The Temporal Calculus of Conditional Objects and Conditional Events

Jerzy Tyszkiewicz[1,2]
Arthur Ramer[2]
Achim Hoffmann[2]

June 15, 1999

[1] Institute of Informatics,
University of Warsaw,
Banacha 2,
02-097 Warszawa,
Poland.
E-mail jty@mimuw.edu.pl.
Supported by the Polish Research Council KBN grant 8 T11C 002 11.

[2] School CSE,
UNSW,
2052 Sydney,
Australia.
E-mail {jty|ramer|achim}@cse.unsw.edu.au.
Supported by the Australian Research Council ARC grant A 49800112
(1998–2000).

School of Computer Science and Engineering
The University of New South Wales
Sydney 2052, Australia

Abstract

We consider the problem of defining conditional objects $(a|b)$, which would allow one to regard the conditional probability $\Pr(a|b)$ as a probability of a well-defined event rather than as a shorthand for $\Pr(ab)/\Pr(b)$. The next issue is to define boolean combinations of conditional objects, and possibly also the operator of further conditioning. These questions have been investigated at least since the times of George Boole, leading to a number of formalisms proposed for conditional objects, mostly of syntactical, proof-theoretic vein.

We propose a unifying, semantical approach, in which conditional events are (projections of) Markov chains, definable in the three-valued extension (TL|TL) of the past tense fragment of propositional linear time logic (TL), or, equivalently, by three-valued counter-free Moore machines. Thus our conditional objects are indeed stochastic processes, one of the central notions of modern probability theory.

Our model precisely fulfills early ideas of de Finetti [6], and, moreover, as we show in a separate paper [31], all the previously proposed algebras of conditional events can be isomorphically embedded in our model.

Contents

1 Preliminaries and statement of the problem

1.1 The problem of conditional objects

Probabilistic reasoning [27] is the basis of Bayesian methods of expert system inferences, of knowledge discovery in databases, and in several other domains of computer, information, and decision sciences. The model of conditioning and conditional objects we discuss serves equally to reason about probabilities over a finite domain X, or probabilistic propositional logic with a finite set of atomic formulae.

Computing of conditional probabilities of the form $\Pr(X|Y_1, \dots, Y_n)$ and, by extension of conditional beliefs, is well understood. Attempts of defining first the *conditional objects* of the basic form $X|Y$, and then defining $\Pr(X|Y)$ as $\Pr((X|Y))$ were proposed, without much success, by some of the founders of probability [2, 6]. They were taken up systematically only about 1980. The development was slow, both because of logical difficulties [22, 16, 17], and even more because the computational model is difficult to construct. (While $a|b$ appears to stand for a sentence 'if b then a', there is no obvious calculation for $\Pr(a|(b|c))$, nor intuitive meaning for $a|(b|c)$, $(a|b) \wedge (c|d)$, and the like.)

The idea of defining conditional objects was entertained by some founders of modern probability [2, 6], but generally abandoned since introduction of the measure-theoretic model. It was revived mostly by philosophers in 1970's [1, 33] with a view towards artificial intelligence reasoning. Formal computational models came in the late 1980's and early 1990's [3, 13, 11]. Only a few of them have been used for few actual calculations of conditionals and their probabilities whose values are open to questions [4, 11].

In this paper we want to give a rigorous (and yet quite natural and intuitive) probabilistic and semantical construction of conditionals, based on ideas proposed by de Finetti over a quarter a century ago [6]. It appears that this single formalism contains fragments precisely corresponding to all the previously considered algebras of conditional events [31]. Seen as a whole, it can be therefore considered as their common generalisation and perhaps *the* calculus of conditionals.

Our system consists of three layers: the logical part is a three valued extension of the past tense fragment of propositional linear time logic, the computation model are three-valued Moore machines (an extension of deterministic finite automata), and the probabilistic semantics is provided by three-valued stochastic processes, which appear to be projections of Markov chains.

1.2 The main idea

The main idea. The main idea of our approach can be seen as an attempt to provide a precise mathematical implementation of the following idea of de Finetti [6, Sect. 5.12]:

> *"In the asymptotic approach, the definition of conditional probability appears quite naturally; it suffices to repeat the definition of probability* (*as the limiting frequency*), *taking into consideration only the trials in which the conditioning event* (*hypothesis*) *is satisfied. Thus,* $P(E|H)$ *is simply the limit of the ratio between the frequency of* EH *and the frequency of* H*. If the limiting frequency of* H *exists and is different from zero, the definition is mathematically equivalent to the compound probability theorem* $P(E|H) = P(EH)/P(H)$*. But even if the frequency of* H *does not tend to a limit, or the limit is zero,* $P(E|H)$ *can nonetheless exist* (*trivial example:* $P(H|H)$ *is always equal to* 1)*."*

We believe that our attempt is successful: our system will have all the properties predicted by de Finetti, and, moreover, as we show in a separate paper [31], subsumes all the previously existing formalisms developed to deal with conditionals, and, finally, appears to be able to handle some well-known paradoxes of probability in an intuitive and yet precise manner.

Three truth values. To be able to take into account only the trials in which the hypothesis is satisfied, one has to introduce a third logical value. Informally, if one considers two players[1]: one betting $(a|b)$ will hold, and the other it will not, if in a random experiment (dice toss, coin flip) b doesn't hold, the game is drawn. The previous works considered it to be an evidence that the definition of conditionals must be necessarily based on many valued logics, the typical choices being three valued.
Note however, that assigning probability to a three-valued c is something like squeezing it to become two-valued. For one then assumes it to be true $\Pr(c)$ of time and false $1 - \Pr(c)$ of time, and the time when c has the third value, typically described as *undefined*, is lost. So, unlike most of our predecessors, we attempt to preserve the three-valuedness of conditionals as a principle, and define their probability only on the top of that.

Bet repetitions. Now, we should allow the players to repeat their bets. Here, unlike most of the previous works, if the players repeat the game, we allow them to bet on properties of the *whole* sequence of outcomes, not just the last one.
This is not uncommon in many random experiments, that the history of the bets influences the present bet somehow.
We present three natural examples, which are natural and have a simple description.

[1] This sounds definitely better than gamblers ;-).

The first possibility is that after each bet we we start over—after the result of the experiment is settled, the (temporal) history is started anew, the next experiment not taking the old results into account.
The second is just the opposite—always the entire history, including earlier experiments, is taken into account.
The third is that no repetition is allowed: after the first experiment is settled, its outcome is deemed to persist forever, and future trials are effectively null. (Regardless of each subsequent element drawn the result is always defined and remains the same.)
Roughly speaking, the first choice is adopted in bridge, the second in black-jack and the third in Russian roulette.
This suggests that a conditional isn't merely an experiment with three possible outcomes. It is indeed a *sequence* of experiments, and the third logical value, often described as *unknown*, is often *not yet known*. It is clearly a temporal concept, and thus we are going to consider conditionals as *temporal objects.* This temporal aspect is clearly of *past tense* type — the result of a bet must depend on the history (including present) of the sequence of outcomes, only.
It is worth noting that there are other approaches which consider implicitly bet repetition in the modelling of conditionals. These include [33, 25, 11, 28].

Summary. What we undertake is thus the development of a calculus of conditional objects identified with temporal rules, which, given a sequence of random elements from the underlying domain, decide after each of the drawn elements if the the conditional becomes defined, and if so, whether it is true or false.
We stipulate that, for any reasonable calculus of conditionals, *forming boolean combinations of conditionals, as well as iterated conditionals, amounts to manipulating on these rules.*
This claim is indeed well motivated: if we fail to associate such rule to a complex conditional object, we do not have any means to say, in a real-life situations, who wins the bet on this conditional and when. So to say, such a conditional would be nonprobabilistic, because one couldn't bet on it!

Novelty of our approach. We would like to stress that virtually none of the results we prove below is entirely new. Most of them are simple extensions or reformulations of already known theorems, as the reader can verify in Section 7.1. The novelty of our approach lies almost entirely in the way we assemble the results to create mathematically precise representation of an otherwise quite clear and intuitive notion. And indeed, we feel very reassured by the fact that we didn't have to invent any new mathematics for our construction. Similarly the proofs we give in this paper are quite straightforward. This is exactly the emergence of previously-unheard-of complicated algebraic structures (dubbed *conditional event algebras* in [12]), which prompted us to have a closer look at conditional events and search for simpler and more intuitive formalisations. Note that probabilists and logicians have been doing quite well without conditional events for decades, which strongly suggests they have had all the tools necessary to use conditionals in an implicit way for a long time already. To the contrary, in the emerging applied areas, and in particular in AI, there is a strong need to have conditional events explicitly present, and this is why we believe in the importance of our results.

2 The tools

2.1 Pre-conditionals

Let $\mathcal{E} = \{a, b, c, d, \dots\}$ be a finite set of basic events, and let Σ be the free Boolean algebra generated by $\mathcal{E}$, and Ω the set of atoms of Σ. Consequently, Σ is isomorphic to the powerset of Ω, and Ω itself is isomorphic to the powerset of $\mathcal{E}$. Any element of Σ will be considered as an event, and, in particular, $\mathcal{E} \subseteq \Sigma$.
The union, intersection and complementation in Σ are denoted by $a \cup b$, $a \cap b$ and $a^{\complement}$, respectively. The least and greatest elements of Σ are denoted $\varnothing$ and Ω, respectively. However, sometimes we use a more compact notation, replacing $\cap$ by juxtaposition. When we turn to logic, it is customary to use yet another notation: $a \vee b$, $a \wedge b$ and $\neg a$, respectively. In this situation Ω appears as *true* and $\varnothing$ as *false*, but 1 and 0, respectively, are incidentally used, as well. Generally we are quite anarchistic in our notation, as long as it does not create ambiguities.
We introduce the set $\mathbf{3} = \{0, 1, \perp\}$ of truth values, interpreted as *true*, *false* and *undefined*, respectively. The subset of $\mathbf{3}$ consisting of 0 and 1 will be denoted $\mathbf{2}$.
It follows from the discussion above that we are going to look for conditionals in the set $\mathcal{PC} = \mathbf{3}^{\Omega^+}$ of three-valued functions c from the set Ω^+ of finite nonempty sequences of atomic events from Ω into $\mathbf{3}$. We will call such functions *pre-conditionals*, since to deserve the name of conditionals they must obey some additional requirements.
Sometimes it is convenient to represent such objects in two other, slightly different, yet equivalent forms:

- The second representation are length-preserving mappings $c_+ : \Omega^+ \to \mathbf{3}^+$ such that $c_+(v)$ is a prefix of $c_+(vw)$. The set of all such mappings will be denoted $\mathcal{PC}_+$.
- The third representation are mappings $c_\infty : \Omega^\infty \to \mathbf{3}^\infty$ such that if $w, v \in \Omega^\infty$ have a common prefix of length n, then $c_\infty(w)$ and $c_\infty(v)$

have a common prefix of length n, too. The set of all such mappings will be denoted $\mathcal{PC}_\infty$.

