Representing Incomplete and Uncertain Temporal Knowledge

Jixin Ma

School of Computing and Mathematical Sciences

The University of Greenwich, London, U.K.

j.ma@gre.ac.uk

1. Introduction

- A Simple Question:

What is "*TIME*"?

- Am I asking a silly question?

What, then is time?

- **If no one asks me,**

  I know;

- **But if I want to explain it to a questioner,**

  I don't know.

Augustine of Hippo (*Confessions XI, XIV*)

Why time?

- Time plays an fundamental role in modelling natural phenomena and human activities concerning the dynamic aspects of the real world.
- Temporal reference is an idea deeply integrated in human common sense, as well as in the domain of computer and information science.
- Many computer based applications need to deal with the temporal dimension of information, the change of information over time and the knowledge about how it changes.

Areas requiring TR:

- Knowledge Representation and Management
- Prediction / Forecast / Planning
- Diagnosis / Explanation (Police, Law, Medical)
- Database Management / Data Mining
- Industrial Process Control
- Historical Reconstruction (Police, Law, Medical)
- Pattern Reconigiotn
- Natural Language Understanding

etc.

Typical feature:

- Complete and absolute temporal information is rarely available and remembered for knowledge representation and reasoning, while only incomplete relative temporal knowledge is derived from humans.

- This is in particular typical for Artificial Intelligence systems, in which temporal knowledge/Information is usually given in some incomplete and/or uncertain form.

Classification of Temporal Information

- ***Absolute temporal references*** **(easy to deal with):**

  E.g., “10 am on the 9th of October 2012” and “the last two weeks of September, 1988”, which refer to times with absolute values;

- ***Relative temporal references*** **(difficult to deal with):**

  E.g., “during the time when the Dean was in his office” and “after mid-night”, which refer to times that are known only by their relative temporal relations to other temporal reference, which again, may be absolute or relative;

- ***Absolute temporal durations*** **(easy to deal with):**

  E.g., “45 minutes” and “16 hours”, which refer to some certain amount of temporal granularity;

- ***Relative temporal durations*** **(difficult to deal with):**

  E.g., “less than 3 hours” and “more than six years but less than 10 years”, which refer to some uncertain amount of temporal granularity.

Two Folds of The Problem

- How to represent various kinds of incomplete and uncertain temporal knowledge?

- How to construct a reliable mechanism for inference, based on this representation?

2. Theories and Models (1)

Point-based theory

- **The most traditional structure of time is the standard point-based theory adopted in classical physics.**

- **Usually, a point-based theory consists of a pair (*P*, ≤), where**
  - ***P* is a set of points,**
  - **≤ is an order (partial or total) over *P*.**

2. Theories and Models (2)

Point-based theory (cont.)

For particular applications, the characteristics of a point-based system may be specified in great detail, e.g.:

- ***linear vs. non-linear*** **(including: *parallel*, *circular*, etc.)**
- ***dense vs. discrete***
- ***bounded vs. unbounded***

- **Some obvious models:**
  - **the *real-numbers time* (*R*, <)**
  - **the *rational-numbers time* (*Q*, <)**
  - **and the *integer-numbers time* (*Z*, <)**

2. Theories and Models (3)

Point-based theory (cont.)

- **In point-based systems, as the derived temporal objects, intervals are usually defined as set, or ordered pairs, of points. E.g.:**

$$Ip = \{\langle p_1, p_2 \rangle \mid p_1 < p_2\}$$

- **Relations over point-based intervals such as "Equal", "Before", "Meets", "Overlaps", "Starts", "During" and "After" (Bruce 1972), as well as the corresponding reverse relations, can be derived from the order relation over time points.**
- **The so-called Dividing Instant Problem**

2. Theories and Models (4)

Interval-based theory

- **An interval-based theory posits a pair (*I*,*R*) [Allen 1983]:**
  - ***I* is a set of intervals**
  - ***R* is a set of binary relations over *I* [All83,84]:**

    {Meets, Met_by, Equal, Before, After, Overlaps, Overlapped_by, Starts, Starts_by, During, Contains, Finishes, Finished_by}
- **The intuitive meaning of Meets(i_1, i_2) is that interval i_1 is one of the immediate predecessors (not necessarily the unique one) of interval i_2.**
- **By formally characterising the Meets relation as primitive, the other 12 binary relations can be derived.**

2. Theories and Models (5)

Interval-based theory (cont.)

- **Interval-based approach avoids the annoying question of whether or not a given point is part of, or a member of a given interval, which is equivalent to the *DIP*.**

- **However, it is not intuitive/convenient to express instantaneous events, e.g., "The court was adjourned at 4:30pm", "The heater is automatically switched on at 6:00am", and so on.**

2. Theories and Models (6)

Point&Interval-based theory

- **A point&interval-based theory [Ma 1994] consists of a triad (*T*,Meets,Dur), where**
  - ***T* is a non-empty set of time elements**
  - **Meets is a binary order relation over T**
  - **Dur is a function from T to R_0^+, the set of non-negative real numbers.**

- **A time element t is called an interval if Dur(t) > 0; otherwise, t is called a point.**

2. Theories and Models (7)

Point&Interval-based theory (cont.)

- **In terms of the single primitive relation Meets, other binary relations over points/intervals can be classified into 4 groups:**
  - **Point – Point:**

    {Equal, Before, After}
  - **Point – Interval:**

    {Before, After, Meets, Met_by, Starts, During, Finishes}
  - **Interval – Point:**

    {Before, After, Meets, Met_by, Started_by, Contains, Finished_by}
  - **Interval – Interval:**

    {Equal, Before, After, Meets, Met_by, Overlaps, Overlapped_by, Starts, Started_by, During, Contains, Finishes, Finished_by}

2. Theories and Models (8)

Relating / Relation	point t_1 to point t_2	Interval t_1 to interval t_2	point t_1 to interval t_2	interval t_1 to point t_2
Equal	t_1 t_2	t_1 t_2	Not Applicable	Not Applicable
Before	t_1 t_2	t_1 t_2	t_1 t_2	t_1 t_2
After	t_1 t_2	t_1 t_2	t_1 t_2	t_1 t_2
Meets	Not Applicable	t_1 t_2	t_1 t_2	t_1 t_2
Met-by	Not Applicable	t_1 t_2	t_1 t_2	t_1 t_2
Overlaps	Not Applicable	t_1 t_2	Not Applicable	Not Applicable
Overlapped-by	Not Applicable	t_1 t_2	Not Applicable	Not Applicable
Starts	Not Applicable	t_1 t_2	t_1 t_2	Not Applicable
Started-by	Not Applicable	t_1 t_2	Not Applicable	t_1 t_2
During	Not Applicable	t_1 t_2	t_1 t_2	Not Applicable
Contains	Not Applicable	t_1 t_2	Not Applicable	t_1 t_2
Finishes	Not Applicable	t_1 t_2	t_1 t_2	Not Applicable
Finished-by	Not Applicable	t_1 t_2	Not Applicable	t_1 t_2

