ISSN 2007-9737

A Logical Interpretation of Silence

Alfonso Garcés Báez, Aurelio López López

Instituto Nacional de Astrofísica, Óptica y Electrónica, Mexico

{agarces, allopez}@ccc.inaoep.mx

Abstract. In human communication process, we often face situations where decisions have to be made, regardless of silence of one of the interlocutors. That is, we have to decide from incomplete information, guessing the knowledge or intentions of the silent person. This behaviour has been studied by several disciplines but barely touched in logic or artificial intelligence. After reviewing some previous studies of silence and conversational implicature of Grice, we focus on a puzzle formerly expressed and solved in Answer Set Programming, to analyze the implications of two different interpretations of silence (Defensive and Acquiescent Silence), in terms of the Says() predicate. Several conclusions are derived from the different possibilities that opened for analysis. In addition, a general strategy for analysis of problems involving testimonies and silence is proposed.

Keywords. Silence, interpretation, dialogue, speech acts, intention, says predicate, says graph, puzzle, answer set programming.

1 Introduction

In human communication process, we often face situations where decisions have to be made, regardless of silence of one of the interlocutors, as often occurs in a dialogue. That is, we have to decide from incomplete information, guessing the knowledge or intentions of the silent person. This behavior has been studied by several disciplines but barely touched in logic or artificial intelligence. We have not found an approach to formalize the use of intentional silence in terms of logic, the closest attempt was that of [9] with an ”informal” logic.

According to Kurzon [10], there are two types of silence, intentional and unintentional.

Intentional silence is a deliberate action not to cooperate with the other party and unintentional silence is psychological in nature.

The interpretation of silence must be contextual. For example, in a normal conversation, silence is interpreted to the detriment of the person who is silent. The immediate reaction is that she hides something. For Kurzon [10], the silence is defined by language, and points to three types of silence:

— Psychological silence. The help of a decoder is necessary.

— Interactive silence. It occurs as an intentional pause in the conversation, allowing the other person to draw inferences related to the meaning of the conversation.

— Socio-cultural silence. When silence is interpreted based on specific cultural codes.

The intentional silence is also a sign of group loyalty. To interpret intentional silence, first we have to discard the modal ”can” that expresses unintentional silence, as in ”I can not speak” [10]. Then, intentional silence can be interpreted with four manners:

1. I may not tell you.
2. I must not tell you.
3. I shall not tell you.
4. I will not tell you.

Where manners 1 and 2 are intentional external silences ”by order”. And manners 3 and 4 are intentional internal silences ”by will”.

Bohnet and Frey [1] state that the variants for the interpretation of silence can be: anonymous and not anonymous, where the latter can be with identification and face-to-face. They also studied silence in the communication process, specifically in the context of prisoner's dilemma.

From the point of view of semiotics, silence is a sign. In a communication scheme that includes the interpretation of silence in its basic form, the speaker has to interpret the silence that the listener sends with a certain intention. The sender becomes the receiver of silence [3].
The categories proposed by Grice [8] to have a good communication are:

1. Quantity. Make your informative contribution as required for current exchange purposes. Do not make more informative contributions than required.

2. Quality. Do not say what you think is false. Do not say that for which you lack adequate evidence.

3. Relation. Be relevant

4. Manner. Avoid obscure expressions. Avoid ambiguity. Be brief, avoid unnecessary prolixity. Be ordered.

Category 4 is related to the way it is said and includes the supermaxim 'Be perspicuous' and various maxims more. A participant in a talk exchange may fail to fulfill with ”Avoid obscure expressions”; he may say, indicate, or allow it to become plain that he is unwilling to cooperate in the way the maxim requires. He may say, for example, 'I cannot say more; my lips are sealed'. Nevertheless, this type of silence is intentional, details the context in which is presented and allows decisions to be made.

This paper offers a first logical approach to the study of intentional silence in a particular context, focusing on a puzzle formally expressed and previously solved, to analyze the implications of two different interpretations of silence, expressed in terms of the predicate Says.

The first intentional silence is Defensive Silence that is proactive, involving awareness and consideration of alternatives, followed by a conscious decision to withhold ideas, information, and opinions as the best personal strategy at the moment [2]. The second intentional silence, on the other hand, suggests disengaged behaviour that is more passive than active. We call it Acquiescent Silence, in this case, who remains silent agrees implicitly with what others say. Several conclusions are derived from both readings. After analyzing the case study, we propose a general strategy for the analysis of problems involving testimonies and silence.

