Formal verification of key properties for several probability logics in the proof assistant Coq

Petar Maksimović

petar.maksimovic@gmail.com

Mathematical Institute of the Serbian Academy of Sciences and Arts
Faculty of Technical Sciences, University of Novi Sad
University of Nice Sophia Antipolis
INRIA Sophia Antipolis Méditerranée

Fourth Workshop on Formal and Automated Theorem Proving and Applications, February 4-5, 2011, Belgrade, Serbia

What is the main idea?

- To be able to represent and reason with uncertain knowledge
- To extend the classical propositional calculus with expressions which refer to probability, with the formulas still remaining either *true* or *false*.
- We introduce probabilistic operators, such as $P_{\geq s}\alpha$ with the intended meaning "the probability of α is at least *s*".
- Many such logics have been developed, the semantics of which is in the style of Kripke (possible worlds)
- Goal: To find a *strongly complete* (a set of formulas T is consistent *if* T is satisfiable) axiomatization for such logics

Why verify these logics?

- To make sure that the proofs of the main meta-theorems are correct, which is an important question. A formally verified proof of the strong completeness theorem, for instance, justifies the use of probabilistic SAT-checkers for problems such as:
  - determining whether probability estimates placed on certain events are consistent,
  - calculating, given probability estimates of certain assumptions, the probability of the conclusion,

  which could arise in various expert systems applying one of the developed probability logics to fields such as game theory, economy and medicine.

Why verify these logics?

- To make sure that the proofs of the main meta-theorems are correct, which is an important question. A formally verified proof of the strong completeness theorem, for instance, justifies the use of probabilistic SAT-checkers for problems such as:
  - determining whether probability estimates placed on certain events are consistent,
  - calculating, given probability estimates of certain assumptions, the probability of the conclusion,

  which could arise in various expert systems applying one of the developed probability logics to fields such as game theory, economy and medicine.
- Formalization of proof techniques. The proof technique which is used to prove strong completeness, for one, could be re-used, with some modifications, in situations when a similar technique is required.

The syntax of $LPP_2^{\mathbb{Q}}$

- Classical formulas (For_C) – defined as usual
    - ϕ – set of propositional letters

The syntax of $LPP_2^{\mathbb{Q}}$

- Classical formulas (For_C) – defined as usual
  - ϕ – set of propositional letters
- Probabilistic formulas (For_P)
  - Probabilistic operators: $P_{\geq s}$, $s \in \mathbb{Q}_{[0,1]}$
  - Connectives: $\neg_p$ and $\rightarrow_p$
  - Basic probabilistic formulas: $P_{\geq s}\alpha$
  - For_P - the smallest set:
    - which contains all of the basic probabilistic formulas, and
    - is closed under the following formation rules: if $A, B \in For_P$, then $\neg_p A, A \rightarrow_p B \in For_P$.

The syntax of $LPP_2^{\mathbb{Q}}$

- Classical formulas (For_C) – defined as usual
  - ϕ – set of propositional letters
- Probabilistic formulas (For_P)
  - Probabilistic operators: $P_{\geq s}, s \in \mathbb{Q}_{[0,1]}$
  - Connectives: $\neg_p$ and $\rightarrow_p$
  - Basic probabilistic formulas: $P_{\geq s}\alpha$
  - For_P - the smallest set:
    - which contains all of the basic probabilistic formulas, and
    - is closed under the following formation rules: if $A, B \in For_P$, then $\neg_p A, A \rightarrow_p B \in For_P$.
- All formulas: $For_{LPP_2^{\mathbb{Q}}} = For_C \cup For_P$

The syntax of $LPP_2^{\mathbb{Q}}$

- Classical formulas (For_C) – defined as usual
  - ϕ – set of propositional letters
- Probabilistic formulas (For_P)
  - Probabilistic operators: $P_{\geq s}, s \in \mathbb{Q}_{[0,1]}$
  - Connectives: $\neg_p$ and $\rightarrow_p$
  - Basic probabilistic formulas: $P_{\geq s}\alpha$
  - For_P - the smallest set:
    - which contains all of the basic probabilistic formulas, and
    - is closed under the following formation rules: if $A, B \in For_P$, then $\neg_p A, A \rightarrow_p B \in For_P$.
- All formulas: $For_{LPP_2^{\mathbb{Q}}} = For_C \cup For_P$
- Abbreviations:
  - The remaining classical and propositional connectives
  - Probabilistic operators: $P_{<s}$, $P_{\leq s}$, $P_{>s}$, $P_{=s}$, $P_{\neq s}$