2. Theories and Models (9)

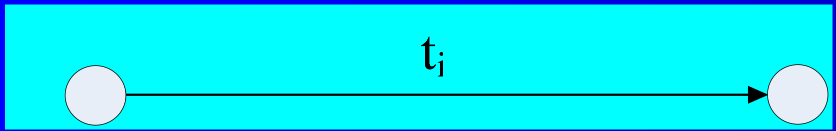
A triad (T, M, D) to express the temporal reference of a given collection of incomplete/uncertain temporal knowledge, where:

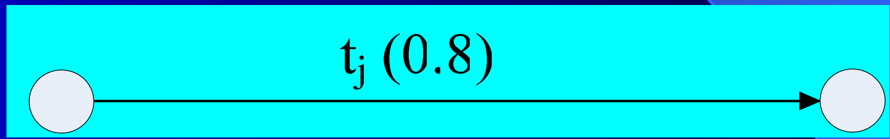
- $T = \{t_1, \ldots, t_n\}$ is a finite set of time elements, expressing the knowledge (possibly incomplete) of what time elements are involved;
- $M = \{Meets(t_i, t_{i(1)}) \vee \ldots \vee Meets(t_i, t_{i(j)}) \mid$ for some i, where $1 \leq i, i(1), i(j), j \leq n\}$ is a collection of disjunctions 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.
- $D = \{Dur(t_i) = r_i \mid$ for some i where $1 \leq i \leq n\}$ is a collection of duration assignments (possibly incomplete) to time elements in T.

3. A Graphical Representation (1)

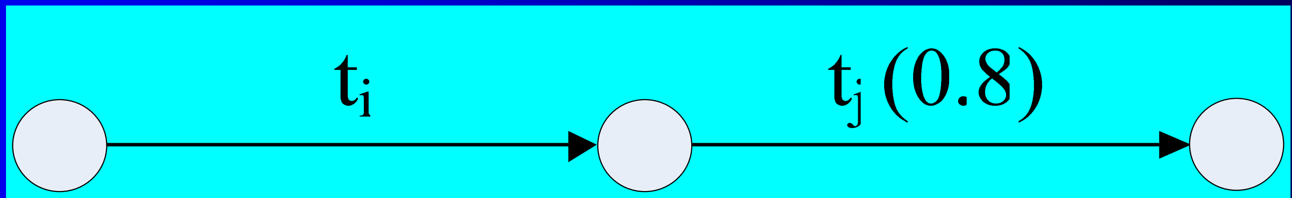
A temporal reference can be graphically expressed in terms of a directed, partially weighted simple graph G, called temporal graph, in which:

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

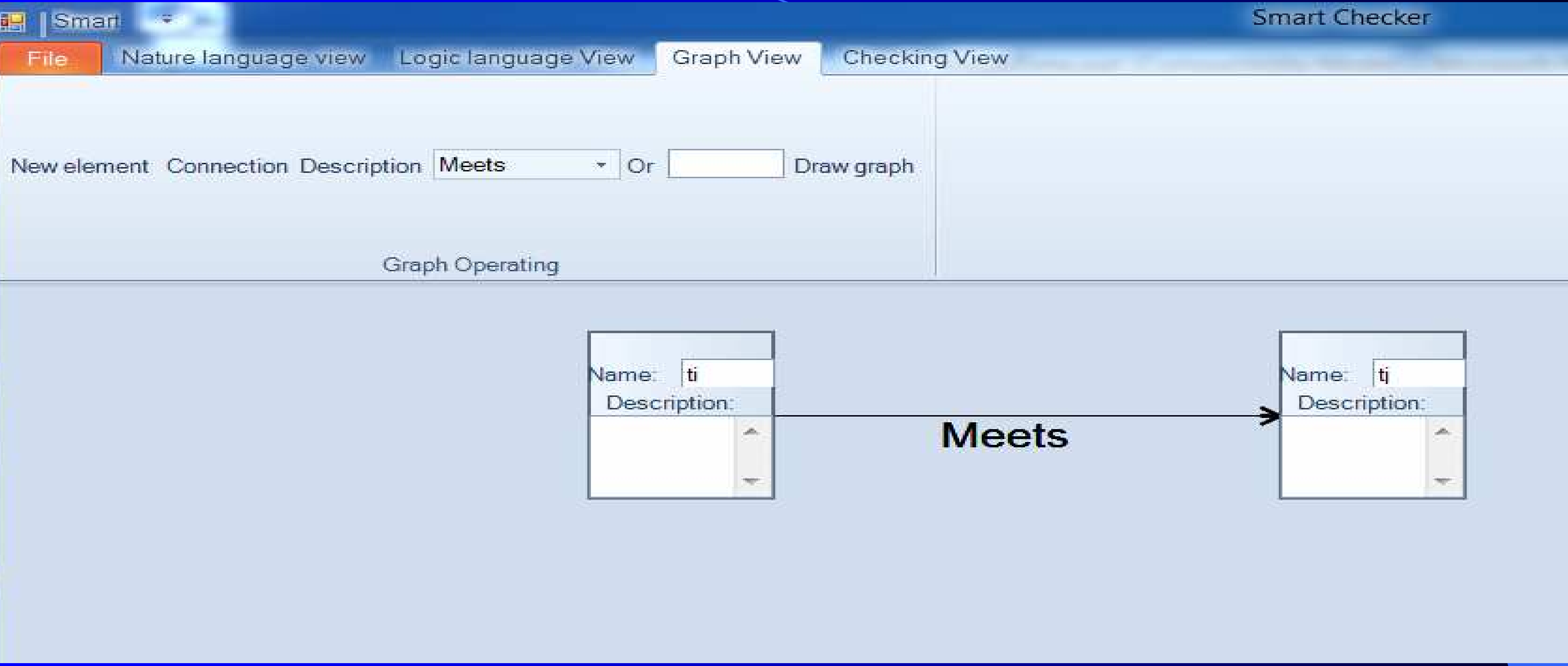


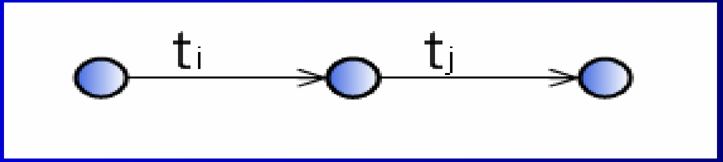


•Relation Meets(t_i, t_j) is presented by means of merging the ending-vertex of time element t_i and the beginning-vertex of time element t_j as the same vertex, of which t_i is an in-arc and t_j is an out-arc, respectively.