This paper is organized as follows: Section 2 shows preliminaries concepts. Section 3 describes a case study for silence, includes the two interpretations and their consequences. Section 4 includes a formulation of a general strategy for analyzing problems involving testimonies. We conclude in Section 5, discussing in addition work in progress.

2 Preliminaries

A *clause* is a formula of the form $H \leftarrow B$ where H and B are arbitrary formulas in principle, called head and body of the clause respectively. There are several types of clauses. If $H = \{\}$ the clause is called a constraint and we can write that clause as $\leftarrow B$. Analogously, if $B = \{\}$ then the clause is called a fact and can be written as $H \leftarrow$. An augmented clause is a clause where H and B are some conjunction, disjunction or denial [6]. Hp and Bp contain positive atoms. Hn and Bn contain negative atoms.

A *logical program* is then a finite set of clauses. If all the clauses in a program are of a certain type, we say that the program is also of that type. For example, a set of augmented clauses specifies an augmented program, a set of free clauses is a free program and so is in the case of the disjunctive and definite programs [6]. Figure 1 shows the classification of types of clauses that can lead to the corresponding logical programs.

A set consisting of literals X, *satisfies* a basic formula F (symbolically, $X \models F$) recursively, as follows [11]:

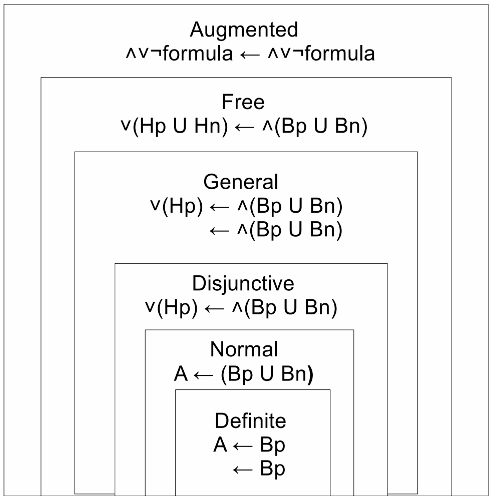

Fig. 1. Types of clauses

— For F elemental, $X \models F$ if $F \in X$ or $F = \top$,

— $X \models (F, G)$ if $X \models F$ and $X \models G$,

— $X \models (F; G)$ if $X \models F$ or $X \models G$.

Let Π be a basic program. A consistent set of literals X is *closed* under Π if, for each rule $F \leftarrow G$ in Π, $X \models F$ when $X \models G$.

We say that X is a *answer set* for the basic program Π if X is the minimum amount of the consistent set of literals closed under Π. For example, consider the program: $q \leftarrow p \vee -p$. The closure under this program is characterized by the following condition: if $p \in X$ or $-p \in X$ then $q \in X$. It is clear that the answer set for that program is empty. If we add the rule p (that is, $p \leftarrow \top$) to this program then $\{p, q\}$ would be the answer set.

The *reduction* of a formula, rule or program Π, relative to a consistent X set of literals is defined recursively, as follows [11]:

— For F elemental, $F^X = F$,

— $(F \wedge G)^X = \left(F^X \wedge G^X\right)$,

— $(F \vee G)^X = \left(F^X \vee G^X\right)$,

— $(not\ F)^X = \{\bot$, if $X \models F^X; \top$, in other case$\}$,

— $(F \leftarrow G)^X = F^X \leftarrow G^X$,

— $\Pi^X = \left\{(F \leftarrow G)^X : F \leftarrow G \in \Pi\right\}$.

For example, let P be such that:

$$P: \quad a \leftarrow \neg\neg a, \quad \neg b \leftarrow c \vee b.$$

If we choose $X = \{a\}$ then the reduction is

$$P^X: \quad a \leftarrow \top, \quad \top \leftarrow c \vee b.$$

One can easily verify that $\{a\}$ is closed under this reduction and the empty set or $\emptyset$ is not, this is the minimum set with such property. So, it follows that $\{a\}$ is a P answer set. However, this shows that the empty set or $\emptyset$ is also a set of P, but produces a different closed *reduction*.

A consistent X set of literals is an *answer set* for a program Π if this is an answer set for the reduction Π^X.

