Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Computer Science 16 (2013) 403 – 412

Procedia Computer Science

Conference on Systems Engineering Research (CSER'13)
Eds.: C.J.J. Paredis, C. Bishop, D. Bodner, Georgia Institute of Technology, Atlanta, GA, March 19-22, 2013.

Ontologies of Time and Time-based Reasoning for MBSE of Cyber-Physical Systems

Leonard Petnga* and Mark Austin**

**Ph.D. Graduate Student, Civil Systems Program, University of Maryland, College Park, MD 20742.*

***Associate Professor, Departments of Civil Engineering and Institute for Systems Research, University of Maryland, College Park, MD 20742.*

Abstract

Our work is concerned with the development of Model-Based Systems Engineering (MBSE) procedures for the behavior modeling and design of Cyber-Physical Systems. This class of problems is defined by a tight integration of software and physical processes, the need to satisfy stringent constraints on performance, safety and a reliance on automation for the management of system functionality. To assure correctness of functionality with respect to requirements, there is a strong need for methods of analysis that can describe system behavior in terms of time, intervals of time, and relationships among intervals of time. Accordingly, this paper discusses temporal semantics and their central role in the development of a new time-based reasoning framework in MBSE for CPS. Three independent but integrated modules compose the system: CPS, ontology and time-reasoning modules. This approach is shown to be mostly appropriate for CPS for which safety and performance are dependent on the correct time-based prediction of the future state of the system. A Python-based prototype implementation has been created to demonstrate the capabilities of the ontological framework and reasoning engine in simple CPS applications.

Keywords: ontology; temporal semantics; Model Based Systems Engineering; time-based reasoning; Cyber-Physical Systems.

1. Introduction

This paper examines temporal semantics and their use in reasoning when cyber-physical system behavior, safety, and performance are dependent on correct time-based predictions of future system state. The central role ontologies can play in capturing and formally representing the time domain is discussed. We review temporal theories and Description Logic (DL) semantics backing decidability of our time reasoning system. The latter makes uses of Allen's interval calculus algebra, and temporal relationships to support the reasoning about time and time intervals. We develop and propose use of a three-part CPS architecture that supports physical behavior modeling, the

* Corresponding author. Tel.: +1-301-405-9926; fax: +1-301-314-9218.
E-mail address: lpetnga@umd.edu.

1877-0509 © 2013 The Authors. Published by Elsevier B.V.
Selection and/or peer-review under responsibility of Georgia Institute of Technology
doi:10.1016/j.procs.2013.01.042

ontologies, and temporal reasoning. Our time-based reasoning framework is applied to a problem that involves behavior modeling and decision making for a vehicle approaching a yellow traffic light. We will see that the time-based reasoning approach improves the overall traffic system performance and safety.

2. Ontologies of time and temporal semantics in MBSE for CPS

2.1. Temporal theories

Time is an English word with various meanings; a concept that is frequently used in linguistic statements and is a central theme in different domains from philosophy to physics passing by knowledge representation and reasoning [1]. This diversity of views of what "time" is has led to various models of its representation as ontology experts try to define a time vocabulary with an explicit specification of intended meanings. Hayes [2] identifies six main concepts of time which, fortunately, are related to one another – among this set we select four to support ontological representation of time in engineering. They are:

1. *Time-interval:* pieces of time located on the temporal continuum (or time-plenum) serve as basic for the temporal theory
2. *Time-duration:* constant amount of time that can be compared and are distinctive from the length of the time interval are used to define time.
3. *Time-point:* the notion of a "point" in time supports this temporal theory; this concept is sometimes assimilated to the one of a "position in temporal coordinate system" which has no duration and is useful in locating event on the time-plenum.
4. *Time-dimension:* time is considered a physical dimension such as length, mass or voltage, with unit and physical properties as specified by Gruber and Olsen [3].

2.2. Ontologies of time and time models

Existing ontologies of time employ a combination of these four concepts, but are otherwise strongly influenced by the targeted need for which they were developed. For example, Hatala et al. [4] rely on a discrete time-space point model to store user paths in their real-time audio museum application. Gruninger [5] introduces time-point based axioms to support formalizing the Process Specification Language (PSL). The time in this theory is based on totally ordered timepoints and it supports branches for possible futures . In OWL-Time [6], the time ontology based upon the Web Ontology Language (OWL), *Instant* and *Interval* are basic mereological individuals, serving as foundational temporal entities (see Fig 1.).

```
:Instant
a       owl:Class ;
rdfs:subClassOf :TemporalEntity .
:Interval
a       owl:Class ;
rdfs:subClassOf :TemporalEntity .
:TemporalEntity
a       owl:Class ;
rdfs:subClassOf :TemporalThing ;
owl:equivalentClass
[ a       owl:Class ;
owl:unionOf (:Instant :Interval)] .
```

