Primal implication as encryption

Vladimir N. Krupski

Lomonosov Moscow State University

June 2013

Propositional Infon Logic (Y. Gurevich, I. Neeman, 2008)
Distributed-Knowledge Authorization Language DKAL

Propositional Infon Logic (Y. Gurevich, I. Neeman, 2008)
Distributed-Knowledge Authorization Language DKAL

- Infon — a message as a piece of information.
- $\Gamma \vdash \varphi$ — "the principal can get (by herself, without any communication) the information φ provided she already has all infons $\psi \in \Gamma$".

Propositional Infon Logic (Y. Gurevich, I. Neeman, 2008)
Distributed-Knowledge Authorization Language DKAL

- Infon — a message as a piece of information.
- $\Gamma \vdash \varphi$ — "the principal can get (by herself, without any communication) the information φ provided she already has all infons $\psi \in \Gamma$".

General Infon Logic = intuitionistic propositional logic + quotation modalities `A_said()`, `B_said()`, ... (PSPACE)

Primal Infon Logic = its efficient fragment. (Linear TIME)

Primal implication $\to_p$

Primal implication $\to_p$

$\Gamma, \varphi \vdash \varphi$, (but $\nvdash \varphi \to_p \varphi$)

Primal implication $\to_p$

$\Gamma, \varphi \vdash \varphi$, (but $\nvdash \varphi \to_p \varphi$)

$$\frac{\Gamma \vdash \psi}{\Gamma \vdash \varphi \to_p \psi}(\to_p I) \quad , \quad \frac{\Gamma \vdash \varphi \quad \Gamma \vdash \varphi \to_p \psi}{\Gamma \vdash \psi}(\to_p E)\,.$$

Primal implication $\to_p$

$\Gamma, \varphi \vdash \varphi$, (but $\nvdash \varphi \to_p \varphi$)

$$\frac{\Gamma \vdash \psi}{\Gamma \vdash \varphi \to_p \psi}(\to_p I) \quad , \quad \frac{\Gamma \vdash \varphi \quad \Gamma \vdash \varphi \to_p \psi}{\Gamma \vdash \psi}(\to_p E) \; .$$

We propose a "cryptographic" interpretation:

- $\varphi \to_p \psi$ — "an infon, containing the information ψ encrypted by a symmetric key (generated from) φ".

Primal implication $\to_p$

$\Gamma, \varphi \vdash \varphi$, (but $\nvdash \varphi \to_p \varphi$)

$$\frac{\Gamma \vdash \psi}{\Gamma \vdash \varphi \to_p \psi}(\to_p I) \quad , \quad \frac{\Gamma \vdash \varphi \quad \Gamma \vdash \varphi \to_p \psi}{\Gamma \vdash \psi}(\to_p E)\ .$$

We propose a "cryptographic" interpretation:

- $\varphi \to_p \psi$ — "an infon, containing the information ψ encrypted by a symmetric key (generated from) φ".
- $(\to_p I)$ allows to encrypt any available message by any key.

Primal implication $\to_p$

$\Gamma, \varphi \vdash \varphi$, (but $\nvdash \varphi \to_p \varphi$)

$$\frac{\Gamma \vdash \psi}{\Gamma \vdash \varphi \to_p \psi}\,(\to_p I) \quad , \quad \frac{\Gamma \vdash \varphi \quad \Gamma \vdash \varphi \to_p \psi}{\Gamma \vdash \psi}\,(\to_p E)\ .$$

We propose a "cryptographic" interpretation:

- $\varphi \to_p \psi$ — "an infon, containing the information ψ encrypted by a symmetric key (generated from) φ".
- $(\to_p I)$ allows to encrypt any available message by any key.
- $(\to_p E)$ allows to extract the information from a ciphertext provided the key is also available.

Primal Infon Logic incorporated into communication protocols

It is a natural tool for manipulating with commitment schemes without detailed analysis of the scheme itself.

Primal Infon Logic incorporated into communication protocols

It is a natural tool for manipulating with commitment schemes without detailed analysis of the scheme itself.

