Organon F 25 (1) 2018: 21-40

Analysis of Time References in Natural Language by Means of Transparent Intensional Logic

MARIE DUŽÍ – JAKUB MACEK[1]

ABSTRACT: In this paper, we deal with sentences containing time references like 'five years ago', 'three years older', 'in five seconds'. It turns out that such sentences are pragmatically incomplete, because there is an elliptic reference to a calendar that makes it possible to determine the length of the time interval associated with a time duration like a year, month, day, or to compute the time interval denoted by terms like 'February 29, 2016'. Since Transparent Intensional Logic (TIL) takes into account two modal parameters, namely possible worlds of type ω and times of type τ, and this system is particularly apt for the analysis of natural language expressions, our background theory is TIL. Within this system, we define time intervals, calendar time durations, and last but not least a method for adding and multiplying time durations in a way that takes into account the leap days and leap seconds. As sample applications, we analyse two sentences, to wit, "A year has 365 days" and "Adam is 5 years older than Bill".

KEYWORDS: Calendar – Gregorian calendar – Julian calendar – time duration – time interval – TIL – time point – time span – Transparent Intensional Logic – typed system – year.

---

1 Received: 9 May 2017 / Accepted: 27 June 2017

✉ Marie Duží

VSB-Technical University of Ostrava, Department of Computer Science FEI
17. listopadu 15/2172, 708 33 Ostrava, Czech Republic

e-mail: marie.duzi@vsb.cz

✉ Jakub Macek

VSB-Technical University of Ostrava, Department of Computer Science FEI
17. listopadu 15/2172, 708 33 Ostrava, Czech Republic

e-mail: jakub.macek.0@gmail.com

0. Introduction

Terms specifying time-referring objects like ‘five years ago’, ‘next month’, ‘for three days’ and sentences containing such terms are part and parcel of our everyday vernacular. The goal of this paper is to present a logical analysis of natural-language terms specifying *time durations* (‘year’, ‘month’, ‘day’, etc.) and their mutual relations in different contexts. As an example, we are going to analyse two sample sentences containing such terms:

“A year has 365 days.”

“Adam is five years older than Bill.”

We believe that these two sample sentences characterise well the issues connected with the analysis of such sentences containing time duration and time references. The first sentence might appear as an analytic one; yet it is not so, as we are going to show below. The second sentence illustrates an ordinary relation-in-intension between two individuals; yet the term ‘five years’ is vague, as we are going to show as well.

Yet, to the best of our knowledge, the analysis of such terms and sentences has been rather neglected by logicians as well as philosophers of language. There are several temporal logics that deal with sentences in the present, past and future tenses. These formal systems are mostly viewed as a special case of modal logic interpreted by means of Kripkean possible-world semantics. The term temporal logic is broadly used to cover all approaches to the representation of the temporal dimension within a logical framework. More narrowly, it is also used to refer to a particular modal system of temporal propositional logic that Arthur Prior introduced in Prior (1957; 1962; and 1967) under the name ‘tense logic’. Despite the great applicability of particular variants of tense logic in the semantics of programming languages, the systems just mentioned suffer a drawback when applied to the semantics of natural language. The drawback is their inability to adequately analyse sentences indicating a point of reference referring to the interval when the sentence was or will be true. Such sentences come attached with a presupposition under which a sentence is true or false.

This issue has been properly analysed in TIL that is an expressive logic apt for the analysis of sentences with presuppositions, because in TIL we work with partial functions, in particular with propositions with truth-value gaps (see Tichý 1980; and also Duží 2010).

In computer science, rigorous analysis of terms specifying time-referring objects is crucial. For instance, Ohlbach (1998) presents *Calendar Logic*, a propositional temporal logic whose operators quantify over time intervals that are specified using the terms of common vernacular, such as 'next week' and 'June 2000'. *Calendar Logic* uses two modal operators, 'sometimes within *T*' and 'always within *T*' where *T* may be one of the (finite) time intervals. This in effect allows *Calendar Logic* to retain the decidability of propositional logic, albeit at the expense of expressivity. A system for refinement of time intervals that captures the complexity of the Gregorian calendar is presented. For example, the following formula specifies the time interval denoted by the term '29th day of February 1998' where the initial interval of 1998 is assigned to the variable x_{year} and further intervals are specified using the functions *February* and *day_within_month*:

$$[1998; year] : day\_within\_month(February(x_{year}); 29)$$

The decidability of the system is the primary focus, and individual time intervals are denoted by the formal language constructs that partition the timeline into a finite number of continuous intervals. For example, consider these two sentences:

"If the temperature was below 0°C on February 20th, 2000, it snowed on February 20th, 2000."

"The temperature was below 0°C the entire February of 2000."

There are three continuous intervals: February 1st, 2000 to February 19th, 2000 (I1), February 20th, 2000 (I2), and February 21st, 2000 to February 28th, 2000 (I3). The final (pure) propositional logic formula for the first sentence is "$T_{I2} \rightarrow S_{I2}$" and the formula for the second sentence is "$T_{I1} \wedge T_{I2} \wedge T_{I3}$".

