A Space-Time Logic

Ulisses Ferreira

e-mail: *ulisses@philosophers-fcs.org*

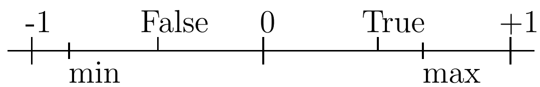
Abstract

It is well known that space and time are two primary notions that have always been present in the human consciousness independently of contexts. In this article, I propose a space-time logic with five epistemic values, together with uncertainty. My work considers the expressiveness of the language and, because of this, I also introduce a deductive system for the present logic in the same syntax. The present author's final aim is mainly to introduce a general framework for writing formal semantics of programming languages that support code mobility e.g. mobile agents, among other more traditional applications such as knowledge representation.

Moreover, the present logic is powerful enough for representing more sophisticated forms of reasoning, such as to weigh up possibilities. For being able to make fair judgments one should consciously attach honesty factors (in this case, floating-point numbers in $[-1, +1]$) to diversified implications between premises and some conclusion in such a way that that conclusion from weighing up possibilities can be based on those factors. As it is well known, this is a very common form of reasoning. Thus, the @-logic also provides uncertainty-based representation combined with deduction for such purposes.

1 Introduction

For many years, I designed a programming language which is called Plain[38]. One of the general purposes of Plain is to provide a *hybrid* programming paradigm. Since 1992, it has been clear that the idea is not the same as *multiple* paradigm, and that is one of the novelties of Plain: Often, a multi-paradigm programming language is just a union of two or more languages of different paradigms, while a programmer may choose which one she is going to use. Thus, from the present perspective, in contrast with the latter, the hybrid paradigm of Plain has some high level of integration. Plain is also based on knowledge-and-belief representation, and can be used for expert systems over MYCIN confidence factors. From this, logic in my work has always had a flavour that contrasted with Boolean logic, for instance. Even in the very beginning of Plain, two values [40] were not nearly enough. A value called ***uu*** has existed in Plain from start. Now, Plain had three values: *tt*, *ff*, and *uu*: true, false and unknown, respectively, as well as a real (or rational, in the end) range between -1 and $+1$, with four variables where $min, max, False, True : [-1, +1]$, and $min \leq max$ and $False \leq True$, as follows.



A typical configuration representing *uu*.

Of course, every computer scientists is interested in logics. My particular interest in logics started when I received an acceptance letter from the Computing Department of the University of London stating that Dov M. Gabbay was willing to supervise me. We do not know each other. However, in my particular case, that was not a good moment of my life for that.

In 90's, I made some design of a temporary language for representing permissions of accesses by users at different levels, and, once I entered Sussex University, there, ended up designing a three-valued logic programming [55, 56] language[39], called Globallog, and which in turn was initially referred to as Kleene after Kleene's death in 1994. I prefered to leave this name for other logic programming languages designers, since the corresponding language was already conceived for being no more than a subset of the hybrid paradigm language, Plain. In any case, this language has been flexible in such a way that programmers might write inconsistent predicates. From this, there is now the need to think in terms of an inconsistent value which has been called *ii*: the fourth value.

In 1996, there was a brief opportunity to study with supervision of some particular Professor but for some reason I could not go. However, his interest on closed-world assumption and negation by failure suggested me to think about adapting some pioneering ideas from Prolog to the logical language, Kleene at that time.

On the other hand, until the year 2000, the $False$ and $True$ thresholds above were constants for each run[37], often $False < True$ had hold in projects, but soon I realized that, for more complex systems, $False$ could increase or $True$ could decrease in such a way that $False = True$ would hold. Under this condition, a new value appeared and I called this value *kk*: the fifth value, knowable but consistent. Now there are already the ingredients required for a five-valued logic[53].

I consider a programming language as something slightly more than a logic: there is control, flow, sometimes side effects and i/o operations, for instance: lots of boring details at low level. Nonetheless, although the pieces of knowledge for designing a logic and a programming language are different, the skills seem to be the same. Thus, I started studying logics.

Since ML was one of the main programming languages by Edinburgh University (UoE), my work required some more knowledge and care to propose PLAIN. In 1997, my first mobile agent in PLAIN run on the Internet with a demonstration to friends, sending e-mail with partial results. Thus, my first argument was that mobile agents[97] were not a suitable application on purely functional languages[60]. Anyway, I presented 10% of what I wanted to do. I would like to have presented more but had previously consulted an oracle and its response was that 10% would be suitable for the purpose. At UoE, a colleague of mine proposed a mobile calculus, and then an idea came up: a *spatio-temporal logic* which could be used for physical aspects of computation, for writing formal semantics of mobile agents as well as statements in programming languages that supported them. I found that the spatio-temporal logic ought to contribute to the general logic, and then did not propose a separate space-time logic, although a classical binary version was built in there in the work. As an example, in terms of its expressiveness in comparison to predicate logics, the whole logic would be able to make clear that the notion of inconsistency is applied to the same place and to the same time only. In late 2000 and early 2001, having deduced the ingredients, I worked somewhat hard on the @-logic while in Nothern Ireland, London, Dublin and a few places around there. In short, a formula $@s \cdot t[P(x)]$ means $P(x)$ at place s time t. The announcement of the @-logic was sent via the types-l mailing list: See www.cis.upenn.edu/~bcpierce/types/archives/current/msg00677.html. Then, I submitted my work on the @-logic for publication to journals such as *Journal of Logic and Computation* and have improved it in the details in parallel to other original and published pieces of work. This is the end of the brief story of what I present here.

Classical logics, in both propositional and predicate forms[19], traditionally have been the most important logics since George Boole's time[27], with many contributions[28, 45, 95, 85, 89]. In the last century, a number of logics have been developed and established as alternatives to classical logic. In particular, intuitionistic logics, in both propositional and predicate forms, have increasingly attracted the attention of mathematicians and non-mathematicians[12]. In fact, computers have played important rôle in many logics with constructive proofs. As regards intuitionistic logic, since Brouwer and Heyting[12], there have been many important contributions in the field. In [26], for instance, the duality in Cartesian closed categories, λ-calculi, intuitionistic and classical logics from syntactic and semantic viewpoints are investigated, while, regarding philosophy of mathematics, in [13, 18], there are two kinds of defense of classical view in mathematics and logic. The range of subjects is broad. Regarding a more informal and philosophical literature, logics and space-time together do not play lesser relevant rôles[90].

Recently, some other contributions to pure logics[49] have appeared. For example, Arthur Prior[24, 77, 78, 79] and others[51] are some of the important contributors to modal logics[21] and temporal logics. A reference on them is [34].

The proposed language of the present logic is based on five values, *ff*, *tt*, *uu*, *kk*, *ii* representing, briefly speaking, *false*, *true*, *unknown*, *known* and *inconsistent*, an idea whose simplification is somewhat similar to the Belnap four-valued logic[11] in the following sense: *uu* means "neither *true* nor *false*" while *ii* means both. Łukasiewicz[44] introduced many-valued logics or infinitely-many-valued logics, both based on a set of values from *false* to *true*, e.g. $\{0/3, 1/3, 2/3, 3/3\}$. Stephen Kleene also introduced his many-valued system[67], with some modifications on Łukasiewicz's. Here, in some sense, I do not adopt degrees of veracity but instead work on pairs of opposites, although I deal with degrees of veracity in another context of the @-logic called uncertainty. On the other hand, the present calculus is somewhat similar to the Łukasiewicz three-valued logic or Kleene three-valued logic as $\neg ii$ also results in ii. The differences to those many-valued logics[70] will become explicit in sections 2 and 2.2.

Some many-valued logics, as well as modal and temporal logics, were introduced having as motivation the representation of forms of veracity referring to the future, i.e. propositions referring to the future are regarded as neither *true* nor *false*[94]. Thus, we can regard the truth value of the propositions as unknown. However, there are many other uses for the representation of lack of information. Here, I do not philosophically[66] discuss on whether it is possible for one to have knowledge about the future. In any case, for any event, unless we experience it somehow and in a particular situation, we do not normally know whether this event happens or not. In space[20] (a reference on spatial cognition is [46]), the need for the notion of lack of information is essentially the same. We often know what somehow reaches us by communication. Otherwise, events are normally unknown. Because of this, the present space-time calculus is based on the present five-valued logic, in particular, in one of the defined implications.

On the one hand, a number of temporal and spatial logics have been introduced[4] for a number of purposes[43] with success, whereas *spatial reasoning* has been deeply studied. On the other hand, there has been a relatively small number of spatial theories on predicate logics and other attempts have been made. For instance, interval temporal logics are suitable for planning systems and scheduling[3, 6]. For time, we propose a more general approach for representing time for actions, events and tasks, than that of James Allen's temporal logics, which is more at the AI or application level than here[5]. His logic considers time as intervals, which is more general than points and makes his calculus very suitable for planners. Thus, points can be represented as $[p_1, p_2]$ where $p_1 = p_2$. In the present work, dated back to 2000, both space and time are represented as sets, a more general form of representation. If we want to represent cyclical events, we are able to do so by considering unions between intervals, for instance. I mention other aspects, for instance, concerning models and derivations, studied in [22, 30, 73, 92]. However, the present piece of work is in the scope of the emerging *philosophy of computer science*, which is in (philosophical) foundations of computer science, and models are not formalized such as using Kripke models[72], in particular, since this article is long.

There are other pieces of work on applying logics in some areas in computer science[50, 84]. Briefly, in addition to the literature on different logics[31, 35, 36], Girard's linear logic[29, 57, 91] and labelled deductive systems[48] are two of best examples of work that can be applied to computer science. A very good paper on introduction to logics in computer science is in [86].

Little work has been done on spatial logical theories. The Region Connection Calculus[80] is a predicate theory on space. On the other hand, the author of [42] concentrates on a more detailed level of abstraction. There are other approaches and spatio-temporal logics, such as [52], which is a logic for multi-agent problem domains. A different approach for agents is shown in [63]. My approach is to introduce a powerful and expressive language, while I abstract details addressed in specific applications, e.g. AI systems. It seems that, to date, there is no space-time logic, even good and recent literature, e.g. book [88], does not contain relatively significant contribution combining both notions in one formal logic. Instead, in parallel to temporal logics, there has been more specific work on the broad subject, for instance, spatio-temporal databases[65], a model and language[96], predicate theories such as RCC-8, in particular, those useful for AI. In contrast with particular space-time logics, an updated bibliography for data mining research is [83]. In [23], the authors observe the similarities between temporal and spatial structures, but they did not collapsed both into sets. Like in [14] which applies *rough* sets[76] to a spatio-temporal context, I collapse and generalize both notions, although I use *sets* (according to that article, rough set theory[75] provides a way of approximating subsets of a set when the set is equipped with a partition or equivalence relation. The same article contains another related issue. It notes that unfortunately the exact location of spatio-temporal objects is often *indeterminate*, which motivates the definition and interpretation of the uu value, as well as the notion of uncertainty, in a space-time logic.), and because I accept sets (hence intervals) for both space and time in a Cartesian space representation, the present logic might also capture other notions such as geometrical or geographical relational operations and so forth.

However, perhaps because technologies such as mobile-code languages, e.g. Java[8, 61], are relatively recent, we have not had symmetry and balance concerning attention to time and space. We have some temporal logics but, to date, space has had little attention from the academic community with rare exceptions such as [15]. Some proposals, in particular predicate theories, have appeared but no space-time logic has been established. It seems that, for various purposes, it would be desirable to combine both approaches, with respect to space and time, in only one logic.

For this article, let $\mathbf{C}$ be the set of all formulae in the space-time logic i.e. the language of the @-logic. Thus, to start explaining the subject, a classical version of the space-time logic can be defined as follows:

Definition 1 *Let φ and ϕ denote two formulae and α be a variable (a quantifier). Thus, a classical @-logic language corresponds to φ (a non-terminal symbol, in formal languages terms) in the grammar as the following:*

$$\begin{aligned}
&\varphi \longmapsto P \mid \neg\varphi \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \varphi \Rightarrow \varphi \mid \varphi \Leftrightarrow \varphi \mid (\exists\alpha)\ \varphi \mid (\forall\alpha)\ \varphi \\
&\varphi \longmapsto @s \cdot t[\varphi] \mid \phi \mid @s \cdot t[\ ] \\
&\phi \longmapsto @s \cdot t[\![\varphi]\!] \\
&\alpha \longmapsto x \mid y \mid \ldots
\end{aligned}$$

where φ is the starting symbol, P stands for a proposition or predicate, α denotes a quantified variable in $\mathbf{C}$, $s \in \mathbb{R}^3$ or $s \subseteq \mathbb{R}^3$ and $t \in \mathbb{R}$ or $t \subseteq \mathbb{R}$, depending on the focus of attention, points or sets, sometimes intervals. The author apologizes for this abuse of notation here. Let ϕ stand for semantics, a function symbol with domain in $\mathbf{C}$ and semantic image. Both my syntax and semantics are simple: if φ is a formula, then

$$@s \cdot t[\varphi]$$

is a formula in the present logic, which is called @-logic, where s indicates the place where φ holds, and t indicates the time when φ holds. A particular case is

$$@s \cdot t[\ ]$$

which intuitively indicates that there is no assertion for space s and time t. This notation is capable of representing an empty data base or theory. Accordingly, by using a slightly different notation, we are able to express $@s \cdot t[\![\varphi]\!]$ as "the meaning of φ at place s and time t".

Note that this language implicitly introduces a conjunction between the space and the time, for every space-time formula. Now, for any expression, everything happens intuitively in the same way as it would have happened in the classical logics, except that now there are variables of space and time, and that the classical logic expression in question is valid in the new context.

There are two standard variables, namely *here* and *now* that can be used in space and time expressions, respectively. There is an alternative pair of variables with the same meanings in symbols, namely $\oplus$ and $\otimes$, respectively.

The notion of space-time can be applied at more than one level. Thus, in this article, I present a space-time deductive system that applies this pair of concepts at two or three levels:

- Deduction (sequents)
- Logic (syntax)
- Object language (the semantics of) or application

A calculus is embeded in this system and vice-versa, in such a way that I address the whole set of deductive rules as either. The present calculus is called @-calculus and, accordingly, the logic that I define here is called @-logic. Thus, I distinguish space and time referred to in the statements from space and time where and/or when the derivation carries out, and from the space and time referred to in nesting statements, i.e. the language permits references to other epistemic levels of space-time.

The time component is based on a flow that can be represented by reals or integer numbers, for instance, depending on one's purpose. As an example, if I set that $i \in \mathbb{N}$ represents the moment when I apply a logical rule, $i +_t 1$ corresponds to the next step, no matter when this step is performed in the real life. Thus, the structures for representing space and time can be two parameters of the @-logic. In this way, if $s \in \mathbb{N}$ is the current bus or train station, or the airport, $s +_s 1$ can be the next, for instance.

In this article, the space-time logic is based on a five-valued logic introduced here, without computational concerns such as decidability[93] and complexity[10], nor proof search in backwards[59]. This calculus combines natural deduction with sequent calculus although this yields redundancies: the exclusion rules of the deduction correspond to the left rules of that calculus as it is known. Briefly, my main concern is not the computerization of this calculus, but instead, to introduce a language that can be used in the formal semantics of mobile-code programming languages, as well as for philosophical purposes.

For further work, properties of the present logic ought to be studied and addressed in another paper. The referred to properties include soundness, completeness and complexity.

Finally, this article is organized as follows: section 2 introduces the propositional logic. In the whole @-logic, the space and time models are parameters of the logic, i.e. one defines them for his or her specific purpose, as long as they are refinements of sets. Since the @-logic is used here e.g. in the calculus, a plain space and time models are defined here for the referred to level of derivations. In subsection 2.2, I present and discuss the five truth values of the language, while in subsection 2.3 I present some motivating examples. Section 2.4 presents some comparisons with interval temporal logic, by defining the primitives in the @-logic. In subsection 2.5, I interpret some tense logic operators in the current logic. Finally, in subsection 2.7 I consider uncertainty as well as analogy and belief. In terms of proof theory, section 3 introduces a pair of consequence relations, while section 4 directly deals with deduction. Section 6 briefly introduces one way to roughly represent material objects on the move, as well as resources. Finally, section 7 concludes the article. The appendix A shows the language syntax using Backus normal form. In appendix B, I show a classical binary formulation of the space-time logic.

2 A five-valued propositional logic

The whole logic which I am introducing here is based on a five-valued logic, with truth values represented in

$$C \stackrel{def}{=} \{uu, kk, ff, tt, ii\}$$

In the present piece of work, as well as the well known ff and tt values, uu stands for *unknown* or *undefined*, kk stands for (possibly) *known*, while ii stands for *inconsistent*. I choose to work on inconsistency[9] because it often appears in contexts of the real world. In this way, mobile agents, for instance, ought to be able to decide and act even when it recognizes the presence of an inconsistent predicate. I consider that ii is stronger than uu and also stronger than kk in some kind of strict reasoning, but can be weaker than either in some forms of lazy computation. I shall explain the "known value", *kk*, in section 2.2 together with reasons for having five values. In advance here, it suffices to say that *kk* means "some other agent might know the truth value" to necessarily exclude the person who reasons. More generally and intuitively, the meanings of the values are the following:

- *ff*: (My partner and) I know that the value is $false$;
- *tt*: (My partner and) I know that the value is $true$;
- *kk*: I know that the value is either $true$ or $false$, but I do not know which of them. However, my partner might know which of them.
- *uu*: I do not know the Boolean value nor whether or not it is consistent. My partner neither;
- *ii*: (My partner and) I know that there is some inconsistency in the subject and, because of this, we two do not know whether the actual value is $true$ (nor whether the same actual value is $false$).

However, "my partner" here represents another agent, for example. Note that *kk* or *uu*, for instance, can abstractly represent uncertainty in some subset of domain $\mathbb{R}$.

2.1 Semantics, notions of space and time

In many articles on interval temporal logics, time is often represented by using real values where, as time goes by, *the present moment* corresponds to a value which normally increases. For some applications, there can be branches along these lines to represent possible "futures". There are other approaches, such as in [84] that can also be useful for applications, including systems specification, and also to express natural sub-languages by using particular cases of modality. The underlying language here is both the natural language[54] and mathematics whenever it suits well. I do not adopt tense logics here. However, one can easily define some modal operators of tense logic as in section 2.5.

As mentioned above, in this article I adopt a form of representing time by making use of a flow. More precisely, I shall define an algebra[33, 64, 71] that includes notions of time and space. Thus, I define $\mathbb{T} \doteq \mathbb{R}$ (or alternatively $\mathbb{T} \doteq \mathcal{P}(\mathbb{R})$, the power set of $\mathbb{R}$.) as an infinite set for representing temporal moments. A temporal model is a structure of kind $M = \langle \mathbb{T}, <_t, \leq_t, =_t, \neq_t, \geq_t, >_t \rangle$ which is a flow of time with the present five-valued logical connectives or operators, for each proposition p resulting in a value in C, the set of the five truth values. The semantics of the five-valued connectives are described in 2.2. There can be relational operators over time instants (the real numbers or so). Let $a, b, c, d \in \mathbb{T}$. Then,

- $a <_t b$ states *"a happens before b"*;
- $[a, b] <_t [c, d]$ states that *"the interval* $[a, b]$ *happens before the interval* $[c, d]$*"* (that is, $a \leq_t b <_t c \leq_t d$). Other 12 relations of Allen's interval temporal logic[4] can also be used;
- $a =_t b$ states *"a and b happen at the same time"*;
- $a \leq_t b$ states $a <_t b \vee a =_t b$.