Example

Alice and Bob live in different places and communicate via a telephone line or by e-mail. They wish to play the following game distantly. Each of them picks a bit, randomly or somehow else. If the bits coincide then Alice wins; otherwise Bob wins. Both of them decide to play fair but don't believe in the fairness of the opponent. So they use cryptography.

To play fair means that they honestly declare their choice of a bit, independently of what the other player said.

$\mathtt{X\_said()}$ — corresponds to the modal logic **K**.

$\mathtt{X\_IsTrustedOn}\,\varphi := \mathtt{X\_said}\,\varphi \to_p \varphi$.

$\texttt{X\_said}()$ — corresponds to the modal logic **K**.

$\texttt{X\_IsTrustedOn}\,\varphi := \texttt{X\_said}\,\varphi \to_p \varphi$.

Policy: Alice

Γ : $\texttt{A\_said}\ m_a$, $\texttt{A\_said}\ k_a$,
$\texttt{A\_IsTrustedOn}\ m_a$,
$\texttt{A\_IsTrustedOn}\ k_a$.

`X_said()` — corresponds to the modal logic **K**.

$\texttt{X\_IsTrustedOn}\,\varphi := \texttt{X\_said}\,\varphi \to_p \varphi$.

Policy: Alice

Γ : $\texttt{A\_said}\ m_a$, $\texttt{A\_said}\ k_a$,
$\texttt{A\_IsTrustedOn}\ m_a$,
$\texttt{A\_IsTrustedOn}\ k_a$.

$\Gamma \vdash k_a \to_p m_a$; $\texttt{SEND}\ k_a \to_p m_a$.

$\mathtt{X\_said()}$ — corresponds to the modal logic **K**.

$\mathtt{X\_IsTrustedOn}\,\varphi := \mathtt{X\_said}\,\varphi \to_p \varphi$.

Policy: Alice

Γ : $\mathtt{A\_said}\ m_a$, $\mathtt{A\_said}\ k_a$,
$\mathtt{A\_IsTrustedOn}\ m_a$,
$\mathtt{A\_IsTrustedOn}\ k_a$.

$\Gamma \vdash k_a \to_p m_a$; $\mathtt{SEND}\ k_a \to_p m_a$.

$$\dfrac{\dfrac{\mathtt{A\_said}\,m_a \quad \mathtt{A\_IsTrustedOn}\,m_a}{m_a}}{k_a \to_p m_a}$$

$\texttt{X\_said}()$ — corresponds to the modal logic **K**.

$\texttt{X\_IsTrustedOn}\,\varphi := \texttt{X\_said}\,\varphi \to_p \varphi$.

Policy: Alice

Γ : $\texttt{A\_said}\ m_a$, $\texttt{A\_said}\ k_a$,
$\texttt{A\_IsTrustedOn}\ m_a$,
$\texttt{A\_IsTrustedOn}\ k_a$.

$\Gamma \vdash k_a \to_p m_a$; $\texttt{SEND}\ k_a \to_p m_a$.

$\texttt{when gets}\ k_b \to_p m_b$:
$\Gamma := \Gamma, \texttt{B\_said}\,(k_b \to_p m_b)$;
$\Gamma \vdash k_a$; $\texttt{SEND}\ k_a$.

$\texttt{X\_said}()$ — corresponds to the modal logic **K**.

$\texttt{X\_IsTrustedOn}\,\varphi := \texttt{X\_said}\,\varphi \to_p \varphi$.

Policy: Alice

Γ : $\texttt{A\_said}\ m_a$, $\texttt{A\_said}\ k_a$,
$\texttt{A\_IsTrustedOn}\ m_a$,
$\texttt{A\_IsTrustedOn}\ k_a$.

$\Gamma \vdash k_a \to_p m_a$; $\texttt{SEND}\ k_a \to_p m_a$.

$\texttt{when gets}\ k_b \to_p m_b$:
$\Gamma := \Gamma, \texttt{B\_said}\,(k_b \to_p m_b)$;
$\Gamma \vdash k_a$; $\texttt{SEND}\ k_a$.

$\texttt{when gets}\ k_b$: $\Gamma := \Gamma, \texttt{B\_said}\,k_b$.

