Chapter 4

Focused Derivations

The sequent calculus as presented in the previous chapter is an excellent foundation for proof search strategies, but it is not yet practical. For a typical sequent there are many choices, such as which left or right rule to use to reduce the goal in the bottom-up construction of a proof. After one step, similar choices arise again, and so on. Without techniques to eliminate some of this non-determinism one would be quickly overwhelmed with multiple choices.

In this chapter we present two techniques to reduce the amount of non-determinism in search. The first are *inversion properties* which hold when the premises of an inference rule are derivable if and only if the conclusion is. This means that we do not lose completeness when applying an invertible rule as soon as it is applicable. The second are *focusing properties* which allow us to chain together non-invertible inference rules with consecutive principal formulas, once again without losing completeness.

While inversion and focusing are motivated by bottom-up proof search, they generally reduce the number of derivations in the search space. For this reason they also apply in top-down search procedures such as the inverse method introduced in Chapter 5.

4.1 Inversion

The simplest way to avoid non-determinism is to consider those propositions on the left or right for which there is a unique way to apply a corresponding left or right rule. For example, to prove $A \wedge B$ we can immediately apply the right rule without losing completeness. On the other hand, to prove $A \vee B$ we can not immediately apply a left rule. As a counterexample consider $B \vee A \Longrightarrow A \vee B$, where we first need to apply a left rule.

On a given sequent, a number of invertible rules may be applicable. However, the order of this choice does not matter. In other words, we have replaced *don't-know* non-determinism by *don't-care* non-determinism.

Determining the invertibility of left rules in order to support this strategy requires some additional considerations. The pure inversion property states that the premises should be derivable if and only if the conclusion is. However, in left rule the principal formula is still present in the premises, which means we can continue to apply the same left rule over and over again leading to non-termination. So we require in addition that the principal formula of a left rule is no longer needed, thereby guaranteeing the termination of the inversion phase of the search.

Theorem 4.1 (Inversion)

1. *If* $\Gamma \Longrightarrow A \wedge B$ *then* $\Gamma \Longrightarrow A$ *and* $\Gamma \Longrightarrow B$.
2. *If* $\Gamma \Longrightarrow A \supset B$ *then* $\Gamma, A \Longrightarrow B$.
3. *If* $\Gamma \Longrightarrow \forall x.\, A$ *then* $\Gamma \Longrightarrow [a/x]A$ *for a new individual parameter* a.
4. *If* $\Gamma \Longrightarrow \neg A$ *then* $\Gamma, A \Longrightarrow p$ *for a new propositional parameter* p.
5. *If* $\Gamma, A \wedge B \Longrightarrow C$ *then* $\Gamma, A, B \Longrightarrow C$.
6. *If* $\Gamma, \top \Longrightarrow C$ *then* $\Gamma \Longrightarrow C$.
7. *If* $\Gamma, A \vee B \Longrightarrow C$ *then* $\Gamma, A \Longrightarrow C$ *and* $\Gamma, B \Longrightarrow C$.
8. *If* $\Gamma, \exists x.\, A \Longrightarrow C$ *then* $\Gamma, [a/x]A \Longrightarrow C$ *for a new individual parameter* a.

Proof: By induction over the structure of the given derivations. Parts (5) and (6) are somewhat different in that they extract an inversion property from two and zero left rules, respectively. The proof is nonetheless routine.

Alternatively, we can take advantage of the admissibility of cut to avoid another inductive proof. For example, to show the first property, we can reason as follows:

$\Gamma \Longrightarrow A \wedge B$	Assumption
$\Gamma, A \wedge B, A \Longrightarrow A$	By rule init
$\Gamma, A \wedge B \Longrightarrow A$	By rule $\wedge L_1$
$\Gamma \Longrightarrow A$	By admissibility of cut (Theorem 3.11)

See also Exercise 4.1. □

The rules $\top R$ and $\bot L$ are a special case: they can be applied eagerly without losing completeness, but these rules have no premises and therefore do not admit a theorem of the form above. None of the other rules permit an inversion property, as the following counterexamples show. These counterexamples can easily be modifed so that they are not initial sequents.

1. $A \vee B \Longrightarrow A \vee B$ (both $\vee R_1$ or $\vee R_2$ lead to an unprovable sequent).
2. $\bot \Longrightarrow \bot$ (no right rule applicable).
3. $\exists x.\, A \Longrightarrow \exists x.\, A$ ($\exists R$ leads to an unprovable sequent).

4. $A \supset B \Longrightarrow A \supset B$ ($\supset$L leads to an unprovable sequent).

5. $\neg A \Longrightarrow \neg A$ ($\neg$L leads to an unprovable sequent).

6. $\forall x.\ A \Longrightarrow \forall x.\ A$ ($\forall$L leads to an unprovable sequent if we erase the original copy of $\forall x.\ A$).

Now we can write out a pure inversion strategy in the form of an inference system. One difficulty with such a system is that the don't-care non-determinism is not directly visible and has to be remarked on separately. We also refer to don't-care non-determinism as *conjunctive non-determinism*: eventually, all applicable rules have to be applied, but their order is irrelevant as far as provability is concerned.

First, we distinguish those kinds of propositions for which either the left or the right rule is *not* invertible. We call them *synchronous* propositions (either on the left or on the right).[1] The remaining propositions are called *asynchronous*. This terminology comes from the study of concurrency where an asynchronously computing processes proceed independently of all other processes, while a synchronously computing process may have to wait for other processes.

$$\begin{array}{rrcl} \text{Left synchronous propositions} & L & ::= & P \mid A_1 \supset A_2 \mid \forall x.\ A \\ \text{Right synchronous propositions} & R & ::= & P \mid A_1 \vee A_2 \mid \bot \mid \exists x.\ A \\ \text{Passive antecedents} & \Delta & ::= & \cdot \mid \Delta, L \end{array}$$

Note that we will revise this classification in Section 4.3. Sequents are composed of four judgments: left and right propositions, each of which may be active or passive. In order to simplify the notation, we collect like judgments into zones, keeping in mind that there can only be one proposition on the right. The active propositions that are decomposed asynchronously will be written in the center, the synchronous ones move to the outside for later consideration.

Sequents are written as

$$\Delta; \Omega \Longrightarrow A; \cdot \quad \text{and} \quad \Delta; \Omega \Longrightarrow \cdot; R$$

where the outer zones containing Δ or R are passive and the inner zones containing Ω or A are active. We still think of Δ as unordered, but it is important that Ω is ordered in order to avoid spurious non-deterministic choices. We must always work on its right end. We break down the principal connectives of asynchronous propositions eagerly, moving synchronous propositions into the passive zones, until all asynchronous connectives have been decomposed. At that point we have to choose one of the passive (synchronous) propositions. If this attempt fails we have to backtrack and try other choices.

In order to prove a sequent $\Gamma \Longrightarrow A$, we initialize our inversion-based procedure with the sequent $\cdot; \Gamma \Longrightarrow A; \cdot$, where the order we choose for the elements of Γ is irrelevant.

[1] For the moment, we do not consider negation explicitly—think of it as defined.

Right Asynchronous Propositions. First, we decompose the right asynchronous connectives.

$$\frac{\Delta; \Omega \Longrightarrow A; \cdot \qquad \Delta; \Omega \Longrightarrow B; \cdot}{\Delta; \Omega \Longrightarrow A \wedge B; \cdot} \wedge\text{R} \qquad \frac{}{\Delta; \Omega \Longrightarrow \top} \top\text{R}$$

$$\frac{\Delta; \Omega, A \Longrightarrow B; \cdot}{\Delta; \Omega \Longrightarrow A \supset B; \cdot} \supset\text{R} \qquad \frac{\Delta; \Omega \Longrightarrow [a/x]A; \cdot}{\Delta; \Omega \Longrightarrow \forall x.\ A; \cdot} \forall\text{R}^a$$

$$\frac{\Delta; \Omega \Longrightarrow \cdot; R}{\Delta; \Omega \Longrightarrow R; \cdot} R\text{R}$$

The last rule moves the right synchronous proposition into the passive zone.

Left Asynchronous Propositions. When the proposition on the right is passive, we break down the left asynchronous connectives in the active zone on the left. Recall that Ω is considered in order, so there is no non-determinism.

$$\frac{\Delta; \Omega, A, B \Longrightarrow \cdot; R}{\Delta; \Omega, A \wedge B \Longrightarrow \cdot; R} \wedge\text{L} \qquad \frac{\Delta; \Omega \Longrightarrow \cdot; R}{\Delta; \Omega, \top \Longrightarrow \cdot; R} \top\text{L}$$

$$\frac{\Delta; \Omega, A \Longrightarrow \cdot; R \qquad \Delta; \Omega, B \Longrightarrow \cdot; R}{\Delta; \Omega, A \vee B \Longrightarrow \cdot; R} \vee\text{L} \qquad \frac{}{\Delta; \Omega, \bot \Longrightarrow \cdot; R} \bot\text{L}$$

$$\frac{\Delta; \Omega, [a/x]A \Longrightarrow \cdot; R}{\Delta; \Omega, \exists x.\ A \Longrightarrow \cdot; R} \exists\text{L}^a$$

$$\frac{\Delta, L; \Omega \Longrightarrow \cdot; R}{\Delta; \Omega, L \Longrightarrow \cdot; R} L\text{L}$$

The last rule allows us to move synchronous propositions into the passive zone.

Right Synchronous Propositions. The active rules always terminate when applied in a bottom-up fashion during proof search (see Lemma 4.7). Now a don't-know non-deterministic choice arises: either we apply a right rule to infer R or a left rule to one of the passive assumptions in Δ. We also refer to don't-know non-determinism as *disjunctive non-determinism* since we have to pick one of several possibilities.

$$\frac{\Delta; \cdot \Longrightarrow A; \cdot}{\Delta; \cdot \Longrightarrow \cdot; A \vee B} \vee\text{R}_1 \qquad \frac{\Delta; \cdot \Longrightarrow B; \cdot}{\Delta; \cdot \Longrightarrow \cdot; A \vee B} \vee\text{R}_2$$

$$\textit{no right rule for } \bot \qquad \frac{\Delta; \cdot \Longrightarrow [t/x]A; \cdot}{\Delta; \cdot \Longrightarrow \cdot; \exists x.\ A} \exists\text{R}$$

In the last case we would have to guess the t, but in practice the t is determined by unification as indicated in Section 4.4.

