World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:7, No:11, 2013

A Visualized Framework for Representing Uncertain and Incomplete Temporal Knowledge

Yue Wang, Jixin Ma, Brian Knight

Abstract**—**This paper presents a visualized computer aided case tool for non-expert, called Visual Time, for representing and reasoning about incomplete and uncertain temporal information. It is both expressive and versatile, allowing logical conjunctions and disjunctions of both absolute and relative temporal relations, such as "Before", "Meets", "Overlaps", "Starts", "During", and "Finishes", etc. In terms of a visualized framework, Visual Time provides a user-friendly environment for describing scenarios with rich temporal structure in natural language, which can be formatted as structured temporal phrases and modeled in terms of Temporal Relationship Diagrams (TRD). A TRD can be automatically and visually transformed into a corresponding Time Graph, supported by automatic consistency checker that derives a verdict to confirm if a given scenario is temporally consistent or inconsistent.

Keywords**—**Time Visualization, Uncertainty, Incompleteness, Consistency Checking.

I. Introduction

GENERALLY speaking, time plays the role of a common universal reference - everything appears to be related by its temporal reference. In particular, the representation and manipulation of natural human understanding of temporal phenomena is a fundamental field of research in Artificial Intelligence, which aims both to emulate human thinking, and to use the methods of human intelligence to underpin computerized solutions. It has been noted that absolute-time-stamping of temporal data provides an efficient indexing method for temporal systems, but suffers from the requirement that precise time values for all temporal data need to be available. Generally speaking, in the domain of Artificial Intelligence, temporal knowledge can be uncertain and incomplete due to various reasons:

- Time references may come from, or go into, more than one possible world (e.g., after arriving at Venice in the morning, the visitor may take a train in the afternoon, or a flight in the evening, to get to Rome);
- Temporal references may be only relative (e.g., "during the time when the officer was in his office", "after 9 o'clock", etc., which refer to times that are known only by their relative temporal relations to other temporal reference), rather than being absolute (e.g., "8 pm on the 8th of August 2008", "the last week of August 2008", which refer to times with absolute values);
- Temporal duration may be only relative (e.g., "less than 6 hours", "more than 12 years but less than 15 years", etc., which refer to some uncertain amount of temporal granularity), rather than being absolute (e.g., "31 minutes", "18 hours", etc., which refer to some certain amount of temporal granularity);
- One may only know event E_A occurred "Before" event $E_{\backslash B}$, without knowing their precise starting and finishing time, what happened between E_A and E_B, or how long was the delay between E_A and E_B.

Uncertain and Incomplete relative temporal knowledge such as these is typically derived from humans, where complete and absolute temporal information is rarely available and remembered for knowledge representation and reasoning. Various approaches to dealing with uncertain or incomplete temporal knowledge have been proposed, where most of them actually devoted themselves on specific applications [1]-[4]. Allen's interval-based time theory [5],[6] is a representative example of temporal systems addressing relative temporal relations including "Meets", "Met_by", "Equal", "Before", "After", "Overlaps", "Overlapped_by", "Starts", "Starts_by", "During", "Contains", "Finishes" and "Finished_by". It has been claimed in the literature that time intervals are more suited for expression of common sense temporal knowledge, especially in the domain of linguistics and artificial intelligence. In addition, approaches like that of Allen that treat intervals as primitive temporal elements can successfully bypass puzzles like the Dividing Instant Problem [5], [7], [8] which is in fact an ancient historical puzzle encountered when attempting to represent what happens at the boundary point that divides two successive intervals. However, as Galton shows in his critical examination of Allen's interval logic [9], a theory of time based only on intervals is not adequate for reasoning correctly about continuous change. In fact, many common sense situations suggest the need for including time points in the temporal ontology as an entity different from intervals. For instance, it is intuitive and convenient to say that instantaneous events such as "The database was updated at 00:00am" [10], "The light was automatically switched on at 8:00pm", and so on, occur at time points rather than over intervals (not even at Allen and Hayes' "moments" [11], no matter how small they are). Therefore, for general treatments, it is appropriate to include both points and intervals as primitives in the underlying time model, for making temporal reference to instantaneous phenomena with zero duration, and periodic phenomena which last for some positive duration, respectively.