3. A Graphical Representation (2)



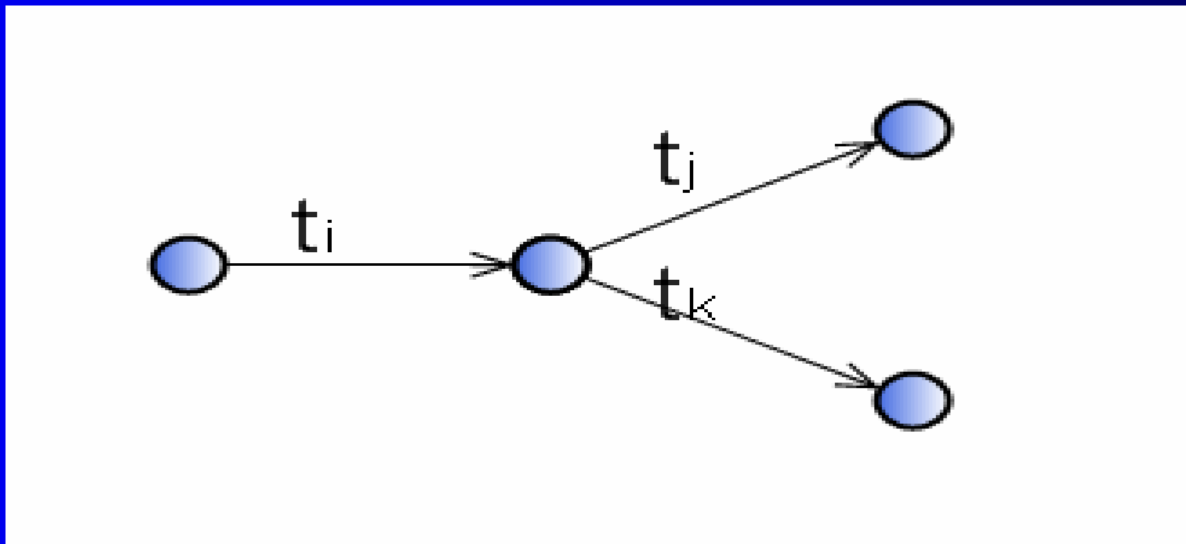


Demon1

3. A Graphical Representation (3)

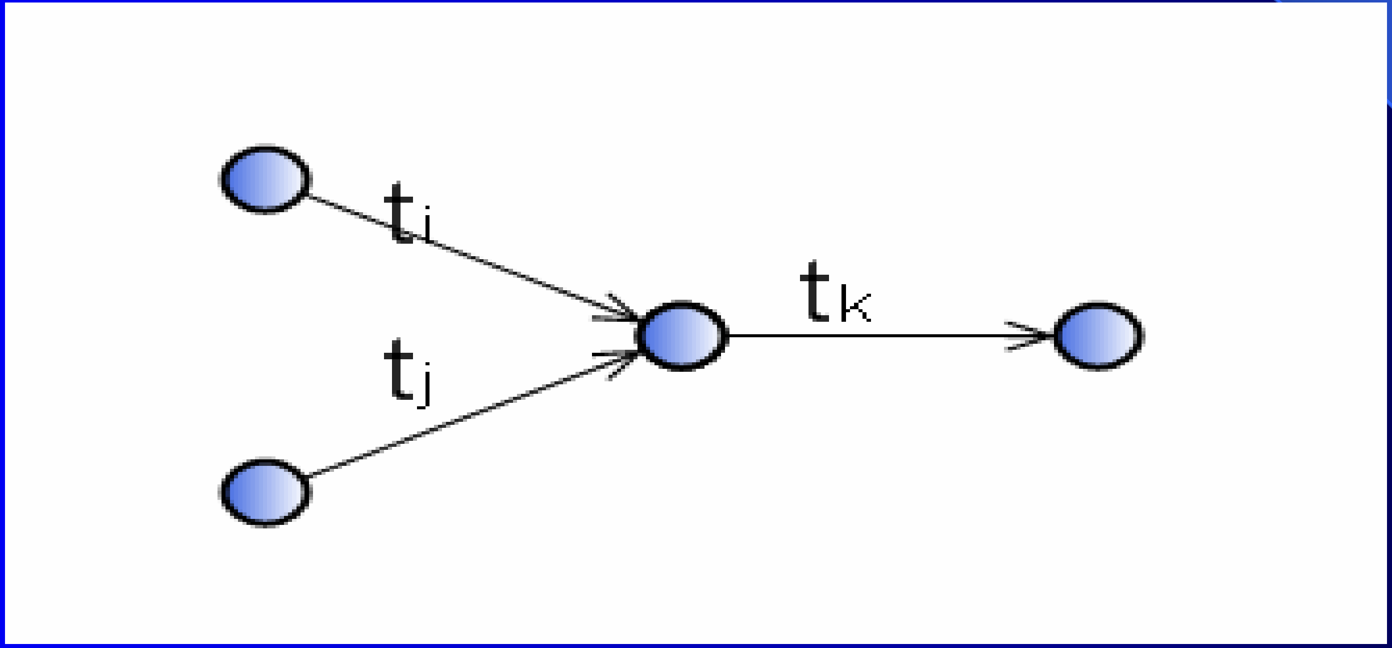
Logical expressions ("$\wedge$" and "$\vee$") of Meets relations are presented as below, respectively:

Meets(t_i, t_j) $\wedge$ 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 vertex, respectively.



