Primal infon logic: proof theory and efficient decidability

S P Suresh, Chennai Mathematical Institute[1]

Seminar on Logic and Cognition
Jadavpur University
October 31, 2013

[1] Joint work with A Baskar (IMSc), Prasad Naldurg (IBM Bangalore), and K R Raghavendra (IIIT Bangalore)

Outline

1. Primal infon logic
2. Some proof theory
3. Algorithm for derivability
4. Complexity analysis

Outline

Infon logic and authorization

- Infon logic: proposed by Yuri Gurevich and Itay Neeman of Microsoft Research.
- Part of the authorization system DKAL.
- In DKAL, principals use infon logic to derive consequences from their own knowledge and communications from other principals.

Infon logic and authorization ...

- If I have $[A \text{ said } (B \text{ can read file } X)] \rightarrow (B \text{ can read file } X)$ in my knowledge set and A communicates $(B \text{ can read file } X)$, I can grant access to B.
- If A tells B that C can read X provided C signs an agreement, it is modelled as $C \text{ agrees } \rightarrow [A \text{ implied } (C \text{ can read } X)]$.
- $A \text{ implied } x$ is less trusted than $A \text{ said } x$: $(A \text{ implied } x) \rightarrow x$ may not hold even when $(A \text{ said } x) \rightarrow x$.
- $A \text{ said } x \rightarrow A \text{ implied } x$.

Infon logic: syntax

- Infon logic is the $(\wedge, \rightarrow)$ fragment of intuitionistic logic, with modalities.
- Syntax of the logic:

$$\Phi ::= p \mid x \wedge y \mid x \rightarrow y \mid \square_a x \mid \blacksquare_a x$$

  where $p \in \mathit{Props}$, $a \in \mathit{Ag}$, and $x, y \in \Phi$.
- $\square_a x$ stands for a said x and $\blacksquare_a x$ stands for a implied x.

Infon logic: proof rules

$\dfrac{}{X, x \vdash x}\, ax$	$\dfrac{X \vdash x}{X, X' \vdash x}\, weaken$
$\dfrac{X \vdash x \quad X \vdash y}{X \vdash x \wedge y}\, \wedge i$	$\dfrac{X \vdash x_0 \wedge x_1}{X \vdash x_i}\, \wedge e_i$
$\dfrac{X, x \vdash y}{X \vdash x \rightarrow y}\, \rightarrow i$	$\dfrac{X \vdash x \quad X \vdash x \rightarrow y}{X \vdash y}\, \rightarrow e$
$\dfrac{X \vdash x}{\square_a X \vdash \square_a x}\, \square_a$	$\dfrac{X, Y \vdash x}{\square_a X, \blacksquare_a Y \vdash \blacksquare_a x}\, \blacksquare_a$

Problem of interest

The derivability problem: Given X and x, determine whether there is a proof of $X \vdash x$.

Example proofs

$$\dfrac{\dfrac{\dfrac{}{x, y \vdash x}\,ax \qquad \dfrac{}{x, y \vdash y}\,ax}{x, y \vdash x \wedge y}\,\wedge i}{\Box_a x, \blacksquare_a y \vdash \blacksquare_a (x \wedge y)}\,\blacksquare_a$$

Example proofs ...

$$
\begin{prooftree}
\AxiomC{}
\RightLabel{ax}
\UnaryInfC{$\Box_a x \wedge \Box_a y \vdash \Box_a x \wedge \Box_a y$}
\RightLabel{$\wedge e_1$}
\UnaryInfC{$\Box_a x \wedge \Box_a y \vdash \Box_a y$}
\AxiomC{}
\RightLabel{ax}
\UnaryInfC{$\Box_a x \wedge \Box_a y \vdash \Box_a x \wedge \Box_a y$}
\RightLabel{$\wedge e_0$}
\UnaryInfC{$\Box_a x \wedge \Box_a y \vdash \Box_a x$}
\AxiomC{}
\RightLabel{ax}
\UnaryInfC{$x, y \vdash x$}
\AxiomC{}
\RightLabel{ax}
\UnaryInfC{$x, y \vdash y$}
\RightLabel{$\wedge i$}
\BinaryInfC{$x, y \vdash x \wedge y$}
\RightLabel{$\Box_a$}
\UnaryInfC{$\Box_a x, \Box_a y \vdash \Box_a (x \wedge y)$}
\RightLabel{$\rightarrow i$}
\UnaryInfC{$\Box_a x \vdash \Box_a y \rightarrow \Box_a (x \wedge y)$}
\RightLabel{$\rightarrow i$}
\UnaryInfC{$\vdash \Box_a x \rightarrow (\Box_a y \rightarrow \Box_a (x \wedge y))$}
\RightLabel{$weaken$}
\UnaryInfC{$\Box_a x \wedge \Box_a y \vdash \Box_a x \rightarrow (\Box_a y \rightarrow \Box_a (x \wedge y))$}
\RightLabel{$\rightarrow e$}
\BinaryInfC{$\Box_a x \wedge \Box_a y \vdash \Box_a y \rightarrow \Box_a (x \wedge y)$}
\RightLabel{$\rightarrow e$}
\BinaryInfC{$\Box_a x \wedge \Box_a y \vdash \Box_a (x \wedge y)$}
\end{prooftree}
$$