Let $Bool \stackrel{def}{=} \{tt, ff\}$. The operators defined in the algebra apply over $\mathbb{T}$. The signature for the above operators is $\mathbb{T} \times \mathbb{T} \longrightarrow Bool$.

Forms of representation for points in space are slightly more complex than in time. I normally consider $\mathbb{T} \doteq \mathbb{R}$ and $\mathbb{S} \doteq \mathbb{R}^3$ when I refer to these notions, and, as notation, $(\forall i)\ s_i \equiv \langle x_i, y_i, z_i \rangle$, with i an index. When more appropriate, I consider $\mathbb{T} \doteq \mathcal{P}(\mathbb{R})$ and $\mathbb{S} \doteq \mathcal{P}(\mathbb{R}^3)$ instead. Letting $Bool \stackrel{def}{=} \{tt, ff\}$, the relational operators in $\mathbb{S}$, namely $<_s : \mathbb{S} \times \mathbb{S} \longrightarrow Bool$, $=_s : \mathbb{S} \times \mathbb{S} \longrightarrow Bool$ and $\leq_s : \mathbb{S} \times \mathbb{S} \longrightarrow Bool$ can be defined as follows:

$$\begin{aligned} s_i <_s s_j \stackrel{def}{=} & \sqrt{x_i^2 + y_i^2 + z_i^2} < \sqrt{x_j^2 + y_j^2 + z_j^2} \vee \\ & \sqrt{x_i^2 + y_i^2 + z_i^2} = \sqrt{x_j^2 + y_j^2 + z_j^2} \wedge \\ & (x_i < x_j \vee x_i =_s x_j \wedge y_i < y_j \vee x_i = x_j \wedge y_i = y_j \wedge z_i < z_j) \end{aligned}$$

The precedence between coordinates can be, of course, different. Likewise, definition for $<_s$ depends on the application and can have many different ways. However, conjunctions have precedence over disjunctions as indicated in appendix A. The other relational operators are the following:

$$s_i =_s s_j \stackrel{def}{=} x_i = x_j \wedge y_i = y_j \wedge z_i = z_j$$

$$s_i \leq_s s_j \stackrel{def}{=} s_i <_s s_j \vee s_i =_s s_j$$

Definition 2 *Let* $\mathbf{C}$ *be the set of all wffs of @-logic (its language) and* C *be the set of the five truth values* uu, kk, ff, tt, ii. *The corresponding space-time model* m *is a mapping of the type*

$$m : \mathbf{C} \longrightarrow C$$

together with an algebra (Notice that **C** and C are two different symbols). For instance:

$$\begin{aligned} M \stackrel{def}{=} \langle & \mathbb{S}, \mathbb{T}, \mathbf{C}, \\ & <_s, =_s, +_s, -_s, <_t, =_t, +_t, -_t, \\ & \cup, \cap, \setminus, ..., \\ & \neg, \ominus, \wedge, \&, \vee, \otimes, ... \rangle \end{aligned}$$

with, informally, the following semantics: $M \models @s \cdot t[\varphi]$ for $s \in \mathbb{S}$ and $t \in \mathbb{T}$, meaning the value of $\varphi \in \mathbf{C}$ at place s and time t, where $\mathbf{C}$ corresponds to all formulae in the @-logic. I also interpret $@ >_s s \cdot t[\varphi]$ and $@s\cdot >_t t[\varphi]$, or alternatively $@s <_s \cdot t[\varphi]$ and $@s \cdot t <_t [\varphi]$ respectively, as shorthands for $(\exists s' \in \mathbb{S}, s' >_t s)\ @s' \cdot t[\varphi]$ and $(\exists t' \in \mathbb{T}, t' >_t t)\ @s \cdot t'[\varphi]$, respectively, including the binding of the respective variable, and the constraints that the variables, namely s' and t', are chosen in such a way that they are not used in φ, or alternatively one renames variables in φ in a similar way. Accordingly, I interpret $@ <_s s \cdot t[\varphi]$ and $@s\cdot <_t t[\varphi]$, or alternatively $@s >_s \cdot t[\varphi]$ and $@s \cdot t >_t [\varphi]$ respectively, as shorthands for $(\exists s' \in \mathbb{S}, s' <_s s)\ @s' \cdot t[\varphi]$ and $(\exists t' \in \mathbb{T}, t' <_t t)\ @s \cdot t'[\varphi]$, respectively, including the same observation and constraints. That is, as well as the corresponding bindings, this relational form of a formula in the @-logic is not interpreted as universally quantified formula but instead as existentially quantified one.

Considering the hypothesis that the world was created at time B (the Big Bang, for instance) and that the future will always exist, I can express this consideration by the formulae $(\exists B \in \mathbb{T})\ (\forall t \in \mathbb{T})\ B \leq_t t$ and $(\forall t_1 \in \mathbb{T})\ (\exists t_2 \in \mathbb{T})\ t_1 <_t t_2$, respectively. Because I am adopting $\mathbb{T} \doteq \mathbb{R}$, $\mathbb{S} \doteq \mathbb{R}^3$ here, I should consider the continuum property:

$$(\forall t_0, t_1 \in \mathbb{T})\ t_0 <_t t_1 \leftrightarrow (\exists t \in \mathbb{T})\ t_0 <_t t \wedge t <_t t_1$$

$$(\forall s_0, s_1 \in \mathbb{S})\ s_0 <_s s_1 \leftrightarrow (\exists s \in \mathbb{S})\ s_0 <_s s \wedge s <_s s_1$$

And for the same reason, for applications where space and time are linear, both are also linear in the above algebra M. The following properties are also in the present algebra:

$$(\forall x, y \in \mathbb{S})\ x <_s y \underline{\vee} x =_s y \underline{\vee} x >_s y$$

$$(\forall x, y \in \mathbb{T})\ x <_t y \underline{\vee} x =_t y \underline{\vee} x >_t y$$

If one wants to capture the idea of alternative pasts, x and y, in some theory atop the @-logic, we write the following:

$$(\exists x, y \in \mathbb{T})\ \neg(y <_t x) \wedge \neg(y =_t x) \wedge \neg(x <_t y)$$

and/or the following, in some theory atop the @-logic, for representing alternative futures, x and y:

$$(\exists x, y \in \mathbb{T})\ \neg(y <_t x) \wedge \neg(y =_t x) \wedge \neg(x <_t y)$$

Or, alternatively and more succinctly, if one wants only one past

$$x <_t \otimes \wedge y <_t \otimes \rightarrow x <_t y \vee x =_t y \vee x >_t y$$

and/or no alternative futures

$$x >_t \otimes \wedge y >_t \otimes \rightarrow x <_t y \vee x =_t y \vee x >_t y$$

Clearly, this kind of choice depends not only on the application but also on the speaker's intention, and this does not necessarily include the philosophical issue of fate versus free will. For here, in this article, like the way in which one may represent space, it suffices to adopt only one straight line for representing time, i.e. $(\forall x, y \in \mathbb{T})\ x <_t y \vee x =_t y \vee x >_t y$ while I keep the present logic general.

As regards the operators $+_s$ and $-_s$, they can be defined precisely for each theory, whereas $+_t$ and $-_t$ are normally interpreted as follows: $t_1 +_t t_2$ represents the sum of a duration t_2 to a time moment t_1 and the expression results in another time moment; and $t_1 -_t t_2$ represents the interval duration *equivalent* to the duration from a time moment t_2 to another moment t_1. The operators $+_t$ and $-_t$ can be defined in a different way elsewhere, but here these definitions suffice. For the calculus that I define in section 4, for example, the temporal expression $t +_t 1$ simply indicates *the next step, if the inference is in the forward direction, otherwise the previous step.*

2.2 The five values

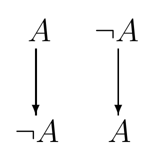
This section opens with the definition of the ontic and strongest five-valued equivalence in the following way:

$\doteq$	u	k	f	t	i
u	t	f	f	f	f
k	f	t	f	f	f
f	f	f	t	f	f
t	f	f	f	t	f
i	f	f	f	f	t

and similarly for discrimination: $A \neq B = \neg(A = B)$. In the @-logic, $=$ is the same as $\doteq$ and we use both interchangeable in this text.

In this section, using the formal syntax defined in the appendix A, I explain a hierarchy of veracity. There may be at least two kinds of *unknown*: "unknown because one does not know the value in the problem domain" (uu) or, alternatively, "unknown because the value is inconsistent" (ii). Thus, in comparison with other logics such as Belnap's, while ii may be interpreted as "the inconsistent value", the present uu and ii are not actually opposite values as uu is the opposite of kk. In fact, there are two views and sets of the connectives, ontic and epistemic. The present work is epistemic and the logic also deals with the concepts of true and false as usual. While $\{\neg, \wedge, \vee\}$ are more ontic operators, $\{\ominus, \&, \parr\}$ are more epistemic ones. To simplify the language during the presentation, I shall refer to them as "ontic" and "epistemic" connectives or operators, although this classification is relative, as well as I am using "connectives" and "operators" with the same meaning, for any reasoning. Thus, $kk \doteq \ominus kk$ and $uu \doteq \ominus uu$, i.e., both formulae are evaluated as *true* whereas $\neg kk \doteq uu$ and $kk \doteq \neg uu$ are valid and make use of the @-logic ontic negation.

For propagating inconsistency, I state $\ominus ii \leftrightarrow ii$, which means that, using sense of humor, "if a formula is inconsistent let alone its negation". The ontic negation of ii would be the value "consistent", which is absent from the logic, for I do not regard this consistency value as interesting for my purpose. I can demonstrate that $\ominus ii \leftrightarrow ii$ makes sense:



In the above illustration, there exist two inconsistent formulae whose negation is represented by one arrow. Notice that there is no involved conjunction nor disjunction, and now I can represent the following structure of truth:

f -	1 knows that sth is false.
t -	1 knows that it is true.
u -	1 does not know because 1 does not have enough knowledge.
i -	1 does not know because it obtains inconsistency.
k -	1 knows, although I do not know which in $\{ff, tt\}$.
kf -	1 knows f because 2 knows that it is false.
kt -	1 knows t because 2 knows that it is true.
ku -	1 knows u because 2 does not know because 2 does not

have enough knowledge.

ki - 1 knows i because 2 does not know because 2 sees inconsistency.

kk - 1 knows the value because 2 knows, but I do not know which.

kkf - 1 knows f because 2 knows f because 3 knows that it is false.

kkt - 1 knows t because 2 knows t because 3 knows that it is true.

kku - 1 knows u because 2 knows u because 3 does not know because 3 does not have enough knowledge.

and so on.

The numbers can indicate machines, for instance. In this setting, I am assuming that numbers do not tell lies. Here, communication is suggested in a sequential form of truth value. For example, to represent that, at time t_0, the machine m_1 knows the veracity of A via machine m_2 and, later, at time t, the communication from the machine m_2 to the machine m_1 was not letting m_1 know the veracity of A for certain reasons, we can briefly state this possibility as follows:

$$@\exists \cdot t_0[@m_1 \cdot \exists[A] \doteq kk \,\&\, (@m_2 \cdot \exists[A] \doteq kk \looparrowright @m_1 \cdot \exists[A] \doteq kk)] \,\&$$
$$(\exists t \in \mathbb{T})\ t >_t t_0 \& \langle t \rangle (@\exists \cdot t[@m_2 \cdot \exists[A] \doteq kk] \,\&\, @\exists \cdot t[@m1 \cdot \exists[A \doteq uu]])$$

I introduce a few implications, e.g. $\looparrowright$ used above, the five-valued intuitionist logic implication, in section 3. For such logics, there exist many truth tables that can be interpreted as an implication, some stronger than others, and I also introduce the implication $\vdash$, which is a more general and weaker one with all the necessary properties and, therefore, capable of supporting the axioms and rules entirely. Thus, in the above example, in the presence of $\exists$ symbol, at the outer level of this formula, I am only concerned about time while, at the inner level, I am only concerned about places.

In general, a truth value is in the form $k^\lambda\gamma$ where λ is a natural number that indicates the number of occurrences of k, and γ is a letter in $\{\mathrm{u}, \mathrm{k}, \mathrm{f}, \mathrm{t}, \mathrm{i}\}$. As a syntax sugar, we can also consider the form γ^{k+1}, e.g. kkkkt = ttttt and kkku = uuuu. For example, in [41], the general form for k is explicitly indexed by the entity that has the knowledge. Although this form in [41] is statically more general, here space and time can be considered instead, which can be used as such indexes together with nesting formulae.

While, for a person $person$, we state "according to $\langle\, person \,\rangle$", the above hierarchy of truths allows the representation of any indirect knowledge of this simple kind. There exists a simplified and static view of the present five values considering only the last letter as a truth value.

I now introduce the connectives of the first set as follows:

a	¬a
u	k
k	u
f	t
t	f
i	i

∧	u	k	f	t	i
u	u	u	f	u	u
k	u	k	f	k	i
f	f	f	f	f	f
t	u	k	f	t	i
i	u	i	f	i	i

∨	u	k	f	t	i
u	u	k	u	t	i
k	k	k	k	t	k
f	u	k	f	t	i
t	t	t	t	t	t
i	i	k	i	t	i

a	(a)Ł
u	u
k	i
f	t
t	f
i	k

→	u	k	f	t	i
u	k	k	k	t	k
k	u	k	u	t	i
f	t	t	t	t	t
t	u	k	f	t	i
i	i	k	i	t	i

↔	u	k	f	t	i
u	u	u	k	u	i
k	u	k	u	k	i
f	k	u	f	f	i
t	u	k	f	t	i
i	i	i	i	i	i

The Ł negation (notation Ł after Łukasiewicz) is left for further work, for those who want to exploit a different feature of the @-logic. Because of this, in the present piece of work, I do not make references to the latter negation, Ł, despite its relevance. $A \leftrightarrow B \stackrel{def}{=} (A \rightarrow B) \wedge (B \rightarrow A)$ is not usual, and this operator is not used in this article either. $\rightarrow$ is placed here only for a better comparison with the following set. The connective $\wedge$ is commutative, associative and has a neutral element, tt. The $\vee$ is commutative, associative and has a neutral element, ff. For the equality connective that I shall define, both De Morgan's laws, $\neg(A \vee B) = \neg A \wedge \neg B$ and $\neg(A \wedge B) = \neg A \vee \neg B$, as well as both absorption laws, $A \vee (A \wedge B) = A$ and $A \wedge (A \vee B) = A$, hold in accordance with my automatic verifications. Furthermore, distributive laws: $A \vee B \wedge C = (A \vee B) \wedge (A \vee C)$ and $A \wedge (B \vee C) = A \wedge B \vee A \wedge C$ are valid. Note that the present logic binds conjunctions tighter than disjunctions, in accordance with appendix A. The more epistemic connectives are the following:

a	⊝a
u	u
k	k
f	t
t	f
i	i

&	u	k	f	t	i
u	u	u	f	u	i
k	u	k	f	k	i
f	f	f	f	f	i
t	u	k	f	t	i
i	i	i	i	i	i

⅋	u	k	f	t	i
u	u	k	u	t	i
k	k	k	k	t	i
f	u	k	f	t	i
t	t	t	t	t	i
i	i	i	i	i	i

↠	u	k	f	t	i
u	u	k	u	t	i
k	k	k	k	t	i
f	t	t	t	t	i
t	u	k	f	t	i
i	i	i	i	i	i

↞↠	u	k	f	t	i
u	u	u	u	u	i
k	u	k	k	k	i
f	u	k	t	f	i
t	u	k	f	t	i
i	i	i	i	i	i

As the results of $A \wedge B$ and $A \vee B$ are the same as $A\&B$ and $A⅋B$, respectively, when $A \neq ii \ \wedge \ B \neq ii$, one can collapse both conjunctions and both disjunctions above in another four-valued logic by dropping *ii* and redefine a four-valued implication and equivalence, if we are sure that there is no inconsistency.

In the logic shown above, a possible interpretation for the operators is with respect to the knowledge on the operands of an arbitrary operation, typically in a programming language context. If one or more values are *ff* or *tt*, the connective gives the corresponding intuitive negation, as above. As an example, two of the tables above can be interpreted as permitting strict and lazy evaluations, if we are a little careful in order to avoid confusion. For instance, *kk* & *uu* can mean that, in a strict evaluation, the first operand is known and that the second one (or, alternatively, the same one) is completely unknown, whereas $kk \vee uu$ can mean the knowledge on the first operand value or no knowledge on the value of the second (or, alternatively, the same) operand in a lazy evaluation. Thus, in accordance with the tables, the first evaluation yields an unknown result whereas the second (lazy) evaluation yields a known result. Here I consider that conjunction and disjunction are commutative connectives. There are other interpretations using these tables. The connective & is commutative, associative and has a neutral element, *tt*. The ⅋ is commutative, associative and has a neutral element, *ff*. De Morgan's laws hold with the negations: $\neg(A⅋B) = \neg A\&\neg B$, $\neg(A\&B) = \neg A⅋\neg B$, $\ominus(A⅋B) = \ominus A\& \ominus B$, $\ominus(A\&B) = \ominus A⅋ \ominus B$. Furthermore, $A⅋B\&C = (A⅋B)\&(A⅋C)$ and $A⅋(B\&C) = (A⅋B)\&(A⅋C)$ is one more important property. However, because my purpose is to propagate *ii* here, in contrast with the first scheme, $A \invamp (A \& B) = A$ and $A \& (A \invamp B) = A$ are not tautologies. While $\rightarrow$ and $\leftrightarrow$ are more ontic, whereas symbols such as $\twoheadrightarrow$ and $\twoheadleftrightarrow$ are more epistemic. While the ontic connectives can be seen as lazy, the epistemic connectives can be seen as strict. $A, B \in \{ff, tt, uu, kk, ii\}$, i.e. for two logical formulae or operands, the first implication can be defined as $A \rightarrow B = \neg A \vee B$ whereas $A \twoheadrightarrow B = \neg A \invamp B$. Furthermore, $A \leftrightarrow B = ((A \rightarrow B) \wedge (B \rightarrow A))$ whereas $A \twoheadleftrightarrow B$, a very epistemic equivalence, cannot be defined in this brief way.