$\mathtt{X\_said}()$ — corresponds to the modal logic **K**.

$\mathtt{X\_IsTrustedOn}\,\varphi := \mathtt{X\_said}\,\varphi \to_p \varphi$.

Policy: Alice

Γ : $\mathtt{A\_said}\ m_a$, $\mathtt{A\_said}\ k_a$,
$\mathtt{A\_IsTrustedOn}\ m_a$,
$\mathtt{A\_IsTrustedOn}\ k_a$.

$\Gamma \vdash k_a \to_p m_a$; $\mathtt{SEND}\ k_a \to_p m_a$.

$\mathtt{when\ \ gets}\ k_b \to_p m_b$:
$\Gamma := \Gamma, \mathtt{B\_said}\,(k_b \to_p m_b)$;
$\Gamma \vdash k_a$; $\mathtt{SEND}\ k_a$.

$\mathtt{when\ \ gets}\ k_b$: $\Gamma := \Gamma, \mathtt{B\_said}\,k_b$.

$\Gamma \vdash \mathtt{B\_said}\ m_b$, $\Gamma \vdash \mathtt{A\_said}\ m_a$.

X_said() — corresponds to the modal logic **K**.

X_IsTrustedOn φ := X_said $\varphi \to_p \varphi$.

Policy: Alice

Γ : A_said m_a, A_said k_a,
A_IsTrustedOn m_a,
A_IsTrustedOn k_a.

$\Gamma \vdash k_a \to_p m_a$; SEND $k_a \to_p m_a$.

when gets $k_b \to_p m_b$:
$\Gamma := \Gamma$, B_said $(k_b \to_p m_b)$;
$\Gamma \vdash k_a$; SEND k_a.

when gets k_b: $\Gamma := \Gamma$, B_said k_b.

$\Gamma \vdash$ B_said m_b, $\Gamma \vdash$ A_said m_a.

Policy: Bob

Γ : B_said m_b, B_said k_b,
B_IsTrustedOn m_b,
B_IsTrustedOn k_b.

$\Gamma \vdash k_b \to_p m_b$; SEND $k_b \to_p m_b$.

when gets $k_a \to_p m_a$:
$\Gamma := \Gamma$, A_said $(k_a \to_p m_a)$;
$\Gamma \vdash k_b$; SEND k_b.

when gets k_a: $\Gamma := \Gamma$, A_said k_a.

$\Gamma \vdash$ A_said m_a, $\Gamma \vdash$ B_said m_b.

What are the values of infon formulas?
What is stored in the memory sells and sent?

The "cryptographic" semantics gives some answer.

What are the values of infon formulas?
What is stored in the memory sells and sent?

The “cryptographic” semantics gives some answer.

- In what follows we do not insist that the encryption is strong in some sense. One may assume that the privacy is protected by the interface: an agent simply has no tools that make the decryption of a ciphertext without key possible.

What are the values of infon formulas?
What is stored in the memory sells and sent?

The "cryptographic" semantics gives some answer.

- In what follows we do not insist that the encryption is strong in some sense. One may assume that the privacy is protected by the interface: an agent simply has no tools that make the decryption of a ciphertext without key possible.

Example

$\texttt{cp := CodePage(hash(}\varphi\texttt{))}$

$\varphi \to_p \psi$ $\texttt{:= convert}$ ψ $\texttt{to cp}$

What are the values of infon formulas?
What is stored in the memory sells and sent?

The "cryptographic" semantics gives some answer.

- In what follows we do not insist that the encryption is strong in some sense. One may assume that the privacy is protected by the interface: an agent simply has no tools that make the decryption of a ciphertext without key possible.
- We consider the purely propositional language and leave the modalities for the future.