On the set $\Omega^+ \cup \Omega^\infty$ one has the natural partial order relation of being a prefix. Suprema of sets in this partial order are denoted by $\bigsqcup$.
In general, c, c_+ and c_∞ denote always three representations of the same pre-conditional, and the subscript (or its lack) indicates what representation we take at the moment, and we choose it according to what is most convenient. The three representations c, c_+ and c_∞ are related by the equalities

$$\begin{aligned}
c(\omega_1 \dots \omega_n) &= \text{last-letter-of}(c_+(\omega_1 \dots \omega_n)), \\
c(\omega_1 \dots \omega_n) &= n\text{th-letter-of}(c_\infty(\omega_1 \dots \omega_n \dots)), \\
c_+(\omega_1 \dots \omega_n) &= c(\omega_1)c(\omega_1\omega_2)\dots c(\omega_1 \dots \omega_n), \\
c_+(\omega_1 \dots \omega_n) &= \text{first-}n\text{-letters-of}(c_\infty(\omega_1 \dots \omega_n \dots)), \\
c_\infty(\omega_1 \dots \omega_n \dots) &= c(\omega_1)c(\omega_1\omega_2)\dots c(\omega_1 \dots \omega_n)\dots, \\
c_\infty(\omega_1 \dots \omega_n \dots) &= \bigsqcup\{c_+(\omega_1 \dots \omega_n) \ / \ n = 1, 2 \dots\}.
\end{aligned} \tag{1}$$

Even though we are on a rather preliminary level of our construction, we can address the general question of defining connectives among pre-conditionals already now. In our setting such a connective is indeed a function from some power of the space of pre-conditionals into itself. However, to fulfill the intuitive requirement that a connective should depend solely on the outcomes of its arguments and that it should refer to the history, only, the following additional condition must be met.
For any connective $\alpha : \mathcal{PC}_+^n \to \mathcal{PC}_+$ and any $\varphi_1, \dots, \varphi_n, \varphi'_1, \dots, \varphi'_n \in \mathcal{PC}_+$, $v, w \in \Omega^+$ satisfying $\varphi_i(w) = \varphi'_i(v)$ for $i = 1, \dots, n$ holds

$$\alpha(\varphi_1, \dots, \varphi_n)(w) = \alpha(\varphi'_1, \dots, \varphi'_n)(v).$$

Note that we permit strong dependence on the history: we do not require the connective to depend just on the present values of its arguments, we allow it to depend on their whole histories. However, if a particular connective α meets the former, stronger requirement, whose formal statement can be obtained from the above condition by replacing $\mathcal{PC}_+$ by $\mathcal{PC}$ everywhere it occurs, we call it a *present tense connective.*
Connectives which are not present tense will be called *past tense*. Any n-ary present tense connective of pre-conditionals is fully characterised by a mapping $\mathbf{3}^n \to \mathbf{3}$. Note that any connective α, not necessarily present tense one, can be completely specified by a mapping $\bigcup_{t>0} \underbrace{\mathbf{3}^t \times \dots \times \mathbf{3}^t}_{n \text{ times}} \to \mathbf{3}$.

Just like their connectives, pre-conditionals can be present tense, too. A pre-conditional $c : \Omega^+ \to \mathbf{3}$ is called *present tense* iff $c(v) = c(w)$ holds whenever last-letter-of(v) = last-letter-of(w). So indeed a present tense pre-conditional is completely determined by a function $\Omega \to \mathbf{3}$.

2.2 The formalisms

Our intention is to distinguish conditionals among pre-conditionals. Therefore, in order to deal with them, we need a formalism aimed at dealing with sequences of symbols from a finite alphabet. There are many candidates of this kind, including regular expressions and their subclasses, grammars of various kinds, deterministic or nondeterministic automata, temporal logics, first order logic and higher order logics.

Our choice, which will be carefully motivated later on, is to use three-valued counterparts of a certain particular class of finite automata and of past tense temporal logic. When the probabilities come into play conditional events of a fixed probability space are represented by Markov chains.

We introduce here briefly the main formalisms used throughout this paper: temporal logic, Moore machines and Markov chains.

2.3 Temporal logic

Let us first define *temporal logic of linear discrete past time*, called TL. We follow the exposition in [7], tailoring the definitions somewhat towards our particular needs.

The formulas are built up from the set $\mathcal{E}$ (the same set of basic events as before), interpreted as propositional variables here, and are closed under the following formula formation rules:

1. Every $a \in \mathcal{E}$ is a formula of temporal logic.
2. If $\varphi, \psi \in$ TL, then their boolean combinations $\varphi \vee \psi$ $\neg\varphi$ are in TL. The other Boolean connectives: $\wedge, \rightarrow, \leftrightarrow, \ldots$ can be defined in terms of $\neg$ and $\vee$, as usual.
3. If $\varphi, \psi \in$ TL, then their past tense temporal combinations $\bullet\varphi$ and $\varphi \,\mathsf{Since}\, \psi$ are in TL, where $\bullet\varphi$ is spelled “previously φ.”

A model of temporal logic is a sequence $\mathcal{M} = s_0, s_1, \ldots, s_n$ of states, each state being a function from $\mathcal{E}$ (the same set of basic events as before) to the boolean values $\{0, 1\}$. Note that a state can be therefore understood as an atomic event from Ω, and $\mathcal{M}$ can be thought of as a word from Ω^+. To be explicit we declare that the states of $\mathcal{M}$ are ordered by $\leq$. Rather than using the indices of states to denote their order, we simply write $s \leq t$ to denote that a state t comes later than, or is equal to, a state s; similarly $s + 1$ denotes the successor state of s. We adopt the convention that, unless explicitly indicated otherwise, a model is always of length $n + 1$, and thus n is always the last state of a model.

For every state s of $\mathcal{M}$ we define inductively what it means that a formula $\varphi \in$ TL is satisfied in the state s of $\mathcal{M}$, symbolically $\mathcal{M}, s \models \varphi$.

1. $\mathcal{M}, s \models a$ iff $s(a) = 1$

2.

$$\mathcal{M}, s \models \neg\varphi :\iff \mathcal{M}, s \not\models \varphi,$$
$$\mathcal{M}, s \models \varphi \vee \psi :\iff \mathcal{M}, s \models \varphi \text{ or } \mathcal{M}, s \models \psi.$$

3.

$$\mathcal{M}, s \models \bullet\varphi :\iff s > 0 \text{ and } \mathcal{M}, s-1 \models \varphi;$$
$$\mathcal{M}, s \models \varphi \,\mathsf{Since}\, \psi :\iff (\exists t \leq s)(\mathcal{M}, t \models \psi \text{ and } (\forall t < w \leq s) M, w \models \varphi).$$

The syntactic abbreviations $\blacksquare\varphi$ and $\blacklozenge\varphi$ are of common use in TL. They are defined by $\blacklozenge\varphi \equiv \mathit{false}\,\mathsf{Since}\,\varphi$ and $\blacksquare\varphi \equiv \neg\blacklozenge\neg\varphi$. The first of them is spelled "once φ" and the latter "always in the past φ".
Their semantics is then equivalent to

$$\mathcal{M}, s \models \blacksquare\varphi :\iff (\forall t \leq s)\mathcal{M}, t \models \varphi;$$
$$\mathcal{M}, s \models \blacklozenge\varphi :\iff (\exists t \leq s)\mathcal{M}, t \models \varphi.$$

Using the given temporal and boolean connectives, one can write down quite complex formulae describing temporal properties of models $\mathcal{M}, s$. We will see several such examples in this paper, and even more can be found in [31].

2.4 Moore machines

In this section we follow [18], tailoring the definitions, again, towards our needs.
A *deterministic finite automaton* is a five-tuple $\mathfrak{A} = (Q, \Omega, \delta, q_0, T)$, where Q is its set of states, Ω (the same set of atomic events as before) is the input alphabet, $q_0 \in Q$ is the initial state and $\delta : Q \times \Omega \to Q$ is the transition function. $T \subseteq Q$ is the set of accepting states.
We picture $\mathfrak{A}$ as a labelled directed graph, whose vertices are elements of Q, a the function δ is represented by directed edges labelled by elements of Ω: the edge labelled by $\omega \in \Omega$ from $q \in Q$ leads to $\delta(q, \omega)$. The initial state is typically indicated by an unlabelled edge "from nowhere" to this state.
As the letters of the input word $w \in \Omega^+$ come in one after another, we walk in the graph, always choosing the edge labelled by the letter we receive. What we do with the word depends on the state we are in upon reaching the end of the word. If it is in T, the automaton accepts the input, otherwise it rejects it.
Formally, to describe the computation of $\mathfrak{A}$ we extend δ to a function $\hat{\delta} : Q \times \Omega^+ \to Q$ in the following way:

$$\hat{\delta}(q,w) = \begin{cases} \delta(q,w) & \text{if } |w| = 1 \\ \delta(\hat{\delta}(q,v),\omega) & \text{if } w = v\omega. \end{cases}$$