Fig. 1. Definition of instant, interval and temporal entity in OWL-Time

As standalone entities, time ontologies are not of great interest as they do nothing more than providing a description of the time domain. Getting the most out of them requires their integration to either other ontologies or directly in applications or models. Unfortunately, the pathway forward is not as simple as one would like. First, augmenting existing ontologies with time lead to diachronic representations where time is the third argument added to initial binary relations. Krieger [7] reviews three of the most relevant approaches and reinterprets a fourth one - the 4D view – using the notion of "time slice"[b] in an attempt to develop a general approach for equipping ontologies with time. Blom et al. [8] follow similar pathway by augmenting EASTADL2 and AUTOSAR models with timing information in specifying the semantics of age and reaction time appropriate for the automotive industry. In addition to expected increase in complexity of the RDF triple graph, the proliferation of objects of different types renders querying and reasoning with such approaches very painful. The proliferation of modelling languages also raises the issues of models consistency and interoperability. The OMG UML profile for modelling and analysis of real-time and embedded (MARTE) [9] systems not only tries to address this problem but it introduces a ``time-point'' model of time with abstractions and concepts (e.g., delay, duration, causal/temporal, clocked/synchronous) appropriate for the analysis of real-time systems behaviour.

2.3. Temporal semantics in CPS modeling

Why we need temporal semantics in modeling CPS? CPS are heterogeneous by nature, with time and concurrency intrinsic to the physical side but not necessarily relevant in the cyber world. Reconciling these differences makes CPS modeling, verification and validation particularly difficult. Also, as Derler indicates in [10], capturing and modeling *Zeno behavior*[c] in CPS models, whether they are desired or not is very hard without sound temporal semantics. Another challenge is the modeling of *distributed behaviors*, which requires the composition of components that are both spatially and temporally distributed. One illustration of the distributive nature of temporal components is the notion of time zone, which also applies to CPS. *Delays* are also intrinsic to CPS given their heterogeneity and distributed computational platforms. Eidson et al. [11] introduce PTIDES, a platform independent programming model for distributed systems based on time synchronization, which handles system delays but does not provide any reasoning capability.

Common temporal semantics for CPS. The recent development of the Lingua Franca Design and Integration Language (LFDIL) [12], a common language for CPS modeling, participates to the goal of unifying relevant semantics in view of formal and full across domains integration, including the temporal domain. Denotational semantics rely on mathematical terms to provide meaning to time constructs. However, the temporal semantics used for modeling is dependent on the theoretical viewpoint adopted, for instance - time-duration or time-interval. Wang et al. [13] put temporal semantics based on *Duration Calculus* into the Architecture Analysis and Design Language (AADL) to support timing behavior modeling for CPS. Allen et al.[14] rely on temporal interval semantics and define an *Interval Calculus* to specify relations between intervals based on the identification of begin and end points as well as *before* relations. Before diving into how this is done let's introduce the corresponding foundational logical theories.

Description Logics (DL) semantics. Horrocks [15] defines DL are a family of logic based knowledge representation formalisms that describe domain in terms of concepts (classes in OWL), roles (properties, relationships) and individuals (objects). Universal ($\forall$), existential ($\exists$), intersection ($\sqcap$), Union ($\sqcup$) and negation ($\neg$) operators are used for restriction specifications to make the language decidable with low complexity. In DL, semantics are defined by interpretations. An interpretation $\mathcal{I}$ is defined as followed:

$\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is the domain of interest (non-empty set) and $\cdot^{\mathcal{I}}$ is an interpretation function that maps:

- Concept name C : a subset $C^{\mathcal{I}}$ of $\Delta^{\mathcal{I}}$
- Role name R : a binary relation $R^{\mathcal{I}}$ over $\Delta^{\mathcal{I}}$

The interpretation function above extends to concept expressions as follows:

[b] This is a reinterpretation of the notion of "Time-interval" introduced earlier.

[c] Type of system behavior where infinitely many events occur in a finite time interval

Concepts/relations Expressions(DL)		Interpretation (FOL[d])
$(C \sqcap D)^{\mathcal{I}}$	$=$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
$(C \sqcup D)^{\mathcal{I}}$	$=$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$
$(\neg C)^{\mathcal{I}}$	$=$	$\Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$
$\{x\}^{\mathcal{I}}$	$=$	$\{x^{\mathcal{I}}\}$
$(\exists R.C)^{\mathcal{I}}$	$=$	$\{x \mid \exists y.\langle x, y\rangle \in R^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\}$
$(\forall R.C)^{\mathcal{I}}$	$=$	$\{x \mid \forall y.(x, y) \in R^{\mathcal{I}} \Rightarrow y \in C^{\mathcal{I}}\}$
$(\leqslant nR)^{\mathcal{I}}$	$=$	$\{x \mid \#\{y \mid \langle x, y\rangle \in R^{\mathcal{I}}\} \leqslant n\}$
$(\geqslant nR)^{\mathcal{I}}$	$=$	$\{x \mid \#\{y \mid \langle x, y\rangle \in R^{\mathcal{I}}\} \geqslant n\}$
$(R^{-})^{\mathcal{I}}$	$=$	$\{(x, y) \mid (y, x) \in R^{\mathcal{I}}\}$

