Abstract. This report defines the basic concepts of rational probable logic (RPL). Rational probable logic combines classical logic and the elementary mathematical theory of probability with the intention of predicting probable inferences. It is rational because the probability measures that are of primary concern all have finite sample spaces for which probability is computed as a rational fraction: $\text{prob}(A) = |A| / \text{size of sample space}$. Relationships with classical logic, computational logic, and probability theory are discussed. There are several interesting theoretical conjectures regarding the decidability of rational probable logic that are difficult for this author and his students. A novel inference method is introduced that uses abstract interpretations of rational probable theories: Monte Carlo methods are used to generated random minimizing models for a theory. Probability estimates for events of the theory are then estimated using the outcomes for these abstract interpretations. This presentation is a preliminary report.

1 Basic definitions

A rational probable logic clause has the form

$$r_1 \leq A_1 \cup A_2 \cup \ldots \cup A_k \leq r_2$$

where the $A_i$ are event, or set, symbols, $\cup$ represents union, and $r_1$ and $r_2$ are rational number and $0 \leq r_1 \leq r_2 \leq 1$.

Think of the displayed clause as saying that a joint event has a probability lying between the two rational numbers $r_1$ and $r_2$.

A rational probable logic theory $T$ consists of a finite set of rational probable clauses.
For example, consider the theory $T_1$ given by the following three clauses

$$
\frac{2}{3} \leq A \\
\frac{1}{3} \leq B \\
1 \leq \sim A \cup \sim B
$$

We use $\sim A$ to denote the complement of set $A$ (not $A$). The intention of this little theory is to say that there are two events, $A$ and $B$, which have the following properties: $A$ has probability at least $2/3$, $B$ has probability at least $1/3$, and the events $A$ and $B$ are disjoint. Absent upper bounds are presumed to be 1; absent lower bounds would similarly be presumed to be 0.

A word regarding notation: It might seem more natural to write the probable clauses in a more typical way, something like

$$
Pr(\sim A \cup \sim B) \geq 1
$$

for example. However, the extra notation seems unnecessary. The interpretations of the theory will involve the probability concept. Think of a probable logic theory itself as a set of abstract symbolic formulas whose probability meaning is revealed in a more precise manner under interpretation.

The event literal set associated with a theory consists of the set of event symbols that appear in the theory. In the previous example, the event literal set is $\{A, B\}$. Note that the event literal set for a theory is finite.

An interpretation $I$ of a probable theory $T$ consists of a finite sample space (a set) $S$ along with a function which assigns elements in the event literal set of $T$ to events in the sample space $S$ (subsets of $S$). If $A$ is an event symbol then $I(A)$ refers to the assigned event (subset) of $S$. It is required that

$$
I(\sim A) = S - I(A)
$$

for all event symbols $A$. The ‘$-$’ symbol refers to ordinary set difference: all elements of set $S$ not in set $I(A)$.

For example, Let $S = \{1,2,3,4,5,6\}$ be the sample space and assign $A$ and $B$ as follows:

$I(A) = \{1,3,5,6\}$, $I(B) = \{2,4\}$. Note then that $I(\sim A) = \{2,4\}$ and $I(\sim B) = \{1,3,5,6\}$.

The probable clause (1) is true for the interpretation $I$ provided that
The ratio or fraction in the middle term of these inequalities is the relative frequency probability of the union in the sample space of the interpretation:

- $O = |I(A_1) \cup I(A_2) \cup \ldots \cup I(A_k)|$ is the size of the union of the events in the interpretation
- $p = O / |S|$ is the relative sample probability of outcome $O$ in sample space $S$
- $r_1 \leq p \leq r_2$ says that relative probability $p$ lies within these bounds

Otherwise, if the clause is not true, then the clause is said to be false for the given interpretation. We also use the terminology that a clause holds when it is true. In addition, the notation $Pr_I(A)$ to denote the relative probability assigned to the event symbol $A$ by interpretation $I$.

$$Pr_I(A) = |I(A)| / |S|$$

It is easy to check that, for the theory $T_1$, the previously specified interpretation makes all the probable clauses of the theory true:

- $Pr_I(A) = |\{1,3,5,6\}| / 6 = 4/6 = 2/3$
- $Pr_I(B) = |\{3,4\}| / 6 = 2/6 = 1/3$
- $Pr_I(\neg A \cup \neg B) = |I(\neg A) \cup I(\neg B)| / 6 = |\{2,4\} \cup \{1,3,5,6\}| / 6 = 1$

So that interpretation is a model for the theory. A model for a probable logic theory is any interpretation that makes all the probable clauses of the theory true. Any theory having a model is said to be consistent. A theory with no model is said to be inconsistent.