P — the $\{\top, \wedge, \to_p\}$-fragment.

$$\frac{}{\top} \qquad \frac{\varphi_1 \quad \varphi_2}{\varphi_1 \wedge \varphi_2} \qquad \frac{\varphi_1 \wedge \varphi_2}{\varphi_i} \qquad \frac{\varphi_2}{\varphi_1 \to_p \varphi_2} \qquad \frac{\varphi_1 \quad \varphi_1 \to_p \varphi_2}{\varphi_2}$$

P — the $\{\top, \wedge, \to_p\}$-fragment.

$$\frac{}{\top} \qquad \frac{\varphi_1 \quad \varphi_2}{\varphi_1 \wedge \varphi_2} \qquad \frac{\varphi_1 \wedge \varphi_2}{\varphi_i} \qquad \frac{\varphi_2}{\varphi_1 \to_p \varphi_2} \qquad \frac{\varphi_1 \quad \varphi_1 \to_p \varphi_2}{\varphi_2}$$

Theorem (L. Beklemishev, Y. Gurevich, 2012)

P *is sound and complete w.r.t. quasi-boolean semantics.*

$\models$ is a quasi-boolean model iff

- $\models \top$,
- $\models \varphi_1 \wedge \varphi_2 \;\Leftrightarrow\; \models \varphi_1$ and $\models \varphi_2$,
- $\models \varphi_2 \;\Rightarrow\; \models \varphi_1 \to_p \varphi_2$,
- $\models \varphi_1 \to_p \varphi_2 \;\Rightarrow\; \not\models \varphi_1$ or $\models \varphi_2$.

P — the $\{\top, \wedge, \to_p\}$-fragment.

$$\frac{}{\top} \qquad \frac{\varphi_1 \quad \varphi_2}{\varphi_1 \wedge \varphi_2} \qquad \frac{\varphi_1 \wedge \varphi_2}{\varphi_i} \qquad \frac{\varphi_2}{\varphi_1 \to_p \varphi_2} \qquad \frac{\varphi_1 \quad \varphi_1 \to_p \varphi_2}{\varphi_2}$$

Theorem (L. Beklemishev, Y. Gurevich, 2012)

P *is sound and complete w.r.t. quasi-boolean semantics.*

$\models$ is a quasi-boolean model iff

- $\models \top$,
- $\models \varphi_1 \wedge \varphi_2 \;\Leftrightarrow\; \models \varphi_1$ and $\models \varphi_2$,
- $\models \varphi_2 \;\Rightarrow\; \models \varphi_1 \to_p \varphi_2$,
- $\models \varphi_1 \to_p \varphi_2 \;\Rightarrow\; \not\models \varphi_1$ or $\models \varphi_2$.

But it is not what we need.

Infon algebra $\mathcal{A} = \langle \Sigma^*, \pi, l, r, enc, dec, E \rangle$

Infon algebra $\mathcal{A} = \langle \Sigma^*, \pi, l, r, enc, dec, E \rangle$

- Σ is a finite alphabet, say $\Sigma = \{0, 1\}$.

Infon algebra $\mathcal{A} = \langle \Sigma^*, \pi, l, r, enc, dec, E \rangle$

- Σ is a finite alphabet, say $\Sigma = \{0, 1\}$.
- $\pi : (\Sigma^*)^2 \longrightarrow \Sigma^*$ is a total pairing function with projections l, r:
$$l(\pi(x, y)) = x, \quad r(\pi(x, y)) = y.$$

Infon algebra $\mathcal{A} = \langle \Sigma^*, \pi, l, r, enc, dec, E \rangle$

- Σ is a finite alphabet, say $\Sigma = \{0, 1\}$.
- $\pi : (\Sigma^*)^2 \to \Sigma^*$ is a total pairing function with projections l, r:

$$l(\pi(x, y)) = x, \quad r(\pi(x, y)) = y.$$

- $enc, dec : (\Sigma^*)^2 \to \Sigma^*$ — encoding/decoding methods, enc is total,

$$dec(x, enc(x, y)) = y.$$

Infon algebra $\mathcal{A} = \langle \Sigma^*, \pi, l, r, enc, dec, E \rangle$

- Σ is a finite alphabet, say $\Sigma = \{0, 1\}$.
- $\pi : (\Sigma^*)^2 \to \Sigma^*$ is a total pairing function with projections l, r:
$$l(\pi(x, y)) = x, \quad r(\pi(x, y)) = y.$$
- $enc, dec : (\Sigma^*)^2 \to \Sigma^*$ — encoding/decoding methods, enc is total,
$$dec(x, enc(x, y)) = y.$$
- $E \subset \Sigma^*$, $E \neq \emptyset$ – the information known by everyone.

