First Order Predicate Logic

Reasoning in Predicate Logic

Wolfgang Schreiner and Wolfgang Windsteiger
Wolfgang.(Schreiner|Windsteiger)@risc.jku.at

Research Institute for Symbolic Computation (RISC)
Johannes Kepler University (JKU), Linz, Austria
http://www.risc.jku.at

How does Mathematics "Work"?

Mathematics = "study of mathematical theories"

Math. theory = "collection of statements that follow from axioms"

Axiom = statement that is assumed to be true

Workflow:

1. Characterize objects of interest by distinguishing properties ⇝ axioms.
2. Investigate what must hold under these circumstances ⇝ theorems.
   2.1 Investigate what might hold ⇝ conjectures.
   2.2 Justify conjectures ⇝ proof.
   A proof turns a conjecture into a theorem.

Example: Natural Numbers with Addition

What characterizes the natural numbers with addition?

1. Objects of interest: 0, s, $+$. We write $n+1$ instead of $s(n)$.
2. No natural number has 0 as its successor:
$$\forall n : \neg(s(n) = 0). \tag{P1}$$
3. Numbers with identical successor are identical:
$$\forall m, n : s(m) = s(n) \rightarrow m = n. \tag{P2}$$
4. Adding 0 from right is neutral:
$$\forall n : n + 0 = n. \tag{P3}$$
5. Adding successor gives successor:
$$\forall m, n : n + (m+1) = (n+m) + 1. \tag{P4}$$
6. If A holds for 0 and always for successors also, then A holds for all n:
$$(A[0/n] \wedge (\forall m : A[m/n] \rightarrow A[m+1/n])) \rightarrow \forall n : A. \tag{P5}$$
Available for every formula A.

Example: Natural Numbers with Addition

1. Observe:

$$
\begin{aligned}
&0+1 = 0+s(0) = s(0+0) = s(0) = 1\\
&0+2 = 0+s(1) = s(0+s(0)) = s(s(0)) = 2\\
&0+3 = 0+s(2) = s(0+s(1)) = s(s(0+s(0))) = s(s(s(0))) = 3\\
&0+4 = 0+s(3) = \ldots = s(s(s(s(0)))) = 4
\end{aligned}
$$

   etc.

2. Conjecture:

$$\forall n : 0+n = n$$

3. Justify: Semantics of $\forall$: check all assignments for n, which would need (in this case) infinitely many checks!
4. Proof: justify statement through a finite sequence of arguments, why the statement must be true.

Formal Reasoning: What Is a Proof?

Forward interpretation:

A proof starts from trivial proof situations (can be proved easily),

progresses step-by-step

until it reaches the final situation, where the goal is proved.

Backward interpretation:

A proof starts from the goal to be proved,

progresses step-by-step

until it reaches trivial proof situations (can be proved easily).

Individual proof steps are guided by inference rules, which are denoted as

$$\text{forward} \downarrow \quad \frac{S_1 \qquad \ldots \qquad S_n}{S} \quad \uparrow \text{backward}$$

Forward interpretation:

If $S_1, \ldots, S_n$ can be proved, then also S can be proved.

Backward interpretation:

In order to prove S, we need to prove $S_1, \ldots, S_n$.

$S_1, \ldots, S_n$, and S: proof situations.

Example

$S, S_1, \ldots, S_6$: sequents. Consider inference rules:

$$R_1\colon \frac{S_2 \qquad S_3}{S_1} \qquad R_2\colon \frac{}{S_4} \qquad R_3\colon \frac{S_1}{S} \qquad R_4\colon \frac{}{S_5}$$

$$R_5\colon \frac{S_4 \qquad S_5}{S_2} \qquad R_6\colon \frac{}{S_6} \qquad R_7\colon \frac{S_6}{S_3}$$

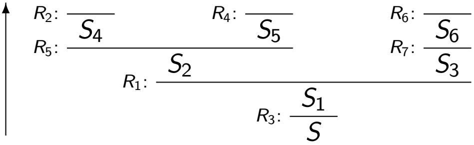
We want to prove S.

$$R_3\colon \cfrac{R_1\colon \cfrac{R_5\colon \cfrac{R_2\colon \cfrac{}{S_4} \qquad R_4\colon \cfrac{}{S_5}}{S_2} \qquad R_7\colon \cfrac{R_6\colon \cfrac{}{S_6}}{S_3}}{S_1}}{S}$$

Proof Generation vs. Proof Presentation

Proof generation: start with sequent to be proved, then work backwards.

