

Constraint-based Temporal Reasoning for E-learning with LODE

Rosella Gennari and Ornella Mich

KRDB, CS Faculty, Free University of Bozen-Bolzano
Piazza Domenicani 3, 39100 Bolzano, Italy
`gennari@inf.unibz.it, mich@itc.it`

Abstract. LODE is a logic-based web tool for Italian deaf children. It aims at stimulating global reasoning on e-stories written in a verbal language. Presently, we are focusing on temporal reasoning, that is, LODE stimulates children to reason with global temporal relations between events possibly distant in a story. This is done through apt exercises and with the support of a constraint programming system. Children can also reinvent the e-story by rearranging its events along a new temporal order; it is the task of the constraint system to determine the consistency of the temporally reorganised story and provide children with feedback. To the best of our knowledge, LODE is the first e-learning tool for Italian deaf children that aims at stimulating global reasoning on whole e-stories.

Key words: constraint programming; automated temporal reasoning; e-learning; assistive technology.

1 Introduction

In the last years, much research in deaf studies and computer science has been devoted to applications for sign languages; roughly speaking, a *sign language* is a gestural-visual language with signs as lexical units, whereas a *verbal language* is an oral-auditive language with words as lexical units. Less attention seems to be paid to e-learning tools for improving the literacy of deaf children in verbal languages. Our LOgic-based e-tool for DEaf children (LODE) belongs in this latter class. In the following, we motivate the need of an e-learning tool such as LODE and then briefly outline our LODE tool.

1.1 The Problem

Learning to read and write effectively can be a difficult task for deaf people. Due to a limited exposition to the language in its spoken form in their first years of life, they lack the primary, natural means of acquiring literacy skills: “deaf children have unique communication needs: unable to hear the continuous, repeated flow of language interchange around them, they are not automatically exposed to the enormous amounts of language stimulation experienced by hearing children” [24].

Research on deaf subjects points out that deaf people hardly achieve verbal language literacy; they tend to reason on single episodes and show difficulties in formulating coherent global relations, such as temporal relations, between episodes of narratives in a verbal language [5], to the effect that their ability of reading does not often go beyond that of a eight-year old child [20]. As reported in [5], this attitude can also depend on the kind of “literacy interventions addressed to deaf children” which tend to “focus on single sentences and the grammatical aspects of text production”. A novel literacy e-tool for them should thus focus on global deductive reasoning.

1.2 Our Proposal

Given that limited literacy skills constitute an obstacle to the integration of deaf people into our society, our purpose is to develop an e-learning tool for deaf children which stimulates the global reasoning on texts written in verbal Italian. To this end, LODE presents children with e-stories and invites children to globally reason on each of them; that is, children are elicited to analyse and produce relations between events, even distant in the story, so that the relations are consistent with the story and possibly implicit in it.

In its current version, LODE focuses on a specific type of relations, namely, temporal relations; it narrates temporally rich stories and then stimulates children to create a coherent network of temporal relations out of each LODE’s story. In order to do so, LODE heavily employs an automated reasoner, namely, a constraint programming system.

Section 2 provides the essential background on reasoning with temporal relations by using constraint programming. Then the paper delves into LODE. Section 3 outlines the educational exercises of LODE and how they aim at stimulating global reasoning with temporal relations. The architecture of LODE is discussed in Sect. 4; there, we explain the pivotal role of constraint programming and how it allows us to implement the educational exercises of LODE. We relate LODE to other e-tools for deaf children in Sect. 5, highlighting its novelty in the e-learning landscape of e-tools for deaf children, and conclude with an assessment of our work in Sect. 7.

2 Background on Temporal Reasoning with Constraints

Temporal Reasoning is a branch of Artificial Intelligence (AI) and involves the *formal representation of time* and a computational *reasoning system* for it. An instance of a temporal reasoning problem is given by the following exercise of LODE; the excerpt is taken from a simplified version of *The Ugly Duckling* by H.C. Andersen.

Mammy duck is brooding: she has five eggs, four are small, and one is big. All of a sudden, while she is still brooding, the small eggshells crack and four little yellow ducklings peep out. Mammy duck watches the big

egg but sees no signs of cracking... So she decides to keep on brooding. After some days, while she is brooding, also the big eggshell cracks and an ugly grey duckling peeps out...