$L(\mathfrak{A}) \subseteq \Omega^+$ is the set of words accepted by $\mathfrak{A}$.
A *Moore machine* $\mathfrak{A}$ is a six-tuple $\mathfrak{A} = (Q, \Omega, \Delta, \delta, h, q_0)$, where (Q, Ω, δ, q_0) is a deterministic finite automaton but the set of accepting states, Δ is a finite output alphabet and h is the output function $Q \to \Delta$. In addition to what $\mathfrak{A}$ does as a finite automaton, at each step it reports to the outside world the value $h(q)$ of the state q in which it is at the moment. Drawing a Moore machine we indicate h by labelling the states of its underlying finite automaton by their values under h. In addition, we almost always make certain graphical simplifications: we merge all the transitions joining the same pair of states into a single transition, labelled by the union (evaluated in Σ) of all the labels. Sometimes we go even farther and drop the label altogether from one transition, which means that all the remaining input letters follow this transition.
Formally, a Moore machine computes a function $f_{\mathfrak{A}} : \Omega^+ \to \Delta^+$ defined by

$$f_{\mathfrak{A}}(\omega_1\omega_2\ldots\omega_n) = h(\hat{\delta}(q_0,\omega_1))h(\hat{\delta}(q_0,\omega_1\omega_2))\ldots h(\hat{\delta}(q_0,\omega_1\omega_2\ldots\omega_n))$$

(note that $|f_{\mathfrak{A}}(\omega_1\omega_2\ldots\omega_n)| = n$, as desired), and a function $g_{\mathfrak{A}} : \Omega^\infty \to \Delta^\infty$ defined by

$$g_{\mathfrak{A}}(\omega_1\omega_2\ldots) = \bigsqcup\{f_{\mathfrak{A}}(\omega_1\omega_2\ldots\omega_n) \ / \ n = 1, 2, \ldots\}.$$

We will be interested in Moore machines which compute **3**-valued functions. This amounts to partitioning the state set Q of $\mathfrak{A}$ into three subsets T, F, B, which we often make into parts of the machine. If we do so, we call the states in T the *accepting states* and the states in F the *rejecting states*. There will be no special name for the states in B.
A Moore machine $\mathfrak{A}$ is called *counter-free* if there is no word $w \in \Omega^+$ and no states $q_1, q_2, \ldots, q_s,\ s > 1$, such that $\hat{\delta}(q_1, w) = q_2, \ldots, \hat{\delta}(q_{s-1}, w) = q_s, \hat{\delta}(q_s, w) = q_1$.

2.5 Markov chains

For us, Markov chains are a synonym of *Markov chains with stationary transitions and finite state space.*
Formally, given a finite set I of *states* and a fixed function $p : I \times I \to [0, 1]$ satisfying

$$(\forall i \in I) \qquad \sum_{j \in I} p(i,j) = 1, \tag{2}$$

the *Markov chain* with state space I and transitions p is a sequence $\mathcal{X} = X_0, X_1, \ldots$ of random variables $X_n : W \to I$, such that

$$\Pr(X_{n+1} = j | X_n = i) = p(i,j). \tag{3}$$

The standard result of probability theory is that there exists a probability triple $(W, \mathfrak{M}, \Pr)$ and a sequence $\mathcal{X}$ such that (3) is satisfied. W is indeed the space of infinite sequences of ordered pairs of elements from I, and Pr is a certain product measure on this set.

One can arrange the values $p(i,j)$ in a matrix $\Pi = (p(i,j); i,j \in I)$. Of course, $p(i,j) \geq 0$ and $\sum_{j \in I} p(i,j) = 1$ for every i. Every real square matrix Π satisfying these conditions is called *stochastic.* Likewise, the initial distribution of $\mathcal{X}$ is that of X_0, which can be conveniently represented by a vector $\Xi_0 = (p(i); i \in I)$. Its choice is independent from the function $p(i,j)$. It is often very convenient to represent Markov chains by matrices, since many manipulations on Markov chains correspond to natural algebraic operations performed on the matrices.

For our purposes, it is convenient to imagine the Markov chain $\mathcal{X}$ in another, equivalent form: Let K_I be the complete directed graph on the vertex set I. First we randomly choose the starting vertex in I, according to the initial distribution. Next, we start walking in K_I; at each step, if we are in the vertex i, we choose the edge (i,j) to follow with probability $p(i,j)$. If we define $X_n =$ (the vertex in which we are after n steps), then X_n is indeed the same X_n as in (3).

So we will be able to *draw* Markov chains. Doing so, we will often omit edges (i,j) with $p(i,j) = 0$.

Classification of states For two states i, j of a Markov chain $\mathcal{X}$ with transition probabilities p we say that i *communicates* with j iff there is a nonzero probability of eventually getting from i to j. Equivalently, it means that there is a sequence $i = i_1, i_2, \ldots, i_n = j$ of states such that $p(i_k, i_{k+1}) > 0$ for $k = 1, \ldots, n-1$. The reflexive relation of mutual communication (i.e., that i communicates with j and j communicates with i or $i = j$) is an equivalence relation on I. Class $[i]$ communicates with class $[j]$ iff i communicates with j.

The relation of communication is a partial ordering relation on classes. The minimal elements in this partial ordering are called *ergodic sets*, and non-minimal elements are called *transient sets.* The elements of ergodic and transient sets are called ergodic and transient states, respectively.

A Markov chain all whose ergodic sets are one-element is called *absorbing*, and its ergodic states are called absorbing.
For ergodic sets one can be further define their *period*. Period of an ergodic state i is the gcd of all the numbers p such that there is a sequence $i = i_1, i_2, \ldots, i_p = i$ of states such that $p(i_k, i_{k+1}) > 0$ for $k = 1, \ldots, p-1$. It can be shown that period is a class property, i.e., all states in one ergodic class have the same period.
An ergodic set is called *aperiodic* iff its period is 1. Equivalently, it means that for every two i, j in this set and all sufficiently large n there exists a sequence $i = i_1, i_2, \ldots, i_n = j$ of states such that $p(i_k, i_{k+1}) > 0$ for $k = 1, \ldots, n-1$.
Every periodic class C of period $p > 1$ can be partitioned into p periodic subclasses $C_1, \ldots, C_p$ such that $\Pr(X_{n+1} \in C_{k+1 \pmod p} | X_n \in C_{k \pmod p}) = 1$ for all k.

3 Constructing conditionals

We make a terminological distinction. If we speak about a *conditional object*, we do not assume any probability space structure imposed on Ω. When we have such structure $(\Omega, \Sigma, \Pr)$, we speak about a *conditional event*, instead.

3.1 Conditional objects

First of all, let us note that any TL formula can be understood as a definition of a pre-conditional from $\mathcal{PC}$, which is indeed **2**-valued. Indeed, states of any model of temporal logic can be interpreted as elements of Ω, and the whole model is thus an element of Ω^+. The value the pre-conditional assigns to model $\mathcal{M}$ is 1 if $\mathcal{M}, n \models \varphi$ and 0 otherwise.
We construct a three-valued extension (TL|TL) of TL as the set of all pairs $(\varphi|\psi)$ of formulas from TL. The operator $(\cdot|\cdot)$ can be understood as a *present tense connective* of pre-conditionals, and, since formulas of TL are **2**-valued, it is sufficient to define its action as follows:

$(x\|y)$			
$x \diagdown y$	0	1	$\bot$
0	$\bot$	0	
1	$\bot$	1	
$\bot$			

Definition 1. A *conditional object of type 1* is a pre-conditional $c \in \mathcal{PC}$, definable in (TL|TL). The set of such conditional objects is denoted $\mathcal{C}$.

Definition 2. A *conditional object of type 2* is a pre-conditional $c_+ \in \mathcal{PC}_+$, such that c_+ is computable by a **3**-valued counter-free Moore machine. The set of such conditional objects is denoted $\mathcal{C}_+$.

Definition 3. A *conditional object of type 3* is a pre-conditional $c_\infty \in \mathcal{PC}_\infty$, such that c_∞ is computable by a **3**-valued counter-free Moore machine. The set of such conditional objects is denoted $\mathcal{C}_\infty$.

The following proposition says that the conditional objects of types 1, 2 and 3 are identical up to the way of representing pre-conditionals.

Theorem 4.

$$\begin{aligned}\mathcal{C}_+ &= \{c_+ \in \mathcal{PC}_+ \ / \ c \in \mathcal{C}\},\\ \mathcal{C}_\infty &= \{c_\infty \in \mathcal{PC}_\infty \ / \ c \in \mathcal{C}\},\\ \mathcal{C} &= \{c \in \mathcal{PC} \ / \ c_\infty \in \mathcal{C}_\infty\}.\end{aligned}$$

Proof. The equalities $\mathcal{C}_+ = \{c_+ \in \mathcal{PC}_+ \ / \ c_\infty \in \mathcal{C}_\infty\}$ and $\mathcal{C}_\infty = \{c_\infty \in \mathcal{PC}_\infty \ / \ c_+ \in \mathcal{C}_+\}$ are obvious. What remains to be proven are $\mathcal{C}_+ = \{c_+ \in \mathcal{PC}_+ \ / \ c \in \mathcal{C}\}$ and $\mathcal{C} = \{c \in \mathcal{PC} \ / \ c_+ \in \mathcal{C}_+\}$
It is well-known [7] that propositional temporal logic of past tense and (finite) deterministic automata are of equal expressive power, i.e., in our terminology, the sets of **2**-valued pre-conditionals from $\mathcal{PC}$ definable in TL and computable by deterministic finite automata are equal. Indeed the translations between temporal logic and automata are effective.
We start with the first equality. Let c be defined by a (TL|TL) formula $(\varphi|\psi)$. Let $\mathfrak{A} = (Q_\mathfrak{A}, \Omega, \delta_\mathfrak{A}, q_\mathfrak{A}, T_\mathfrak{A})$ and $\mathfrak{B} = (Q_\mathfrak{B}, \Omega, \delta_\mathfrak{B}, q_\mathfrak{B}, T_\mathfrak{B})$ be deterministic finite automata, computing the functions $\Omega^+ \to \mathbf{2}$ defined by φ and ψ, respectively.
Consider the Moore machine $(\mathfrak{A}|\mathfrak{B}) = (Q_\mathfrak{A} \times Q_\mathfrak{B}, \Omega, \mathbf{3}, \delta, h, (q_\mathfrak{A}, q_\mathfrak{B}))$, where

$$\delta((p,q),\omega) = (\delta_\mathfrak{A}(p,\omega), \delta_\mathfrak{B}(q,\omega)),$$

$$h((p,q)) = \begin{cases} 1 & \text{if } p \in T_A \text{ and } q \in T_\mathfrak{B},\\ 0 & \text{if } p \notin T_A \text{ and } q \in T_\mathfrak{B},\\ \perp & \text{otherwise.}\end{cases}$$

It is immediate to see that $(\mathfrak{A}|\mathfrak{B})$ computes exactly $(\varphi|\psi)_+$.
To prove the second equality, let $\mathfrak{A} = (Q, \Omega, \mathbf{3}, \delta, h, q_0)$ be a Moore machine computing c_+. We construct two deterministic finite automata $\mathfrak{A}_1 = (Q, \Omega, \delta, q_0, \vec{h}^{-1}(\{1\}))$ and $\mathfrak{A}_2 = (Q, \Omega, \delta, q_0, \vec{h}^{-1}(\{0,1\})$ from $\mathfrak{A}$, where $\vec{h}^{-1}$ stands for the co-image under h. Now let φ_1 and φ_2 be TL formulae corresponding to $\mathfrak{A}_1$ and $\mathfrak{A}_2$, respectively.
It is again immediate to see that $(\varphi_1|\varphi_2)$ defines exactly the conditional in $\mathcal{C}$ computed in $\mathcal{C}_+$ by $\mathfrak{A}$. □

Consequently, we can freely choose between the three available representations of conditional objects. Doing so, we regard (TL|TL) to be the *logic of conditional objects*, while Moore machines represent their *machine representation*. All these representations are equivalent, thanks to Theorem 4.

