Probability Logics in Coq

Petar Maksimović

petarmax@mi.sanu.ac.rs

Mathematical Institute of the Serbian Academy of Sciences and Arts
INRIA Sophia Antipolis Méditerranée

TYPES 2013.
April 26th 2013, Toulouse, France

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 at the Mathematical Institute in Belgrade over the past 15+ years.
- Semantics 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 REALLY correct. 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 decision-support 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 REALLY correct. 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 decision-support 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 syntax of $LPP_2^{\mathbb{Q}}$

- Rational numbers in the unit interval:

```
Record Q01 : Type := mkQ01 { origQ :> Qc;
                             Q01_bcond : 0 <= origQ;
                             Q01_tcond : origQ <= 1}.
```

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

- Rational numbers in the unit interval:

```
Record Q01 : Type := mkQ01 { origQ :> Qc;
                             Q01_bcond : 0 <= origQ;
                             Q01_tcond : origQ <= 1}.
```

- Probabilistic formulas:

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

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

- Rational numbers in the unit interval:

```
Record Q01 : Type := mkQ01 { origQ :> Qc;
                             Q01_bcond : 0 <= origQ;
                             Q01_tcond : origQ <= 1}.
```

- 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 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 $\mu : H \to \mathbb{Q}_{[0,1]}$, 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 $\mu : H \to \mathbb{Q}_{[0,1]}$, and
  - v is a valuation function $v : W \times \phi \to \{true, false\}$
- $[\alpha]_M = \{w | v(w, \alpha) = true\}$, $\alpha \in For_C$

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 $LPP_{2,Meas}^{\mathbb{Q}}$.

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

- Worlds: elements of type U which are in a given set W

```
Record ElemW : Type := mkElemW { origU :> U;
                                 inW : In U W origU }.
```

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

- Worlds: elements of type U which are in a given set W

```
Record ElemW : Type := mkElemW { origU :> U;
                                 inW : In U W origU }.
```

- Algebra of Subsets:

```
Inductive AOS (H : Ensemble (Ensemble U)) : Prop :=
AOS_intro : AOS_Empty_set_In H ->
            AOS_Closed_under_Complement H ->
            AOS_Closed_under_Union H -> AOS H.
```

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

- Worlds: elements of type U which are in a given set W

```
Record ElemW : Type := mkElemW { origU :> U;
                                 inW : In U W origU }.
```

- Algebra of Subsets:

```
Inductive AOS (H : Ensemble (Ensemble U)) : Prop :=
AOS_intro : AOS_Empty_set_In H ->
            AOS_Closed_under_Complement H ->
            AOS_Closed_under_Union H -> AOS H.
```

- Measure:

```
Inductive FAM_Qc (M : EE U -> Qc) (H : EE U) : Prop :=
FAM_Qc_Intro : AOS _ H -> FAM_Qc_Nonnegative M ->
               FAM_Qc_Empty_set M -> FAM_Qc_Full_set M ->
               FAM_Qc_Additive M H -> FAM_Qc M H.
```

Measurable Models in Coq

- A Measurable Model:

```
Record Model_Cand : Type := mkMCand {
       MC_Worlds : Ensemble ElemWS;
       MC_Algebra : EE ElemWS);
       MC_Measure : EE ElemWS -> Qc;
       MC_Valuation : ElemWS -> nat -> Prop;
       MC_InhElemWS : inhabited ElemWS;
       MC_FullWorlds : MC_Worlds = Full_set ElemWS;
       MC_IsModel : FAM_Qc _ MC_Measure MC_Algebra;
       MC_Meas : ∀ fC : forC, In _ MC_Algebra (worlds fC)}.
```

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 $M \models B$.

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 $M \models B$.

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 $M \models B$.
- Coq's built-in implication, along with the `Classical` library.