The objective of this paper is to propose a visualized framework to assist representing and reasoning about uncertain and incomplete temporal knowledge, which could be used by non-experts to quickly visualize and possibly correct

Yue Wang,Jixin Ma, andBrian Knight are with the University of Greenwich, SE109LS UK (e-mail: Y.wang@gre.ac.uk, J.ma@gre.ac.uk, B.Knight@ greenwich.ac.uk).

inconsistencies. Section II introduces a graphical representation for temporal knowledge, based on a time theory that treats both points and intervals as primitive on the same footing. It allows both logical conjunctions and disjunctions of relative temporal relations. Section III presents a visualized case tool, called Visual Time, for representing and reasoning about incomplete and uncertain temporal knowledge. Visual Time provides a user-friendly environment for describing scenarios with rich temporal structure in natural language, which can be formatted as structured temporal phrases and modeled in terms of Temporal Relationship Diagrams (TRD). A TRD can be automatically and visually expressed as a Time Graph, supported by an automatic consistency checker, which can derive a verdict that confirms if a given scenario is temporally consistent or inconsistent. Finally, Section IV provides the summary and concludes the paper.

II. The Temporal Basis

A. The Underline Time Theory

A time theory can be solely based on points [12], [13], solely based on intervals [5], [6], or based on both points and intervals [14].

As discussed in the introduction above, for the reason of general treatment, in this paper, we shall simply adopt the time theory proposed in [14], which takes a nonempty set, **Time**, of primitive time elements, with an immediate predecessor relation, Meets, over time elements, and a duration assignment function, Dur, from time elements to non-negative real numbers. If Dur(t) = 0, then t is called a point; otherwise, that is Dur(t) >0, t is called an interval. The basic set of axioms concerning the triad (**Time**, Meets, Dur) are as below:

T1. $\forall t_1,t_2,t_3,t_4(\text{Meets}(t_1, t_2) \wedge \text{Meets}(t_1, t_3) \wedge \text{Meets}(t_4, t_2) \Rightarrow \text{Meets}(t_4, t_3))$

That is, if a time element meets two other time elements, then any time element that meets one of these two must also meets the other.

T2. $\forall t \exists t_1,t_2(\text{Meets}(t_1, t) \wedge \text{Meets}(t, t_2))$

That is, each time element has at least one immediate predecessor, as well as at least one immediate successor.

T3. $\forall t_1,t_2,t_3,t_4(\text{Meets}(t_1, t_2) \wedge \text{Meets}(t_3, t_4)$
$\Rightarrow \text{Meets}(t_1, t_4)$
$\nabla \exists t'(\text{Meets}(t_1, t') \wedge \text{Meets}(t', t_4))$
$\nabla \exists t''(\text{Meets}(t_3, t'') \wedge \text{Meets}(t'', t_2)))$

where∇ stands for "exclusive or". That is, any two meeting places are either identical or there is at least a time element standing between the two meeting places if they are not identical.

T4. $\forall t_1,t_2,t_3,t_4(\text{Meets}(t_3, t_1) \wedge \text{Meets}(t_1, t_4) \wedge \text{Meets}(t_3, t_2) \wedge \text{Meets}(t_2, t_4)) \Rightarrow t_1 = t_2)$

That is, the time element between any two meeting places is unique.

N.B. For any two adjacent time elements, that is time elements t_1 and t_2 such that Meets(t_1, t_2), $t_1 \oplus t_2$ denotes their ordered union.

T5. $\forall t_1,t_2(\text{Meets}(t_1, t_2) \Rightarrow \text{Dur}(t_1) > 0 \vee \text{Dur}(t_2) > 0)$

That is, time elements with zero duration cannot meet each other.

T6. $\forall t_1,t_2(\text{Meets}(t_1, t_2) \Rightarrow \text{Dur}(t_1 \oplus t_2) = \text{Dur}(t_1) + \text{Dur}(t_2))$

As emphasized in the introduction, in the domain of Artificial Intelligence, temporal knowledge can be uncertain and incomplete. On one hand, for a given pair of time elements t_1 and t_2, it may be unknown which of the 30 possible temporal relations as classified in the above certainly holds between t_1 and t_2. We shall formalize this uncertain temporal knowledge in term of temporal relations jointed by disjunctive connectives. On the other hand, for a given situation, the corresponding temporal knowledge of what time elements are involved, and what are the exact durations of these time elements, may be only partially known.

Analogous to the 13 relations introduced by Allen, accordingly, 30 exclusive temporal relations over time elements including both time points and time intervals can be concluded, which can be derived from the single Meets order relation and classified into the following 4 groups:

- Relations relating a point to a point:
  {Equal, Before, After}
- Relations relating a point to an interval:
  {Before, Meets, Starts, During, Finishes, Met-by, After}
- Relations relating an interval to a point:
  {Before, Meets, Started-by, Contains, Finished-by, Met-by, After}
- Relations relating an interval to an interval:
  {Equal, Before, Meets, Overlaps, Starts, During, Finishes, Finished-by, Contains, Started-by, Overlapped-by, Met-by, After}

In fact, each of these relative relations can be defined in terms of the single relation Meets. For instance:

$$\text{Before}(t_1, t_2) \Leftrightarrow \exists t(\text{Meets}(t_1, t) \wedge \text{Meets}(t, t_2))$$

Therefore, all the knowledge about the temporal relations over a given collection of time elements (points and/or intervals) can be transformed and stored as a table of Meets relations in the knowledge base.