The classes $\mathcal{C}$, $\mathcal{C}_+$ and $\mathcal{C}_\infty$ represent the *semantics* of conditional objects, and again we can freely choose the particular kind of semantical objects, thanks to (1).
As an example, the simple conditional $(a|b) \in (\mathrm{TL}|\mathrm{TL})$ is computed by the following Moore machine.

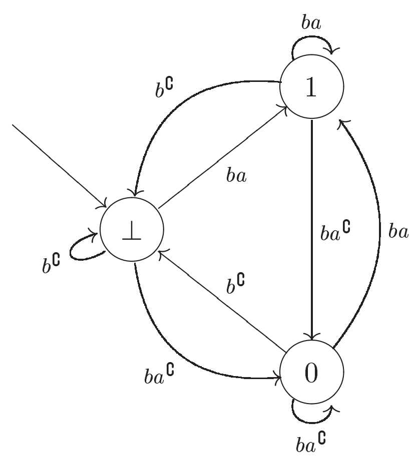

Figure 1: Moore machine representing conditional object $(a|b)$.

The above Moore machine, as it is easily seen, acts exactly according to the rule "ignore $b^{\complement}$'s, decide depending on the truth status of a when b appears". So indeed it represents the repetitions of the experiment for $(a|b)$ according to the "bridge" repetition rule *start history anew.*

3.2 Conditional events

We will be using the name *conditional events* to refer to conditionals considered with a probability space in the background.
Let $(\Omega, \mathcal{P}(\Omega), \Pr)$ be a probability space.

Definition 5 (Conditional event). Let $c \in \mathcal{C}$ be a conditional object over Ω. Suppose Ω is endowed with a probability space structure $(\Omega, \Sigma, \Pr)$. With c we associate the sequence $\mathcal{Y} = \mathcal{Y}(c) = Y_1, Y_2, \ldots$ of random variables $\Omega^\infty \to \mathbf{3}$, defined by the formula

$$Y_n(w) = n\text{-th-letter-of}(c_\infty(w)), \tag{4}$$

where Ω^∞ is considered with the product probability structure.
We call $\mathcal{Y}$ the *conditional event* associated with c, and denote it $[\![c]\!]$, while Y_n is then denoted $[\![c]\!]_n$. Note that we do not include the probability space in the notation. It will be always clear what $(\Omega, \mathcal{P}(\Omega), \Pr)$ is.

In particular, $\Pr([\![c]\!]_n = 1)$ is the probability that at time n the conditional is true, $\Pr([\![c]\!]_n = 0)$ is the probability that at time n the conditional is false, and $\Pr([\![c]\!] = \perp)$ is the probability that at time n the conditional is undefined.

Definition 6 (Probability of conditional events).
We define the *de Finetti probability at time n* of a conditional c by the formula

$$\Pr_n(c) = \frac{\Pr([\![c]\!]_n = 1)}{\Pr([\![c]\!]_n = 0 \text{ or } 1)}. \tag{5}$$

If the denominator is 0, $\Pr_n(c)$ is undefined.
The *de Finetti probability* of c is

$$\Pr(c) = \lim_{n\to\infty} \Pr_n(c), \tag{6}$$

provided that $\Pr_n(c)$ is defined for all sufficiently large n and the limit exists.
We will regard $[\![c]\!]$ as *probabilistic semantics* of c.
If $\varphi \in \mathrm{TL}$ then we write $\Pr(\varphi)$ for $\Pr((\varphi|\mathit{true}))$.

It is perhaps reasonable to explain why we want the conditional event and its probability to be defined in this way. The main motivation is that we want the conditional event and its probability to be natural and intuitive. And we achieve this by using the recipe of de Finetti, which in our case materializes in the above definitions.

4 Underlying Markov chains, Bayes' Formula and classification of conditional events

4.1 Underlying Markov chains

Let c be a conditional object and let $\mathfrak{A} = (Q, \Omega, \delta, \mathbf{3}, h, q_0)$ be a counter-free Moore machine which computes c_∞.
We define a Markov chain $\mathcal{X} = \mathcal{X}(\mathfrak{A})$ by taking the set of states of $\mathcal{X}$ to be the set Q of states of $\mathfrak{A}$, and the transition function p to be defined by

$$p(q, q') = \sum_{\substack{\omega\in\Omega \\ \delta(q,\omega)=q'}} \Pr(\{\omega\}).$$

Indeed, for every q we have

$$\sum_{q'} p(q,q') = \sum_{q'} \sum_{\substack{\omega\in\Omega \\ \delta(q,\omega)=q'}} \Pr(\{\omega\}) = \sum_{\omega\in\Omega} \Pr(\{\omega\}) = 1,$$

which means that the function p satisfies (2), which is the criterion for being a transition probability function of a Markov chain. The initial probability distribution is defined by

$$p(q) = \begin{cases} 1 & \text{if } q = q_0, \text{ the initial state of } \mathfrak{A}, \\ 0 & \text{otherwise.} \end{cases}$$

Therefore we have indeed converted $\mathfrak{A}$ into a Markov chain $\mathcal{X}$.
In the pictorial representation of the conversion process is much simpler: we take the drawing of $\mathfrak{A}$, and replace all the letters from Ω marking transitions by their probabilities according to Pr, and then contract multiple transitions between the same states into a single one, summing up their probabilities.

Theorem 7. *$\mathcal{X}$ is a Markov chain in which only transient and aperiodic states exist.*

Proof. Suppose $\mathcal{X}$ has a periodic set C of period $p > 1$, and $C_1, \ldots, C_p$ its division into periodic subclasses. Let $\omega \in \Omega$ be any atomic event with $\Pr(\{\omega\}) > 0$. Let $q \in C_1$. Since $\Pr(X_{n+1} \in C_{k+1 \pmod p} | X_n \in C_{k \pmod p}) = 1$ for all k, it follows that $\delta^1(q,\omega) = \delta(q,\omega) \in C_{2 \pmod p}$, and likewise $\delta^{k+1}(q,\omega) = \delta(\delta^k(q,\omega)) \in C_{k+1 \pmod p}$ for $k \geq 1$.
However, C is finite, so there must be $s \neq t$ such that $\delta^s(q,\omega) = \delta^t(\omega)$.
The sequence

$$\delta^s(q,\omega), \delta^{s+1}(q,\omega), \ldots, \delta^t(q,\omega) = \delta^s(q,\omega)$$

thus violates the assumption that $\mathfrak{A}$ is counter-free. □

The next corollary follows by the classical result about finite Markov chains.

Corollary 8. *For every state i of $\mathcal{X}$, the limit $\lim_{n\to\infty} \Pr(X_n = i)$ exists.*

Using $h : Q \to \mathbf{3}$, the *acceptance mapping of* $\mathfrak{A}$, we get

Theorem 9. $[\![c]\!] = h(\mathcal{X})$. □

Note that $[\![c]\!]$ defined above need not be a Markov chain itself, but it is a simple projection of a Markov chain, extracting all the invariant information. Of course, it will be typically very beneficial to work most of the time with $\mathcal{X}$, having the whole theory of Markov chains as a tool-set, and only then to move to $[\![c]\!]$.

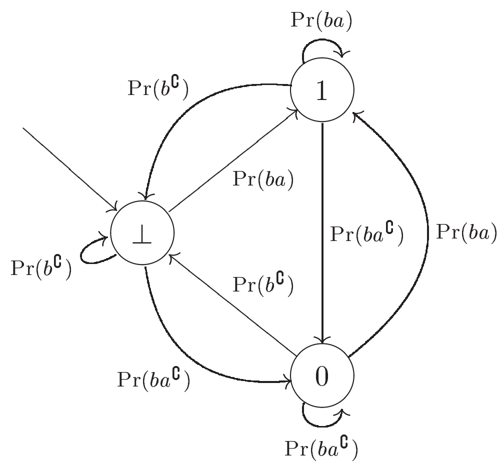

Figure 2: Markov chain corresponding to the Moore machine on Fig. 1.

Let us examine the previously given definition of $(a|b)$ to see what its probability is.
The Markov chain looks as follows:
where the initial distribution assumes probability 1 given to the state pointed to by the arrow "from nowhere".
It is easy to check that $\Pr((a|b)) = \Pr(ba)/\Pr(b)$, provided that $\Pr(b) > 0$. Indeed, for every n holds $\Pr([\![(a|b)]\!]_n = 1) = \Pr(ba)$ and $\Pr([\![(a|b)]\!]_n = 0) = \Pr(ba^{\complement})$, so $\Pr([\![(a|b)]\!]_n = 0 \text{ or } 1) = \Pr(ba) + \Pr(ba^{\complement}) = \Pr(b)$. It is so because, no matter in which state we are, these are the probabilities of getting to 1 and 0 in the next step, respectively. This evaluation will follow from Bayes' Formula below, too.