Left Synchronous Propositions. Left synchronous propositions may be needed more than once, so they are duplicated in the application of the left rules.

$$\frac{\Delta, A \supset B; \cdot \Longrightarrow A; \cdot \qquad \Delta, A \supset B; B \Longrightarrow \cdot; R}{\Delta, A \supset B; \cdot \Longrightarrow \cdot; R} \supset\mathrm{L}$$

$$\frac{\Delta, \forall x.\ A; [t/x]A \Longrightarrow \cdot; R}{\Delta, \forall x.\ A; \cdot \Longrightarrow \cdot; R} \forall\mathrm{L}$$

Initial Sequents. This leaves the question of initial sequents, which is easily handled by allowing an passive atomic proposition on the left to match a passive atomic proposition on the right.

$$\frac{}{\Delta, P; \cdot \Longrightarrow \cdot; P} \mathrm{init}$$

The judgments $\Delta; \Omega \Longrightarrow A; \cdot$ and $\Delta; \Omega \Longrightarrow \cdot; R$ are hypothetical in Δ, but *not* hypothetical in Ω in the usual sense. This is because proposition in Ω do not persist, because they have to be empty in the initial sequents, and because they must be considered in order. In other words, contraction, weakening, and exchange are not available for Ω. These turn out to be admissible, but the structure of the proof is changed globally. Therefore we consider it an *ordered hypothetical judgment* where each hypothesis must be used exactly once in a derivation, in the given order. We do not formalize this notion any further, but just remark that appropriate versions of the substitution property can be devised to explain its meaning.

First, the soundness theorem is straightforward, since inversion proofs merely eliminate some disjunctive non-determinism.

Theorem 4.2 (Soundness of Inversion Proofs)
If $\Delta; \Omega \Longrightarrow A; \cdot$ *or* $\Delta; \Omega \Longrightarrow \cdot; A$ *then* $\Delta, \Omega \Longrightarrow A$.

Proof: By a straightforward induction over the given derivation, applying weakening in some cases. □

The completeness theorem requires a number of inversion lemmas. For a possible alternative path, see Exercise 4.2. The first set of results expresses the invertibility of the rules concerning the active propositions. That is, we can immediately apply any invertible rule witout losing completeness. The second set of results expresses the opposite: we can always postpone the non-invertible rules until all invertible rules have been applied.

We use the notation $\Delta; \Omega \Longrightarrow \rho$ to stand for $\Delta; \Omega \Longrightarrow A; \cdot$ or $\Delta; \Omega \Longrightarrow \cdot; R$.

Lemma 4.3 (Inversion on Asynchronous Connectives)

1. $\Delta; \Omega \Longrightarrow A \wedge B; \cdot$ *iff* $\Delta; \Omega \Longrightarrow A; \cdot$ *and* $\Delta; \Omega \Longrightarrow B; \cdot$.

2. $\Delta; \Omega \Longrightarrow A \supset B; \cdot$ *iff* $\Delta; \Omega, A \Longrightarrow B; \cdot$.

3. $\Delta; \Omega \Longrightarrow \forall x.\ A; \cdot$ *iff* $\Delta; \Omega \Longrightarrow [a/x]A; \cdot$ *for any new parameter* a.

4. $\Delta; \Omega \Longrightarrow R; \cdot$ *iff* $\Delta; \Omega \Longrightarrow \cdot; R$ *for* R *right synchronous.*

5. $\Delta; \Omega_1, A \wedge B, \Omega_2 \Longrightarrow \rho$ *iff* $\Delta; \Omega_1, A, B, \Omega_2 \Longrightarrow \rho$.

6. $\Delta; \Omega_1, \top, \Omega_2 \Longrightarrow \rho$ *iff* $\Delta; \Omega_1, \Omega_2 \Longrightarrow \rho$.

7. $\Delta; \Omega_1, A \vee B, \Omega_2 \Longrightarrow \rho$ *iff* $\Delta; \Omega_1, A, \Omega_2 \Longrightarrow \rho$ *and* $\Delta; \Omega_1, B, \Omega_2 \Longrightarrow \rho$.

8. $\Delta; \Omega_1, \exists x. A, \Omega_2 \Longrightarrow \rho$ *iff* $\Delta; \Omega_1, [a/x]A, \Omega_2 \Longrightarrow \rho$ *for any new param.* a.

9. $\Delta; \Omega_1, L, \Omega_2 \Longrightarrow \rho$ *iff* $\Delta, L; \Omega_1, \Omega_2 \Longrightarrow \rho$ *for* L *left synchronous.*

Proof: In each direction the result is either immediate by a rule, by inversion, or follows by a straightforward induction on the structure of the given derivation. □

The dual lemma shows that rules acting on synchronous propositions can be postponed until after the asynchronous rules. We define the *active size* of a sequent $\Delta; \Omega \Longrightarrow A; \cdot$ or $\Delta; \Omega \Longrightarrow \cdot; R$ as the number of logical quantifiers, connectives, constants, and atomic propositions in Ω and A. Note that the active size of a sequent is 0 if and only if it has the form $\Delta; \cdot \Longrightarrow \cdot; R$.

Lemma 4.4 (Postponement of Synchronous Connectives)

1. *If* $\Delta; \Omega \Longrightarrow A; \cdot$ *or* $\Delta; \Omega \Longrightarrow \cdot; A$ *then* $\Delta; \Omega \Longrightarrow \cdot; A \vee B$.

2. *If* $\Delta; \Omega \Longrightarrow B; \cdot$ *or* $\Delta; \Omega \Longrightarrow \cdot; B$ *then* $\Delta; \Omega \Longrightarrow \cdot; A \vee B$.

3. *If* $\Delta; \Omega \Longrightarrow [t/x]A; \cdot$ *or* $\Delta; \Omega \Longrightarrow \cdot; [t/x]A$ *then* $\Delta; \Omega \Longrightarrow \cdot; \exists x.\ A$.

4. *If* $(\Delta, A \supset B); (\Omega_1, \Omega_2) \Longrightarrow A; \cdot$ *and* $(\Delta, A \supset B); (\Omega_1, B, \Omega_2) \Longrightarrow \rho$ *then* $(\Delta, A \supset B); (\Omega_1, \Omega_2) \Longrightarrow \rho$.

5. *If* $(\Delta, \forall x.\ A); (\Omega_1, [t/x]A, \Omega_2) \Longrightarrow \rho$ *then* $(\Delta, \forall x.\ A); (\Omega_1, \Omega_2) \Longrightarrow \rho$.

Proof: By induction on the active size of the given sequent. For the right rules (parts (1), (2), and (3)), the base cases are $\Omega = \cdot$, in which case the conclusion follows directly by a rule. For the left rules, the base case is $\Omega = \cdot$ and $\rho = \cdot; R$, in which case the conclusion follows directly by a rule. In all other case we apply inversion to an element of Ω (Lemma 4.3) or C (if $\rho = C; \cdot$) and appeal to the induction hypothesis. Since the right-hand sides of the inversion principles have smaller active size than the left-hand sides, we are correct in applying the induction hypothesis. We show two cases in the proof of part (4).

Case: $\Omega = \cdot$ and $\rho = \cdot; R$.

$(\Delta, A \supset B); B \Longrightarrow \cdot; R$	Assumption
$(\Delta, A \supset B); \cdot \Longrightarrow A; \cdot$	Assumption
$(\Delta, A \supset B); \cdot \Longrightarrow \cdot; R$	By rule $\supset$L

Case: $\Omega = \Omega', C \vee D$.

$(\Delta, A \supset B); \Omega', C \vee D, B \Longrightarrow \rho$	Assumption
$(\Delta, A \supset B); \Omega', C, B \Longrightarrow \rho$ and	
$(\Delta, A \supset B); \Omega', D, B \Longrightarrow \rho$	By inversion
$(\Delta, A \supset B); \Omega', C \vee D \Longrightarrow A; \cdot$	Assumption
$(\Delta, A \supset B); \Omega', C \Longrightarrow A; \cdot$ and	
$(\Delta, A \supset B); \Omega', D \Longrightarrow A; \cdot$	By inversion
$(\Delta, A \supset B); \Omega', C \Longrightarrow \rho$	By i.h. on Ω', C
$(\Delta, A \supset B); \Omega', D \Longrightarrow \rho$	By i.h. on Ω', D
$(\Delta, A \supset B); \Omega', C \vee D \Longrightarrow \rho$	By rule $\vee$L

□

For the proof of completeness, and also to permit some optimizations in the search procedure, we need to show that weakening and contraction for propositions in Ω are admissible, at the price of possibly lengthening the derivation. Note that weakening and contraction for Δ is trivial, since inversion sequents *are* hypothetical in Δ.

Lemma 4.5 (Structural Properties of Inversion Sequents)

1. *If* $\Delta; \Omega \Longrightarrow \rho$ *then* $(\Delta, A); \Omega \Longrightarrow \rho$.
2. *If* $(\Delta, A, A); \Omega \Longrightarrow \rho$ *then* $(\Delta, A); \Omega \Longrightarrow \rho$.
3. *If* $\Delta; (\Omega_1, \Omega_2) \Longrightarrow \rho$ *then* $\Delta; (\Omega_1, A, \Omega_2) \Longrightarrow \rho$.
4. *If* $\Delta; (\Omega_1, A, A, \Omega_2) \Longrightarrow \rho$ *then* $\Delta; (\Omega_1, A, \Omega_2) \Longrightarrow \rho$.

Proof: Parts (1) and (2) follow as usual by straightforward structural inductions over the given derivations. Parts (3) and (4) follow by induction on the structure of A, taking advantage of the inversion properties for asynchronous propositions (Lemma 4.3) and parts (1) and (2) for synchronous propositions. □

Theorem 4.6 (Completeness of Inversion Proofs)
If $\Omega \Longrightarrow A$ *then* $\cdot; \Omega \Longrightarrow A; \cdot$.

Proof: By induction on the structure of the given sequent derivation $\mathcal{S}$, taking advantage of the inversion, postponement, and structural properties proven in this section. We think of the ordinary left rules of the sequent calculus as operating on some proposition in the middle of Ω, rather than explicitly dealing with exchange. We consider in turn: invertible right rules, invertible left rules, initial sequents, non-invertible right rules and non-invertible left rules.

Case:

$$\mathcal{S} = \cfrac{\begin{array}{cc} \mathcal{S}_1 & \mathcal{S}_2 \\ \Omega \Longrightarrow A_1 & \Omega \Longrightarrow A_2 \end{array}}{\Omega \Longrightarrow A_1 \wedge A_2} \wedge\mathrm{R}$$

$\cdot; \Omega \Longrightarrow A_1; \cdot$ By i.h. on $\mathcal{S}_1$
$\cdot; \Omega \Longrightarrow A_2; \cdot$ By i.h. on $\mathcal{S}_2$
$\cdot; \Omega \Longrightarrow A_1 \wedge A_2; \cdot$ By Lemma 4.3(1)

Cases: The right invertible rules $\supset$R and $\forall$R and also the case for $\top$R are similar to the case for $\wedge$R.