In Ohlbach & Gabbay (2004), the approach is extended to fuzzy time intervals, and the notion of time duration within this system is defined.

Hobbs & Pan (2004) proposes a similar system with the intention to represent time-based statements within OWL ontologies.[2]

The approach handles the property of leap days, but leap seconds are intentionally left out. To facilitate this, several predicates are defined, but the notion of duration is represented using predefined constants such as **Day**. For example, the following formula states that if *m* is an interval that is the month of February of the interval *y* and the interval *y* is a leap year, then *m* has 29 days:[3]

$$February(m, y) \wedge leapYear(y) \rightarrow Hath(29, {*}Day{*}, m)$$

For comparison, we propose an approach that applies Tichý's Transparent Intensional Logic (TIL) with procedural semantics based on ramified type hierarchy. In TIL we furnish the three different time objects (points, intervals, and durations) with types within a ramified hierarchy of types. In other words, we analyse the natural-language terms denoting them in a fine-grained way as any other terms of natural language. This applies, inter alia, to the objects of calendars (e.g. the *Gregorian calendar*) as well, and allows us to render the meaning of sentences like "Any year in the Gregorian calendar is longer than the same year in the Julian calendar." For instance, the year 2017 in the Gregorian calendar is longer than the year 2017 in the Julian calendar.

The rest of this paper is organized as follows. In Section 1 we introduce a fragment of TIL that we need for the analysis of time references. Section 2 deals with two basic terms that are used in time modelling, namely 'time point' (referring to a single point in time) and 'time interval' (standing for, e.g., the year 2017). In TIL, time is modelled as a set of real numbers, and these two terms are defined accordingly; *time point* as a real number and *time interval* as an interval of real numbers. Section 3 presents the main *novel contribution*, which are the definitions of *time duration* and *calendar time duration.* The fact that various years have different lengths is taken into account as well as the different notions of a *year* according to the Gregorian and the Julian *calendar*. Here we also

---

2 The Web Ontology Language (OWL) is a family of knowledge representation languages for authoring ontologies.

3 Citation is exactly as in the paper.

deal with the problem of leap days and leap seconds. Section 4 presents an analysis of the sentence “A year has 365 days”. Here we put forward several building blocks, notably the addition of *calendar time durations* (e.g. *a year and a day*), multiplication of *calendar time durations* (e.g. *356 days*) and a *modifier of calendar time duration.* In Section 5, we analyse the other sample sentence “Adam is five years older than Bill”. Two possible alternatives with slight technical differences are proposed, and the advantages and disadvantages of both are discussed. In the concluding Section 6, the proposed solutions are summarized and further research suggested. The latter includes, inter alia, an analysis of the sort(s) of calendars that are actually used by people in their everyday lives, such as the calendars that are implemented in cell phones and computers.

1. Fundamentals of TIL

As mentioned above, our background theory is TIL, namely the version presented in Duží et al. (2010) – see also Tichý (1998) and Tichý (2004). From a formal point of view, TIL is a partial, typed lambda calculus with a procedural semantics. This means that we explicate the meanings of expressions as abstract procedures encoded by the expressions. These procedures are rigorously defined as TIL *constructions.* All the entities of the stratified ontology of TIL receive a type. Thus, the core of TIL consists of the definition of the type hierarchy and the definition of constructions. For the sake of simplicity, we first define types of order 1 that include types of non-procedural objects, then four kinds of constructions, and finally the ramified hierarchy of types of order n.

Definition 1 (*types of order 1*)

Let B be a *base*, where a base is a collection of pair-wise disjoint, non-empty sets. Then:

(i) Every member of B is an elementary *type of order 1 over B.*

(ii) Let $\alpha, \beta_1, \ldots, \beta_m$ ($m > 0$) be types of order 1 over B. Then the collection $(\alpha\ \beta_1 \ldots \beta_m)$ of all m-ary partial mappings from $\beta_1 \times \ldots \times \beta_m$ into α is a functional *type of* order *1 over B.*

(iii) Nothing is a *type of order 1 over B* unless it so follows from (i) and (ii). □

For the purposes of natural-language analysis, we are currently assuming the following base of *atomic types*, which form part of the ontological commitments of TIL:

ο: the type of truth-values = {T, F}

ι: the type of individuals (the universe of discourse)

τ: the type of real numbers (doubling as time points)

ω: the type of logically possible worlds (the logical space)

As mentioned above, in TIL we have two mutually independent modal parameters, namely possible worlds and times. Thus, unlike Montague's IL logic, we can apply explicit intensionalisation and temporalisation, which we need for the analysis of empirical sentences containing time references.[4]

Definition 2 (*construction*)

(i) A *variable* x is the *construction* that constructs an object X of the respective type assigned to x as the range of x dependently on a valuation v; x *v-constructs* X.

(ii) Where X is an object whatsoever, *Trivialization* is the construction ${}^0X$. ${}^0X$ constructs X without any change of X.