Conversational Implicature is a potential inference that is not a logical implication and is connected with the meaning of the word ”say”.

Cooperative Principle consists of the participants making their conversational contribution as required in the scenario in which this occurs for the accepted purpose or direction of the speech exchange in which they were engaged [8]. In this sense, we employ the predicate ”Says” that has emerged previously in logics for access control [4]. Such predicate can be restricted by varied axioms, but we are using it here simply to express that “somebody” is asserting “something”, and considering that also intentional silence ”says” something when is interpreted in its context for decision making.

We define the predicate ”says” in the sense of Grice, as: *Says(X, Y)*, it expresses that the agent X says Y (predicate). A predicates set ”says” can have a $SaysGraph$ associated. A $SaysGraph = < V, Ap >$, where *V* is a set of agents and *Ap* is a set of Predicate Arcs. A $SaysGraph$ represents the relations of the subset X (Says) of a program P, defined later.

Likewise, we give the following definition:

A *Predicate Arc* is a directed arc. The origin of the arc, labeled with a predicate, corresponds to the agent asserting something and the destination or destinations correspond to the agent or agents referred to.

If the origin of the arc is a black dot, the predicate includes, as an argument, the same agent that asserts.

Next, we define informally Defensive and Acquiescent Silence, as described by Dyne, Ang, and Botero [2], and then formalize such types of silence for further application and analysis.

Defensive Silence consists of withholding relevant ideas, information, or opinions as a form of self-protection, based on fear [2].

Let *P* be a logical program or a knowledge base (KB); $A_1,\ A_2,\ldots,\ A_n$ are agents; $p_1,\ p_2,\ldots,\ p_m$ are predicates; $X_{A_1} = \{Says(A_1,p_1),...,Says(A_1,p_l)\} = \{Says(A_1,*)\};\ \ldots;X_{A_k} = \{Says(A_k,*)\}$; $X = X_{A_1}\ UX_{A_2}\ U\ldots UX_{A_n};\ X \subset P;\ (1 \leq k \leq n);\ (1 \leq l \leq m)$.

P_{A_i} is *Total Defensive Silence (TDS)* for $A_i, (1 \leq i \leq n)$; where P_{A_i}=$P - X_{A_i}$.

$P_{A_{ij}}$ is *Partial Defensive Silence (PDS)* for $A_i, (1 \leq i \leq n), (1 \leq j \leq m)$; where $P_{A_{ij}}$=$P - \{Says(A_i,p_j)\}$.

Acquiescent Silence expresses witholding relevant ideas, information, or opinions, based on resignation [2].

P'_{A_i} is *Acquiescent Silence (AS)* for A_i; where P'_{A_i}=$P_{A_i}U\{Says(A_j,*)\} \circ \lambda$, P_{A_i} is *TDS* for $A_i,\ \lambda = \{A_j/A_i\}, (1 \leq i,j \leq n)$ and $(j \neq i)$.
Here the operator $\circ$ with λ sustitution denotes the replacement of A_j for A_i on *Says* subset.

We employ the Answer-Set Programming (ASP) paradigm to explore the implications of silence in our case study, given that is closely related to intuitionist logic, i.e. both are based on the concept of *proof* rather than *truth* (previously shown in [7] for intuitionist logic). This is a logical programming branch that computes stable models for difficult problems [6], where a stable model is a belief system that holds for a rational agent. This approach:

- Goes beyond answering queries.
- Is used to solve computational problems by reducing them to finding answer sets of programs.
- In principle, any NP-complete problem can be solved in this way using ASP without disjunction.
- With disjunction, we can solve more-complex problems.

Clingo is an implementation of ASP that allows to find, if there exists, the answer set or stable model of a logical program [5]. Clingo is used to generate answer sets for the problem of a case study, and explore the implications of the silence interpretations formulated. Python is used to update the KB or logical program [14].

3 A Case Study for Silence

There is a puzzle, previously modeled and solved in [6], that includes testimonies of different people, and allows to model and explore our two interpretations of of silence. In this puzzle, a mystery related to the murder of a person is raised, where one can assume that a judge requests and records the testimony of three suspects:

Vinny has been murdered, and Andy, Ben, and Cole are suspects.

Andy says he did not do it. He says that Ben was the victim's friend but that Cole hated the victim. Ben says he was out of town the day of the murder, and besides he didn't even know the guy. Cole says he is innocent and he saw Andy and Ben with the victim just before the murder.