Case:

$$\mathcal{S} = \cfrac{\begin{array}{cc} \mathcal{S}_1 & \mathcal{S}_2 \\ \Omega_1, B_1 \vee B_2, B_1, \Omega_2 \Longrightarrow A & \Omega_1, B_1 \vee B_2, B_2, \Omega_2 \Longrightarrow A \end{array}}{\Omega_1, B_1 \vee B_2, \Omega_2 \Longrightarrow A} \vee\mathrm{L}$$

$\cdot; \Omega_1, B_1 \vee B_2, B_1, \Omega_2 \Longrightarrow A; \cdot$ By i.h. on $\mathcal{S}_1$
$\cdot; \Omega_1, B_1 \vee B_2, B_2, \Omega_2 \Longrightarrow A; \cdot$ By i.h. on $\mathcal{S}_2$
$\cdot; \Omega_1, B_1 \vee B_2, B_1 \vee B_2, \Omega_2 \Longrightarrow A; \cdot$ By Lemma 4.3(7)
$\cdot; \Omega_1, B_1 \vee B_2, \Omega_2 \Longrightarrow A; \cdot$ By contraction (Lemma 4.5)

Cases: The left invertible rule $\exists$L and also the case for $\bot$L are similar to the case for $\vee$L.

Case:

$$\mathcal{S} = \cfrac{\begin{array}{c} \mathcal{S}_1 \\ \Omega_1, B_1 \wedge B_2, B_1, \Omega_2 \Longrightarrow A \end{array}}{\Omega_1, B_1 \wedge B_2, \Omega_2 \Longrightarrow A} \wedge\mathrm{L}_1$$

$\cdot; \Omega_1, B_1 \wedge B_2, B_1, \Omega_2 \Longrightarrow A; \cdot$ By i.h. on $\mathcal{S}_1$
$\cdot; \Omega_1, B_1 \wedge B_2, B_1, B_2, \Omega_2 \Longrightarrow A; \cdot$ By weakening (Lemma 4.5)
$\cdot; \Omega_1, B_1 \wedge B_2, B_1 \wedge B_2, \Omega_2 \Longrightarrow A$ By Lemma 4.3(5)
$\cdot; \Omega_1, B_1 \wedge B_2, \Omega_2 \Longrightarrow A$ By contraction (Lemma 4.5)

Case: The case for $\wedge\mathrm{L}_2$ is symmetric to $\wedge\mathrm{L}_1$. Note that there is no left rule for $\top$ in the sequent calculus, so the $\top$L rule on inversion sequents arises only from weakening (see the following case).

Case:

$$\mathcal{S} = \cfrac{}{\Omega_1, P, \Omega_2 \Longrightarrow P} \mathrm{init}$$

$$
\begin{array}{ll}
P; \cdot \Longrightarrow \cdot; P & \text{By rule init} \\
\cdot; P \Longrightarrow \cdot; P & \text{By rule } L\text{L} \\
\cdot; P \Longrightarrow P; \cdot & \text{By rule } R\text{R} \\
\cdot; \Omega_1, P, \Omega_2 \Longrightarrow P; \cdot & \text{By weakening (Lemma 4.5)}
\end{array}
$$

Case:

$$
\mathcal{S} = \cfrac{\begin{array}{c}\mathcal{S}_1 \\ \Omega \Longrightarrow A_1\end{array}}{\Omega \Longrightarrow A_1 \vee A_2} \vee\text{R}_1
$$

$$
\begin{array}{ll}
\cdot; \Omega \Longrightarrow A_1; \cdot & \text{By i.h. on } \mathcal{S}_1 \\
\cdot; \Omega \Longrightarrow \cdot; A_1 \vee A_2 & \text{By postponement (Lemma 4.4)} \\
\cdot; \Omega \Longrightarrow A_1 \vee A_2; \cdot & \text{By rule } R\text{R}
\end{array}
$$

Cases: The cases for the non-invertible right rules $\vee\text{R}_2$ and $\exists\text{R}$ are similar to $\vee\text{R}_1$.

Case:

$$
\mathcal{S} = \cfrac{\begin{array}{c}\mathcal{S}_1 \\ \Omega_1, B_1 \supset B_2, \Omega_2 \Longrightarrow B_1\end{array} \qquad \begin{array}{c}\mathcal{S}_2 \\ \Omega_1, B_1 \supset B_2, B_2, \Omega_2 \Longrightarrow A\end{array}}{\Omega_1, B_1 \supset B_2, \Omega_2 \Longrightarrow A} \supset\text{L}
$$

$$
\begin{array}{ll}
\cdot; \Omega_1, B_1 \supset B_2, \Omega_2 \Longrightarrow B_1; \cdot & \text{By i.h. on } \mathcal{S}_1 \\
B_1 \supset B_2; \Omega_1, \Omega_2 \Longrightarrow B_1; \cdot & \text{By inversion (Lemma 4.3(9))} \\
\cdot; \Omega_1, B_1 \supset B_2, B_2, \Omega_2 \Longrightarrow A; \cdot & \text{By i.h. on } \mathcal{S}_2 \\
B_1 \supset B_2; \Omega_1, B_2, \Omega_2 \Longrightarrow A; \cdot & \text{By inversion (Lemma 4.3(9))} \\
B_1 \supset B_2; \Omega_1, \Omega_2 \Longrightarrow A; \cdot & \text{By postponement (Lemma 4.4)} \\
\cdot; \Omega_1, B_1 \supset B_2, \Omega_2 \Longrightarrow A; \cdot & \text{By Lemma 4.3(9)}
\end{array}
$$

Case: The cases for the non-invertible left rule $\forall\text{L}$ is similar to $\supset\text{L}$.

□

We can also show that the active rules always terminate, which is important for the algorithm.

Lemma 4.7 (Termination of Active Rules)
Given a goal $\Delta; \Omega \Longrightarrow \rho$. Any sequence of applications of active rules terminates.

Proof: By induction on the active size of the given sequent. □

Next we describe a non-deterministic algorithm for proof search. There are a number of ways to eliminate the remaining disjunctive non-determinism. Typical is depth-first search, made complete by iterative deepening. The choice of the term t in the rules $\exists\text{R}$ and $\forall\text{L}$ is later solved by introducing free variables and equational constraints into the search procedures which are solved by unification (see Section 4.4). Many futher refinements and improvements are possible on this procedures, but not discussed here.

Given a goal $\Delta; \Omega \Longrightarrow \rho$.

1. If $\Omega = \cdot$ and $\rho = \cdot; P$ succeed if P is in Δ.

2. If $\Omega = \cdot$ and $\rho = \cdot; R$, but the previous case does not apply, guess an inference rule to reduce the goal. In the cases of $\exists$R and $\forall$L we also have to guess a term t. Solve each subgoal by recursively applying the procedure. This case represents a disjunctive choice (don't know non-determinism). If no rule applies, we fail.

3. If Ω is non-empty or $\rho = A; \cdot$, use the unique applicable active rule and solve each of the subgoals by recursively applying the procedure.

This search procedure is clearly sound, because the inversion proof system is sound (Theorem 4.2). Furthermore, if there is a derivation the procedure will (in principle) always terminate and find some derivation if it guesses correctly in step (2).

4.2 Backchaining

While the inversion properties from the previous section are critical for constructing efficient theorem provers, they far from sufficient. The difficulty is that many non-deterministic choices remain. In this section we discuss a particular strategy called *backchaining* which has applications outside of theorem proving, for example, in logic programming. We restrict ourselves to *Horn logic*, a particularly simple logic that is useful in many circumstances. In the next section we describe *focusing*, which is the generalization of backchaining to full intuitionistic logic.

In many theorem proving problems we are in a situation where we have a number of propositions describing a *theory* and then a proposition we would like to prove with respect to that theory. Theories are often given in the form of propositions $\forall x_1 \ldots \forall x_n.\ P_1 \wedge \ldots \wedge P_k \supset P$. These hypotheses are synchronous (in the sense of the previous section), that is, we have to choose between them when trying to prove some atomic proposition Q. Backchaining rests on two observations. The first is that search remains complete if we only try to use those assumptions where P and Q can be made equal by instantiating $x_1, \ldots, x_n$ with appropriate terms. The second is that once we decide which assumption to use, we can apply a whole sequence of left rules (here $\forall$L and $\supset$L) without considering any other synchronous assumption.

Both of these observation are of crucial importance. The first cuts down on the number of assumptions we may use. The second drastically reduces the non-determinism. To see the latter, consider a theory with m clauses defining a predicate p and that ach clause has n universal quantifiers. With backchaining (and unification, see Section 4.4) we create one choice with m alternatives. With just the inversion strategy, we have m choices in the first step, then $m+1$ choices in the second step after instantiating one quantifier, and so on, yielding

$m(m+1)\cdots(m+p)$ choices. As the main theorem of this section and the next shows, these choices are redundant.

We first define Horn clauses in a form that is slightly more general than what is usually given in the literature.

$$\begin{array}{rrcl}
\text{Horn clauses} & D & ::= & P \mid G \supset D \mid \forall x.\, D \\
\text{Horn goals} & G & ::= & P \mid G_1 \wedge G_1 \mid \top \\
\text{Horn theories} & \Delta & ::= & \cdot \mid \Delta, D
\end{array}$$

Some further generalizations are possible; important for us is the absence of implications and universal quantification in goals as well as existential, disjunction, and falsehood in clauses.

A theorem proving problem in Horn logic is stated as

$$\Delta \Longrightarrow G$$

where Δ is a Horn theory and G is a Horn goal, that is, a conjunction of atomic propositions.

As two simple examples of Horn theories we consider even and odd numbers, and graph reachability.

