

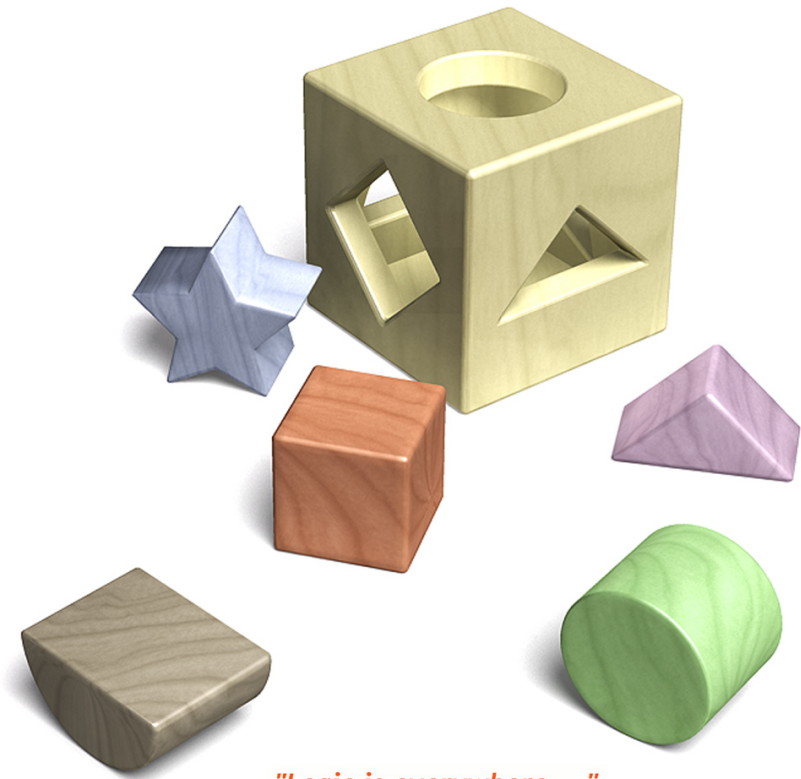


Four-valued Logics for Paraconsistent Reasoning

Andreas Lang

International Center for Computational Logic
Technische Universität Dresden, Germany

- ▶ **Semantic Web Reasoning**
- ▶ **Paraconsistent Reasoning**
- ▶ **Four-valued Approach**



"Logic is everywhere ..."



Semantic Web

- ▶ **Universal medium for information exchange**

- ▶ **Enables machines to**
 - ▷ Interpret information
 - ▷ Perform reasoning tasks

- ▶ **Extension of reasoning capabilities**
 - ▷ Usage of information from diverse sources
 - ▷ Reasoning on the joined information



Different Viewpoints

- ▶ **Viewpoint of a German:**
 - ▷ **“The writer Franz Kafka is German.”**

- ▶ **Viewpoint of a Czech:**
 - ▷ **“The writer Franz Kafka is Czech.”**

- ▶ **Viewpoint of a historian:**
 - ▷ **“Franz Kafka is AustroHungarian.”**
 - ▷ **“Franz Kafka is not German.”**
 - ▷ **“Franz Kafka is not Czech.”**



Contradicting Information

- ▶ **Joining information may cause inconsistencies**
- ▶ **Formalized example**
 - ▶ $\Sigma_1 = \{ \mathit{German}(\mathit{FranzKafka}), \mathit{Writer}(\mathit{FranzKafka}), \mathit{German}(\mathit{ThomasMann}), \mathit{Writer}(\mathit{ThomasMann}) \}$
 - ▶ $\Sigma_2 = \{ \mathit{Czech}(\mathit{FranzKafka}), \mathit{Writer}(\mathit{FranzKafka}), \mathit{Czech}(\mathit{MilanKundera}), \mathit{Writer}(\mathit{MilanKundera}) \}$
 - ▶ $\Sigma_3 = \{ \mathit{Writer}(\mathit{ArthurSchnitzler}), \mathit{AustroHungarian}(\mathit{FranzKafka}), \neg \mathit{German}(\mathit{FranzKafka}), \neg \mathit{Czech}(\mathit{FranzKafka}) \}$
 - ▶ $\Sigma_4 = \Sigma_1 \cup \Sigma_2 \cup \Sigma_3$



Problem of Inconsistencies – the Principle of Explosion

- ▶ Rule of classical logic: *Anything follows from a contradiction*
 - ▶ $\Sigma_4 \models \text{Irish}(\text{JoséCela})$ since Σ_4 has no classical model
- ▶ Σ_4 becomes trivial and we lose all information
- ▶ However, valuable information exist, e.g., *Writer(FranzKafka)*
- ▶ Desire to draw conclusions from consistent information
- ▶ Allowing Σ_4 to have a model avoids explosion



What is Paraconsistency?