Here is an example of an interpretation which is not a model. Let $S$ be as before, but this time assign $A$ to $\{1,2,3,4\}$ and $B$ to $\{3,6\}$. Note that the third clause is false under this interpretation.

Given a probable theory $T$ we say that a clause $C$ in the theory is independent provided that there is some interpretation of the theory that makes all of the other clauses true but makes $C$ false. If a clause is not independent it is said to be dependent.

Thus, the third clause of the theory in the previous example is independent. It is easy to show that all of clauses of the example theory are independent.

An example of a theory having a dependent clause would be the following $T_2$.
Notice that the second clause is dependent: any interpretation making the first clause true must necessarily also make the second clause true. On the other hand, the first clause in $T_2$ is independent.

Suppose that $T$ is a consistent theory and that $A$ is an event literal of $T$. Define the *minimal expectation* $e(A)$ as follows.

$$e(A) = \text{g.l.b.} \{ \Pr_M(A) \mid M \text{ is a model of } T \}$$

where ‘g.l.b.’ is the greatest lower bound (a real number) of the set of rational numbers. The real number $e(A)$ is well defined if $T$ is consistent, but it is an open question as of this writing whether $e(A)$ can ever be irrational.

A *positive* rational probable logic theory is one all of whose clauses have the form

$$r_1 \leq A_1 \cup A_2 \cup \ldots \cup A_k \leq r_2$$

where

- $r_2 = 1$ and
- *exactly one* of the set literals $A_i$ is positive.

Such a probable logic clause can thus be conveniently written in either the form

$$r \leq A$$

in case there is exactly one set literal, or

$$r \leq A_1 \leftarrow A_2, \ldots, A_k$$

in the more general case where we may assume that $A_1$ is the one positive literal and the others are negative, where the expression $A_1 \leftarrow A_2, \ldots, A_k$ is shorthand for

$$A_1 \cup \neg A_2 \cup \ldots \cup \neg A_k$$

These logical expressions are called *Horn formulas* [2] in logic (where $\cup$ is logical disjunction and $\leftarrow$ is formal implication). The expressions are logically equivalent.
Here is an example of a positive theory $T_3$:

$$
\begin{align*}
1/2 & \leq A \\
1/3 & \leq B \\
1 & \leq C \iff A \\
1 & \leq C \iff B \\
1 & \leq D \iff A, B
\end{align*}
$$

First, let us rewrite the clauses of $T_3$ using our original notation:

$$
\begin{align*}
1/2 & \leq A \leq 1 \\
1/3 & \leq B \leq 1 \\
1 & \leq C \cup \neg A \leq 1 \\
1 & \leq C \cup \neg B \leq 1 \\
1 & \leq D \cup \neg A \cup \neg B \leq 1
\end{align*}
$$

Hopefully, the reader understands why we prefer the sparser representation.

Here is an intuitive "reading" of the probable clauses of $T_3$:

- Event $A$ has probability at least $1/2$
- Event $B$ has probability at least $1/3$
- It is certain (probability = 1) that $C$ if $A$
- It is certain that $C$ if $B$
- It is certain that $D$ if both $A$ and $B$

We will return to this example $T_3$ later.

It is easy to invent theories that do not have any model. For example, $T_4$

$$
\begin{align*}
2/3 & \leq A \\
2/3 & \leq B \\
1 & \leq \neg A \cup \neg B
\end{align*}
$$

Why? Note that the third clause says that events $A$ and $B$ would be disjoint in a model, and so the two events could not both have probability at least $2/3$. Note that $T_4$ is not positive.

General probable logic theories can easily be inconsistent, in the probable logic sense. For positive theories the situation is very different: Any positive rational probable logic theory has a model. We will expand on this in the next section.

In the remainder of this document, RPL will be shorthand for rational probable logic as specified by the definitions in this first section.
2 Logic and resolution

It is important to make a comparison with computational logic and clausal resolution, even though we believe that these methods of computational logic will ultimately turn out to be essentially deficient. Certain connections need to be reviewed in order understand the deterministic approaches and to move away from their limitations.

A disjunctive logic clause has the form

\[ A_1 \lor A_2 \lor \ldots \lor A_k \]  \hspace{1cm} (11)

Where ‘\( \lor \)’ denotes logical disjunction.