The syntax of $LPP_2^{\mathbb{Q}}$

- Classical formulas (For_C) – defined as usual
  - ϕ – set of propositional letters
- Probabilistic formulas (For_P)
  - Probabilistic operators: $P_{\geq s}, s \in \mathbb{Q}_{[0,1]}$
  - Connectives: $\neg_p$ and $\rightarrow_p$
  - Basic probabilistic formulas: $P_{\geq s}\alpha$
  - For_P - the smallest set:
    - which contains all of the basic probabilistic formulas, and
    - is closed under the following formation rules: if $A, B \in For_P$, then $\neg_p A, A \rightarrow_p B \in For_P$.
- All formulas: $For_{LPP_2^{\mathbb{Q}}} = For_C \cup For_P$
- Abbreviations:
  - The remaining classical and propositional connectives
  - Probabilistic operators: $P_{<s}$, $P_{\leq s}$, $P_{>s}$, $P_{=s}$, $P_{\neq s}$
- Not permitted: $\alpha \wedge_c P_{\geq 1}\beta$, $P_{\geq 1}(P_{\geq 0}\alpha)$

The semantics of $LPP_2^{\mathbb{Q}}$

- Based on the possible–world approach

The semantics of $LPP_2^{\mathbb{Q}}$

- Based on the possible–world approach
- An $LPP_2^{\mathbb{Q}}$-model – a structure $M = \langle W, H, \mu, v \rangle$:
  - W is a non-empty set of objects we will call worlds
  - H is an algebra of subsets on W
  - μ – finitely additive measure $\mu : H \to \mathbb{Q}_{[0,1]}$, and
  - v is a valuation function $v : W \times \phi \to \{true, false\}$

The semantics of $LPP_2^{\mathbb{Q}}$

- Based on the possible–world approach
- An $LPP_2^{\mathbb{Q}}$-model – a structure $M = \langle W, H, \mu, v \rangle$:
  - W is a non-empty set of objects we will call worlds
  - H is an algebra of subsets on W
  - μ – finitely additive measure $\mu : H \rightarrow \mathbb{Q}_{[0,1]}$, and
  - v is a valuation function $v : W \times \phi \rightarrow \{true, false\}$
- $[\alpha]_M = \{w | v(w, \alpha) = true\}$, $\alpha \in For_C$

The semantics of $LPP_2^{\mathbb{Q}}$

- Based on the possible–world approach
- An $LPP_2^{\mathbb{Q}}$-model – a structure $M = \langle W, H, \mu, v \rangle$:
  - W is a non-empty set of objects we will call worlds
  - H is an algebra of subsets on W
  - μ – finitely additive measure $\mu : H \to \mathbb{Q}_{[0,1]}$, and
  - v is a valuation function $v : W \times \phi \to \{true, false\}$
- $[\alpha]_M = \{w | v(w, \alpha) = true\}$, $\alpha \in For_C$
- M is measurable if $[\alpha]_M \in H$, for all $\alpha \in For_C$
- We will onward focus on the class of all measurable models, which we will denote by $LPP_{2,Meas}^{\mathbb{Q}}$.

Satistfiability and validity in $LPP_2^{\mathbb{Q}}$

- The satisfiability relation $\models \subseteq LPP_{2,Meas}^{\mathbb{Q}} \times For_{LPP_2^{\mathbb{Q}}}$ satisfies the following conditions, for every measurable model $M = \langle W, H, \mu, v \rangle$ and every formula F:
  - if $F \in For_C$, $M \models F$ iff $v(w, F) = true$, for all $w \in W$,
  - if $F \equiv P_{\geq r}\alpha$, $M \models F$ iff $\mu([\alpha]_M) \geq r$.
  - if $F \equiv \neg_p A$, $M \models F$ iff $M \not\models A$.
  - if $F \equiv A \to_p B$, $M \models F$ iff $M \not\models A$ or $M \models B$.