In this paper, we shall use a triad (T, M, D) to express the temporal reference of a given collection of temporal knowledge, where:

- T = $\{t_1, \ldots, t_n\}$ is a finite set of time elements, expressing the knowledge (possibly incomplete) of what time elements are involved;

- M = {Meets(t_i, $t_{i(1)}$) ∧ … ∧Meets(t_i, $t_{i(j)}$) | for some i, where 1≤i, i(1), i(j), j≤ n} is a collection of disjunctions of Meets relations over T, expressing the knowledge (possibly incomplete) as how the time elements in T are related to each other by the Meets relations.
- D = {Dur(t_i) = r_i | for some i where 1 ≤ i ≤ n} is a collection of duration assignments (possibly incomplete) to time elements in T.

A temporal reference G = (T, M, D) can be graphically expressed in terms of a directed, partially weighted simple graph G, called temporal graph, in which:

(1) Each time element is denoted as an arrowed-arc with a beginning-node and an ending node; and for time elements with known duration, the corresponding arcs are weighted by their durations respectively.

(2) The relation Meets(t_i, t_j) is presented by means of unifying the ending-note of time element t_i and the beginning-node of time element t_j. In other words, Meets(t_i, t_j)is denoted by the node structure where t_i is an in-arc and t_j is an out-arc of a same node, respectively.

(3) Logical expressions ("∧" and "∨") of Meets relations are presented as below, respectively:

a. Meets(t_i, t_j) ∧Meets(t_i, t_k) is denote by defining t_i as an in-arc and t_j and t_k as two out-arcs of the same node, respectively(see Fig. 1 (b)).

b. Meets(t_i, t_k) ∧Meets(t_i, t_k) is denote by defining t_i and $t_j$as two in-arcs and t_k as an out-arcs of the same node, respectively(see Fig. 1 (a)).

c. Meets(t_i, t_j) ∨ Meets(t_i, t_k) is denoted by definingt$_i$ as duplicated identical out-arcs of a node, and defining one of the two t_is as an in-arc and t_j as an out-arc of another node; and defining the other t_i as as an in-arc and t_k as an out-arc of the third node respectively (see Fig. 1 (c)).

d. Meets(t_i, t_k) ∨ Meets(t_i, t_k) is denoted by definingt$_k$ as duplicated identical in-arcs of a node, and defining t_i as an in-arc and one the two t_ks as an out-arc of another node; and defining t_i as as an in-arc and the other t_k as an out-arc of the third node respectively (see Fig. 1 (d)).

Combinations of conjunction and disjunction can also be denoted:

e. Meets(t_{i1},t_k)∧Meets(t_{i2},t_k)∨Meets(t_j,t_k)(see Fig. 1 (e))
f. Meets(t_i,t_{k1})∧Meets(t_i,t_{k2})∨Meets(t_i,t_j)(see Fig. 1 (f))
g. Meets(t_i,t_{k1})∧(Meets(t_i,t_{k2}) ∨Meets(t_i,t_j))(see Fig. 1 (g))
h. Meets(t_{i1},t_k)∧(Meets(t_{i2},t_k)∨Meets(t_j,t_k))(see Fig. 1 (h))

As an illustration, consider the following temporal reference G = (T, M, D), where, for the reason of simple expression, comma "," in M and D stands for logical connective "∧":

T = {t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8, t_9, t_{10}}

M = {Meets(t_1,t_2), (Meets(t_2,t_3)∨Meets(t_2,t_4)),
Meets(t_3,t_7), Meets(t_3,t_8), Meets(t_4,t_5), Meets(t_4,t_6), Meets(t_5,t_7), Meets(t_5,t_8), Meets(t_7,t_9), Meets(t_7,t_{10}), Meets(t_8,t_{12}),Meets(t_6,t_{11})∨Meets(t_9,t_{11})∨Meets(t_{10},t_{11})),
Meets(t_{11},t_{12})}

D = {Dur(t_2)=1, Dur(t_3)=2.3, Dur(t_6)=3, Dur(t_8)=2.6,
Dur(t_{10})=0.8, Dur(t_{12})=1}

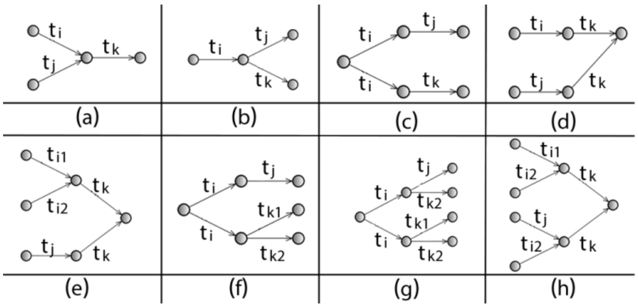

Fig. 1 Graphical representations of Meets relations

The temporal graph (T, M, D) is shown in Fig. 2.