4.2 Bayes' Formula

First of all, let us note that for each $\star \in \mathbf{3}$ the limit $\lim_{n\to\infty} \Pr([\![c]\!]_n = \star)$ exists, since, for any choice of a Moore machine $\mathfrak{A}$ computing c_+ and assuming $\mathcal{X} = \mathcal{X}(\mathfrak{A})$, $\Pr([\![c]\!]_n = \star)$ is a sum of $\Pr(X_n = i)$ over all states i of $\mathcal{X}$ with $h(i) = \star$, and the latter probabilities converge by Corollary 8.
A conditional event is called *regular* iff $\lim_{n\to\infty} \Pr([\![c]\!]_n = 0 \text{ or } 1) > 0$. In particular, for regular conditionals the limit in (6) always exists and is equal to

$$\frac{\lim_{n\to\infty} \Pr([\![c]\!]_n = 1)}{\lim_{n\to\infty} \Pr([\![c]\!]_n = 0 \text{ or } 1)}.$$

Turning to the logical representation of conditionals, we have thus

Theorem 10 (Bayes' Formula). *For* $(\varphi|\psi) \in (\mathrm{TL}|\mathrm{TL})$

$$\Pr((\varphi|\psi)) = \frac{\Pr(\varphi \wedge \psi)}{\Pr(\psi)}$$

whenever the right-hand-side above is well-defined. □

Note that Bayes' Formula has been expected by de Finetti for the frequency based conditionals.

4.3 Classifying conditional events

It is interesting to consider the conditionals c for which $\lim_{n\to\infty} \Pr([\![c]\!]_n = 0 \text{ or } 1) = 0$. We can distinguish two types of such conditional events: those for which $\Pr([\![c]\!]_n = 0 \text{ or } 1)$ is identically 0 for infinitely many n, and those for which it is nonzero for all but finitely many n. The former will be called *degenerate*, the latter *strange*. We call *strictly degenerate* those degenerate events, for which $\Pr([\![c]\!]_n = 0 \text{ or } 1)$ for all but finitely many n.
The degenerate conditional events correspond to bets which infinitely often cannot be resolved, because they are undefined, and strictly degenerate events are those which are almost never defined.
Strange conditional events are more interesting. The Bayes' Formula is senseless for them, so we have to use some ad hoc methods to see if their de Finetti probability exists or not.
The first example shows that the sequence $\Pr_n(c)$ can be nonconvergent for strange c.
Consider $c_1 = (a|\blacksquare((\bullet\, a \to a^{\complement}) \wedge (\bullet\, a^{\complement} \to a) \wedge (\neg \bullet\, \mathit{true} \to a)))$, where $0 < \Pr(a) < 1$. The long temporal formula asserts that a always follows $a^{\complement}$ and $a^{\complement}$ always follows a, and at the beginning of the process ($n = 1$), where $\bullet\, \mathit{true}$ is false, a holds.
It is easily verified that

$$\Pr_n([\![c_1]\!]) = \begin{cases} 0 & \text{if } n \text{ is even,} \\ 1 & \text{if } n \text{ is odd.} \end{cases}$$

Thus the finite-time behaviour of this conditional is not probabilistic—its truth value depends solely on the *age* of the system. So for somebody expecting a pure game of chances its behaviour must seem strange (and hence the name of this class of conditional events).
Note that we have just discovered the next feature of conditionals expected by de Finetti: nonconvergence of the limiting frequency when probability of the 'given' part tends to 0.
However, again following de Finetti, if $(\varphi|\varphi)$ is strange, its de Finetti probability is 1. E.g., $\Pr(\blacksquare((\bullet\, a \to a^{\complement}) \wedge (\bullet\, a^{\complement} \to a) \wedge (\neg \bullet\, \mathit{true} \to a))|\, \blacksquare((\bullet\, a \to a^{\complement}) \wedge (\bullet\, a^{\complement} \to a) \wedge (\neg \bullet\, \mathit{true} \to a))) = 1$.

Moreover, for $c_2 = (a|\blacksquare((\bullet\, a \to a^{\complement}) \wedge (\bullet\, a^{\complement} \to a)))$ we have

$$\Pr_n(\llbracket c_2 \rrbracket) = \begin{cases} 1 - \Pr(a) & \text{if } n \text{ is even,} \\ \Pr(a) & \text{if } n \text{ is odd.} \end{cases}$$

Indeed, here the 'given' part requires that a'a and $a^{\complement}$'s alternate, but does not specify what is the case at the beginning of the process. So the probability of the whole conditional at odd times is the probability that a has happened at time 1, and at even times it is the probability that a has not happened at time 1. Therefore, when $\Pr(a) = 1/2$, $\Pr(c_2)$ exists and is 1/2. So de Finetti probabilities which are neither 0 nor 1 are possible for strange conditionals events.

However, there is a theory of convergence for strange conditional events. It is based on sub-Markov chains and requires quite different tools than what we use here. Therefore we have decided to present it in another paper [32]. As an appetizer we offer here one sufficient condition for the existence of the de Finetti probability, derived from [21].

Theorem 11. *Let* $\llbracket c \rrbracket$ *be a strange conditional event and* $\mathcal{X}$ *its underlying Markov chain. Remove from* $\mathcal{X}$ *all states from which no state* i *with* $h(i) = 0$ *or* 1 *is reachable. The reduced* $\mathcal{X}'$ *is a sub-Markov chain—the formula* (2) *for* $\mathcal{X}$ *has* $\leq$ *in place of* $=$ *. If this sub-Markov chain is formally a single aperiodic ergodic class, then the de Finetti probability of* c *exists.* □

5 Connectives of conditionals

5.1 Present tense connectives

Let us recall that present tense connectives are those, whose definition in (TL|TL) does not use temporal connectives, and therefore depends on the present, only. Equivalently, an n-ary present tense connective is completely characterised by a function $\mathbf{3}^n \to \mathbf{3}$.

Here are several possible choices for the conjunction, which is always defined as a pointwise application of the following **3** valued functions. Above we display the notation for the corresponding kind of conjunction.

$x \wedge_{\mathrm{SAC}} y$

$x \diagdown y$	0	1	$\perp$
0	0	0	0
1	0	1	1
$\perp$	0	1	$\perp$

$x \wedge_{\mathrm{GNW}} y$

$x \diagdown y$	0	1	$\perp$
0	0	0	0
1	0	1	$\perp$
$\perp$	0	$\perp$	$\perp$

$x \wedge_{\mathrm{Sch}} y$

$x \diagdown y$	0	1	$\perp$
0	0	0	$\perp$
1	0	1	$\perp$
$\perp$	$\perp$	$\perp$	$\perp$

$\sim x$	
x	$\sim x$
0	1
1	0
$\bot$	$\bot$

$x \vee_{\rm SAC} y$			
$x \diagdown y$	0	1	$\bot$
0	0	1	0
1	1	1	1
$\bot$	0	1	$\bot$

$x \vee_{\rm GNW} y$			
$x \diagdown y$	0	1	$\bot$
0	0	1	1
1	1	$\bot$	1
$\bot$	$\bot$	$\bot$	$\bot$

$x \vee_{\rm Sch} y$			
$x \diagdown y$	0	1	$\bot$
0	0	1	$\bot$
1	1	1	$\bot$
$\bot$	$\bot$	$\bot$	$\bot$

.

They can be equivalently described by syntactical manipulations in (TL|TL). The reduction rules are as follows:

$$\begin{aligned} (a|b) \wedge_{\rm SAC} (c|d) &= (abcd \vee abd^{\complement} \vee cdb^{\complement} | b \vee d) \\ (a|b) \wedge_{\rm GNW} (c|d) &= (abcd | a^{\complement} d \vee c^{\complement} d \vee abcd) \\ (a|b) \wedge_{\rm Sch} (c|d) &= (abcd|bd) \\ \sim (a|b) &= (a^{\complement}|b) \\ (a|b) \vee_{\rm SAC} (c|d) &= (ab \vee cd | b \vee d) \\ (a|b) \vee_{\rm GNW} (c|d) &= (ab \vee cd | ab \vee cd \vee bd) \\ (a|b) \vee_{\rm Sch} (c|d) &= (ab \vee cd|bd). \end{aligned} \tag{7}$$

The first is based on the principle "if any of the arguments becomes defined, act!". A good example would be a quotation from [5]:

> *"One of the most dramatic examples of the unrecognised use of compound conditioning was the first military strategy of our nation. As the Colonialists waited for the British to attack, the signal was 'One if by land and two if by sea'. This is the conjunction of two conditionals with uncertainty!"*

Of course, if the above was understood as a conjunction of two conditionals, the situation was crying for the use of $\wedge_{\rm SAC}$, whose definition has been proposed independently by Schay, Adams and Calabrese (the author of the quotation).

The conjunction $\wedge_{\rm GNW}$ represents a moderate approach, which in case of an apparent evidence for 0 reports 0, but otherwise it prefers to report unknown in a case of any doubt. Note that this conjunction is essentially the same as *lazy evaluation*, known from programming languages.

Finally, the conjunction $\wedge_{\rm Sch}$ is least defined, and acts (classically) only if both arguments become defined. It corresponds to the *strict evaluation*.

We have given an example for the use of $\wedge_{\mathrm{SAC}}$. The uses of $\wedge_{\mathrm{GNW}}$ and $\wedge_{\mathrm{Sch}}$ can be found in any computer program executed in parallel, which uses either lazy or strict evaluation of its logical conditions. And indeed both of them happily coexist in many programming languages, in that one of them is the standard choice, the programmer can however explicitly override the default and choose the other evaluation strategy.

Let us mention that all the three systems above are in fact well-known, classical so to say three-valued logics: $\langle \wedge_{\mathrm{GNW}}, \vee_{\mathrm{GNW}}, \sim \rangle$ is the logic of Lukasiewicz, $\langle \wedge_{\mathrm{SAC}}, \vee_{\mathrm{SAC}}, \sim \rangle$ is the logic of Sobociński, and $\langle \wedge_{\mathrm{Sch}}, \vee_{\mathrm{Sch}}, \sim \rangle$ is the logic of Bochvar.

