Rewrite Systems on Logics: Calculi

<table>
<thead>
<tr>
<th></th>
<th>Validity</th>
<th>Satisfiability</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Sound</strong></td>
<td>If the calculus derives a proof of validity for the formula, it is valid.</td>
<td>If the calculus derives satisfiability of the formula, it has a model.</td>
</tr>
<tr>
<td><strong>Complete</strong></td>
<td>If the formula is valid, a proof of validity is derivable by the calculus.</td>
<td>If the formula has a model, the calculus derives satisfiability.</td>
</tr>
<tr>
<td><strong>Strongly Complete</strong></td>
<td>For any validity proof of the formula, there is a derivation in the calculus producing this proof.</td>
<td>For any model of the formula, there is a derivation in the calculus producing this model.</td>
</tr>
</tbody>
</table>
Propositional Logic: Syntax

2.1.1 Definition (Propositional Formula)

The set $PROP(\Sigma)$ of *propositional formulas* over a signature $\Sigma$, is inductively defined by:

<table>
<thead>
<tr>
<th>$PROP(\Sigma)$</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\bot$</td>
<td>connective $\bot$ denotes “false”</td>
</tr>
<tr>
<td>$\top$</td>
<td>connective $\top$ denotes “true”</td>
</tr>
<tr>
<td>$P$</td>
<td>for any propositional variable $P \in \Sigma$</td>
</tr>
<tr>
<td>$(\neg \phi)$</td>
<td>connective $\neg$ denotes “negation”</td>
</tr>
<tr>
<td>$(\phi \land \psi)$</td>
<td>connective $\land$ denotes “conjunction”</td>
</tr>
<tr>
<td>$(\phi \lor \psi)$</td>
<td>connective $\lor$ denotes “disjunction”</td>
</tr>
<tr>
<td>$(\phi \rightarrow \psi)$</td>
<td>connective $\rightarrow$ denotes “implication”</td>
</tr>
<tr>
<td>$(\phi \leftrightarrow \psi)$</td>
<td>connective $\leftrightarrow$ denotes “equivalence”</td>
</tr>
</tbody>
</table>

where $\phi, \psi \in PROP(\Sigma)$.
Propositional Logic: Semantics

2.2.1 Definition ((Partial) Valuation)

A $\Sigma$-valuation is a map

$$\mathcal{A} : \Sigma \rightarrow \{0, 1\}.$$ 

where $\{0, 1\}$ is the set of truth values. A partial $\Sigma$-valuation is a map $\mathcal{A'} : \Sigma' \rightarrow \{0, 1\}$ where $\Sigma' \subseteq \Sigma$. 
2.2.2 Definition (Semantics)

A $\Sigma$-valuation $\mathcal{A}$ is inductively extended from propositional variables to propositional formulas $\phi, \psi \in \text{PROP}(\Sigma)$ by

- $\mathcal{A}(\bot) := 0$
- $\mathcal{A}(\top) := 1$
- $\mathcal{A}(\neg \phi) := 1 - \mathcal{A}(\phi)$
- $\mathcal{A}(\phi \land \psi) := \min(\{\mathcal{A}(\phi), \mathcal{A}(\psi)\})$
- $\mathcal{A}(\phi \lor \psi) := \max(\{\mathcal{A}(\phi), \mathcal{A}(\psi)\})$
- $\mathcal{A}(\phi \rightarrow \psi) := \max(\{1 - \mathcal{A}(\phi), \mathcal{A}(\psi)\})$
- $\mathcal{A}(\phi \leftrightarrow \psi) := \text{if } \mathcal{A}(\phi) = \mathcal{A}(\psi) \text{ then } 1 \text{ else } 0$
If $A(\phi) = 1$ for some $\Sigma$-valuation $A$ of a formula $\phi$ then $\phi$ is **satisfiable** and we write $A \models \phi$. In this case $A$ is a **model** of $\phi$.

If $A(\phi) = 1$ for all $\Sigma$-valuations $A$ of a formula $\phi$ then $\phi$ is **valid** and we write $\models \phi$.

If there is no $\Sigma$-valuation $A$ for a formula $\phi$ where $A(\phi) = 1$ we say $\phi$ is **unsatisfiable**.

A formula $\phi$ **entails** $\psi$, written $\phi \models \psi$, if for all $\Sigma$-valuations $A$ whenever $A \models \phi$ then $A \models \psi$. 
Propositional Logic: Operations

2.1.2 Definition (Atom, Literal, Clause)

A propositional variable $P$ is called an *atom*. It is also called a *(positive) literal* and its negation $\neg P$ is called a *(negative) literal*.

The functions $\text{comp}$ and $\text{atom}$ map a literal to its complement, or atom, respectively: if $\text{comp}(\neg P) = P$ and $\text{comp}(P) = \neg P$, $\text{atom}(\neg P) = P$ and $\text{atom}(P) = P$ for all $P \in \Sigma$. Literals are denoted by letters $L, K$. Two literals $P$ and $\neg P$ are called *complementary*.