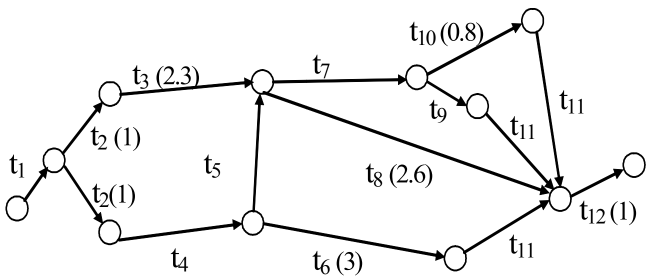

Fig. 2 Graphical representation of G = (T, M, D)

B. Consistency Checking

A temporal reference (T, M, D) is defined as temporal consistent if at least one of its temporal scenarios is temporal consistent.

The necessary and sufficient condition for the consistency of a temporal scenario G_s can be given as below:

1) For each simple circuit in G_s, the directed sum of weights is zero;
2) For any two adjacent time elements, the directed sum of weights is bigger than zero.

Here, condition 1) guarantees that there exists a valid duration assignment function Dur to the time elements in G_s consistent with D; and condition 2) ensures that no two time points meet each other, that is between any two time points, there is an interval standing between them.

The consistency checking for a temporal scenario with duration constraints involves searching for simple circuits, and constructing a numerical constraint for each circuit. The existence of a solution(s) to this set of constraints implies the consistency of the temporal scenario and hence of the temporal reference, where each solution gives a possible case for that can subsume the addressed temporal scenario. In fact, the consistency checker for temporal references can be transformed into linear programming problem.

For instance, the temporal scenarios G_s = (T_s, M_s, D_s) is consistent since one of its temporal scenarios, e.g., temporal scenario G_s as shown in Fig. 2, is consistent. In fact, by assigning duration value of 1 to t_7, and 0.8 to t_{11}, will make temporal scenario G_s shown in Fig. 3 consistent.

III. Visual Time

In what follows, we present a visualized framework for representing and reasoning about uncertain and incomplete temporal knowledge. Various views, including "Natural Language Description", "Time Relation Diagram" (TRD), "Temporal Relation", "Meets Table" and "Time Graph", are integrated in the framework, which can express given temporal scenarios as both narrative, diagrams and graphs. A multi-threading scanning algorithm based on the Depth-First Traversal algorithm is developed for automatic consistency checking.

The structure of such a framework can be expressed as Fig. 3.

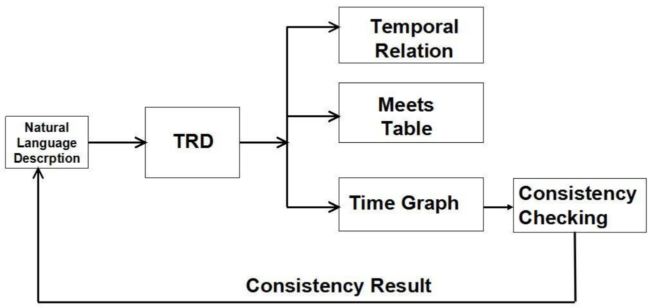

Fig. 3 Framework structure

Fig. 4 shows the environment/use-interface of Visual Time, where "File" button provides usual functions for file managements.

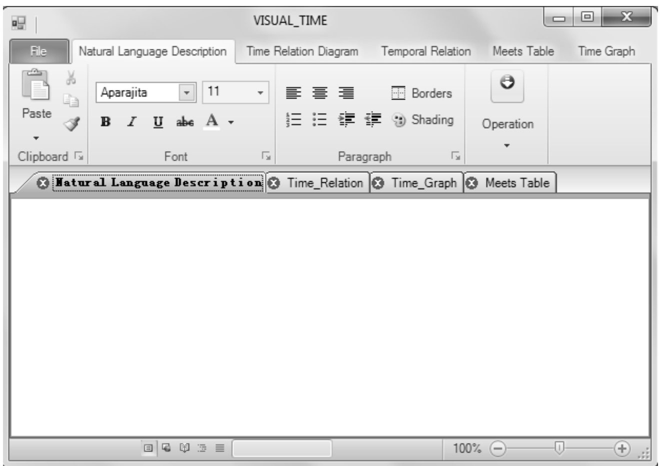

Fig. 4 Visual Time environment

In what follows, we demonstrate the main functions of Visual Time by considering the following scenario:

Two persons, Peter and Jack, are suspected of committing a murder during the daytime. In court, Jack and Peter gave the following statements, respectively:

- Peter's statements:

I got home with Jack before 1pm. We had our lunch, and when Jack left I put on a video. The video lasts 2 hours. Before it finished, Robert arrived. When the video finished we went to the train station and waited until Jack came at 4 pm.