For even/odd number we have constants $\mathsf{0}$ and s to represent the natural numbers in unary form. As usual, we abbreviate $\mathsf{0}()$ with just $\mathsf{0}$.

$$\begin{array}{l}
\mathsf{even}(\mathsf{0}), \\
\forall x.\, \mathsf{even}(x) \supset \mathsf{odd}(\mathsf{s}(x)), \\
\forall x.\, \mathsf{odd}(x) \supset \mathsf{even}(\mathsf{s}(x))
\end{array}$$

For reachability in a directed graph we assume we have a constant for each node in the graph and an assumption $\mathsf{edge}(a, b)$ for each edge from node a to node b. In addition we assume

$$\begin{array}{l}
\forall x.\, \forall y.\, \mathsf{edge}(x, y) \supset \mathsf{reach}(x, y), \\
\forall x.\, \forall y.\, \forall z.\, \mathsf{reach}(x, y) \wedge \mathsf{reach}(y, z) \supset \mathsf{reach}(x, z)
\end{array}$$

In the even/odd example, we would like for backchaining to reduce the goal $\mathsf{even}(\mathsf{s}(\mathsf{s}(\mathsf{0})))$ to the subgoal $\mathsf{odd}(\mathsf{s}(\mathsf{0}))$. In this case this reduction should be essentially deterministic, because only the last clause could match the goal. We formalize backchaining with the following two judgments.

$$\begin{array}{ll}
\Delta \overset{u}{\Longrightarrow} G & \text{Horn theory } \Delta \text{ proves } G \text{ uniformly} \\
\Delta; D \overset{u}{\Longrightarrow} P & \text{Backchaining on Horn clause } D \text{ proves } P
\end{array}$$

First the rules of uniform proof, which are rather simple. The critical one is the last, which selects a Horn clause from Δ for backchaining.

$$\frac{\Delta \overset{u}{\Longrightarrow} G_1 \qquad \Delta \overset{u}{\Longrightarrow} G_2}{\Delta \overset{u}{\Longrightarrow} G_1 \wedge G_2}\wedge R \qquad \frac{}{\Delta \overset{u}{\Longrightarrow} \top}\top R$$

$$\frac{\Delta; D \overset{u}{\Longrightarrow} P \qquad (D \text{ in } \Delta)}{\Delta \overset{u}{\Longrightarrow} P}\text{select}$$

The rules for backchaining consider the possible forms of the Horn clause, decomposing it by a left rule. When using this as a proof search procedure by interpreting it bottom-up, we imagine using unification variables instead of guessing terms, and solving left-most premises first.

$$\frac{}{\Delta; P \overset{u}{\Longrightarrow} P}\,\text{init} \qquad (\Delta; P \overset{u}{\Longrightarrow} Q \text{ fails for } P \neq Q)$$

$$\frac{\Delta; D \overset{u}{\Longrightarrow} P \qquad \Delta \overset{u}{\Longrightarrow} G}{\Delta; G \supset D \overset{u}{\Longrightarrow} P}\,{\supset}\text{L} \qquad \frac{\Delta; [t/x]D \overset{u}{\Longrightarrow} P}{\Delta; \forall x.\, D \overset{u}{\Longrightarrow} P}\,\forall\text{L}$$

It is not difficult to see that this indeed captures the intended proof search strategy for backchaining. It is also rather straightforward to prove it sound and complete.

Theorem 4.8 (Soundness of Uniform Proofs in Horn Theories)

1. *If* $\Delta \overset{u}{\Longrightarrow} G$ *then* $\Delta \Longrightarrow G$.
2. *If* $\Delta; D \overset{u}{\Longrightarrow} G$ *then* $\Delta, D \Longrightarrow G$.

Proof: By straightforward induction over the given derivations. In the case of the select rule, we require the admissibility of contraction in the sequent calculus. □

For the completeness direction we need a postponement lemma, similar to the case of inversion proofs. This lemma demonstrates that the left rules of the sequent calculus are admissible for the passive propositions of uniform sequents.

Lemma 4.9 (Postponement for Uniform Proofs)

1. *If* $\Delta, G \supset D, D; D' \overset{u}{\Longrightarrow} P$ *and* $\Delta, G \supset D \overset{u}{\Longrightarrow} G$ *then* $\Delta, G \supset D; D' \overset{u}{\Longrightarrow} P$
2. *If* $\Delta, G \supset D, D \overset{u}{\Longrightarrow} G'$ *and* $\Delta, G \supset D \overset{u}{\Longrightarrow} G$ *then* $\Delta, G \supset D \overset{u}{\Longrightarrow} G'$
3. *If* $\Delta, \forall x.\, D, [t/x]D; D' \overset{u}{\Longrightarrow} P$ *then* $\Delta, \forall x.\, D; D' \overset{u}{\Longrightarrow} P$
4. *If* $\Delta, \forall x.\, D, [t/x]D \overset{u}{\Longrightarrow} G'$ *then* $\Delta, \forall x.\, D \overset{u}{\Longrightarrow} G'$

Proof: By straightforward inductions over the first given derivation. □

Theorem 4.10 (Completness of Uniform Proofs in Horn Theories)

1. *If* $\Delta \Longrightarrow G$ *then* $\Delta \overset{u}{\Longrightarrow} G$.
2. *If* $\Delta \Longrightarrow P$ *then there is a* D *in* Δ *such that* $\Delta; D \overset{u}{\Longrightarrow} P$.

Proof: Part (1) follows by inversion properties of the sequent calculus. We show one case of Part (2).

Case:

$$\mathcal{S} = \cfrac{\begin{array}{cc} \mathcal{S}_1 & \mathcal{S}_2 \\ \Delta', G \supset D \Longrightarrow G & \Delta', G \supset D, D \Longrightarrow P \end{array}}{\Delta', G \supset D \Longrightarrow P} \supset\text{L}$$

$\Delta', G \supset D, D; D' \xrightarrow{u} P$ for some D' in $\Delta', G \supset D, D$	By i.h. on $\mathcal{S}_2$
$\Delta', G \supset D \xrightarrow{u} G$	By i.h. on $\mathcal{S}_1$
$\Delta', G \supset D; D' \xrightarrow{u} P$	By Lemma 4.9
If D' in $\Delta', G \supset D$ we are done	
If $D' = D$:	
$\Delta', G \supset D; G \supset D \xrightarrow{u} P$	By rule $\supset$L

□

Horn theories have a number of important properties. Some of these stem from the fact that during proof search, the collection of assumptions Δ never changes, nor will there ever be any new parameters introduced. This allows us to give an *inductive* interpretation to the set of clauses. For example, we could reason inductively about properties of even numbers, rather than just reason in first-order logic.

A related property is that Horn clauses can be seen to define *inference rules*. For example, we can translate the theory defining the even and odd numbers into the rules

$$\frac{}{\mathsf{even}(0)} \qquad \frac{\mathsf{even}(t)}{\mathsf{odd}(\mathsf{s}(t))} \qquad \frac{\mathsf{odd}(t)}{\mathsf{even}(\mathsf{s}(t))}$$

In fact, one can see the uniform proof system and backchaining as implementing precisely these rules. In other words, we can also *compile* a Horn theory into a set of inference rules and then prove Horn goals from no assumptions, but using the additional rules.

This view is also interesting in that it provides the basis for a forward-reasoning procedure for Horn logic that resembles the inverse method. However, all sequents we ever consider have an empty left-hand side! That is, from some atomic facts, using unary inference rules (possibly with multiple premises), we derive further facts. We illustrate this way of proceeding using our second Horn theory which implements a particular graph. First, we turning the theory

$$\begin{array}{l} \forall x.\, \forall y.\, \mathsf{edge}(x, y) \supset \mathsf{reach}(x, y), \\ \forall x.\, \forall y.\, \forall z.\, \mathsf{reach}(x, y) \wedge \mathsf{reach}(y, z) \supset \mathsf{reach}(x, z) \end{array}$$

into the inference rules

$$\frac{\mathsf{edge}(s, t)}{\mathsf{reach}(s, t)} \qquad \frac{\mathsf{reach}(s, t) \qquad \mathsf{reach}(t, u)}{\mathsf{reach}(s, u)}$$

Second, assume we start with facts

$$\mathsf{edge}(a,b), \mathsf{edge}(b,c)$$

Applying all possible rules we obtain

$$\begin{array}{l}\mathsf{edge}(a,b), \mathsf{edge}(b,c),\\ \mathsf{reach}(a,b), \mathsf{reach}(b,c)\end{array}$$

After one more step we have

$$\begin{array}{l}\mathsf{edge}(a,b), \mathsf{edge}(b,c),\\ \mathsf{reach}(a,b), \mathsf{reach}(b,c),\\ \mathsf{reach}(a,c)\end{array}$$

Now applying any more rules does not add any more facts: the set of facts is *saturated*. We can now see if the goal (e.g., $\mathsf{reach}(c,a)$) is in the saturated set or not. If yes it is true, if not it cannot be derived from the given facts.

The above strategy can be generalized to the case of facts with free variables (which are universally interpreted) and is known under the name of *unit resolution*.

It is interesting that the forward chaining strategy works particularly well for Horn theories such as for reach which can easily be seen to be terminating. This is because no new terms are constructed during the inferences. On the other hand, the backward chaining strategy we exemplified using even and odd can easily be seen to be terminating in the backward directions because the term involved get smaller.

As far as I know, it is still an open research problem how backward chaining and forward chaining (here illustrated with unit resolution) can be profitably combined. Also, the relationship between the inverse method and unit (or general) resolution is unclear in the sense that we do not know of a proposal that effectively combines these strategies.

4.3 Focusing

The search procedure based on inversion developed in Section 4.1 still has an unacceptable amount of don't know non-determinism. For the Horn fragment, we addressed this issue in Section 4.2; here we combine backchaining with inversion in order to obtain a method that works for full intuitionistic logic.

We first recall the problem with the inversion strategy. The problem lies in the undisciplined use and proliferation of assumptions whose left rule is not invertible.

In a typical situation we have some universally quantified implications as assumptions. For example, Δ could be

$$\begin{array}{l}\forall x_1.\,\forall y_1.\,\forall z_1.\; P_1(x_1,y_1,z_1) \supset Q_1(x_1,y_1,z_1) \supset R_1(x_1,y_1,z_1),\\ \forall x_2.\,\forall y_2.\,\forall z_2.\; P_2(x_2,y_2,z_2) \supset Q_2(x_2,y_2,z_2) \supset R_2(x_2,y_2,z_2)\end{array}$$

If the right-hand side is passive, we now have to apply $\forall$L to one of the two assumptions. We assume we guess the first one and that we can guess an appropriate term t_1. After the $\forall$L rule and a left transition, we are left with

$$\begin{array}{l}\forall x_1.\, \forall y_1.\, \forall z_1.\, P_1(x_1, y_1, z_1) \supset Q_1(x_1, y_1, z_1) \supset R_1(x_1, y_1, z_1),\\ \forall x_2.\, \forall y_2.\, \forall z_2.\, P_2(x_2, y_2, z_2) \supset Q_2(x_2, y_2, z_2) \supset R_2(x_2, y_2, z_2),\\ \forall y_1.\, \forall z_1.\, P_1(t_1, y_1, z_1) \supset Q_1(t_1, y_1, z_1) \supset R_1(t_1, y_1, z_1).\end{array}$$

Again, we are confronted with a don't know non-deterministic choice, now between 3 possibilities. One can see that the number of possible choices quickly explodes. We can observe that the pattern above does not coincide with mathematical practice. Usually one applies an assumption or lemma of the form above by instantiating all the quantifiers and all preconditions at once. This strategy called *focusing* is a refinement of the inversion strategy.

Roughly, when all propositions in a sequent are synchronous, we *focus* either on an assumption or the proposition we are trying to prove and then apply a sequence of non-invertible rules to the chosen proposition. This phase stops when either an invertible connective or an atomic proposition is reached.