A case for the *cut* rule

- Proof search is difficult if arbitrarily large formulas can occur in all proofs of $X \vdash x$.
- The *cut* rule can help in handling chains of implications.

$$\frac{X \vdash x \quad Y \vdash y}{X, Y - x \vdash y}\ \mathit{cut}$$

- Does not add power:

$$\dfrac{\begin{array}{c}\pi_1\\ \vdots\\ X \vdash x\end{array} \qquad \dfrac{\begin{array}{c}\pi_2\\ \vdots\\ Y \vdash y\end{array}}{Y - x \vdash x \to y} \to i}{X, Y - x \vdash y} \to e$$

A proof with *cut*

$$
\dfrac{
  \dfrac{
    \dfrac{}{\Box_a x \wedge \Box_a y \vdash \Box_a x \wedge \Box_a y}\,ax
  }{\Box_a x \wedge \Box_a y \vdash \Box_a y}\,\wedge e_1
  \qquad
  \dfrac{
    \dfrac{
      \dfrac{}{\Box_a x \wedge \Box_a y \vdash \Box_a x \wedge \Box_a y}\,ax
    }{\Box_a x \wedge \Box_a y \vdash \Box_a x}\,\wedge e_0
    \qquad
    \dfrac{
      \dfrac{
        \dfrac{}{x, y \vdash x}\,ax
        \qquad
        \dfrac{}{x, y \vdash y}\,ax
      }{x, y \vdash x \wedge y}\,\wedge i
    }{\Box_a x, \Box_a y \vdash \Box_a (x \wedge y)}\,\Box_a
  }{\Box_a x \wedge \Box_a y, \Box_a y \vdash \Box_a (x \wedge y)}\,cut
}{\Box_a x \wedge \Box_a y \vdash \Box_a (x \wedge y)}\,cut
$$

Primal infon logic

- Statman 1979 proves that the derivability problem for intuitionistic logic (even the $\rightarrow$-fragment) is PSPACE-complete.
- The main culprit is the $\rightarrow i$ rule.
- A variant suggests itself – primal implication:

$$\frac{X \vdash y}{X \vdash x \rightarrow y}$$

- A form of weakening.
- Shades of encryption:

$$\frac{X \vdash t}{X \vdash sk(A) \rightarrow t}$$

Primal infon logic: proof rules

$$\frac{}{X, x \vdash x}\ ax \qquad \frac{X \vdash x}{X, X' \vdash x}\ weaken$$

$$\frac{X \vdash x \quad X \vdash y}{X \vdash x \wedge y}\ \wedge i \qquad \frac{X \vdash x_0 \wedge x_1}{X \vdash x_i}\ \wedge e_i$$

$$\frac{X \vdash y}{X \vdash x \to y}\ \to i \qquad \frac{X \vdash x \quad X \vdash x \to y}{X \vdash y}\ \to e$$

$$\frac{X \vdash x}{\Box_a X \vdash \Box_a x}\ \Box_a \qquad \frac{X, Y \vdash x}{\Box_a X, \blacksquare_a Y \vdash \blacksquare_a x}\ \blacksquare_a$$

$$\frac{X \vdash x \quad Y \vdash y}{X, Y - x \vdash y}\ cut$$

Why the *cut* rule?

- The *cut* rule ought to be admissible in any reasonable system.
- But it can be shown that there is no *cut*-free proof of $\square_a x \wedge \square_a y \vdash \square_a(x \wedge y)$.
- Thus we add *cut* as an explicit rule.

Semantics

A Kripke structure for infon logic is a tuple $(W, \leq, C, S, I)$ where

- $(W, \leq)$ is a partially ordered set.
- $C : Props \to \wp(W)$ maps each $p \in Props$ to a cone.
- $S : a \mapsto S_a$ and $I : a \mapsto I_a$, where for all $a \in Ag$
    - $S_a \subseteq W \times W$ and $I_a \subseteq W \times W$.
    - $I_a \subseteq S_a$.
    - If $u \leq w$ and wS_av then uS_av.
    - If $u \leq w$ and wI_av then uI_av.

Semantics ...

