Truth. Does not change.

$$\frac{}{\cdot \longrightarrow \top}\,\top R$$

Implication. The possibility of empty right-hand sides requires a third right rule for implication. Again, in an implementation the three rules might be combined into a more efficient one.

$$\frac{\Gamma, A \longrightarrow B}{\Gamma \longrightarrow A \supset B}\,{\supset}R_1 \qquad \frac{\Gamma \longrightarrow B}{\Gamma \longrightarrow A \supset B}\,{\supset}R_2 \qquad \frac{\Gamma, A \longrightarrow \cdot}{\Gamma \longrightarrow A \supset B}\,{\supset}R_3$$

$$\frac{\Gamma_1 \longrightarrow A \qquad \Gamma_2, B \longrightarrow \gamma}{\Gamma_1 \cup \Gamma_2, A \supset B \longrightarrow \gamma}\,{\supset}L$$

Disjunction. The rule for disjunction on the right remains the same, but the left rule now has to account for several possibilities, depending on whether the right-hand sides of the premises are empty. Essentially, we take the union of the right-hand sides of the two premises, except that the result must be a singleton or empty for the sequent to be well-formed.

$$\frac{\Gamma \longrightarrow A}{\Gamma \longrightarrow A \vee B}\,{\vee}R_1 \qquad \frac{\Gamma \longrightarrow B}{\Gamma \longrightarrow A \vee B}\,{\vee}R_2$$

$$\frac{\Gamma_1, A \longrightarrow \gamma_1 \qquad \Gamma_2, B \longrightarrow \gamma_2}{\Gamma_1 \cup \Gamma_2, A \vee B \longrightarrow \gamma_1 \cup \gamma_2}\,{\vee}L$$

In detail, either γ_1 or γ_2 is empty, or $\gamma_1 = \gamma_2 = C = \gamma_1 \cup \gamma_2$. The rule does not apply otherwise.

The statement of the soundness theorem does not change much with empty succedents.

Theorem 5.3 (Soundness of Forward Sequent Calculus)

1. *If* $\Gamma \longrightarrow C$ *then* $\Gamma \Longrightarrow C$*, and*
2. *if* $\Gamma \longrightarrow \cdot$ *then* $\Gamma \Longrightarrow C$ *for all* C*.*

Proof: By induction on the derivation $\mathcal{F}$ of $\Gamma \longrightarrow \gamma$. □

In the completeness theorem, we now need to allow possible weakening on the left or on the right.

Theorem 5.4 (Completeness of Forward Sequent Calculus)

1. *If* $\Gamma \Longrightarrow C$ *then* $\Gamma' \longrightarrow C$ *or* $\Gamma' \longrightarrow \cdot$ *for some* $\Gamma' \subseteq \Gamma$*.*

Proof: By induction on the derivation $\mathcal{S}$ of $\Gamma \Longrightarrow C$. □

5.3 The Subformula Property

It is a general property of cut-free sequent calculi that all propositions occurring in a derivation are subformulas of the endsequent. In the forward direction we can therefore restrict the application of a rule to the case where the principal formula in the conclusion is a subformula of the goal sequent. We refine this property further by tracking positive and negative subformula occurrences. We then restrict left rule to introduce only negative subformulas of the goal sequent and right rules to positive subformulas of the goal sequent. To this end we introduce signed formulas.

$$\begin{array}{lrcl} \text{Positive} & A^+ & ::= & P^+ \mid A_1^+ \wedge A_2^+ \mid A_1^- \supset A_2^+ \mid A_1^+ \vee A_2^+ \mid \top^+ \mid \bot^+ \mid \neg A^- \\ \text{Negative} & A^- & ::= & P^- \mid A_1^- \wedge A_2^- \mid A_1^+ \supset A_2^- \mid A_1^- \vee A_2^- \mid \top^- \mid \bot^- \mid \neg A^+ \end{array}$$

It is obvious that every proposition can be annotated both positively and negatively, and that such an annotation is unique. We write Γ^- for a context $A_1^-, \ldots, A_n^-$ and γ^+ for an empty succedent or C^+. All inference rules for the sequent calculus can be annotated so that for a goal sequent $\Gamma^- \longrightarrow \gamma^+$, each sequent arising in the derivation has the same form, with only negative propositions on the left and positive propositions on the right (see Exercise 5.1). We say that A is a subformula of Γ or γ if A is a subformula of an element of Γ or γ, respectively, and similarly for signed propositions.