▶ A set of formulas Σ is *paraconsistent* iff

▶ it is inconsistent, i.e.,

we find for a formula F that: $\Sigma \models F$

and for the negation of F that: $\Sigma \models \neg F$

▶ but **non-trivial**, i.e.,

Σ has a model



What is a Paraconsistent Logic?

Let $\langle \mathcal{L}, \models \rangle$ be a logic, \mathcal{L} its set of formulas, \models its entailment relation

- ▶ $\langle \mathcal{L}, \models \rangle$ is *paraconsistent* iff
it allows paraconsistent sets of formulas
- ▶ $\langle \mathcal{L}, \models \rangle$ is *fully paraconsistent* iff
all subsets of \mathcal{L} are non-trivial



Our Objective

- ▶ **Join sets of formulas by avoiding trivialization**
 - ▷ A trivial set of formulas leads to explosion
 - ▷ Even paraconsistent logics may explode

- ▶ **Obtain a logic that cannot explode**
 - ▷ A fully paraconsistent logic

- ▶ **Perform reasoning tasks on inconsistent information**
 - ▷ Information is specified in classical logic
 - ▷ Interpret information in a paraconsistent sense



Proceeding in Thesis

- ▶ Development of a paraconsistent first-order logic
- ▶ Definition of a semantic mapping to description logics
- ▶ Elaboration of paraconsistent description logics corresponding to \mathcal{ALC} and \mathcal{SHIQ}
- ▶ Implementation of paraconsistent DL-Reasoning



This presentation covers

- ▶ **Paraconsistent first-order logic**
- ▶ **Reasoning in our paraconsistent logic**
- ▶ **Discussion of the realization of:**
 - ▷ **Subsumption, denoted by \sqsubseteq**
 - ▷ **Equivalence, denoted by \equiv**