(iii) Let $X, Y_1, \ldots, Y_m$ be constructions. Then *Composition* $[X\ Y_1 \ldots Y_m]$ is the following *construction*. If X *v-constructs* a function g of a type $(\alpha\beta_1\ldots\beta_m)$, and $Y_1, \ldots, Y_m$ *v-construct* entities $\mathrm{B}_1, \ldots, \mathrm{B}_m$ of types $\beta_1, \ldots, \beta_m$, respectively, then the *Composition* $[X\ Y_1 \ldots Y_m]$ *v-constructs* the value (an entity, if any, of type α) of g on the tuple argument $\langle \mathrm{B}_1, \ldots, \mathrm{B}_m \rangle$. Otherwise the *Composition* $[X\ Y_1 \ldots Y_m]$ does not *v-construct* anything and so is *v-improper*.

(iv) The *Closure* $[\lambda x_1 \ldots x_m\ Y]$ is the following *construction*. Let $x_1, x_2, \ldots, x_m$ be pair-wise distinct variables v-constructing entities

[4] See Duží et.al (2010, § 2.4.3) for criticism of Montague's implicit intensionalisation.

of types $\beta_1, \ldots, \beta_m$ and Y a *construction* typed to v-construct an α-entity. Then $[\lambda x_1 \ldots x_m\ Y]$ is the *construction* λ-*Closure*. It *v-constructs* the following function f of the type $(\alpha\beta_1\ldots\beta_m)$. Let $v(B_1/x_1, \ldots, B_m/x_m)$ be a valuation identical with v at least up to assigning objects $B_1/\beta_1, \ldots, B_m/\beta_m$ to variables $x_1, \ldots, x_m$. If Y is $v(B_1/x_1, \ldots, B_m/x_m)$-improper (see iii), then f is undefined at $\langle B_1, \ldots, B_m\rangle$. Otherwise the value of f at $\langle B_1, \ldots, B_m\rangle$ is the α-entity $v(B_1/x_1, \ldots, B_m/x_m)$-constructd by Y.

(v) Nothing is a *construction*, unless it so follows from (i) through (iv). □

Remark. Definition 2 leaves out constructions *Single* and *Double Execution*, ^1X and ^2X, which we do not need for the present study.

Definition 3 (*ramified hierarchy of types*)

T_1 (*types of order 1*). See Def. 1.

C_n (*constructions of order n*)

(i) Let x be a variable ranging over a type of order n. Then x is a *construction of order n over B*.

(ii) Let X be a member of a type of order n. Then ^0X is a *construction of order n over B*.

(iii) Let $X, X_1, \ldots, X_m$ $(m > 0)$ be *constructions* of order n over B. Then $[X\ X_1\ldots X_m]$ is a *construction of order n over B*.

(iv) Let $x_1, \ldots, x_m, X$ $(m > 0)$ be *constructions* of order n over B. Then $[\lambda x_1\ldots x_m\ X]$ is a *construction of order n over B*.

(v) Nothing is a *construction of order n over B* unless it so follows from C_n (i)-(iv).

T_{n+1} (*types of order n + 1*).

Let $*_n$ be the collection of all constructions of order n over B. Then

(i) $*_n$ and every type of order n are *types of order* $n + 1$.

(ii) If $m > 0$ and $\alpha, \beta_1, \ldots, \beta_m$ are types of order $n + 1$ over B, then $(\alpha\ \beta_1 \ldots \beta_m)$ (see T_1 ii)) is a *type of order* $n + 1$ *over B*.

(iii) Nothing is a *type of order n + 1 over B* unless it so follows from T_{n+1} (i) and (ii). □

Remark. As a notational convention, 'a/α' means that the object a is of type α, while '$C \rightarrow \alpha$' means that the construction C is typed to *v*-construct objects of type α. Where C is a construction, the frequently used Composition $[[C\ w]\ t]$ will be abbreviated as C_{wt}.

Empirical expressions and sentences denote so-called *PWS-intensions*, which are functions of type $((\alpha\tau)\omega)$, abbreviated as '$\alpha_{\tau\omega}$', that is, mappings from possible worlds to *chronologies* of objects of type α. Note that in TIL we have two independent modal parameters at our disposal, namely possible worlds of type ω and times of type τ, which is another reason we recommend TIL as a theory apt for analysis of empirical expressions with time references. Throughout this paper we use variables w and t ranging over ω and τ, respectively. Intensions, being functions of $\alpha_{\tau\omega}$, are *v*-constructed by Closures of the form $\lambda w \lambda t\ C$, where $C \rightarrow \alpha$.[5]

2. Time point and time interval

In TIL, time is modelled by the type τ, the set of real numbers.[6] Therefore, any *time point* is modelled as a real number. Thus, the binary relation of equality between two *time points* is defined as the identity relation $=/(o\tau\tau)$. The binary relation of precedence between two *time points* is defined as the less-than relation $</(o\tau\tau)$. Furthermore, the binary relation $\leq$ is defined as $t_1 \leq t_2$ iff $t_1 = t_2$ or $t_1 < t_2$. For the sake of simplicity, when applying these relations, we use infix notation (and without Trivialization) as is common in mathematics.