Figure 2 presents the testimony of the suspects through the predicate Says() as a SaysGraph. We must assume that all the people involved tell the truth except, possibly, the murderer. The story and testimony of these three people is formulated in the program Mystery.lp for Clingo (see Appendix A).

The program for the puzzle produces as a result: murderer(ben). This means that according to the testimonies and the rules of common sense knowledge provided, the murderer is Ben.

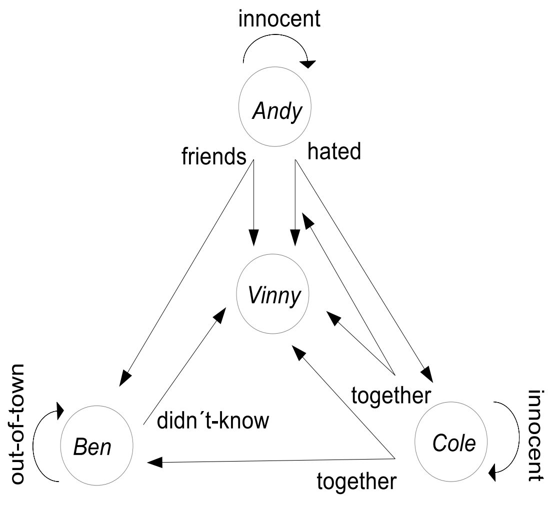

Fig. 2. SaysGraph for original puzzle

3.1 Interpreting Silence

Based on the formulation of the puzzle previously described, we proceed to explore two interpretations of intentional silence linked to such context.

The first interpretation is a Defensive Silence, i.e. an agent intentionally simply remains silent, mainly by fear. While the second corresponds to Acquiescent Silence, understood as asserting with silence what others have said, commonly by resignation. We explore in both cases, the consequences of the interpretation.

3.1.1 Defensive Silence

Defensive silence, previously defined, is intentional and proactive behavior that is intended to protect the user from external threats.

If an agent investigating a case faces this kind of silence of one or more of those involved, he can not count on their testimonies. So, we have to remove the declaration of those people, as a rule.

So expressing this kind of silence in the context of our case study (puzzle); what would happen if silence with common sense is presented as a possibility? What conclusions the interrogator or judge can reach if some of the suspects decide to intentionally remain quiet?

Applying this first rule to each person giving his testimony and executing the Python program:

```
t_def_silence(Mystery.lp,andy)
t_def_silence(Mystery.lp,ben)
t_def_silence(Mystery.lp,cole)
```

Table 1. Defensive Silence model for agent

Silent agent(s)	Presumable culprit
{}	{ben}
{andy}	{cole, ben}
{ben}	{cole, ben, andy}
{cole}	{ben, andy}
{andy, ben}	{cole, ben, andy}
{ben, cole}	{cole, andy, ben}
{andy, cole}	{cole, ben, andy}
{andy, ben, cole}	{cole, ben, andy}

The possible outcomes (guilty) when a one or more suspects decide intentionally to omit their testimonies are presented in Table 1. In this, we can notice that the culprit can be anyone depending on who decides to stay silent. For the possibilities, we can comment:

1. {} corresponds to the original scheme where nobody is silent, i.e. every testimony is taken into account. The only model for this case is Ben, as before.

2. When Andy is silent, the offender turns out to be either Ben or Cole. Each answer corresponds to a model, as shown below:

   ```
   Answer: 1
   murderer(cole).
   Answer: 2
   murderer(ben).
   SATISFIABLE
   ```

   The last line indicates that there are no more models.

3. When Ben is silent, any of the three suspects may be guilty. Intuitively we can think that Ben's silence has more decision capability since anyone involved can turn out as guilty.

4. Cole's silence can turn Andy or Ben guilty.

ISSN 2007-9737

5. With the remaining possibilities, related to more than one person, any of the three involved can be guilty.

About the problem in general, we can add that Andy's silence can be in his own benefit, in the same way as Cole.

For instance, in the case of Andy, in terms of logic programming, the rules that have to be deleted are: $\{Says(andy, *)\}$. The existing relations are depicted in Figure 3.

We can further detail the analysis by considering partial silence. We can now wonder: What part of the testimony could be convenient to silence in the case of suspects? Who would be the culprit in the event that some person decide to remain partially silent? What possibilities would each of the suspects have if, before giving their allegation, they have access to the testimony of others?