- Jack's statements:

Peter and me went to his home and arrived there before 1pm. When we finished our lunch there, Peter put on a video, and I left and went to the supermarket. I stayed there for between 1 and 2 hours. Then I drove to my home to collect some mail. It takes between 1.5 to 2 hours to reach my home; and about the same to the train station. I arrived at the train station at 4 pm.

In addition, being a witness, Robert made the following statements:

- Robert's statements:

I left home at 2 pm and went to Peter's house. He was playing a video, and we waited till it finished. Then we went together to the train station and waited for Jack. Jack got to the train station at 4pm.

A. Natural Language Description View

In "Nature Language Description (NLD)" view, one can input these statements in the textbox (by means of direct typing in, Insert or Copy/Paste, etc.) as shown in as in Fig. 5.

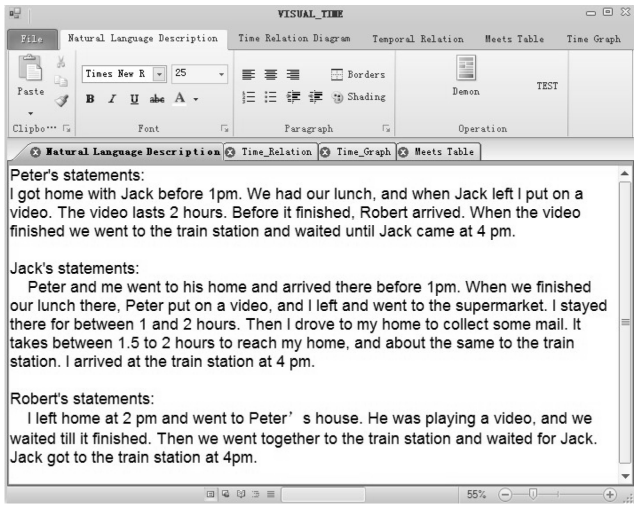

Fig. 5 Natural Language Description

B. Time Relation Diagram View

Similar to Peter Chan's Entity relationship Diagram (ERD), Time Relation Diagrams (TRD) are proposed here specially for modeling temporal information of given scenarios described in natural language, in which, time elements, duration andtemporal relations between time elements can be graphically represented as diagrams. TRD allows logical expressions of both absolute and relative temporal information, including both logical conjunctions and disjunctions.

In "Time Relations Diagram" view, any time element with its property details such as name, duration, and so on, can be added and modified, and any of the 13 possible temporal relations including "Meets", "Met_by", "Equal", "Before", "After", "Overlaps", "Overlapped_by", "Starts", "Starts_by", "During", "Contains", "Finishes" and "Finished_by", can be chosen to link any pair of given time elements.

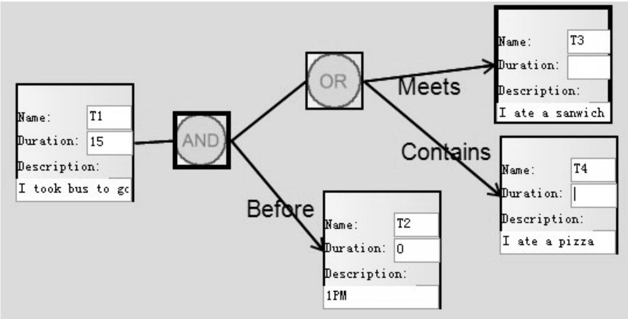

Fig. 6 A Sample TRD

Fig. 6 is a simple example of TRD, where there are 4 time elements, T1, T2, T3 and T4; the duration of (interval) T1 (interval) and (point) T2 are 15 and 0 respectively, and the duration of T3 and T4 are unknown (incomplete knowledge). The temporal relations are:

Before (T1, T2) $\wedge$ (Meets(T1, T3) $\vee$ Contains(T1, T4))where logical disjunction, "$\vee$", denotes the uncertainty of the temporal relation knowledge.

For the court scenario as described in Fig. 5 in the above, the corresponding TRD is as that shown in Fig. 7.