The focusing strategy is defined by four judgments

$$\begin{array}{ll}\Delta; \Omega \stackrel{a}{\Longrightarrow} A; \cdot & \text{Decompose right asynchronous proposition}\\ \Delta; \Omega \stackrel{a}{\Longrightarrow} \cdot; R & \text{Decompose left asynchronous propositions}\\ \Delta; A \stackrel{s}{\Longrightarrow} \cdot; R & \text{Focus on left synchronous proposition}\\ \Delta; \cdot \stackrel{s}{\Longrightarrow} A; \cdot & \text{Focus on right synchronous proposition}\end{array}$$

The first two judgment are very similar to the inversion strategy. When we have the situation $\Delta; \cdot \stackrel{a}{\Longrightarrow} \cdot; R$ where Δ consists of left synchronous propositions and R is right synchronous, we focus either on R or on some proposition L in Δ and chain together inferences on the those propositions.

As in the inversion judgment, the proposition on the outside of the four zones are passive, while the ones on the inside are actively decomposed.

For the strategy to be maximally effective, we slightly generalize our classification of connectives, permitting conjunction and truth to be viewed as either synchronous or asynchronous, depending on what is convenient. This allows us to extend the phases maximally, removing as much non-determinism as possible.

	Asynchronous	Synchronous
Right	$\wedge, \top, \supset, \forall$	$\wedge, \top, \vee, \bot, \exists$
Left	$\wedge, \top, \vee, \bot, \exists$	$\wedge, \top, \supset, \forall$

We now use R for propositions that are *not* right asynchronous $(\vee, \bot, \exists, P)$ and L for propositions that are *not* left asynchronous $(\supset, \forall, P)$.

Except for the special status of conjunction and truth, each connective has unique and complementary status on the left and on the right. Andreoli's original analysis [And92] was carried out in linear logic, which did not show these anomalies. This is because there are two forms of conjunction (additive and multiplicative), each with a unique status.

We first repeat the inversion rules which constitute an asynchronous phase during search.

Right Asynchronous Propositions. First, we decompose the right asynchronous connectives.

$$\frac{\Delta; \Omega \overset{a}{\Longrightarrow} A; \cdot \qquad \Delta; \Omega \overset{a}{\Longrightarrow} B; \cdot}{\Delta; \Omega \overset{a}{\Longrightarrow} A \wedge B; \cdot} \wedge\mathrm{R} \qquad \frac{}{\Delta; \Omega \overset{a}{\Longrightarrow} \top; \cdot} \top\mathrm{R}$$

$$\frac{\Delta; \Omega, A \overset{a}{\Longrightarrow} B; \cdot}{\Delta; \Omega \overset{a}{\Longrightarrow} A \supset B; \cdot} \supset\mathrm{R} \qquad \frac{\Delta; \Omega \overset{a}{\Longrightarrow} [a/x]A; \cdot}{\Delta; \Omega \overset{a}{\Longrightarrow} \forall x.\ A; \cdot} \forall\mathrm{R}^a$$

$$\frac{\Delta; \Omega \overset{a}{\Longrightarrow} \cdot; R \quad (R = A \vee B, \bot, \exists x.\ A, P)}{\Delta; \Omega \overset{a}{\Longrightarrow} R; \cdot} R\mathrm{R}$$

Left Asynchronous Propositions. Next we break down the left asynchronous propositions. Recall that Ω is considered in order, so the rules are deterministic.

$$\frac{\Delta; \Omega, A, B \overset{a}{\Longrightarrow} \cdot; R}{\Delta; \Omega, A \wedge B \overset{a}{\Longrightarrow} \cdot; R} \wedge\mathrm{L} \qquad \frac{\Delta; \Omega \overset{a}{\Longrightarrow} \cdot; R}{\Delta; \Omega, \top \overset{a}{\Longrightarrow} \cdot; R} \top\mathrm{L}$$

$$\frac{\Delta; \Omega, A \overset{a}{\Longrightarrow} \cdot; R \qquad \Delta; \Omega, B \overset{a}{\Longrightarrow} \cdot; R}{\Delta; \Omega, A \vee B \overset{a}{\Longrightarrow} \cdot; R} \vee\mathrm{L} \qquad \frac{}{\Delta; \Omega, \bot \overset{a}{\Longrightarrow} \cdot; R} \bot\mathrm{L}$$

$$\frac{\Delta; \Omega, [a/x]A \overset{a}{\Longrightarrow} \cdot; R}{\Delta; \Omega, \exists x.\ A \overset{a}{\Longrightarrow} \cdot; R} \exists\mathrm{L}^a$$

$$\frac{\Delta, L; \Omega \overset{a}{\Longrightarrow} \cdot; R \quad (L = A \supset B, \forall x.\ A, P)}{\Delta; \Omega, L \overset{a}{\Longrightarrow} \cdot; R} L\mathrm{L}$$

Focus. Next we need to decide which proposition among Δ and R to focus on. While we allow focusing on an atomic assumption, focusing on the succedent requires it to be non-atomic. The reason is our handling of initial sequents. For uniformity we also include $\bot$, even though focusing on it will fail in the next step.

$$\frac{(\Delta, L); L \overset{s}{\Longrightarrow} \cdot; R}{(\Delta, L); \cdot \overset{a}{\Longrightarrow} \cdot; R} \mathrm{focus}L \qquad \frac{\Delta; \cdot \overset{s}{\Longrightarrow} R; \cdot \quad (R = A \vee B, \bot, \exists x.\ A)}{\Delta; \cdot \overset{a}{\Longrightarrow} \cdot; R} \mathrm{focus}R$$

Right Synchronous Propositions. The non-invertible rules on the right maintain the focus on principal formula of the inference. When we have reduced the right-hand side to an asynchronous (but not synchronous) or atomic proposition, we blur our focus and initiate an asynchronous phase.

$$\frac{\Delta; \cdot \overset{s}{\Longrightarrow} A; \cdot}{\Delta; \cdot \overset{s}{\Longrightarrow} A \vee B; \cdot}\vee\mathrm{R}_1 \qquad \frac{\Delta; \cdot \overset{s}{\Longrightarrow} B; \cdot}{\Delta; \cdot \overset{s}{\Longrightarrow} A \vee B; \cdot}\vee\mathrm{R}_2$$

$$\text{no right focus rule for } \bot \qquad \frac{\Delta; \cdot \overset{s}{\Longrightarrow} [t/x]A; \cdot}{\Delta; \cdot \overset{s}{\Longrightarrow} \exists x.\ A; \cdot}\exists\mathrm{R}$$

$$\frac{\Delta; \cdot \overset{a}{\Longrightarrow} A; \cdot \quad (A = B \supset C, \forall x.\ B, P)}{\Delta; \cdot \overset{s}{\Longrightarrow} A; \cdot}\mathrm{blur}R$$

Left Synchronous Propositions. The non-invertible rules on the left also maintain their focus on the principal formula of the inference. When we have reached an asynchronous (but not synchronous) proposition, we blur our focus and initiate an asynchrounous phase.

$$\frac{\Delta; B \overset{s}{\Longrightarrow} \cdot; R \qquad \Delta; \cdot \overset{s}{\Longrightarrow} A; \cdot}{\Delta; A \supset B \overset{s}{\Longrightarrow} \cdot; R}\supset\mathrm{L} \qquad \frac{\Delta; [t/x]A \overset{s}{\Longrightarrow} \cdot; R}{\Delta; \forall x.\ A \overset{s}{\Longrightarrow} \cdot; R}\forall\mathrm{L}$$

$$\frac{\Delta; A \overset{s}{\Longrightarrow} \cdot; R}{\Delta; A \wedge B \overset{s}{\Longrightarrow} \cdot; R}\wedge\mathrm{L}_1 \qquad \frac{\Delta; B \overset{s}{\Longrightarrow} \cdot; R}{\Delta; A \wedge B \overset{s}{\Longrightarrow} \cdot; R}\wedge\mathrm{L}_2$$

$$\text{no rule for } \top\mathrm{L} \qquad \frac{\Delta; A \overset{a}{\Longrightarrow} \cdot; R \quad (A = B \vee C, \bot, \exists x.\ B)}{\Delta; A \overset{s}{\Longrightarrow} \cdot; R}\mathrm{blur}L$$

$$\frac{}{\Delta; P \overset{s}{\Longrightarrow} \cdot; P}\mathrm{init} \qquad \text{no rule for } \Delta; P \overset{s}{\Longrightarrow} \cdot; Q \text{ for } P \neq Q$$

Note that the second premise of the $\supset$L rule is still a focused sequent. From a practical point of view it is important to continue with the focusing steps in the first premise before attempting to prove the second premise, because the decomposition of B may ultimately fail when an atomic proposition is reached. Such a failure would render the possibly difficult proof of A useless.

There is a slight, but important asymmetry in the initial sequents: we require that we have focused on the left proposition.

If one shows only applications of the decision rules in a derivation, the format is very close to *assertion-level proofs* as proposed by Huang [Hua94]. His motivation was the development of a formalism appropriate for the presentation of mathematical proofs in a human-readable form. This provides independent evidence for the value of focusing proofs. Focusing derivations themselves were developed by Andreoli [And92] in the context of classical linear logic. An adaptation to intuitionistic linear logic was given by Howe [How98] which is related the calculus LJT devised by Herbelin [Her95]. Herbelin's goal was to devise a sequent calculus whose derivations are in bijective correspondence to normal natural deductions. Due to the $\vee$, $\perp$ and $\exists$ elimination rules, this is not the case here.

The search procedure which works with focusing sequents is similar to the one for inversion. After the detailed development of inversion proofs, we will not repeat or extend the development here, but refer the interested reader to the literature. The techniques are very similar to the ones shown in Section 4.1.

4.4 Unification

When proving a proposition of the form $\exists x.\ A$ by its right rule in the sequent or focusing calculus, we must supply a term t and then prove $[t/x]A$. The domain of quantification may include infinitely many terms (such as the natural numbers), so this choice cannot be resolved simply by trying all possible terms t. Similarly, when we use a hypothesis of the form $\forall x.\ A$ we must supply a term t to substitute for x. We refer to this a *existential non-determinism*.

Fortunately, there is a technique called *unification* which is sound and complete for syntactic equality between terms. The basic idea is quite simple: we postpone the choice of t and instead substitute a new *existential variable* (often called *meta-variable* or *logic variable*) X for x and continue with the bottom-up construction of a derivation. When we reach initial sequents we check if there is a substitution for the existential variables such that the hypothesis matches the conclusion. If so, we apply this instantiation globally to the partial derivation and continue to search for proofs of other subgoals. Finding an instantiation for existential variables under which two propositions or terms match is called *unification*. It is decidable if a unifying substitution or *unifier* exists, and if so, we can effectively compute it in linear time. Moreover, we can do so with a minimal commitment and we do not need to choose between various possible unifiers.

Because of its central importance in both backward- and forward-directed search, unification has been thoroughly investigated. Herbrand [Her30] is given credit for the first description of a unification algorithm in a footnote of his thesis, but it was not until 1965 that it was introduced into automated deduction through the seminal work by Alan Robinson [Rob65, Rob71]. The first algorithms were exponential, and later almost linear [Hue76, MM82] and linear algorithms [MM76, PW78] were discovered. In the practice of theorem proving, generally variants of Robinson's algorithm are still used, due to its low constant overhead on the kind of problems encountered in practice. For further discussion and a survey of unification, see [Kni89]. We describe a variant of Robinson's algorithm.