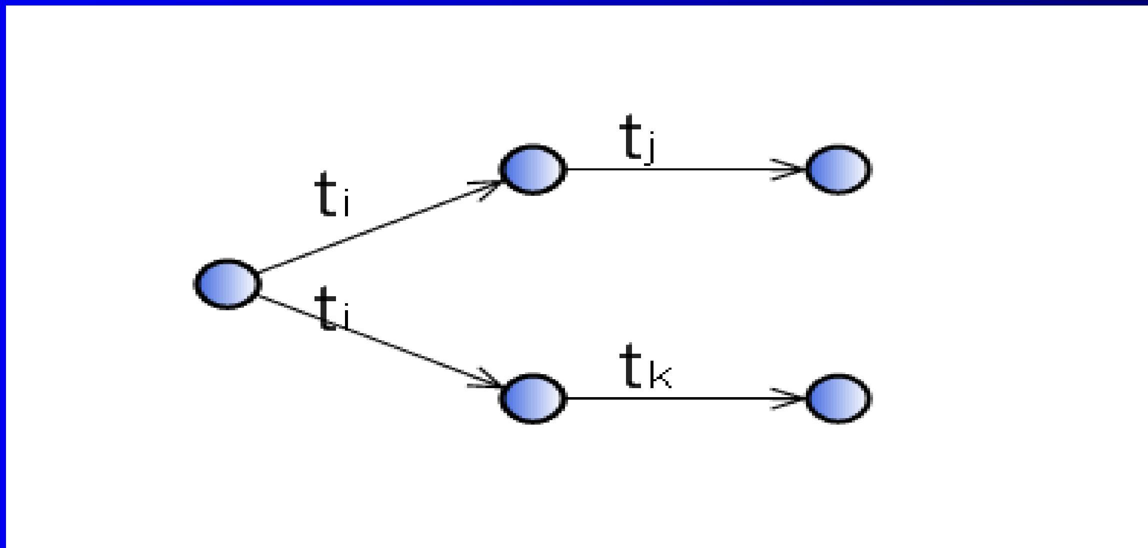
3. A Graphical Representation (4)

$Meets(t_i, t_k) \wedge Meets(t_j, t_k)$ is denote by defining t_i and t_j as two in-arcs and t_k as an out-arcs of the same vertex, respectively.



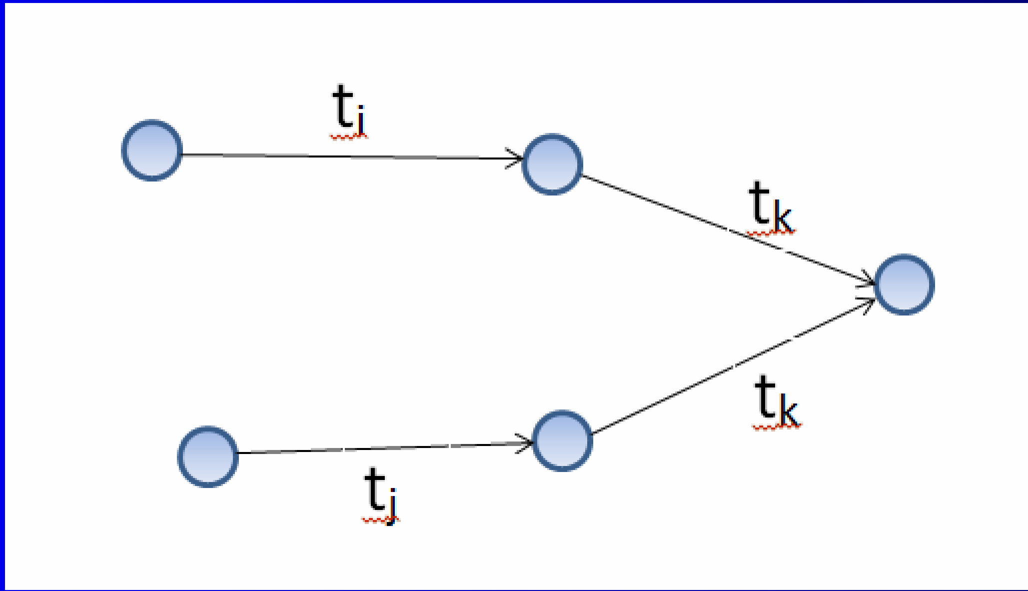
3. A Graphical Representation (5)

Meets(t_i, t_j) $\vee$ Meets(t_i, t_k) is denoted by defining t_i as duplicated identical out-arcs of the same vertex, and defining one of the two t_is as an in-arc and t_j as an out-arc of another vertex; and defining the other t_i as an in-arc and t_k as an out-arc of the third vertex respectively.



3. A Graphical Representation (6)

Meets(t_i, t_k) $\vee$ Meets(t_j, t_k) is denoted by defining t_k as duplicated identical in-arcs of the same vertex, and defining t_i as an in-arc and one the two t_ks as an out-arc of another vertex; and defining t_j as an in-arc and the other t_k as an out-arc of the third vertex respectively.