If \( T \) is a probable logic theory define

\[ L(T) = \{ A_1 \lor A_2 \lor \ldots \lor A_k \mid r_1 \leq A_1 \cup A_2 \cup \ldots \cup A_k \leq r_2 \ \text{belongs to} \ T \} \]  \hspace{1cm} (12)

\( L(T) \) is the finite set of logical disjuncts corresponding to unions appearing in \( T \).

Using our previous example, \( L(T_4) \) consists of the following disjunctive clauses

\[ \begin{align*}
A \\
B \\
\neg A \lor \neg B
\end{align*} \]

We are using the ‘\( \neg \)’ symbol here for both logical negation and set complement

Note that \( L(T_4) \) is logically inconsistent but that the probable logic theory \( T_4 \) certainly does have probable logic models.

Logic Resolution inference rule. The disjunctive clause \( B \lor C \) is a logical consequence of logical clauses \( A \lor B \) and \( \neg A \lor C \). (\( B \) and \( C \) themselves are also disjunctive clauses, \( A \) is a logical symbol.) This is often expressed in the following tabular form:

\[ \begin{align*}
A \lor B \\
\neg A \lor C \\
\hline
B \lor C
\end{align*} \]

\( B \lor C \) is the resolvent of the clauses above the line. J.A. Robinson was the key inventor of logic resolution [6]. There is a very similar appearing inference rule for probable logic.

A famous theorem states that a set of disjunctive logic clauses is logically inconsistent if, and only, if there is a resolution deduction of NULL, the empty clause, using the clauses in the set. Here is a pictorial representation of deduction of NULL for \( L(T_4) \)
There is an inference rule for probable clauses that is similar to resolution and can be considered as a kind of generalization of logical resolution.

**Probable Logic Resolution inference rule**

\[
\begin{align*}
    p & \leq A \cup B \leq q \\
    r & \leq \neg A \cup C \leq s
\end{align*}
\]

\[\begin{align*}
    u & \leq B \cup C \leq 1 \\
    v & \leq \neg B \cup \neg C \leq 1
\end{align*}\]

where

\[u = \max(0, p+r-1)\]

\[v = \min(1, 2-q-s)\]

The probable clauses below the line are said to be *probable resolvents* of the two clauses above the line. What this schema means is that any probable logic interpretation which makes both of the formulas above the line true also makes the formula below the line true. See the appendix for proofs.

The resolvent

\[v \leq \neg B \cup \neg C \leq 1\] (13)

is written as a disjunction since probable logic clauses use disjunctive expressions. It could also be expressed in a dual manner:

\[0 \leq B \cap C \leq q+s-1\] (14)

where \(B \cap C\) represents the *intersection* of \(B\) and \(C\). Note that \(1-(2-q-s) = q+s-1\).

Note that if \(q = 1\) and \(s = 1\) then \(v = 0\) and the resolvent

\[0 \leq \neg B \cup \neg C \leq 1\] (15)
gives no useful information. Thus, in particular, the second resolvent is irrelevant for positive probable logic theories.

The reader should test their understanding by inventing an example having equality:

\[
\begin{align*}
  u &= B \cup C \\
  v &= \neg B \cup \neg C
\end{align*}
\]

This does not mean that the formulas for \( u \) and \( v \) are functionally optimal. (It merely means that in some cases the boundary values are actually achieved for the given formulas.)

**Proposition 1.** Suppose that \( X \) and \( Y \) are events in a probability space, that \( \text{prob}(X) \geq x \) and \( \text{prob}(Y) \geq y \) and that \( f(X,Y) \) is a Boolean function containing \( X \cap Y \) as a subset. Then \( \text{prob}(f(X,Y)) \geq x + y - 1 \).

**Proof.**

\[
\text{prob}(f(X,Y)) \geq \text{prob}(X \ Y) = \text{prob}(X) + \text{prob}(Y) - \text{prob}(X \cap Y) \geq x + y - 1 \quad \diamondsuit
\]

**Proposition 2.** Suppose that \( A \) is an event in a probability sample space and that \( B \) and \( C \) are unions of events, and suppose that both of the following probability estimates hold

\[
\begin{align*}
  p &\leq \text{prob}(A \cup B) \leq q \\
  r &\leq \text{prob}(\neg A \cup C) \leq s
\end{align*}
\]

Then (a)

\[
u \leq \text{prob}(B \cup C) \leq 1
\]

and (b)

\[
v \leq \text{prob}(\neg B \cup \neg C) \leq 1
\]

where

\[
\begin{align*}
  u &= \max(0, p+r-1) \\
  v &= \min(1, 2-q-s)
\end{align*}
\]