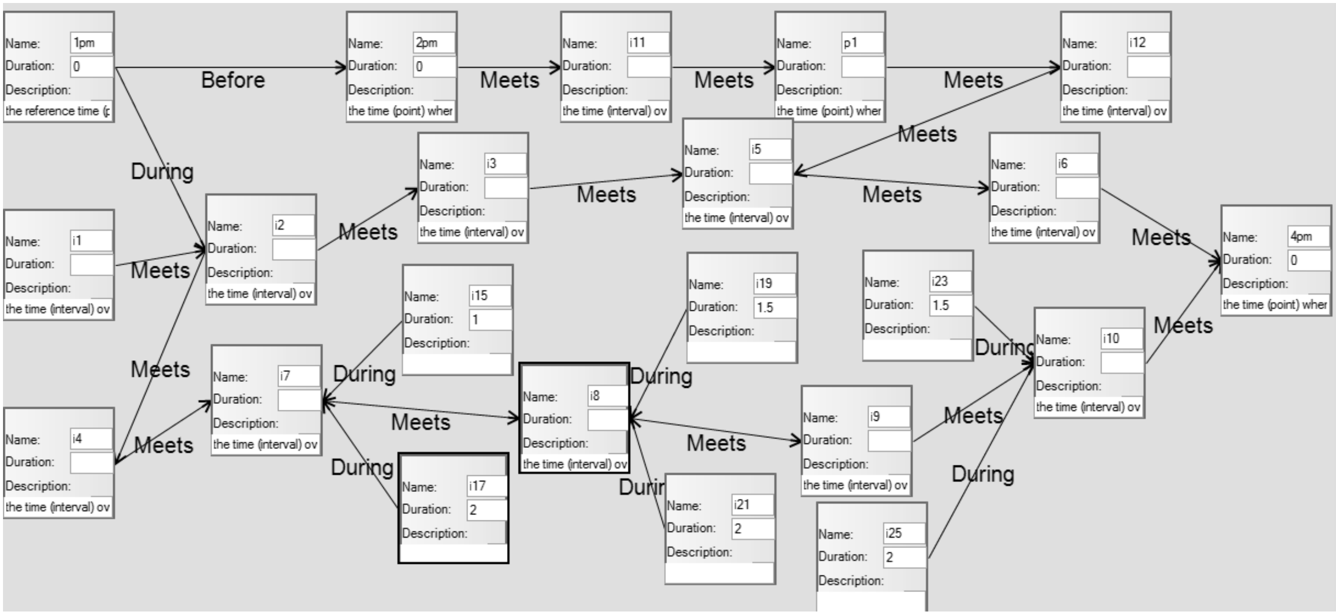

Fig. 7 Time Relations Diagram

For a given TRD, the algorithm described in Fig. 8 can automatically derive the corresponding temporal relations, and in turn transfer them into a single Meets Table and automatically draw out the corresponding Time Graph.

C. Temporal Relation View

In "Temporal Relation" view, the corresponding temporal relations between time elements as modeled in the TRD can be displayed automatically. Fig. 9 shows the temporal relations as for the court scenario considered here as the example.

D. Meets Table View

The Meets Table view, the temporal relations are transformed in terms of the Single meets relation. Fig. 10 displays the Meets Table corresponds to those temporal relations expressed in "Temporal Relation" view as shown in Fig. 8.