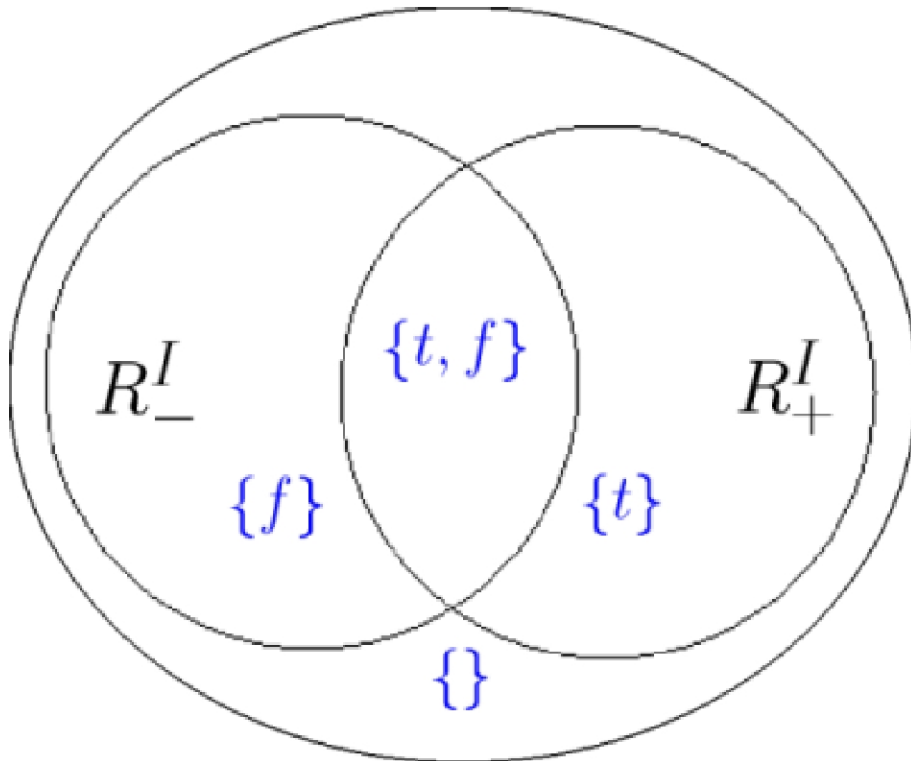


How to obtain Paraconsistency

- ▶ A logic allowing formulas to be *true* and *false* at the same time
- ▶ Intuitively providing additional truth values
- ▶ Multi-valued logics, e.g., four-valued logic (Nuel Belnap)
- ▶ Four-valued logics are a generalization of two-valued logics
- ▶ Formulas of classical logic are formulas of four-valued logic too



Idea of Four-valued Logics



An interpretation assigns to a relation symbol R two sets R_+^I and R_-^I

The truth value of $R(\vec{a})$ depends on the set membership of \vec{a}^I

$$R(\vec{a})^I = \{\} \text{ iff } \vec{a}^I \notin R_+^I \text{ and } \vec{a}^I \notin R_-^I$$

$$R(\vec{a})^I = \{f\} \text{ iff } \vec{a}^I \notin R_+^I \text{ and } \vec{a}^I \in R_-^I$$

$$R(\vec{a})^I = \{t\} \text{ iff } \vec{a}^I \in R_+^I \text{ and } \vec{a}^I \notin R_-^I$$

$$R(\vec{a})^I = \{t, f\} \text{ iff } \vec{a}^I \in R_+^I \\ \text{and } \vec{a}^I \in R_-^I$$



Logical Consequence in Four-valued Logics (I)

▶ **Definition of entailment relations \models_t and \models_f**

- ▶ $I, \sigma \models_t R(\vec{x})$ **iff** $\vec{x}^{I, \sigma} \in R_+^I$
- ▶ $I, \sigma \models_f R(\vec{x})$ **iff** $\vec{x}^{I, \sigma} \in R_-^I$
- ▶ $I, \sigma \models_t \neg F$ **iff** $I, \sigma \models_f F$
- ▶ $I, \sigma \models_f \neg F$ **iff** $I, \sigma \models_t F$
- ▶ $I, \sigma \models_t F \wedge G$ **iff** $I, \sigma \models_t F$ **and** $I, \sigma \models_t G$
- ▶ $I, \sigma \models_f F \wedge G$ **iff** $I, \sigma \models_f F$ **or** $I, \sigma \models_f G$
- ▶ ...



Logical Consequence in Four-valued Logics (II)

- ▶ ***Satisfiable*** is a formula F iff $I, \sigma \models_t F$ holds for some I, σ
 - ▷ $\{t\}$ and $\{t, f\}$ are both ***designated*** truth values

- ▶ I is a ***model*** of F iff $I, \sigma \models_t F$ holds for all σ



Logical Consequence in Four-valued Logics (III)

- ▶ A set of formulas Γ *entails* a formula F iff all interpretations that are models of Γ are models of F
 - ▶ We write $\Gamma \models_t F$
- ▶ Formulas F and G are *semantically equivalent* iff $F^{I,\sigma} = G^{I,\sigma}$
 - ▶ We write $F \equiv G$



Benefit of Four-valued Logics

- ▶ A formula can be both *true* and *false* at the same time
 - ▷ Inconsistencies do not trivialize a set of formulas
 $F^{I,\sigma} = \{t, f\}$ and $G^{I,\sigma} = \{f\}$ but $F \wedge \neg F \not\vdash_t G$
- ▶ Full paraconsistency w.r.t. conjunction, disjunction and negation
- ▶ Intuitive calculation with conjunction, disjunction and negation
- ▶ Use of formulas of classical first-order logic



Difficulties in Four-valued Logics

- ▶ **Lack of four-valued reasoning systems**
- ▶ **Use of a truth or a false formula**
- ▶ **Definition of an implication connective**
- ▶ **Definition of an equivalence connective**
- ▶ **Definition of an equality relation**



Difficulties in Four-valued Logics

- ▶ **Lack of four-valued reasoning systems**
- ▶ Use of a truth or a false formula
- ▶ Definition of an implication connective
- ▶ Definition of an equivalence connective
- ▶ Definition of an equality relation