Before we describe the unification algorithm itself, we relate it to the problem of proof search. We use here the sequent calculus with atomic initial sequents, but it should be clear that precisely the same technique of *residuation* applies to focused derivations. We enrich the judgment $\Gamma \stackrel{-}{\Longrightarrow} A$ by a *residual proposition*

F such that

1. if $\Gamma \stackrel{-}{\Longrightarrow} A$ then $\Gamma \stackrel{-}{\Longrightarrow} A \setminus F$ and F is true, and

2. if $\Gamma \stackrel{-}{\Longrightarrow} A \setminus F$ and F is true then $\Gamma \stackrel{-}{\Longrightarrow} A$.

Generally, we cannot prove such properties directly by induction, but we need to generalize them, exhibiting the close relationship between the derivations of the sequents and residual formulas F.

Residual formulas F are amenable to specialized procedures such as unification, since they are drawn from a simpler logic or deductive system than the general propositions A. In practice they are often solved *incrementally* rather than collected throughout a derivation and only solved at the end. This is important for the early detection of failures during proof search. Incremental solution of residual formulas is the topic of Exercise **??**.

What do we need in the residual propositions so that existential choices and equalities between atomic propositions can be expressed? The basic proposition is one of equality between atomic propositions, $P_1 \doteq P_2$. We also have conjunction $F_1 \wedge F_2$, since equalities may be collected from several subgoals, and $\top$ if there are no residual propositions to be proven. Finally, we need the existential quantifier $\exists x.\, F$ to express the scope of existential variables, and $\forall x.\, F$ to express the scope of parameters introduced in a derivation. We add equality between terms, since it is required to describe the unification algorithm itself. We refer to the logic with these connectives as *unification logic*, defined via a deductive system.

$$\textit{Formulas} \quad F \quad ::= \quad P_1 \doteq P_2 \mid t_1 \doteq t_2 \mid F_1 \wedge F_2 \mid \top \mid \exists x.\, F \mid \forall x.\, F$$

The main judgment "F *is valid*", written $\models F$, is defined by the following rules, which are consistent with, but more specialized than the rules for these connectives in intuitionistic natural deduction (see Exercise **??**).

$$\frac{}{\models P \doteq P}\,{\doteq}\mathrm{I} \qquad\qquad \frac{}{\models t \doteq t}\,{\doteq}\mathrm{I}'$$

$$\frac{\models F_1 \qquad \models F_2}{\models F_1 \wedge F_2}\,\wedge\mathrm{I} \qquad\qquad \frac{}{\models \top}\,\top\mathrm{I}$$

$$\frac{\models [t/x]F}{\models \exists x.\, F}\,\exists\mathrm{I} \qquad\qquad \frac{\models [a/x]F}{\models \forall x.\, F}\,\forall\mathrm{I}^a$$

The $\forall\mathrm{I}^a$ rule is subject to the usual proviso that a is a new parameter not occurring in $\forall x.\, F$. There are no elimination rules, since we do not need to consider hypotheses about the validity of a formula F which is the primary reason for the simplicity of theorem proving in the unification logic.

We enrich the sequent calculus with residual formulas from the unification logic, postponing all existential choices. Recall that in practice we merge residuation and solution in order to discover unprovable residual formulas as soon as possible. This merging of the phases is not represented in our system.

Initial Sequents. Initial sequents residuate an equality between its principal propositions. Any solution to the equation will unify P' and P, which means that this will translate to a correct application of the initial sequent rule in the original system.

$$\frac{}{\Gamma, P' \overset{-}{\Longrightarrow} P \setminus P' \doteq P}\,\text{init}$$

Propositional Connectives. We just give a few sample rules for the connectives which do not involve quantifiers, since all of them simply propagate or combine unification formulas, regardless whether they are additive, multiplicative, or exponential.

$$\frac{\Gamma, A \overset{-}{\Longrightarrow} B \setminus F}{\Gamma \overset{-}{\Longrightarrow} A \supset B \setminus F}\,{\supset}\text{R} \qquad \frac{}{\Gamma \overset{-}{\Longrightarrow} \top \setminus \top}\,\top\text{R}$$

$$\frac{\Gamma, A \supset B \overset{-}{\Longrightarrow} A \setminus F_1 \qquad \Gamma, A \supset B, B \overset{-}{\Longrightarrow} C \setminus F_2}{\Gamma, A \supset B \overset{-}{\Longrightarrow} C \setminus F_1 \wedge F_2}\,{\supset}\text{L}$$

Quantifiers. These are the critical rules. Since we residuate the existential choices entirely, the $\exists$R and $\forall$L rules instantiate a quantifier by a new *parameter*, which is existentially quantified in the residual formula in both cases. Similarly, the $\forall$R and $\exists$L rule introduce a parameter which is universally quantified in the residual formula.

$$\frac{\Gamma \overset{-}{\Longrightarrow} [a/x]A \setminus [a/x]F}{\Gamma \overset{-}{\Longrightarrow} \forall x.\ A \setminus \forall x.\ F}\,\forall\text{R}^a \qquad \frac{\Gamma, \forall x.\ A, [a/x]A \overset{-}{\Longrightarrow} C \setminus [a/x]F}{\Gamma, \forall x.\ A \overset{-}{\Longrightarrow} C \setminus \exists x.\ F}\,\forall\text{L}^a$$

$$\frac{\Gamma \overset{-}{\Longrightarrow} [a/x]A \setminus [a/x]F}{\Gamma \overset{-}{\Longrightarrow} \exists x.\ A \setminus \exists x.\ F}\,\exists\text{R}^a \qquad \frac{\Gamma, \exists x.\ A, [a/x]A \overset{-}{\Longrightarrow} C \setminus [a/x]F}{\Gamma, \exists x.\ A \overset{-}{\Longrightarrow} C \setminus \forall x.\ A}\,\exists\text{L}^a$$

The soundness of residuating equalities and existential choices in this manner is straightforward.

Theorem 4.11 (Soundness of Equality Residuation)
If $\Gamma \overset{-}{\Longrightarrow} A \setminus F$ *and* $\models F$ *then* $\Gamma \overset{-}{\Longrightarrow} A$.

Proof: By induction on the structure of the given derivation $\mathcal{R}$. We show the critical cases. Note how in the case of the $\exists$R rule the derivation of $\models \exists x.\ F$ provides the essential witness term t.

Case:

$$\mathcal{R} = \frac{}{\Gamma, P' \overset{=}{\Longrightarrow} P \setminus P' \doteq P}\ \text{init}$$

$\models P' \doteq P$	By assumption
$P' = P$	By inversion
$\Gamma, P' \overset{=}{\Longrightarrow} P$	By rule init

Case:

$$\mathcal{R} = \frac{\begin{array}{c}\mathcal{R}_1 \\ \Gamma \overset{=}{\Longrightarrow} [a/x]A_1 \setminus [a/x]F_1\end{array}}{\Gamma \overset{=}{\Longrightarrow} \exists x.\ A_1 \setminus \exists x.\ F_1}\ \exists\text{R}^a$$

$\models \exists x.\ F_1$	By assumption
$\models [t/x]F_1$ for some t	By inversion
$\Gamma \overset{=}{\Longrightarrow} [t/x]A_1 \setminus [t/x]F_1$	By substitution for parameter a
$\Gamma \overset{=}{\Longrightarrow} [t/x]A_1$	By i.h.
$\Gamma \overset{=}{\Longrightarrow} \exists x.\ A_1$	By rule $\exists$R

Case:

$$\mathcal{R} = \frac{\begin{array}{c}\mathcal{R}_1 \\ \Gamma \overset{=}{\Longrightarrow} [a/x]A_1 \setminus [a/x]F_1\end{array}}{\Gamma \overset{=}{\Longrightarrow} \forall x.\ A_1 \setminus \forall x.\ F_1}\ \forall\text{R}^a$$

$\models \forall x.\ F_1$	By assumption
$\models [b/x]F_1$ for a new parameter b	By inversion
$\models [a/x]F_1$	By substititution of a for b
$\Gamma \overset{=}{\Longrightarrow} [a/x]A_1$	By i.h.
$\Gamma \overset{=}{\Longrightarrow} \forall x.\ A_1$	By rule $\forall$R

$\square$

The opposite direction is more difficult. The desired theorem:

If $\Gamma \overset{=}{\Longrightarrow} A$ *then* $\Gamma \overset{=}{\Longrightarrow} A \setminus F$ *for some* F *with* $\models F$

cannot be proved directly by induction, since the premisses of the two derivations are different in the $\exists$R and $\forall$L rules. However, one can be obtained from the other by substituting terms for parameters. Since this must be done simultaneously, we introduce a new notation.

$$\textit{Parameter Substitution} \quad \rho \quad ::= \quad \cdot \mid \rho, t/a$$

We assume all the parameters a substituted for by ρ are distinct to avoid ambiguity. We write $A[\rho]$, $F[\rho]$, and $\Gamma[\rho]$, for the result of applying the substitution ρ to a proposition, formula, or context, respectively.

Lemma 4.12 *If* $\Gamma \stackrel{=}{\Longrightarrow} A$ *where* $A = A'[\rho]$, $\Gamma = \Gamma'[\rho]$ *then* $\Gamma' \stackrel{=}{\Longrightarrow} A' \setminus F$ *for some* F *such that* $\models F[\rho]$.

Proof: The proof proceeds by induction on the structure of the given derivation $\mathcal{D}$. We show only two cases, the second of which required the generalization of the induction hypothesis.

Case:

$$\mathcal{D} = \cfrac{}{\Gamma_1, P \stackrel{=}{\Longrightarrow} P}\,\text{init}$$

$\Gamma_1 = \Gamma_1'[\rho]$, $P = P'[\rho]$, and $P = P''[\rho]$	Assumption
$\Gamma_1', P' \stackrel{=}{\Longrightarrow} P'' \setminus P' \doteq P''$	By rule init
$\models P'[\rho] \doteq P''[\rho]$	By rule $\doteq$I

Case:

$$\mathcal{D} = \cfrac{\begin{array}{c}\mathcal{D}_1\\ \Gamma \stackrel{=}{\Longrightarrow} [t/x]A_1\end{array}}{\Gamma \stackrel{=}{\Longrightarrow} \exists x.\ A_1}\,\exists\text{R}$$

$\exists x.\ A_1 = A'[\rho]$	Assumption
$A' = \exists x.\ A_1'$ for a new parameter a with	
$[a/x]A_1 = ([a/x]A_1')[\rho, a/a]$	By definition of substitution
$[t/x]A_1 = ([a/x]A_1')[\rho, t/a]$	By substitution for parameter a
$\Gamma = \Gamma'[\rho]$	Assumption
$\Gamma'[\rho] = \Gamma'[\rho, t/a]$	Since a is new
$\Gamma' \stackrel{=}{\Longrightarrow} [a/x]A_1' \setminus [a/x]F_1$, and	
$\models ([a/x]F_1)[\rho, t/a]$	By i.h.
$\Gamma' \stackrel{=}{\Longrightarrow} \exists x.\ A_1' \setminus \exists x.\ F_1$	By rule $\exists$R
$\models (\exists x.\ F_1)[\rho]$	By rule $\exists$R and definition of substitution

□

Theorem 4.13 (Completeness of Equality Residuation)
If $\Gamma \stackrel{=}{\Longrightarrow} A$ *then* $\Gamma \stackrel{=}{\Longrightarrow} A \setminus F$ *for some* F *and* $\models F$.