**proof.** For (a) let \( X = A \cup B \) and \( Y = \neg A \cup C \). The reader can verify that \( B \cup C \) does indeed contain the subset \( X \cap Y \); now apply **Proposition 1**. Part (b) follows from the following fact: if \( x \leq A \cap B \) and \( y \leq \neg A \cap C \) then \( x + y \leq B \cup C \). The details are left to the reader. \( \diamondsuit \)

It is possible to define deduction using the probable logic version of resolution, in a manner similar to ordinary (binary) resolution deduction:
A probable resolution deduction from a probable logic theory $T$ consists of a sequence of probable clauses $C_1, C_2, \ldots, C_z$ such that each $C_j$ is either a clause of $T$ or is a probable logic resolution inference using two probable clauses $C_x$ and $C_y$ where $x$ and $y$ are smaller than $j$, or $C_j$ is a factor of some $C_i$, where $i < j$.

A factor of a clause removes any repeated literals. For example, consider the positive theory $T_4$

\[
\begin{align*}
1 & \leq C \leftarrow A, B \\
1 & \leq B \leftarrow A \\
1/2 & \leq A
\end{align*}
\]

A resolution deduction showing $1/2 \leq C$ is the following

\[
\begin{align*}
1 & \leq C \leftarrow A, B \\
2 & \leq B \leftarrow A \\
3 & \leq C \leftarrow A, A \ % \text{ resolve 1 and 2} \\
4 & \leq C \leftarrow A \ % \text{ factor 3} \\
5 & 1/2 \leq A \\
6 & 1/2 \leq C
\end{align*}
\]

Without factoring, one would have to resolve twice on clause 3 and get $1/2 + 1/2 - 1 = 0 \leq C$, too small an estimate.

In the following proposition $\phi$ refers to the empty event (set).

**Proposition 3.** If a probable resolution deduction sequence from probable logic theory $T$ contains a probable clause of the form ‘$\phi \geq r$’ where $r > 0$ then $T$ does not have a probable model. It is inconsistent.

Here is an example. Reconsider $T_5$

\[
\begin{align*}
2/3 & \leq A \\
2/3 & \leq B \\
1 & \leq \neg A \cup \neg B
\end{align*}
\]

And here is a probable deduction...

\[
\begin{align*}
1 & \leq \neg A \cup \neg B \\
2 & \leq 2/3 \leq A \\
3 & 2/3 \leq \neg B \ % \text{ resolve 1 and 2} \\
4 & 2/3 \leq B \\
5 & 1/3 \leq \phi \ % \text{ resolve 3 and 4}
\end{align*}
\]

The last clause (5) of this deduction shows that $T_5$ is an inconsistent probable logic theory. Here is a more graphical representation of this deduction.
Fig. 2 probable resolution deduction

Unlike the case for logical resolution there are inconsistent probable logic theories for which there is no resolution deduction of $\phi > 0$. $T_6$ is such an example:

$$
\begin{align*}
1 & \leq \neg A \cup \neg B \\
1 & \leq \neg A \cup \neg B \\
1 & \leq \neg B \cup \neg C \\
2/3 & \leq A \\
2/3 & \leq B \\
1/3 & \leq C
\end{align*}
$$

Note that the first three clauses say that A, B and C are mutually disjoint. Thus, they cannot have respective probabilities 2/3, 2/3, and 1/3. So cannot possibly have a model. But there is no probable resolution deduction demonstrating this fact. Thus probable resolution deduction is incomplete.

Now, of course, we could invent other inference rules, such as the following one, which we call expansion. A and B need to be set event literals:

$$
\begin{align*}
p & \leq \neg A \cup \neg B \\
q & \leq A \\
r & \leq B \\
\hline
q+r-p & \leq A \cup B
\end{align*}
$$

It is an interesting exercise to show that 1) the expansion rule is valid and 2) probable resolution together with this expansion rule suffice to derive a contradiction $\phi > 0$ from the probable theory $T_6$.

The goal of finding a complete set of inference rules for deciding consistency for probable logic theories is elusive, and probably unattainable.

P. Suppes [X] gave some rules similar in motivation to the characterization of probable resolution given in this section.
3 Standard RPL models

Consider again the positive RPL theory $T_3$:

\[
\begin{align*}
1/2 &\leq A \\
1/3 &\leq B \\
1 &\leq C <- A \\
1 &\leq C <- B \\
1 &\leq D <- A, B
\end{align*}
\]

The denominators of the bounding fractions are 1, 2, 3 and their least common multiple is 6. It seems reasonable to specify models having a sample space containing exactly 6 elements. Using sample space \{1,2,3,4,5,6\}, let $A = \{1,2,3\}$, $B = \{4,5\}$, $C = A \cup B = \{1,2,3,4,5\}$ and $D = \{\}$. The reader should verify that this gives a model for $T_3$.

