *Autonomous Agents and Multi-Agent Systems 6(2)*. USA: Kluwer Academic Publishers, pp. 185-219.

Vasconcelos W. (2004). "Norm Verification and Analysis of Electronic Institutions". In *2004 Workshop on Declarative Agent Languages and Technologies (DALT-04)*. USA, pp. 141-155.

Viroli M. and Omicini A. (2002). "Specifying agent observable behaviour". In *1st International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2002) (2)*. ACM: pp. 712-720.

Walton C. (2004). "Multi-Agent Dialogue Protocols". In *Proceedings of the Eighth International Symposium on Artificial Intelligence and Mathematics*. USA.

Walton D. and Krabbe E. C. W. (1995). *Commitment in Dialogue: Basic Concepts of Interpersonal Reasoning.* USA: SUNY press.

Wei J., Cheung S. C. and Wang X. (2001). "Towards a methodology for formal design and analysis of agent interaction protocols: An investigation in electronic commerce". In *Proceedings of the International Software Engineering Symposium*.

Widom J., Gries D. and Schneider F. B. (1987). "Completeness and incompleteness of trace-based network proof systems". In *POPL '87: Proceedings of the 14th ACM SIGACT-SIGPLAN symposium on Principles of programming languages*. USA: ACM Press, pp. 27-38.

Woods W. A. (1999). "FIPA specification: agent communication language". In *Foundation for Intelligent Physical Agents, part 2*. URL: www.fipa.org.

Wooldridge M. (2000). "Semantic Issues in the Verification of Agent Communication Languages". In Jennings Nicholas, Sycara Katia and Georgeff Michael, eds., *Autonomous Agents and Multi-Agent Systems 3(1)*. Kluwer Academic publishers, pp. 9-31.