Satistfiability and validity in $LPP_2^{\mathbb{Q}}$

- The satisfiability relation $\models \subseteq LPP^{\mathbb{Q}}_{2,Meas} \times For_{LPP_2^{\mathbb{Q}}}$ satisfies the following conditions, for every measurable model $M = \langle W, H, \mu, v \rangle$ and every formula F:
  - if $F \in For_C$, $M \models F$ iff $v(w, F) = true$, for all $w \in W$,
  - if $F \equiv P_{\geq r}\alpha$, $M \models F$ iff $\mu([\alpha]_M) \geq r$.
  - if $F \equiv \neg_p A$, $M \models F$ iff $M \not\models A$.
  - if $F \equiv A \rightarrow_p B$, $M \models F$ iff $M \not\models A$ or $M \models B$.
- A formula F is:

Satistfiability and validity in $LPP_2^{\mathbb{Q}}$

- The satisfiability relation $\models \subseteq LPP_{2,Meas}^{\mathbb{Q}} \times For_{LPP_2^{\mathbb{Q}}}$ satisfies the following conditions, for every measurable model $M = \langle W, H, \mu, v \rangle$ and every formula F:
  - if $F \in For_C$, $M \models F$ iff $v(w, F) = true$, for all $w \in W$,
  - if $F \equiv P_{\geq r}\alpha$, $M \models F$ iff $\mu([\alpha]_M) \geq r$.
  - if $F \equiv \neg_p A$, $M \models F$ iff $M \not\models A$.
  - if $F \equiv A \rightarrow_p B$, $M \models F$ iff $M \not\models A$ or $M \models B$.
- A formula F is:
  - *Satisfiable*, if there exists an LPP_2^n-measurable model M such that $M \models F$

Satistfiability and validity in $LPP_2^{\mathbb{Q}}$

- The satisfiability relation $\models \subseteq LPP_{2,Meas}^{\mathbb{Q}} \times For_{LPP_2^{\mathbb{Q}}}$ satisfies the following conditions, for every measurable model $M = \langle W, H, \mu, v \rangle$ and every formula F:
  - if $F \in For_C$, $M \models F$ iff $v(w, F) = true$, for all $w \in W$,
  - if $F \equiv P_{\geq r}\alpha$, $M \models F$ iff $\mu([\alpha]_M) \geq r$.
  - if $F \equiv \neg_p A$, $M \models F$ iff $M \not\models A$.
  - if $F \equiv A \rightarrow_p B$, $M \models F$ iff $M \not\models A$ or $M \models B$.
- A formula F is:
  - *Satisfiable*, if there exists an LPP_2^n-measurable model M such that $M \models F$
  - *Valid*, if $M \models F$, for all LPP_2^n-measurable models M

Satistfiability and validity in $LPP_2^{\mathbb{Q}}$

- The satisfiability relation $\models \subseteq LPP^{\mathbb{Q}}_{2,Meas} \times For_{LPP_2^{\mathbb{Q}}}$ satisfies the following conditions, for every measurable model $M = \langle W, H, \mu, v \rangle$ and every formula F:
  - if $F \in For_C$, $M \models F$ iff $v(w, F) = true$, for all $w \in W$,
  - if $F \equiv P_{\geq r}\alpha$, $M \models F$ iff $\mu([\alpha]_M) \geq r$.
  - if $F \equiv \neg_p A$, $M \models F$ iff $M \not\models A$.
  - if $F \equiv A \rightarrow_p B$, $M \models F$ iff $M \not\models A$ or $M \models B$.
- A formula F is:
  - *Satisfiable*, if there exists an LPP_2^n-measurable model M such that $M \models F$
  - *Valid*, if $M \models F$, for all LPP_2^n-measurable models M
- A set of formulas T is satisfiable if there exists an LPP_2^n-measurable model M such that $M \models F$, for all $F \in T$.

A complete axiomatization of $LPP_2^{\mathbb{Q}}$ – $Ax_{LPP_2^{\mathbb{Q}}}$

- Axioms:

AC1. $\alpha \rightarrow_c (\beta \rightarrow_c \alpha)$
AC2. $(\alpha \rightarrow_c (\beta \rightarrow_c \gamma)) \rightarrow_c ((\alpha \rightarrow_c \beta) \rightarrow_c (\alpha \rightarrow_c \gamma))$
AC3. $(\neg_c \beta \rightarrow_c \neg_c \alpha) \rightarrow ((\neg_c \beta \rightarrow_c \alpha) \rightarrow \beta)$

A complete axiomatization of $LPP_2^{\mathbb{Q}}$ – $Ax_{LPP_2^{\mathbb{Q}}}$

- Axioms:

AC1. $\alpha \to_c (\beta \to_c \alpha)$
AC2. $(\alpha \to_c (\beta \to_c \gamma)) \to_c ((\alpha \to_c \beta) \to_c (\alpha \to_c \gamma))$
AC3. $(\neg_c\beta \to_c \neg_c\alpha) \to ((\neg_c\beta \to_c \alpha) \to \beta)$

AP1. $A \to_p (B \to_p A)$
AP2. $(A \to_p (B \to_p C)) \to_p ((A \to_p B) \to_p (A \to_c C))$
AP3. $(\neg_p B \to_p \neg_p A) \to ((\neg_p B \to_p A) \to B)$