Suppose that

\[ T = \{ r_1 \leq A_1 \cup A_2 \cup ... \cup A_k \leq r_2 \mid i = 1, ..., k \} \quad (17) \]

is a probable logic theory and that each $r_i = n_i/d_i$ expresses the fraction in lowest terms. Let $d$ be the least common multiple of the set of all of the denominators:

\[ d = \text{lcm} \{ d_i \} \quad (18) \]

Then a standard interpretation of $T$ is any interpretation over the set $d = \{1,...,d\}$. In a standard interpretation, event symbols are assigned to elements of $2^d$, the set of subsets of $d$. A standard model is any model over the set $d$.

There is nothing special about the set $d$ used to define a standard interpretation. Any set with $d$ elements would serve equally well.

An assertive probable logic theory is one for which all of the $r$'s are 1. For example the following theory $T_6$ is assertive:

\[
\begin{align*}
1 &\leq A \\
1 &\leq B \\
1 &\leq \sim A \cup \sim B \cup \sim C
\end{align*}
\]

Note that $T_6$ is not a positive theory, and that $T_6$ is consistent.

The following lemma shows that assertive theories are essentially logical theories, and provides some basic support for a claim that probable logic generalizes classical propositional logic.

**Proposition 4.** Suppose that $T$ is an assertive probable logic theory. Then the following are equivalent.
1. \( L(T) \) is logically consistent
2. \( T \) has a standard probable model
3. \( T \) has a probable model

**proof.** We show that 1 implies 2, 2 implies 3, and 3 implies 1...

1 \( \implies \) 2: Assume that \( L(T) \) is logically consistent. Then there is a logical interpretation \( i \) of the symbols of \( T \) such that each logical clause of \( L(T) \) is **true**. That is, if \( A_1 \lor A_2 \lor \ldots \lor A_k \) is a clause of \( L(T) \) then \( i(A_j) \) is **true** for at least one \( j = 1, \ldots, k \). That is, \( i \) is a logical model for \( L(T) \). Define a standard probable interpretation \( I \) of \( T \) as follows:

\[
I(A) = \emptyset \text{ if } i(A) = \text{false}, \text{ else } I(A) = \{1\} \text{ if } i(A) = \text{true}.
\]

Then \( I \) is a standard probable logic model for \( T \) (why).

2 \( \implies \) 3 is obvious.

3 \( \implies \) 1: Suppose that \( L(T) \) is not logically consistent. By a famous theorem of resolution logic, there must be some resolution deduction of NIL using the clauses of \( L(T) \). This means that there is a sequence \( C_1, C_2, \ldots, C_z \) of disjunctive clauses such that each \( C_j \) is either a clause of \( L(T) \) or is a resolvent of some \( C_k \) and \( C_l \) where \( k \) and \( l \) are smaller than \( j \), and \( C_z = \text{NIL} \). Thus there is also a sequence of probable resolvents ending with \( \phi \geq 1 \). If \( T \) had a probable model then this last probable clause would have to be true for that model, which is impossible. Thus not 1 \( \implies \) not 3, so 3 \( \implies \) 1. ♦

It is possible, of course, to concoct nonstandard interpretations for assertive theories. For example, consider the probable theory again. Define interpretation \( I: \{A,B,C\} \rightarrow \{1,2,3,4\} \) as follows: \( I(A) = I(B) = I(\neg C) = \{1,2,3,4\} \). \( I \) is a probable model of the theory over the set \( 4 = \{1,2,3,4\} \). We do not presume that such models are necessarily interesting. It is a curiosity, however, that having a nonstandard model requires a logic theory to be consistent!

**Proposition 5.** Suppose that \( T \) is a probable logic theory for which \( L(T) \) is logically consistent. Then \( T \) itself has a standard probable model.

**proof.** The proof is essentially the same as the proof of 1 \( \implies \) 2 for **Proposition 4**. The only difference is that the resulting probabilities (all 1s) are surely larger or equal to the rational numbers that actually appear in the original clauses of \( T \). ♦

The following conjecture, if true, would be a very important result. It would show, for example, that having a model (being satisfiable) for a probable logic theory is algorithmically decidable.

**Open Question 1.** Is the following statement true? Any rational probable logic theory \( T \) has a model if, and only if, \( T \) has a standard model.