For comparison, I present the tables in Belnap four-valued logic. I present the tables below without implication and equivalence, for Belnap did not show them[11], and because of his work on entailment. His n value (none) corresponds to this uu value (u in my truth tables here), the b value (both) roughly corresponds to this kk value (k in my truth tables here). On the other hand, for helping comparisons, I add the i value to Belnap logic, and the usual properties are still valid between the three connectives, with some exceptions, e.g. $A \wedge ff = ff$ and $A \vee tt = tt$ no longer hold. I shall refer to the resulting five-valued scheme as Belnap-based five-valued logic. The tables become as follows:

a	¬a
u	k
k	u
f	t
t	f
i	i

∧	u	k	f	t	i
u	u	f	f	u	i
k	f	k	f	k	i
f	f	f	f	f	i
t	u	k	f	t	i
i	i	i	i	i	i

∨	u	k	f	t	i
u	u	t	u	t	i
k	t	k	k	t	i
f	u	k	f	t	i
t	t	t	t	t	i
i	i	i	i	i	i

Belnap Four-Valued Logic joined with ii

In [62], in chapter 2, Gupta and Belnap illustrate with schemes for two, three and four values. For the scheme with four values, they present the above conjunction but with the same negation as $\ominus$, except that I have one additional value, ii. Therefore, both the present $\neg$ and $\ominus$ are in fact relatively old connectives and exist since seventies, in the last century. Briefly, the key difference between my truth tables and Belnap's is $uu \wedge kk = ff$ in his tables, i.e. one difference between the @-logic and Belnap four-valued logic is that, while his $A \wedge B$ results in ff for A having value uu and B having value kk, this operation with these values results in uu in the @-logic. The other table results are exactly the same.

In the present five-valued logic, a formula is a *tautology* if and only if it results in tt for all models. Similarly, a formula is a *contradiction* if and only if it results in *ff* for all models. A formula is a *contingency* if and only if the following holds: there exists some model from which the formula produces value *tt* and there exists some model from which the formula produces value *ff*. The present classification is not mutually exclusive. Obviously, since I assume that the world is naturally consistent, a formula of the @-logic is said to be *consistent-valued* if and only if it does not result in *ii* in any model, and *unknown-valued* if and only if it results in *uu* in a model. A formula is *(possibly) known-valued* if and only if one of the following two holds: either it is a contingency and results in *kk* in the set of all models, or results in *kk* in all models. In this article, the number of previous occurrences of k in that complex and sequential truth values can be implicitly represented by nesting space-time references, allowed by the logic grammar.

Even for an established logic, I consider that, if I initially define an equivalence connective independent from the implication, and define the implication as e.g. $A \Rightarrow B \stackrel{def}{=} \neg A \vee B \vee (A \Leftrightarrow B)$, a more general form is obtained. For a version of five-valued implication that has the properties of a classical logic, including the third-middle law, the truth table is the following:

$\multimap$	u	k	f	t	i
u	t	k	k	t	k
k	u	t	u	t	u
f	t	t	t	t	t
t	u	k	f	t	i
i	t	t	t	t	t

However, the formulae $A \wedge B \multimap A$, $A \& B \multimap A$, $A \multimap A \vee B$ and $A \multimap A \invamp B$, like the implications introduced above, are not tautologies for $\multimap$. On the other hand, a deontic logic can be informally conceived in the following fashion: let φ be a formula of the @-logic and $\ominus\varphi$ denote obligation on φ, $\vartriangle \varphi$ denote permissibility on φ. Accordingly, $\neg \ominus \varphi \doteq \vartriangle \neg\varphi$ and $\ominus\neg\varphi \doteq \neg \vartriangle \varphi$. I than combine such modalities with the epistemic values, e.g. "one does not know φ if and only if he or she does not know whether φ is obligatory (or whether φ is permissible)." etc. Finally, the definitions in this paragraph suffices for modality, while other authors can extend the present set of rules with other more specific rules. Such modal operators are welcome to @-logic. While $\Diamond$ represents possibility, $\Box$ does not represent necessity in the real world, but instead sureness. The rules correspond to the implications $A \vdash \Diamond A$ and $\Box A \vdash A$ in Gentzen's style.

I shall introduce in a due course yet another implication symbol, $\looparrowright$, which has the properties of the intuitionistic logic, according to a well-known scheme that I reproduce below, with some adaptation.Furthermore, the notion of ii might be interesting in other contexts, including when one speaks regarding space or time, for instance, one can choose a hotel or a restaurant: one thinks "this one is suitable, that one is not so". $A \wedge \neg A$ is a particular case of $A \wedge B$ and, hence, not inconsistent for us as a principle, but inconsistency, and not necessarily falsity or contradiction, might temporarily appear in the inference, e.g. if someone refers to a larger place (or time) with the same set of hotels or restaurants. I shall identify inconsistency in the @-logic as follows: $B \Rightarrow \neg A$ and $B \Rightarrow A$. I represent that A is inconsistent in the @-logic as $A \doteq ii$.

2.3 Examples

In this section I present examples in the proposed language, that is formalized in appendix A. Here, I attempt to demonstrate the suitability of the @-logic as a language for representing knowledge and belief.Thus, as an example, "day" and "night" are taken as vague variables, where ignorance and inconsistency may arise in some possible form of knowledge representation. Note that words whose first letter is lower-case can be variables or predicates, except space or time variables which can be in either case. Words whose first letter is upper-case can be constants or time or space variables.

The sun always shines everywhere:

$$@\forall \cdot \forall[shines(Sun)]$$

John is working now:

$$@\exists \cdot \otimes[isworking(John)]$$

Marry is now traveling from London to York:

$$(\exists p, f \in \mathbb{T})\ @London \cdot p <_t \otimes[is(Mary)] \wedge @\exists \cdot \otimes[travels(Mary)] \\ \wedge @York \cdot f >_t \otimes[\Diamond is(Mary)]$$

All women were girls at some time in the past:

$$@\exists \cdot \otimes[woman(x)] \rightarrow (\exists t <_t \otimes)\ @\exists \cdot t[girl(x)]$$

In addition to the pair of modal $\Box$ and $\Diamond$, one can define others[36]. Operators concerning space, such as:

$$[s]p \;=\; @\forall \cdot t[p] \qquad for\ some\ moment\ of\ time\ t.$$

$$\langle s \rangle p \; = \; \neg \; [s] \neg p$$

...and for time, such as:

$$[t]p \; = \; @s \cdot \geq_t \otimes [p] \qquad \textit{restricted to some particular place } s.$$

$$\langle t \rangle p \; = \; \neg [t] \; \neg p$$

John is standing up:

$$@s \cdot <_t \otimes [issitdown(John)] \wedge \langle t \rangle isstandup(John)$$

It is day in Brazil iff it is night in Japan:

$$(\forall t) \; @Brazil \cdot t[day] \doteq @Japan \cdot t[night]$$

If someone orders a book, within five days he or she will receive it:

$$(\forall b, s) \; @s \cdot t[orders(b)] \rightarrow @s \cdot \leq_t t +_t 5d[receives(b)]$$

If someone loses his or her passport, he or she will not receive it within 15 days:

$$(\forall p, s, t) \; @\exists \cdot t[looses(s, p)] \rightarrow @\exists \cdot <_t t +_t 15d[\neg receives(s, p)]$$

Another example is a well-known puzzle explained by Melvin Fitting in [41] which I briefly recast here in the logic including space and time. Then, a child can be represented as space relative to time, although in general this does not seem to be a very appropriate way of representing a person's knowledge. In the puzzle, the mother talks to both logician children, here denoted by A and B, asking some questions to them over time from t_0 to t_7:

t_0: "At least one of you have a muddy forehead".
t_1: - both children listen.
t_2: "Does either of you know whether your own forehead is muddy?"
t_3: - both children think.
t_4: - nobody answers.
t_5: "Does either of you know whether your own forehead is muddy?"
t_6: - both children think.
t_7: - both children answer "Mine is".

Firstly, the following rules can be stated

$$(\forall t, t' \in \mathbb{T}, t \leq_t t')\ @x \cdot t[\varphi] \doteq tt \rightarrow @x \cdot t'[\varphi] \doteq tt$$

$$(\forall t, t' \in \mathbb{T}, t \leq_t t')\ @x \cdot t[\varphi] \doteq f\!f \rightarrow @x \cdot t'[\varphi] \doteq f\!f$$

that express monotonicity with respect to knowledge over the time, i.e. that both children's knowledge was not forgotten after having learnt. Here I present an ordinary reasoning without paying attention to details etc. Adopting $m(A)$ and $m(B)$ as propositions stating "A's head is muddy" and "B's head is muddy", respectively, one can represent either children's knowledge (A represents the child who thinks whereas B represents the opposite child) as follows, according to what their mother says and what the children see:

$$(m(A) \vee m(B)) \wedge m(B) \wedge @A \cdot t_1[m(A) \doteq kk] \wedge$$
$$@A \cdot t_4[@B \cdot t_4[m(B) \doteq kk]\ ] \wedge @A \cdot t_6[m(A)] \wedge @B \cdot t_6[m(B)]$$

I further may express that what one says is understood by the others immediately, that is, $(\forall i)\ @S \cdot t_i[speak(W)] \rightarrow @S' \cdot t_{i+1}[W]$, and to represent a truth state known by both children we have the following rule, $x1$:

$$x1: \qquad (\forall A \in \mathbb{S})\ \varphi \rightarrow @A \cdot t[\varphi]$$

which is also a consequence of rule $\forall sw\mathcal{R}$ of the calculus, present in section 4.3.3. Furthermore, I need a rule in the micro-theory as follows

$$x2: \qquad \neg m(A) \rightarrow @B \cdot t_3[m(B)]$$

which states that if B had seen that A's forehead was not muddy, B would have replied "Mine is" at t_4. But since nobody replied, one can use *modus ponens* at t_6 over the contrapositional form of rule $x2$ as well as applying $x1$, and from

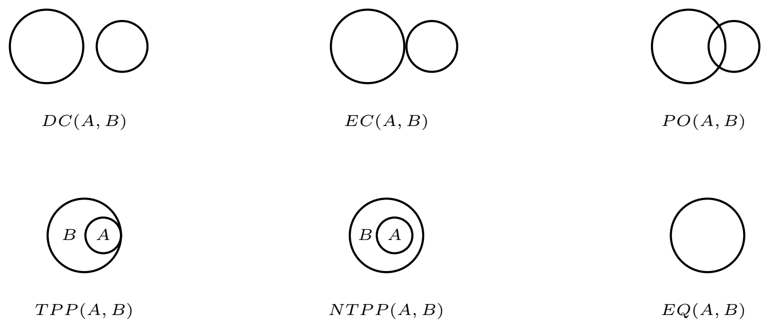
$$\neg @B \cdot t_6[m(B)] \rightarrow\ (m(A) \rightarrow @A \cdot t_6[m(A)])$$

we can infer that

$$@A \cdot t_6[m(A)]$$

2.4 RCC and Interval Temporal Logics

The RCC-8 region-region predicates[23] are the following:



Together with two inverse predicates that are definable from the predicates TPP and $NTPP$.

Considering RCC and only circular regions as a simplification of mine, although in the original paper the predicates are generic and the regions are circular as a choice for their diagrams, I define the primitives of the RCC-8 region-region relations in the @-logic as follows:

- A disconnected from B: $DC(A, B) \stackrel{def}{=} @A \cdot \exists[p] \wedge @B \cdot \exists[q] \wedge (A \cap B = \emptyset)$

- A externally connected to B: $EC(A, B) \stackrel{def}{=} @A \cdot \exists[p] \wedge @B \cdot \exists[q] \wedge (A \cup B \neq A \wedge A \cup B \neq B \wedge A \cap B \neq \emptyset \wedge (a, b \in A \cap B \rightarrow a = b))$. Note that this definition uses a property of circles. The following definition makes use of a related property. In turn, the subsequent two definitions are based on the present predicate.

- A partially overlapping B: $PO(A, B) \stackrel{def}{=} @A \cdot \exists[p] \wedge @B \cdot \exists[q] \wedge (A \cup B \neq A \wedge A \cup B \neq B \wedge (\exists a, b \in A \cap B)\ a \neq b)$

- A tangential proper part of B: $TPP(A, B) \stackrel{def}{=} @A \cdot \exists[p] \wedge @B \cdot \exists[q] \wedge (A \subset B \wedge (\exists C)\ EC(A, C) \wedge EC(B, C))$

- A non-tangential proper-part of B: $NTTP(A, B) \stackrel{def}{=} @A \cdot \exists[p] \wedge @B \cdot \exists[q] \wedge (A \subset B \wedge (\forall C)\ EC(B, C) \rightarrow DC(A, C))$

- A equals B: $EQUALS(A, B) \stackrel{def}{=} @A \cdot \exists[p] \wedge @B \cdot \exists[q] \wedge (A = B)$

Once again, the inverse relations are definable from the above. Non-circular regions can be defined with the notion of a border as a subset for each region. Notation: let $s, A \in \mathcal{P}(\mathbb{S})$, two spaces. Thus the notation $s.b$, $s.b \subset s$, is used to denote the border of s with property $(\forall A)\ DC(A, s.b) \leftrightarrow A \subseteq (s \setminus s.b) \veebar DC(A, s)$. Thus, the relations for different regions can be defined as follows, and I write the predicates in lower-case letters, as this is the standard in this article:

$$dc(A,B) \stackrel{def}{=} dc(A.b, B.b) \wedge A \cap B = \emptyset$$
$$ec(A,B) \stackrel{def}{=} ec(A.b, B.b) \wedge A \cap (B \setminus B.b) = \emptyset \wedge B \cap (A \setminus A.b) = \emptyset$$
$$po(A,B) \stackrel{def}{=} (A \setminus A.b) \cap (B \setminus B.b) \neq \emptyset \wedge A \cup B \neq A \wedge A \cup B \neq B$$
$$tpp(A,B) \stackrel{def}{=} A \subset B \wedge A.b \cap B.b \neq \emptyset$$
$$ntpp(A,B) \stackrel{def}{=} A \subset B \wedge A.b \cap B.b = \emptyset$$

In addition to the following deductive axioms, I can make deductions taking into consideration that the above relations, including $equals(A,B)$, undefined above, are complete and mutually exclusive. In other words, if a region is not related to another by five of these relations, then it is related by the remaining relation.

The above predicates can be used in space expressions, and, for some work in the near future, I shall combine them with $\{uu, kk, ff, tt, ii\}$ and with $\{\cup, \cap, \setminus\}$. Difference from a set to another will also be omitted here in this article in the @-logic. I am now going to present the following scheme for theories more refined than the @-calculus, with exportation, according to my interpretation, which is different from [42]. Thus, for $\perp$ meaning classical falsity,

(1) : $dc(A,B) \rightarrow dc(B,A)$ (commutativity)
(2) : $dc(A,B) \rightarrow (tpp(C,B) \rightarrow dc(A,B))$
(3) : $dc(A,B) \rightarrow (ntpp(C,B) \rightarrow dc(A,B))$
(4) : $ec(A,B) \rightarrow ec(B,A)$
(5) : $ec(A,C) \rightarrow (tpp(C,B) \rightarrow (dc(A,B) \rightarrow \perp))$
(6) : $ec(A,C) \rightarrow (tpp(B,C) \rightarrow (dc(A,B) \vee ec(A,B)))$
(7) : $ec(C,D) \rightarrow (tpp(C,A) \rightarrow (tpp(D,B) \rightarrow ec(A,B)))$
(8) : $ec(A,C) \rightarrow (ntpp(B,C) \rightarrow dc(A,B))$
(9) : $ec(A,C) \rightarrow (ntpp(C,B) \rightarrow \neg dc(A,B))$
(10) : $ec(A,C) \rightarrow (ntpp(C,B) \rightarrow po(A,B))$
(11) : $po(A,B) \rightarrow po(B,A)$ (commutativity)
(12) : $po(A,C) \rightarrow (tpp(C,B) \rightarrow po(A,B))$
(13) : $po(A,C) \rightarrow (ntpp(C,B) \rightarrow po(A,B))$
(14) : $tpp(C,A) \rightarrow (tpp(C,B) \rightarrow$
$(A \neq B \rightarrow po(A,B) \vee tpp(A,B) \vee tpp(B,A) \vee ntpp(A,B) \vee ntpp(B,A)))$
(15) : $tpp(C,A) \rightarrow (ntpp(C,B) \rightarrow$
$(A \neq B \rightarrow po(A,B) \vee tpp(A,B) \vee tpp(B,A) \vee ntpp(A,B) \vee ntpp(B,A)))$
(16) : $ntpp(C,A) \rightarrow (tpp(C,B) \rightarrow$

$(A \neq B \to po(A,B) \vee tpp(A,B) \vee tpp(B,A) \vee ntpp(A,B) \vee ntpp(B,A)))$
$(17): \ ntpp(C,A) \to (ntpp(C,B) \to$
$(A \neq B \to po(A,B) \vee tpp(A,B) \vee tpp(B,A) \vee ntpp(A,B) \vee ntpp(B,A)))$
$(18): \ tpp(A,B) \to (tpp(B,C) \to tpp(A,C) \vee ntpp(A,C))$
$(19): \ tpp(A,B) \to (ntpp(B,C) \to ntpp(A,C))$
$(20): \ ntpp(A,B) \to (tpp(B,C) \vee ntpp(B,C) \to ntpp(A,C))$
$(21): \ ntpp(A,B) \to (po(A,C) \to po(C,B) \vee tpp(C,B) \vee ntpp(C,B))$
$(22): \ dc(A,B) \to (ec(A,B) \to \perp)$
$(23): \ dc(A,B) \to (po(A,B) \to \perp)$
$(24): \ dc(A,B) \to (tpp(A,B) \to \perp)$
$(25): \ dc(A,B) \to (ntpp(A,B) \to \perp)$
$(26): \ ec(A,B) \to (po(A,B) \to \perp)$
$(27): \ ec(A,B) \to (tpp(A,B) \to \perp)$
$(28): \ ec(A,B) \to (ntpp(A,B) \to \perp)$
$(29): \ po(A,B) \to (tpp(A,B) \to \perp)$
$(30): \ po(A,B) \to (ntpp(A,B) \to \perp)$
$(31): \ tpp(A,B) \to (ntpp(A,B) \to \perp)$
$(32): \ tpp(A,B) \to (ntpp(B,A) \to \perp)$
$(33): \ tpp(A,B) \to (tpp(B,A) \to \perp)$
$(34): \ ntpp(A,B) \to (ntpp(B,A) \to \perp)$
$(35): \ ntpp(A,B) \to (ntpp(B,C) \to ntpp(A,C))$ (nttp transitivity)
$(36): \ po(A,C) \to (tpp(C,B) \to (dc(A,B) \to \perp))$
$(37): \ po(A,C) \to (ntpp(C,B) \to (dc(A,B) \to \perp))$
$(38): \ tpp(A,C) \to (tpp(C,B) \to (dc(A,B) \to \perp))$
$(39): \ tpp(A,C) \to (ntpp(C,B) \to (dc(A,B) \to \perp))$
$(40): \ (dc(A,B) \to \perp) \to (nttp(A,C) \to (ec(B,C) \to \perp))$
$(41): \ po(B,C) \to (tpp(A,B) \to ((tpp(C,A) \to \perp) \wedge (ntpp(C,A) \to \perp)))$
$(42): \ po(B,C) \to (ntpp(A,B) \to ((tpp(C,A) \to \perp) \wedge (ntpp(C,A) \to \perp)))$
$(43): \ tpp(A,B) \to (tpp(C,B) \to (ntpp(A,C) \to \perp))$
$(44): \ ntpp(A,B) \to (tpp(C,B) \to ((tpp(C,A) \to \perp) \wedge (ntpp(C,A) \to \perp)))$

I omit relations whose consequent is in the negative form, such as in axiom (9) as an example. There are other more complex axioms or rules for the above scheme.