Exercise: *do the small eggshells crack before the big eggshell cracks?*

We use it in the remainder to explain the time representation, in Subs. 2.1, and the computational reasoning system, in Subs. 2.2, that LODE adopts.

2.1 Time Representation à la Allen

Here we adopt intervals as the primitive entities for representing time; each interval is uniquely associated with a time event. Between any two pairs of events, there is an *atomic Allen* relation, namely, a relation of the form

before, meets, overlaps, starts, during, finishes, equals

or rel^{-1} , where rel is one of the above relations and rel^{-1} is the inverse of rel . See Fig. 1 for an intuitive graphical representation of the atomic Allen relations between two events, $e1$ and $e2$. The Allen relations are employed whenever

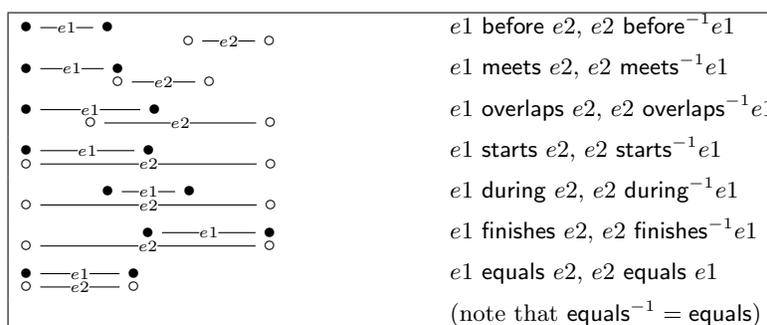


Fig. 1. The atomic Allen relations.

temporal information boils down to qualitative relations between events, e.g.,

“the small eggshells crack *while* Mammy duck broods”; (1)

in terms of the Allen relations, the sentence (1) states that the relation *during* can hold between the event “the small eggshells crack” and the event “Mammy duck broods”. By way of contrast, “the small eggshells crack *after 2 days*” is a quantitative temporal information.

As Allen arguments [1], “uncertain information” can be represented by means of unions of the atomic Allen relations. Formally, let A denote the class of the atomic Allen relations. Then the class 2^A forms a relational algebra with the operations of union \cup , composition \odot , and inverse $^{-1}$. In particular, the composition operation is used in deductions with the Allen relations. For instance, the

result of the composition `before` \odot `before` is `before`; this means that

```
if  e1 before e2 and e2 before e3
then e1 before e3.
```

2.2 Constraint Programming for Automated Temporal Reasoning

Constraint programming is a flourishing programming paradigm implemented in several constraint programming systems, e.g., `ECLiPSe`, a Constraint Logic Programming (CLP) system. Performing automated temporal reasoning with constraint programming means first modelling a temporal problem in a suitable formalism and then solving it in an automated manner with a constraint programming system. In the following, we introduce the basics on constraint modelling and solving, taking them from [4].

Constraint modelling. Given the temporal reasoning problem above, the constraint programmer *models* it as a *constraint satisfaction problem*. In essence, this is given by

- finitely many variables, x_1, \dots, x_n ,
- each ranging on a domain D_i of values,
- and a set of constraints, namely, relations of the form $C \subseteq D_{i_1} \times \dots \times D_{i_m}$.

A tuple of domain values a_1, \dots, a_n for the variables x_1, \dots, x_n is a *solution* to the problem if it belongs to all the constraints of the problem. When a value a_i for x_i participates in a solution to the problem, we will say that a_i is *consistent* with the problem.

A temporal reasoning problem can be modelled as a constraint problem as done in [3]. We expound this modelling in the context of `LODE` in Sect. 4. As an example, take the temporal reasoning problem in p. 2. The ordered pair of events “the small eggshells crack” and “the big eggshell cracks” gives a variable; its domain is the set of the atomic Allen relations; the temporal reasoning problem restricts the domain of the variable to `before`, and this is a unary constraint. The relation `before` is consistent with the considered problem.

Constraint solving. Once a temporal reasoning problem is modelled as a constraint problem in a suitable programming language (e.g., `CLP`), a constraint programming system (e.g., `ECLiPSe`) can be invoked to *solve* it. In this setting, to solve a problem means

- to decide on the consistency of a relation with the problem, e.g., `before` between “the big eggshells crack” and “the small eggshell cracks”,
- or to deduce an/all the Allen relation/relations between two events, e.g., “the big eggshell cracks” and “the small eggshells crack”, which are consistent with the problem and implicit in it.