5.2 Past tense connectives

The following connective is tightly related to very close to the conjunction of the *product space conditional event algebra* introduced in [11]. Detailed discussion of embeddings of existing algebras of conditional events into (TL|TL) is included in the companion paper [31]. Our new conjunction, denoted $\wedge^\star$, is defined precisely when at least one of its arguments is defined, so it resembles $\wedge_{\mathrm{SAC}}$ in this respect, but instead of assigning the other argument a default value when it is undefined, like SAC does, it uses its most recent defined value, instead. However, when the other argument hasn't ever been defined, it is assumed to act like *false*.

In the language of (TL|TL) $(a|b) \wedge^\star (c|d)$ can be expressed by

$$((b^{\complement}\,\mathsf{Since}(a \wedge b)) \wedge (d^{\complement}\,\mathsf{Since}(c \wedge d))|b \vee d).$$

5.3 Conclusion

We believe that there is no reason to restrict our attention to any particular choice of an operation extending the classical conjunction, and call is *the conjunction of conditionals.* There are indeed many reasonable such extensions, which correspond to different intuitions and situations, they can coexist in a single formalism, and any restriction in this respect necessarily narrows the applicability of the formalism.

We believe that neither of the choices discussed in this paragraph is *the* conjunction of conditionals. There are indeed many possible choices, and all of them have their own merits. In fact already the original system of Schay consisted of five operations: $\sim, \wedge_{\mathrm{SAC}}, \vee_{\mathrm{SAC}}, \wedge_{\mathrm{Sch}}$ and $\vee_{\mathrm{Sch}}$. Moreover, he was aware that these operations still do not make the algebra functionally complete (even in the narrowed sense, restricted to defining only operations which are undefined for all undefined arguments). And in order to remedy this he suggested to use one of several additional operators, one of them being $\wedge_{\mathrm{GNW}}$! So for him all those operations could coexist in one system.

6 Three prisoner's puzzle

In order to demonstrate that our formalism allows for a precise treatment of problems with conditioning and probabilities, let us consider the following classical example of a probabilistic "paradox". We will take this opportunity to highlight some of the practical issues of modelling using (TL|TL) and Moore machines approach. Therefore our analysis will be very detailed.

6.1 The puzzle

The three prisoner's puzzle [27] is the following:

> Three prisoners are sentenced for execution. One day before their scheduled execution, prisoner A learns that two of them have been pardoned. A calculates a probability of 2/3 for him being pardoned. Then he asks the Guard: "Name me one of my fellows who will be executed". The Guard tells him, that B will be executed. Based on that information, A recalculates the probability of being pardoned as 1/2, since now only one pardon remains for him and C (the third prisoner) to share! However, he could apply the same argument if the Guard named C. Furthermore, he knew beforehand that at least one of his fellows will be pardoned — so what did he gain (or lose) by the answer?

The intuitive explanation is that after learning the Guard's testimony $G(B)$ that B will be executed, A should revise the probability of the event $P(A)$ (of him being pardoned) by computing $P(\bullet P(A)|G(B))$, and the probability evaluation yields in this case 2/3, as expected.
However, what he indeed calculated was $P(\bullet G(B)|P(A))$, assuming effectively that the pardon had been given with equal probabilities to all the pairs possible *after* Guard's testimony. This probability turns out to be 1/2.

6.2 Probability tree model

First we present a simple probability tree analysis of the paradox, using the method which originates with Huygens [19, 29] and is indeed almost as old as mathematically rigorous probability theory itself. We begin in the leftmost circle (before pardon), then each of the three pardoned pairs leads us to three next circles, indicating the situation after the pardon. Finally, we have all the possible testimonies of the Guard. All edges originating from the same circle are equiprobable. After Guard's testimony $G(B)$, only the two top circles on the right are possible, and their probabilities are in the proportion 2 : 1, the more probable one being the one in which A is pardoned, while he is executed in the other one. So indeed even after the testimony the probability that A is pardoned remains 2/3.

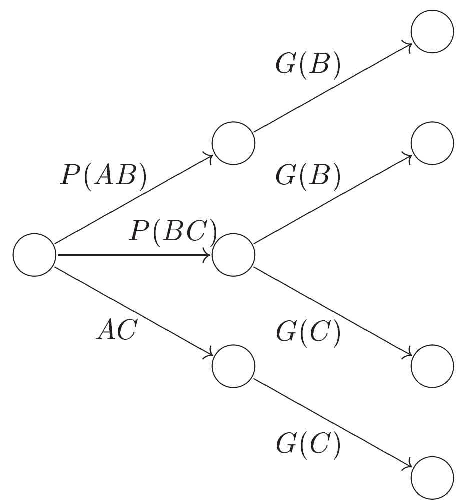

Figure 3: Probability tree analysis of the three prisoner puzzle.

6.3 (TL|TL) and Moore machine models

However, the tree shown above strongly resembles a Moore machine. And indeed, we augment it with the necessary details below. The most substantial change is that the Moore machine requires the same set of atomic possibilities is given at each state, which determine the next transition. Therefore:

- The Guard testifies something irrelevant while the court decides the pardons, and the court decides something irrelevant while the Guard testifies. This change is made invisible by our convention of collapsing transitions and applying subsequently Boolean algebra simplifications, except that

- In cases when the Guard has no choice, we must replace the existing transition label by the full event, because the Guard has prescribed answer no matter whom he would like to name,

- And except that we have to decide about transitions from the states which are terminal in the tree model. Because we believe that after being pardoned nobody can be prosecuted again for the same crime, and we do not believe in reincarnation, either, our choice is to use self-loops in the terminal states, yielding a "Russian roulette" model.

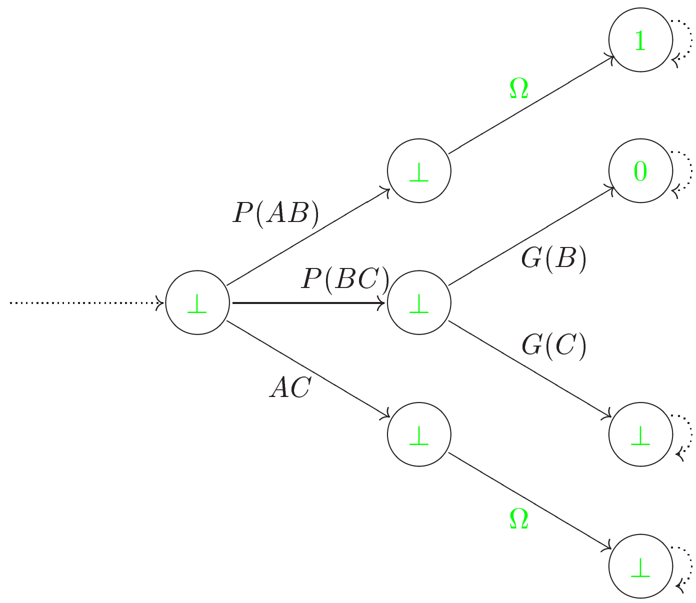

Figure 4: Probability tree analysis of the three prisoner puzzle with extensions necessary to convert the diagram into a Moore machine.

This provides a next piece of evidence that our definition of conditional events is natural and close to intuitions. In fact, one can embed the whole probability tree model into the formalism of Russian roulette Markov chains [29], and thus shows that our model of conditionals extends the method of probability trees.

Next we attempt to model the same paradox syntactically in (TL|TL). The construction of a correct (TL|TL) representation is a little bit more complicated than the formula $P(\bullet\, P(A)|G(B))$ we have suggested previously, as this requires specifying the actions of the Guard, whose probabilities are affected by the pardon decision. So we assume that the Guard always tosses a coin. If he gets heads (H), he tells the alphabetically first name among those applicable, and in case of tails (T) the alphabetically last among them. This indicates the need to consider the strategy followed by the Guard. And in fact, the probabilities A calculates depend on what he assumes about this strategy. So indeed now the answers of the Guard are shorthands for the combinations of the pardon decision and the coin toss outcome. Therefore $G(B)$ is $(\bullet\, P(AB) \wedge (H \vee T)) \vee (\bullet\, P(BC) \wedge H)$.

Moreover, we have to decide what should be modelled by the conditional object, and what by the probability assignment, which turns the former into a stochastic process. The general rule is that the more of the modelling is encoded in the probability assignment, the simpler the conditional and

its Moore machine are. On the other hand, encoding everything in the probability distribution is difficult and prone to errors, as the example of the poor prisoner shows. An, needless to say, a good model is one in which the proportions are just right. More on that below.
So formally the conditional looks now as follows:

$$\big((\bullet\, AB) \vee (\bullet\, AC) \,\big|\, ((\bullet\, AB) \wedge (H \vee T)) \vee ((\bullet\, BC) \wedge H)\big)\,, \tag{8}$$

with $\mathcal{E} = \{P(AB), P(BC), AC, H, T\}$, where the events $P(AB)$, $P(BC)$ and AC mutually exclusive and equiprobable, and similarly H and T mutually exclusive and equiprobable. (Our construction will easily handle non-equal probabilities, i.e., biased pardon decision and/or biased coin, too.) So the set Ω of atomic events is $\{AB\,H, AB\,T, BC\,H, BC\,T, AC\,H, AC\,T\}$, and these events are equiprobable under our probability assignment. However, we will be able to calculate the probability of (8) without the equiprobability assumption, too.
Note that, e.g., assuming events A, B and C to be nonexclusive individual pardon decisions of probability $1/3$ each, leads to more complicated conditional expression, because a substantial amount of coding effort must used just to ensure that always precisely two prisoners are pardoned. This makes the Moore machine more complicated, too. So this is certainly not a good model, because what can be easily taken care of by the probability assignment is instead modelled by logical methods. Such a model can be of course correct,[2] but good means for us more than just correct.
But if we attempt to draw the Moore machine of our conditional, we discover that it is quite different from that on Fig. 4.
The overall structure of the Moore machine is as follows: The entry states and transitions are dotted. Each of the three lines of three states (they form roughly edges of a triangle), consists of states with the same, already known pardon decision in the next experiment, while the current experiment's outcome is represented as the label of the state. Transitions are shown for one state on each edge only, because their targets depend on the input only, and not on the source within that edge. And this is why we can calculate the probability of (8) in a quite straightforward way. For time greater than 1 the probability of getting in two steps to a state with a given label does not depend on the current state nor on the time. Essentially, after the first step the edge of the triangle is chosen, which corresponds to the move to one of the states in the middle column of Fig. 4. In the second step we move to the state with the label equal to the destination label from Fig. 4, and the edge it is found within depends on the next experiment, already. The similarity

[2] Although unnecessary complications certainly increase the risk of mistakes and make verification of the model harder.