Perhaps the best approach at this time would be to try to find a counterexample to the conjecture. Thus, we should look for some (hopefully small) example of a probable logic theory which has a model, but does not have a standard model.

The conjecture, if true, would certainly describe a beautiful mathematical relationship for finite set theory. Define a **finite set problem** to be a finite collection of expressions of the form
where the $A_i$ are set literals (set symbols or complements) and $a$ and $b$ are nonnegative integers with $a \leq b$. A finite set problem can be thought of as a Venn diagram involving the set symbols, together with the restrictions on the sizes of the set combinations given by the inequalities.

![Venn Diagram](image)

Fig 3. finite set problem with Venn diagram

Any finite set problem can be converted to an RPL theory: rewrite each expression in the form

$$a/L \leq A_1 \cup A_2 \cup \ldots \cup A_k \leq b/L$$  \hspace{1cm} (20)

where $L$ is the least common multiple of all the $a$'s and $b$'s from all of the expressions. Clearly, any standard model for the RPL theory can be used to construct a solution for the original finite set problem over a domain of size $L$. Conversely, any RPL theory can be converted to a finite set problem by multiplying all of the rational numbers by the least common multiple $L$ of the set of their denominators. If the finite set problem has a solution over domain of size $L$, then the original RPL theory has a standard model.

Models of consistent probable logic theories can be as large as desired, as shown by the following proposition.

**Proposition 6.** Suppose that $T$ is a probable logic theory that has a model over a sample space of size $k$ and suppose that $m$ is any positive integer. Then $T$ has a model over a sample space of size $k^m$.

**proof.** Suppose that $T$ has a model over sample space $S$ of size $k$. Then it is easy to construct a model over $S^m = S \times S \times \ldots \times S$ (the Cartesian product space with $m$ factors) having size $k^m$. The details showing that this is a model for $T$ provide some exercise for the reader.

One can formulate a different proof for **Proposition 5** using **Proposition 6**: Suppose that $L(T)$ is logically consistent. Then the probable logic theory which is similar to $T$ but has all 1's for the clauses has a standard model over a sample space of size 1. Using **Proposition 6**, this altered probable logic theory has a model over a sample space of size $k = k^*1$, where $k$ is the least common multiple of the denominators of the probabilities in the original clauses of $P$. This probable model for the altered theory is also a probable model for the original theory.
Other interesting open questions are the following.

**Open Question 2.** If $T$ is a consistent RPL theory and $A$ is an event symbol of $T$, is there a standard model $M$ such that $\text{pr}_M(A) = e(A)$?

**Open Question 3.** If there an effective procedure that computes $e(A)$, for consistent theories?

Special cases of the open questions would restrict the theories to be positive.

The basic problem which remains is the following: Given a probable logic theory $T$ and an event $A$ appearing in $T$, what probability should be assigned to $A$ based upon $T$?

Return to the positive theory $T_3$ for a moment:

1/2 ≤ A 
1/3 ≤ B 
1 ≤ C ← A 
1 ≤ C ← B 
1 ≤ D ← A, B

Were we to use probable resolution to deduce probabilities, we would get 1/2 for $A$, 1/3 for $B$, 1/2 and 1/3 for $C$ (needing to be combined somehow) and 0 for $D$. It’s an interesting elementary exercise to show that if one insists on a model having $\text{pr}(A) = 1/2$, $\text{pr}(B) = 1/3$ and $\text{pr}(D) = 0$, then it follows that we must have $\text{pr}(C) \geq 5/6$. (Hint: draw a Venn diagram!) So the question is: how does one combine multiple inferences, like the 1/2 and 1/3 probabilities for $C$. In the literature, various approaches have been tried, most using variations of Dempster-Shafer theory [1,7].

In the next section we propose another approach to inferring probabilities, which uses randomly generated models for RPL theories.

**4 Abstract interpretation of positive RPL theories using Monte Carlo methods**

The crux of the approach covered in this section is the generation of models for a probable logic theory $T$ using *Monte Carlo methods*. The idea is very simple: randomly generated bitsets can be used to model the events that occur in a probable logic theory. The methods explained here are tailored for positive probable logic theories. The generated models seek to jointly minimize all of the events so as to make each clause of the theory true. Answers to queries about events in $T$ report the probabilities of the events that are computed by the random model generator.