The LODE's exercises rely on both these solving capabilities of a constraint programming system, namely, that of deciding on and that of deducing new Allen relations, consistent with a LODE's story. First we need to present such exercises, then we can return to constraint programming by explaining its role in the LODE's architecture and in solving the exercises.

3 LODE: Educational Exercises

LODE presents a list of e-stories the child can choose among. They are simplified versions of traditional children tales, such as *The Ugly Duckling*, so that the language is more suitable to a 8-year old deaf child; they are also enriched with explicit temporal relations so as to focus the attention of the child on temporal reasoning. The child has to choose a story from the list in order to begin his/her work session.

3.1 Dictionary of Difficult Words

The first exercises build a sort of dictionary of the most unusual words for deaf children. Single words are proposed on the screen together with an image explaining their meaning and a short textual explanation; example sentences are also available. This preliminary phase simplifies the comprehension of the story and the association grapheme-meaning in beginning readers, a step which may be necessary with young deaf users. Children can also consult back the dictionary during the other work sessions of LODE.

Future versions of LODE will also feature a translation of the dictionary words into Italian Sign Language (*Lingua Italiana dei Segni*, LIS) to facilitate their comprehension to LIS speakers.

3.2 Global Reasoning Exercises

Then the chosen story is presented, split across different pages. There are two or three sentences with an explanatory image on each page. The text is visually predominant so that the child must concentrate on it and not on the image. Every few pages, the child starts a new exercise session for *reasoning* on the tale. In LODE, we have two main types of reasoning exercises: comprehension exercises and production exercises.

Comprehension. In comprehension exercises, the child is presented with temporal relations connecting events of the story; the relations may be implicit in it. More precisely, the child is proposed four temporal relations. The child is asked to judge which relations are inconsistent with the text he/she has already read, playing the role of the teacher who eliminates the incoherent ones. The four cases are constructed with the assistance of the constraint programming system to determine which temporal relations are (in)consistent with the story.

Production. The child is asked to tackle three main kinds of production exercises, explained in the following.

- P1. The child is shown an unordered sequence of temporal events extracted from the story; his/her task is to drag the events into the right temporal sequence, namely, one which is consistent with the story. The consistency is decided by the constraint programming system.
- P2. The child is shown scattered sentence units extracted from the given story; then he/she should compose a grammatically correct sentence with them, forming a temporal relation consistent with the story and which may be implicit in the story. For instance, suppose that the available sentence units are: before, while, after, the big eggshell cracks, the small eggshells crack. Two are the possible correct sentences the child can compose, consistent with the tale. One is: the small eggshells crack before the big eggshell cracks. The other sentence is: the big eggshell cracks after the small eggshells crack. If the child composes a wrong sentence, because it is ungrammatical or inconsistent with the story, LODE will suggest how to correct the sentence with the help of the constraint programming system and a natural language processor for Italian.
- P3. The child can also reinvent the chosen e-story by using selected events of the e-story. More precisely, first LODE proposes a set of events extracted from the chosen e-story. Then the child re-creates his/her own story by reordering the events along the timeline, to the child's liking. Anytime, the child can check whether the temporal relations are consistent in his/her story, e.g., it is not the case that an event is simultaneously before and during another; he/she can also ask for the assistance of LODE, or better, of the constraint programming system of LODE in setting new temporal relations between events, consistent with his/her story.

3.3 Final Remarks

LODE assists children in all the exercises, e.g., they are shown the events they may reason on. The difficulty of the exercises for reasoning increases with the portion of the story the child has to reason on. Thus, first LODE proposes the simpler exercises: these relate two temporal events which occur in the portion of the tale, temporally rich, that the child has just read. If the score reached so far by the child is reasonably good, then LODE proposes the more challenging exercises, namely, those that require a deep global understanding of the story and the creation of global temporal relations: these exercises relate two temporal events, one of the current session and the other of a previous session—the farther is this session the more difficult is the exercise.

Moreover, note that the comprehension exercises aim at stimulating the deduction of global relations between events of the story; the production exercises demand this and something else, that is, to compose parts of the story. Therefore, the production exercises also aim at teaching children Italian grammar. To stimulate children to write, web users of LODE will also be invited to collaborate and exchange their productions via a blog.