Concepts, roles and individuals build up to the DL knowledge base $\mathcal{K}$ <$\mathcal{T}$, $\mathcal{A}$> of a domain D . Here, $\mathcal{T}$ is a set of terminological Tbox axioms and $\mathcal{A}$ is a set of assertional Abox axioms. x,y are individual names.

3. A time based reasoning framework for CPS

Our time-based reasoning framework makes use of Allen's interval calculus algebra and temporal relationships.

3.1. Justification of our system

For formal approaches to the verification of CPS to work, we need formal models to capture the appropriate granularity of time, considering its frequent under-specification in natural language expressions. For instance, consider the following expressions: (a) The last Olympics games started on July 27, and (b) In 2012, the Olympics games started in July. The absence of reference for a *year* for the chosen *month* (i.e., July) makes statement (a) underspecified. With the assumption that the *month* is the granularity of our time measurement, statement (b) is fully specified otherwise it could not be the case; for instance if, instead of *month*, the granularity of time is *day*. A number of researchers [16,17,18] have determined that interval-based models are more appropriate for formal analysis having time-dependent behavior than point-based models. In particular, interval-based temporal logics built over Allen's Interval Algebra [17] or Moszkowski's Interval Temporal Logic[18] appear as adequate choices to support practical temporal reasoning for the class of CPS systems with above-mentioned characteristics. However, it's critical for intervals to be fully specified using the granularity of time implemented in the cyber part of the CPS. In this work, we adopt interval calculus approach, which assumes that all intervals are "proper" with a *before* relationship between their beginning and end instants, which are fully specified time points. One of the key benefices of this specification is the avoidance of situations where instances of time intervals are underspecified; that is, either its *beginsAt* and/or *endsAt* properties have no values or the assigned values are underspecified as noticed by Krieger [7]. Moreover, this allows the formulation of restricted axioms which, when expressed in an ontology language, will ensure that time reasoning is decidable.

3.2. Allen's Temporal interval Calculus and OWL Time Ontology

The strength of Allen's interval algebra resides in its capability to manipulate interval and express temporal properties and their evolution over those intervals. At the core of this algebra is the relationship between time intervals. Thus, given two time intervals *I1* and *I2*, a time-point t and a proposition ϕ, we might ask a variety of questions over the time domain such as: (1) Mereological or "part-of" questions (e.g., Is the interval *I1* a sub-interval of *I2*? Does *t* occur within *I1*? Is the interval *I1* equals to *I2*? What interval represents the temporal

---

[d] First Order Logic

intersection of *I1* and *I2*? Does interval *I1* contains interval *I2*?); (2) Topological or "connects" questions (e.g., Does interval *I1* happens before or after interval *I2*? Do intervals *I1* and *I2* meet? Do intervals *I1* and *I2* start and/or end at the same instants?) and (3) Logical or "rules-based" questions (e.g., Does the proposition ϕ hold within the interval *I1*? If ϕ holds during the interval *I1* does it hold during *I2* too? Does the proposition ϕ hold before or after the interval *I1*?). Allen [17, 19] has identified and specified thirteen (13) relationships between any ordered pair of "convex" time intervals as the core of his Interval Algebra. The main seven (7) relationships are illustrated in Fig. 2 below; Six (6) inverse relations also exist.

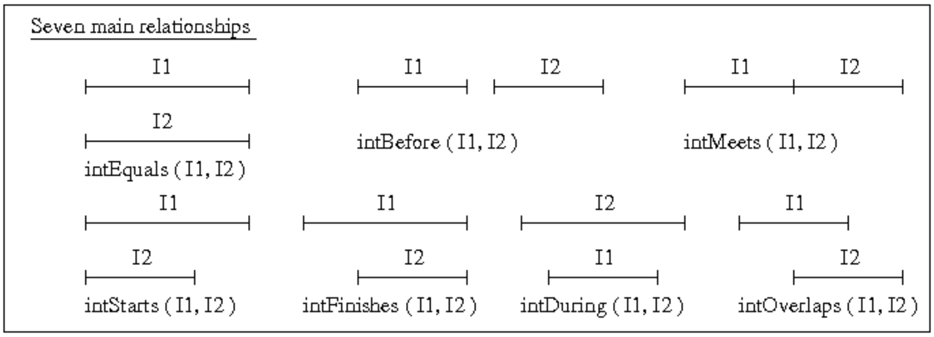

Fig. 2. Seven main relationships among intervals of time (as proposed by Allen).