Definition

A set $M \subseteq \Sigma^*$ is *closed* if $E \subseteq M$ and M satisfies the closure conditions:

- $a, b \in M \Leftrightarrow \pi(a, b) \in M$,
- $a \in \Sigma^*,\ b \in M \Rightarrow enc(a, b) \in M$.
- $a, enc(a, b) \in M \Rightarrow b \in M$,

Definition

A set $M \subseteq \Sigma^*$ is *closed* if $E \subseteq M$ and M satisfies the closure conditions:

- $a, b \in M \Leftrightarrow \pi(a, b) \in M$,
- $a \in \Sigma^*,\ b \in M \Rightarrow enc(a, b) \in M$.
- $a, enc(a, b) \in M \Rightarrow b \in M$,

A closed set M represents the information that is potentially available to an agent in a local state. M contains all public and some private texts.

Definition

A set $M \subseteq \Sigma^*$ is *closed* if $E \subseteq M$ and M satisfies the closure conditions:

- $a, b \in M \Leftrightarrow \pi(a, b) \in M$,
- $a \in \Sigma^*,\ b \in M \Rightarrow enc(a, b) \in M$.
- $a, enc(a, b) \in M \Rightarrow b \in M$,

A closed set M represents the information that is potentially available to an agent in a local state. M contains all public and some private texts.
The agent can combine several texts in a single multi-part document using π and extract its parts by means of projections.

Definition

A set $M \subseteq \Sigma^*$ is *closed* if $E \subseteq M$ and M satisfies the closure conditions:

- $a, b \in M \Leftrightarrow \pi(a, b) \in M$,
- $a \in \Sigma^*,\ b \in M \Rightarrow enc(a, b) \in M$.
- $a, enc(a, b) \in M \Rightarrow b \in M$,

A closed set M represents the information that is potentially available to an agent in a local state. M contains all public and some private texts.
The agent can combine several texts in a single multi-part document using π and extract its parts by means of projections.

She has access to the encryption tool *enc*, so she can convert a plaintext into a ciphertext. The backward conversion (by *dec*) is also available provided she has the encryption key.

Definition

A model is a triple $\langle \mathcal{A}, M, v\rangle$ where $\mathcal{A}$ is an infon algebra, $M \subseteq \Sigma^*$ is a closed set and $v\colon Fm \to \Sigma^*$ is an evaluation,

- $v(\top) \in E$,
- $v(\varphi_1 \wedge \varphi_2) = \pi(v(\varphi_1),\ v(\varphi_2))$,
- $v(\varphi_1 \to_p \varphi_2) = enc(v(\varphi_1), v(\varphi_2))$.

Definition

A model is a triple $\langle \mathcal{A}, M, v \rangle$ where $\mathcal{A}$ is an infon algebra, $M \subseteq \Sigma^*$ is a closed set and $v \colon Fm \rightarrow \Sigma^*$ is an evaluation,

- $v(\top) \in E$,
- $v(\varphi_1 \wedge \varphi_2) = \pi(v(\varphi_1),\ v(\varphi_2))$,
- $v(\varphi_1 \rightarrow_p \varphi_2) = enc(v(\varphi_1), v(\varphi_2))$.

Theorem (Soundness and Completeness)

*$\Gamma \vdash \varphi$ in **P** iff $v(\varphi) \in M$ for every model $\langle \mathcal{A}, M, v \rangle$ with $v(\Gamma) \subseteq M$.*

Definition

A model is a triple $\langle \mathcal{A}, M, v \rangle$ where $\mathcal{A}$ is an infon algebra, $M \subseteq \Sigma^*$ is a closed set and $v \colon Fm \to \Sigma^*$ is an evaluation,

- $v(\top) \in E$,
- $v(\varphi_1 \wedge \varphi_2) = \pi(v(\varphi_1),\ v(\varphi_2))$,
- $v(\varphi_1 \to_p \varphi_2) = enc(v(\varphi_1), v(\varphi_2))$.