Through all its exercises, LODE stimulates children to learn and reason on e-stories in an inductive and implicit manner.

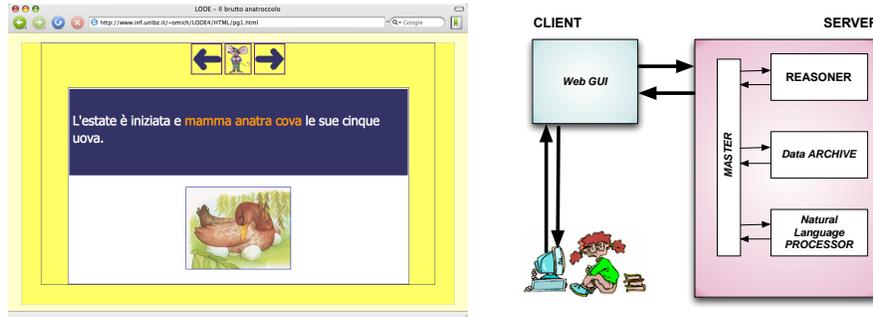


Fig. 2. LODE: a screen-shot of a client session on the left and the client-server architecture on the right.

4 LODE: the Architecture and the Constraint-based Module

LODE has a web-based client-server architecture; see Fig. 2. We opt for this for several reasons. First, it makes LODE independent of the Operating System (OS). Therefore, users are free to run LODE on the preferred OS with their web browser. Second, it makes easier the updating of LODE; new features can be implemented without affecting the users, e.g., no need of installing new versions of LODE. Third, a web-based architecture promotes collaborative study: when they are on-line, the LODE users can work together and exchange their own stories or comments.

4.1 The Modular Architecture

The client is a graphical user interface (GUI); see the left side of Fig. 2. This is an AJAX application compatible with most web browsers, e.g., Firefox-Mozilla, Internet Explorer, Safari, Opera. It works as the interface between the LODE user and the real system, the server, which runs on a remote machine.

The server has a modular structure. The main modules are: 1) the *stories' database*, 2) the *Constraint-based Automated Reasoner* and 3) the *Natural Language Processor*; see the right side Fig. 2.

- 1) The current stories' database is a simple repository structured as a file system. It contains temporally enriched versions of famous children stories, in XHTML format. Events and relations are tagged in XHTML à la TimeML [23], the main difference being that the used Allen relations can be non-atomic; see Fig. 3 for an example.

```

<EVENT class="E1">
  The small eggshells crack
</EVENT>
<EVENT class="E2">
  Mammy duck watches the big egg
</EVENT>
<TLINK event="E1" relatedToEvent="E2" relType="before OR meets"/>
<EVENT class="E3">
  The big eggshell cracks
</EVENT>
<TLINK event="E2" relatedToEvent="E3" relType="before"/>

```

Fig. 3. A sample of a tagged story in LODE.

- 2) The *Constraint-based Automated Reasoner* is composed of three main parts:
 - a) ECL^iPS^e , the constraint (logic) programming system; b) the *knowledge base*, namely, an ECL^iPS^e program with the inverse and composition operations for the Allen relations; c) the *domain knowledge*, consisting of constraint problems modelling the temporal information of the e-stories in the database. The Constraint-based Automated Reasoner is employed in the composition and production exercises to assist children in their deductions.
- 3) The *Natural Language Processor* will check if the user's sentences are grammatically correct in the production exercises.

The most important module of the architecture is the Constraint-based Automated Reasoner.

4.2 The Constraint-based Automated Reasoner

Each e-story in the database is modelled as a constraint problem in a semi-automated manner; the problem is included in the domain knowledge and then solved by the chosen constraint programming system. Let us make precise what we mean by modelling and solving in the context of LODE; for their more general presentation, refer back to Subsect. 2.2.

Constraint modelling in LODE. First we illustrate the main steps of the modelling with an example and then we generalise it.