Now we are going to deal with time intervals. Unfortunately, the term 'time interval' is commonly used with two different meanings. It either denotes a *time duration*, such as *20 seconds* as in the sentence "The light changes colour every 20 seconds", or a particular *interval* of time points, i.e. the set of real numbers/time points with the property that any number that is in between two numbers in the set is also included in the set. Time

---

5 Jespersen talks about this logical form characteristic of empirical expressions as *explicit intensionalization and temporalization*; see Jespersen (2005).

6 In practice, time can be modelled in a different way. For example, for the purposes of programming, discretization of time is necessary; in such cases, time can be modelled as a set of integer numbers.

duration will be dealt with in the next section. To avoid confusion, from now on we will terminologically distinguish 'time duration' and 'time interval'. The latter notion is mathematically defined as follows.

Definition 4

Let t_1, t_2 be time points such that $t_1 < t_2$. Then a *time interval between the time points* t_1 and t_2 is a bounded half-open interval $[t_1, t_2)$ excluding the point t_2. Hence, it is a set of real numbers constructed by $\lambda t\ [t_1 \leq t < t_2]$.

The reason we define *time interval* as a half-closed mathematical interval excluding the last end-point is this. We assume that *time is linear and continuous.*[7] If we went for discrete time, then the interval could be closed. Yet on the assumption of a time continuum, the alternatives would cause severe problems. The first alternative, an open interval, for instance (the first moment of 2017, the last moment of 2017) obviously excludes the first and the last moments of 2017. The second alternative, a closed interval [first moment of 2017, last moment of 2017], presumes the existence of a last moment of 2017. If there were such a *time point x*, there would be infinitely many time points between *x* and the first moment of 2018. Therefore, *x* would not be the last moment of 2017. The third alternative would be the half-open interval excluding the first end-point. Yet, since it is natural to deal with time as flowing forward, we choose the half-open interval excluding the last end-point.[8] Next, we define the *length of a time interval*:

[7] We do not deal here with 'branching time' theories; see, for instance, Placek (2012). These theories have many useful applications in computer science in the research on parallel and concurrent processes; see Nain & Vardi (2007).

[8] Another reason for this approach is explained in detail in Hobbs & Pan (2004, 76). The authors suggest that "we get a cleaner treatment if, for example, all times of the form 12:xx a.m., including 12:00 a.m., are part of the same hour and day, and all times of the form 10:15:xx, including 10:15:00, are part of the same minute" and support this claim by practical examples.

Definition 5

The *length L of time interval* $[t_1, t_2)$ is a (non-negative) real number $L = t_2 - t_1$.

Depending on the calendar, there are time intervals that play a special role in our everyday lives, like the one denoted by 'the day of September 11, 2001'. In general, many time intervals receive a name in our everyday vernaculars, like 'the year 2017', the 'month January of 2017', etc. For example, let t_1 be the first moment of January 1st, 2017, and t_2 the first moment of January 1st, 2018. Then the interval $[t_1, t_2)$ is the *year 2017*. Yet, as mentioned above, the length of particular time intervals, such as the year 2017, depends on calendars.

3. Calendars and time duration

Time durations[9] are objects that are denoted by expressions such as 'year', 'month', 'day', 'hour', 'minute', 'second', '5 years', '5 years ago', 'a year and a month' and '15 hours 30 minutes'. These objects are dependent on a particular calendar for their duration. There have been many *calendars* in use around the world. Some of them, such as the *Gregorian calendar* and *Julian calendar*, have relatively minor differences, others, such as the fiscal calendar for accounting and budget purposes, may define different rules for various *time durations*, or the duration of a year according to a solar or a lunar calendar also differ.

The Cambridge Dictionary[10] defines *calendar* as "the system used to measure and arrange the days, weeks, months, and special events of the year according to a belief system or tradition" and the *Gregorian calendar* as a "system used in many parts of the world to divide the 365 days of the year into weeks and months, and to number the years".[11]

9 Sometimes the alternative term 'time span' is used.

10 See *http://dictionary.cambridge.org/dictionary/english/calendar*.

11 While in this paper we discuss ordinary, recently used calendars that operate with objects such as *months* and *minutes*, historically there have been different peculiar calendars like the one operating with the "sinking-bowl" of water for measuring intervals of time in India; see Plofker (2011). As far as we know, the Ancient Britons, probably

In what follows we take a *calendar method* as being an object of type $*_n$, i.e., a construction. This is a simplification, for sure, yet for our purposes this simplification is harmless. From the practical point of view, it is more important to analyse the structure of a calendar time durations, because the reasonable definitions of calendar methods computing these durations are rare.[12] A *calendar* (e.g. the *Gregorian calendar*) is then an empirical function of type $((*_n\tau)\omega)$, or '$*_{\tau\omega}$' for short, that yields a *calendar method* for a given world and time. We define a calendar in this way, because calendar methods are based on empirical observations (such as the solar cycle or the lunar cycle), and they can be adjusted from time to time.