Reflection for each suspect using the programs: *p_def_silence(kb,agent,predicate)* and Clingo.

1. In the case of Andy, with the silence of his first or second statement, the culprit can be Ben, with the silence of the third one, Cole also appears as presumable guilty. Table 2 shows the possibilities of Andy.

2. Ben is the most affected with his silence, either total (Table 1) or partial since he comes out in every model sharing the suspicion with somebody else, as Table 3 shows.

3. Cole can also decide, without incriminating himself, whom to reveal as guilty. Table 4 shows the different answers obtained.

3.1.2 Acquiescent Silence

The second interpretation of silence is related with the old saying "silence is consent", expressing a passive disengaged attitude, previously defined.

In this interpretation, we operationalize it by omitting the whole person's testimony and inserting new rules related with what he is implicitly assuming with his silence. For example, in the case of Ben, we execute the program *acq_silence(Mystery.pl,ben)*, which leads to:

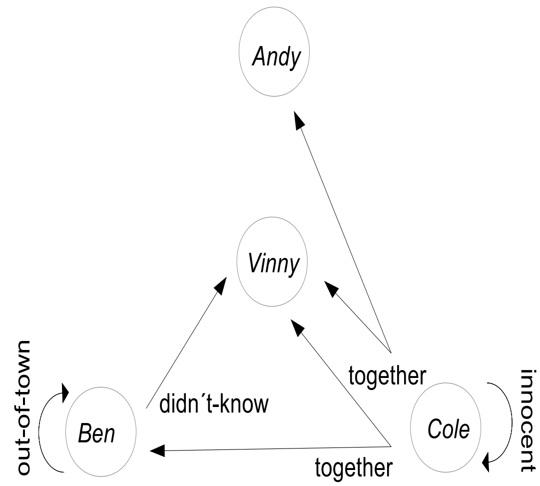

Fig. 3. SaysGraph for Andy's Defensive Silence

Table 2. Partial Defensive Silence models for Andy

Silenced testimony (predicate)	Presumable culprit
{says(andy, innocent(andy))}	{ben}
{says(andy, hated(cole,vinny))}	{ben}
{says(andy, friends(ben,vinny))}	{ben, cole}

Table 3. Partial Defensive Silence models for Ben

Silenced testimony (predicate)	Presumable culprit
{says(ben,out-of-town(ben))}	{andy, ben}
{says(ben,know(ben,vinny))}	{ben, cole}

Table 4. Partial Defensive Silence models for Cole

Silenced testimony (predicate)	Presumable culprit
{says(cole,innocent(cole))}	{ben}
{says(cole,together(andy,vinny))}	{ben}
{says(cole,together(ben,vinny))}	{andy, ben}

1. Ignore the following assertions, since he is not declaring anything: $\{Says(ben, p_i)\}, (1 \leq i \leq 2)$.

2. Add the following assertions, to model his consent on what others say: $Says(ben, q)$, where $\{Says(A, q)\} \subset P$ and $A \neq ben$.

Figure 4 presents the Acquiescent Silence for Ben, where he turns out to be guilty and is obtained by the execution of the program clingo 0 Mystery-as-ben.pl. The answer was similar to model of the original problem.

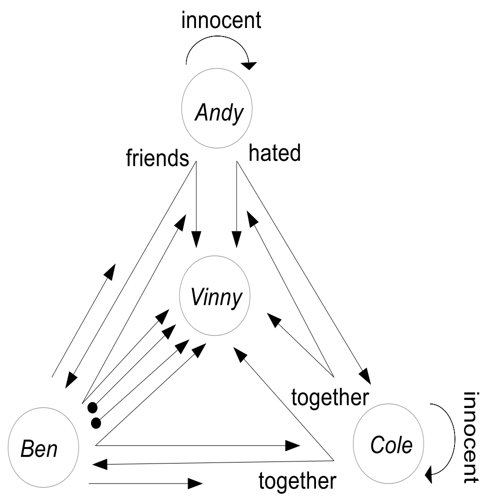

Fig. 4. SaysGraph of Ben's Acquiescent Silence

Table 5. Acquiescent Silence for agent

Silent agent(s)	Presumable culprit
{}	{ben}
{andy}	UNSATISFIABLE
{ben}	{ben}
{cole}	UNSATISFIABLE
{andy, ben}	{ben, andy}
{ben, cole}	{ben, cole}
{andy, cole}	{cole, andy}
{andy, ben, cole}	{cole, ben, andy}

Table 5 shows the solutions reached for the puzzle when one or several persons are silenced under the interpretation of Acquiescent Silence. Again, the first line corresponds to the original situation where everybody has declared, leading to Ben as the murderer.