Concerning interval temporal logics[2], there are 13 relations, six of which are obtained from the six other relations, and one relation for equality. In the present logic syntax and semantics, I can build those relations by defining temporal predicates:

- A before B: $BEFORE(A,B) \stackrel{def}{=} @\exists\cdot[a_0,a_f][A]\&@\exists\cdot[b_0,b_f][B]\&a_f <_t b_0$

- A meets B: $MEETS(A,B) \stackrel{def}{=} @\exists \cdot [a_0, a_f][A] \& @\exists \cdot [b_0, b_f][B] \& a_f =_t b_0$
- A overlaps B: $OVERLAPS(A,B) \stackrel{def}{=} @\exists \cdot [a_0, a_f][A] \& @\exists \cdot [b_0, b_f][B] \& a_0 <_t b_0 <_t a_f <_t b_f$
- A starts B: $STARTS(A,B) \stackrel{def}{=} @\exists \cdot [a_0, a_f][A] \& @\exists \cdot [b_0, b_f][B] \& a_0 =_t b_0 \& a_f <_t b_f$
- A contains B: $CONTAINS(A,B) \stackrel{def}{=} @\exists \cdot [a_0, a_f][A] \& @\exists \cdot [b_0, b_f][B] \& a_0 <_t b_0 \& b_f <_t a_f$
- A ends B: $ENDS(A,B) \stackrel{def}{=} @\exists \cdot [a_0, a_f][A] \& @\exists \cdot [b_0, b_f][B] \& a_0 >_t b_0 \& a_f =_t b_f$
- A equals B: $EQUALS(A,B) \stackrel{def}{=} @\exists \cdot a[A] \& @\exists \cdot b[B] \& a =_t b$

Thus, by using the above relations for time and space (for Region Connection Calculus is very similar in a sense), there is a unifying predicate theory involving both notions formally.

As an example of representation of action or events, some information in a domain that reasons about transportation of cargo can be extracted from [5]:

- The train starts at the originating city S.
- The trip takes between 4 and 6 hours.
- The train will be on track segment $A1$ then it will cross junction $J1$, and be on track segment $A2$ for the rest of the trip.
- The train will end up at destination city D.

Therefore, we can represent the train trip as follows:

- $@S \cdot t[Train]$
- $t_f \in [t +_t 4h, t +_t 6h]$
- $(\forall t_1, t_2, t_3, t \leq_t t_1 <_t t_2 <_t t_3 \leq_t t_f)\ @A1 \cdot t_1[Train] \wedge @J1 \cdot t_2[Train] \wedge @A2 \cdot t_3[Train]$
- $@D \cdot t_f[Train]$

2.5 Application: Tense logic operators

Although I do not adopt tense, I regard these operators as definable in the following way:

$F\varphi \stackrel{def}{=} (\exists t)\ @ \oplus \cdot t >_t \otimes[\varphi]$: φ will be the case;

$P\varphi \stackrel{def}{=} (\exists t)\ @ \oplus \cdot t <_t \otimes[\varphi]$: φ was the case;

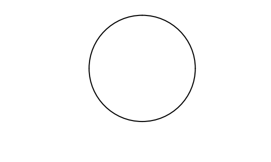
$G\varphi \stackrel{def}{=} (\forall t)\ @ \oplus \cdot t >_t \otimes[\varphi]$: φ is always going to be the case;

$H\varphi \stackrel{def}{=} (\forall t)\ @ \oplus \cdot t <_t \otimes[\varphi]$: φ was always the case.

2.6 Cycles: An Illustration

One can represent time in a somewhat subjective and flexible way using three dimensions. If one chooses a spiral to represent a view of time, one of the orthogonal projections of the spiral on the 2D plane is a circle. Thus time goes by cyclically and here time is represented using an angle. On the other hand, if one views a time line as a wave, one can perceive the infiniteness of it. If one wants to get a straight line one projects the wave uniformly in one of the dimensions.

The subjectivity and flexibility concerning this circle is that one can associate angles to the daily life. Each angle is a fraction of an hour, for example. In another form of representation, the same angle can be the same fraction of a year, for example. Thus one obtains different spirals, each of which represents the current focus of attention. In this way, the observer is included in the notion of time. The time variable represented by a single real number can be easily converted from/to this tuple.



Two Views of the Time Flow

As an idea that is orthogonal to alternative pasts and alternative futures, the above picture informally illustrates two different views for a common notion of time. The choice between them does not change my formalism, except for a few operators on angles, as they will be left here.

Thus, there can also be an alternative form of representation as follows: a pair of real numbers where the second element is the angle while the first element is the number of complete circles before that angle (the correspondence between one circle and one unit of temporal notion that belongs to the real world is implicit, e.g. one hour or week or year or other). Let π represent here the well-known transcendental number, which corresponds to the ratio of the circumference to its diameter. For $\langle \alpha, \beta \rangle$ where $\alpha \in \mathbb{Z}$ and $\beta \in [0, 2\pi)$, the straight line representation for the time is

$$t = 2\pi \times_t \alpha +_t \beta$$

and thus one feels free to choose when to use α and when to use β, or both (t).

2.7 Analogy, Belief and Uncertainty

Although computer scientists do not normally think on certainty factors while writing the semantics of a programming language or writing some other academic pieces of work, sometimes scientists want to express uncertainty over propositions. Thus, because the expressiveness of the @-logic is an approach or mine, in this section, I introduce a notation that can capture analogy, induction, belief as well as some models of uncertainty. This notation is orthogonal to the rest of the logic and analogy and belief can be seen as orthogonal notions with respect to uncertainty, that is, these notions can help each other in the expressiveness of the language.

If φ is a formula in @-logic, then $\bigcirc\varphi$ is a formula in @-logic that means "φ is believed to be true". Properties: $\neg \bigcirc \varphi \doteq \bigcirc\neg\varphi$, $\ominus \bigcirc \varphi \doteq \bigcirc \ominus \varphi$. Knowledge is not dual to belief, and that pseudo duality does not hold among the known modal logics. Both $\neg \bigcirc \varphi \wedge \varphi$ and $\neg \bigcirc \varphi \wedge \neg\varphi$, as well as $\neg \bigcirc \varphi \wedge \bigcirc\neg\varphi$, are often acceptable in natural languages. Moreover, $\bigcirc \bigcirc \varphi$ seems to be more uncertain belief than $\bigcirc\varphi$, but that is subjective.

'Analogy' is both a feature and a process of reasoning based on similar features, when two objects are compared. Here I use the second meaning. Intuitively, I also understand that analogy is an instantaneous form of synthetic reasoning based on intuition or a kind of personal perception, and, because of this, I prefer not to define the semantics in a universal way. Instead, I only standardize its symbol and syntax in the @-logic.

Thus, let φ_1 and φ_2 be two formulae. Then, to express that φ_1 is analogous to φ_2 I write $\varphi_1 \bowtie \varphi_2$, or, alternatively for analogy is commutative, one may also write $\varphi_2 \bowtie \varphi_1$. $\bowtie$ is probably commutative, associative and distributive even over other operators. Although analogy is clearly commutative, I do not do this, for I consider that analogy is personal. I have the same attitude towards the following properties, usual for other operations:

Reflexive:	$\varphi_1 \bowtie \varphi_1$
Symmetric:	$\varphi_1 \bowtie \varphi_2 \rightarrow \varphi_2 \bowtie \varphi_1$
Transitive:	$\varphi_1 \bowtie \varphi_2 \wedge \varphi_2 \bowtie \varphi_3 \rightarrow \varphi_1 \bowtie \varphi_3$

It is reasonable to state that $\bowtie$ is reflexive and symmetric, but the transitive property does not universaly hold. Because of this, I leave the issue of analogy half open, although analogy can be used together with uncertainty.

For induction, a sequence of formulae separated by comma with the "..." symbol as its sufix indicates induction over the formulae. This induction is not mathematical induction, but instead a non-valid form of reasoning that humans make use in their lives. If a student submits his or her PhD thesis to a panel, the panel will make the final decision based on this kind of induction as follows:

$$E_1, E_2, ... \rightarrow Result$$

where $E_i,\ i \in \mathbb{N}$, indicates any list of examiners and *Result* indicates the result that the student obtains. To stress the importance of induction, any practice of democracy is based on this form of reasoning, although mathematically invalid.

For uncertainty, I initially have to define the truth values as a subset of $\mathbb{R}$: in this article, $\mathbb{P} = \{x \in \mathbb{R} :\ -1.0 \leq x \leq +1.0\}$ plays this rôle. I use the Greek letter ψ to denote an uncertainty formula, i.e. a ψ-formula, e.g. $\psi(n\ \varphi)$ for some formula φ, where n is a pair $\langle x, y\rangle : \mathbb{P} \times \mathbb{P}$ where $x, y \in \mathbb{P}$ are the certainty thresholds, *false* and *true* respectively, for $\psi(n\ \varphi)$. The variables of the pair $n = \langle x, y\rangle$ are individually denoted as $n.x$ and $n.y$ respectively. In any case, n and φ allow the valuation system to know whether $\psi(n\ \varphi)$ is *true* or *false*, for example, where $n.x \leq n.y$. In the present language, this result is in $\{ff, tt, uu, kk, ii\}$ in accordance with n, φ and a few rules. In advance, as a simplified and informal example, the formula $\psi(\langle 0, 0.5\rangle\ \varphi)$ indicates that if φ has certainty degree greater than or equal to 0.5, the formula is interpreted as tt. On the other hand, if φ has certainty degree less than 0, the formula is interpreted as ff. Otherwise, its resulting value is uu. This certainty degree will be defined later in this section.

It is well known that, for more complex systems, the notation of n as well as the interval can be different, e.g. $[0, 1]$ is normally used for representing probability[74].

With respect to the nature of veracity, it is easy to observe, for example, that a car is German, and write the proposition "the car is German". Then another person can look at the car and easily state "that is true". This may happen because the nature of the information that can be represented is in a sense objective (all that one needs to do is to recognize the German company).

Or, alternatively, the simplification that one makes on the information from the real world converts a naturally subjective piece of information, given the natural complexity of the world, to another objective piece of information in a corresponding manner, e.g. one can still ask "yes, but how much in that car is German?" even if the answer happens to be 100%, as the world can be seen as fuzzy[68]. In this piece of work, I do not state the uncertainty explicitly in any rule of the deductive system presented below, because that is a matter of vocabulary in that purely deductive context. As long as there is a mapping without loss of generality, the more simplified the language the clearer its concepts and constructs. Different values and different certainty thresholds can be assigned to different views of the same object, person etc in the real world, and that is because the truth value of φ results from the subjective nature of some object. The use of the ψ letter in the present article is for suggesting a more subjective nature of human factors. A solely fuzzy view of the universe is in [68].

With respect to the present sub-model of uncertainty, a truth value is a pair of values, $\langle v, n\rangle$, where v is a value in $\{ff, tt, uu, kk, ii\}$ and n is another pair $\langle \alpha, \omega\rangle$ of values in $\mathbb{P}$, where α means minimum and ω means maximum, i.e. $\alpha \leq \omega$ (otherwise, the formula is said to be inconsistent). Thus, if V is some logical value on uncertainty, I simplify this notation by making use of $V.\alpha$ and $V.\omega$ to denote this pair of values. For a formula $\psi(m\ \varphi)$, if φ is evaluated and results in $\langle v, n\rangle$ here, or simply $\varphi = \langle v, n\rangle$ as a value of a model, the resulting value of the whole evaluation of $\psi(m\ \varphi)$ is one of the resulting values as follows:

- $\langle uu, n\rangle$ iff $n.\alpha \geq m.x \wedge n.\omega < m.y \wedge m.x < 0 \leq m.y \wedge n.\alpha \leq n.\omega$
- $\langle kk, n\rangle$ iff $(n.\alpha < m.x \vee n.\omega \geq m.y) \wedge m.x < 0 \leq m.y \wedge n.\alpha \leq n.\omega$
- $\langle ff, n\rangle$ iff $n.\omega < m.x \wedge m.x < 0 \leq m.y \wedge n.\alpha \leq n.\omega$
- $\langle tt, n\rangle$ iff $n.\alpha \geq m.y \wedge m.x < 0 \leq m.y \wedge n.\alpha \leq n.\omega$
- $\langle ii, \langle 1, -1\rangle\rangle$, otherwise.

For example, two main interpretations for uu, and two main interpretations for kk, can be made: in the first and second items above, the evalutation system regards the certainty degrees as public. For an interpretation with private certainty degrees, I simply consider the following cases:

- $\langle uu, \langle 0, 0\rangle\rangle$ iff $n.\alpha \geq m.x \wedge n.\omega < m.y \wedge m.x < 0 \leq m.y \wedge n.\alpha \leq n.\omega$
- $\langle kk, \langle 0, 0\rangle\rangle$ iff $(n.\alpha < m.x \vee n.\omega \geq m.y) \wedge m.x < 0 \leq m.y \wedge n.\alpha \leq n.\omega$

To allow the evaluation of φ as a pair I provide a notation for an uncertain formula. Thus, if φ is a (possible ψ-) formula, the evaluation of an expression $\varphi?\beta$ is in accordance with the following:

$$\langle v, \langle \beta \times \gamma.\alpha, \beta \times \gamma.\omega \rangle \rangle \text{ if } \varphi \text{ results in } \langle v, \gamma \rangle$$

Here I define some helpful constructs, bearing in mind that they are optional, by making use of the meta-level predicate of the form $@\forall \cdot t[\![\varphi]\!]$ to stand for "the meaning of the formula φ at time t", in this case in the type $C \times (\mathbb{P} \times \mathbb{P})$. I first formalize the construct $\varphi\ ?\ \beta$ (β is its certainty factor), as explained before:

$$\frac{@\forall \cdot t_1[\![\varphi]\!] = \langle v, \gamma \rangle}{@\forall \cdot t_2[\![\varphi\ ?\ \beta]\!] = \langle v, \langle \beta \times \gamma.\alpha, \beta \times \gamma.\omega \rangle \rangle}$$

where $t_1 <_t t_2$, and this condition also holds throughout this section.

I still need a few more words on implication. I support the idea that deductive logics are not capable of capturing a really relevance implication, despite Belnap's fantastic and historical work on relevance logic among others. I observe that words such as "because" have a conjunctive component, for example, if one says "Ann is using an umbrella because it is raining" is different from *if it rains Ann uses an umbrella*. In the former, the person who states indicates four important and conjunctive ideas, in addition to the context: that Ann is using umbrella, that it is raining, her *awareness* that it is raining, and her intention, which depends on cause-effect relationships. There are some natural-language conditionals that, to be regarded as valid, indicate that both the antecedent and consequent are $false$[69]. Moreover, in the real world and using the natural language, we normally have causes before their consequences, often with a particular time interval as a constraint between actions and/or events, and it may happen that the relationship is not clear. On the one hand, logics is a study of reasoning in one of its broadest senses. On the other hand, temporal relations, which might seem to be information outside logics, may be in the core of the meaning of conditional, implication and/or entailment in the same sense of logics. Therefore, what is the truth and logical meaning for such ordinary implications? I regard that probably exist some stronger implications in comparison to others. Uncertainty is a more general tool for defining connectives and the importance of certainty factors in natural-language inferences seems to be clear. Although I do not define implication as $\neg A \vee B$, the rules for implication between two uncertainty formulae can be as follows:

Intuitively, for two formulae A, B, and constant $n \in \mathbb{R}$ ($0 \leq n \leq 1$), a *conditional* can be formalized as a 3-tuple of the form $\langle A, B, n \rangle$, and suggested notation $A \xrightarrow{n} B$, together with the following property:

- Awereness of A happens before the awareness of B;
- If $A \xrightarrow{m} B$ and $B \xrightarrow{n} C$, we obtain $A \xrightarrow{m\times n} C$

with the following semantics:
first and second conditionals

$$\frac{@\forall \cdot t_1[\![A]\!] = \langle v_1, m_1\rangle \quad @\forall \cdot t_1[\![B]\!] = \langle v_2, m_2\rangle \quad v_1, v_2 \in \{f\!f, tt\}}{@\forall \cdot t_2[\![A \xrightarrow{n} B]\!] = \langle v_1 \rightarrow v_2, \langle m_1.\alpha \times m_2.\alpha \times n, m_1.\omega \times m_2.\omega \times n\rangle\rangle}$$

third conditional

$$\frac{@\forall \cdot t_1[\![A]\!] = \langle v_1, m_1\rangle \quad @\forall \cdot t_1[\![B]\!] = \langle v_2, m_2\rangle \quad v_1 = v_2 = f\!f}{@\forall \cdot t_2[\![A \xrightarrow{n} B]\!] = \langle v_1 \rightarrow v_2, \langle m_1.\alpha \times m_2.\alpha \times n, m_1.\omega \times m_2.\omega \times n\rangle\rangle}$$

for the case e.g. "if Ann had studied less she would not pass the exam" where both the antecedent and consequent are interpreted as false for the conditional be valid. This interpretation for conditionals are only two suggestions.

As well as *modus ponens*, entailment is useful for applications such as expert systems. For being sufficiently general, it ought to have a certainty factor attached. As an example of interpretation,

$$\frac{\begin{array}{c} v_1, v_2 \in \{uu, kk, tt\} \quad @\forall \cdot t_1[\![A]\!] = \langle v_1, \beta_1\rangle \\ @\forall \cdot t_1[\![A \xrightarrow{\beta_2} B]\!] = \langle tt, \langle +1.0, +1.0\rangle\rangle \quad @\forall \cdot t_1[\![B]\!] = \langle v_2, \beta_3\rangle \end{array}}{@\forall \cdot t_2[\![B]\!] = \langle v_1 \vee v_2, \langle max(\beta_1.\alpha \times \beta_2, \beta_3.\alpha), max(\beta_1.\omega \times \beta_2, \beta_3.\omega)\rangle\rangle}$$

and symmetrically for $\{uu, kk, f\!f\}$ (although with some redundancy) and, finally, one rule that propagates ii.