Hobbs & Pan [20] provide formal definitions of these interval relations in terms of *before* relations among their beginning and end points, which exist, given their "proper" nature. Moreover, these intervals are closed i.e. if we consider intervals over a time domain $\mathcal{T}$ with $\{beginAt, endAt, t\} \subseteq \mathcal{T}$, then t is within the *closed* interval *[beginAt, endAt]* if *beginAt <= t* and *t <= endAt*. For instance, if I_1 and I_2 are two time intervals, the above *intOverlaps* relationship is defined as followed:

$$\begin{aligned} intOverlaps(I1,I2) \equiv & [ProperInterval(I1) \wedge ProperInterval(I2)] \\ & \wedge (\exists\ t2, t3)[ends(t2, I1) \wedge begins(t3, I2) \wedge before(t3, t2) \\ & \wedge (\forall\ t1)[begins(t1, I1) \Longrightarrow before(t1, t3)] \\ & \wedge (\forall\ t4)[ends(t4, I2) \Longrightarrow before(t2, t4)]]]] \end{aligned}$$

Clock, time zone and calendar definitions are added to this interval framework to serve as the foundation of the OWL time ontology for the semantic web published by World Wide Web Consortium(W3C)[6]. This time ontology is expressed in OWL DL which is a First Order Logic (FOL) restriction based on $\mathcal{SHOIN}$ DL. This DL is decidable thanks partially to well defined semantics and proven reasoning algorithms. Thus, OWL time ontology is an excellent candidate for our time-based reasoning framework for CPS.

3.3. System architecture and description

The system architecture for our time-based reasoning framework is shown on figure 3 below. It's comprised of three integrated modules:

Module 1: The Cyber-Physical System. This module is the actual CPS of interest, with each of its main subsystems having corresponding time model. The physical and cyber worlds communicate through an interface comprised of a network fabric between the various platforms, along with other physical interfaces. Physical time is the time at sensors, actuators and other network interfaces while the cyber time is the one of the computation. In order for the time-based reasoner to operate properly and allow for efficient decision making, there is a need for synchronization of the different clocks. One solution to this challenge, as introduced in [11], is to deliver events to the component of the system along with its corresponding time stamp.

Module 2: The Ontologies. The ontologies module formally describes the different physical and non-physical domains involved in the CPS modeling problem. The ontologies embed the abstraction of those domains and provide the intended meaning of concepts and statements in the domain. The appropriate mathematical framework is the DL formalisms introduced above. We are investigating use of the PHYSYS ontology [21] for capturing and representing the physical domain of the CPS. PHYSYS is a "generic" but formal ontology based upon system dynamic theory as perceived and used in engineering modeling, simulation and design. This makes it an excellent fit in MBSE approaches for CPS design. Moreover, it consists of three interdependent conceptual viewpoints or engineering ontologies formalizing structural (component), mathematical and process (dynamic) views on physical systems. Its process ontology specifies the behavioral view, using physical mechanisms that are applications of physical laws and principles to energy flows (in dimension energy/time). Analogies that exist between different physics such as mechanics, electricity, magnetism, hydraulic or thermodynamics are exploited along with system theory to produce a unified ontological view of physical systems using bond graph formalisms. In support of reasoning, there is a need to extract the dynamics of the physical system in term of time. This is the function performed by the *Converter* which, after the Bond Graph of the system is translated into differential equations and solved (process not shown), maps the behavior entities of the process ontology to time ontologies entities. The methodology introduced in [7] (see section 2.2) allows for the conversion of non-time entities into time entities and could be considered for this function too. Those time entities are time slices, which rejoined the abovementioned notion of time interval.