Notice that there is no model (solution) in cases 2 and 4, where UNSATISFIABLE is obtained. These situations can be interpreted that there is no evidence to blame any of the suspects, possibly leading to a mistrial. So, under this scheme, Andy and Cole are those who could benefit from remaining silent.

Table 6. Combining the two types of silence

Defensive	Acquiescent	Declarant	Presumable culprit
andy	ben	cole	{ben}
andy	cole	ben	{cole}
ben	andy	cole	{andy}
ben	cole	andy	{cole}
cole	andy	ben	{andy}
cole	ben	andy	{cole, andy, ben}

In cases 5, 6 and 7, the person who speaks is out of suspicion. In the latter case, as expected from common sense, when everybody is silent (no one has revealed any information), anyone can be the culprit.

3.2 Combining Types of Silence

In a real life situation, we can have that people remain silent by different reasons, i.e. we can have simultaneously different types of silence. If we find that some of those declaring recur to different types of silence, what would be the consequences in the case under consideration? In particular, for the mistery, who will be guilty if the Defensive and Acquiescent silences are combined in the testimony?

We can also analyze scenarios where the two kinds of silence are displayed by participants, e.g. one is recurring to defensive silence and other to acquiescent silence. Table 6 shows the possibilities when Defensive Silence and Acquiescent Silence are combined among the three agents involved. Of course, one of them has to remain as declarant.

We can notice that under these different scenarios, the models reduce to point to only one person, except in the case of the last row, that came out equivalent as both displaying defensive silence (Table 1).

3.3 Discussion

An important fact that we want to highlight is when two programs are ”equivalent” with respect to the semantic answer set. We consider a definition for ”equivalence”, where two programs are equivalent if they have the same answer set. Intuitively, the original program, whose known solution is Ben, and several programs obtained under the two interpretations of silence, are equivalent from the point of view of the obtained result.

Table 7. Defensive Silence versus Acquiescent Silence

Silent agent(s)	Defensive Silence	Acquiescent Silence
{}	{ben}	{ben}
{andy}	{ben, cole}	UNSATISFIABLE
{ben}	{cole, andy, ben}	{ben}
{cole}	{andy, ben}	UNSATISFIABLE
{andy, ben}	{cole, ben, andy}	{ben, andy}
{ben, cole}	{cole, andy, ben}	{ben, cole}
{andy, cole}	{cole, ben, andy}	{cole, andy}
{andy, ben, cole}	{cole, ben, andy}	{cole, ben, andy}

Also, this formulation as speech acts (expressed in terms of the predicate Says) under the assumption of silence of one or more of the interlocutors is an example of non monotonicity, since allows to draw tentative conclusions, in particular:

— Under Defensive Silence, with the silence of Andy, the first found answer led to Ben as a solution. In this case, we have three rules less in the knowledge base.

— Under Acquiescent Silence, with the silence of Ben, the culprit comes out also as Ben. In this case, we require four additional rules (i.e. -2+6).

Contrasting the two types of silence (Table 7), we can notice that Defensive Silence opens possibilities (that is one of the reasons for the right to remain silent) while Acquiescent Silence restricts them. People who recur to Acquiescent Silence tend to appear as guilty, except for cases where no solution is found and that correspond to the silence of Andy and silence of Cole.

It was also noteworthy the interaction of the two types of silence for the case under consideration, leading mostly to answer sets with one element.