A complete axiomatization of $LPP_2^{\mathbb{Q}}$ – $Ax_{LPP_2^{\mathbb{Q}}}$

- Axioms:

AC1. $\alpha \to_c (\beta \to_c \alpha)$
AC2. $(\alpha \to_c (\beta \to_c \gamma)) \to_c ((\alpha \to_c \beta) \to_c (\alpha \to_c \gamma))$
AC3. $(\neg_c\beta \to_c \neg_c\alpha) \to ((\neg_c\beta \to_c \alpha) \to \beta)$

AP1. $A \to_p (B \to_p A)$
AP2. $(A \to_p (B \to_p C)) \to_p ((A \to_p B) \to_p (A \to_c C))$
AP3. $(\neg_p B \to_p \neg_p A) \to ((\neg_p B \to_p A) \to B)$

AP4. $P_{\geq 0}\alpha$
AP5. $P_{\leq r}\alpha \to_p P_{<s}\alpha$, for $s > r$
AP6. $P_{<r}\alpha \to_p P_{\leq r}\alpha$
AP7. $P_{\geq r}\alpha \to_p (P_{\geq s}\beta \to_p (P_{\geq 1}\neg_c(\alpha \wedge_c \beta) \to_p P_{\geq r+s}(\alpha \vee_c \beta)))$, $r + s \leq 1$
AP8. $P_{\leq r}\alpha \to_p (P_{<s}\beta \to_p P_{<r+s}(\alpha \vee_c \beta))$, $r + s \leq 1$
AP9. $P_{\geq 1}(\alpha \to_c \beta) \to_p (P_{\geq r}\alpha \to_p P_{\geq r}\beta)$

A complete axiomatization of $LPP_2^{\mathbb{Q}}$ – $Ax_{LPP_2^{\mathbb{Q}}}$

- Inference rules:
  - *Modus ponens for classical formulas*:
    from α and $\alpha \rightarrow_c \beta$, infer β
  - *Modus ponens for probabilistic formulas*:
    from A and $A \rightarrow_p B$, infer B

A complete axiomatization of $LPP_2^{\mathbb{Q}}$ – $Ax_{LPP_2^{\mathbb{Q}}}$

- Inference rules:
  - *Modus ponens for classical formulas*:
    from α and $\alpha \rightarrow_c \beta$, infer β
  - *Modus ponens for probabilistic formulas*:
    from A and $A \rightarrow_p B$, infer B
  - *Probabilistic necessitation*:
    from α, infer $P_{\geq 1}\alpha$

A complete axiomatization of $LPP_2^{\mathbb{Q}}$ – $Ax_{LPP_2^{\mathbb{Q}}}$

- Inference rules:
  - *Modus ponens for classical formulas*:
    from α and $\alpha \to_c \beta$, infer β
  - *Modus ponens for probabilistic formulas*:
    from A and $A \to_p B$, infer B
  - *Probabilistic necessitation*:
    from α, infer $P_{\geq 1}\alpha$
  - *The Serbian-Russian rule*:
    from $\{A \to_p P_{\neq s}\alpha\}_{s \in \mathbb{Q}_{[0,1]}}$, infer $A \to_p \bot_p$

A complete axiomatization of $LPP_2^{\mathbb{Q}}$ – $Ax_{LPP_2^{\mathbb{Q}}}$

- Inference rules:
  - *Modus ponens for classical formulas*:
    from α and $\alpha \rightarrow_c \beta$, infer β
  - *Modus ponens for probabilistic formulas*:
    from A and $A \rightarrow_p B$, infer B
  - *Probabilistic necessitation*:
    from α, infer $P_{\geq 1}\alpha$
  - *The Serbian-Russian rule*:
    from $\{A \rightarrow_p P_{\neq s}\alpha\}_{s \in \mathbb{Q}_{[0,1]}}$, infer $A \rightarrow_p \bot_p$
- Note that the last inference rule is infinitary – it has countably many premises. As a consequence, we will have infinite proofs.

Syntactic notions in $LPP_2^{\mathbb{Q}}$

- A formula Φ is **derivable** from a set of formulas (premises) T (denoted by $T \vdash \Phi$) if there exists a finite sequence of formulas $\Phi_0, \ldots, \Phi_k, \Phi$, such that each Φ_i is either in the set T, is an instance of one of the axiom schemata, or is obtained from the preceding formulas by using one of the inference rules. We call such a sequence a **proof** of Φ from T. A formula Φ is a **theorem** (denoted by $\vdash \Phi$) if it is derivable from the empty set of formulas.