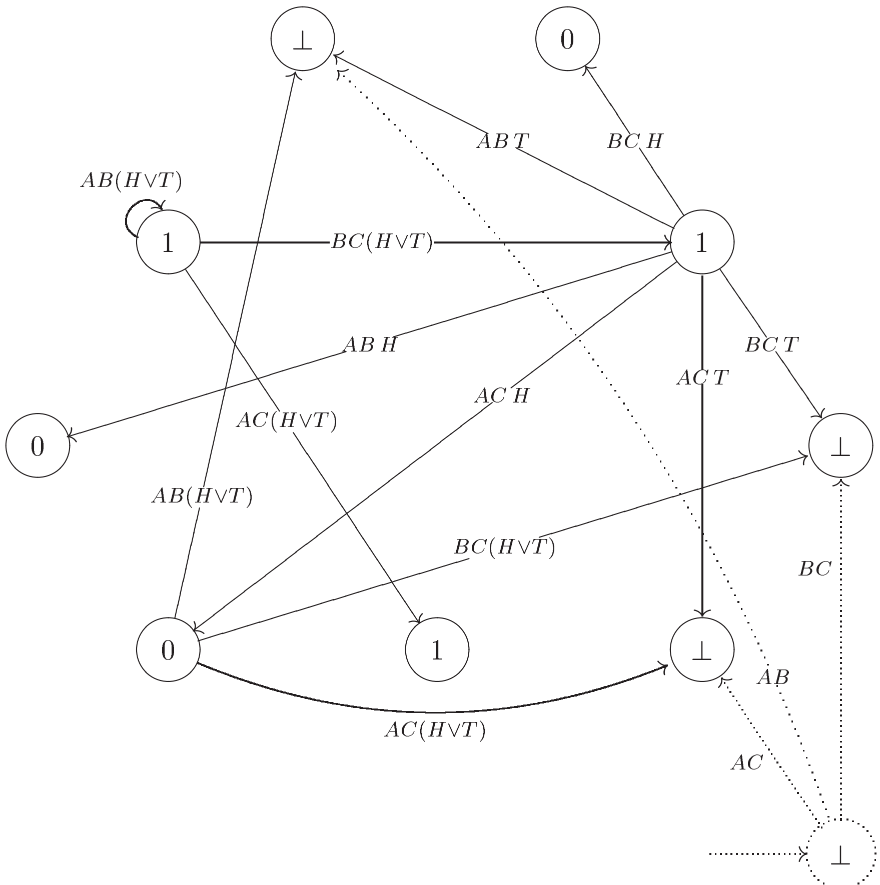

Figure 5: Moore machine corresponding to formula (8).

is even stronger if we compare Fig. 5 with Fig. 9 rather than with Fig. 4. A formal calculation, using matrix calculus, can be found in Section 6.4 below. The most substantial difference is that (8) is not a “Russian roulette” model! To note this set time to 3 and see: the present outcomes depend on the pardon decisions made at time 2, while the Guard was testifying in the previous round of the experiment, and while we are hearing the testimony of the Guard now, the pardons are already decided as a part of the next experiment. So the probabilistic choices which we described as irrelevant for the Moore machine model, are parts of the previous/next repetition schema here. The overlapping experiments do not interfere, however, so this does not affect probabilities. Furthermore, all the final outcome undefined values have been merged into one state. Finally, there are entry states which are visited just once and correspond to the situation at time 1, when the Guard says something, but there is no pardon decision to compare it with.
A modified version of (8), which is Russian roulette, is as follows:

$$((@_1 AB) \vee (@_1 AC)\,|((@_1 AB) \wedge @_2(H \vee T)) \vee ((@_1 BC) \wedge @_2 H))\,, \qquad (9)$$

where $@_1\alpha$ is $\blacklozenge(\neg \bullet \mathit{true} \wedge \alpha)$ and $@_2\alpha$ is $\blacklozenge(\bullet \mathit{true} \wedge \neg \bullet \bullet \mathit{true} \wedge \alpha)$, and express that α is true at time 1 and 2, respectively.

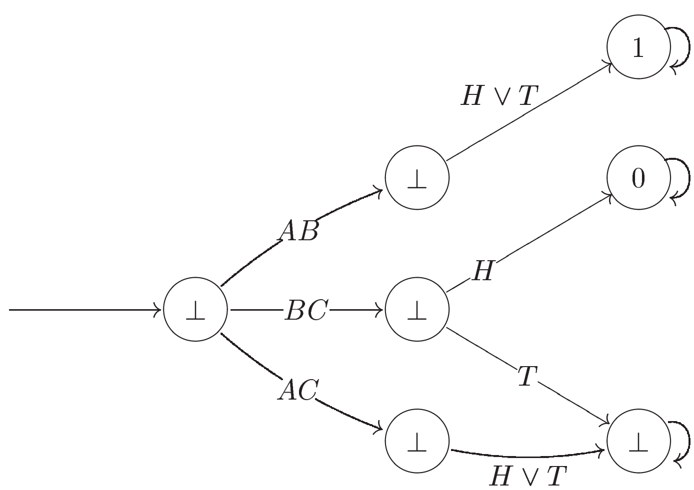

Figure 6: Moore machine of (9). It is the minimalization of the Moore machine from Fig. 4, so they are indeed logically indistinguishable.

The general conclusion is that simple Moore machines can correspond to complicated (TL|TL) formulas, and simple (TL|TL) descriptions can yield complicated Moore machines. If we additionally take into account that it is hard to expect that any computer program will be ever able transform human-readable representations of one kind to human-readable representations of the other kind[3], we recommend that the whole process of modelling is done using only one of the formalisms, without mixing them.

6.4 Algorithm for calculating the probability

Of course, the natural method to compute probability of a given regular conditional c in our model is to refer to an underlying Markov chain $\mathcal{X}$, perform the computations there, and then use the formula

$$\Pr(c) = \frac{\sum_{i:h(i)=1} \lim_{n\to\infty} \Pr(X_n = i)}{\sum_{i:h(i)=1 \text{ or } 0} \lim_{n\to\infty} \Pr(X_n = i)},$$

which follows directly from the Bayes' Formula.

The calculation of $\lim_{n\to\infty} \Pr(X_n = i)$ is generally known to be polynomial time in the number of states of the Markov chain, assuming unit cost of arithmetical operations [20]. The book [30] contains the account of state-of-the-art algorithms for numerical calculations of the limiting probabilities.

As an example we calculate here the probability of the formula (8), using the simplest possible approach, assuming all the events from Ω have nonzero probability.

We assume the following numbering of the states of the Markov chain from Fig. 5:

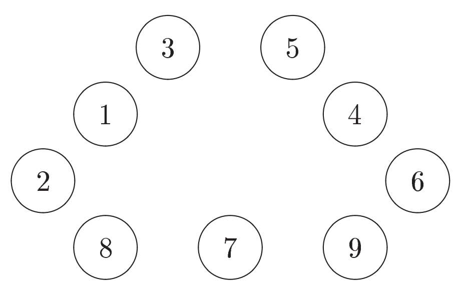

Figure 7: Numbering of the states of Markov chain resulting from the Moore machine in Fig. 5.

[3]In both cases even graphical layout can have a huge impact on the readability of the model!

Then the matrix Π of transition probabilities is

$$\begin{bmatrix}
\bar{AB} & 0 & 0 & \bar{BC} & 0 & 0 & \bar{AC} & 0 & 0 \\
\bar{AB} & 0 & 0 & \bar{BC} & 0 & 0 & \bar{AC} & 0 & 0 \\
\bar{AB} & 0 & 0 & \bar{BC} & 0 & 0 & \bar{AC} & 0 & 0 \\
0 & \bar{AB}\,\bar{H} & \bar{AB}\,\bar{T} & 0 & \bar{BC}\,\bar{H} & \bar{BC}\,\bar{T} & 0 & \bar{AC}\,\bar{H} & \bar{AC}\,\bar{T} \\
0 & \bar{AB}\,\bar{H} & \bar{AB}\,\bar{T} & 0 & \bar{BC}\,\bar{H} & \bar{BC}\,\bar{T} & 0 & \bar{AC}\,\bar{H} & \bar{AC}\,\bar{T} \\
0 & \bar{AB}\,\bar{H} & \bar{AB}\,\bar{T} & 0 & \bar{BC}\,\bar{H} & \bar{BC}\,\bar{T} & 0 & \bar{AC}\,\bar{H} & \bar{AC}\,\bar{T} \\
0 & 0 & \bar{AB} & 0 & 0 & \bar{BC} & 0 & 0 & \bar{AC} \\
0 & 0 & \bar{AB} & 0 & 0 & \bar{BC} & 0 & 0 & \bar{AC} \\
0 & 0 & \bar{AB} & 0 & 0 & \bar{BC} & 0 & 0 & \bar{AC}
\end{bmatrix}$$

where $\bar{AB}$ stands for $\Pr(AB)$, and similarly for arguments BC, AC, H, T (the matrix does not fit into the page when the standard notation is used). It can be directly checked that the square of this matrix has all entries nonnegative, hence the whole represents a single ergodic class. (This is what breaks down when some elements from Ω have probability 0. It this is permitted, one has to consider a few more cases.) It is known that in such cases the limiting probability does not depend on the initial probabilities of getting into this class, therefore we can ignore the dotted (transient) states from Fig. 5. The limiting probabilities can be found, given $\Pi = (p_{ij})$, by finding the only solution of the system of linear equations

$$\begin{cases}
\sum_{i=1}^{9} x_i & = & 1, \\
\sum_{i=1}^{9} p_{i1} x_i & = & x_1, \\
\sum_{i=1}^{9} p_{i2} x_i & = & x_2, \\
\cdots & = & \cdots \\
\sum_{i=1}^{9} p_{i9} x_i & = & x_9,
\end{cases}$$

which yields the following unique solution:

$$\begin{array}{lll}
x_1 = \bar{AB}^2 & x_2 = \bar{BC}\,\bar{AB}\,\bar{H} & x_3 = \bar{AB}(1 - \bar{AB} - \bar{BC}\,\bar{H}) \\
x_4 = \bar{AB}\,\bar{BC} & x_5 = \bar{BC}^2\,\bar{H} & x_6 = \bar{BC}(1 - \bar{AB} - \bar{BC}\,\bar{H}) \\
x_7 = \bar{AC}\,\bar{AB} & x_8 = \bar{BC}\,\bar{AC}\,\bar{H} & x_9 = 1 - \bar{AB}(1 + \bar{AC} + \bar{BC}\,\bar{H}) - \bar{BC}
\end{array}$$

and the de Finetti probability of the conditional represented by the Moore machine in question is $\dfrac{\Pr(AB)}{\Pr(BC)\Pr(H) + \Pr(AB)}$, as expected. In particular, in the equiprobable case the value is 2/3.