Theorem (Soundness and Completeness)

$\Gamma \vdash \varphi$ in **P** *iff $v(\varphi) \in M$ for every model $\langle \mathcal{A}, M, v \rangle$ with $v(\Gamma) \subseteq M$.*

Theorem (Uniform model)

There exists an interpretation $\langle \mathcal{A}, v \rangle$ with the following property: for any context Γ there exists a model $\langle \mathcal{A}, M, v \rangle$ with $v(\Gamma) \subseteq M$, such that $\Gamma \nvdash \varphi$ implies $v(\varphi) \notin M$ for all infons φ.

Constant $\perp$ and backdoors

Constant $\perp$ and backdoors

P$[\perp]$:

$$\frac{\perp}{\varphi}\,(\perp E)$$

$\perp$ as superuser permissions, makes communications and all other tools useless for the owner.

P$[\perp_w]$:

$$\frac{\perp \quad \varphi \to_p \psi}{\psi}\,(\perp_w E)$$

$\perp$ as a universal key, provides the ability to decrypt any available ciphertext.

Constant $\perp$ and backdoors

$\mathbf{P}[\perp]$:

$$\frac{\perp}{\varphi}\,(\perp E)$$

$\perp$ as superuser permissions, makes communications and all other tools useless for the owner.

$\mathbf{P}[\perp_w]$:

$$\frac{\perp \quad \varphi \to_p \psi}{\psi}\,(\perp_w E)$$

$\perp$ as a universal key, provides the ability to decrypt any available ciphertext.

$$\Sigma_\perp = \Sigma \cup \{\mathbf{f}\}, \quad v : Fm \to \Sigma_\perp^*, \quad v(\perp) = \mathbf{f}.$$

Constant $\perp$ and backdoors

P$[\perp]$:

$$\frac{\perp}{\varphi}\,(\perp E)$$

$\perp$ as superuser permissions, makes communications and all other tools useless for the owner.

P$[\perp_w]$:

$$\frac{\perp \quad \varphi \to_p \psi}{\psi}\,(\perp_w E)$$

$\perp$ as a universal key, provides the ability to decrypt any available ciphertext.

$$\Sigma_\perp = \Sigma \cup \{\mathbf{f}\}, \quad v : Fm \to \Sigma_\perp^*, \quad v(\perp) = \mathbf{f}.$$

$\mathbf{f} \in M\, , a \in \Sigma_\perp^* \Rightarrow a \in M$

$\mathbf{f}, enc(a, b) \in M \Rightarrow b \in M$

Constant $\perp$ and backdoors

P[$\perp$] :

$$\frac{\perp}{\varphi}\,(\perp E)$$

$\perp$ as superuser permissions, makes communications and all other tools useless for the owner.

P[$\perp_w$] :

$$\frac{\perp \quad \varphi \to_p \psi}{\psi}\,(\perp_w E)$$

$\perp$ as a universal key, provides the ability to decrypt any available ciphertext.

$$\Sigma_\perp = \Sigma \cup \{\mathbf{f}\}, \quad v : Fm \to \Sigma_\perp^*, \quad v(\perp) = \mathbf{f}.$$

$\mathbf{f} \in M\,, a \in \Sigma_\perp^* \Rightarrow a \in M$

$\mathbf{f}, enc(a, b) \in M \Rightarrow b \in M$

$crack(\mathbf{f}, enc(a, b)) = b$

Theorem

The completeness results for $\mathbf{P}[\bot]$ *and* $\mathbf{P}[\bot_w]$ *are just the same.*

Theorem

The completeness results for $\mathbf{P}[\bot]$ *and* $\mathbf{P}[\bot_w]$ *are just the same.*

Complexity: all known primal infon logics have linear time complexity. $\mathbf{P}[\bot_w]$ is a new one.

Theorem

The completeness results for $\mathbf{P}[\perp]$ *and* $\mathbf{P}[\perp_w]$ *are just the same.*

Complexity: all known primal infon logics have linear time complexity. $\mathbf{P}[\perp_w]$ is a new one.