Lack of four-valued reasoning systems

- ▶ We transform formulas F into negation normal form F'
- ▶ Function λ maps formula F' of four-valued first-order logic to a formula of a negation-free classical first-order logic

$$\lambda(F') \stackrel{\text{def}}{=} \begin{cases} R_+(\vec{x}) & \text{for positive literals } R(\vec{x}), \\ R_-(\vec{x}) & \text{for negative literals } \neg R(\vec{x}), \\ \lambda(G) \circ \lambda(H) & \text{for two formulas } G \text{ and } H \\ & \text{with } \circ \in \{\wedge, \vee\}, \\ (Qx)\lambda(F) & \text{for a formula } F \text{ with } Q \in \{\forall, \exists\}. \end{cases}$$

- ▶ Application of reasoning systems for classical logic on $\lambda(F')$



Example Mapping to negation-free Classical Logic

- ▶ Translation of Σ_4 in negation normal form
 - ▷ $\Sigma'_4 = \{ \mathbf{German}(\mathbf{FranzKafka}), \mathbf{Writer}(\mathbf{FranzKafka}), \neg \mathbf{German}(\mathbf{FranzKafka}), \dots \}$

- ▶ Mapping of Σ'_4 to negation free classical first-order logic
 - ▷ $\lambda(\Sigma'_4) = \{ \mathbf{German}_+(\mathbf{FranzKafka}), \mathbf{Writer}_+(\mathbf{FranzKafka}), \mathbf{German}_-(\mathbf{FranzKafka}), \dots \}$



Reasoning with standard reasoning systems

- ▶ $\lambda(\Sigma'_4)$ does not explode and
 - ▶ $\{ \mathbf{German}_+(FranzKafka), \mathbf{Writer}_+(FranzKafka), \mathbf{German}_-(FranzKafka), \dots \}$
- ▶ Hence, $\Sigma_4 \models_t \mathbf{Writer}(FranzKafka)$
- ▶ and $\Sigma_4 \not\models_t \mathbf{Irish}(JoséCela)$



Difficulties in Four-valued Logics

- ▷ Lack of four-valued reasoning systems
- ▶ **Use of a truth or a false formula**
- ▷ Definition of an implication connective
- ▷ Definition of an equivalence connective
- ▷ Definition of an equality relation



The truth and the false formula \top and \perp

- ▶ Description logics require a *top* and *bottom* concept
- ▶ Correspondence to top and false formula in first-order logic
- ▶ Intended semantics of \top is to be always true and of \perp to be always false
- ▶ Allowing \top or \perp makes the logic trivializable:

$$\neg \top \equiv \perp \quad \rightsquigarrow \quad \perp \vee F \equiv F$$



The truth and the false formula \top and \perp

- ▶ Full paraconsistency cannot be obtained
 - ▷ A set of formulas containing \perp is trivial
- ▶ Joining non-trivial sets of formulas yields a non-trivial set
- ▶ Easy to check for occurrences of \top and \perp
- ▶ Careful use of \top and \perp may be allowed



Difficulties in Four-valued Logics

- ▷ Lack of four-valued reasoning systems
- ▷ Use of a truth or a false formula
- ▶ **Definition of an implication connective**
- ▷ Definition of an equivalence connective
- ▷ Definition of an equality relation



Requirement for Implication Connective

- ▶ **Concept subsumption is basic scheme in DLs**
 - ▶ $(D \sqsubseteq C)$ corresponds to $(\forall x)(D(x) \rightarrow C(x))$

- ▶ **Giving semantics of DL by first-order logic relies on implication**
 - ▶ $(\forall R.C)$ corresponds to $(\forall y)(R(x, y) \rightarrow C(y))$
 - ▶ $(\leq 3R.C)$ corresponds to

$$(\forall y_1, y_2, y_3, y_4)(\bigwedge_{i=1}^4 (R(x, y_i) \wedge C(y_i)) \rightarrow \bigvee_{i,j} y_i \approx y_j)$$



Definition of an Implication Connective (I)

- ▶ **Requirements for an implication connective**
 1. **Full paraconsistency has to be preserved**
 - ▶ to avoid that our logic becomes trivializable
 2. **The deduction theorem has to be satisfied**
 - ▶ to express consequence relations as formulas
 3. **It should be expressible by \wedge, \vee, \neg**
 - ▶ to map formulas to classical logic