7 Related work and possible extensions

7.1 Related work

- Using temporal logic in reasoning about knowledge is nothing new. Indeed, many logics of knowledge incorporate temporal operators, see [8]. However, to the best of our knowledge, (TL|TL) is the very first multi-valued temporal logic to be considered. In particular, the above mentioned logics of knowledge are two-valued. Moreover, (TL|TL) is the first natural use of past tense temporal logic in computer science. Most of the established formalisms which use propositional temporal logic, indeed use its future tense fragment.

- Computing of conditional probabilities $\Pr(\varphi|\psi)$ is nothing new, either, and has been considered by several authors in depth, including [23, 15, 14], mostly for first order logic or unordered structures.

- Finally, Markov chains have already been used for evaluation of probabilities of logical statements. In particular, our Bayes' Formula is a simple extension of a theorem of Ehrenfeucht (see [24]), phrased there as a theorem about first order logic of ordered unary structures (over which first order logic is equally as expressive as propositional temporal logic, see [7]).

7.2 Possible extensions.

- (TL|TL) is not closed under its own connectives, since the nesting of the conditioning operator $(\cdot|\cdot)$ with other connectives (let alone itself) is not allowed, and since the temporal connectives cannot be applied to a conditional pair. As a consequence, operations on conditionals are defined by disassembling the pairs and reassembling them afterwards, to yield a pair in the correct syntactical form again.

  We would like to have an equivalent logic with much better syntactical structure. This should be possible by extending the ideas of multivalued modal logics, investigated in [26, 9, 10], by a multivalued counterparts of Since. The logic would then assume the form of a propositional logic with multivalued temporal connectives and conditioning.

  The big question is whether one can retain the Bayes' Formula then. The existing attempts in the present tense logics of conditionals suggest it might be difficult.

- (TL|TL) does not match exactly the class of automata, which for any assignment of probabilities yield a Markov chain with all states either transient or aperiodic. In such Markov chains all the limiting probabilities do exist, and thus every such Markov chain can be meaningfully considered to represent an extended kind of a conditional. Indeed, below is a simple example of such an automaton.

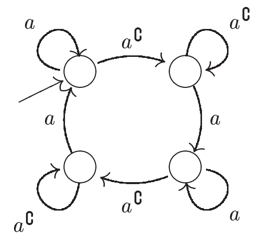

Figure 8: It is not hard to verify that, no matter what probability is assigned to the event a, the resulting Markov chain has only transient and acyclic states. However, the automaton is not acyclic, since it has two states, reachable by a path labelled $aa^{\complement}$ from each other.

We would like to have an extension of (TL|TL), matching exactly the class of Markov chains with only transient and aperiodic states, to take the advantage of the maximal class of Markov chains for which the limiting probabilities exist, and thus all the definitions given in the paper make sense. We expect the logic to be obtained by extending the multivalued temporal logic proposed suggested above, rather than by extending the present syntax.

- In the forthcoming paper [32] we consider probabilities of strange conditional events, analyzing the bahaviour of their associated so-called sub-Markov chains. Our results are similar to those in [21]. In particular, this work extends the results of [23, 15, 14] to first order logic of *ordered* unary structures, per the equivalence mentioned above.

Acknowledgement. The first author wishes to thank Igor Walukiewicz for valuable informations concerning temporal logic.

References

[1] E. W. Adams. On the logic of high probability. *J. Philos. Logic*, 15(3):255–279, 1986.

[2] G. Boole. *An investigation of the laws of thought, on which are founded the mathematical theories of logic and probabilities.* Dover Publications, Inc., New York, 1957.

[3] P. Calabrese. An algebraic synthesis of the foundations of logic and probability. *Inform. Sci.*, 42(3):187–237, 1987.

[4] P. G. Calabrese. A theory of conditional information with applications. *IEEE Trans. Systems Man Cybernet.*, 24(12):1676–1684, 1994. Special issue on conditional event algebra (San Diego, CA, 1992).

[5] P. G. Calabrese. Conditional events: doing for logic what fractions do for integer arithmetic. Presented at the International Conference on the Notion of Event in Probabilistic Epistemology, University of Trieste, Italy, May 1996, 1997.

[6] B. de Finetti. *Probability, induction and statistics. The art of guessing.* John Wiley & Sons, London-New York-Sydney, 1972. Wiley Series in Probability and Mathematical Statistics.

[7] E. A. Emerson. Temporal and modal logic. In *Handbook of theoretical computer science, Vol. B*, pages 995–1072. Elsevier, Amsterdam, 1990.

[8] R. Fagin, J. Y. Halpern, Y. Moses, and M. Y. Vardi. *Reasoning about knowledge.* MIT Press, Cambridge, MA, 1995.

[9] M. Fitting. Many-valued modal logics. II. *Fund. Inform.*, 17(1-2):55–73, 1992.

[10] M. C. Fitting. Many-valued modal logics. *Fund. Inform.*, 15(3-4):235–254, 1991.

[11] I. R. Goodman. Toward a comprehensive theory of linguistic and probabilistic evidence: two new approaches to conditional event algebra. *IEEE Trans. Systems Man Cybernet.*, 24(12):1685–1698, 1994. Special issue on conditional event algebra (San Diego, CA, 1992).

[12] I. R. Goodman, R. P. S. Mahler, and H. T. Nguyen. *Mathematics of data fusion.* Kluwer Academic Publishers Group, Dordrecht, 1997.

[13] I. R. Goodman, H. T. Nguyen, and E. A. Walker. *Conditional inference and logic for intelligent systems.* North-Holland Publishing Co., Amsterdam, 1991. A theory of measure-free conditioning.

[14] A. J. Grove, J. Y. Halpern, and D. Koller. Asymptotic conditional probabilities: the non-unary case. *J. Symbolic Logic*, 61(1):250–276, 1996.

[15] A. J. Grove, J. Y. Halpern, and D. Koller. Asymptotic conditional probabilities: the unary case. *SIAM J. Comput.*, 25(1):1–51, 1996.

[16] A. Hájek and N. Hall. The hypothesis of the conditional construal of conditional probability. In *Probability and conditionals*, pages 75–111. Cambridge Univ. Press, Cambridge, 1994.

[17] N. Hall. Back in the CCCP. In *Probability and conditionals*, pages 141–160. Cambridge Univ. Press, Cambridge, 1994.

[18] J. E. Hopcroft and J. D. Ullman. *Introduction to automata theory, languages, and computation*. Addison-Wesley Publishing Co., Reading, Mass., 1979. Addison-Wesley Series in Computer Science.

[19] C. Huygens. *Oeuvres complétes*, volume 14, pages 151–155. Martinus Nijhoff, La Haye, 1920. Transcription of a manuscript written in August 1676.

[20] J. G. Kemeny and J. L. Snell. *Finite Markov chains*. Springer-Verlag, New York-Heidelberg, 1976. Reprinting of the 1960 original, Undergraduate Texts in Mathematics.

[21] H. Kesten. A ratio limit theorem for (sub) Markov chains on $\{1, 2, \cdots\}$ with bounded jumps. *Adv. in Appl. Probab.*, 27(3):652–691, 1995.

[22] D. Lewis. Probabilities of conditionals and conditional probabilities. *Philos. Review*, 85:297–315, 1976.

[23] M. I. Liogon'kiĭ. On the conditional satisfiability ratio of logical formulae. *Mat. Zametki*, 6:651–662, 1969.

[24] J. F. Lynch. Almost sure theories. *Ann. Math. Logic*, 18(2):91–135, 1980.

[25] V. McGee. Conditional probabilities and compounds of conditionals. *Philosophical Review*, 4:485–541, 1989.

[26] C. G. Morgan. Local and global operators and many-valued modal logics. *Notre Dame J. Formal Logic*, 20(2):401–411, 1979.

[27] J. Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann, San Mateo, CA, 1988.

[28] A. Ramer. Combinatorial interpretation of uncertainty and conditioning. In G. Antoniou, editor, *Learning and Reasoning with Complex Representations*, volume 1359 of *LNCS*, pages 248–255. Springer Verlag, 1998.

[29] G. Shafer. *The art of causal conjecture.* MIT Press, Cambridge, MA, 1996.

[30] W. J. Stewart. *Introduction to the numerical solution of Markov chains.* Princeton University Press, Princeton, NJ, 1994.

[31] J. Tyszkiewicz, A. Hoffmann, and A. Ramer. Embedding conditional event algebras into temporal calculus of conditionals. Manuscript, 1999.

[32] J. Tyszkiewicz, A. Hoffmann, and A. Ramer. Strange conditional events and subMarkov chains. In preparation, 1999.

[33] B. C. van Fraassen. Probabilities of conditionals. In *Foundations of probability theory, statistical inference, and statistical theories of science (Proc. Internat. Res. Colloq., Univ. Western Ontario, London, Ont., 1973), Vol. I*, pages 261–308. Univ. Western Ontario, Ser. Philos. Sci., Vol. 6. Reidel, Dordrecht, 1977. With a discussion by the author and R. N. Giere, a letter to the author by R. Stalnaker, and a response.