There are time durations that differ in different calendars. Yet even within one and the same calendar these time durations are not of the same length. This is due to leap seconds and leap days. Thus, we define:

Definition 6

A *time duration* is a function of type $(\tau\tau)$. A *time interval* $[t_1, t_2)$ has a *time duration d* iff $[d\ t_1] = t_2$. A *calendar time duration* is a function from *calendar methods* to time durations. Hence a *calendar time duration* is a function of type $((\tau\tau)*_n)$.

For example, a *year* is a *calendar time duration* that for a given *calendar method c* associates any *time point* t_1 with the *time point* t_2 that comes one year after t_1 (according to a given calendar method). Note the difference between the *length* of an *interval* and *time duration* of an *interval*. The *length* of a given *interval* is an exact real number, whereas its *time duration*, for instance a *year in the Gregorian calendar*, does not determine a definite number. It can be 365 or 366 *days in the Gregorian calendar*, the lengths of particular *days* can also differ due to leap seconds, etc. Only

under the influence of the Druids, used similar bowls for measuring intervals of time. The bowls had a small hole in the bottom, and in use it was placed on the surface of water, which slowly leaked into it until, after a certain interval of time, the bowl sank. The interval was the unit of time; in the case of the bowl found in County Antrim, Northern Ireland, it was approximately one hour.

[12] This topic would be a subject of further research that is out of the scope of the present paper.

when obtaining additional pieces of information, like the exact point of the beginning of the *interval* and a *calendar*, is one able to compute rigorously the actual *calendar time duration* of a given *interval*.

For some applications, it may be feasible to define *time duration* in a simpler way, for instance as a time difference in seconds. However, this is not acceptable when analysing natural language. As mentioned above, due to the existence of leap days and leap seconds, various *time intervals* that have the same *time duration*, for instance a *minute*, may have different lengths. This is particularly obvious of the *calendar time duration month.* Saying that this or that lasted a month one is not conveying much information. It can be 28, 29, 30 or 31 days.

Things are even more complicated with leap days, even within the *Gregorian calendar*. The most complicated problem of *calendar time durations* is the question what day follows exactly *one year after the 29*th *of February* of a leap year, for instance *2016.*[13] Intuitively, one would say that it must be a regular day in the calendar, not a virtual one. For sure, because this question is important, for instance for legal purposes, to compute the age of criminal responsibility. In England children under 10 cannot be arrested or charged with a crime. If a child was born on the 29th of February 2000 and commits a criminal act on the 28th of February 2010, are they responsible? Unfortunately, there is no consensus on the solution of this problem.[14]

For the moment, let us assume that[15] [16]

---

[13] Note that the same problem applies to leap seconds, for example the December 31st, 2005, 18:59:60 leap second.

[14] To illustrate, we have tested several programming platforms; the .NET framework class DateTime gives February 28th, 2017 as *one year after February 29*th, *2016*, the same as the Java class GregorianCalendar (Java forces the programmers to choose explicitly the calendar they want to use). The PHP class DateTime, however, yields March 1st, 2017.

[15] The technical details of the addition of *time* are explicated in TIL in Section 5. However, in accordance with our intuition it should hold that the addition of *time duration d* to a *time interval* $[t_1, t_2)$ yields a *time interval* $[d(t_1), d(t_2))$.

[16] In what follows, we assume the *Gregorian calendar* for the purpose of obtaining *time durations* from *calendar time durations*.

February 29th 2016 + 1 year = February 28th 2017

The following additions are less problematic:

February 28th 2017 + 1 year = February 28th 2018

February 28th 2018 + 1 year = February 28th 2019

February 28th 2019 + 1 year = February 28th 2020

The troubling part is immediately apparent, because one would also assume that

February 29th 2016 + 4 years = February 29th 2020

From this, it consequently follows that:

(((*February 29th 2016* + 1 *year)* + *1 year)* + 1 *year)* + *1 year* ≠ *February 29th 2016 + 4 years*

Thus, when we define the operation of adding *calendar time durations*, the following holds:

((1 *year* + *1 year)* + 1 *year)* + *1 year ≠ 4 years*

In general, we cannot define the operation of multiplication of a *calendar time duration* as a series of additions of that *calendar time duration*. Moreover, the same problem arises with *negative calendar time durations*. We have

February 28th 2017 + (-1 year) = February 28th 2016

But

(February 29th 2016 + 1 *year)* + *(-1 year)* ≠ *February 29th 2016*

Intuitively, it should be clear what we mean by '+ 1 year' or '-1 year'; yet the rigorous definition is needed. Thus we define.

Definition 7

A *calendar time duration d* is *positive* according to a *calendar (method) c* iff for all *time points t* holds that $t < [[d\ c]\ t]$. A *calendar time duration d* is *negative* according to a *calendar (method) c* iff for all *time points t* holds that $t > [[d\ c]\ t]$. Finally, a *calendar time duration d* is *zero* according to a *calendar (method) c* iff for all *time points t* holds that $t = [[d\ c]\ t]$.