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 $M \models B$.
- Coq's built-in implication, along with the `Classical` library.
- A formula F is:

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 $M \models B$.
- Coq's built-in implication, along with the `Classical` library.
- A formula F is:
  - *Satisfiable*, if there exists an $LPP_2^{\mathbb{Q}}$-measurable model M such that $M \models F$

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 $M \models B$.
- Coq's built-in implication, along with the `Classical` library.
- A formula F is:
  - *Satisfiable*, if there exists an $LPP_2^{\mathbb{Q}}$-measurable model M such that $M \models F$
  - *Valid*, if $M \models F$, for all $LPP_2^{\mathbb{Q}}$-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 $M \models B$.
- Coq's built-in implication, along with the `Classical` library.
- A formula F is:
  - *Satisfiable*, if there exists an $LPP_2^{\mathbb{Q}}$-measurable model M such that $M \models F$
  - *Valid*, if $M \models F$, for all $LPP_2^{\mathbb{Q}}$-measurable models M
- A set of formulas T is satisfiable if there exists an $LPP_2^{\mathbb{Q}}$-measurable model M such that $M \models F$, for all $F \in T$.

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

- Axiom schemata:

ACT. All instances of classical propositional tautologies for classical formulas.
APT. All instances of classical propositional tautologies for probabilistic formulas.

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

- Axiom schemata:

ACT. All instances of classical propositional tautologies for classical formulas.
APT. All instances of classical propositional tautologies for probabilistic formulas.

AP1. $P_{\geq 0}\alpha$
AP2. $P_{\leq r}\alpha \rightarrow_p P_{<s}\alpha$, for $s > r$
AP3. $P_{<r}\alpha \rightarrow_p P_{\leq r}\alpha$
AP4. $P_{\geq r}\alpha \rightarrow_p (P_{\geq s}\beta \rightarrow_p (P_{\geq 1}\neg_c(\alpha \wedge_c \beta) \rightarrow_p P_{\geq r+s}(\alpha \vee_c \beta)))$, $r + s \leq 1$
AP5. $P_{\leq r}\alpha \rightarrow_p (P_{<s}\beta \rightarrow_p P_{<r+s}(\alpha \vee_c \beta))$, $r + s \leq 1$
AP6. $P_{\geq 1}(\alpha \rightarrow_c \beta) \rightarrow_p (P_{\geq r}\alpha \rightarrow_p P_{\geq r}\beta)$

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

- Axiom schemata:

ACT. All instances of classical propositional tautologies for classical formulas.
APT. All instances of classical propositional tautologies for probabilistic formulas.

AP1. $P_{\geq 0}\alpha$
AP2. $P_{\leq r}\alpha \rightarrow_p P_{<s}\alpha$, for $s > r$
AP3. $P_{<r}\alpha \rightarrow_p P_{\leq r}\alpha$
AP4. $P_{\geq r}\alpha \rightarrow_p (P_{\geq s}\beta \rightarrow_p (P_{\geq 1}\neg_c(\alpha \wedge_c \beta) \rightarrow_p P_{\geq r+s}(\alpha \vee_c \beta)))$, $r + s \leq 1$
AP5. $P_{\leq r}\alpha \rightarrow_p (P_{<s}\beta \rightarrow_p P_{<r+s}(\alpha \vee_c \beta))$, $r + s \leq 1$
AP6. $P_{\geq 1}(\alpha \rightarrow_c \beta) \rightarrow_p (P_{\geq r}\alpha \rightarrow_p P_{\geq r}\beta)$

- Sample encoding of an axiom schema:

```
Definition AxAP05 (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 $LPP_2^{\mathbb{Q}}$ – $Ax_{LPP_2^{\mathbb{Q}}}$

- 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 $\alpha \rightarrow_c \beta$, infer β.
  - *Modus ponens for probabilistic formulas*:
    from A and $A \rightarrow_p B$, infer B.
  - *Probabilistic Necessitation*:
    from α, infer $P_{\geq 1}\alpha$.
  - *Domain Enforcement*:
    from $\{A \rightarrow_p P_{\neq s}\alpha\}_{s \in \mathbb{Q}_{[0,1]}}$, infer $A \rightarrow_p \bot_p$.

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 $P_{\geq 1}\alpha$.
  - *Domain Enforcement*:
    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.

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 $P_{\geq 1}\alpha$.
  - *Domain Enforcement*:
    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.
- Encoding of the DE inference rule:

```
∀ (T : Ensemble FOR) (F : forP) (A : forC) (proofs : Q01 -> proof_term),
  (∀ r : Q01, derives T (proofs r) (Prob (ImpP F (Pne r A)))) ->
  derives T (dIRDE F A proofs) (Prob (ImpP F (FalP (Pge Q01_0 A)))).
```

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

- A formula Φ is **derivable** from a set of formulas (context) T (denoted by $T \vdash \Phi$) if there exists a tree of formulas such that Φ is its root, each of the leaves is either in the set T or is an instance of one of the axiom schemata, while the remaining nodes are obtained through the use of inference rules, and are connected accordingly. 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 (context) T (denoted by $T \vdash \Phi$) if there exists a tree of formulas such that Φ is its root, each of the leaves is either in the set T or is an instance of one of the axiom schemata, while the remaining nodes are obtained through the use of inference rules, and are connected accordingly. 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 **inconsistent** if $T \vdash \bot_c$ or $T \vdash \bot_p$, and is consistent otherwise.

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

- A formula Φ is **derivable** from a set of formulas (context) T (denoted by $T \vdash \Phi$) if there exists a tree of formulas such that Φ is its root, each of the leaves is either in the set T or is an instance of one of the axiom schemata, while the remaining nodes are obtained through the use of inference rules, and are connected accordingly. 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 **inconsistent** if $T \vdash \bot_c$ or $T \vdash \bot_p$, and is consistent otherwise.
- 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 $LPP_2^{\mathbb{Q}}$

- **Soundness:** If a formula Φ is a theorem of $Ax_{LPP_2^{\mathbb{Q}}}$, then it is $LPP^{\mathbb{Q}}_{2,Meas}$–valid.

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

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

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

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

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

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

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

Proving Soundness and the Deduction Theorem

- Soundness is proven in the standard way, by proving that each instance of the axiom schemata is valid, and that the inference rules preserve validity.

Proving Soundness and the Deduction Theorem

- Soundness is proven in the standard way, by proving that each instance of the axiom schemata is valid, and that the inference rules preserve validity.

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

Proving Soundness and the Deduction Theorem

- Soundness is proven in the standard way, by proving that each instance of the axiom schemata is valid, and that the inference rules preserve validity.

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

- The deduction theorem is proven twice, once for classical and once for probabilistic formulas. One interesting required fact is that proofs of derivability for classical formulas do not depend on the probabilistic formulas

Proving Soundness and the Deduction Theorem

- Soundness is proven in the standard way, by proving that each instance of the axiom schemata is valid, and that the inference rules preserve validity.

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

- The deduction theorem is proven twice, once for classical and once for probabilistic formulas. One interesting required fact is that proofs of derivability for classical formulas do not depend on the probabilistic formulas

```
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)).
```

On the Road to Strong Completeness

- Consistent sets with embedded proofs of consistency:

```
Record ConsT := mkCons { origT :> Ensemble FOR;
                         is_Cons : Consistent origT }.
```

On the Road to Strong Completeness

- Consistent sets with embedded proofs of consistency:

```
Record ConsT := mkCons { origT :> Ensemble FOR;
                         is_Cons : Consistent origT }.
```

- Extending a consistent set of formulas T to a maximally consistent set of formulas T^*.

On the Road to Strong Completeness

- Consistent sets with embedded proofs of consistency:

```
Record ConsT := mkCons { origT :> Ensemble FOR;
                         is_Cons : Consistent origT }.
```

- Extending a consistent set of formulas T to a maximally consistent set of formulas T^*.
- Start with $T_0 = T \cup CC(T) \cup (P_{\geq 1}(CC(T))$.

On the Road to Strong Completeness

- Consistent sets with embedded proofs of consistency:

```
Record ConsT := mkCons { origT :> Ensemble FOR;
                         is_Cons : Consistent origT }.
```

- Extending a consistent set of formulas T to a maximally consistent set of formulas T^*.
- Start with $T_0 = T \cup CC(T) \cup (P_{\geq 1}(CC(T))$.
- Enumerate all classical and probabilistic formulas (axiom).

On the Road to Strong Completeness

- Consistent sets with embedded proofs of consistency:

```
Record ConsT := mkCons { origT :> Ensemble FOR;
                         is_Cons : Consistent origT }.
```

- Extending a consistent set of formulas T to a maximally consistent set of formulas T^*.
- Start with $T_0 = T \cup CC(T) \cup (P_{\geq 1}(CC(T))$.
- Enumerate all classical and probabilistic formulas (axiom).
- Construct (this is a "classical construction")

$$T_{n+1} = \begin{cases} T_n \cup \{P_{=p(\alpha_n)}\alpha_n\} \cup \{A_n\} & \text{if } T_n \cup \{A_n\} \text{ is consistent,} \\ T_n \cup \{P_{=p(\alpha_n)}\alpha_n\} \cup \{\neg_p A_n\} & \text{otherwise.} \end{cases}$$

On the Road to Strong Completeness

- Consistent sets with embedded proofs of consistency:

```
Record ConsT := mkCons { origT :> Ensemble FOR;
                         is_Cons : Consistent origT }.
```

- Extending a consistent set of formulas T to a maximally consistent set of formulas T^*.
- Start with $T_0 = T \cup CC(T) \cup (P_{\geq 1}(CC(T))$.
- Enumerate all classical and probabilistic formulas (axiom).
- Construct (this is a "classical construction")

$$T_{n+1} = \begin{cases} T_n \cup \{P_{=p(\alpha_n)}\alpha_n\} \cup \{A_n\} & \text{if } T_n \cup \{A_n\} \text{ is consistent,} \\ T_n \cup \{P_{=p(\alpha_n)}\alpha_n\} \cup \{\neg_p A_n\} & \text{otherwise.} \end{cases}$$

- Maximally consistent extension: $T^* = \bigcup_{i=1}^{\infty} T_i$.

```
Definition MC_Extension (T : Ensemble FOR) : Ensemble FOR :=
    (fun A : FOR ⇒ ∃ k : nat, In FOR (Tn_const T k) A).
```

The Canonical Model

We have a consistent set of formulas T and its extension to a maximally consistent set T^*. We construct a canonical measurable model for the set T as follows:

The Canonical Model

We have a consistent set of formulas T and its extension to a maximally consistent set T^*. We construct a canonical measurable model for the set T as follows:

- Worlds:

$$W = \{w \mid w : \phi \to \{true, false\}, \{w_{ext}(\alpha) = 1 \mid \alpha \in For_C, T \vdash \alpha\}\}.$$

```
Worlds = Full_set (ElemW (nat -> Prop) (fun v : nat -> Prop =>
   satSetC v (ClasDCons T)))
```

The Canonical Model

We have a consistent set of formulas T and its extension to a maximally consistent set T^*. We construct a canonical measurable model for the set T as follows:

- Worlds:

$$W = \{w \mid w : \phi \to \{true, false\}, \{w_{ext}(\alpha) = 1 \mid \alpha \in For_C, T \vdash \alpha\}\}.$$

```
Worlds = Full_set (ElemW (nat -> Prop) (fun v : nat -> Prop =>
   satSetC v (ClasDCons T)))
```

- Algebra of subsets: $H = \{[\alpha] \mid \alpha \in For_C\}$,

The Canonical Model

We have a consistent set of formulas T and its extension to a maximally consistent set T^*. We construct a canonical measurable model for the set T as follows:

- Worlds:

$$W = \{w \mid w : \phi \to \{true, false\}, \{w_{ext}(\alpha) = 1 \mid \alpha \in For_C, T \vdash \alpha\}\}.$$

```
Worlds = Full_set (ElemW (nat -> Prop) (fun v : nat -> Prop =>
   satSetC v (ClasDCons T)))
```

- Algebra of subsets: $H = \{[\alpha] \mid \alpha \in For_C\}$,
- The measure: $\mu([\alpha]) = q(\alpha)$, where $T^* \vdash P_{=q(\alpha)}\alpha$.

The Canonical Model

We have a consistent set of formulas T and its extension to a maximally consistent set T^*. We construct a canonical measurable model for the set T as follows:

- Worlds:

$$W = \{w \mid w : \phi \to \{true, false\}, \{w_{ext}(\alpha) = 1 \mid \alpha \in For_C, T \vdash \alpha\}\}.$$