Example. Hereby is part of the excerpt of the LODE's story introduced in Sect. 2:

The small eggshells crack. Mammy duck watches the big
egg. . . After some days, [...] the big eggshell cracks. (2)

The excerpt has 3 temporal events, tagged in the XHTML code as illustrated in Fig. 3:

- “the small eggshells crack”, classified as E1;
- “Mammy duck watches the big egg”, classified as E2;
- “the big eggshell cracks”, classified as E3;

The corresponding constraint problem, stored in the domain knowledge of ECL^iPS^e , has variables

- E1E2 with domain the set of the atomic Allen relations between E1 and E2,
- E2E3 with domain the set of the atomic Allen relations between E2 and E3,
- E1E3 with domain the set of the atomic Allen relations between E1 and E3,

and constraints of two main types:

- C1. unary constraints formalising the temporal relations of the excerpt (which are tagged with TLINK in the XHTML code in Fig. 3), that is: a constraint stating that E1E2 is *before* or *meets*; a constraint stating that E2E3 is *before*;
- C2. a ternary constraint on E1, E2 and E3 formalising the Allen composition operation; the ternary constraint on E1, E2 and E3 yields that *before* can hold between E1 and E3 since *before* can hold between E1 and E2, and between E2 and E3.

See Fig. 4 for the straightforward translation of the above constraints in the language of ECL^iPS^e .

```

E1E2 &:: [before,meets],
E2E3 &:: [before],
allen_composition(E1E2, E2E3, E1E3)
        
```

Fig. 4. The constraint model in ECL^iPS^e corresponding to the sample in Fig. 3.

Model. As illustrated in the above example, here we adopt the same constraint model for qualitative temporal reasoning as in [3]. More precisely, the knowledge base contains

- the set A of atomic Allen relations providing the domain of the variables,
- a ternary relation, namely, `allen_composition`, for the composition of the Allen relations, e.g., `allen_composition(before, before, before)`.

Let $E := \{E1, \dots, En\}$ be the set of events tagged in a story of LODE. Then the domain knowledge of the constraint-based module contains the following:

- variables E_iE_j of ordered pairs of events E_i and E_j of E ; the domain of the E_iE_j is the set A of the atomic Allen relations;
- for each variable E_iE_j , a unary constraint $C(E_iE_j)$ on E_iE_j which restricts the relations from A to those that are declared in the story (e.g., see item C1 above);

- for each triple E_i , E_j and E_k of events from E , a ternary constraint $C(E_iE_j, E_jE_k, E_iE_k)$ stating that each relation in E_iE_k comes from the composition of E_iE_j and E_jE_k (e.g., see item C2 above).

Constraint solving in LODE. In LODE, we employ both the following solving capabilities of a constraint programming system such as ECL^iPS^e :

- that of deciding on the consistency of a temporal constraint problem in the domain knowledge,
- that of deducing a/all the consistent Allen relation/relations between two events of a temporal constraint problem in the domain knowledge.

These constraint solving capabilities are employed in LODE in two main manners:

- to assist the LODE developers in the creation of the exercises; the constraint programming system allows us to automatically *deduce* the relations between events which are consistent with the e-story, such as *before* between E_1 and E_3 of (2);
- to assist children in the resolution of exercises. In the comprehension exercises and in some production exercises (see Sect. 3), the constraint programming system can be used to *decide* on the consistency of the relations proposed by children. Moreover, in specific production exercises, a child can recreate his/her own story, setting a different order among events of the story along the timeline; then he/she can (transparently) query the constraint system to *decide* on the consistency of the recreated story or to *deduce* a/the consistent relation/relations between the considered events.

Final remarks: why constraint programming in LODE. Constraint programming is thus a backbone of LODE; in the following, we try to sum up the main reasons for using constraint programming in LODE; we split our summary in two parts, one for constraint modelling, the other for constraint solving.

Constraint modelling.

- First of all, the constraint modelling is human readable as it is closer to the structure of the temporal reasoning problem; by way of contrast, a SAT encoding would be less manageable and readable for the modeller.
- The modelling can also be semi-automated: events and explicit relations of an e-story are first tagged in XHTML; then a script automatically reads the XHTML tags and translates them into variables and constraints of the corresponding constraint problem.
- Last but not least, the constraint model for temporal reasoning that we adopted here allows us to employ a generic constraint programming system, such as ECL^iPS^e , as is. In other words, this constraint model does not demand to implement dedicated algorithms for temporal reasoning with Allen relations; for instance, with this model, we can exploit constraint propagation in the form of hyper-arc consistency which is already available in the *propia* library of ECL^iPS^e ; instead, with the constraint model described in [12], we should use and implement path consistency.