Another known operation may be called *composition*, which simulates some form of inductive reasoning. The operations above make use of the *min* and *max* functions, but there are contexts in which more than one formula together should increase certainty. The more the pieces of evidence, the greater the confidence should be. MYCIN[87] was the first expert system that used a similar idea. Let the 2-ary ϕ be the following auxiliary function:

$$\phi(x_1, x_2) = \begin{cases} x_1 + (1 - x_1) \times x_2 & \text{if } 0 \leq x_1, x_2 \leq +1, \\ x_1 + x_2 & \text{if } x_1 < 0 \wedge x_2 \geq 0 \vee x_1 \geq 0 \wedge x_2 < 0, \\ x_1 + (1 + x_1) \times x_2 & \text{if } -1 \leq x_1, x_2 < 0; \end{cases}$$

The definition and syntax are: φ_1 and φ_2 are two formulae if and only if $\{\psi : \varphi_1, \varphi_2\}$ is an uncertainty formula called composition. More generaly, if φ_1 is a formula, then $\{\psi : \varphi_1, \Gamma\}$ is an uncertainty formula called composition, where Γ is a non-terminal symbol which denotes a sequence of formulae in the object (final) language. The semantics for composition is in accordance with the following rules using ϕ:

Given the A and B formulae,

$$\frac{@\forall \cdot t_1[\![A]\!] = \langle v_1, \beta_1 \rangle \quad @\forall \cdot t_1[\![B]\!] = \langle v_2, \beta_2 \rangle}{@\forall \cdot t_2[\![\ \{\psi : A, B\}\ ]\!] = \langle tt,\ \langle \phi(\beta_1.\alpha, \beta_2.\alpha), \phi(\beta_1.\omega, \beta_2.\omega)\rangle\rangle}$$

For more than two formulae,

$$\frac{@\forall \cdot t_1[\![A]\!] = \langle v_1, \beta_1 \rangle \quad @\forall \cdot t_1[\![\{\psi : \Gamma\}]\!] = \langle v_2, \beta_2 \rangle}{@\forall \cdot t_2[\![\ \{\psi : A, \Gamma\}\ ]\!] = \langle tt,\ \langle \phi(\beta_1.\alpha, \beta_2.\alpha), \phi(\beta_1.\omega, \beta_2.\omega)\rangle\rangle}$$

For aall truth values in form $\langle v, \beta \rangle$, the certainty degree is simply obtained as follows:

$$?\langle v, \beta \rangle = f(\beta) = \frac{\beta.\alpha + \beta.\omega}{2}$$

where f is a locally defined symbol.

From now on I shall not make explicit use of uncertainty, except to introduce a number of examples in the next paragraphs. Instead, I assume that uncertainty can be implicit, as stated above, for all formulae. The axioms and rules of the deductive system or the @-calculus deal with the final value, i.e. I simply use the value in $\{uu, kk, ff, tt, ii\}$. That is, if $\psi(a\ A)$ results in $\langle v, \alpha \rangle$ where $v \in \{ff, tt, uu, kk, ii\}$, then v is simply used instead.

2.8 A Few More Examples

I present some examples that make use of knowledge representation with uncertainty.

If one tosses d (a one-dollar coin) on a table T, one obtains 50% of probability of getting head after 10 seconds:

$$@T \cdot t[toss(d)] \rightarrow @T \cdot t +_t 10s[get(head)?0]$$

If the patient had x but now the test of x presents a degree of certainty less than 0.1, the patient does not have x.

$$@s\cdot <_t \otimes[has(patient, x)] \wedge @s \cdot \otimes[diagnose(patient, x) = y \wedge y < 0.1] \\ \rightarrow \neg @s\cdot \geq_t \otimes[has(patient, x)]$$

As another example, for a person who knows the Boolean value of A but does not know the Boolean value of B, the formula $A \rightarrow B$ might result in tt or uu, say, in equal probabilities. Thus, we can write the following propositions:

- $val(A)$ indicates the value of A.
- $val(B)$ denotes the value of B.
- $imp(A, B, r)$ represents that the result from $A \to B$ is r.
- $prob(x, y)$ denotes formula x with probability y.

In the @-logic, we can write as follows:

$$val(A) = kk \wedge val(B) = uu \to \\ prob(imp(A, B, tt), 0.5) \wedge prob(imp(A, B, uu), 0.5)$$

The above example does not require uncertainty. However, observe the following: there is one diagnosis d and three symptoms s_1, s_2, s_3 with certainty factors 0.3, 0.6 and *unknown*, respectively, together with one rule with more details:

$$\langle tt, 0.3\rangle \to s_1 \qquad \langle tt, 0.6\rangle \to s_2 \qquad \langle uu, 0\rangle \to s_3 \\ \psi(\langle -0.5, +0.9\rangle \ \{\psi : s_1?0.8,\ s_2?0.5,\ s_3?0.3\}\ ) \to d$$

3 Sequents

In [48], Gabbay states a scheme for a *linear logic* in Hilbert style and using the classical implication symbol:

Identity:	$A \Rightarrow A$
Commutativity:	$(A \Rightarrow (B \Rightarrow C)) \Rightarrow (B \Rightarrow (A \Rightarrow C))$
Prefixing:	$(C \Rightarrow A) \Rightarrow ((B \Rightarrow C) \Rightarrow (B \Rightarrow A))$
Suffixing:	$(C \Rightarrow A) \Rightarrow ((A \Rightarrow B) \Rightarrow (C \Rightarrow B))$

The *relevance logic*[7, 81] is based on the schema above plus

$$(A \Rightarrow (B \Rightarrow C)) \Rightarrow ((A \Rightarrow B) \Rightarrow (A \Rightarrow C))$$

The *intuitionistic logic* is based on the relevance logic scheme plus

$$A \Rightarrow (B \Rightarrow A)$$

and, finally, by adding the following schema

$$((A \Rightarrow B) \Rightarrow A) \Rightarrow A.$$

to the previous one, we obtain the schema for *classical logic*. The original approach on the @-logic was to choose the calculus for one of the above logics, and then correct problems. There are rules of inference that are specific for the values other than *true* and *false*.

In this way, if I let A, B be formulae, the axioms in the classical logic $(A \Rightarrow B) \Rightarrow ((A \Rightarrow (B \Rightarrow ff)) \Rightarrow (A \Rightarrow ff))$ and $A \Rightarrow ((A \Rightarrow ff) \Rightarrow B)$ had not been tautologies if I would want to propose a paraconsistent and relevance logic[32], together with some extra rules in the calculus. The latter axiom is also the sixth axiom above, that one which complements the scheme for intuitionistic logic.

In the present logic, there is not a single notion of contradiction as a primitive because there are five values (including *ff* and *ii*) and two different consequence relations: weak and strong. The proposal of a pair of two consequence relations is probably a novelty.

An implication with the properties of the above scheme for intuitionistic logic is the following:

$\looparrowright$	u	k	f	t	i
u	t	k	f	t	i
k	u	t	f	t	i
f	t	t	t	t	t
t	u	k	f	t	i
i	u	k	f	t	t

In advance, the $\looparrowright$ implication, as well as most implications, do not support entirely the @-calculus, only a few rules. However, I introduce a weaker implication for the present calculus that has the properties of the classical scheme as well as makes the rules tautologies with the first tables (i.e. the connectives $\{\neg, \wedge, \vee\}$. $\vdash$ is also tautological for the truth tables of the second scheme if there is no inconsistency in the calculus presented later), for the principle of contraposition does not need to hold in the above schemes.

$\vdash$	u	k	f	t	i
u	t	t	t	t	t
k	t	t	t	t	t
f	t	t	t	t	t
t	u	k	f	t	i
i	t	t	t	t	t

and now principles such as *simplification*, $A \wedge B \vdash A$ or $A \& B \vdash A$ for example, hold with the schemes presented above. However, because of ii, $A \vdash A \parr B$ is not a tautology where $A = tt$ and $B = ii$, although both $(A \vdash B) \vdash (A \vdash B \parr C)$ and $(A \vdash C) \vdash (A \vdash B \parr C)$ are tautologies. Furthermore, $A \vdash A \vee B$, for example, is a tautology. Thus, for the present deductive system together with the @-calculus, I propose the above definition of $\vdash$.

Space can be seen as an abstract notion whereas a sequent in the calculus is also an expression of one of the following alternative forms

- $\Delta \vdash C$ iff both time and space can be represented implicitly, or
- $@s \cdot t_0[\Delta] \vdash @s \cdot t_1[C]$ or, equivalently, $@\Delta \cdot [t_0, t_1][C]$, as long as s is not used in Δ, C, t_0 or t_1.

In the second form, s and $[t_0, t_1]$ explicitly state the space and time where and when the expression $\Delta \vdash C$ takes place, respectively. Thus, here, there is no mobile derivation. Further, I can simplify my notation here by writing $@s \cdot t[\Delta \vdash C]$ or $@\Delta \cdot t[C]$ that indicates that a derivation makes use of the assumption Δ; starts at some time in which I am not interested, and finishes at time t.

The @-logic has another consequence relation, $\Vdash$. While $\vdash$, also called weak sequent, yields weak proof, $\Vdash$ (the strong sequent) yields strong proof. $\vdash$ and $\Vdash$ yield derivations. Weak and strong proofs may form a pair of novelties. Thus, $(\Delta \Vdash A) \stackrel{def}{=} (\Delta \vdash A) \& \neg(\Delta \vdash \neg A)$.

4 Deduction

In this section I initially concentrate on derivations. Let A be a formula in the present language. As usual, a proof for A here is a tree of steps from a set of valid assumptions (the leaves) that leads us to conclude that the logical formula A is *true* (the root) for all values in any model. On the other hand, a *derivation* is a more general notion. It does not imply that the assumed formulae and the final formula are valid.

The @-calculus works as follows: there is a set of assumed formulae and one final formula, where each variable or formula can have one of the five values presented here: $\{ff, tt, uu, kk, ii\}$.

Deductions are based on axioms and rules of inference. A Rule is a meta-level implication and here I assume the @-logic $\vdash$ implication to follow their semantics. As usual, I also represent rules of inference by using fractional notation, where

$$\frac{@\Delta_1 \cdot t_1[C_1] \quad @\Delta_2 \cdot t_2[C_2] \quad ... \quad @\Delta_n \cdot t_n[C_n]}{@\Delta_1, \Delta_2, ..., \Delta_n \cdot t[C]}$$

corresponds to, at a higher level,

$$@\Delta_1 \cdot t_1[C_1] \wedge @\Delta_2 \cdot t_2[C_2] \wedge ... \wedge @\Delta_n \cdot t_n[C_n] \vdash @\Delta_1 \cup \Delta_2 \cup ..., \Delta_n \cdot t[C]$$

I use comma instead of the $\cup$ set operation as I use multi-sets.

Here, I introduce the properties of the present calculus.

Reflexivity: $@s \cdot t[\Delta, \{C\} \vdash C]$ which captures inclusion: $C \in \Delta \rightarrow (\forall s \in \mathbb{S}, t \in \mathbb{T})\ @s \cdot t[\Delta \vdash C]$, another property.

Monotonicity:

$$\frac{@\Delta \cdot t[C]}{@\Delta, \Gamma \cdot t +_t 1[C]}$$

Premise Commutativity:

For any rule, since time is relevant here, I include an axiom for exchanging premisses considering the evaluation time as follows:

$$\frac{@\Delta \cdot t[A] \quad @\Gamma \cdot t +_t 1/2[B]}{@\Delta, \Gamma \cdot t +_t 1[C]} \doteq \frac{@\Gamma \cdot t[B] \quad @\Delta \cdot t +_t 1/2[A]}{@\Delta, \Gamma \cdot t +_t 1[C]}$$

As usual, the premises are also associative.

The cut rule is computationally redundant, as demonstrated in a theorem by Gentzen[1].

For the @-logic, I regard that derivations have place and also work as time goes by.

As an example of notation for interpretation of derivation, the rule for the introduction of the implication operator is shown:

$$\frac{\begin{array}{c} @s \cdot t_1[A] \\ \vdots \\ @s \cdot t[C] \end{array}}{@s \cdot t +_t 1[A \rightarrow C]}$$

where $t_1 <_t t$. In the above case, I assume A, obtain C and deduce $A \rightarrow C$. That is, this holds because the rule is sound with respect to the five-valued truth table for the implication symbol (here, if the premise is tt, the conclusion is tt).

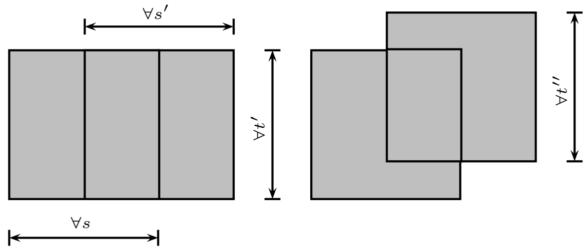
Here I am assuming that derivations do record the sequence of applications of the rules in such a way that the sequence of steps is the output from its computation

In both structural and logical rules, the time is placed explicit, every rule should be stated only as implication.

Negative and Positive Wholeness

In natural language, both space and time can be referred to to mean existentially or with wholeness. So far, expressions of the @-logic form, such as $@s \cdot t[\varphi]$, do not allow deductions, unless I am more precise in the space or the time expressions. Alternatively, a more practical terminology for formalization is set: positive and negative wholeness, respectively.

In the following picture on the left, the rectangles will represent the scopes of two formulae, $@s \cdot t[A]$ and $@s' \cdot t[A]$ with the same time (ordinate) and a common place (abscissa). On the right, a more general situation: two formulae with a little common space and a little common time.



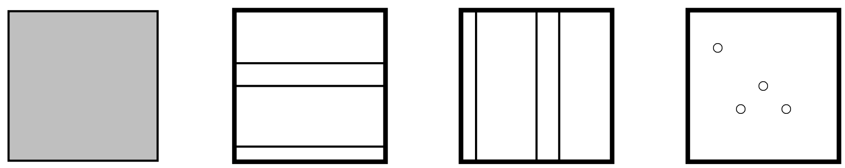
To indicate that the place (or time) indicates the negative wholeness, one places the symbol ∃ before the space (or time) expression. Accordingly, one places ∀ before the space (or time) expression to indicate that the notion is positive. Although $@s \cdot t[\varphi]$ without ∃ or ∀ is a valid expression in the @-logic, it does not allow deductions.

As an example of the positive wholeness, on the one hand, if one knows that in the *state* of São Paulo (in Brazil) citizens have the habits h_1, e.g. they like sports, formally $@\forall SP \cdot \exists[habits(p, h_1)]$, one is allowed to deduce that, in Santos city, which in turn is in the state of São Paulo, citizens have habits h_1, that is, $@\forall Santos \cdot \exists[habits(p, h_1)]$. On the other hand, if one understands that in the *state* of São Paulo citizens have the habits h_2, possibly the same ones but with *negative* wholeness, say $@\exists SP \cdot \exists[habits(p, h_2)]$, one is allowed to deduce that, in Brazil, citizens have habits h_2, say $@\exists Brazil \cdot \exists[habits(p, h_2)]$. The expression $@\exists SP \cdot \exists[habits(p, h_2)]$ formally states that it is allowed to deduce that, in South America for instance, or even in America in some sense, there are citizens who have habits h_2. However, by using the last formula, one cannot deduce that, in São Paulo *city*, which is located inside São Paulo state, there are citizens who have habits h_2.

One may argue that, in a formula, the syntactical positions for space and for time are rigid and, therefore, if one wants to combine the positive and negative wholeness, one loses flexibility. That opinion is not a good standpoint, for space and time expressions themselves do not bind variables, and where both indicate that the inner formula is the scope. In this way, the following expression $(\forall s \in \mathbb{S})\ (\exists t \in \mathbb{T})\ @s \cdot t[\varphi]$ and the expression $(\exists t \in \mathbb{T})\ (\forall s \in \mathbb{S})\ @s \cdot t[\varphi]$ are not equivalent, while the expression $@s \cdot \exists[@\exists \cdot t[\varphi]] \doteq @\exists \cdot t[@s \cdot \exists[\varphi]]$ holds. In nesting formulae, the positive wholeness of space or time does not have priority over the negative wholeness and vice-versa. For example, depending on the interpretation, the formula $@\exists s \cdot \forall t[@\forall s \cdot \exists t[\varphi]]$ might not imply $@\forall s \cdot \forall t[\varphi]$. Alternatively, one may want to write $@\exists s \cdot \forall t[\varphi] \wedge @\forall s \cdot \exists t[\varphi]$ instead or, better, $@\forall s \cdot \forall t[\varphi] \wedge @\exists s \cdot \exists t[\varphi]$.

The following picture demonstrates possible combinations of space and time wholeness, respectively,

$@\forall s \cdot \forall t[A]$ $\quad$ $@\forall s \cdot \exists t[A]$ $\quad$ $@\exists s \cdot \forall t[A]$ $\quad$ $@\exists s \cdot \exists t[A]$



4.1 Axioms

Identity:

$$\overline{@\{C\} \cdot t[C]}$$

The other axioms are defined in specific contexts.

4.2 Structural Rules

In this section, I present the structural rules of the @-calculus. Other space-time logics can be obtained by removing some of structural rules[82]. The structural rules almost in Gentzen's style are the following:

Hypothesis:

$$\frac{}{@s \cdot t[\Delta, \{C\} \vdash C]}\mathcal{Y}$$

Here, I use comma instead of the $\cup$ set operation as I use multi-sets. Therefore, this notation does not impose an order between two finite multi-sets of formulae, in such a way that there is no need for the so called *exchange* rule. The contraction rule is the following:

Contraction:

$$\frac{@\Delta, \{A, A\} \cdot t[C]}{@\Delta, \{A\} \cdot t +_t 1[C]}C\mathcal{L}$$

An essay on contraction is [47]. For proof theory without contraction, references, for example, are [16, 17, 59].

Weakening:

$$\frac{@\Delta \cdot t[C]}{@\Delta, \{A\} \cdot t +_t 1[C]}\mathcal{W}$$

Weakening explicitly expresses the monotonicity property.

4.3 Logical Rules

In this section, the logical rules are presented. The rules for $\ominus$, &, ⅋ and $\twoheadrightarrow$ are not presented since the structures of the rules are equivalent to the rules for $\neg$, $\wedge$, $\vee$ and $\rightarrow$, respectively. More than this, rules with $\multimap$ are not presented for the same reason with respect to $\rightarrow$. Therefore, I am going to present rules for the fragment $\{\neg, \wedge, \vee, \rightarrow\}$.

Deduction:

$$\frac{@\Delta \cdot t[A \rightarrow C]}{@\Delta, \{A\} \cdot t +_t 1[C]}\mathcal{D}\uparrow \qquad \frac{@\Delta, \{A\} \cdot t[C]}{@\Delta \cdot t +_t 1[A \rightarrow C]}\mathcal{D}\downarrow$$

Excluded 6th:

$$\frac{\neg @\Delta \cdot t[A \doteq kk] \quad \neg @\Delta \cdot t +_t 1/4[A \doteq ff] \quad \neg @\Delta \cdot t +_t 1/2[A \doteq tt] \quad \neg @\Delta \cdot t +_t 3/4[A \doteq ii]}{@\Delta \cdot t +_t 1[A \doteq uu]}$$

but all sequences of formulae in the premise can appear in any order.

4.3.1 Introduction:

The introduction rules are part of the deduction as well as the calculus.