Definition of an Implication Connective (II)

- ▶ Item 2 and item 3 are mutually exclusive
- ▶ We drop the third requirement
- ▶ Fortunately, transformation in classical logic is possible
 - ▶ $R(x) \rightarrow S(y)$ is mapped to $\neg R_+(x) \vee S_+(y)$



Difficulties in Four-valued Logics

- ▷ Lack of four-valued reasoning systems
- ▷ Use of a truth or a false formula
- ▷ Definition of an implication connective
- ▶ **Definition of an equivalence connective**
- ▷ Definition of an equality relation



Definition of an Equivalence Connective (I)

- ▶ **Common definition:** $F \leftrightarrow G \stackrel{def}{=} (F \rightarrow G) \wedge (G \rightarrow F)$
- ▶ **Corresponding truth table**

\leftrightarrow	$\{\}$	$\{f\}$	$\{t\}$	$\{t, f\}$
$\{\}$	$\{t\}$	$\{t\}$	$\{\}$	$\{\}$
$\{f\}$	$\{t\}$	$\{t, f\}$	$\{\}$	$\{f\}$
$\{t\}$	$\{\}$	$\{\}$	$\{t\}$	$\{t\}$
$\{t, f\}$	$\{\}$	$\{f\}$	$\{t\}$	$\{t, f\}$

- ▶ **Equivalence not properly described by \leftrightarrow**



Definition of an Equivalence Connective (I)

- ▶ **Common definition:** $F \leftrightarrow G \stackrel{def}{=} (F \rightarrow G) \wedge (G \rightarrow F)$
- ▶ **Corresponding truth table**

\leftrightarrow	$\{\}$	$\{f\}$	$\{t\}$	$\{t, f\}$
$\{\}$	$\{t\}$	$\{t\}$	$\{\}$	$\{\}$
$\{f\}$	$\{t\}$	$\{t, f\}$	$\{\}$	$\{f\}$
$\{t\}$	$\{\}$	$\{\}$	$\{t\}$	$\{t\}$
$\{t, f\}$	$\{\}$	$\{f\}$	$\{t\}$	$\{t, f\}$

- ▶ **Equivalence not properly described by \leftrightarrow**



Definition of an Equivalence Connective (II)

- ▶ Definition of \Leftrightarrow satisfies intuitive equivalence semantics
- ▶ Corresponding truth table

\Leftrightarrow	$\{\}$	$\{f\}$	$\{t\}$	$\{t, f\}$
$\{\}$	$\{t\}$	$\{f\}$	$\{f\}$	$\{f\}$
$\{f\}$	$\{f\}$	$\{t\}$	$\{f\}$	$\{f\}$
$\{t\}$	$\{f\}$	$\{f\}$	$\{t\}$	$\{f\}$
$\{t, f\}$	$\{f\}$	$\{f\}$	$\{f\}$	$\{t, f\}$

- ▶ Problematic in DL since equivalence often described by subsumption



Subsumption and Equivalence in \mathcal{ALC}_4

▶ Let A and B be two concepts

▶ $A \sqsubseteq B$ if and only if

▷ $A_+ \subseteq B_+$

▶ $A \equiv B$ if and only if

▷ $(A_+ = B_+)$ and $(A_- = B_-)$

▷ $(A_+ \subseteq B_+)$ and $(B_+ \subseteq A_+)$ and $(A_- \subseteq B_-)$ and $(B_- \subseteq A_-)$



Difficulties in Four-valued Logics

- ▷ Lack of four-valued reasoning systems
- ▷ Use of a truth or a false formula
- ▷ Definition of an implication connective
- ▷ Definition of an equivalence connective
- ▶ **Definition of an equality relation**



The Equality Relation \approx

- ▶ Requirement for equality in description logic \mathcal{SHIQ}
 - ▶ E.g. number restriction relies on equality
- ▶ Intended semantics of $a \approx b$ is: $a^{I,\sigma} = b^{I,\sigma}$
- ▶ Such a definition makes the logic trivializable because $\neg a \approx a \equiv \perp$