Theorem 5.5 (Signed Subformula Property)
Given a derivation $\mathcal{S}$ of $\Gamma^- \longrightarrow \gamma^+$. Then each sequent in $\mathcal{S}$ has the form $A_1^-, \ldots, A_n^- \longrightarrow B^+$ or $A_1^-, \ldots, A_n^- \longrightarrow \cdot$ where all A_i^- and B^+ are signed subformulas of Γ^- or γ^+.

Proof: By straightforward induction on the structure of $\mathcal{S}$. □

Note that this is a very strong theorem, since it asserts not only that every provable goal sequent has a derivation consisting of subformulas, but that *all* derivations of a provable sequent consist only of subformulas. A sequent not consisting of subformulas cannot contribute to a derivation of a goal sequent in the (cut-free) forward sequent calculus.

The subformula property immediately gives rise to a procedure for forward theorem proving. We start with all initial sequents of the form $A^- \longrightarrow A^+$ where both A^- and A^+ are signed subformulas of the goal sequent. We also have to add $\cdot \longrightarrow \top^+$ and $\bot^- \longrightarrow \cdot$ if $\top^+$ or $\bot^-$ are subformulas of the goal sequent, respectively.

Then we apply all possible inference rules where the principal proposition in the conclusion is a subformula of the goal sequent. We stop with success when we have generated the goal sequent, or if the goal sequent can be obtained from a generated sequent by weakening. We fail if any possible way of applying inference rules yields only sequents already in the database. In that case the goal sequent cannot be derivable if we have not encountered it (or a strengthened form of it) already.

We now show an example derivation in a linearized format. The goal sequent is $A \supset (B \supset C) \longrightarrow ((A \wedge B) \supset C)$. After signing each subformula we obtain

$$(A^+ \supset (B^+ \supset C^-)^-)^- \longrightarrow (((A^- \wedge B^-)^-) \supset C^+)^+$$

If show only the top-level sign, this leads to the following list of signed subformulas.

$$\begin{array}{l} A^+, B^+, C^-, A^-, B^-, C^+, \\ (B \supset C)^-, (A \wedge B)^-, \\ (A \supset (B \supset C))^-, ((A \wedge B) \supset C)^+ \end{array}$$

This means we have both positive and negative occurrences of A, B, and C and we have to consider three initial sequents.

1	$A^- \longrightarrow A^+$	init
2	$B^- \longrightarrow B^+$	init
3	$C^- \longrightarrow C^+$	init
4	$(A \wedge B)^- \longrightarrow A^+$	$\wedge L_1$ 1
5	$(A \wedge B)^- \longrightarrow B^+$	$\wedge L_1$ 2
6	$(A \wedge B)^-, (B \supset C)^- \longrightarrow C^+$	$\supset L$ 5 3
7	$(A \wedge B)^-, (A \supset (B \supset C))^- \longrightarrow C^+$	$\supset L$ 4 6
8	$(A \supset (B \supset C))^- \longrightarrow ((A \wedge B) \supset C)^+$	$\supset R_1$ 7

We use the horizontal lines to indicate iterations of an algorithm which derives all possible new consequences from the sequents already established. We have elided those sequents that do not contribute to the final derivation. For example, in the first step we can use $\supset R_2$ to conclude $C^- \longrightarrow ((A \wedge B) \supset C)^+$, from $C^- \longrightarrow C^+$, since the succedent is a positive subformula of the goal sequent.

Note that the inference of line 7 contains an implicit contraction, since $(A \wedge B)^-$ is an assumption in both premises (4 and 6).