The approach here has some similarities to the known methods of *abstract interpretation* used to validate software code. Abstract interpretation generates input for code that will test the computed outputs for correctness. The methods discussed here do not test correctness of the program (the positive RPL theory), but rather they compute outputs for the program using statistical methods.
Recall that a positive probable logic theory \( T \) consists of a finite number of clauses having either the form

\[
r \leq A \tag{21}
\]

or the form

\[
r \leq A_1 \leftarrow A_2, \ldots, A_k \tag{22}
\]

where \( r \) is a rational number. It needs to be stressed that the second form is \textbf{not} intended to be interpreted as a conditional probability rule.

Suppose that we specify a sample size \( N \), where \( N \) is a positive integer. Events in \( T \) will be assigned to bitsets of size \( N \). A bitset of size \( N \) is essentially a function \( b : N \to \{0,1\} \). For example if \( N = 5 \) and \( N = \{1,2,3,4,5\} \) then we could write a bitset \( b = <0,1,0,1,1> \) to mean \( b(1) = 0 \) (false), \( b(2) = 1 \) (true), etc.

When we interpret the clause \( (21) \) over the sample space \( N \), we need to assign \( A \) to a bitset having at least \( r \times N \) elements (set true). (We mean the smallest nonnegative integer not less than \( r \times N \), of course.)

To be explicit, take \( N = 10 \) and suppose the clause is

\[
\frac{3}{8} \leq Q
\]

Now, \( 3/8 \times 10 = 4 \) so in order to interpret this clause so that it is \textbf{true} we need to assign \( Q \) to a bitset having at least 4 elements. For example, the bitset \( <1,0,0,1,1,0,0,1,0> \) assigned to \( Q \) would make the clause true, minimally. The assignment of \( Q \) to \( <1,1,1,1,0,0,1,1,0> \) would also make the clause true, but \textbf{not} minimally, because this bitset has more than 4 elements. The minimal bitset assignments (such as the explicit example given) can be easily generated using elementary algorithms employing \textit{random number generators}. Note however that \( N = 10 \) would not be a standard model size for this clause as discussed in section 3. We will not restrict our computations to standard models.

Continuing with our specific example, suppose now that we also wish to make it so that the clause

\[
\frac{3}{4} \leq P \leftarrow Q
\]

comes out to also be true in our interpretation using bitsets of size \( N=10 \). Assuming that our model generator has already assigned bitset \( <1,0,0,1,1,0,0,1,0> \) to \( Q \), then it has, in effect also assigned \( <0,1,1,0,0,1,1,1,0> \) to \( \neg Q \). So, in order to make the last display clause true, our generator will need to assign a bitset to \( P \) that makes it so that
because \( \frac{3}{4} \times 10 = 8 \). Thus our minimal model generator needs to randomly assign 2 true bits to \( P \) that are also in \( Q \), for example \( P = <1,0,0,0,0,0,0,0,1,0> \) would suffice.

If we had wished to generated a model of size \( N=10 \) minimally satisfying a clause like

\[
\frac{5}{8} \leq P <- Q, R
\]

then we would have first have randomly generated a minimal bitset for \( Q \), and a minimal bitset for \( R \) and then sought to randomly generates bits to set for \( P \) so that \( P \cup \neg Q \cup \neg R \geq 7 \). This would be accomplished by choosing random elements in \( Q \cap R \) to add to \( P \), until we get equality (= 7).

These specific calculations describe the flavor of the intended random model generator that we will describe further. A query regarding the minimal probability of \( P \) would then answer 2/10. In this simple example the answer would be deterministic. All computed models would have exactly the same minimal probability assigned to \( P \), as long as the sample space has size 10. Different sample space sizes would of course yield rational fractions for the probability of \( P \) that approximate 2/10 most nearly.

We will see that in general the minimal required probability for a query will itself be variable and will be determined by some distribution whose values are determined by the model assignment process.

**PRPL Model building algorithm**

We now describe minimal model assignment methods when there may be several clauses each having the same head event literal. Suppose that \( T \) is a positive rational probable logic (PRPL) theory and suppose that all of the clauses having head \( A \) are conveniently expressed in the following form

\[
\begin{align*}
r_1 & \leq A <- B_1 \\
r_2 & \leq A <- B_2 \\
\vdots \\
r_k & \leq A <- B_k
\end{align*}
\]

That is, there are \( k \) clauses having head event literal \( A \) and these clauses have bodies \( B_i \), respectively. The \( B_i \) can be absent in the case of atomic clauses and \( B_i \) could represent a conjunction of event literals in the cases of nonatomic clauses.