Conjunction:

$$\frac{@\Delta, \{A\} \cdot t[C]}{@\Delta, \{A \wedge B\} \cdot t +_t 1[C]} \wedge \mathcal{IL}_1 \qquad \frac{@\Delta, \{B\} \cdot t[C]}{@\Delta, \{A \wedge B\} \cdot t +_t 1[C]} \wedge \mathcal{IL}_2$$

$$\frac{@\Delta \cdot t[A] \quad @\Gamma \cdot t +_t 1/2[B]}{@\Delta, \Gamma \cdot t +_t 1[A \wedge B]} \wedge \mathcal{IR}$$

Similarly, for inconsistent deduction:

$$\frac{@\Delta, \{A\} \cdot t[C]}{@\Delta, \{A \doteq ii\} \cdot t +_t 1[C]}\mathcal{I}ii\mathcal{L}_1 \qquad \frac{@\Delta, \{\neg A\} \cdot t[C]}{@\Delta, \{A \doteq ii\} \cdot t +_t 1[C]}\mathcal{I}ii\mathcal{L}_2$$

$$\frac{@\Delta \cdot t[A] \quad @\Gamma \cdot t +_t 1/2[\neg A]}{@\Delta, \Gamma \cdot t +_t 1[A \doteq ii]}\mathcal{I}ii\mathcal{R}$$

Disjunction:

$$\frac{@\Delta, \{A\} \cdot t[C] \qquad @\Gamma, \{B\} \cdot t +_t 1/2[C]}{@\Delta, \Gamma, \{A \vee B\} \cdot t +_t 1[C]} \vee \mathcal{IL}$$

$$\frac{@\Delta \cdot t[A]}{@\Delta \cdot t +_t 1[A \vee B]} \vee \mathcal{IR}_1 \qquad \frac{@\Delta \cdot t[B]}{@\Delta \cdot t +_t 1[A \vee B]} \vee \mathcal{IR}_2$$

Negation:

$$\frac{@\Delta \cdot t[A]}{@\Delta, \{\neg A\} \cdot t +_t 1[ii]} \neg \mathcal{IL} \quad \frac{@\Delta, \{A\} \cdot t[ff]}{@\Delta \cdot t +_t 1[\neg A]} \neg \mathcal{IR}$$

Implication:

$$\frac{@\Delta \cdot t[A] \qquad @\Gamma, \{B\} \cdot t +_t 1/2[C]}{@\Delta, \Gamma, \{A \to B\} \cdot t +_t 1[C]} \to \mathcal{IL} \qquad \frac{@\Delta \cdot t[B]}{@\Delta \cdot t +_t 1[A \to B]} \to \mathcal{IR}$$

4.3.2 Elimination:

The elimination rules are part of the deductive system but not part of the calculus.

Conjunction:

$$\frac{@\Delta, \{A \wedge B\} \cdot t[C]}{@\Delta, \{A, B\} \cdot t +_t 1[C]} \wedge \mathcal{EL}$$

$$\frac{@\Delta \cdot t[A \wedge B]}{@\Delta \cdot t +_t 1[A]} \wedge \mathcal{ER}_1 \qquad \frac{@\Delta \cdot t[A \wedge B]}{@\Delta \cdot t +_t 1[B]} \wedge \mathcal{ER}_2$$

Similarly,

$$\frac{@\Delta, \{A \doteq ii\} \cdot t[C]}{@\Delta, \{A, \neg A\} \cdot t +_t 1[C]} ii\mathcal{EL}$$

$$\frac{@\Delta \cdot t[A \doteq ii]}{@\Delta \cdot t +_t 1[A]} ii\mathcal{ER}_1 \qquad \frac{@\Delta \cdot t[A \doteq ii]}{@\Delta \cdot t +_t 1[\neg A]} ii\mathcal{ER}_2$$

Disjunction:

$$\frac{@\Delta, \{A \vee B\} \cdot t[C]}{@\Delta, \{A\} \cdot t +_t 1[C]} \vee \mathcal{EL}_1 \qquad \frac{@\Delta, \{A \vee B\} \cdot t[C]}{@\Delta, \{B\} \cdot t +_t 1[C]} \vee \mathcal{EL}_2$$

$$\frac{@\Delta \cdot t[A \vee B] \quad @\Delta_1 \cdot t +_t 1/3[A \to C] \quad @\Delta_2 \cdot t +_t 2/3[B \to C]}{@\Delta, \Delta_1, \Delta_2 \cdot t +_t 1[C]} \vee \mathcal{ER}$$

and there also exist the following two rules:

$$\frac{@\Delta \cdot t[A \vee B]}{@\Delta \cdot t[A] \vee @\Delta \cdot t[B]} \vee \mathcal{E} \vee \mathcal{R} \qquad \frac{@\Delta \cdot t[A \vee B]}{@\Delta \cdot t[A] \bindnasrepma @\Delta \cdot t[B]} \vee \mathcal{E} \bindnasrepma \mathcal{R}$$

Negation:

$$\frac{@\Delta, \{\neg A\} \cdot t[\mathit{ff}]}{@\Delta \cdot t +_t 1[A]} \neg\mathcal{EL} \quad \frac{@\Delta, \{A\} \cdot t[\mathit{ff}]}{@\Delta \cdot t +_t 1[\neg A]} \neg\mathcal{ER}$$

Implication:

$$\frac{@\Delta, \{A \to B\} \cdot t[C]}{@\Delta, \{B\} \cdot t +_t 1[C]} \to \mathcal{EL} \qquad \frac{@\Delta \cdot t[A] \quad @\Gamma \cdot t +_t 1/2[A \to C]}{@\Delta, \Gamma \cdot t +_t 1[C]} \to \mathcal{ER}$$

The left rule, above, is not part of the linear logic or relevance logic. The above right rule is what is often called *modus ponens*.

4.3.3 Space and Time:

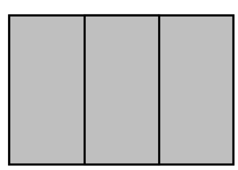
From now on, for the following axioms and deductive rules, $s, s' \subset \mathbb{S}; t', t'' \subset \mathbb{T}$, while t ought to remain as before: $t \in \mathbb{T}$.

$\neg\mathcal{A}1$ $\quad \neg @\forall s \cdot \forall t'[A] \doteq @\exists s \cdot \exists t'[\neg A]$
$\neg\mathcal{A}2$ $\quad \neg @\forall s \cdot \exists t'[A] \doteq @\exists s \cdot \forall t'[\neg A]$
$\neg\mathcal{A}3$ $\quad \neg @\exists s \cdot \forall t'[A] \doteq @\forall s \cdot \exists t'[\neg A]$
$\neg\mathcal{A}4$ $\quad \neg @\exists s \cdot \exists t'[A] \doteq @\forall s \cdot \forall t'[\neg A]$

$\wedge\mathcal{AP}$ $\quad @P \cdot Q[A] \wedge @P \cdot Q[B] \doteq @P \cdot Q[A \wedge B]$
$\vee\mathcal{AP}$ $\quad @P \cdot Q[A] \vee @P \cdot Q[B] \doteq @P \cdot Q[A \vee B]$

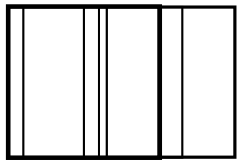
The above axioms can be used as two-way rules. I present the set of rules with an implicit correspondence between uu and empty space:

Space $\cup$ introduction

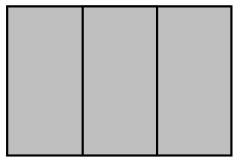


$$\frac{@\Delta, \{@\forall s \cdot t'[A], @\forall s' \cdot t'[A]\} \cdot t[C]}{@\Delta, \{@\forall s \cup s' \cdot t'[A]\} \cdot t +_t 1[C]} \forall s \cup \mathcal{IL}$$

$$\frac{@\Delta \cdot t[@\forall s \cdot \forall t'[A]] \quad @\Gamma \cdot t +_t 1/2[@\forall s' \cdot \forall t'[A]]}{@\Delta, \Gamma \cdot t +_t 1[@\forall s \cup s' \cdot \forall t'[A]]} \forall s \cup \mathcal{IR}$$



or

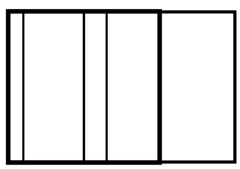


$$\frac{@\Delta, \{@\exists s \cdot t'[A]\} \cdot t[C] \quad @\Gamma, \{@\exists s' \cdot t'[A]\} \cdot t +_t 1/2[C]}{@\Delta, \Gamma, \{@\exists s \cup s' \cdot t'[A]\} \cdot t +_t 1[C]} \exists s \cup \mathcal{IL}$$

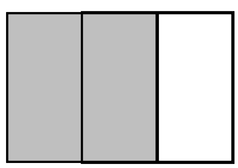
A rule for expansion:

$$\frac{@\Delta \cdot t[@\exists s \cdot t'[A]]}{@\Delta \cdot t +_t 1[@\exists s \cup s' \cdot t'[A]]} \exists s \cup \mathcal{IR}$$

and, for both $\exists s$ and $\forall s$, an alternative and interesting rule for



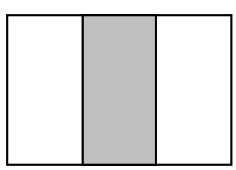
or

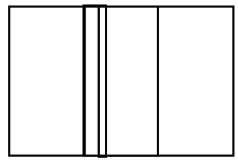


follows:

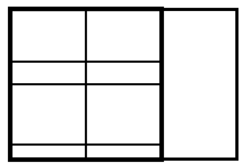
$$\frac{@\Delta \cdot t[@s \cdot \forall t'[A]]}{@\Delta \cdot t +_t 1[@\exists s \cup s' \cdot \forall t'[A]]} s\forall \cup \mathcal{IR}$$

Space $\cap$ introduction

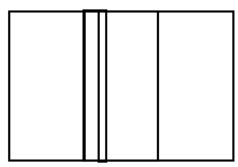




$$\frac{@\Delta, \{@\exists s \cdot t'[A], @\exists s' \cdot t'[A]\} \cdot t[C]}{@\Delta, \{@s \cap s' \cdot t'[A]\} \cdot t +_t 1[C]} s \cap \mathcal{IL}$$



$$\frac{@\Delta \cdot t[@\forall s \cdot t'[A]]}{@\Delta, \Gamma \cdot t +_t 1[@\forall s \cap s' \cdot t'[A]]} \forall s \cap \mathcal{IR}$$

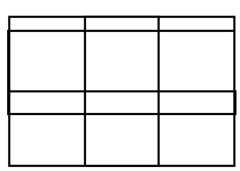


$$\frac{@\Delta, \{@\exists s \cdot t'[A], @\exists s' \cdot t'[A]\} \cdot t[C]}{@\Delta, \{@\exists s \cap s' \cdot t'[A]\} \cdot t +_t 1[C]} \exists s \cap \mathcal{IL}$$

It can be somewhat interesting to observe that the formula $@\exists s \cap s' \cdot t'[A]$ does not follow from $@\exists s \cdot t'[A] \wedge @\exists s' \cdot t'[A]$, but instead from $\forall\exists sw\mathcal{R}$, defined later.

Space ∪ elimination

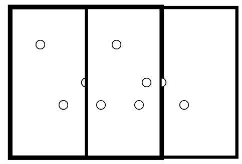
$$\frac{@\Delta, \{@\forall s \cup s' \cdot \forall t'[A]\} \cdot t[C]}{@\Delta, \{@\forall s \cdot \forall t'[A], @\forall s' \cdot \forall t'[A]\} \cdot t +_t 1[C]} \forall s \cup \mathcal{EL}$$



$$\frac{@\Delta \cdot t[@\forall s \cup s' \cdot t'[A]]}{@\Delta \cdot t +_t 1[@\forall s \cdot t'[A]]} \forall s \cup \mathcal{ER}$$

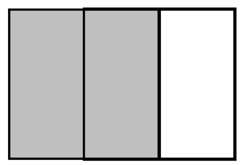
$$\frac{@\Delta, \{@\exists s \cup s' \cdot t'[A]\} \cdot t[C]}{@\Delta, \{@\exists s \cdot t'[A]\} \cdot t +_t 1[C]} \exists s \cup \mathcal{EL}$$

$$\frac{@\Delta, \{@\exists s \cup s' \cdot \forall t'[A]\} \cdot t[C]}{@\Delta, \{@s \cdot \forall t'[A]\} \cdot t +_t 1[C]} s\forall \cup \mathcal{EL}$$

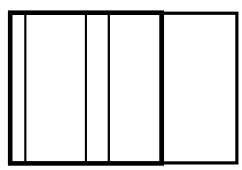


$$\frac{@\Delta \cdot t[@\exists s \cup s' \cdot t'[A]]}{@\Delta \cdot t[@\exists s \cdot t'[A]] \vee @\Delta \cdot t[@\exists s' \cdot t'[A]]} \exists s \cup \mathcal{ER}$$

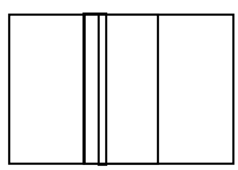
Space ∩ elimination:



$$\frac{@\Delta, \{@\forall s \cap s' \cdot t'[A]\} \cdot t[C]}{@\Delta, \{@\forall s \cdot t'[A]\} \cdot t +_t 1[C]} \forall s \cap \mathcal{EL}$$



$$\frac{@\Delta \cdot t[@\forall s \cap s' \cdot t'[A]]}{@\Delta \cdot t +_t 1[@\exists s \cdot t'[A]]} \forall\exists s \cap \mathcal{ER}$$



$$\frac{@\Delta \cdot t[@\exists s \cap s' \cdot t'[A]]}{@\Delta \cdot t +_t 1[@\exists s \cdot t'[A]]} \exists s \cap \mathcal{ER}$$

Space weakening - At the same time:

$$\frac{@\Delta, \{@\exists s \cup s' \cdot t'[A]\} \cdot t[C]}{@\Delta, \{@\exists s \cap s' \cdot t'[A]\} \cdot t +_t 1[C]} \exists sw\mathcal{L} \qquad \frac{@\Delta \cdot t[@\exists s \cap s' \cdot t'[A]]}{@\Delta \cdot t +_t 1[@\exists s \cup s' \cdot t'[A]]} \exists sw\mathcal{R}$$

$$\frac{@\Delta, \{@\forall s \cap s' \cdot t'[A]\} \cdot t[C]}{@\Delta, \{@\forall s \cup s' \cdot t'[A]\} \cdot t +_t 1[C]} \forall sw\mathcal{L} \qquad \frac{@\Delta \cdot t[@\forall s \cup s' \cdot t'[A]]}{@\Delta \cdot t +_t 1[@\forall s \cap s' \cdot t'[A]]} \forall sw\mathcal{R}$$

Different Spaces and Times
For the following rules, $s \neq s' \wedge t' \neq t''$.

$$\frac{@\Delta, \{@\forall s \cdot \exists t'[A]\} \cdot t[C] \quad @\Gamma, \{@\forall s \cdot \exists t'[A]\} \cdot t +_t 1/2[C]}{@\Delta, \Gamma, \{@\forall s \cup s' \cdot \exists t' \cup t''[A]\} \cdot t +_t 1[C]} \forall s \exists t \cup \mathcal{IL}$$

$$\frac{@\Delta \cdot t[@\forall s \cup s' \cdot \exists t' \cup t''[A]]}{@\Delta \cdot t +_t 1[@\forall s \cdot \exists t'[A] \vee @\forall s \cdot \exists t'[A]]} \forall s \exists t \cup \mathcal{ER}$$

$$\frac{@\Delta, \{@\exists s \cdot \forall t'[A]\} \cdot t[C] \quad @\Gamma, \{@\exists s \cdot \forall t'[A]\} \cdot t +_t 1/2[C]}{@\Delta, \Gamma, \{@\exists s \cup s' \cdot \forall t' \cup t''[A]\} \cdot t +_t 1[C]} \exists s \forall t \cup \mathcal{IL}$$

$$\frac{@\Delta \cdot t[@\exists s \cup s' \cdot \forall t' \cup t''[A]]}{@\Delta \cdot t +_t 1[@\exists s \cdot \forall t'[A] \vee @\exists s \cdot \forall t'[A]]} \exists s \forall t \cup \mathcal{ER}$$

$$\frac{@\Delta, \{@\forall s \cdot \forall t'[A], @\forall s' \cdot \forall t''[A]\} \cdot t[C]}{@\Delta, \{@\forall s \cup s' \cdot \forall t' \cup t''[A]\} \cdot t +_t 1[C]} \forall st \mathcal{IL} \divideontimes$$

$$\frac{@\Delta \cdot t[@\forall s \cdot \forall t'[A]] \quad @\Gamma \cdot t +_t 1/2[@\forall s' \cdot \forall t''[A]]}{@\Delta, \Gamma \cdot t +_t 1[@\forall s \cap s' \cdot \forall t' \cap t''[A]]} \forall st \mathcal{IR} \divideontimes$$

$$\frac{@\Delta, \{@\forall s \cap s' \cdot \forall t' \cap t''[A]\} \cdot t[C]}{@\Delta, \{@\forall s \cdot \forall t'[A], @\forall s' \cdot \forall t''[A]\} \cdot t +_t 1[C]} \forall st \mathcal{EL} \divideontimes$$

$$\frac{@\Delta \cdot t[@\forall s \cup s' \cdot \forall t' \cup t''[A]]}{@\Delta \cdot t +_t 1[@\forall s \cdot \forall t'[A]]} \forall st \mathcal{ER} \divideontimes$$

although $@\forall s \cup s' \cdot \exists t' \cup t''[A]$ does not imply $@\forall s \cdot \exists t'[A]$ or $@\forall s' \cdot \exists t''[A]$. Accordingly, $@\exists s \cup s' \cdot \forall t' \cup t''[A]$ does not imply $@\exists s \cdot \forall t'[A]$ or $@\exists s \cdot \forall t'[A]$.