Constraint solving.

- Thanks to the solving capabilities of the constraint programming system, users of LODE can create new temporally consistent e-stories out of the available ones: a user of LODE can query the constraint programming system to decide on the consistency of the relations in his/her story, or to deduce new temporal relations that can be consistently added to his/her story.
- Using constraint programming spares the LODE developers manual work in the creation of the exercises, a task which is tedious and prone to human errors: instead of creating, manually, a series of exercises, e-stories are first modelled as constraint problems; then the relations of the reasoning exercises, introduced in Sect. 3, are deduced using the constraint programming system.
- Moreover, the exercises created in this manner can be easily updated, new exercises can be introduced and solved on the fly using the constraint programming system.
- Although the efficiency in solving temporal problems is not a critical feature of LODE at present, it may become a critical issue in future and more ambitious versions of the tool; in this respect, the model and the different constraint propagation and search procedures already available in the chosen constraint programming system will be of pivotal importance.

5 Related Work

Currently, research in deaf studies and computer science seems to mostly revolve around applications for sign languages, such as LIS, e.g., see [7,16]. Considerable less attention seems to be devoted to the development of e-learning tools for the literacy of deaf children in verbal languages. This impression is confirmed by our overview of this type of e-learning tools. We present the main ones related to LODE in the remainder.

5.1 Italian Tools

In Italy, three systems were developed in the '90s to tackle different aspects of verbal Italian lexicon or grammar: *Articoli* [6] aims at teaching Italian articles (e.g., gender agreement); *Carotino* [8] is an interactive tool for teaching simple Italian phrases; *Pro-Peanuts* [21] deals with the correct use of pronouns. Please, note that none of these tools were developed exclusively for deaf children. This is clear in a tool such as Carotino, where instructions for children presume a certain knowledge of written verbal Italian and do not focus on issues with verbal Italian which are specific to deaf children.

There are several tools that aim at teaching stories to deaf or hearing-impaired children. In order to facilitate the integration of a deaf girl into an Italian primary school, teachers and students of the school created *Fabulis* [11], a collection of famous stories for children narrated using text and images, based

on gestures and LIS signs. Another application born at school is *Nuvolina* [19], the result of a project realised in a fourth class of an Italian primary school. Also this project aimed at integrating a deaf girl into the class. *Nuvolina* is a multimedia tale with contents in Italian, English and French, written and spoken. The version in verbal Italian is also presented in LIS by means of short videos.

In the area of bilingual tools, employing LIS and verbal Italian, we found *Gli Animali della Savana* [2]. This is a multimedia software based on text, images and videos, featuring an actor who translates the written text in LIS; assisted by a cartoon (a lion), the user navigates through a series of pages presenting the life of 10 wild animals. A more recent and ambitious project is *Tell me a Dictionary* [15,22], the purpose of which is to offer both deaf and hearing children an interactive instrument to discover and compare the lexicon of LIS and Italian. *Tell me a Dictionary* is a multimedia series of six DVDs plus book volumes [15].

We also found references to a tool developed in 1994, *Corso di Lettura* [10], which aims at improving the reading capabilities of hearing-impaired children. Alas, we could not find further information on the tool besides this.

5.2 English Tools

The primary goal of the ICICLE [14,17] researchers was to employ natural language processing and generation to tutor deaf students on their written English. ICICLE's interaction with the user takes the form of a cycle of user input and system response. The cycle begins when a user submits a piece of writing to be reviewed by the system. The system then performs a syntactic analysis on this writing, determines its errors, and constructs a response in the form of tutorial feedback. This feedback is aimed towards making the student aware of the nature of the errors found in the writing and giving him or her the information needed to correct them.

CornerStones [18] is a technology-infused approach to literacy development for early primary children who are deaf or hard of hearing. Academic experts in literacy and deafness, along with teachers of deaf students participated in its development. An essential element of Cornerstones is a story taken from the PBS's literacy series *Between the Lions*, complemented by versions of the story in American Sign Language and other visual-spatial systems for communicating with deaf children. Cornerstones developers evaluated their system with children and teachers and results of their evaluation demonstrated an increase in students' knowledge of selected words from pre-test to post-test.