Module 3: Time-based Reasoning Engine. It receives inputs from the CPS and ontology modules and provides DL support for reasoning efficiently about time intervals. The Tbox of the DL knowledge base for our system contains "terminological" time axioms. They comprise the formal definition of concepts – temporal entities - as well as a set of roles that are mostly mereological and topological types of relations embedded in the structure of the time ontology. These axioms also provide *type* definition to temporal objects contained in the Abox which encompasses assertional axioms on the time domain. The reasoning system: (1) checks for (un)satisfiability of propositions constructed with the combination of Tbox and Abox elements in order to ensure consistency of the time interval knowledge base and, (2) infers new relations between input/existing time concepts (along with time objects) and the time domain or between themselves. Tableau algorithms can be used to test and check for consistency in the database and support the construction of a clash-free tree for input temporal concepts. Put together, those trees compose triple graphs of time concepts that can be queried. Triples contained in the graph are of the form *Subject-Predicate-Object*. Then, user defined inference rules further shrink the size of the graph thanks to predefined filters leading to a small (or empty) list of triples that serves as the basis for decision making. This process should be fully aligned with our CPS control strategy. In this framework, both *Subject* and *Objects* in triples are proper time intervals and *Predicates* are fully compatible with Allen's time interval specification as defined in Section 3.2. Our reasoning framework can be fully expressed using RDF[e] which is a basic relational language with simple ontological primitives. OWL also provides a powerful but still decidable ontology language to the same end.

4. Prototype Implementation: Behavior of a Car approaching a Yellow Traffic Light

4.1. Overview

To illustrate the use and effectiveness of the time-based reasoning framework, we consider the problem of decision making that occurs when a car approaches a yellow traffic light. For experimental purposes, two cases will be considered: (1) a human-driven car, and (2) an autonomous smart car. The driver's reaction time in case (1) increases the stopping distance while the smart car will have an almost-zero reaction time. Therefore, we seek to understand how this will translate to improved intersection throughput and enhanced traffic system safety.

This system is an "ideal" CPS in the sense that vehicles (physical system) interact with the light (cyber system) with the aim of maximizing traffic throughput while ensuring safe crossing of vehicles and pedestrians at the intersection. In actual real-world traffic light systems, timing of the yellow light for vehicles is generally set solely based on the expected throughput of incoming vehicles, not their physical characteristics. When the light turns yellow as the car

[e] Resource Description Framework

approaches the intersection, the driver must decide whether to stop the car or keep going. Therefore, as illustrated in Fig 4, not knowing the following three pieces of information can lead to performance and/or safety issues for this system: (1) duration θ_Y of the yellow light before it turns red; (2) stopping distance S of the vehicle given the actual speed, thus the corresponding duration θ_S; (3) duration θ_B it will take to arrive at the intersection given the actual speed and position of the vehicle.