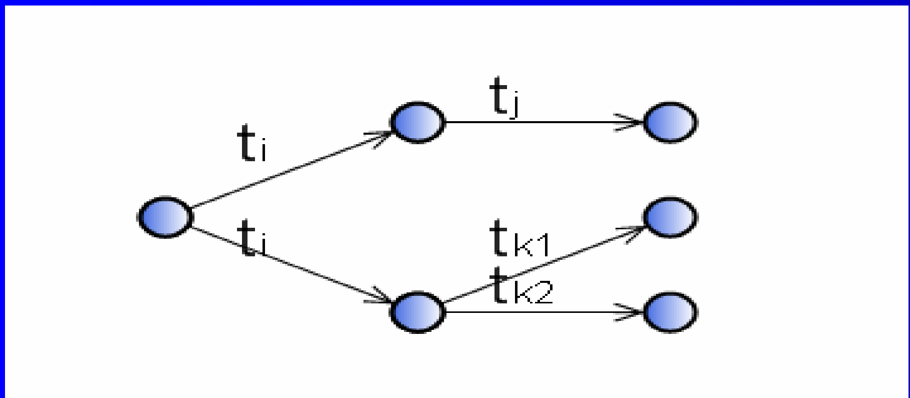


3. A Graphical Representation (7)

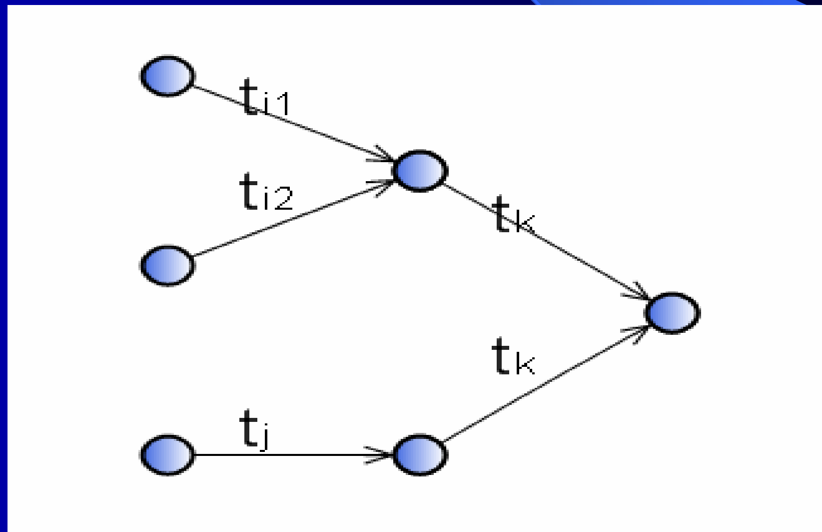
Two examples of expressing the logic combination:

first "AND" then "OR"

Meets(t_i, t_j)
$\vee$Meets(t_i, t_{k1}) $\wedge$ Meets(t_i, t_{k2})



Meets(t_j, t_k)
$\vee$ Meets(t_{i1} t_k) $\wedge$ Meets(t_{i2}, t_k)

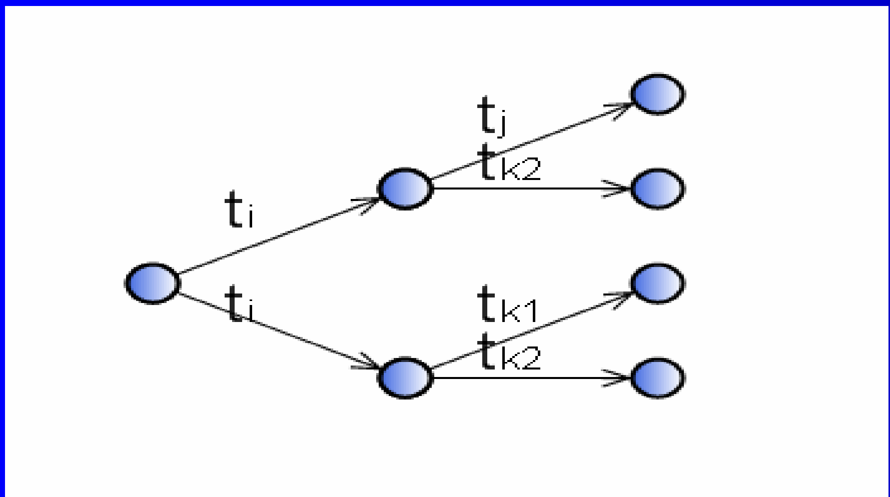


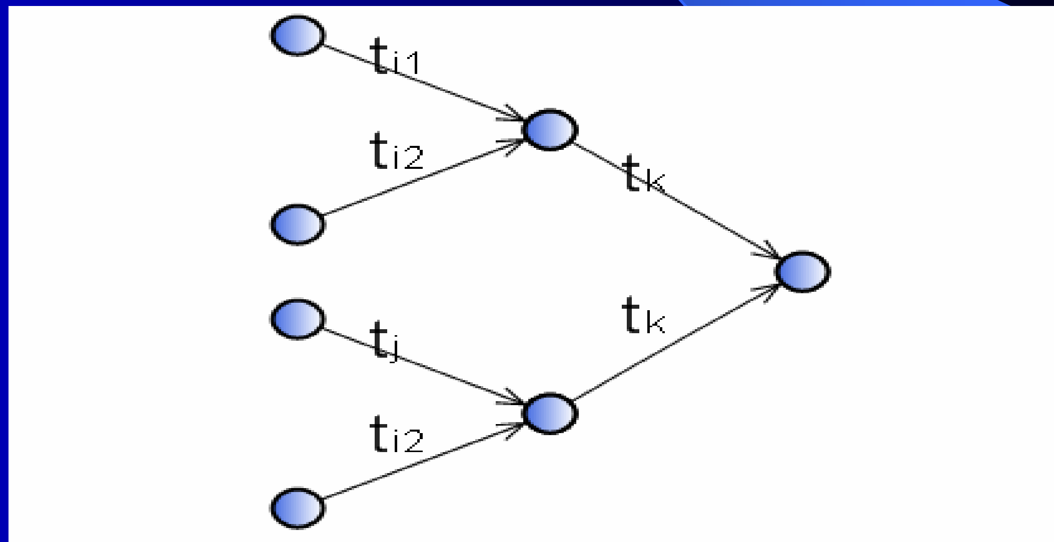
3. A Graphical Representation (8)

Two examples of expressing the logic combination: first "OR" then "AND".

$(\text{Meets}(t_i, t_j) \vee \text{Meets}(t_i, t_{k1})) \wedge \text{Meets}(t_i, t_{k2})$

$(\text{Meets}(t_j, t_k) \vee \text{Meets}(t_{i1}\ t_k)) \wedge \text{Meets}(t_{i2}, t_k)$





3. A Graphical Representation (9)

An illustration example, where comma "," standing for logical connective "$\wedge$":