Syntactic notions in $LPP_2^{\mathbb{Q}}$

- A formula Φ is **derivable** from a set of formulas (premises) T (denoted by $T \vdash \Phi$) if there exists a finite sequence of formulas $\Phi_0, \ldots, \Phi_k, \Phi$, such that each Φ_i is either in the set T, is an instance of one of the axiom schemata, or is obtained from the preceding formulas by using one of the inference rules. We call such a sequence a **proof** of Φ from T. A formula Φ is a **theorem** (denoted by $\vdash \Phi$) if it is derivable from the empty set of formulas.
- A set of formulas T is **consistent** if there exists at least one classical formula α and at least one probabilistic formula A which are not derivable from it, and otherwise is **inconsistent**. Alternatively, a set of formulas T is inconsistent if $T \vdash \bot_c$ or $T \vdash \bot_p$.

Syntactic notions in $LPP_2^{\mathbb{Q}}$

- A formula Φ is **derivable** from a set of formulas (premises) T (denoted by $T \vdash \Phi$) if there exists a finite sequence of formulas $\Phi_0, \dots, \Phi_k, \Phi$, such that each Φ_i is either in the set T, is an instance of one of the axiom schemata, or is obtained from the preceding formulas by using one of the inference rules. We call such a sequence a **proof** of Φ from T. A formula Φ is a **theorem** (denoted by $\vdash \Phi$) if it is derivable from the empty set of formulas.
- A set of formulas T is **consistent** if there exists at least one classical formula α and at least one probabilistic formula A which are not derivable from it, and otherwise is **inconsistent**. Alternatively, a set of formulas T is inconsistent if $T \vdash \bot_c$ or $T \vdash \bot_p$.
- A set of formulas T is **maximally consistent** if it is consistent and the following holds:
    - for each $\alpha \in For_C$: if $T \vdash \alpha$, then $\alpha \in T$ and $P_{\geq 1}\alpha \in T$,
    - for each $A \in For_P$: either $A \in T$ or $\neg_p A \in T$.

Meta-theoretic properties of $LPP_2^{\mathbb{Q}}$

- **The Deduction Theorem:** $T \vdash A \rightarrow_{c(p)} B$ iff $T, A \vdash B$

Meta-theoretic properties of $LPP_2^{\mathbb{Q}}$

- **The Deduction Theorem:** $T \vdash A \rightarrow_{c(p)} B$ iff $T, A \vdash B$
- **Soundness:** If a formula Φ is a theorem of $Ax_{LPP_2^{\mathbb{Q}}}$, then it is $LPP_{2,Meas}^{\mathbb{Q}}$–valid.

Meta-theoretic properties of $LPP_2^{\mathbb{Q}}$

- **The Deduction Theorem:** $T \vdash A \rightarrow_{c(p)} B$ iff $T, A \vdash B$
- **Soundness:** If a formula Φ is a theorem of $Ax_{LPP_2^{\mathbb{Q}}}$, then it is $LPP_{2,Meas}^{\mathbb{Q}}$–valid.
- **Strong Completeness:** A set of formulas T is $Ax_{LPP_2^{\mathbb{Q}}}$–consistent if and only if it is $LPP_{2,Meas}^{\mathbb{Q}}$–satisfiable.

Meta-theoretic properties of $LPP_2^{\mathbb{Q}}$

- **The Deduction Theorem:** $T \vdash A \rightarrow_{c(p)} B$ iff $T, A \vdash B$
- **Soundness:** If a formula Φ is a theorem of $Ax_{LPP_2^{\mathbb{Q}}}$, then it is $LPP_{2,Meas}^{\mathbb{Q}}$–valid.
- **Strong Completeness:** A set of formulas T is $Ax_{LPP_2^{\mathbb{Q}}}$–consistent if and only if it is $LPP_{2,Meas}^{\mathbb{Q}}$–satisfiable.
- **Simple Completeness:** If a formula Φ is $LPP_{2,Meas}^{\mathbb{Q}}$–valid, then it is a theorem of $Ax_{LPP_2^{\mathbb{Q}}}$.