For example, *15 hours 30 minutes* is a *positive calendar time duration* (like in "The cricket match lasted 15 hours and 30 minutes") and *5 years ago* is a *negative calendar time duration* (like in "I last saw him 5 years ago.") according to the *Gregorian* as well *Julian calendar*.

A *zero-calendar time duration* is used in our common vernacular to communicate that one event followed another without delay, that is, *immediately* as in "After being mixed, the liquid turned red immediately." For comparison, consider a similar sentence, "After being mixed, the liquid turned red 2 minutes later" with the *positive calendar time duration 2 minutes*.

4. "A year has 365 days"

Since the sentence does not mention any calendar, its meaning is pragmatically incomplete; it means that the construction encoded by the sentence is typed to *v*-construct a proposition, but it is an open construction with a free variable $c \rightarrow *_{\tau\omega}$ ranging over calendars. From the linguistic point of view, this is a case of *ellipsis*. Only when obtaining a piece of information about a pragmatic context, the sentence can be completed so that its meaning would be a closed construction denoting a proposition the truth-condition of which can be evaluated like, for instance, is the case of the sentence "A year has 365 days according to the Julian calendar".

Both expressions 'a year' and '365 days' denote a *calendar time duration*. Thus, the meaning of the sentence "a year has 365 days" is this open construction:

$$\lambda w \lambda t\ [{}^0{=}_{(\tau\tau)}\ [{}^0Year\ c_{wt}]\ [{}^0365Days\ c_{wt}]]$$

$Year$, $365Days/((\tau\tau)*_n)$, $=_{(\tau\tau)}/(o(\tau\tau)(\tau\tau))$, $c \rightarrow *_{\tau\omega}$, $w \rightarrow \omega$, $t \rightarrow \tau$.

Clearly, the construction of the function denoted by the term '365 days' can be further refined. To facilitate this, we are going to define the operation of *adding calendar time durations*, the entity *modifier of calendar time duration* and the operation of *multiplying* a *calendar time duration* by a number.

Definition 8

The *addition* of *calendar time durations*, *AddTD*, is defined as follows:
$^0AddTD = \lambda d_1\, \lambda d_2\, \lambda e\, \lambda t\; [[d_2\; e]\; [[d_1\; e]\; t]]$
$d_1, d_2 \rightarrow ((\tau\tau)*_n)$, $e \rightarrow *_n$, $t \rightarrow \tau$, $AddTD\; /\; ((((\tau\tau)*_n)((\tau\tau)*_n))((\tau\tau)*_n))$

Definition 9

A *modifier of calendar time duration* is a function from *calendar time durations* to *calendar time durations*.[17] The TIL type of a *modifier of calendar time durations* is thus $(((\tau\tau)*_n)((\tau\tau)*_n))$.

Note that applying the function *AddTD* to a *calendar time duration* d_1 yields a *modifier of calendar time durations*. If this modifier is applied to a *calendar time duration* d_2, it yields the *calendar time duration* that is the sum of d_1 and d_2.

As a consequence of the above observation regarding *February 29th, 2016*, we cannot define the multiplication of *calendar time durations* simply as adding time durations, since the details are different for different calendars and their irregularities (such as leap days).

Definition 10

The operation *MulTD* of *multiplying calendar time duration* is a function of type $((((\tau\tau)*_n)((\tau\tau)*_n))\tau)$ that associates a real number x with a *modifier of a calendar time duration* M such that M applied to a *calendar time duration* d yields as its value a *calendar time duration* that is x times longer than d.

[17] Note that there is no requirement for a *modifier of a calendar time duration* to be a total function.

Example: For the Gregorian calendar (*GrC*) it holds that

$$[[[^0MulTD\ ^0 0.5]\ ^0Year]\ ^0GrC_{wt}]\ ^0{=}_{(\tau\tau)}\ [[[^0MulTD\ ^0 6]\ ^0Month]\ ^0GrC_{wt}].$$

Also, since y*ear* is a *positive time duration* according to the Gregorian calendar, $[[[^0MulTD\ ^0\text{-}1]\ ^0Year]\ ^0GrC_{wt}]$ *v*-constructs the negative time duration a *year ago* (again according to the Gregorian calendar).

The resulting *modifier of a calendar time duration* may yield for some *calendar time durations* that are defined in a certain *calendar* a *calendar time duration* that is undefined in this *calendar*, depending on the multiplication number. For example, a week is well-defined in the Gregorian calendar (as 7 days), and any integer multiplication, such as 2 weeks, 3 weeks, and so on are well-defined as well. However, a quarter of a week is not. Therefore, the modifier *multiply by 0.25* applied to *week* yields a *calendar time duration* that is undefined in the Gregorian calendar (but may very well be defined in other calendars).