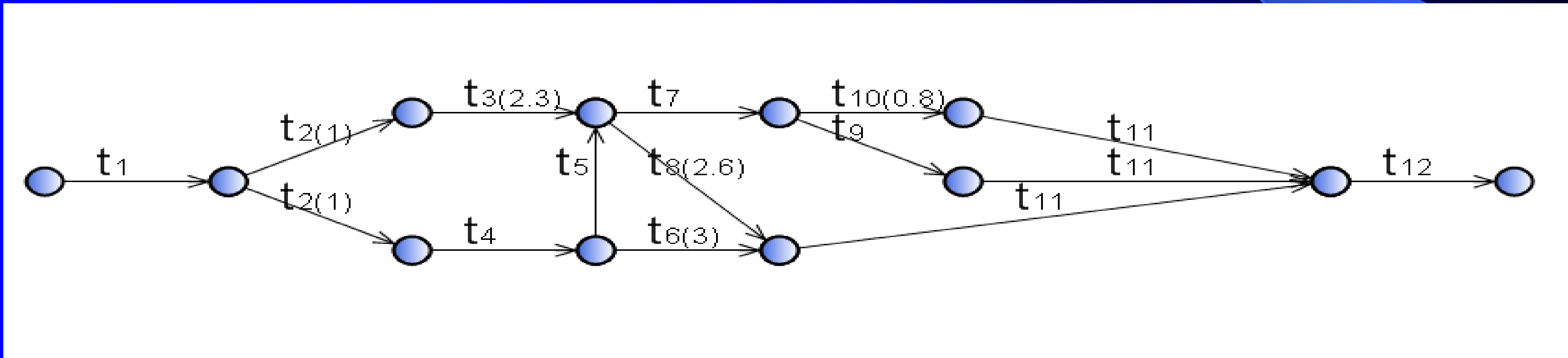
- **T_1** = $\{t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8, t_9, t_{10}\}$

- **M_1** = $\{Meets(t_1, t_2), (Meets(t_2, t_3) \vee Meets(t_2, t_4)), Meets(t_3, t_7), Meets(t_3, t_7), Meets(t_4, t_5), Meets(t_4, t_6), Meets(t_5, t_7), Meets(t_5, t_8), Meets(t_7, t_9), Meets(t_7, t_{10}), Meets(t_{11}, t_{12}), (Meets(t_8, t_{11}), Meets(t_6, t_{11}) \vee Meets(t_9, t_{11}) \vee Meets(t_{10}, t_{11}))\}$

- **D_1** = $\{Dur(t_2) = 1, Dur(t_3) = 2.3, Dur(t_6) = 3, Dur(t_8) = 2.6, Dur(t_{10}) = 0.8, Dur(t_{12}) = 1\}$

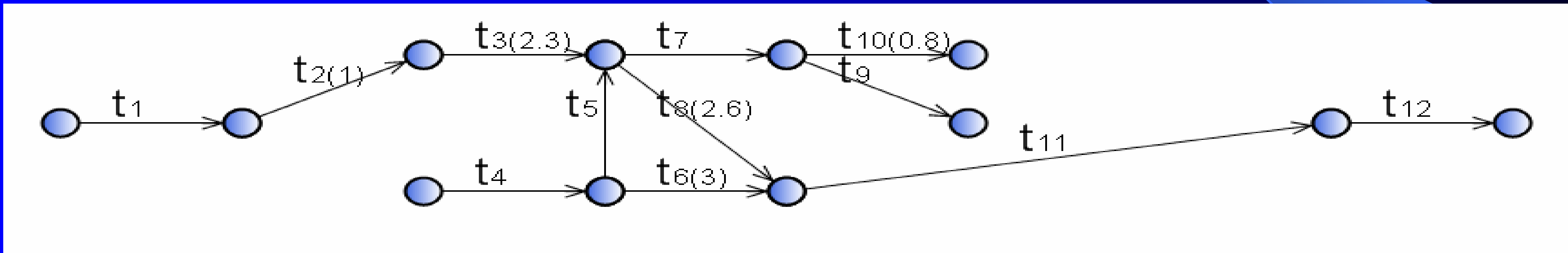
3. A Graphical Representation (10)

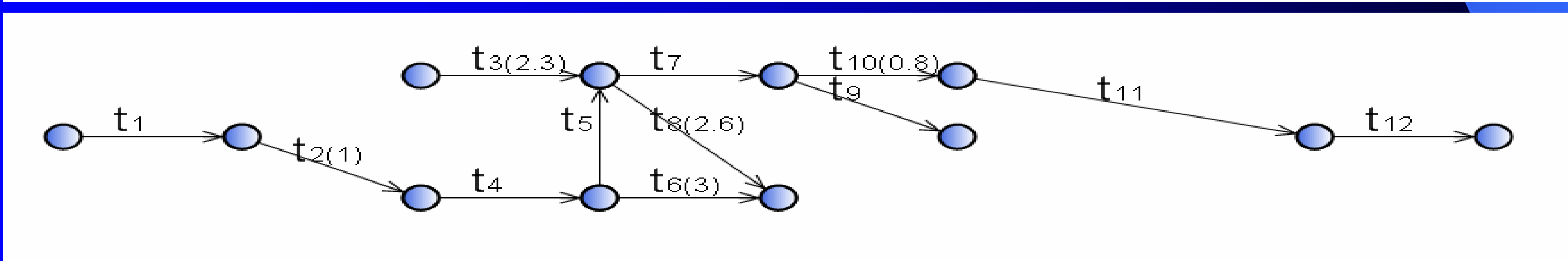
Demon2



3. A Graphical Representation (11)

For a given temporal graph G, a temporal scenario G^{ts} is defined as a maximal sub-graph of G with no duplicated time elements. For instances, below are 2 of the 6 (=2*3) temporal scenarios of temporal graph for ($\mathbf{T_1}$, $\mathbf{M_1}$, $\mathbf{D_1}$).





4. Temporal Consistency Checking (1)

A temporal reference

(T, M, D)

is defined as temporal consistent if at least one of its temporal scenarios is temporal consistent.

4. Temporal Consistency Checking (2)

The necessary and sufficient condition for the consistency of a scenario G^{ts} , can be given as below:

For each simple circuit in the graph of scenario G^{ts}, the directed sum of weights is zero;

For any two adjacent time elements, the directed sum of weights is bigger than zero.

Condition 1. guarantees that there exists a valid duration assignment function Dur to the time elements in scenario G^{ts} agreeing upon D;

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.

4. Temporal Consistency Checking (3)

- 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 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 reference ($\mathbf{T_1}$, $\mathbf{M_1}$, $\mathbf{D_1}$) is consistent since one of its temporal scenarios, e.g., temporal scenario 1 is consistent. In fact, by assigning duration value of 0.4 to t_4, it will make both temporal scenarios 1 and 2 consistent.

5. Case Study (1)

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.