Meta-theoretic properties of $LPP_2^{\mathbb{Q}}$

- **The Deduction Theorem:** $T \vdash A \rightarrow_{c(p)} B$ iff $T, A \vdash B$
- **Soundness:** If a formula Φ is a theorem of $Ax_{LPP_2^{\mathbb{Q}}}$, then it is $LPP^{\mathbb{Q}}_{2,Meas}$–valid.
- **Strong Completeness:** A set of formulas T is $Ax_{LPP_2^{\mathbb{Q}}}$–consistent if and only if it is $LPP^{\mathbb{Q}}_{2,Meas}$–satisfiable.
- **Simple Completeness:** If a formula Φ is $LPP^{\mathbb{Q}}_{2,Meas}$–valid, then it is a theorem of $Ax_{LPP_2^{\mathbb{Q}}}$.
- **Non-compactness:** Let T be a set of formulas. It does not hold that if every finite subset of T is $LPP^{\mathbb{Q}}_{2,Meas}$–satisfiable, then T is $LPP^{\mathbb{Q}}_{2,Meas}$–satisfiable.

The syntax of $LPP_2^{\mathbb{Q}}$

- Probabilistic formulas:

```
Inductive forP : Type :=
 | Pge : Q01 → forC → forP
 | NegP : forP → forP
 | ImpP : forP → forP → forP.
```

The syntax of $LPP_2^{\mathbb{Q}}$

- Probabilistic formulas:

```
Inductive forP : Type :=
 | Pge : Q01 → forC → forP
 | NegP : forP → forP
 | ImpP : forP → forP → forP.
```

- All formulas:

```
Inductive FOR : Type :=
 | Clas : forC → FOR
 | Prob : forP → FOR.
```

The syntax of $LPP_2^{\mathbb{Q}}$

- Probabilistic formulas:

```
Inductive forP : Type :=
 | Pge : Q01 → forC → forP
 | NegP : forP → forP
 | ImpP : forP → forP → forP.
```

- All formulas:

```
Inductive FOR : Type :=
 | Clas : forC → FOR
 | Prob : forP → FOR.
```

- Abbreviations:

```
Definition OrP (A : forP) (B : forP) : forP := ImpP (NegP A) B.
Definition Plt (s : Q01) (A : forC) : forP := NegP (Pge s A).
```

The semantics of $LPP_2^{\mathbb{Q}}$

- A Model Candidate:

```
Record Model_Cand : Type := mkMCand {
       MC_Worlds : Ensemble ElemWS;
       MC_Algebra : Ensemble (Ensemble ElemWS);
       MC_Measure : Measure;
       MC_Valuation : ElemWS -> nat -> Prop;
       MC_ElemWS_Cd : inhabited ElemWS;
       MC_Worlds_Cd : MC_Worlds = Full_set ElemWS}.
```

The semantics of $LPP_2^{\mathbb{Q}}$

- A Model Candidate:

```
Record Model_Cand : Type := mkMCand {
       MC_Worlds : Ensemble ElemWS;
       MC_Algebra : Ensemble (Ensemble ElemWS);
       MC_Measure : Measure;
       MC_Valuation : ElemWS -> nat -> Prop;
       MC_ElemWS_Cd : inhabited ElemWS;
       MC_Worlds_Cd : MC_Worlds = Full_set ElemWS}.
```

- Satisfiability:

```
Definition Satisfiable (T : Ensemble FOR) : Prop :=
∃ Model : Model_Meas, modelsSet Model T.
```

The semantics of $LPP_2^{\mathbb{Q}}$

- A Model Candidate:

```
Record Model_Cand : Type := mkMCand {
        MC_Worlds : Ensemble ElemWS;
        MC_Algebra : Ensemble (Ensemble ElemWS);
        MC_Measure : Measure;
        MC_Valuation : ElemWS -> nat -> Prop;
        MC_ElemWS_Cd : inhabited ElemWS;
        MC_Worlds_Cd : MC_Worlds = Full_set ElemWS}.
```

- Satisfiability:

```
Definition Satisfiable (T : Ensemble FOR) : Prop :=
∃ Model : Model_Meas, modelsSet Model T.
```

- Validity:

```
Definition Valid (F : FOR) : Prop :=
∀ Model : Model_Meas, models Model F.
```

A complete axiomatization of $LPP_2^{\mathbb{Q}}$ – $Ax_{LPP_2^{\mathbb{Q}}}$

- Encodings of axioms:

AP1. $\alpha \to_p (\beta \to_p \alpha)$

```
Definition AxAP01 (A B : forP) : FOR := Prob (ImpP A (ImpP B A)).
```

A complete axiomatization of $LPP_2^{\mathbb{Q}}$ – $Ax_{LPP_2^{\mathbb{Q}}}$

- Encodings of axioms:

AP1. $\alpha \rightarrow_p (\beta \rightarrow_p \alpha)$

```
Definition AxAP01 (A B : forP) : FOR := Prob (ImpP A (ImpP B A)).
```