FtL [9] has not been developed for deaf or hard of hearing children, but this type of users has also been considered. FtL is a comprehensive computer-based reading program that has been designed to teach beginning and early readers to read with good comprehension. FtL consists of three integrated components: a Managed Learning Environment (MLE) that tracks and displays student progress and manages an individual study plan for each student; Foundational Skills Reading Exercises, which teach and practice basic reading skills, such as alphabet knowledge and word decoding, providing the foundation for fluent reading; Interactive Books, which represent the state of the art in integration of

human language and animation technologies to enable conversational interaction with a Virtual Tutor that teaches fluent reading and comprehension of text. The final evaluation of FtL produced significant learning gains for letter and word recognition for kindergarten students.

5.3 Final Remarks

According to our overview of Italian and non-Italian projects for deaf children, and to the best of our knowledge, LODE is the first web e-learning tool that tackles literacy issues of deaf children which go beyond the lexicon and grammar of a verbal language, that is: LODE addresses *global deductive reasoning* on stories, and LODE does it with the support of a constraint-based automated reasoner. Table 1 gives a summative analysis of LODE, comparing it with the principal and assessed tools for the literacy improvement of deaf children.

6 Future Work

LODE is being evaluated into three main phases. The first evaluation phase is almost over; LIS interpreters and teachers for deaf children from the Talking Hands Association and the Italian National Institute for the Assistance and Protection of the Deaf (ENS), logopaedists and a cognitive psychologist have tested a preliminary version of LODE and provided us with positive informative feedback on its learning goals and strategies.

The second and third evaluation phases will directly involve deaf children; the second phase will be done in class with the assistance of a teacher from ENS and a cognitive psychologist; the third phase will involve children at home. The second evaluation phase will test the usability of LODE and, in particular, which is the most effective way of visually representing the exercises of LODE. The third evaluation phase will test the effectiveness of LODE and will involve circa 15 deaf children. As confirmed by our own experience, the assistance of children and their teachers is fundamental for developing a tool interesting and useful for children.

Last but not least, the application of constraint programming in LODE goes beyond temporal reasoning: after the evaluation phase, we are going to extend LODE to other kinds of global deductive reasoning, critical for deaf children and for which constraint programming can be beneficial.

7 Conclusions

In this paper, we presented our e-learning web-based tool for deaf children: LODE. The tool tackles problematic issues encountered in their written productions in a verbal language, related to deductive global reasoning; see Subject. 1.1. Presently, we are focusing on global temporal reasoning. In Sect. 2, we introduced

Table 1. Tools for literacy improvement: a comparative analysis

	ICICLE	Cornerstones	FtL	LODE
Content type	user's input	famous stories	interactive books	famous stories
Use of sign language	yes	yes	no	planned
Dialogue interface	yes	no	yes	yes
Reading comprehension	no	no	yes	yes
Active feedback	yes	no	no	yes
Syntax/grammar analysis	yes	no	no	planned
Speech recognition	no	no	yes	no
Global deductive reasoning	no	no	no	yes

temporal reasoning with Allen relations and motivated their use in the context of LODE; note that the relations are visually represented as explained in [13] because LODE is for deaf children. In Sect. 3, we summed up the educational exercises of LODE; we showed how the so-called reasoning exercises aim at stimulating global reasoning on stories, and not on isolated sentences of the stories.

In particular, these exercises of LODE demand the use of the automated reasoner embedded in LODE, namely, a constraint programming system. In Sect. 4, explaining the web-based architecture of LODE, we presented in details and motivated such a use of constraint programming in LODE, both in terms of the adopted constraint model for temporal reasoning problems and in terms of the constraint solving capabilities of a system such as ECLⁱPS^e.

In Sect. 5, we overviewed and compared with LODE several e-learning tools that address literacy issues of deaf or hearing-impaired people. According to our overview and to the best of our knowledge, LODE is the first web-based e-learning tool which aims at stimulating global deductive reasoning on whole e-stories in a verbal language, such as Italian, by employing a constraint-based automated reasoner.

Acknowledgement. Among the others, we wish to thank B. Arfé, S. Brand, the ENS of Trento (in particular, N. Hy Thien), Talking Hands (in particular, F. De Carli), M. Valente and the anonymous referees for their assistance.