The Equality Relation \approx

- ▶ **Joining non-trivial sets may result in a trivial set**

- ▶ $\Sigma_1 = \{(\leq 3R.C)\}$

- ▶ $\Sigma_2 = \{(\geq 5R.C)\}$

- ▶ $\Sigma_3 = \Sigma_1 \cup \Sigma_2$

- ▶ **Difficult to check a set of formulas for expressions that may trivialize the set**



Proposal for Redefinition of Equality – Open for Discussion

- ▶ **Alternative definition for semantics of \approx**
 - ▷ $I, \sigma \models_t x \approx y$ **iff** $x^{I, \sigma} = y^{I, \sigma}$
 - ▷ $I, \sigma \models_f x \approx y$ **for all x and y**

- ▶ **$\{a \approx b, \neg a \approx b\}$ is not a trivial set**
 - ▷ $\{a \approx b, \neg a \approx b\} \models_t \neg a \approx a$ **and**
 - ▷ $\{a \approx b, \neg a \approx b\} \models_t a \approx a$

- ▶ **Negation of $a \approx b$ cannot be expressed by \neg**
but as *failed to prove* $a \approx b$



Summary and Outlook

- ▶ **Paraconsistent first-order logic successfully designed**
- ▶ **Paraconsistent \mathcal{ALC} successfully designed**
- ▶ **Paraconsistent DL reasoning for KAON2 implemented**
- ▶ **Paraconsistent \mathcal{SHIQ} in discussion**
 - ▷ **Equality is required but problematic**



Proposals for Discussion

- ▶ **Definition and use of the equality relation \approx**
- ▶ **Equivalence connective \equiv in description logics**
- ▶ **Mapping to classical logic**
- ▶ **Other proposals for application to DLs**



References

- ▶ Ofer Arieli and Arnon Avron,
 - ▷ *The Value of the Four Values*

- ▶ Peter F. Patel-Schneider,
 - ▷ *A Four-Valued Semantics for Terminological Logics*

- ▶ Umberto Straccia,
 - ▷ *A Sequent Calculus for Reasoning in Four-valued Description Logics*



Definition of Ontology

An ontology defines the terms used to describe and represent an area of knowledge.

Ontologies are used to share domain information.

Ontologies include computer-usable definitions of basic concepts in the domain and the relationships among them.

We abstract the term ontology as a set of formulas of a logic.



Model and Interpretation

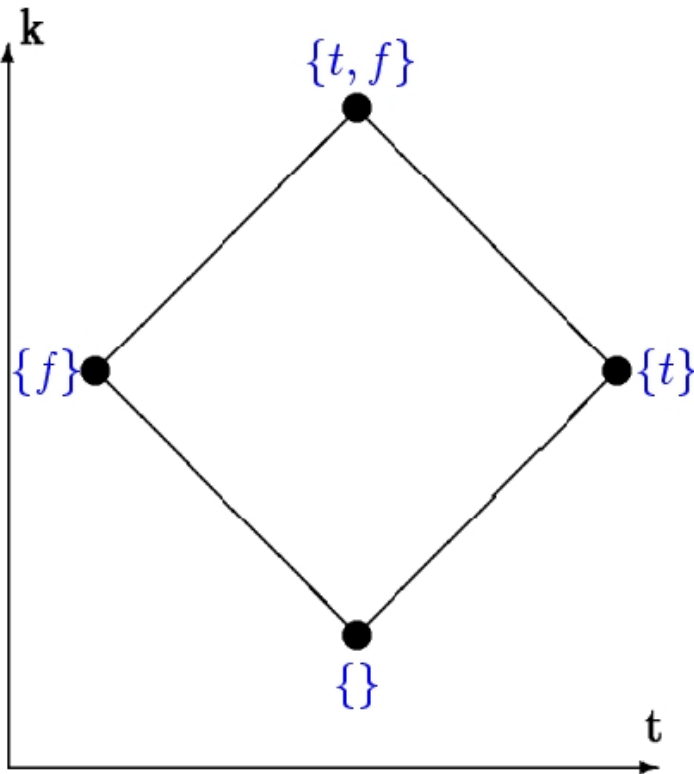
Let $\langle \mathcal{L}, \models \rangle$ be a logic, \mathcal{L} its set of formulas, \models its entailment relation and let Γ be a set of formulas