We assign a cone $C(z)$ to every formula z.

- $C(x \wedge y) = C(x) \cap C(y)$.
- $C(\square_a x) = \{u \mid \forall v : uS_a v \Rightarrow v \in C(x)\}$.
- $C(\blacksquare_a x) = \{u \mid \forall v : uI_a v \Rightarrow v \in C(x)\}$.
- Full infon logic: $C(x \rightarrow y) = \{u \mid \forall v \geq u : v \in C(x) \Rightarrow v \in C(y)\}$.
- Primal infon logic: $C(x \rightarrow y)$ is an arbitrary cone C such that

$$C(y) \subseteq C \subseteq \{u \mid \forall v \geq u : v \in C(x) \Rightarrow v \in C(y)\}.$$

Semantics ...

Theorem

For both full infon logic and primal infon logic, the following are equivalent for any sequent s.

1. *s is provable.*
2. *s is valid.*
3. *Every finite Kripke structure models s.*
4. *There is a proof of s that uses only subformulas of s.*

Known results

- Full infon logic is PSPACE-complete. (Gurevich and Neeman (2009).)
- Primal constructive logic (PIL without the modalities) is solvable in linear time [GN09].
- Primal infon logic with only the $\Box_a$ modalities is solvable in linear time [GN09].
- Primal infon logic extended with disjunctions is PSPACE-complete. Proved by Beklemishev and Gurevich (2012).
- Gurevich and Savateev (2011) have proved exponential lower bounds on proof size in primal infon logic.
- [BNRS13]: Primal infon logic is solvable in polynomial time ($O(N^3)$ algorithm).

Outline

cut and subformula property

- The *cut* rule also renders proof search difficult, by violating the subformula property.
- Standard solution: prove that every provable sequent has a *cut*-free proof.
- But …*cut* is not eliminable in PIL.
- What do we do?

Sequent calculus system for PIL

$\dfrac{}{X, x \vdash x}\ ax$	$\dfrac{X \vdash x}{X, X' \vdash x}\ weaken$
$\dfrac{X \vdash x \quad X \vdash y}{X \vdash x \wedge y}\ \wedge r$	$\dfrac{X, x_i \vdash y}{X, x_0 \wedge x_1 \vdash y}\ \wedge \ell_i$
$\dfrac{X \vdash y}{X \vdash x \rightarrow y}\ \rightarrow r$	$\dfrac{X \vdash x \quad X, y \vdash z}{X, x \rightarrow y \vdash z}\ \rightarrow \ell$
$\dfrac{X \vdash x}{\Box_a X \vdash \Box_a x}\ \Box_a$	$\dfrac{X, Y \vdash x}{\Box_a X, \blacksquare_a Y \vdash \blacksquare_a x}\ \blacksquare_a$

$$\dfrac{X \vdash x \quad Y \vdash y}{X, Y - x \vdash y}\ cut$$

Equivalence

Translate

$$\dfrac{\begin{array}{c}\pi_1\\ \vdots\\ X \vdash x \to y\end{array} \quad \begin{array}{c}\pi_2\\ \vdots\\ X \vdash x\end{array}}{X_1 \vdash y} \to e$$

to

$$\dfrac{\begin{array}{c}\pi_1'\\ \vdots\\ X \vdash x \to y\end{array} \quad \dfrac{\begin{array}{c}\pi_2'\\ \vdots\\ X \vdash x\end{array} \quad \dfrac{}{X, y \vdash y}\, ax}{X, x \to y \vdash y} \to \ell}{X \vdash y}\, cut$$

No new formulas!

Equivalence

Translate

$$\dfrac{\begin{array}{c}\pi'_1\\ \vdots\\ X \vdash x\end{array} \quad \begin{array}{c}\pi'_2\\ \vdots\\ X, y \vdash z\end{array}}{X, x \to y \vdash z} \to \ell$$

to

$$\dfrac{\dfrac{\dfrac{}{X, x \to y \vdash x \to y}\,ax \quad \begin{array}{c}\pi_1\\ \vdots\\ X \vdash x\end{array}}{X, x \to y \vdash y} \to e \quad \begin{array}{c}\pi_2\\ \vdots\\ X, y \vdash z\end{array}}{X, x \to y \vdash z}\,cut$$

No new formulas!