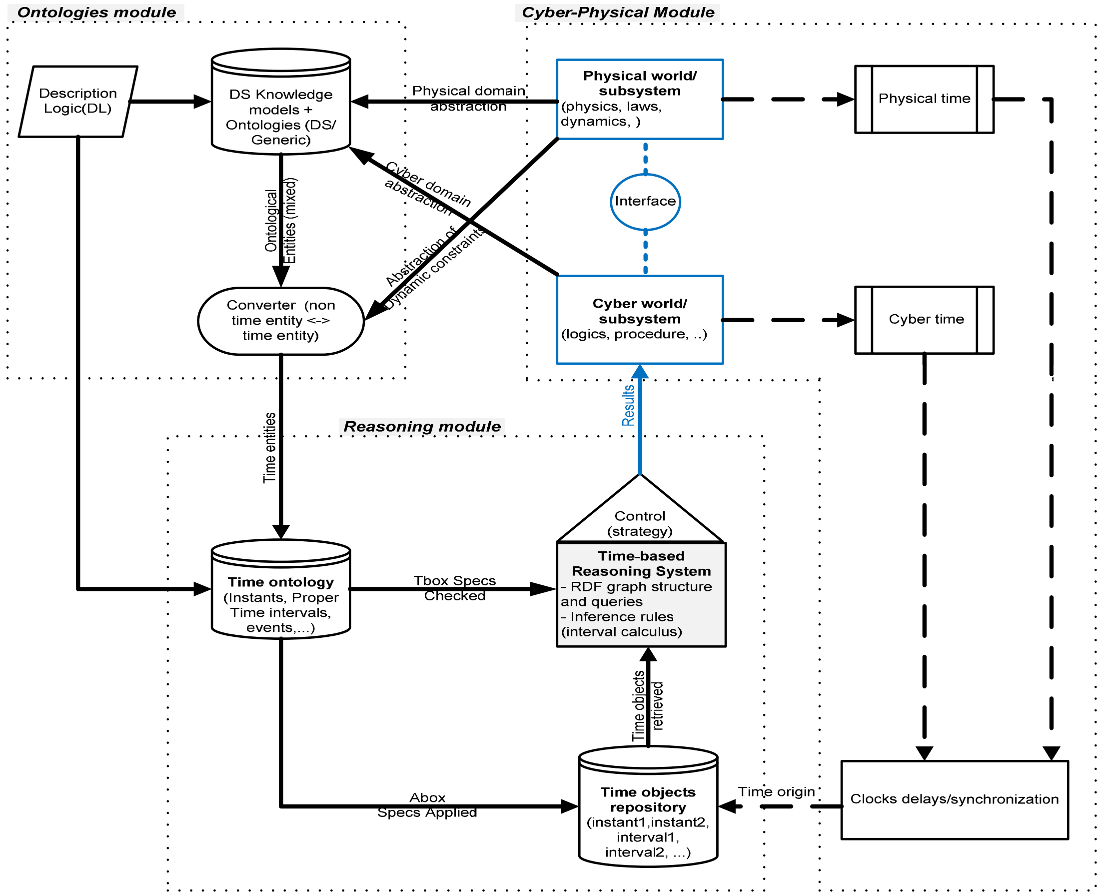

Fig. 3. High level architecture of the Time-based Reasoning System for CPS

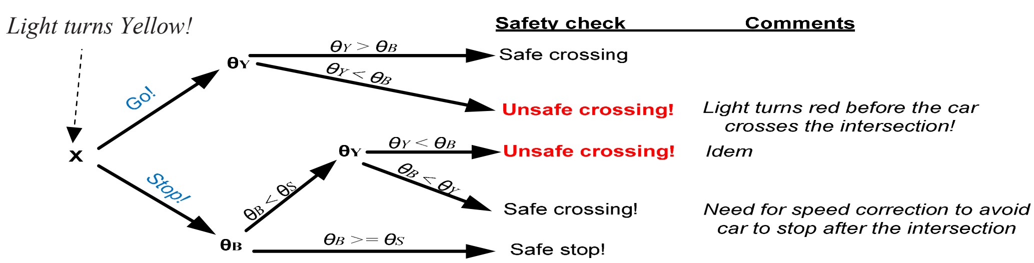

Fig. 4. Time-based decision tree of a human-driven car for the yellow light

4.2. Time-based reasoning system for an optimized traffic light: prototype

The decision tree above shows that none of the two decision pathways - assuming they are irreversible - is one hundred percent safe for the traffic light system! Moreover, even if the car crosses the intersection before the light turns red after failing to stop, there are still performance issues (overall reduced throughput) requiring correction.

Figure 5 shows the decision tree of a smart car whose behavior is consistent with the optimized traffic light system. Figure 6 introduces a prototype of the improved traffic light system implementing our time based reasoning framework. The system is augmented with workspaces for engineering, time ontology and cyber models implemented in *Python*, a leading object oriented scientific programming language. The three modules of the reasoning framework are implemented as followed:

Ontology Models: Given the simplicity of this model there is no need for the whole PHYSYS ontology here. However, a basic *space ontology* specifying position and distance concepts used by both Safety and Dynamic models of the vehicle is important. We note the central place of the *time ontology* whose semantics are used to properly specify light and car dynamics. An excerpt of its implementation in Python shows the definition of the time interval as specified by Allen's temporal algebra.

Engineering Models: In Fig. 6 we distinguish cyber models (yellow shading) from physical models (white shading). They model the behavior of the car as prescribed by both engineering specifications and physics of the vehicle and in accordance with the definitions provided by the supportive ontologies. Dynamic and safety models are interconnected since the *stopping distance* of the car is a quadratic function of its velocity [22], which is a linear function of time (assuming a constant acceleration). The light model is used to represent the logic (cyber part) of the traffic light based on the light and time ontologies. The timing and duration of the lights are specified based on the time ontology while the light ontology supports specifications such as sequence and colors (green, yellow and red). Put together, these models make up an implementation of the cyber-physical module of our framework.

Time-Based Reasoning Model: The time-based reasoning model makes use of the direct and real-time flow of time data from both the light (cyber module) and car dynamic (physics) to reason about the safety and performance of the system based on inference rules. The space-time conversion interface provides the position of the car in term of time instants based on the dynamic and safety models. Thus, the reasoner can project from the current instant (tX) the time intervals for both stopping the car (if the driver brakes) and arriving at the intersection (if the car keeps going). These time points are respectively the instants at which the car arrives at S or S' i.e. tS/tS' and B (tB) as shown on the simulation view of the reasoning model. Similarly, time interval for the Yellow (as well as others) light can be determined based on the set timing. The simulator takes charge of the synchronization of the clocks either at the instant the light turns yellow (tX) or at start time. The DL structure supporting time ontology along with its reasoning machinery (that we implemented in python) is called to (1) identify the nature of the light and its projected color when the car will arrive at B; (2) build the RDF graph for all possible triples involving the two time intervals. We then apply inference rules from six of the seven Allen's basic intervals[f] to the graph and extract the "list" of triples (which should contain the wining one) for the light. Finally, we walk through the list and identify the exact relationship between the two intervals.
This results in the determination of the appropriate course of action for the car. Thus, knowing those temporal information ahead leads to efficient decision making, better performance and improved safety for our system as shown by the new decision tree in Fig. 5.