A disjunction of literals $L_1 \vee \ldots \vee L_n$ is called a *clause*. A clause is identified with the multiset of its literals.
2.1.3 Definition (Position)

A *position* is a word over \( \mathbb{N} \). The set of positions of a formula \( \phi \) is inductively defined by

\[
\begin{align*}
pos(\phi) & := \{\epsilon\} \text{ if } \phi \in \{\top, \bot\} \text{ or } \phi \in \Sigma \\
pos(\neg \phi) & := \{\epsilon\} \cup \{1p \mid p \in \pos(\phi)\} \\
pos(\phi \circ \psi) & := \{\epsilon\} \cup \{1p \mid p \in \pos(\phi)\} \cup \{2p \mid p \in \pos(\psi)\}
\end{align*}
\]

where \( \circ \in \{\land, \lor, \rightarrow, \leftrightarrow\} \).
The prefix order $\leq$ on positions is defined by $p \leq q$ if there is some $p'$ such that $pp' = q$. Note that the prefix order is partial, e.g., the positions 12 and 21 are not comparable, they are “parallel”, see below.

The relation $<$ is the strict part of $\leq$, i.e., $p < q$ if $p \leq q$ but not $q \leq p$.

The relation $\parallel$ denotes incomparable, also called parallel positions, i.e., $p \parallel q$ if neither $p \leq q$, nor $q \leq p$.

A position $p$ is above $q$ if $p \leq q$, $p$ is strictly above $q$ if $p < q$, and $p$ and $q$ are parallel if $p \parallel q$. 
The size of a formula $\phi$ is given by the cardinality of $\text{pos}(\phi)$:

$$|\phi| := |\text{pos}(\phi)|.$$ 

The subformula of $\phi$ at position $p \in \text{pos}(\phi)$ is inductively defined by $\phi|_\epsilon := \phi$, $\neg \phi|_p := \phi|_p$, and $(\phi_1 \circ \phi_2)|_i p := \phi_i|_p$ where $i \in \{1, 2\}$, $\circ \in \{\land, \lor, \rightarrow, \leftrightarrow\}$.

Finally, the replacement of a subformula at position $p \in \text{pos}(\phi)$ by a formula $\psi$ is inductively defined by $\phi[\psi]_\epsilon := \psi$, $(\neg \phi)[\psi]|_p := \neg \phi[\psi]|_p$, and $(\phi_1 \circ \phi_2)[\psi]|_i p := (\phi_1[\psi]|_p \circ \phi_2)$, $(\phi_1 \circ \phi_2)[\psi]|_2 p := (\phi_1 \circ \phi_2[\psi]|_p)$, where $\circ \in \{\land, \lor, \rightarrow, \leftrightarrow\}$. 
2.1.5 Definition (Polarity)

The polarity of the subformula $\phi|_p$ of $\phi$ at position $p \in \text{pos}(\phi)$ is inductively defined by

\[
\begin{align*}
\text{pol}(\phi, \epsilon) & := 1 \\
\text{pol}(\neg \phi, 1p) & := -\text{pol}(\phi, p) \\
\text{pol}(\phi_1 \circ \phi_2, ip) & := \text{pol}(\phi_i, p) & \text{if } \circ \in \{\land, \lor\}, i \in \{1, 2\} \\
\text{pol}(\phi_1 \rightarrow \phi_2, 1p) & := -\text{pol}(\phi_1, p) \\
\text{pol}(\phi_1 \rightarrow \phi_2, 2p) & := \text{pol}(\phi_2, p) \\
\text{pol}(\phi_1 \leftrightarrow \phi_2, ip) & := 0 & \text{if } i \in \{1, 2\}
\end{align*}
\]
Valuations can be nicely represented by sets or sequences of literals that do not contain complementary literals nor duplicates.

If $\mathcal{A}$ is a (partial) valuation of domain $\Sigma$ then it can be represented by the set
$$\{ P \mid P \in \Sigma \text{ and } \mathcal{A}(P) = 1 \} \cup \{ \neg P \mid P \in \Sigma \text{ and } \mathcal{A}(P) = 0 \}.$$  

Another, equivalent representation are *Herbrand* interpretations that are sets of positive literals, where all atoms not contained in an Herbrand interpretation are false. If $\mathcal{A}$ is a total valuation of domain $\Sigma$ then it corresponds to the Herbrand interpretation
$$\{ P \mid P \in \Sigma \text{ and } \mathcal{A}(P) = 1 \}.$$
2.2.4 Theorem (Deduction Theorem)

\( \phi \models \psi \text{ iff } \models \phi \rightarrow \psi \)
2.2.6 Lemma (Formula Replacement)

Let $\phi$ be a propositional formula containing a subformula $\psi$ at position $p$, i.e., $\phi|_p = \psi$. Furthermore, assume $\models \psi \leftrightarrow \chi$. Then $\models \phi \leftrightarrow \phi[\chi]_p$. 
2.4.1 Definition (α-, β-Formulas)