5. Case Study (2)

In addition, being a witness, Robert made this 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.**

We can use the following temporal references for the corresponding statements in the above scenario:

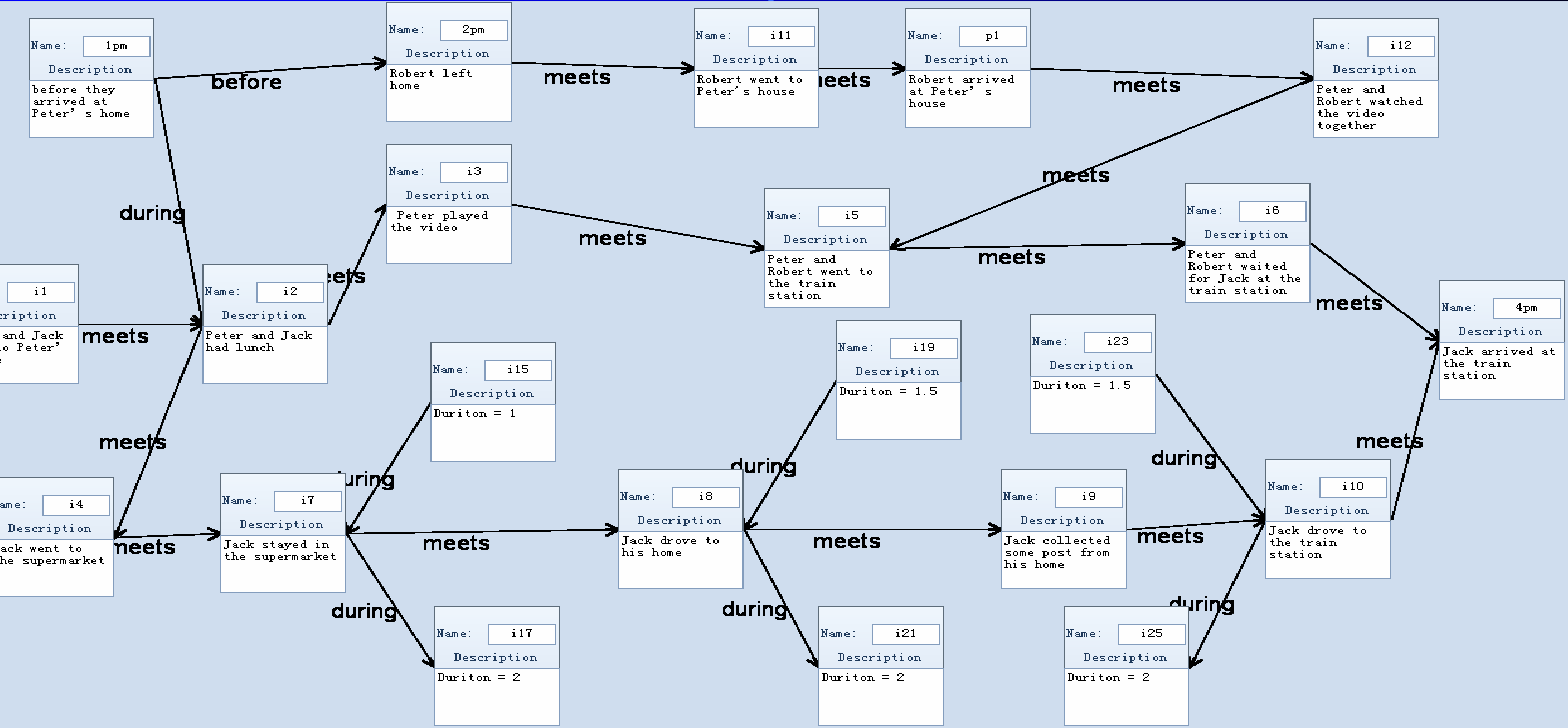
5. Case Study (3)

i_1: the time (interval) over which Peter and Jack went to Peter’s home;
1pm: the reference time (point) before which they arrived at Peter’s home;
i_2: the time (interval) over which Peter and Jack had lunch;
i_3: the time (interval) over which Peter played the video ($Dur(i_2) = 2$);
i_4: the time (interval) over which Jack went to the supermarket;
p_1: the time (point) when Robert arrived at Peter’s house;
i_5: the time (interval) over which Peter and Robert went to the train station;
i_6: the time (interval) over which Peter and Robert waited for Jack at the train station;
4pm: the time (point) when Jack arrived at the train station;
i_7: the time (interval) over which Jack stayed in the supermarket ($1 < Dur(i_7) < 2$);
i_8: the time (interval) over which Jack drove to his home ($1.5 < Dur(i_8) < 2$);
i_9: the time (interval) over which Jack collected some post from his home;
i_{10}: the time (interval) over which Jack drove to the train station ($1.5 < Dur(i_{10}) < 2$);
2pm: the time (point) when Robert left home;
i_{11}: the time (interval) over which Robert went to Peter's house;
i_{12}: the time (interval) over which Peter and Robert watched the video together;
i_{13}, ..., i_{27}: some extra relative time elements which are used for expressing the correspondingly relative duration knowledge, e.g., with i_{19}, i_{20}, i_{21}, i_{22}, and $Dur(i_{19}) = 1.5$ and $Dur(i_{21}) = 2$, we can get $1.5 < Dur(i_8) < 2$.

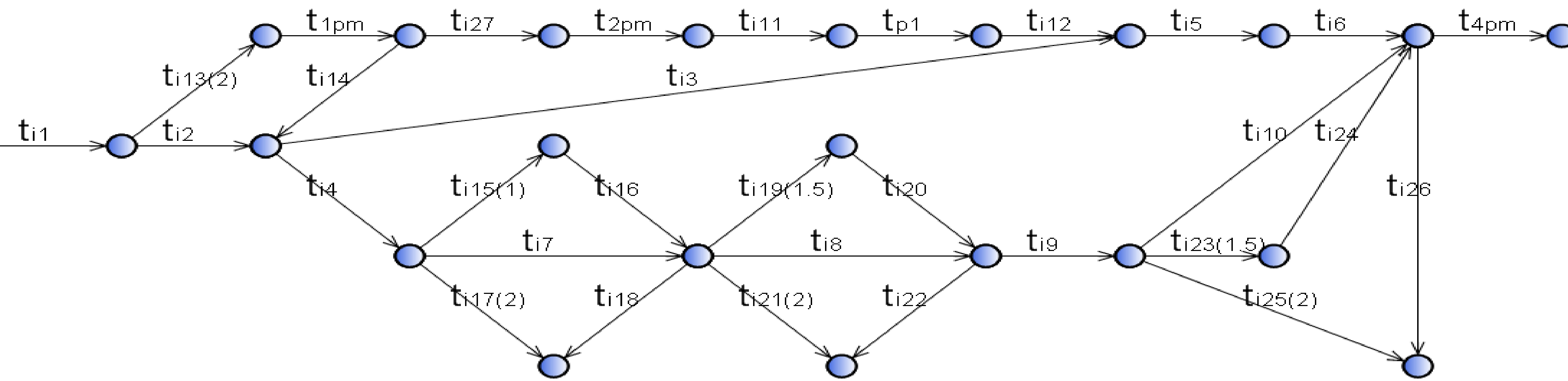
5. Case Study (4)