4 A Strategy for Analysis

We now formulate a strategy for bringing intentional silence in the analysis of problems involving testimonies. Assuming that testimonies of different people involved are already available, the strategy is formulated as follows:

1. Identify agents and predicates.

2. Formalize the statements using the predicate ”Says”.

3. Add definitions and common sense rules according to the problem at hand.

4. Identify the types of silence ocurring in the problem.

5. Generate a KB to model the problem, including agent statements, common sense knowledge, and identified types of silence. Depending on the type of silence of the agents, one or more of the following programs have to be executed, for the corresponding agent, to define the knowledge base accordingly:

```
t_def_silence(kb.pl,agent)
p_def_silence(kb.pl,agent, predicate)
acq_silence(kb.pl,agent)
```

6. Apply ASP to get the models taking into account the corresponding acts of silence on the KB:

```
clingo 0 kb.pl
```

7. Analyze the different scenarios obtained.

A key step in the strategy is 4. Here, the obvious case is when one of those involved recurs to his right to remain silent. We can then proceed to consider, one at a time, the two types of silence for such person.

However, other situations can emerge, for instance when two declarants A and B separately coincide in statements *p* and *q*, but let say A in addition declares *r*. We can then hypothesize an acquiescent silence of B, or even a partial defensive silence, since he is omiting *r*, and proceed accordingly to represent and analyze the problem.

A second puzzle is analyzed to illustrate the application of the strategy . This is formulated as follows [13]:

The Island of Knights and Knaves has two types of inhabitants: knights, who always tell the truth, and knaves, who always lie. One day, three inhabitants (A, B, and C) of the island met a foreign tourist and gave the following information about themselves:

— *A said that B and C are both knights.*

— *B said that A is a knave and C is a knight.*

What types are A, B, and C?.

1. Agents: a, b and c. Predicates: knight and knave.

2. From the agent declarations, the following assertions are obtained, expressed in terms of predicate says:

```
says(a,knight(b)).
says(a,knight(c)).
says(b,knave(a)).
says(b,knight(c)).
```

3. Definitions and common sense knowledge:

```
knave(P) :- agent(P), says(P,S),
S==False.
knight(P) :- agent(P), not knave(P).
1{knight(P);knave(P)}1 :- agent(P).
:- knave(a),knight(b),knight(c).
:- knave(b),knave(a),knight(c).
```

4. Given that the declarations of A and B differ, the silence to consider is defensive.

5. Two programs to execute to set silence, one at a time.

```
t_def_silence(knight-knave.pl,a)
t_def-Silence(knight-knave.pl,b)
```

6. And then to obtain answer sets, again execute one at a time.

```
clingo 0 knight-knave-tds-a.pl
clingo 0 knight-knave-tds-b.pl
```

7. The solution to the puzzle as formulated is: *knave(a), knave(b), knave(c)*.
   After defensive silence of A, we got:

   — *knave(b), knight(a), knave(c)*.

   — *knave(b), knight(a), knight(c)*.

   — *knave(b), knave(a), knave(c)*.

   And after defensive silence of B, we obtained:

   — *knave(a), knave(b), knave(c)*

   — *knave(a), knave(b), knight(c)*

   — *knave(a), knight(b), knave(c)*

   Under both scenarios analyzed, we got the solution of the original puzzle, i.e. the three agents are knaves. In addition, we can observe that agent C does not give information, which can be interpreted as a unintentional silence. With the intentional silence of agents A and B, agent C maintains a constant behavior.

   Considering silence in this puzzle led to conclude that (total) defensive silence benefits the agent who practices it.

5 Conclusions

Silence expresses valuable information that can be employed for decision making. In particular, when the intentional silent is interpreted according to its context, we achieve implicatures.

Understanding and modeling the implications of silence can be useful in agent interaction, either human or virtual. For instance, we foresee a useful analysis of different scenario in legal cases involving testimonies of varied people (witnesses) and different kinds of silence. The strategy sketched can serve as a basis for a system supporting judges or prosecutors for decision making.

Each model generated with a logical program in the ASP paradigm can have a formal proof in intuitionist logic, according to [12]. Thus, each of the applications of the interpretations of silence and their possible combinations, intuitively, represent a formal proof within the belief system constructed from the assertions (statements) of the rational agents involved.

As future work, we plan to extend the interpretations to incorporate prosocial silence, i.e. retaining work-related information or opinions with the goal of benefiting other people or an organization.

It remains to bring the interpretations of silence to a more general framework for agent interaction, beyond testimonies and puzzles. Also in this direction, we are exploring to consider payoffs of agents involved in the interaction, as well as to the predicates to know who or what has more gains with silence, as an instrument in making decisions.

The silence can be active, conscious, intentional, strategic, and purposeful.