Time $\cup$ introduction

Since time is symmetric to space, it turns out that to repeat similar diagrams here are unnecessary. Instead, I place the rules without comments.

$$\frac{@\Delta, \{@s \cdot \forall t'[A], @s \cdot \forall t''[A]\} \cdot t[C]}{@\Delta, \{@s \cdot \forall t' \cup t''[A]\} \cdot t +_t 1[C]} \forall t \cup \mathcal{IL}$$

$$\frac{@\Delta \cdot t[@\forall s \cdot \forall t'[A]] \quad @\Gamma \cdot t +_t 1/2[@\forall s \cdot \forall t''[A]]}{@\Delta, \Gamma \cdot t +_t 1[@\forall s \cdot \forall t' \cup t''[A]]} \forall t \cup \mathcal{IR}$$

$$\frac{@\Delta, \{@s \cdot \exists t'[A]\} \cdot t[C] \quad @\Gamma, \{@s \cdot \exists t''[A]\} \cdot t +_t 1/2[C]}{@\Delta, \Gamma, \{@s \cdot \exists t' \cup t''[A]\} \cdot t +_t 1[C]} \exists t \cup \mathcal{IL}$$

$$\frac{@\Delta \cdot t[@s \cdot \exists t'[A]]}{@\Delta \cdot t +_t 1[@s \cdot \exists t' \cup t''[A]]} \exists t \cup \mathcal{IR}$$

$$\frac{@\Delta \cdot t[@\forall s \cdot t'[A]]}{@\Delta \cdot t +_t 1[@\forall s \cdot \exists t' \cup t''[A]]} t\forall \cup \mathcal{IR}$$

Time ∩ introduction

$$\frac{@\Delta, \{@s \cdot \exists t'[A], @s \cdot \exists t''[A]\} \cdot t[C]}{@\Delta, \{@s \cdot t' \cap t''[A]\} \cdot t +_t 1[C]} t \cap \mathcal{IL}$$

$$\frac{@\Delta \cdot t[@s \cdot \forall t'[A]]}{@\Delta, \Gamma \cdot t +_t 1[@s \cdot \forall t' \cap t''[A]]} \forall t \cap \mathcal{IR}$$

$$\frac{@\Delta, \{@s \cdot \exists t'[A], @s \cdot \exists t''[A]\} \cdot t[C]}{@\Delta, \{@s \cdot \exists t' \cap t''[A]\} \cdot t +_t 1[C]} \exists t \cap \mathcal{IL}$$

Time ∪ elimination

$$\frac{@\Delta, \{@\forall s \cdot \forall t' \cup t''[A]\} \cdot t[C]}{@\Delta, \{@\forall s \cdot \forall t'[A], @\forall s \cdot \forall t''[A]\} \cdot t +_t 1[C]} \forall t \cup \mathcal{EL}$$

$$\frac{@\Delta \cdot t[@s \cdot \forall t' \cup t''[A]]}{@\Delta \cdot t +_t 1[@s \cdot \forall t'[A]]} \forall t \cup \mathcal{ER}$$

$$\frac{@\Delta, \{@s \cdot \exists t' \cup t''[A]\} \cdot t[C]}{@\Delta, \{@s \cdot \exists t'[A]\} \cdot t +_t 1[C]} \exists t \cup \mathcal{EL}$$

$$\frac{@\Delta, \{@\forall s \cdot \exists t' \cup t''[A]\} \cdot t[C]}{@\Delta, \{@\forall s \cdot t'[A]\} \cdot t +_t 1[C]} t\forall \cup \mathcal{EL}$$

$$\frac{@\Delta \cdot t[@s \cdot \exists t' \cup t''[A]]}{@\Delta \cdot t[@s \cdot \exists t'[A]] \vee @\Delta \cdot t[@s \cdot \exists t''[A]]} \exists t \cup \mathcal{ER}$$

Time ∩ elimination

$$\frac{@\Delta, \{@s \cdot \forall t' \cap t''[A]\} \cdot t[C]}{@\Delta, \{@s \cdot \forall t'[A]\} \cdot t +_t 1[C]} \forall t \cap \mathcal{EL}$$

$$\frac{@\Delta \cdot t[@s \cdot \forall t' \cap t''[A]]}{@\Delta \cdot t +_t 1[@s \cdot \exists t'[A]]} \forall \exists t \cap \mathcal{ER}$$

$$\frac{@\Delta \cdot t[@s \cdot \exists t' \cap t''[A]]}{@\Delta \cdot t +_t 1[@s \cdot \exists t'[A]]} \exists t \cap \mathcal{ER}$$

Time Weakening - At the same place:

$$\frac{@\Delta, \{@s \cdot \exists t' \cup t''[A]\} \cdot t[C]}{@\Delta, \{@s \cdot \exists t' \cap t''[A]\} \cdot t +_t 1[C]} \exists tw\mathcal{L} \qquad \frac{@\Delta \cdot t[@s \cdot \exists t' \cap t''[A]]}{@\Delta \cdot t +_t 1[@s \cdot \exists t' \cup t''[A]]} \exists tw\mathcal{R}$$

$$\frac{@\Delta, \{@s \cdot \forall t' \cap t''[A]\} \cdot t[C]}{@\Delta, \{@s \cdot \forall t' \cup t''[A]\} \cdot t +_t 1[C]} \forall tw\mathcal{L} \qquad \frac{@\Delta \cdot t[@s \cdot \forall t' \cup t''[A]]}{@\Delta \cdot t +_t 1[@s \cdot \forall t' \cap t''[A]]} \forall tw\mathcal{R}$$

Space-Time $\forall$ and $\exists$:

$$\frac{@\Delta, \{@s \cdot t'[A]\} \cdot t[C]}{@\Delta, \{@\forall \cdot t'[A]\} \cdot t +_t 1[C]}\forall s\mathcal{IL} \quad \frac{@\Delta \cdot t[@\forall \cdot t'[A]]}{@\Delta \cdot t +_t 1[@s \cdot t'[A]]}\forall s\mathcal{ER}$$

$$\frac{@\Delta, \{@s \cdot t'[A]\} \cdot t[C]}{@\Delta, \{@s \cdot \forall[A]\} \cdot t +_t 1[C]}\forall t\mathcal{IL} \quad \frac{@\Delta \cdot t[@s \cdot \forall[A]]}{@\Delta \cdot t +_t 1[@s \cdot t'[A]]}\forall t\mathcal{ER}$$

$$\frac{@\Delta, \{@\exists \cdot t'[A]\} \cdot t[C]}{@\Delta, \{@s \cdot t'[A]\} \cdot t +_t 1[C]}\exists s\mathcal{EL} \quad \frac{@\Delta \cdot t[@s \cdot t'[A]]}{@\Delta \cdot t +_t 1[@\exists \cdot t'[A]]}\exists s\mathcal{IR}$$

$$\frac{@\Delta, \{@s \cdot \exists[A]\} \cdot t[C]}{@\Delta, \{@s \cdot t'[A]\} \cdot t +_t 1[C]}\exists t\mathcal{EL} \quad \frac{@\Delta \cdot t[@s \cdot t'[A]]}{@\Delta \cdot t +_t 1[@s \cdot \exists[A]]}\exists t\mathcal{IR}$$

Wholeness and Nesting Formulae:

$$\frac{@\Delta, \{@\exists s \cdot t'[A]\} \cdot t[C]}{@\Delta, \{@\forall s \cdot t'[A]\} \cdot t +_t 1[C]}\exists\forall sw\mathcal{L} \quad \frac{@\Delta \cdot t[@\forall s \cdot t'[A]]}{@\Delta \cdot t +_t 1[@\exists s \cdot t'[A]]}\forall\exists sw\mathcal{R}$$

$$\frac{@\Delta, \{@s \cdot \exists t'[A]\} \cdot t[C]}{@\Delta, \{@s \cdot \forall t'[A]\} \cdot t +_t 1[C]}\exists\forall tw\mathcal{L} \quad \frac{@\Delta \cdot t[@s \cdot \forall t'[A]]}{@\Delta \cdot t +_t 1[@s \cdot \exists t'[A]]}\forall\exists tw\mathcal{R}$$

Some theories on the @-logic might have rules for nesting formulae. I present some axioms only as an example:

$$\begin{array}{ll} \forall\forall s\mathcal{N}: & @\forall s \cdot t'[@\forall s \cdot t'[A]] \doteq @\forall s \cdot t'[A] \\ \exists\exists s\mathcal{N}: & @\exists s \cdot t'[@\exists s \cdot t'[A]] \doteq @\exists s \cdot t'[A] \\ \forall\forall t\mathcal{N}: & @s \cdot \forall t'[@s \cdot \forall t'[A]] \doteq @s \cdot \forall t'[A] \\ \exists\exists t\mathcal{N}: & @s \cdot \exists t'[@s \cdot \exists t'[A]] \doteq @s \cdot \exists t'[A] \end{array}$$

5 A Space-Time Operational Semantics

In this section, I illustrate an application of the present logic to operational semantics of programming languages. Informally, I adopt the following conventions:

- σ: a state of the computation, seen as a set.
- $\sigma(m/X)$: σ, in particular, $X = m \in \sigma$.
- $\langle \mathbf{true}, \sigma \rangle$: tt in state σ.
- $\langle \mathbf{false}, \sigma \rangle$: $f\!f$ in state σ.

- $@s \cdot t[\![A]\!]$: the meaning of A (it requires that $A \doteq tt$) at place s and time t.
- n, m: two real numbers.
- ϵ: time spent to execute the referred to operation.
- $\leadsto$: evaluation of some operation and its meaning is obtained.

Traditionally, the formal semantics of programming languages do not require one to state the space-time components. For a semantic rule, it is assumed that the antecedents refer to executions before the execution of the statement that appears in the consequent in the rule. However, for mobile code languages, it becomes important to make it explicit that such statements do not change the locality while some other statements do change locality. Moreover, time becomes a major issue in global environments such as the Internet.

The @-logic can be used as a Space-Time semantics for more general purpose programming languages, or simply for those languages that support code mobility.

Here, I present an operational semantics of the well known ***while*** language, extracted from [98] with slight changes in addition to the present author's notation, to make it explicit that their constructs do not change locality.

5.1 The evaluation of Boolean expressions

$$@s \cdot t[\![\langle \mathbf{true}, \sigma\rangle]\!] \leadsto @s \cdot t +_t \epsilon[\![\mathbf{true}]\!] \qquad @s \cdot t[\![\langle \mathbf{false}, \sigma\rangle]\!] \leadsto @s \cdot t +_t \epsilon[\![\mathbf{false}]\!]$$

where ϵ is the time for executing the operation.

$$\frac{@s \cdot t_0[\![\langle a_0, \sigma\rangle]\!] \leadsto @s \cdot t_0 +_t \epsilon_0[\![n]\!] \quad @s \cdot t_0 +_t \epsilon_0[\![\langle a_1, \sigma\rangle]\!] \leadsto @s \cdot t_0 +_t \epsilon_0 +_t \epsilon_1[\![m]\!] \quad n = m}{@s \cdot t_0[\![\langle a_0 = a_1, \sigma\rangle]\!] \leadsto @s \cdot t_0 +_t \epsilon_0 +_t \epsilon_1[\![\mathbf{true}]\!]}$$

Notice that the present author's notation allows the semantics to make explicit that a_0 is performed before a_1. For a parallel version, the rule would be slightly different from the one above.

$$\frac{@s \cdot t_0[\![\langle a_0, \sigma\rangle]\!] \leadsto @s \cdot t_0 +_t \epsilon_0[\![n]\!] \quad @s \cdot t_0 +_t \epsilon_0[\![\langle a_1, \sigma\rangle]\!] \leadsto @s \cdot t_0 +_t \epsilon_0 +_t \epsilon_1[\![m]\!] \quad n \neq m}{@s \cdot t_0[\![\langle a_0 = a_1, \sigma\rangle]\!] \leadsto @s \cdot t_0 +_t \epsilon_0 +_t \epsilon_1[\![\mathbf{false}]\!]}$$

For the less than or equal to operator, there exist two extra rules such as:

$$\frac{@s \cdot t_0[\![\langle a_0, \sigma\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon_0[\![n]\!] \quad @s \cdot t_0 +_t \epsilon_0[\![\langle a_1, \sigma\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon_0 +_t \epsilon_1[\![m]\!] \quad n \leq m}{@s \cdot t_0[\![\langle a_0 \leq a_1, \sigma\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon_0 +_t \epsilon_1[\![\mathbf{true}]\!]}$$

$$\frac{@s \cdot t_0[\![\langle a_0, \sigma\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon_0[\![n]\!] \quad @s \cdot t_0 +_t \epsilon_0[\![\langle a_1, \sigma\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon_0 +_t \epsilon_1[\![m]\!] \quad n > m}{@s \cdot t_0[\![\langle a_0 \leq a_1, \sigma\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon_0 +_t \epsilon_1[\![\mathbf{false}]\!]}$$

More two rules for the negation:

$$\frac{@s \cdot t_0[\![\langle b, \sigma\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon_0[\![\mathbf{true}]\!]}{@s \cdot t_0[\![\langle \neg b, \sigma\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon_0 +_t \epsilon_1[\![\mathbf{false}]\!]}$$

$$\frac{@s \cdot t_0[\![\langle b, \sigma\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon_0[\![\mathbf{false}]\!]}{@s \cdot t_0[\![\langle \neg b, \sigma\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon_0 +_t \epsilon_1[\![\mathbf{true}]\!]}$$

Conjunction:

$$\frac{\begin{array}{c} @s \cdot t_0[\![\langle b_0, \sigma\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon_0[\![\alpha]\!] \\ @s \cdot t_0 +_t \epsilon_0[\![\langle b_1, \sigma\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon_0 +_t \epsilon_1[\![\beta]\!] \end{array}}{@s \cdot t_0[\![\langle b_0 \ \&\ b_1, \sigma\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon_0 +_t \epsilon_1 +_t \epsilon_2[\![\alpha \ \&\ \beta]\!]}$$

Disjunction:

$$\frac{\begin{array}{c} @s \cdot t_0[\![\langle b_0, \sigma\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon_0[\![\alpha]\!] \\ @s \cdot t_0 +_t \epsilon_0[\![\langle b_1, \sigma\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon_0 +_t \epsilon_1[\![\beta]\!] \end{array}}{@s \cdot t_0[\![\langle b_0 \vee b_1, \sigma\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon_0 +_t \epsilon_1 +_t \epsilon_2[\![\alpha \vee \beta]\!]}$$

5.2 The execution of commands

In this section, I present an operational semantics of the commands in the *while* language.

Atomic commands:

$$@s \cdot t[\![\langle \mathbf{skip}, \sigma\rangle]\!] \rightsquigarrow @s \cdot t +_t \epsilon[\![\sigma]\!]$$

$$\frac{@s \cdot t_0[\![\langle a, \sigma\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon_0[\![m]\!]}{@s \cdot t_0[\![\langle X := a, \sigma\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon_0 +_t \epsilon_1[\![\sigma(m/X)]\!]}$$

Sequencing:

$$\frac{@s \cdot t_0[\![\langle c_0, \sigma\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon_0[\![\sigma'']\!] \quad @s \cdot t_0 +_t \epsilon_0[\![\langle c_1, \sigma''\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon_0 +_t \epsilon_1[\![\sigma']\!]}{@s \cdot t_0[\![\langle c_0; c_1, \sigma\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon_0 +_t \epsilon_1[\![\sigma']\!]}$$

Conditionals[25]:

$$\frac{@s \cdot t_0[\![\langle b, \sigma\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon_0[\![\mathbf{true}]\!] \quad @s \cdot t_0 +_t \epsilon_0[\![\langle c_1, \sigma\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon_0 +_t \epsilon_1[\![\sigma']\!]}{@s \cdot t_0[\![\langle \mathbf{if}\ b\ \mathbf{then}\ c_1\ \mathbf{else}\ c_2, \sigma\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon_0 +_t \epsilon_1[\![\sigma']\!]}$$

$$\frac{@s \cdot t_0[\![\langle b, \sigma\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon_0[\![\mathbf{false}]\!] \quad @s \cdot t_0 +_t \epsilon_0[\![\langle c_2, \sigma\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon_0 +_t \epsilon_1[\![\sigma']\!]}{@s \cdot t_0[\![\langle \mathbf{if}\ b\ \mathbf{then}\ c_1\ \mathbf{else}\ c_2, \sigma\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon_0 +_t \epsilon_1[\![\sigma']\!]}$$

While-loops:

$$\frac{@s \cdot t_0[\![\langle b, \sigma\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon[\![\mathbf{false}]\!]}{@s \cdot t_0[\![\langle \mathbf{while}\ b\ \mathbf{do}\ c, \sigma\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon +_t \Delta t[\![\sigma]\!]}$$

$$\frac{@s \cdot t_0[\![\langle b, \sigma\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon_0[\![\mathbf{true}]\!] \quad @s \cdot t_0 +_t \epsilon_0[\![\langle c, \sigma\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon_0 +_t \epsilon_1[\![\sigma'']\!] \quad @s \cdot t_0 +_t \epsilon_0 +_t \epsilon_1[\![\langle \mathbf{while}\ b\ \mathbf{do}\ c, \sigma''\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon_0 +_t \epsilon_1 +_t \epsilon_2[\![\sigma']\!]}{@s \cdot t_0[\![\langle \mathbf{while}\ b\ \mathbf{do}\ c, \sigma\rangle]\!] \rightsquigarrow @s \cdot t_0 +_t \epsilon_0 +_t \epsilon_1 +_t \epsilon_2[\![\sigma']\!]}$$

6 Mobility and Resources

In this section, I demonstrate how the @-logic can capture material resources and mobility of objects. The issue of matter is discussed in other contributions such as [14].

To achieve this purpose, let *Obj* denote the set of all physical objects in the real world. Therefore, the property of uniqueness of its elements can be written as:

$$\begin{aligned}\forall(o_1, o_2 \in Obj,\ t \in \mathbb{T},\ s_1, s_2 \in \mathbb{S}).\ & (s_1 \not\subset s_2 \wedge s_1 \not\supset s_2) \wedge \\ & (@\exists s_1 \cdot \forall t[o_1] \wedge @\exists s_2 \cdot \forall t[o_1] \rightarrow s_1 = s_2) \wedge \\ & (@\forall s_1 \cdot \forall t[o_1] \wedge @\forall s_1 \cdot \forall t[o_2] \rightarrow o_1 = o_2)\end{aligned}$$

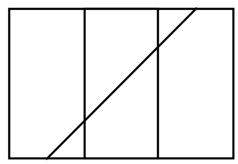
Thus, in the world described above, I can also represent mobility of an object slower than the speed of light in the following abstract way:

$$\nearrow (x, \langle p_0, t_0\rangle, \langle p_1, t_1\rangle) \stackrel{def}{=} @p_0 \cdot t_0[x] \wedge @p_1 \cdot t_1[x]$$

where $p_0, p_1 \in \mathbb{S}$, and x asserts the existence of some particular object, or, more abstractly,

$$\nearrow (x, p_0, p_1) \stackrel{def}{=} \exists t_0, t_1 \in \mathbb{T}.\ (t_0 <_t t_1) \wedge @p_0 \cdot t_0[x] \wedge @p_1 \cdot t_1[x]$$

or, given the above property, one can represent mobility graphically, from p_0 to p_1 during $t_1 -_t t_0$ interval, that is $@[p_0, p_1] \cdot [t_0, t_1][x]$, in accordance with the following space-time diagram:



which is in the same style of interpretation as the other space-time diagrams, presented above.