Thus the more detailed analysis of the sentence "A year has 365 days" comes down to this construction.

$$\lambda w \lambda t\ [^0{=}_{(\tau\tau)}\ [^0Year\ c_{wt}]\ [[[^0MulTD\ ^0 365]\ ^0Day]\ c_{wt}]]$$

$Year$, $Day/((\tau\tau)*_n)$, $365/\tau$, $=_{(\tau\tau)}/(o(\tau\tau)(\tau\tau))$, $c \rightarrow *_{\tau\omega}$

In the interest of better readability, we may improve this analysis by defining the shorthand function *Days*:

$$^0Days = \lambda x\ [[^0MulTD\ x]\ ^0Day]$$

The analysis of "A year has 365 days" is then this construction:

$$\lambda w \lambda t\ [^0{=}_{(\tau\tau)}\ [^0Year\ c_{wt}]\ [[^0Days\ ^0 365]\ c_{wt}]]$$

5. "Adam is 5 years older than Bill"

The sentence "Adam is 5 years older than Bill" does not mention any specific calendar, hence its meaning is again pragmatically incomplete. Traditionally, we consider the age of a person to be a particular number, e.g. "Adam is 27" (years old). It is, however, the often-unspoken part with the word "years" that raises the question "*according to which calendar*"? And the same applies to the difference in age of two people.

For the sake of simplicity, we introduce two additional pieces of shorthand for *calendar time duration*:

$$^0Years = \lambda x\ [[^0MulTD\ x]\ ^0Year]$$
$$^0Months = \lambda x\ [[^0MulTD\ x]\ ^0Month]$$

As always, we start with the type analysis. Both "Adam" and "Bill" denote individuals and the term "5 years" denotes a *calendar time duration*. The expression "is 5 years older than" denotes a relation-in-intension between individuals, i.e.[18]

$$5\_Years\_Older/(o\iota\iota)_{\tau\omega}$$

Thus a coarse-grained analysis of our sentence is simply this construction:

$$\lambda w \lambda t\ [^0 5\_Years\_Older_{wt}\ ^0Adam\ ^0Bill]$$

However, refinement of this analysis is rather complicated. First, there is an ambiguity. Either 'five years older' means exactly five years older, or approximately five years older.

The first option is the simpler one and allows us to define the relation of being *five years older* by means of the entity *Older* of type $(o\iota\iota(\tau\tau))_{\tau\omega}$: the ternary relation-in-intension between two individuals and a *time duration*.

$$^0 5\_Years\_Older = \lambda w \lambda t\ \lambda xy\ [^0Older_{wt}\ x\ y\ [[^0Years\ ^05]\ c_{wt}]]$$

Types: x, $y \rightarrow \iota$; $[[^0Years\ ^05]\ c_{wt}] \rightarrow (\tau\tau)$; $c \rightarrow *_{\tau\omega}$: a calendar; $=/((o\iota\iota)_{\tau\omega}(o\iota\iota)_{\tau\omega})$: the identity of binary relations-in-intension.

---

[18] In general, a TIL type of an n-ary relation-in-intension is $(((o\alpha_1\alpha_2\ldots\alpha_n)\tau)\omega)$, or '$(o\alpha_1\alpha_2\ldots\alpha_n)_{\tau\omega}$' for short, where α_i may be any TIL type.

The refined analysis of the first option is thus:

$\lambda w \lambda t\ [{}^0Older_{wt}\ {}^0Adam\ {}^0Bill\ [[{}^0Years\ {}^05]\ c_{wt}]]$

To analyse the second option, we must introduce a measure of tolerance. Intuitively, the term "60-months-in-the-Gregorian-calendar" seems more specific and therefore would allow for less tolerance. An inaccuracy of 1 month might be negligible for 5 years, significant for 60 months and too much for 1826 days (a best guess for the number of days in 5 years).[19]

This different level of tolerance is, however, lost in the first analysis. Let us again assume that the used *calendar c* is the *Gregorian calendar*. According to this calendar the *time duration* of *5 years* is the same as the *time duration* of *60 months*. From this it follows that the proposition *v*-constructed by the following construction is the same as the one *v*-constructed by the previous construction.

$\lambda w \lambda t\ [{}^0Older_{wt}\ {}^0Adam\ {}^0Bill\ [[{}^0Months\ {}^060]\ c_{wt}]]$

In other words, these two constructions are equivalent by producing one and the same proposition:

$$\lambda w \lambda t\ [{}^0Older_{wt}\ {}^0Adam\ {}^0Bill\ [[{}^0Years\ {}^05]\ {}^0GrC_{wt}]] = \lambda w \lambda t\ [{}^0Older_{wt}\ {}^0Adam\ {}^0Bill\ [[{}^0Months\ {}^060]\ {}^0GrC_{wt}]]$$

To allow for different tolerance for different units of time (years, months, …) we apply the relation-in-intension *OlderCal* of type $(o\iota\iota((\tau\tau)*_n)*_n)_{\tau\omega}$; a relation-in-intension between two individuals, a *calendar time duration* and a *calendar method*. The analysis of the second reading of the sentence is then:

$\lambda w \lambda t\ [{}^0OlderCal_{wt}\ {}^0Adam\ {}^0Bill\ [{}^0Years\ {}^05]\ c_{wt}]$

[19] The difference is important in a multi-valued logic where the increasing tolerance results in a lower degree of truthfulness.