Cut elimination for PIL in sequent calculus form

- Proof is along standard lines.
- Immediately implies the subformula property.
- $\Box_a x \wedge \Box_a y \vdash \Box_a(x \wedge y)$ is proved as follows:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{x, y \vdash x}\,ax \qquad \dfrac{}{x, y \vdash y}\,ax}{x, y \vdash x \wedge y}\,\wedge r}{\Box_a x, \Box_a y \vdash \Box_a(x \wedge y)}\,\Box_a}{\Box_a x \wedge \Box_a y, \Box_a y \vdash \Box_a(x \wedge y)}\,\wedge\ell_1}{\Box_a x \wedge \Box_a y \vdash \Box_a(x \wedge y)}\,\wedge\ell_2$$

Subformula property for PIL

- If $X \vdash_{nd} x$ then $X \vdash_{sc} x$.
- If $X \vdash_{sc} x$ then there is a cut-free sequent calculus proof of $X \vdash x$.
- All formulas occurring in any cut-free sequent calculus proof of $X \vdash x$ belong to $sf(X \cup \{x\})$.
- This last proof can be translated to a natural deduction proof respecting the subformula property.

Outline

1. Primal infon logic
2. Some proof theory
3. Algorithm for derivability
4. Complexity analysis

Setting it up

- Given X_0 and x_0, to check if $X_0 \vdash x_0$.
- Let $Y_0 = sf(X_0 \cup \{x_0\})$.
- Let $|Y_0| = N$.
- $closure'(X) = \{x \mid X \vdash x \text{ without using the modality rules}\}$.
- For $X \subseteq Y_0$, $closure'(X)$ computable in $O(N)$ time.
- $closure(X) = \{x \mid X \vdash x\}$.

Setting it up …

- Let $\mathscr{C}$ be the set of modal contexts in Y_0.
- For each $\sigma \in \mathscr{C}$ define $f_\sigma : \wp(Y_0) \to \wp(Y_0)$ and $g_\sigma : \wp(Y_0) \to \wp(Y_0)$.
- f_σ handles applications of the *cut* rule.
- g_σ handles one application of each of the modality rules.
- Mutually recursive procedures.

Theorem

For all $X \subseteq Y_0$, $f_\varepsilon(X) = \mathit{closure}(X)$.

Computing *closure*(X)

function $f_\sigma(X)$
    if $(\sigma \notin \mathscr{C}$ or $X = \varnothing)$ **then**
        return $\varnothing$
    end if
    $Y \leftarrow X$
    while $Y \neq g_\sigma(Y)$ **do**
        $Y \leftarrow g_\sigma(Y)$
    end while
    return Y
end function

Computing $closure(X)$

function $g_\sigma(X)$
  for all $a \in Ag : Y_a \leftarrow \Box_a f_{\sigma\Box_a}(\Box_a^{-1}(X))$
  for all $a \in Ag : Z_a \leftarrow \blacksquare_a f_{\sigma\blacksquare_a}(\Box_a^{-1}(X) \cup \blacksquare_a^{-1}(X))$
  return $closure'(X \cup \bigcup_{a \in Ag}(Y_a \cup Z_a))$
end function

Outline

Number of distinct recursive calls

With respect to the run of $f_\varepsilon(X_0)$

- $(\sigma, X) \rightarrow_f (\tau, Y)$ if $f_\sigma(X)$ is an (temporally) earlier recursive call than $f_\tau(Y)$.
- $(\sigma, X) \rightarrow_g (\tau, Y)$ if $g_\sigma(X)$ is an (temporally) earlier recursive call than $g_\tau(Y)$.

Lemma

Suppose $\sigma \in \mathscr{C}$, *and* $X, Y \subseteq Y_0$.

1. *If* $(\sigma, X) \rightarrow_f (\sigma, Y)$ *then* $f_\sigma(X) \subseteq Y$.
2. *If* $(\sigma, X) \rightarrow_g (\sigma, Y)$ *then* $g_\sigma(X) \subseteq Y$.

Computing $closure(X)$ with memoization

Initialization: for all $\sigma \in \mathscr{C} : G_\sigma \leftarrow \emptyset$

function $f(\sigma, X)$
    if $\sigma \notin \mathscr{C}$ or $X = \emptyset$ **then**
        return $\emptyset$
    end if
    $Y \leftarrow X$
    while $Y \neq G_\sigma$ **do**   ▷ $G_\sigma = g(\sigma, G_\sigma)$ before the start of the loop.
        $G_\sigma \leftarrow Y$
        $Y \leftarrow g(\sigma, Y)$
    end while   ▷ $G_\sigma = g(\sigma, G_\sigma)$ at the end of the loop.
    return G_σ
end function

$O(N^3)$ complexity

- At most N modal contexts.
- For each context σ, across all calls to f_σ, at most N recursive calls to g_σ.
- At most N^2 calls to g_σ, across all σ.
- Each g_σ makes a constant number of recursive calls to f_τ's.
- Each g_σ takes $O(N)$ time to compute $\mathit{closure}'$.
- Overall time: $O(N^3)$.

Questions?

Thank you!