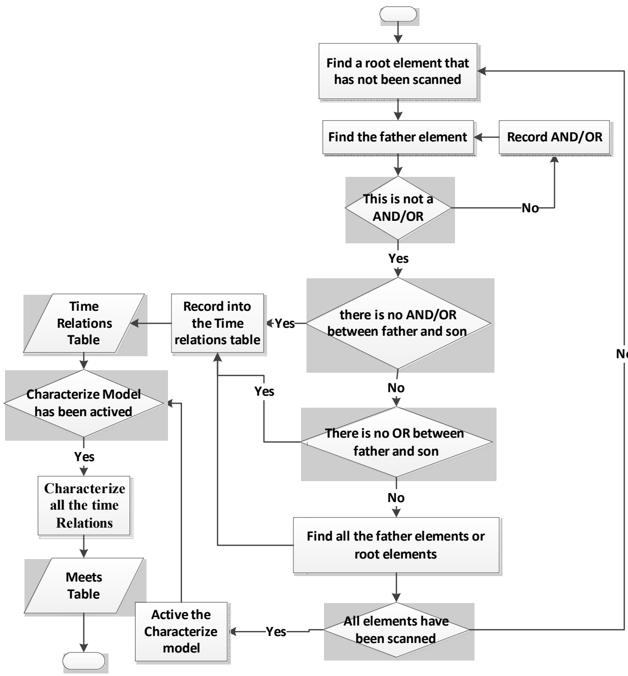

Fig. 8 Flow Chart of TRD Processing

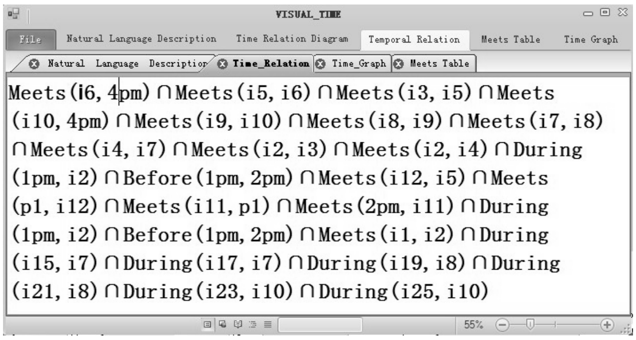

Fig. 9 Time Relations

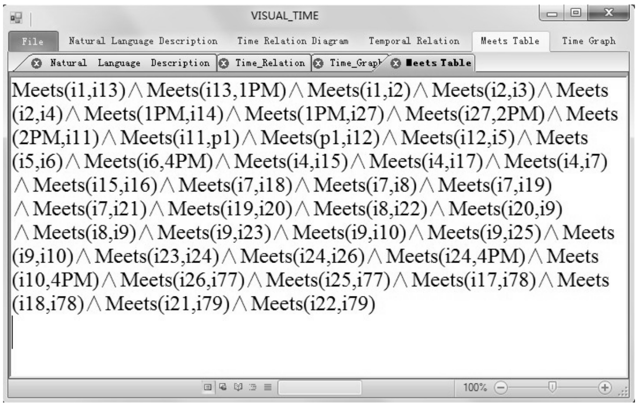

Fig. 10 Meets table

E. "Time Graph" View

For a given TRD, one can simply click the "Time Graph" button to automatically display the corresponding graphical representation in "Time Graph" view as shown Fig. 10. This graphical representation is exactly as that is defined in Section II. Fig. 11 presents the time graph of the court scenario.

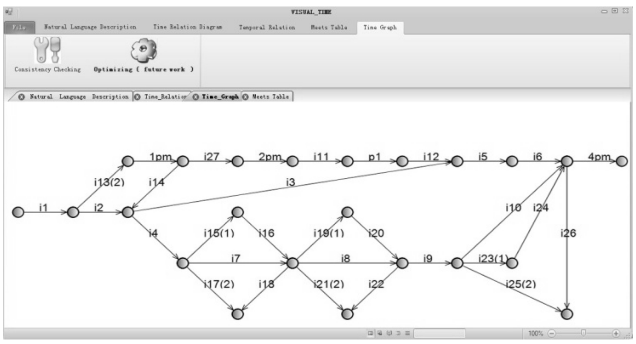

Fig. 11 The graphical representation

F. Consistency Checking

To check the consistency of a given temporal graph (scenario), one can simply click the "Consistency Checking" button in the "Time Graph" view, and the system will automatically deliver a verdict to confirm if the given graph, and therefore the corresponding scenarios, is temporally consistent or not. For the court example considered here, it confirms that it is inconsistent, and highlights the part that leads to the inconsistency.

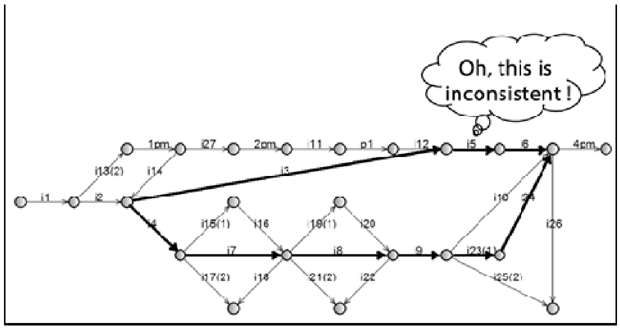

Fig. 12 Consistency Checking result

Hence, the collection of statements made by Jack, Peter and Robert is inconsistent; and therefore we can directly confirm that some statements are untrue. Suppose the video can be checked that it did actually last for two hours, we can confirm that there must be some falsity in either Robert's or Jack's statements. If it can be proved that Robert did left home at 2 pm, then Jack must have lied, when making his statements. Otherwise, to convince that his statements are true, Jack must prove that Robert left home at some time before 2 o'clock in the afternoon.