- i1 meets i2
- i2 meets i4
- 1pm during i2
- 1pm before 2pm
- 2pm meets i11
- i11 meets p1
- p1 meets i12
- i12 meets i5
- i2 meets i3
- i3 meets i5
- i5 meets i6
- i6 meets 4pm
- i4 meets i7
- i15 during i7
- i7 during i17
- i7 meets i8
- i19 during i8
- i8 during i 21
- i8 meets i9
- i9 meets i10
- i23 during i10
- i10 during i25
- i10 meets 4pm

5. Case Study (5)



5. Case Study (6)



5.Case Study (7)

Consider if the above temporal knowledge is consistent or inconsistent:

Since each interval has a positive duration and each point has a non-negative duration, we can easily see that: $\boldsymbol{Dur(i_5) + Dur(i_6) < 2}$

In addition, since $\boldsymbol{Dur(i_3) = 2}$**, hence:** $\boldsymbol{Dur(i_3) + Dur(i_5) + Dur(i_6) < 2 + 2 = 4}$

However,

$$\boldsymbol{Dur(i_4) + Dur(i_7) + Dur(i_8) + Dur(i_9) + Dur(i_{10}) > 0 + 1 + 1.5 + 0 + 1.5 = 4}$$

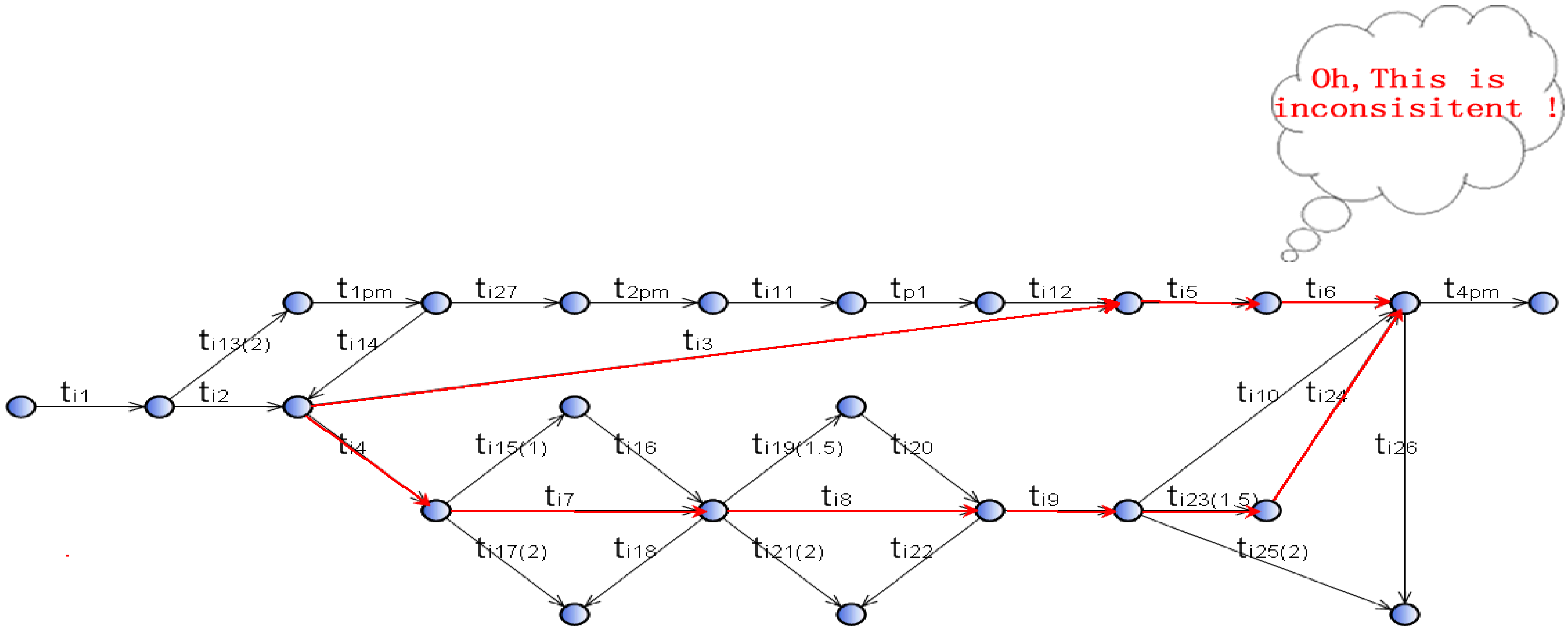
Therefore, for the simple circuit $\boldsymbol{i_3, i_5, i_6, i_{10}, i_9, i_8, i_7, i_4}$**, as shown in the following figure, there does not exist any possible duration assignment over the relevant time elements agreeing upon** ***Dur*****, such that**

$$\boldsymbol{Dur(i_3) + Dur(i_5) + Dur(i_6) = Dur(i_4) + Dur(i_7) + Dur(i_8) + Dur(i_9) + Dur(i_{10})}$$

that is,

$$\boldsymbol{Dur(i_3) + Dur(i_5) + Dur(i_6) - Dur(i_4) - Dur(i_7) - Dur(i_8) - Dur(i_9) - Dur(i_{10}) = 0}$$

5.Case Study (8)



5. Case Study (9)

Therefore, the temporal knowledge shown in the above is inconsistent, and hence some statements are untrue.

Suppose the video can be checked that it did actually last for two hours, then 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.

Software showing

Demon3

6. Conclusions

- **In terms of directed and partially weighted simple graph, a graphical representation for uncertain and incomplete temporal knowledge is proposed.**
- **It allows logical expressions of both absolute and relative temporal relations.**
- **Based on the graphical representation of a given collection of partial temporal knowledge, it can be checked if the corresponding temporal reference is temporally consistent or inconsistent, and derive the corresponding explanations →→(Future Work)**