References

1. J.F. Allen. Maintaining Knowledge about Temporal Intervals. *Comm. of ACM*, 26(832–843), 1983.
2. Animali della Savana, Gli. Retrieved October 24, 2006, from http://www.areato.org/noquadri/ausiliDinamici/AusDnm.00_Titolo.Asp?IDAUSILIO=106&FORMATO=G.
3. K.R. Apt and S. Brand. Constraint-based Qualitative Simulation. In *Proc. of the 12th International Symposium on Temporal Representation and Reasoning (TIME'05)*, pages 26–34, 2005.

4. K.R. Apt and M.G. Wallace. *Constraint Logic Programming using ECLiPSe*. Cambridge University Press, 2006.
5. B. Arfé and P. Boscolo. Causal Coherence in Deaf and Hearing Students' Written Narratives. *Discourse Processes*, 42(3):271–300, 2006.
6. Articoli. Retrieved October 24, 2006, from <http://www.anastasis.it/AMBIENTI/NodoCMS/CaricaPagina.asp?ID=36>.
7. S. Bartolini, P. Bennati, and R. Giorgi. Sistema per la Traduzione in Lingua Italiana dei Segni: Blue Sign Translator / Wireless Sign System. In *Proc. of the 50th AIES National Conference*, 2004.
8. Carotino. Retrieved October 24, 2006, from <http://www.anastasis.it/AMBIENTI/NodoCMS/CaricaPagina.asp?ID=40>.
9. R. Cole, D. Massaro, B. Rundle, K. Shobaki, J. Wouters, M. Cohen, J. Beskow, P. Stone, P. Connors, A. Tarachow, and D. Solcher. New Tools for Interactive Speech and Language Training: Using Animated Conversational Agents in the Classrooms of Profoundly Deaf Children.
10. Corso di Lettura. Retrieved October 24, 2006, from <http://www.anastasis.it/AMBIENTI/NodoCMS/CaricaPagina.asp?ID=35>.
11. Fabulis. Retrieved October 24, 2006, from <http://www.bonavitacola.net/fabulis/>.
12. R. Gennari. Temporal Reasoning and Constraint Programming: a Survey. *CWI Quarterly*, 11(2–3), 1998, Amsterdam.
13. R. Gennari and O. Mich. LODE: Global Reasoning on e-Stories for Deaf Children. In *Proc. of the Engineered Applications of Semantic Web (SWEA 2007)*, invited session of the 11th KES conference, to be published in the LNAI/LNCS series, 2007.
14. ICICLE. Retrieved January 24, 2007, from <http://www.eecis.udel.edu/research/icicle/>.
15. E. Insolera, G. Militano, E. Radutzky, and A. Rossini. Pilot Learning Strategies in Step with New Technologies: LIS and Italian in a Bilingual Multimedia Context 'Tell me a Dictionary'. In C. Vettori, editor, *Proc. of the 2nd Workshop on the Representation and Processing of Sign Languages, LREC 2006*, 2006.
16. M. Mertzani, C. Denmark, and L. Day. Forming Sign Language Learning Environments in Cyberspace. In C. Vettori, editor, *Proc. of the 2nd Workshop on the Representation and Processing of Sign Languages, LREC 2006*, 2006.
17. L.N. Michaud, K.F. McCoy, and C. Pennington. An Intelligent Tutoring System for Deaf Learners of Written English. In *Proc. of ASSETS'00*, 2000.
18. NCAM-CornerStones. Retrieved January 24, 2007, from <http://ncam.wgbh.org/cornerstones/overview.html>.
19. Nuvolina. Retrieved October 24, 2006, from <http://www.areato.org/noquadri/ausiliDinamici/AusDnm.01.Dettaglio.Asp?IDAUSILIO=229&IDSEZIONE=5&FORMATO=G&VETRINA=N>, 1998.
20. P.V. Paul. *Literacy and Deafness: the Development of Reading, Writing, and Literate Thought*. Allyn & Bacon, 1998.
21. Pro-peanuts. Retrieved October 24, 2006, from <http://www.ciscl.unisi.it/persona/chesi/laurea/str.htm>, 1998.
22. Tell me a Dictionary. Retrieved October 24, 2006, from <http://www.lismedia.it/demo01/home.html>, 2005.
23. TimeML. Retrieved March 10, 2007, from <http://www.cs.brandeis.edu/~jamesp/arda/time/>.
24. UNESCO. Education of Deaf Children and Young People. Guides for Special Needs Education. Paris, 1987.