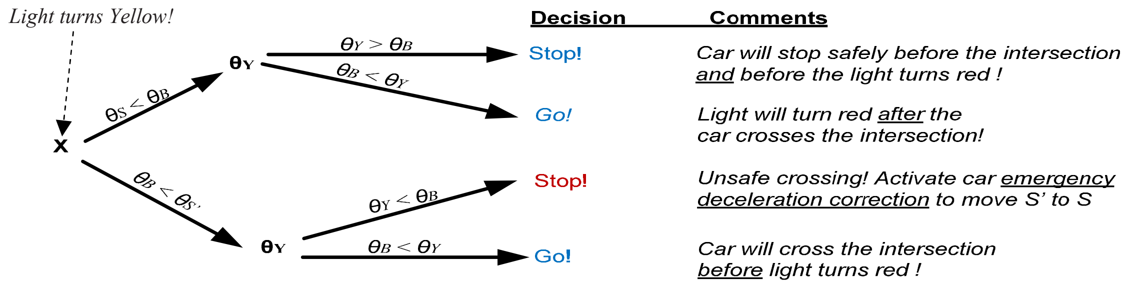

Fig. 5. Time-based decision tree of a smart car for the yellow light.

---

[f] Inference rule for the "Before" relationship is excluded since at this point, the two intervals share at least one time-point which is the instant the car will arrive at B.