Theorem

"$\Gamma \vdash \varphi$ *in* $\mathbf{P}[\perp_w]$*" is linear time decidable.*

Theorem

The completeness results for $\mathbf{P}[\bot]$ *and* $\mathbf{P}[\bot_w]$ *are just the same.*

Complexity: all known primal infon logics have linear time complexity. $\mathbf{P}[\bot_w]$ is a new one.

Theorem

"$\Gamma \vdash \varphi$ *in* $\mathbf{P}[\bot_w]$*" is linear time decidable.*

- `if` $\Gamma \vdash \varphi$ in $\mathbf{P}$, `return ''yes'';`
- `else if` $\Gamma \nvdash \bot$ in $\mathbf{P}$, `return ''no'';`
- `else return` $At^+(\varphi) \subseteq At^+(\Gamma)$.

where $At^+(\varphi)$ is the set of all atoms that occur "positive" in φ;
$At^+(\varphi \to_p \psi) = At^+(\psi)$.

Primal disjunction $\vee_p$

Primal disjunction $\vee_p$

P$[\vee_p]$ is the purely propositional part of **PPIL** (the recent stable formulation of the primal infon logic, C. Cotrini, Y. Gurevish, 2012)

$$\frac{\varphi}{\varphi \vee_p \psi} \qquad \frac{\psi}{\varphi \vee_p \psi}$$

(no elimination rules for $\vee_p$)

Primal disjunction $\vee_p$

$\mathbf{P}[\vee_p]$ is the purely propositional part of **PPIL** (the recent stable formulation of the primal infon logic, C. Cotrini, Y. Gurevish, 2012)

$$\frac{\varphi}{\varphi \vee_p \psi} \qquad \frac{\psi}{\varphi \vee_p \psi} \qquad \text{(no elimination rules for } \vee_p\text{)}$$

"Cryptograpic" interpretation: $(\varphi_1 \vee_p \varphi_2)$ is a group key.

Primal disjunction $\vee_p$

$\mathbf{P}[\vee_p]$ is the purely propositional part of **PPIL** (the recent stable formulation of the primal infon logic, C. Cotrini, Y. Gurevish, 2012)

$$\frac{\varphi}{\varphi \vee_p \psi} \qquad \frac{\psi}{\varphi \vee_p \psi}$$

(no elimination rules for $\vee_p$)

"Cryptograpic" interpretation: $(\varphi_1 \vee_p \varphi_2)$ is a group key.

$(\varphi_1 \vee_p \varphi_2) \to_p \psi$ is a ciphertext that can be decrypted by anyone who has at least one of the keys φ_1 or φ_2:

Primal disjunction $\vee_p$

P[$\vee_p$] is the purely propositional part of **PPIL** (the recent stable formulation of the primal infon logic, C. Cotrini, Y. Gurevish, 2012)

$$\frac{\varphi}{\varphi \vee_p \psi} \qquad \frac{\psi}{\varphi \vee_p \psi} \qquad \text{(no elimination rules for } \vee_p\text{)}$$

"Cryptograpic" interpretation: $(\varphi_1 \vee_p \varphi_2)$ is a group key.

$(\varphi_1 \vee_p \varphi_2) \rightarrow_p \psi$ is a ciphertext that can be decrypted by anyone who has at least one of the keys φ_1 or φ_2:

$$\frac{\varphi_1 \quad (\varphi_1 \vee_p \varphi_2) \rightarrow_p \psi}{\psi} \qquad \frac{\varphi_2 \quad (\varphi_1 \vee_p \varphi_2) \rightarrow_p \psi}{\psi} \qquad \text{are admissible}$$

Primal disjunction $\vee_p$

P$[\vee_p]$ is the purely propositional part of **PPIL** (the recent stable formulation of the primal infon logic, C. Cotrini, Y. Gurevish, 2012)

$$\frac{\varphi}{\varphi \vee_p \psi} \qquad \frac{\psi}{\varphi \vee_p \psi} \qquad \text{(no elimination rules for } \vee_p\text{)}$$

"Cryptograpic" interpretation: $(\varphi_1 \vee_p \varphi_2)$ is a group key.