AP8. $P_{\leq r}\alpha \rightarrow_p (P_{<s}\beta \rightarrow_p P_{<r+s}(\alpha \vee_c \beta))$, $r + s \leq 1$

```
Definition AxAP08 (A B : forC) (r s : Q01) (H : r + s ≤ 1) : FOR :=
Prob (ImpP (Ple r A) (ImpP (Plt s B) (Plt (r + s) (OrC A B)))).
```

A complete axiomatization of $LPP_2^{\mathbb{Q}}$ – $Ax_{LPP_2^{\mathbb{Q}}}$

- Encodings of axioms:

AP1. $\alpha \rightarrow_p (\beta \rightarrow_p \alpha)$

```
Definition AxAP01 (A B : forP) : FOR := Prob (ImpP A (ImpP B A)).
```

AP8. $P_{\leq r}\alpha \rightarrow_p (P_{<s}\beta \rightarrow_p P_{<r+s}(\alpha \vee_c \beta))$, $r + s \leq 1$

```
Definition AxAP08 (A B : forC) (r s : Q01) (H : r + s ≤ 1) : FOR :=
Prob (ImpP (Ple r A) (ImpP (Plt s B) (Plt (r + s) (OrC A B)))).
```

- Encodings of inference rules:

```
dbyIRMPc : ∀ (T : Ensemble FOR) (A B : forC),
derives T (Clas A) → derives T (Clas (ImpC A B)) →
→ derives T (Clas B)
```

A complete axiomatization of $LPP_2^{\mathbb{Q}}$ – $Ax_{LPP_2^{\mathbb{Q}}}$

- Encodings of axioms:

AP1. $\alpha \rightarrow_p (\beta \rightarrow_p \alpha)$

```
Definition AxAP01 (A B : forP) : FOR := Prob (ImpP A (ImpP B A)).
```

AP8. $P_{\leq r}\alpha \rightarrow_p (P_{<s}\beta \rightarrow_p P_{<r+s}(\alpha \vee_c \beta))$, $r + s \leq 1$

```
Definition AxAP08 (A B : forC) (r s : Q01) (H : r + s ≤ 1) : FOR :=
Prob (ImpP (Ple r A) (ImpP (Plt s B) (Plt (r + s) (OrC A B)))).
```

- Encodings of inference rules:

```
dbyIRMPc : ∀ (T : Ensemble FOR) (A B : forC),
derives T (Clas A) → derives T (Clas (ImpC A B)) →
→ derives T (Clas B)
```

```
dbyIRPN : ∀ (T : Ensemble FOR) (A : forC),
derives T (Clas A) → derives T (Prob (Pge 1 A))
```

Syntactic notions in $LPP_2^{\mathbb{Q}}$

- Consistency:

```
Definition Consistent (T : Ensemble FOR) : Prop :=
(∃ f : forC, ¬ Derivable n T (Clas f)) ∧
(∃ f : forP, ¬ Derivable n T (Prob f)).
```

Syntactic notions in $LPP_2^{\mathbb{Q}}$

- Consistency:

```
Definition Consistent (T : Ensemble FOR) : Prop :=
(∃ f : forC, ¬ Derivable n T (Clas f)) ∧
(∃ f : forP, ¬ Derivable n T (Prob f)).
```

- Alternate Inconsistency:

```
Definition Inconsistent_Alternate (T : Ensemble FOR) : Prop :=
(∀ f : forC, Derivable n T (Clas f)) ∨
(∀ f : forP, Derivable n T (Prob f)).
```

Syntactic notions in $LPP_2^{\mathbb{Q}}$

- Consistency:

```
Definition Consistent (T : Ensemble FOR) : Prop :=
(∃ f : forC, ¬ Derivable n T (Clas f)) ∧
(∃ f : forP, ¬ Derivable n T (Prob f)).
```

- Alternate Inconsistency:

```
Definition Inconsistent_Alternate (T : Ensemble FOR) : Prop :=
(∀ f : forC, Derivable n T (Clas f)) ∨
(∀ f : forP, Derivable n T (Prob f)).
```

- Maximal Consistency:

```
Definition Max_Consistent (T : Ensemble FOR) : Prop :=
Consistent T ∧
(∀ A : forC, Derivable n T (Clas A) → In FOR T (Clas A) ∧ In FOR
T (Prob (Pge 1 A))) ∧
(∀ A : forP, In FOR T (Prob A) ⊻ In FOR T (Prob (NegP A))).
```