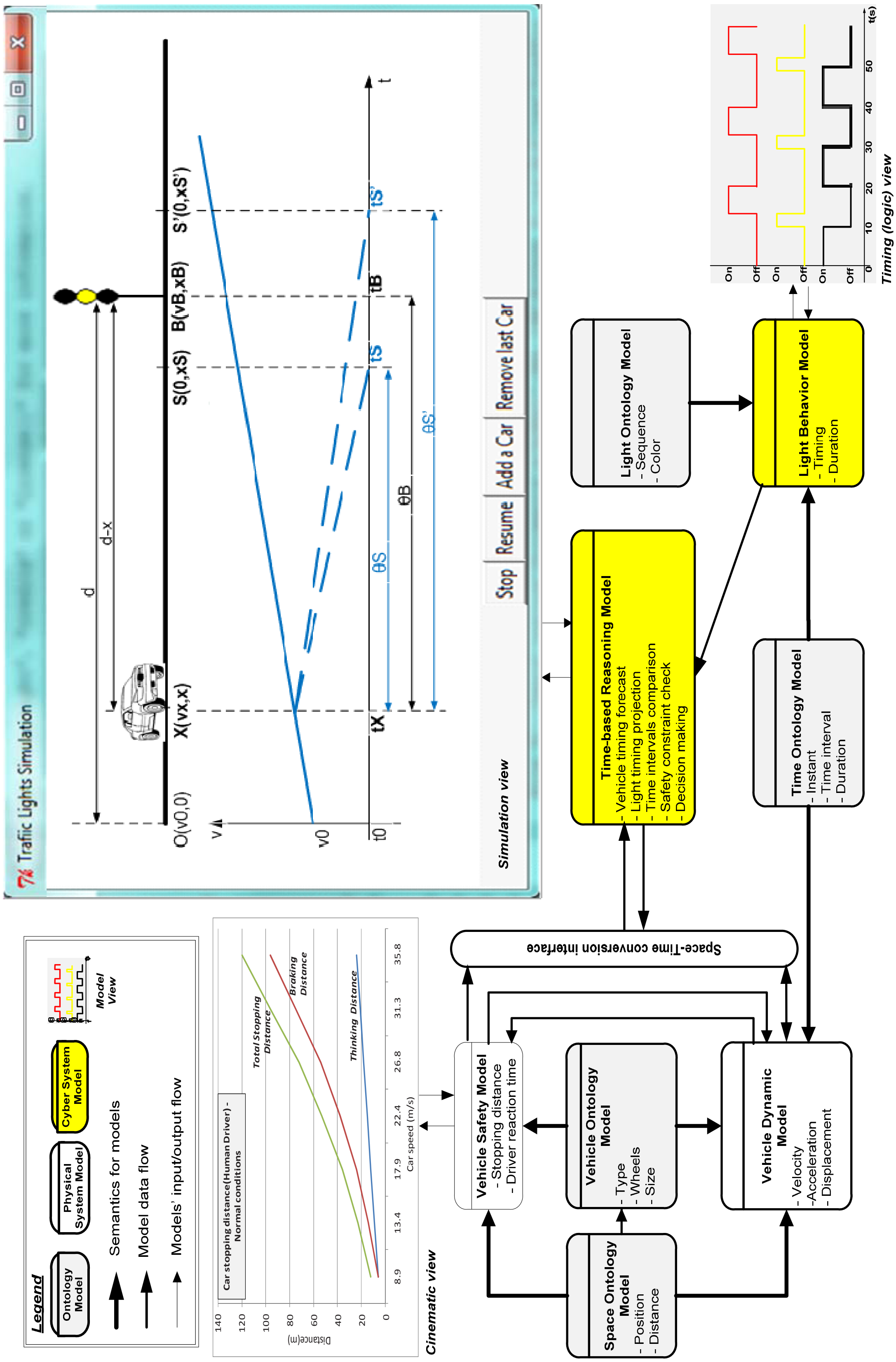

Fig 6: Schematic of the Traffic Light time-based reasoning system

5. Conclusion and future work

We have introduced a new time-based reasoning approach in MBSE for CPS. Our framework supports evaluation of CPS for which safety and performance are dependent on the correct time-based prediction of the future state of the system. Its capability goes far beyond the simple use of temporal semantics for answering mereological, topological and logical questions on the time domain. Mechanisms to reason on multi-time and handle non-time domains spanning CPS models are also provided. These results are been achieved thanks to Allen's interval calculus algebra and temporal relationships as well as DL semantics and system backing decidability of our modular time-based reasoning framework. A Python-based implementation of the framework is under development.

Acknowledgements

The work reported here is part of an NIST[g] funded program dedicated to the development of standards for CPS design, modeling, construction, verification and validation.

References

1. Laboratory for Applied Ontologies (2005), The Ontology of Time: Formal Ontology for Knowledge Representation and Natural Language, Processing – ICT School Doctorate Course, May 2005
2. Hayes(1994), A catalog of temporal theories, 1994
3. Gruber et al.(1994), An ontology for engineering mathematics. In Doyle, Torasso, & Sandewall, Eds., Fourth International Conference on Principles of Knowledge Representation and Reasoning , Morgan Kaufmann.
4. Hatala et al.(2005), Rules and ontologies in support of real-time ubiquitous application, Journal of Web Semantics (2005)
5. Grunninger (2003), Ontology of the Process Specification Language, in Handbook on Ontologies, Staab et al.(eds) (2004)
6. W3C(2006), Time Ontology in OWL, accessible at : http://www.w3.org/TR/owl-time/
7. Krieger(2009?), A General Methodology for equipping Ontologies With Time, German Research Center for Artificial Intelligence(DFKI)
8. Blom et al.(2009), Annotation with timing constraints in the context of EAST_ADL2 and AUTOSTAR – the Timing Augmented Description Language, Blom(2009)
9. OMG(2011), UML Profile for MARTE_Modeling and Analysis of Real Time Embedded Systems. V1.1 (OMG2011-06-02)
10. Derler(2012), Modeling Cyber-Physical Systems, Proceeding of the IEE| Vol. 100, No. 1, January 2012
11. Eidson(2010), A time-centric Model for Cyber-Physical Applications, MoDELS 2010 ACES-MB Workshop Proceedings, Oslo October 4, 2010
12. Masin et al.(2011), Meta II: Lingua Franca Design and Integration Language, IBM Research-Haifa Research Laboratory(August 2011)
13. Wang(2009), Modeling Timing Behavior for Cyber-Physical Systems, International Conference on Computational Intelligence and Software Engineering – CiSE(2009), 11-13 Dec 2009
14. Allen et al.(1997), Actions and events in interval temporal logic. In spatial and Temporal Reasoning. O. Stock, ed., kluwer, Dordrecht, Netherlands, 205-245
15. Horrocks (2005), Description Logic Reasoning, The University of Manchester (2005)
16. Fisher(2011), Practical Formal Methods Using Temporal Logic, 2011 John Wiley & Sons, Ltd
17. Allen J.F.(1983), Maintaining Knowledge about Temporal Intervals. Communications of the ACM, 26(11):832-843, 1983
18. Moszkowski et al.(1984), Reasoning in interval Temporal Logic and Tempura. In *Proc. AMC/NCF/ONR Workshop on Logics of Programs,* volume 164 of LNCS, pages 371 – 383. Springer, 1984
19. Allen J.F.(1984), *Towards a general theory of Action and Time*. Artificial Intelligence, 23(2):123-154, 1984
20. Hobbs et al. (2004), An ontology of time for the Semantic Web, ACM transactions on Asian Language processing(TALIP), 3(1):66-85
21. Borst et al.(1997), Engineering ontologies, Int. J. Human-Computer Studies(1997) 46, 365-406
22. Transport Research Laboratory, UK, 2007, Road Safety Authority, 2007 . Data accessible at www.drivingschoolireland.com/speed.html

[g] US National Institute of Science and Technology