Proof: From Lemma 4.12 with $A' = A$, $\Gamma' = \Gamma$, and ρ the identity substitution on the parameters in Γ and A. $\square$

Next we describe an algorithm for proving residuated formulas, that is, an algorithm for unification. We do this in two steps: first we solve the problem in the fragment without parameters and universal quantifiers and then we extend the solution to the general case.

There are numerous ways of describing unification algorithsm in the literature. We view it here as a process of transformation on a collection of constraints. In the first instance we consider global unification, where we are given a single constraint formula (as generated by equality residuation, for example) and we have to determine if it is true. Later, we will generalize the view in order just partially transform the constraints to a normal form which is easily seen to have most general solutions. This latter view will be particularly useful when constraints are generated incrementally during proof search.

A collection of equational constraints is simply a collection of formulas in the unification logic or an indication that the constraints are inconsistent (#).

$$\text{Constraints} \quad C \quad ::= \quad \cdot \mid F, C \mid \#$$

We will freely exchange formulas among the constraints, just as we freely exchange assumptions in the sequent calculus. The empty constraint "·" corresponds to success, a contradiction to failure of proving the unification formula. Constraints may contain free *unification variables* X which are interpreted existentially. They are also known as *existential variables* or *logic variables*. Note that unification variables are never bound. We group the rules into several classes. The first, breaks down the structure of the formulas in C.

$$\begin{array}{lcll}
F_1 \wedge F_2, C & \mapsto & F_1, F_2, C & \\
\top, C & \mapsto & C & \\
\exists x.\, F, C & \mapsto & [X/x]F, C & \text{where } X \text{ not free in } F \text{ or } C
\end{array}$$

The second group of rules breaks down equalities into simpler equalities.

$$\begin{array}{lcll}
p(t_1, \ldots, t_n) \doteq p(s_1 \ldots, s_n), C & \mapsto & t_1 \doteq s_1, \ldots, t_n \doteq s_n, C & \\
f(t_1, \ldots, t_n) \doteq f(s_1 \ldots, s_n), C & \mapsto & t_1 \doteq s_1, \ldots, t_n \doteq s_n, C & \\
p(t_1, \ldots, t_n) \doteq q(s_1 \ldots, s_n), C & \mapsto & \# & \text{where } p \neq q \\
f(t_1, \ldots, t_n) \doteq g(s_1 \ldots, s_n), C & \mapsto & \# & \text{where } f \neq g
\end{array}$$

Note that equations of predicate or function symbols without arguments ($n = 0$) will either be simply removed or be inconsistent.

Finally, we will be left with equations where one of the two sides is a unification variable (we are not yet considering parameters). In that case, we must consider the right-hand side and distinguish several cases:

$$\begin{array}{lcll}
X \doteq X, C & \mapsto & C & \\
X \doteq t, C & \mapsto & [t/X]C & \text{provided } X \text{ not free in } t \\
t \doteq X, C & \mapsto & [t/X]C & \text{provided } X \text{ not free in } t \\
X \doteq t, C & \mapsto & \# & \text{if } t \neq X \text{ and } X \text{ free in } t \\
t \doteq X, C & \mapsto & \# & \text{if } t \neq X \text{ and } X \text{ free in } t
\end{array}$$

The conditions on these rules are necessary in order to recognize cases such as $X \doteq f(X)$, which has no solution: No matter which term we substitute for X, the right-hand side will always have one more function symbol than the left-hand side, so the equation cannot be satisfied. We refer to the condition "*X not free in t*" as the *occurs-check*.

Note that the whole algorithm depends critically on the function symbols being uninterpreted. As a trivial example, consider $+(3,4) \doteq +(2,5)$ on which the above algorithm would fail. Slighly trickier is something like $X \doteq -(-(X))$ which is true for any integer X, but violates the occurs-check.

As a first step in the correctness proof we can verify that a unification will always terminate.

Lemma 4.14 (Termination of Unification) *Any sequence of reductions $C \mapsto C' \mapsto C'' \ldots$ must terminate and yield either $\#$ or the empty set of constraints $(\cdot)$.*

Proof: By nested induction, first on the number of variables (unification variables X or bound variables $\exists x$) in C, second on the total size of the constraint, counting quantifiers, logical connectives, and variables occurrences.

The first set of rules for structural decomposition and the rule for eliminating $X \doteq X$ decreases the size of the constraints, without increasing the number of variables. The set of rules for variables (except for $X \doteq X$) reduces the number of variables in C by substitution for all occurrences of a variable (possibly increasing the total size of the constraint). □

In order to show the correctness of the unification algorithm, we would like to show that each step preserves provability. That is, if $C \mapsto C'$ then C is provable iff C' is provable. However, a difficulty arises in the case of existential quantification, since we step from $\exists x.\ F$ to $[X/x]F$ and we have not defined what it means for a formula with a unification variable to be provable. Intuitively, it should mean that not the formula itself, but some instance of it is provable. Hence we define that a constraint is *satisfiable* to mean that there is an instance that is provable. In order to define the concept of an *instance* we define simultaneous substitution for the unification variables of a term.

The second concept we need is that of a *substitution* for existential variables. We use a new notation, because this form of substitution is quite different from substitutions for bound variables x or parameters a.

$$\textit{Substitutions} \quad \theta \quad ::= \quad \cdot \mid \theta, t/X$$

We require that all variables X defined by a substitution are distinct. We write $\mathrm{dom}(\theta)$ for the variables defined by a substitution and $\mathrm{cod}(\theta)$ for all the variables occuring in the terms t. For a ground substitution $\mathrm{cod}(\theta)$ is empty. For the technical development it is convenient to assume that the domain and co-domain of a substitution share no variables. This rules out "circular" substitutions such as $f(X)/X$ and it also disallows identity substitutions X/X. The latter restriction can be dropped, but it does no harm and is closer to the implementation.

As for contexts, we consider the order of the definitions in a substitution to be irrelevant.

We write $t[\theta]$, $A[\theta]$, and $\Gamma[\theta]$ for the application of a substitution to a term, proposition, or context. This is defined to be the identity on existential variables that are not explicitly defined in the substitution.

We also need an operation of composition, written as $\theta_1 \circ \theta_2$ with the property that $t[\theta_1 \circ \theta_2] = (t[\theta_1])[\theta_2]$ and similarly for propositions and contexts. Composition is defined by

$$\begin{aligned} (\cdot) \circ \theta_2 &= \theta_2 \\ (\theta_1, t/X) \circ \theta_2 &= (\theta_1 \circ \theta_2), t[\theta_2]/X \end{aligned}$$

In order for composition to be well-defined and have the desired properties we require that $\mathrm{dom}(\theta_1)$, $\mathrm{dom}(\theta_2)$ and $\mathrm{cod}(\theta_2)$ are disjoint, but of course variables in the co-domain of θ_1 can be defined by θ_2.

Now we define that *constraint* $C = F_1, \ldots, F_n$ *is satisfiable* if there exists a substitution θ for unification variables in C such that $\models F_i[\theta]$ for all $1 \leq i \leq n$. We write C *sat* if C is satisfiable.

Theorem 4.15 (Preservation of Satisfiability)
If $C \mapsto C'$ *then* C *sat iff* C' *sat*

Proof: In both directions, the proof is by cases on the definition of $C \mapsto C'$. We show a three cases from left-to-right. The other cases and opposite direction are similar.

Assume $C \mapsto C'$ and C *sat*. We have to show the C' *sat*.

Case: $\exists x.\ F, C_1 \mapsto [X/x]F, C_1$.

$\exists x.\ F, C_1$ *sat*	Assumption
For some θ, $\models (\exists x.\ F)[\theta]$	
and $\models F_1[\theta]$ for every F_1 in C_1	By defn. of *sat*
$\models \exists x.\ (F[\theta])$	By defn. of substitution
$\models [t/x](F[\theta])$	By inversion
$\models ([t/x]F)[\theta]$	By props. of substitution
$\models ([X/x]F)[\theta, t/X]$	Since X not in F or t
$\models F_1[\theta, t/X]$ for any F_1 in C_1	Since X not in C_1
$[X/x]F, C_1$ *sat*	By defn. of *sat*

Case: $X \doteq t, C_1 \mapsto [t/X]C_1$ where X not in t.

$X \doteq t, C_1$ *sat*	Assumption
For some θ, $\models (X \doteq t)[\theta]$	
and $\models F_1[\theta]$ for every F_1 in C_1	By defn. of *sat*
$\models X[\theta] \doteq t[\theta]$	By defn. of substitution
$X[\theta] = t[\theta]$	By inversion
$\theta = (\theta', t[\theta]/X)$	By defn. of substitution

$t[\theta]/X = t[\theta']/X$	Since X not in t
$\models F_1[\theta', t[\theta']/X]$ for any F_1 in C_1	From above
$\models ([t/X]F_1)[\theta']$	By props. of substitution
$[t/X]C_1$ *sat*	By defn. of *sat*

Case: $X \doteq t, C_1 \mapsto \#$ where X in t, $X \neq t$.

$X \doteq t, C_1$ *sat*	Assumption
$\models (X \doteq t)[\theta]$ for some θ	By defn. of *sat*
$\models X[\theta] \doteq t[\theta]$	By defn. of substitution
$X[\theta] = t[\theta]$	By inversion
$X[\theta] = f(\ldots X \ldots)[\theta]$	Since X in t, $X \neq t$
$X[\theta] = f(\ldots X[\theta] \ldots)$	By defn. of substitution
Contradiction	Right-hand side has more function symbols than left-hand side

This case is impossible

□

The argument above requires some elementary reasoning about substitution. Those proofs are usually straightforward by induction on the structure of the term we substitute in, as long as the right condition on occurrences of variables are known.

Termination of unification together with preservation of satisfiability gives us the correctness of unification as a procedure.