A formula \( \phi \) is called an \( \alpha \)-formula if \( \phi \) is a formula \( \neg \neg \phi_1, \phi_1 \land \phi_2, \phi_1 \leftrightarrow \phi_2, \neg(\phi_1 \lor \phi_2), \) or \( \neg(\phi_1 \rightarrow \phi_2). \)

A formula \( \phi \) is called a \( \beta \)-formula if \( \phi \) is a formula \( \phi_1 \lor \phi_2, \phi_1 \rightarrow \phi_2, \neg(\phi_1 \land \phi_2), \) or \( \neg(\phi_1 \leftrightarrow \phi_2). \)
Given an \( \alpha \)- or \( \beta \)-formula \( \phi \), its direct descendants are as follows:

<table>
<thead>
<tr>
<th>( \alpha )</th>
<th>Left Descendant</th>
<th>Right Descendant</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \neg \neg \phi )</td>
<td>( \phi )</td>
<td>( \phi )</td>
</tr>
<tr>
<td>( \phi_1 \land \phi_2 )</td>
<td>( \phi_1 )</td>
<td>( \phi_2 )</td>
</tr>
<tr>
<td>( \phi_1 \leftrightarrow \phi_2 )</td>
<td>( \phi_1 \rightarrow \phi_2 )</td>
<td>( \phi_2 \rightarrow \phi_1 )</td>
</tr>
<tr>
<td>( \neg (\phi_1 \lor \phi_2) )</td>
<td>( \neg \phi_1 )</td>
<td>( \neg \phi_2 )</td>
</tr>
<tr>
<td>( \neg (\phi_1 \rightarrow \phi_2) )</td>
<td>( \phi_1 )</td>
<td>( \neg \phi_2 )</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>( \beta )</th>
<th>Left Descendant</th>
<th>Right Descendant</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \phi_1 \lor \phi_2 )</td>
<td>( \phi_1 )</td>
<td>( \phi_2 )</td>
</tr>
<tr>
<td>( \phi_1 \rightarrow \phi_2 )</td>
<td>( \neg \phi_1 )</td>
<td>( \phi_2 )</td>
</tr>
<tr>
<td>( \neg (\phi_1 \land \phi_2) )</td>
<td>( \neg \phi_1 )</td>
<td>( \neg \phi_2 )</td>
</tr>
<tr>
<td>( \neg (\phi_1 \leftrightarrow \phi_2) )</td>
<td>( \neg (\phi_1 \rightarrow \phi_2) )</td>
<td>( \neg (\phi_2 \rightarrow \phi_1) )</td>
</tr>
</tbody>
</table>
2.4.3 Proposition ()

For any valuation $A$:

(i) if $\phi$ is an $\alpha$-formula then $A(\phi) = 1$ iff $A(\phi_1) = 1$ and $A(\phi_2) = 1$ for its descendants $\phi_1, \phi_2$.

(ii) if $\phi$ is a $\beta$-formula then $A(\phi) = 1$ iff $A(\phi_1) = 1$ or $A(\phi_2) = 1$ for its descendants $\phi_1, \phi_2$. 
Tableau Rewrite System

The tableau calculus operates on states that are sets of sequences of formulas. Semantically, the set represents a disjunction of sequences that are interpreted as conjunctions of the respective formulas.

A sequence of formulas \((\phi_1, \ldots, \phi_n)\) is called closed if there are two formulas \(\phi_i\) and \(\phi_j\) in the sequence where \(\phi_i = \text{comp}(\phi_j)\).

A state is closed if all its formula sequences are closed.

The tableau calculus is a calculus showing unsatisfiability of a formula. Such calculi are called refutational calculi. Recall a formula \(\phi\) is valid iff \(\neg\phi\) is unsatisfiable.
A formula $\phi$ occurring in some sequence is called *open* if in case $\phi$ is an $\alpha$-formula not both direct descendants are already part of the sequence and if it is a $\beta$-formula none of its descendants is part of the sequence.
Tableau Rewrite Rules

**α-Expansion**

\[ N \uplus \{ (\phi_1, \ldots, \psi, \ldots, \phi_n) \} \Rightarrow_T \]

\[ N \uplus \{ (\phi_1, \ldots, \psi, \ldots, \phi_n, \psi_1, \psi_2) \} \]

provided \( \psi \) is an open \( \alpha \)-formula, \( \psi_1, \psi_2 \) its direct descendants and the sequence is not closed.

**β-Expansion**

\[ N \uplus \{ (\phi_1, \ldots, \psi, \ldots, \phi_n) \} \Rightarrow_T \]

\[ N \uplus \{ (\phi_1, \ldots, \psi, \ldots, \phi_n, \psi_1) \} \uplus \{ (\phi_1, \ldots, \psi, \ldots, \phi_n, \psi_2) \} \]

provided \( \psi \) is an open \( \beta \)-formula, \( \psi_1, \psi_2 \) its direct descendants and the sequence is not closed.