```
Worlds = Full_set (ElemW (nat -> Prop) (fun v : nat -> Prop =>
   satSetC v (ClasDCons T)))
```

- Algebra of subsets: $H = \{[\alpha] \mid \alpha \in For_C\}$,
- The measure: $\mu([\alpha]) = q(\alpha)$, where $T^* \vdash P_{=q(\alpha)}\alpha$.
- The valuation:

$$v(w, p_i) = w(p_i)$$

The Canonical Model

We have a consistent set of formulas T and its extension to a maximally consistent set T^*. We construct a canonical measurable model for the set T as follows:

- Worlds:

$$W = \{w \mid w : \phi \to \{true, false\}, \{w_{ext}(\alpha) = 1 \mid \alpha \in For_C, T \vdash \alpha\}\}.$$

```
Worlds = Full_set (ElemW (nat -> Prop) (fun v : nat -> Prop =>
   satSetC v (ClasDCons T)))
```

- Algebra of subsets: $H = \{[\alpha] \mid \alpha \in For_C\}$,
- The measure: $\mu([\alpha]) = q(\alpha)$, where $T^* \vdash P_{=q(\alpha)}\alpha$.
- The valuation:

$$v(w, p_i) = w(p_i)$$

We show that $M^* = \langle W, H, \mu, v\rangle$ is a measurable model.

Strong Completeness

- A set of formulas T is $LPP_2^{\mathbb{Q}}$-consistent *iff* it is $LPP_{2,Meas}^{\mathbb{Q}}$-satisfiable.

Strong Completeness

- A set of formulas T is $LPP_2^{\mathbb{Q}}$-consistent *iff* it is $LPP_{2,Meas}^{\mathbb{Q}}$-satisfiable.

The two directions are proven separately. Left-to-right, we show that the model in question is the canonical model M^*, while the right-to-left follows from the soundness of the system.

Strong Completeness

- A set of formulas T is $LPP_2^{\mathbb{Q}}$-consistent *iff* it is $LPP_{2,Meas}^{\mathbb{Q}}$-satisfiable.

The two directions are proven separately. Left-to-right, we show that the model in question is the canonical model M^*, while the right-to-left follows from the soundness of the system.

```
Theorem LPP2_Q_Strong_Completeness : ∀ T : Ensemble FOR,
    Consistent T <-> Satisfiable T.
```

Proving Simple Completeness

This result is obtained easily from Strong Completeness, given the lemma that if a set of classical formulas T is consistent in the sense of classical logic, then it is also $LPP_2^{\mathbb{Q}}$-consistent.

If a formula F is $LPP_{2,Meas}^{\mathbb{Q}}$-valid, then it is a theorem of $LPP_2^{\mathbb{Q}}$.

```
Theorem LPP2_Q_Simple_Completeness :
        ∀ F : FOR, Valid F -> isTheorem F.
```

Proving Non-compactness by Counter-example

Take the set $T = \{P_{<\frac{1}{n}}\alpha |\ n \in \mathbb{N}^+\}$. Then:

- Using the DE inference rule, we obtain that T is inconsistent.
- By the strong completeness theorem, T is not satisfiable.
- Every finite subset of T is consistent, therefore satisfiable.

This gives us the non-compactness theorem for $LPP_2^{\mathbb{Q}}$:

There exists a set of $LPP_2^{\mathbb{Q}}$-formulas which is unsatisfiable, but whose every finite subset is satisfiable.

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

Other formally verified probability logics

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

Other formally verified probability logics

- $LPP_1^{\mathbb{Q}}$, a probability logic similar to $LPP_2^{\mathbb{Q}}$, but with iterations of probabilistic operators allowed. The semantics is somewhat more complicated.
- $LPP_2^{Fr(n)}$, 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.

Other formally verified probability logics

- $LPP_1^{\mathbb{Q}}$, a probability logic similar to $LPP_2^{\mathbb{Q}}$, but with iterations of probabilistic operators allowed. The semantics is somewhat more complicated.
- $LPP_2^{Fr(n)}$, 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?