One can build a theory that would be aware of resources. I observe that space and time have always been resources, since prehistory. I can generate material resources from both notions as I do in the following illustration: I have five dollars (D) and, for this price, I can either order (O) one pizza (I) or pasta (A). By letting t be the time for the action of buying one of these meals, I can formalize this situation as follows:

$$@P\cdot <_t u[D \geq 5] \wedge @\exists \cdot [u, t][O] \wedge @P\cdot >_t t[] \wedge @hand\cdot >_t t[I \veebar A]$$

where P indicates a content (e.g. the pocket), $\veebar$ is the exclusive or, and I can generate a rule for that transaction itself:

$$(\forall d, u)((@P \cdot t[D = d \wedge d \geq 5] \wedge$$
$$(\exists \kappa)\ (\kappa = u \wedge @\exists \cdot [t, \kappa][O])) \rightarrow @P\cdot >_t u[D = d - 5] \wedge @hand\cdot >_t u[I \veebar A])$$

This example is based on [58].

7 Conclusion

The combination of physical and psychological notions of mobility, space, time and knowledge is so common in the daily life that it is not difficult to find good examples of it: while an observer is sitting down in a café, she can see two men starting to shake hands, then a bus stops between her and the scene blocking her view. Then, she no longer knows when the men stop shaking hands, but can guess that it is not for so long, depending on the place of that scene and her cultural background. The present calculus and deduction can be somewhat uselful.

When weighing up possibilities in any situation, one thinks carefully in order to make use of importance factors from premises to some associated conclusion. For performing such an inference, uncertainty is appropriate and provided in the @-logic. In this way, negative factors *tend* to refute hypotheses while positive factors *tend* to prove them. Thus, the result from reasoning means the average of the negative and positive results, and this is a significant skill for mobile agents.

The notions of space and time can be viewed as either abstract or physical. For example, the Pascal assignment instruction $a := a + b$ can be easily expressed as

$$@a \cdot t[value = a'] \& @b \cdot t +_t 1[value = b'] \& @a \cdot t +_t 2[value = a' + b']$$

where $t \in \mathbb{Z}$.

The grammar is expressive enough to represent sequents using sets of formulae as notion of space as exemplified here, such as $@\Delta \cdot t[C]$, although, for the @-calculus, one can make use of the form $@s \cdot t[\Delta \vdash C]$.

For further work, one can move the mobility operation from the application level to the level of structural rules, and then one obtains mobile derivations. In this way, one cannot make use of a formula at some place P until the formula arrives at P. On the other hand, once the formula is used in some derivation, it cannot escape to another place, for the operation that causes mobility is explicit. Now, the notion of time is not relative to the rule, but instead to the whole derivation. These features simulate the behavior of mobile code systems abstracting details of uncertainty and other variables in the real world.

References

[1] Peter Aczel, Harold Simmons, and Stanley S. Wainer, editors. *Proof Theory*. Cambridge University Press, 1992.

[2] James Allen. An interval-based representation of temporal knowledge. In *Proceedings of 7th IJCAI*, pages 221–226, 1981.

[3] James Allen. Towards a general theory of action and time. *Artificial Intelligence*, 23:123–154, 1984.

[4] James Allen. Time and time again: The many ways to represent time. *International Journal of Intelligent Systems*, 6(4):341–355, July 1991.

[5] James Allen and George Ferguson. Actions and events in interval temporal logic. *Journal of Logic and Computation*, 4(5), 1994.

[6] James Allen and James Hendler, editors. *Readings in Planning*. Representation and Reasoning. Morgan Kaufmann, San Mateo, California, 1990.

[7] Alan Ross Anderson and Nuel D. Belnap Junior. *Entailment: The Logic of Relevance and Necessity*, volume 1. Princeton University Press, 1975.

[8] Ken Arnold and James Goslin. *The Java Programming Language*. Addison-Wesley Publishing Company, 1996.

[9] Diderik Batens, Chris Mortensen, Graham Priest, and Jean-Paul Van Bendegem, editors. *Frontiers of Paraconsistent Logic*, volume 8 of *Studies in Logic and Computation*. Research Studies Press Ltd, 2000.

[10] Paul W. Beame and Samuel R. Buss, editors. *Proof Complexity and Feasible Arithmetics: DIMACS Workshop 96*, volume 39 of *DIMACS series in discrete mathematics and theoretical computer science*. American Mathematical Society, 1996.

[11] Nuel D. Belnap Junior. A useful four-valued logic. In J Michael Dunn and George Epstein, editors, *Proceedings of the Fifth International Symposium on Multiple-Valued Logic*, Modern Uses of Multiple-Valued Logic, pages 8–37. Indiana University, D Reidel Publishing Company, 1975.

[12] Paul Benacerraf and Hilary Putnam, editors. *Philosophy of Mathematics: selected readings*. Prentice-Hall, Inc., 1964.

[13] Paul Bernays. *Philosophy of Mathematics*, chapter On Platonism in Mathematics, pages 274–286. Prentice-Hall, Inc., 1964.

[14] Thomas Bittner. Rough sets in spatio-temporal data mining. In *Temporal, Spatial and Spatio-Temporal Data Mining*, volume LNAI 2007 of *Lecture Notes in Artificial Intelligence*, pages 89–104. Springer, September 2000.

[15] Wayne D. Blizard. A formal theory of objects, space and time. *The Journal of Symbolic Logic*, 55(1):74–89, March 1990.

[16] Robert Bull. *Logic and Reality: essays on the legacy of Arthur Prior*, chapter Logics without Contraction I, pages 317–336. Oxford University Press, 1996.

[17] Martin Bunder. *Logic and Reality: essays on the legacy of Arthur Prior*, chapter Logics without Contraction II, pages 337–349. Oxford University Press, 1996.

[18] John P. Burgess. *Proof, Logic and Formalization*, chapter Proofs About Proofs: a Defense of Classical Logic. Part I: the aim of classical logic, pages 8–23. Routledge, 1992.

[19] Stanley N. Burris. *Logic for Mathematics and Computer Science*. Prentice Hall, Inc., 1998.

[20] Luca Cardelli and Andrew D. Gordon. *Foundations of Software Science and Computational Structures*, volume 1378 of *Lecture Notices in Computer Science*, chapter Mobile Ambients, pages 140–155. Springer-Verlag, 1998. Also Proceedings of FoSSaCS'98.

[21] Alexander Chagrov and Michael Zakharyaschev. *Modal Logic*, volume 35 of *Oxford Logic Guides*. Oxford University Press, 1997.

[22] C. C. Chang and H. J. Keisler. *Model Theory*, volume 73 of *Studies in Logic and the Foundations of Mathematics*. North-Holland Publishing Company and American Elsevier Publishing Company, Inc., 1973.

[23] A. G. Cohn, J. M. Gooday, and B. Bennett. A comparison of structures in spatial and temporal logics. *Philosophy and the Cognitive Sciences*, 1994. Also http://www.comp.leeds.ac.uk/spacenet/gooday.html.

[24] B. J. Copeland, editor. *Logic and Reality: essays on the legacy of Arthur Prior*. Oxford University Press, 1996.

[25] G. Crocco and Luis Farias del Cerro. *Conditionals: from philosophy to computer science*. Studies in Logic and Computation. Clarendon Press, Oxford University, 1995.

[26] Tristan Crolard. Subtractive logic. *Theoretical Computer Science*, 254(1–2):151–185, March 2001.

[27] Martin Davis. *The Universal Computer: the road form Leibniz to Turing*, chapter 2 Boole Turns Logic into Algebra, pages 21–40. W. W. Norton & Company, 2000.

[28] Martin Davis. *The Universal Computer: the road form Leibniz to Turing*. W. W. Norton & Company, 2000.

[29] Giorgio Delzanno and Maurizio Martelli. Proofs as computations in linear logic. *Theoretical Computer Science*, 258(1–2):269–297, May 2001.

[30] Kees Doets. *Basic Model Theory*. CSLI International: Center for the Study of Language and Information and FoLLI: the European Association for Logic, Language and Information, 1996.

[31] Michael Dummett. *The Philosophy of Mathematics*, chapter The Philosophical Basis of Intuitionistic Logic. Oxford Readings in Philosophy. Oxford University Press, 1996.

[32] J. Michael Dunn. *Handbook of Philosophical Logic*, volume III: Alternatives to Classical Logic of *Synthese library; v. 166*, chapter Relevance Logic and Entailment, pages 117–224. Kluwer Academic Publisher, 1986.

[33] John R. Durbin. *Modern Algebra: an introduction*. John Wiley & Sons, Inc., fourth edition, 2000.

[34] E. Allen Emerson. *Handbook of Theoretical Computer Science*, volume B Formal Models and Semantics, chapter 16 Temporal and Modal Logic, pages 995–1072. The MIT Press/Elsevier, 1990.

[35] Richard L. Epstein. *Predicate Logic*. Oxford University Press, 1994.

[36] Richard L. Epstein. *The Semantics Foundations of Logic*. Oxford University Press, second edition, 1995.

[37] J. U. Ferreira, P. S. Nicolletti, and Hélio M. Silva. Tratamento de incertezas na linguagem lidia. In *Proceedings of VI Simpósio Brasileiro de Inteligência Artificial*. Brazilian Computing Society, 1989.

[38] Ulisses Ferreira. *uu* for programming languages. *ACM SIGPLAN Notices*, 35(8):20–30, August 2000.

[39] Ulisses Ferreira. A prolog-like language for the internet. In *Proceedings of CAITA-04*, Purdue, Indiana, USA, July 2004. IPSI Belgrade.

[40] Melvin Fitting. Kleene's logic, generalized. Technical report, City University of New York, Lehman College, Department of Mathematics and Computer Science, New York, 1990.

[41] Melvin Fitting. *Handbook of Logic in Artificial Intelligence and Logic Programming*, volume 1: Logical Foundations, chapter Basic Modal Logic, pages 365–448. Oxford University Press, 1993.

[42] Margaret M. Fleck. The topology of boundaries. *Artificial Intelligence*, 80(1):1–27, January 1996.

[43] Wan Fokkink. *Introduction to Process Algebra*. Texts in theoretical computer science. Springer-Verlag, 2000.

[44] Jan Łukasiewicz. *Jan Łukasiewicz Selected Works*. Series on Studies in Logic and Foundations of Mathematics. North-Holland Publishing Company and PWN - Polish Scientific Publishers, 1970.

[45] Gottlob Frege. *The Basic Laws of Arithmetic: exposition of the system*. University of California Press, 1964.

[46] Christian Freksa, Wilfried Brauer, Christopher Habel, and Karl F. Wender, editors. *Spatial Cognition* II*: Integrating Abstract Theories, Empirical Studies, Formal Methods and Practical Applications*, volume 1849 of *Lecture Notes in Artificial Intelligence*. Springer, 2000.

[47] André Fuhrmann. *An Essay on Contraction*. Studies in Logic, Language and Information. CSLI Publications and FoLLI, 1997.

[48] Dov M. Gabbay. *Labelled Deductive Systems*, volume 1 of *Oxford Logic Guides 33*. Oxford University Press, 1996.

[49] Dov M. Gabbay. *Elementary Logics: A Procedural Perspective*. Prentice Hall Series in Computer Science. Prentice Hall Europe, 1998.

[50] Dov M. Gabbay and Ian Hodkinson. *Logic and Reality: essays on the legacy of Arthur Prior*, chapter Temporal Logic in the Context of Databases, pages 69–87. Oxford University Press, 1996.

[51] Dov M. Gabbay, Ian Hodkinson, and Mark Reynolds, editors. *Temporal Logic: mathematical foundations and computational aspects*, volume 1. Oxford University Press Inc., New York, 1994.

[52] Denis Gagné, Wanlin Pang, and André Trudel. Spatio-temporal logic for 2d multi-agent problem domains. *Expert Systems with Applications*, 12(1):141–145, 1997.

[53] L. T. F. Gamut. *Logic, Language and Meaning*, volume 1 Introduction to Logic, chapter Beyond Standard Logic, pages 156–194. The University of Chicago Press, 1991.

[54] L. T. F. Gamut. *Logic, Language and Meaning*, volume 2 Intensional Logic and Logical Grammar. The University of Chicago Press, 1991.

[55] Michael Gelfond and Vladimir Lifschitz. Logic programs with classical negation. In *Proceedings of 7th International Conference on Logic Programming*, pages 579–597, Cambridge MA, 1990. The MIT Press.

[56] Michael Gelfond and Vladimir Lifschitz. Classical negation in logic programs and disjunctive databases. *New Generation Computing. Ohmsha Ltd and Spring-Verlag*, pages 365–385, 1991.

[57] Jean-Yves Girard. Linear logic. *Theoretical Computer Science*, 50:1–102, 1987.

[58] Jean-Yves Girard. *Advances in Linear Logic*, chapter Linear Logic: its syntax and semantics, pages 1–42. Cambridge University Press, 1995.

[59] Jean-Yves Girard, Paul Taylor, and Yves Lafont. *Proofs and Types.* Cambridge University Press, 1993.

[60] Andrew D. Gordon. *Functional Programming and Input/Output.* Distinguished Dissertations in Computer Science. Cambridge University Press, 1994.

[61] James Gosling and H. McGilton. The java language environment: A white paper. Technical report, Sun Microsystems, October 1995.

[62] Anil Gupta and Nuel Belnap Junior. *The Revision Theory of Truth*, chapter 2 Fixed Points: Some Basic Facts, page 43. The MIT Press, 1993.

[63] John F. Horty. *Agency and Deontic Logic*. Oxford University Press, 2001.

[64] Thomas W. Hungerford. *Algebra*, volume 73 of *Graduate Texts in Mathematics*. Springer-Verlag, 1974. Corrected eighth printing, 1996.

[65] Christian S. Jensen, Markus Schneider, Bernhard Seeger, and Vassilis J. Tsotras, editors. *Advances in Spatial and Temporal Databases*, volume 2121 of *Lecture Notes in Computer Science*. Springer, July 2001.

[66] Immanuel Kant. *Logic*. Bobbs-Merrill, Indianapolis, 1974. Original in German Logik. Translation with an introduction by Robert S Hartman and Wolfgang Schwarz.

[67] Stephen C. Kleene. *Introduction of Metamathematics*. D. Van Nostrand, Princeton, 1952.

[68] Bart Kosko. *Fuzzy Thinking: The New Science of Fuzzy Logic*. HarperCollinsPublishers, Flamingo, 1994.

[69] William G. Lycan. *Real Conditionals*. Oxford University Press, 2001.

[70] Grzegorz Malinowski. *Many-Valued Logics*. Number 25 in Oxford Logic Guides. Claredon Press, Oxford University, 1993.

[71] K. Meinke and J. V. Tucker. *Handbook of Logic in Computer Science*, volume 1: Mathematical Structures, chapter Universal Algebra, pages 189–411. Oxford University Press, 1992.

[72] John C. Mitchell. *Foundations for Programming Languages*, chapter Categories and Recursive Types, pages 490–505. Foundations of Computing. The MIT Press, 1996. Section 7.3 Kripke Lambda Models and Functor Categories.

[73] Sara Negri and Jan Von Plato. *Structural Proof Theory*. Cambridge University Press, 2001.

[74] Zoran Ognjanovic and Miodrag Rašković. Some first-order probability logics. *Theoretical Computer Science*, 247(1–2):191–212, September 2000.

[75] Z. Pawlak. Rough sets. *International Journal Comput. Inform*, 11:341–356, 1982.

[76] Lech Polkowski and Andrzej Skowron, editors. *Rough Sets and Current Trends in Computing: first international conference / RSCTC'98*, volume LNAI 1424 of *Lecture Notes in Artificial Intelligence*. Springer, Warsaw, Poland, June 1998. Proceedings.

[77] Arthur Prior. *Formal Logic*. Oxford University Press, 1955.

[78] Arthur Prior. *Past, Present and Future*. Oxford University Press, 1967.

[79] Arthur Prior. *Logic and Reality: essays on the legacy of Arthur Prior*, chapter Two Essays on Temporal Realism, pages 43–51. Oxford University Press, 1996.

[80] David Randell, Zhan Cui, and Tony Cohn. A spatial logic based on regions and connection. In *Proceedings 3rd International Conference on Knowledge Representation and Reasoning*, pages 165–176, San Mateo, 1992. Morgan Kaufmann.

[81] Stephen Read. *Relevant Logic: a philosophical examination of inference*. Basil Blackwell, 1988.

[82] Greg Restall. *An Introduction to Substructural Logics*. Routledge, 2000.

[83] John F. Roddick, Kathleen Hornsby, and Myra Spiliopoulou. An updated bibliography of temporal, spatial, and spatio-temporal data mining research. In *Temporal, Spatial and Spatio-Temporal Data Mining*, volume LNAI 2007 of *Lecture Notes in Artificial Intelligence*, pages 147–163. Springer, September 2000.

[84] Rita Rodriguez and Frank Anger. *Logic and Reality: essays on the legacy of Arthur Prior*, chapter Prior's Temporal Legacy in Computer Science, pages 89–105. Oxford University Press, 1996.

[85] Bertrand Russell. *Logic and Knowledge: essays 1901-1950*. Routledge, 1997.

[86] Mark Ryan and Martin Sadler. *Handbook of Logic in Computer Science*, volume 1: Mathematical Structures, chapter Valuation Systems and Consequence Relations, pages 1–78. Oxford University Press, 1992.

[87] E. H. Shortlife. *Computer-Based Medical Consultations: MYCIN*. New York, 1976. Elsevier.

[88] John F. Sowa. *Knowledge Representation: logical, philosophical, and computational foundations*. Brooks/Cole, 511 Forest Lodge Road, Pacific Grove,CA, 2000.

[89] N. I. Styazhkin. *History of Mathematical Logic from Leibniz to Peano*. The M.I.T. Press, 1969.

[90] Beverley Tasker. *The Logic of Space-Time: (Zero Infinity Becoming)*, volume 22 of *The Loebertas Series of Philosophical Monographs*. Loebertas, 1998.

[91] Anne Sjerp Troelstra. *Lectures on Linear Logic*. Number 29 in CSLI lecture notes. Center for the Study of Language and Information, CSLI/SRI International, 1992.

[92] Anne Sjerp Troelstra and H. Schwichtenberg. *Basic Proof Theory*, volume 43 Basic Proof Theory of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press, 1996.

[93] Alan M. Turing. *The Undecidable - basic papers on undecidable propositions, unsolvable problems and computable functions*, chapter On Computable Numbers, With an Application to the Entscheidungsproblem, pages 115–151. Raven Press, Hewlett, New York, 1965. A correction, pages 152–154.

[94] Alasdair Urquhart. *Handbook of Philosophical Logic*, volume 3: Alternatives to Classical Logic of *Synthese library; v. 166*, chapter Many-Valued Logic, pages 71–116. Kluwer Academic Publishers, 1986.

[95] Jean van Heijenoort. *From Frege to Gödel*. Harvard University Press, 1967.

[96] Michalis Vazirginiannis and Ouri Wolfson. A spatiotemporal model and language for moving objects on road networks. In *7th International Symposium, SSTD 2001*, pages 20–35. Springer, July 2001. LNCS, 2121.

[97] Jan Vitek, editor. *Mobile Object Systems: towards the programmable Internet*. Number 1222 in Lecture Notes in Computer Science. Springer-Verlag, 1997.

[98] Glynn Winskel. *The Formal Semantics of Programming Languages: an introduction*. The MIT Press, fourth edition, 1997.