$(\varphi_1 \vee_p \varphi_2) \rightarrow_p \psi$ is a ciphertext that can be decrypted by anyone who has at least one of the keys φ_1 or φ_2:

$(\varphi_1 \rightarrow_p \psi) \wedge (\varphi_2 \rightarrow_p \psi)$ makes the same in **P**, but here ψ is repeated twice.

Primal disjunction $\vee_p$

$\mathbf{P}[\vee_p]$ is the purely propositional part of **PPIL** (the recent stable formulation of the primal infon logic, C. Cotrini, Y. Gurevish, 2012)

$$\frac{\varphi}{\varphi \vee_p \psi} \qquad \frac{\psi}{\varphi \vee_p \psi} \qquad \text{(no elimination rules for } \vee_p\text{)}$$

"Cryptograpic" interpretation: $(\varphi_1 \vee_p \varphi_2)$ is a group key.

$(\varphi_1 \vee_p \varphi_2) \to_p \psi$ is a ciphertext that can be decrypted by anyone who has at least one of the keys φ_1 or φ_2:

$$\frac{\varphi_1 \quad (\varphi_1 \vee_p \varphi_2) \to_p \psi}{\psi} \qquad \frac{\varphi_2 \quad (\varphi_1 \vee_p \varphi_2) \to_p \psi}{\psi} \qquad \text{are admissible}$$

$\mathbf{P}[\vee_p]$ *can emulate* $\perp_w$

$\mathbf{P}[\vee_p]$ *can emulate* $\bot_w$

- $q^* = q$ for $q \in At \cup \{\top, \bot\}$,
- $(\varphi \wedge \psi)^* = \varphi^* \wedge \psi^*$,
- $(\varphi \to_p \psi)^* = (\bot \vee_p \varphi^*) \ \to_p \ \psi^*$.

$\mathbf{P}[\vee_p]$ *can emulate* $\perp_w$

- $q^* = q$ for $q \in At \cup \{\top, \perp\}$,
- $(\varphi \wedge \psi)^* = \varphi^* \wedge \psi^*$,
- $(\varphi \to_p \psi)^* = (\perp \vee_p \varphi^*) \ \to_p \ \psi^*$.

The translation $\varphi \mapsto \varphi^*$ is a linear time reduction.

$\mathbf{P}[\vee_p]$ *can emulate* $\perp_w$

- $q^* = q$ for $q \in At \cup \{\top, \perp\}$,
- $(\varphi \wedge \psi)^* = \varphi^* \wedge \psi^*$,
- $(\varphi \to_p \psi)^* = (\perp \vee_p \varphi^*) \ \to_p \ \psi^*$.

The translation $\varphi \mapsto \varphi^*$ is a linear time reduction.

Theorem

$\Gamma \vdash \varphi$ *in* $\mathbf{P}[\perp_w]$ *iff* $\Gamma^* \vdash \varphi^*$ *in* $\mathbf{P}[\vee_p]$.

$\mathbf{P}[\vee_p]$ *can emulate* $\perp_w$

- $q^* = q$ for $q \in At \cup \{\top, \perp\}$,
- $(\varphi \wedge \psi)^* = \varphi^* \wedge \psi^*$,
- $(\varphi \to_p \psi)^* = (\perp \vee_p \varphi^*) \ \to_p \ \psi^*$.

The translation $\varphi \mapsto \varphi^*$ is a linear time reduction.

Theorem

$\Gamma \vdash \varphi$ *in* $\mathbf{P}[\perp_w]$ *iff* $\Gamma^* \vdash \varphi^*$ *in* $\mathbf{P}[\vee_p]$.

It is also possible to reduce $\mathtt{P}[\perp_w]$ to $\mathtt{P}$, but it requires exponential space and time:

1. $\varphi \mapsto \varphi^*$;
2. `replace` $(\perp \vee_p \psi) \to_p \eta$ `with` $(\perp \to_p \eta) \wedge (\psi \to_p \eta)$.