5.4 Naming Subformulas

Without any further optimizations, the check if a given inference rule should be used in the forward direction is complicated, since we have to repeatedly scan the goal sequent for subformula occurrences. An integral part of the inverse method is to avoid this scan by introducing names for non-atomic subformulas and then specialize the inference rules to work only the names. We will not be formal about this optimization, since we view it as an implementation technique, but not an improvement of a logical nature. By expanding all newly defined names we obtain a derivation as in the previous section.

We return to the previous example to illustrate the technique. The goal sequent is $A \supset (B \supset C) \longrightarrow (A \wedge B) \supset C)$. After naming each subformula we obtain the signed atomic propositions

$$A^+, B^+, C^-, A^-, B^-, C^+,$$

and the new names

$$\begin{array}{rcl} L_1^- & = & B^+ \supset C^- \\ L_2^- & = & A^- \wedge B^- \\ L_3^- & = & A^+ \supset L_1^- \\ L_4^+ & = & L_2^- \supset C^+ \end{array}$$

We can now write out the general sequent calculus inference rules, specialized to the above labels. Since the goal sequent contains no negative occurrence of negation or falsehood, we may restrict the right-hand sides of all rules to be non-empty. This means only two implication right rules are necessary instead of three for L_4^+.

$$\frac{\Gamma_1 \longrightarrow B^+ \qquad \Gamma_2, C^- \longrightarrow \gamma}{\Gamma_1 \cup \Gamma_2, L_1^- \longrightarrow \gamma} \supset\!\mathrm{L}\ (L_1^-)$$

$$\frac{\Gamma, A^- \longrightarrow \gamma}{\Gamma, L_2^- \longrightarrow \gamma} \wedge\mathrm{L}_1\ (L_2^-) \qquad \frac{\Gamma, B^- \longrightarrow \gamma}{\Gamma, L_2^- \longrightarrow \gamma} \wedge\mathrm{L}_2\ (L_2^-)$$

$$\frac{\Gamma_1 \longrightarrow A^+ \qquad \Gamma_2, L_1^- \longrightarrow \gamma}{\Gamma_1 \cup \Gamma_2, L_3^- \longrightarrow \gamma} \supset\!\mathrm{L}\ (L_3^-)$$

$$\frac{\Gamma, L_2^- \longrightarrow C^+}{\Gamma \longrightarrow L_4^+} \supset\!\mathrm{R}_1\ (L_4^+) \qquad \frac{\Gamma \longrightarrow C^+}{\Gamma \longrightarrow L_4^+} \supset\!\mathrm{R}_2\ (L_4^+)$$

In its labeled form, the derivation above looks as follows.

1	$A^- \longrightarrow A^+$	init
2	$B^- \longrightarrow B^+$	init
3	$C^- \longrightarrow C^+$	init
4	$L_2^- \longrightarrow A^+$	$\wedge\mathrm{L}_1$ 1
5	$L_2^- \longrightarrow B^+$	$\wedge\mathrm{L}_1$ 2
6	$L_2^-, L_1^- \longrightarrow C^+$	$\supset$L 5 3
7	$L_2^-, L_3^- \longrightarrow C^+$	$\supset$L 4 6
8	$L_3^- \longrightarrow L_4^+$	$\supset\mathrm{R}_1$ 7

In the algorithm for labeling subterms we can avoid some redundancy if we give identical subterms the same label. However, this is not required for soundness and completeness, it only trims the search space.

Another choice arises for initial sequents. As in backwards search, we may restrict ourselves to atomic initial sequents or we may allow arbitrary labeled subformulas as long as they occur both negatively and positively. Tammet [Tam96] reports that allowing non-atomic initial sequents led to significant speed-up on a certain class of test problems. Of course, in their named form, even non-atomic sequents have the simple form $L^- \longrightarrow L^+$ for a label L.

5.5 Forward Subsumption

For the propositional case, we can obtain a decision procedure from the inverse method. We stop with success if we have reached the goal sequent (or a strengthened form of it) and with failure if any possible application of an inference rule leads to a sequent that is already present. This means we should devise a data structure or algorithm which allows us to check easily if the conclusion of an inference rule application is already present in the database of derived sequents. This check for equality should allow for permutations of hypotheses.