Main meta-theoretic results

- The Deduction Theorem:

```
Theorem LPP2_Q_Deduction_Theorem_Classical :
∀ (T : Ensemble FOR) (A B : forC),
Derivable n (Union FOR T (Singleton FOR (Clas A))) (Clas B)
↔ Derivable n T (Clas (ImpC A B)).
```

```
Theorem LPP2_Q_Deduction_Theorem_Probabilistic :
∀ (T : Ensemble FOR) (A B : forP),
Derivable n (Union FOR T (Singleton FOR (Prob A))) (Prob B)
↔ Derivable n T (Prob (ImpP A B)).
```

Main meta-theoretic results

- The Deduction Theorem:

```
Theorem LPP2_Q_Deduction_Theorem_Classical :
∀ (T : Ensemble FOR) (A B : forC),
Derivable n (Union FOR T (Singleton FOR (Clas A))) (Clas B)
↔ Derivable n T (Clas (ImpC A B)).
```

```
Theorem LPP2_Q_Deduction_Theorem_Probabilistic :
∀ (T : Ensemble FOR) (A B : forP),
Derivable n (Union FOR T (Singleton FOR (Prob A))) (Prob B)
↔ Derivable n T (Prob (ImpP A B)).
```

- Soundness:

```
Theorem LPP2_Q_Soundness : ∀ (F : FOR), isTheorem F → Valid F.
```

Main meta-theoretic results

- Strong Completeness:

```
Theorem LPP2_Q_Strong_Completeness :
∀ T : Ensemble FOR, Consistent T ↔ Satisfiable T.
```

Main meta-theoretic results

- Strong Completeness:

```
Theorem LPP2_Q_Strong_Completeness :
∀ T : Ensemble FOR, Consistent T ↔ Satisfiable T.
```

- Simple Completeness:

```
Theorem LPP2_Q_Simple_Completeness :
∀ F : FOR, Valid F → isTheorem F.
```

Main meta-theoretic results

- Strong Completeness:

```
Theorem LPP2_Q_Strong_Completeness :
∀ T : Ensemble FOR, Consistent T ↔ Satisfiable T.
```

- Simple Completeness:

```
Theorem LPP2_Q_Simple_Completeness :
∀ F : FOR, Valid F → isTheorem F.
```

- Non-compactness:

```
Theorem LPP2_Q_NonCompactness : ∃ T : Ensemble FOR,
(∀ T' : Ensemble FOR, Finite FOR T' → Included FOR T' T →
→ Satisfiable T') ∧ ¬ Satisfiable T.
```

Other formalized probability logics

- $LPP_1^{\mathbb{Q}}$, a probability logic similar to $LPP_2^{\mathbb{Q}}$, but with iterations of probabilistic operators allowed.

Other formalized probability logics

- $LPP_1^{\mathbb{Q}}$, a probability logic similar to $LPP_2^{\mathbb{Q}}$, but with iterations of probabilistic operators allowed.
- $LPP_2^{Fr(n)}$, a probability logic similar to $LPP_2^{\mathbb{Q}}$, with the main difference being that the measure μ can take only values from the (finite) set $Fr(n) = \{0, \frac{1}{n}, \dots, \frac{n-1}{n}, 1\}$. This logic has no infinitary rules, and has the compactness property.

Other formalized probability logics

- $LPP_1^{\mathbb{Q}}$, a probability logic similar to $LPP_2^{\mathbb{Q}}$, but with iterations of probabilistic operators allowed.
- $LPP_2^{Fr(n)}$, a probability logic similar to $LPP_2^{\mathbb{Q}}$, with the main difference being that the measure μ can take only values from the (finite) set $Fr(n) = \{0, \frac{1}{n}, \ldots, \frac{n-1}{n}, 1\}$. This logic has no infinitary rules, and has the compactness property.
- $LPP_1^{Fr(n)}$, a probability logic similar to $LPP_2^{Fr(n)}$, but with iterations of probabilistic operators allowed.

Future work

- Formalizing the proof of the decidability for some/all of the already formalized probability logics

Future work

- Formalizing the proof of the decidability for some/all of the already formalized probability logics
- Formalizing other probability logics
  - With the qualitative probability operator
  - With a real–valued measure
  - With an intuitionistic base
  - With infinitesimals

Future work

- Formalizing the proof of the decidability for some/all of the already formalized probability logics
- Formalizing other probability logics
  - With the qualitative probability operator
  - With a real–valued measure
  - With an intuitionistic base
  - With infinitesimals
- Formalizing and extracting a certified probabilistic SAT–checker.

The Usual Way of Ending a Presentation

Thank you for your attention.
Any questions?