6 Appendix: Mistery.lp for Clingo

```
% Predicates:
agent(andy; ben; cole; vinny).

% Andy says:
% He didn't do it.
says(andy,innocent(andy)).
% Cole hated Vinny.
says(andy,hated(cole, vinny)).
% Ben and Vinny were friends.
says(andy,friends(ben, vinny)).
% Ben says:
% He was out of town.
says(ben,outoftown(ben)).
% He didn't know Vinny.
says(ben,didnotknow(ben, vinny)).
% Cole says:
% He is innocent.
says(cole,innocent(cole)).
% He saw Andy and Ben
says(cole,together(andy, vinny)).
% with the victim.
says(cole,together(ben, vinny)).

% Everyone, except possibly for
%the murderer, is telling the truth:
holds(S) :- agent(P),says(P,S),
not murderer(P).
% Relation together is symmetric and
%transitive:
holds(together(A,B)) :- agent(A),
agent(B), holds(together(B,A)).
holds(together(A,B)) :- agent(A),
agent(B), agent(C),
holds(together(A,C)), holds(together(C,B)).
% Relation friends is symmetric:
holds(friends(A,B)) :- agent(A), agent(B),
holds(friends(B,A)).

% Murderers are not innocent:
:- agent(P),holds(innocent(P)),
holds(murderer(P)).
% A person cannot be together with someone
%who is out of town:
:- agent(A),agent(B), holds(outoftown(A)),
holds(together(A,B)).
% Friends know each other:
:- agent(A),agent(B), holds(didnotknow(A,
B)), holds(friends(A,B)).
% A person who was out of town cannot be
%the murderer:
:- agent(P), holds(murderer(P)),
holds(outoftown(P)).

% The murderer is either andy, ben or cole,
%(exclusively):
1{murderer(andy);murderer(ben);
murderer(cole)}1.

% For display:
show murderer/1.
% Solution:
% Answer: 1
% murderer(ben).
% SATISFIABLE
```

Acknowledgements

The work was supported by Benemérita Universidad Autónoma de Puebla.

References

1. **Bohnet, I. & Frey, B. S. (1999).** The sound of silence in prisoner's dilemma and dictator games. *Journal of economic behavior & organization*, Vol. 38, No. 1, pp. 43–57.

ISSN 2007-9737

2. **Dyne, L. V., Ang, S., & Botero, I. C. (2003).** Conceptualizing employee silence and employee voice as multidimensional constructs. *Journal of Management Studies*, Vol. 40, No. 6, pp. 1359–1392.

3. **Eco, U. & Cantarell, F. S. (1994).** *Signo*. Labor.

4. **Garg, D. & Abadi, M. (2008).** A modal deconstruction of access control logics. *International Conference on Foundations of Software Science and Computational Structures*, Springer, pp. 216–230.

5. **Gebser, M., Kaminski, R., Kaufmann, B., Ostrowski, M., Schaub, T., & Wanko, P. (2016).** Theory solving made easy with clingo 5. *OASIcs-OpenAccess Series in Informatics*, volume 52, Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

6. **Gelfond, M. & Kahl, Y. (2014).** *Knowledge representation, reasoning, and the design of intelligent agents: The answer-set programming approach*. Cambridge University Press.

7. **Gödel, K. & Zum intuitionistischen Aussagenkalkül, Anzeiger Akademie der (1932).** Wissenschaften Wien, Math.-naturwissensch. *Klasse*, Vol. 69, pp. 65–66.

8. **Grice, H. P. (1975).** *Logic and conversation*.

9. **Khatchadourian, H. (2015).** *How to Do Things with Silence*, volume 63. Walter de Gruyter GmbH & Co KG.

10. **Kurzon, D. (1995).** The right of silence: A socio-pragmatic model of interpretation. *Journal of Pragmatics*, Vol. 23, No. 1, pp. 55–69.

11. **Lifschitz, V. (2008).** What is answer set programming? *AAAI*, volume 8, pp. 1594–1597.

12. **Pearce, D. (1999).** Stable inference as intuitionistic validity. *The Journal of Logic Programming*, Vol. 38, No. 1, pp. 79–91.

13. **Smullyan, R. M. (2012).** *Forever undecided*. Knopf.

14. **Van der Plas, J. (2016).** *Python data science handbook: Essential tools for working with data*. O'Reilly Media, Inc.

Article received on 30/10/2019; accepted on 03/03/2020.
Corresponding author is Aurelio López López.