We can improve this further by not just checking equality modulo permutations, but taking weakening into account. For example, if we have derived $L_1^-, L_2^- \longrightarrow L_4^+$ then the sequent $L_1^-, L_2^-, L_3^- \longrightarrow L_4^+$ is redundant and could simply be obtained from the previous sequent by weakening. Similarly, $L_1^- \longrightarrow \cdot$ has more information than $L_1^- \longrightarrow L_2^+$, so the latter clause does not need to be kept if we have the former clause. Note that we already need this form of weakening to determine success if the goal sequent has assumptions. We say the a sequent S subsumes a sequent S' (written as $S \leq S'$) if S' can be obtains from S by weakening on the right and left.

In the propositional case, there is a relatively simple way to implement subsumption. We introduce a total ordering among all atomic propositions and also the new literals introduced during the naming process. Then we keep the antecedents of each sequent as an ordered list of atoms and literals. The union operation required in the implementation of inference rules with two premises, and the subset test required for subsumption can now both be implemented efficiently.

The reverse, called *backward subsumption* discards a previously derived sequent S if the new sequent S' subsumes S. Generally, backward subsumption is considered less fundamentally important. For example, it is not necessary to obtain a decision procedure for the propositional case. Implementations generally appear to be optimized for efficient forward subsumption.

[*the remainder of this section is speculative*]

However, it seems possible to exploit backward subsumption in a stronger way. Instead of simply deleting the subsumed sequent, we could strengthen its consequences, essentially by replaying the rules applied to it on the stronger sequent.

5.6 Proof Terms for the Inverse Method

The simplicity of the proof for the completeness theorem (Theorem 5.4) indicates that a proof term assignment should be relatively straightforward. The implicit contraction necessary when taking the union of two sets of antecedents presents the only complication. A straightforward solution seems to be to label each antecedent not with just a single variable, but with a set of variables. When taking the union of two sets of antecedents, we also need to take the union of the corresponding label sets. But this would require globally different variables for labeling antecedents in order to avoid interference between the premises of two-premise rules. Another possibility would be to assign a unique label to each negative subformula of the goal sequent and simply use this label in the proof term. This strategy will have to be reexamined in the first-order case, since a given literal may appear with different arguments.

Note that proof term assignment in the forward sequent calculus can be done *on-line* or *off-line*. In the on-line method we construct an appropriate proof term for each sequent at each inference step in a partial derivation. In the off-line method we keep track of the minimal information so we can recover the actual sequence of inference steps to arrive a the final conclusion. From this we reconstruct a proof term only once a complete sequent derivation has been found.

The on-line method would be preferable if we could use the proof term information to guide further inferences or subsumption; otherwise the off-line method is preferable since the overhead is reduced to a a validation phase once a proof has been found.

5.7 Exercises

Exercise 5.1 Show the forward sequent calculus on signed propositions and prove that if $\Gamma \longrightarrow A$ then $\Gamma^- \longrightarrow A^+$.

Exercise 5.2 In the exercise we explore add the connective $A \equiv B$ as a primitive to inverse method.

1. Following Exercise 2.6, introduce appropriate left and right rules to the backward sequent calculus.

2. Transform the rules to be appropriate for the forward sequent calculus.

3. Extend the notion of positive and negative subformula.

4. Extend the technique of subformula naming and inference rule specialization.

5. Show inverse derivations for each of the following.

   (a) Reflexivity: $\longrightarrow A \equiv A$.

   (b) Symmetry: $A \equiv B \longrightarrow B \equiv A$.

   (c) Transitivity: $A \equiv B, B \equiv C \longrightarrow A \equiv C$.

6. Compare your technique with thinking of $A \equiv B$ as a syntactic abbreviation for $(A \supset B) \wedge (B \supset A)$. Do you see significant advantages or disadvantages of your method?