G. Links between NLD and TRD

Another function of Visual Time is that it provides an automatic link between phrases in "Natural Language Description" view and the corresponding time elements in "Time Relation Diagram" view, and vice versa. As for the court example, Fig.12 shows that the given scenario is inconsistent, and the time elements which make it inconsistent are i3, i5, i6, i4, i7, i8, i9, i23, i24. Fig.13 presents the links by automatically underlining the phrases corresponding to time elements i3, i5, i6, i4, i7, i8, i9, i23 and i24.

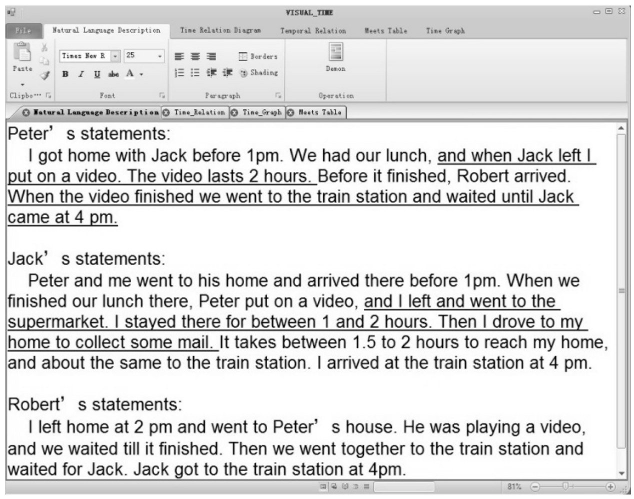

Fig. 13 Links between NLD and TRD

IV. Conclusion and Future Work

In this paper, we have presented a visualized framework, Visual Time, for representing and reasoning about uncertain and incomplete temporal Knowledge. The Temporal Relationship Diagram (TRD) for a given temporal scenario expressed in natural languages can be used as a logical base for a lot of applications, e.g., Business Process Modelling, etc. Temporal order relations can be automatically produced, and transformed into a "Meets" table, and based on which, the drawing of the corresponding time graph can be done automatically as well. The consistency checking is also automatic, providing both visual and audio verdict as for if a given temporal scenario is temporally consistent or not. This is an implementable system able to solve real-world (and full-scale) problems. It integrates interdisciplinary research activities including Temporal Logic, Natural Language Technology and Visual Modelling, etc.

There are some questions remain as for the future work. For instances: (1) how to automatically assign time references to temporal statements? (2) When a scenarios is inconstant, how to define and find the minimal effort to make it consistent? (3) For a consistent scenario, what's the minimal (or maximal) model to keep it consistent?

REFERENCES

[1] Tomasz Imielinski, WitoldLipski. Incomplete Information in Relational Databases.Journal of the ACM,1994, 31(4),pp. 761-791.
[2] Todd Mansell. A method for planning given uncertain and incomplete information, In Proceedings of the Ninth international conference on Uncertainty in artificial intelligence,1993, pp. 350-358.
[3] Mohamed Soliman, IhabIlyas, Shalev Ben-David. Supporting ranking queries on uncertain and incomplete data. The International Journal on Very Large Data Bases Archive. 2010, 19(4), pp. 477-501.
[4] Jianbing Ma, Weiru Liu and Paul Miller. Event modelling and reasoning with uncertain information for distributed sensor networks, In Proceedings of the 4th international conference on Scalable uncertainty management,2010, pp.236-249.
[5] James Allen. Maintaining knowledge about temporal intervals. Communications of the ACM, 1983, 26(11),pp. 832-843.
[6] James Allen. Towards a General Theory of Action and Time. Artificial Intelligence, 1984, 23, pp. 123-154.
[7] J. van Benthem, The Logic of Time. Kluwer Academic, Dordrech,1983.
[8] Brian Knight and Jixin Ma. Representing the Dividing Instant, the Computer Journal, 2003.46(2),pp.213-222.
[9] A. Galton. Critical Examination of Allen's Theory of Action and Time, Artificial Intelligence, 1990, 42,pp.159-188.
[10] J. Jensen, J. Clifford, S. Gadia, A. Segev and R. Snodgrass, A Glossary of Temporal Database Concepts. SIGMOD RECORD, 1992, 21(3),pp.35-43.
[11] James Allen and Pat Hayes, Moments and Points in an Interval-based Temporal-based Logic, Computational Intelligence, 1989, 5,pp. 225-238.
[12] J. McCarthy and Hayes. Some philosophical problems from the standpoint of artificial intelligence. In Meltzer, B. And Michie, D. (eds), Machine Intelligence, 4:463-502. Edinburgh University Press, 1969, Edinburgh.
[13] B. Bruce. A model for temporal references and application in a question answering program. Artificial Intelligence,1972,3, pp.1-25.
[14] Jixin Ma and Brian Knight, A General Temporal Theory. The Computer Journal, 1994, 37(2):114-123.