Suppose that a model of size \( N \) is to be generated and let \( n_i \) be the smallest nonnegative integer not less than \( N \times r_i \). Rewrite the program clauses for \( A \) as a collection of event set constraints:

\[
\begin{align*}
n_1 & \leq A_1 \\
n_2 & \leq A_2
\end{align*}
\]
\[ \ldots \]
\[ n_k \leq A_k \]

where, for each i,

\[ A_i = A \cup \neg B_i \quad \text{if the corresponding clause was nonatomic} \]
\[ = A \quad \text{if the corresponding clause was atomic} \]

To build a minimal model for A, we need to satisfy all of these inequalities minimally. To do this, proceed as follows:

- Build minimal bitsets for all of the individual literals in all of the nonvacuous bodies \( B_1, B_2, \ldots, B_k \).
- Until each of the k inequality constraints is satisfied minimally (=), generate a random bit that, if added to \( A_i \), enlarges some \( A_i \).
- If any event literal is reencountered, use its previously computed value.

To build a minimal model for a conjunctive query \( A_1, A_2, \ldots, A_s \) of event literals, use the basic build method above for each \( A_i \), being sure to reuse model event sets if any repeat literal is encountered.

Notice that, given a positive rational theory \( T \), the build method can be used to construct a partial model starting with a particular query literal. Clauses that are not needed to build a partial model for the query literal do not actually have their corresponding event sets generated, so the algorithm does not necessarily generate a model for all of \( T \). For example, if we use the algorithm to generate a partial model for \( A \) in the theory

\[ 1 \leq A \iff B \]
\[ 3/4 \leq B \]
\[ 1 \leq C \iff A \]

no events are actually need to be generated for C.

**Proposition 5.** The model building algorithm constructs a model for the set of clauses of positive rational theory \( T \) that depend upon a particular query literal (a subtheory). Moreover, there is a model for all of \( T \) which assigns the same events to the subtheory.

**proof.** The first claim is clear since the model building algorithm constructs event sets that do indeed satisfy all of the relevant clauses. We leave the other claim as an exercise for the reader. [Hint: What conjunctive query would accomplish this?]

The model building algorithm is easily implemented as a top-down query engine for PRPL theories (as described in the next section). In fact, the model building algorithm shares much of the same control logic as is used for SLD deduction for logic resolution theories. See [3]. This is not too surprising since our model theory for PRPL shares much in common with the standard Tarskian semantics for logic theories.
5 Experimental implementation of the PRPL model building algorithm

The reference [2] provides a prototype query engine for PRPL (we say “purple”) theories. The prototype is written in Java and can be used by anyone having the appropriate Java runtime engine. Future versions are anticipated, including Prolog implementations. In this section we describe the prototype PRPL interpreter and explain a few preliminary experiments.

Download RPL.jar to your desktop. This program computes models using the model building algorithm described in Section 4. Start RPL.jar by double-clicking on its icon (on your desktop). Figure 4 shows a screen snapshot of the RPL interpreter GUI …

![PRPL interpreter](image.png)

**Fig. 4 PRPL interpreter**

The following sample program was loaded into the interpreter -- the program’s file icon was dropped on the "/" label on the toolbar -- and appears in the program panel in Figure 4.

```prpl
% sample prpl program

1.0 <= a :- b, c.
0.5 <= b.
0.5 <= c.
```

Notice that (for now at least) we are using a Prolog-like PRPL clause syntax.

The three textfields on the toolbar are for the goal, the sample size, and the number of times to repeat the model generation. The user clicks on the button with the runner to generate models. In the report panel of Figure 4, are the results of three experiments. The first experiment was to generate one model (size = 100) and the result was a probability of 26/100 for a. The second experiment used sample size of 1000 and 100 trials, and the result was a median probability of 25/100, a minimum probability of 19/100 and a standard
deviation of 3.0/100. The last experiment used a sample size of 1000 and 100 repeats. Notice the larger deviation in the third experiment with the larger sample size.

It would be helpful to be able to use the model building algorithm as a basis for making predictions about likely outcomes for PRPL theories. For example, given two competing outcomes (goals), how should we decide which one is more likely. Of course, one can run experiments using the interpreter and look at the reports and use them as evidence that one outcome is more likely than another. But, what is really needed is some solid mathematical criteria for discrimination based on such an interpreter. What are the probability distributions that are (partially) computed by the model building algorithm? In particular, it would be good to know more generally how sensitive the outcomes are to sample size.

6 Further work

This paper explains work-in-progress (1998-2005). It is expected that some details will change over time. We continue to work on the open questions of the earlier sections, and more theory regarding the distributions computed by the model building algorithm, as well as extensions for RPL theories with variables.

References


[3] deleted web link