A *model* of Γ is an interpretation that evaluates all $F \in \Gamma$ to be true

An *interpretation* I consists of a non-empty domain \mathcal{D} and a mapping I that assigns truth values to formulas of \mathcal{L}



Representation by a Logical Bilattice



meet-operation on truth-lattice
corresponds to *conjunction*

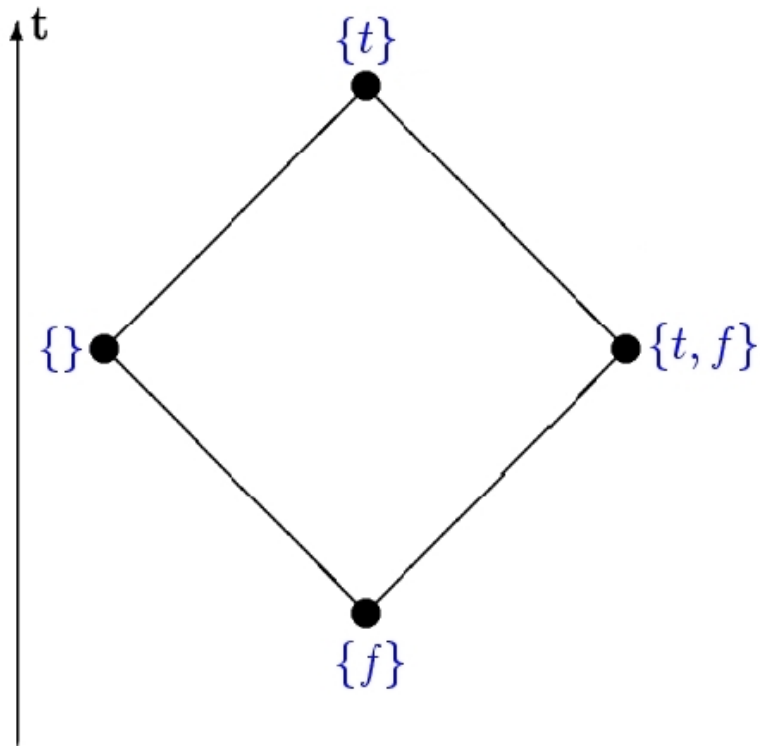
join-operation on truth-lattice
corresponds to *disjunction*

Negation is defined as:

α	$\{\}$	$\{f\}$	$\{t\}$	$\{t, f\}$
$\neg\alpha$	$\{\}$	$\{t\}$	$\{f\}$	$\{t, f\}$



The Truth-Lattice



meet-operation on truth-lattice
corresponds to *conjunction*

join-operation on truth-lattice
corresponds to *disjunction*

Negation is defined as:

α	$\{\}$	$\{f\}$	$\{t\}$	$\{t, f\}$
$\neg\alpha$	$\{\}$	$\{t\}$	$\{f\}$	$\{t, f\}$



Representation of a simple Equivalence

The four-valued first-order logic formula

$$F_4 = R(x) \Leftrightarrow S(x)$$

Representation in classical first-order logic by

$$\begin{aligned} F_2 = & ((\neg R_-(x) \wedge \neg S_+(x)) \vee R_+(x) \vee S_-(x)) \\ & \wedge ((\neg R_+(x) \wedge \neg S_-(x)) \vee R_-(x) \vee S_+(x)) \\ & \wedge (R_+(x) \leftrightarrow S_+(x)) \wedge (R_-(x) \leftrightarrow S_-(x)) \end{aligned}$$



Three-Valued Implication \rightarrow_3

- ▶ Definition of \rightarrow_3 with the properties
 - ▷ Full paraconsistency
 - ▷ Deduction theorem holds
 - ▷ Possible negation normal form transformation

\rightarrow_3	$\{f\}$	$\{t\}$	$\{t, f\}$
$\{f\}$	$\{t, f\}$	$\{t\}$	$\{t, f\}$
$\{t\}$	$\{f\}$	$\{t\}$	$\{t, f\}$
$\{t, f\}$	$\{f\}$	$\{t\}$	$\{t, f\}$