4.5 Unification with Parameters

The generalization of the algorithm above to account for universal quantifiers and parameters is not completely straightforward. The difficulty is that $\forall x.\ \exists y.\ y \doteq x$ is valid, while $\exists y.\ \forall x.\ y \doteq x$ is not. In unification logic, the fact that the second cannot be derived is due to the parameter restriction.

$$\dfrac{\dfrac{\dfrac{}{\models a \doteq a}\doteq\mathrm{I}}{\models \forall x.\ a \doteq x}\forall\mathrm{I}^a??}{\models \exists y.\ \forall x.\ y \doteq x}\exists\mathrm{I}$$

In this derivation, the application of $\forall\mathrm{I}^a$ is incorrect. However, if we had a way to postpone choosing the instantiation for y, say, by supplying an existential variable instead, then the situation is far less clear.

$$\dfrac{\dfrac{\dfrac{\text{“}a/Y\text{”}??}{\models Y \doteq a}\doteq\mathrm{I}}{\models \forall x.\ Y \doteq x}\forall\mathrm{I}^a??}{\models \exists y.\ \forall x.\ y \doteq x}\exists\mathrm{I}$$

In this derivation, it is the substitution of a for Y which will invalidate the derivation at the $\forall\mathrm{I}^a$ rule application. Up to that point we could not really fail. Written in our transformation notation:

$$\begin{array}{ll} & \exists y.\, \forall x.\, y \doteq x \\ \mapsto & \forall x.\, Y \doteq x \\ \mapsto & Y \doteq a \\ \mapsto^{??} & \cdot \end{array}$$

From this very simple example it seems clear that we need to prohibit final step: Y may not be instantiated with a term that mentions parameter a. There are two approaches to encoding this restriction. More or less standard in theorem proving is *Skolemization* which we pursue in Exercise 4.3. The dual solution notes for each existential variable which parameters may occur in its substitution term. In the example above, Y was introduced at a point where a did not yet occur, so the substitution of a for Y should be rejected.

In order to describe this concisely, we add a *parameter context* Ψ to the judgment which lists distinct parameters.

$$\textit{Parameter Context} \quad \Psi \quad ::= \quad \cdot \mid \Psi, a$$

We annotate each judgment with the parameter context and introduce the new judgment "*t is closed with respect to Ψ*", written as $\Psi \models t$ *term*. It is defined by the following rules.

$$\frac{}{\Psi_1, a, \Psi_2 \vdash a\ \mathit{term}}\,\mathrm{parm} \qquad \frac{\Psi \vdash t_1\ \mathit{term} \ \cdots\ \Psi \vdash t_n\ \mathit{term}}{\Psi \vdash f(t_1, \ldots, t_n)\ \mathit{term}}\,\mathrm{root}$$

We modify the validity judgment for unification formulas to guarantee this condition.

$$\frac{\Psi \vdash t\ \mathit{term} \qquad \Psi \models [t/x]F}{\Psi \models \exists x.\, F}\,\exists\mathrm{I} \qquad \frac{\Psi, a \models [a/x]F}{\Psi \models \forall x.\, F}\,\forall\mathrm{I}^a$$

Now the state of the unification algorithm (that is, the current set of constraints) must record the parameter context. We write this as $\Psi \rhd C$. Ψ is simply carried along from left to right in most transformations.

$$\begin{array}{lcll}
(\Psi \rhd F_1 \wedge F_2, C) & \mapsto & (\Psi \rhd F_1, F_2, C) & \\
(\Psi \rhd \top, C) & \mapsto & (\Psi \rhd C) & \\
(\Psi \rhd f(t_1, \ldots, t_n) \doteq f(s_1 \ldots, s_n), C) & \mapsto & (\Psi \rhd t_1 \doteq s_1, \ldots, t_n \doteq s_n, C) & \\
(\Psi \rhd f(t_1, \ldots, t_n) \doteq g(s_1 \ldots, s_n), C) & \mapsto & (\Psi \rhd \#) & \text{where } f \neq g \\
(\Psi \rhd a \doteq a, C) & \mapsto & (\Psi \rhd C) & \\
(\Psi \rhd a \doteq b, C) & \mapsto & (\Psi \rhd \#) & \text{where } a \neq b \\
(\Psi \rhd a \doteq f(t_1, \ldots, t_n)) & \mapsto & (\Psi \rhd \#) & \\
(\Psi \rhd f(t_1, \ldots, t_n) \doteq a) & \mapsto & (\Psi \rhd \#) &
\end{array}$$

The notion of an existential variable must now be generalized to track the set of parameters its substituend may depend on. We write X_Δ for a unification variable X that may depend on all the parameters in Δ, but no others. All occurrences of a variable X must be annotated with the same Δ—we think of Δ as an intrinsic property of X.

$$\begin{array}{lcll}(\Psi \rhd \forall x.\ F, C) & \mapsto & (\Psi, a \rhd [a/x]F, C) & \text{where } a \text{ not in } \Psi,\ F, \text{ or } C\\ (\Psi \rhd \exists x.\ F, C) & \mapsto & (\Psi \rhd [X_\Psi/x]F, C) & \text{where } X \text{ not free in } F \text{ or } C\end{array}$$

An equation $X_\Psi \doteq t$ could now be solved immediately, if all parameters of t are contained in Ψ and X does not occur in t. A first attempt at such a rule would be

$$(\Psi \rhd X_\Delta \doteq t, C) \quad \mapsto \quad (\Psi \rhd [t/X]C) \quad \text{where } \Delta \vdash t\ \mathit{term} \text{ and } X \text{ not free in } t$$

However, in general t will not be closed so we cannot prove that $\Delta \vdash t\ \mathit{term}$. For example, consider the constraint

$$a \rhd X_{\cdot} \doteq f(Y_a) \wedge Y_a \doteq a$$

where X cannot depend on any parameters and Y can depend on a. This should have no solution, since $X_{\cdot}$ would have to be equal to $f(a)$, which is not permissible. On the other hand,

$$a \rhd X_{\cdot} \doteq f(Y_a) \wedge Y_a \doteq c$$

for a constant c has a solution where Y_a is c and $X_{\cdot}$ is $f(c)$. So when we process an equation $X_\Delta = t$ we need to restrict any variable in t so it can depend only on the parameters in Δ. In the example above, we would substitute $Y'_{\cdot}$ for Y_a.

In order to describe this restriction, we introduce a new form of constraints which expresses the judgment $\Delta \vdash t\ \mathit{term}$ in the presence of unification variables. We write it as $t\mid_\Delta$, thinking of it as the restriction of t to Δ. It is implemented by the following transformations.

$$\begin{array}{lcll}(\Psi \rhd f(t_1, \ldots, t_n)\mid_\Delta, C) & \mapsto & (\Psi \rhd t_1\mid_\Delta, \ldots, t_n\mid_\Delta, C) & \\ (\Psi \rhd a\mid_\Delta, C) & \mapsto & (\Psi \rhd C) & \text{if } a \in \Delta\\ (\Psi \rhd a\mid_\Delta, C) & \mapsto & (\Psi \rhd \#) & \text{if } a \notin \Delta\\ (\Psi \rhd Y_{\Delta'}\mid_\Delta, C) & \mapsto & (\Psi \rhd [Y_{\Delta' \cap \Delta}/Y]C) & \end{array}$$

the collection of the above four rules implement a process called *pruning*. Now we can finally write down the correct rule for existential variables.

$$(\Psi \rhd X_\Delta \doteq t, C) \quad \mapsto \quad (\Psi \rhd t\mid_\Delta, [t/X]C) \quad \text{provided } X \text{ not free in } t$$

From an implementation point of view, it makes sense to first solve $t\mid_\Delta$ before substitution t for X. In fact, it is probably beneficial to combine it with the occurs-check to the term t need only be traversed once.

The soundness and completeness theorems from above extend to the problem with parameters, but become more difficult. The principal new notion we need is an *admissible substitution* θ which has the property that for every existential variable X_Δ we have $\Delta \vdash X[\theta]\ \mathit{term}$ (see Exercise 4.4).

The ML implementation takes advantage of the fact that whenever a variable must be restricted, one of the two contexts is a prefix of the other. This is because every equation in a formula F lies beneath a path of possibly alternating quantifiers, a so-called *mixed quantifier prefix*. When we apply the rules above algorithmically, we instantiate each existentially quantified variable with a new free existential variable which depends on all parameters which were introduced for the universally quantified variables to its left. Clearly, then, for any two variables in the same equation, one context is a prefix of the other. Our ML implementation does take advantage of this observation by simplifying the intersection operation.

We can take this optimization a step further and only record with an integer (a kind of time stamp), which parameters an existential variable may depend on. This improves the efficiency of the algorithm even further, since we only need to calculate the minimum of two integers instead of intersecting two contexts during restriction. In the ML code for this class, we did not optimize to this extent.

4.6 Exercises

Exercise 4.1 Give an alternative proof of the inversion properties (Theorem 4.1) which does not use induction, but instead relies on admissibility of cut in the sequent calculus (Theorem 3.11).

Exercise 4.2 Formulate one or several cut rules directly on inversion sequents as presented in Section 4.1 and prove that they are admissible. Does this simplify the development of the completeness result for inversion proofs? Show how admissibility might be used, or illustrate why it is not much help.

Exercise 4.3 An alternative to indexing unification variables with the parameters they may depend on is *Skolemization*. Instead of changing the notion of unification variable, we change the notion of parameter, replacing it by a so-called *Skolem function*. The two quantifier rules become

$$
\begin{array}{lcll}
\forall x.\, F, C & \mapsto & [f(X_1, \ldots, X_n)/x]F, C & \text{where } f \text{ not in } F, \text{ or } C, \text{ and } X_1, \ldots, X_n \\
 & & & \text{are all free unification variables in } F \\
\exists x.\, F, C & \mapsto & [X/x]F, C & \text{where } X \text{ not free in } F \text{ or } C
\end{array}
$$

Now, incorrect dependencies are avoided due to the occurs-check. Reconsider our simple example:

$$
\begin{array}{ll}
 & \exists y.\, \forall x.\, y \doteq x \\
\mapsto & \forall x.\, Y \doteq x \\
\mapsto & Y \doteq f(Y) \\
\mapsto & \#
\end{array}
$$

Skolemization is attractive because it allows us to use a simpler algorithm for unification. Moreover, in some logics such as classical logic it can be applied statically, before we ever attempt to prove the proposition, completely eliminating parameters from consideration. On the other hand, Skolemization is unsound in some higher-order logics. Also, it is more difficult to recover a proof of proposition if we Skolemize during search.

Prove the correctness of the unification algorithm for the full unification logic (including universal quantifiers) which employs Skolemization.

Exercise 4.4 Extend the proofs of termination and preservation of satisfiability from the purely existential case in Section 4.4 to allow for the presence of parameters as sketched in Section 4.5. An important concept will likely be that of *admissible substitution* θ which has the property that for every existential variable X_Δ we have $\Delta \vdash X[\theta]$ *term*. You should be careful to make a precise connection between the constraint $t \mid_\Delta$ and the judgment $\Delta \vdash t$ *term* (where the latter is not defined for unification variables).