6. Conclusion

In this paper, we have defined several basic notions needed for the analysis of sentences involving time references, of which the most important are *time duration* and *calendar time duration*, including *year*, *month*, *day*. Moreover, we proposed a method for dealing with *adding* and *multiplying calendar time durations* so as to be able to present a fine-grained analysis of terms like '*15 hours and 30 minutes*' or '*5 years ago*' respecting leap days and seconds. To this end we defined a *modifier of calendar time duration*.

There are two interconnected avenues of *further research* that we believe will result in significant contribution to the topic. First, the connection between *time intervals* and *calendar time durations*; we should be able to compute the *time interval* denoted by a *calendar time duration* like for instance that denoted by the term 'year 2017'. Furthermore, there is a *calendar* object involved in the specification of time points denoted by expressions such as 'January 1^{st}, 2017, 15:30'. This requires further investigation both of the structure of a calendar *date* (i.e. the object denoted by 'January 1^{st}, 2017, 15:30' before any particular calendar is taken into consideration) and specialties of individual common *calendars*. These include phenomena such as time zones and daylight savings time, and also the fact that the official reference points for many calendars (e.g. the birth of Jesus Christ or the creation of the world) are imprecise at best and made up at worst. It is therefore unreasonable to claim that the time in any computer is computed on the basis of these reference points, because in order that the computer could compute time, it must have some constant time point to start with. To this end is usually used some external impulse, for instance, synchronization with Internet time. Thus the reference point is not settled at zero; rather, the computer takes as the starting time-point, for instance the time "it is now the time 2017-06-01, 14:36:00" and computes time from this reference point.

Acknowledgements

This research has been supported by the Grant Agency of the Czech Republic, project No. GA15-13277S, *Hyperintensional logic for natural language analysis*, and by the internal grant agency of VSB-TU Ostrava, project SGS No. SP2017/133, "Knowledge modeling and its applications in software engineering III".

References

DUŽÍ, M., JESPERSEN, B. & MATERNA, P. (2010): *Procedural Semantics for Hyperintensional Logic. Foundations and Applications of Transparent Intensional Logic*. Berlin: Springer, series Logic, Epistemology, and the Unity of Science, vol. 17.

DUŽÍ, M. (2010): Tenses and Truth-Conditions: a Plea for *If-then-else.* In: Peliš, M. (ed.): *Logica Yearbook 2009.* London: College Publications, 63-80.

HOBBS, J. R. & PAN, F. (2004): An Ontology of Time for the Semantic Web. *ACM Transactions on Asian Language Information Processing* 3, 66-85.

JESPERSEN, B. (2005): Explicit Intensionalization, Anti-Actualism, and How Smith's Murderer Might not Have Murdered Smith. *Dialectica* 59, No. 3, 285-314.

NAIN, S. & VARDI, M. Y. (2007): Branching vs. Linear Time: Semantical Perspective. In: Namjoshi, K. S., Yoneda, T., Higashino, T. & Okamura, Y. (eds.): *Automated Technology for Verification and Analysis. ATVA 2007. Lecture Notes in Computer Science*. Vol. 4762, 19-34.

OHLBACH, H. J. (2004): Calendrical Calculations with Time Partionings and Fuzzy Time Intervals. *Lecture Notes in Computer Science* 3208, 118-133.

OHLBACH, H. J. & GABBAY, D. (1998): Calendar Logic. *Journal of Applied Non-Classical Logics* 8, No. 4, 291-323.

PLACEK, T. (2012): On Individuals in Branching Histories. *Synthese* 188, No. 1, 23-39.

PLOFKER, K. (2011): The Mathematics of Measuring Time: Astronomical Timekeeping and the Sinking-Bowl Water-Clock in India. In: Jardine, D. & Shell-Gellasch, A. (eds.): *Mathematical Time Capsules: Historical Modules for the Mathematics Classroom* 77, 55-62.

PRIOR, A. N. (1957): *Time and Modality*. Oxford: Oxford University Press.

PRIOR, A. N. (1962): Tense Logic and the Continuity of Time. *Studia Logica* 13, 133-148.

PRIOR, A. N. (1967): *Past*, *Present and Future*. Oxford: Clarendon Press.

TICHÝ, P. (1980): The Logic of Temporal Discourse. *Linguistics and Philosophy* 3, 343-369. Reprinted in: Tichý (2004), 373-369.

TICHÝ, P. (1988): *The Foundations of Frege's Logic.* Berlin & New York: deGruyter.

TICHÝ, P. (2004): *Collected Papers in Logic and Philosophy*. Svoboda, V, Jespersen, B. & Cheyne, C. (eds.), Prague: Filosofia, Czech Academy of Sciences & Dunedin: University of Otago Press.