Negation Normal Form Representation of \rightarrow_3

The three-valued first-order logic formula

$$R(x) \rightarrow_3 S(x)$$

Additional involution connective \sim_3

α	$\{f\}$	$\{t\}$	$\{t, f\}$
$\sim_3 \alpha$	$\{t, f\}$	$\{t\}$	$\{f\}$

Transformation in negation normal form

$$R(x) \rightarrow_3 S(x) \rightsquigarrow R(x) \vee \neg S(x) \wedge \sim_3 \neg S(x)$$



The Deduction Theorem

Let \models be an entailment relation, let \rightarrow be an implication connective,

Let Γ be a set of formulas, and let F and G be formulas

Deduction Theorem states relationship between \models and \rightarrow

$$\Gamma \cup \{F\} \models G \quad \text{iff} \quad \Gamma \models F \rightarrow G$$



Truth Table Connectives Satisfying Deduction Theorem

\oplus	$\{\}$	$\{f\}$	$\{t\}$	$\{t, f\}$
$\{\}$	$\{t\} / \{t, f\}$	$\{t\} / \{t, f\}$	$\{t\} / \{t, f\}$	$\{t\} / \{t, f\}$
$\{f\}$	$\{t\} / \{t, f\}$	$\{t\} / \{t, f\}$	$\{t\} / \{t, f\}$	$\{t\} / \{t, f\}$
$\{t\}$	$\{\} / \{f\}$	$\{\} / \{f\}$	$\{t\} / \{t, f\}$	$\{t\} / \{t, f\}$
$\{t, f\}$	$\{\} / \{f\}$	$\{\} / \{f\}$	$\{t\} / \{t, f\}$	$\{t\} / \{t, f\}$



Truth Table Connectives Satisfying Deduction Theorem

\oplus	$\{\}$	$\{f\}$	$\{t\}$	$\{t, f\}$
$\{\}$	$\{t\} / \{t, f\}$	$\{t\} / \{t, f\}$	$\{t\} / \{t, f\}$	$\{t\} / \{t, f\}$
$\{f\}$	$\{t\} / \{t, f\}$	$\{t\} / \{t, f\}$	$\{t\} / \{t, f\}$	$\{t\} / \{t, f\}$
$\{t\}$	$\{\} / \{f\}$	$\{\} / \{f\}$	$\{t\} / \{t, f\}$	$\{t\} / \{t, f\}$
$\{t, f\}$	$\{\} / \{f\}$	$\{\} / \{f\}$	$\{t\} / \{t, f\}$	$\{t\} / \{t, f\}$

\oplus is a non-monotonic connective



Truth Table Connectives Satisfying Deduction Theorem

\oplus	$\{\}$	$\{f\}$	$\{t\}$	$\{t, f\}$
$\{\}$	$\{t\} / \{t, f\}$	$\{t\} / \{t, f\}$	$\{t\} / \{t, f\}$	$\{t\} / \{t, f\}$
$\{f\}$	$\{t\} / \{t, f\}$	$\{t\} / \{t, f\}$	$\{t\} / \{t, f\}$	$\{t\} / \{t, f\}$
$\{t\}$	$\{\} / \{f\}$	$\{\} / \{f\}$	$\{t\} / \{t, f\}$	$\{t\} / \{t, f\}$
$\{t, f\}$	$\{\} / \{f\}$	$\{\} / \{f\}$	$\{t\} / \{t, f\}$	$\{t\} / \{t, f\}$

\rightarrow	$\{\}$	$\{f\}$	$\{t\}$	$\{t, f\}$
$\{\}$	$\{t\}$	$\{t\}$	$\{t\}$	$\{t\}$
$\{f\}$	$\{t\}$	$\{t, f\}$	$\{t\}$	$\{t, f\}$
$\{t\}$	$\{\}$	$\{\}$	$\{t\}$	$\{t\}$
$\{t, f\}$	$\{\}$	$\{f\}$	$\{t\}$	$\{t, f\}$



Definition of Monotonicity

Let $\langle \mathcal{D}, \leq \rangle$ be a partially ordered set.

An operation $f : \mathcal{D}^n \rightarrow \mathcal{D}$ is called *monotonic*

(with respect to \leq) if $f(x) \leq f(y)$ whenever $x \leq y$

(we say that $x = (x_1, \dots, x_n) \leq y = (y_1, \dots, y_n)$

iff we find for all $1 \leq i \leq n$ that $x_i \leq y_i$).

We define monotonicity w.r.t. to knowledge lattice \leq_k