Read and apply rules from bottom to top.



Backward style proof presentation: In order to prove S we have to prove, by R_3, S_1. For this, by R_1, we have to

1. prove S_2: by R_5 we have to prove S_4 and S_5, which are guaranteed by R_2 and R_4, respectively. Now we still have to
2. prove S_3: by R_7 it is sufficient to prove S_6, which we know from R_6. q.e.d.

Proof Generation vs. Proof Presentation

Proof presentation: often done in forward reasoning style, i.e. start with known facts and work forward until the sequent to be proved is reached.

Read and apply rules from top to bottom.

$$R_3:\ \dfrac{R_1:\ \dfrac{R_5:\ \dfrac{R_2:\ \dfrac{}{S_4} \qquad R_4:\ \dfrac{}{S_5}}{S_2} \qquad R_7:\ \dfrac{R_6:\ \dfrac{}{S_6}}{S_3}}{S_1}}{S}$$

Forward style proof presentation: We know S_4 and S_5 can be proved, hence by R_5, S_2 can be proved. Furthermore we know that S_6 can be proved, hence by R_7, also S_3 can be proved. Together with S_2, by R_1, we know that S_1 can be proved, and therefore, by R_3, also S. q.e.d.

Note: proof cannot be generated in this way.

Formal Proofs

A formal proof can be seen as a tree, where

1. every node is a sequent,
2. if $S_1, \ldots, S_n$ are the children nodes of a node S, then there must be an inference rule of the form $\dfrac{S_1 \ \ldots \ S_n}{S}$.

Special case $n = 0$: A leaf has 0 children, hence

for every leaf S in the tree there must be a rule $\dfrac{}{S}$.

A formal proof of S is a formal proof with root S.

A Sketch of a Simple Proof Generation Procedure

Input: S
Output: P s.t. P is a formal proof S.

$P :=$ tree containing only the root node S

$Q := \{S\}$

while Q not empty

choose a rule $\dfrac{S_1 \;\ldots\; S_n}{s}$ such that $s \in Q$

replace s in Q by $S_1, \ldots, S_n$

add $S_1, \ldots, S_n$ as children nodes of s in P

return P

Depending on 1) the rules and 2) the choice of the rule in the loop, the procedure might not terminate or might not give a complete proof.

Inference Rules: A Closer Look

Proof situations are written as sequents of the form $H_1, \ldots, H_k \vdash C$, where

$H_1, \ldots, H_k \vdash C$ intuitively means the goal C follows from the assumptions $\{H_1, \ldots, H_k\}$.

Special case $k = 0$: there are no assumptions!

Proof situation $\vdash C$ means: we have to prove that C is valid.

In the sequel, we describe inference rules as schematic patterns

$$\text{name:} \quad \frac{K_1 \ldots \vdash C_1 \qquad \ldots \qquad K_n \ldots \vdash C_n}{K \ldots \vdash C}$$

where letters stand for individual formulas or terms and
"$K \ldots$" stand for sequences of formulas.

Choice of Inference Rules: A Closer Look

Convention: formula sequences are orderless, i.e.

$$K \ldots, F_1 \wedge F_2 \vdash \neg G$$

expresses that

1. the assumptions contain a formula with outermost symbol "$\wedge$" and
2. the goal is a formula with outermost symbol "$\neg$".

In the "proof generation procedure" above:

$$\text{choose a rule } \frac{S_1 \ \ldots \ S_n}{s} \text{ such that } s \in Q$$

means

$$\text{choose a rule } \frac{S_1 \ \ldots \ S_n}{s} \text{ such that } s \text{ "matches" some } q \in Q.$$

Now $S_1, \ldots, S_n$ actually mean variants of the schematic patterns, where variables are replaced by those parts of s that are fixed by above "matching" (see examples later).

Proof Rules for Predicate Logic

One could give a (minimal) set of inference rules for first order predicate logic, which can be shown to be sound and complete, i.e.

1. every formula, which has a formal proof, is also semantically true and
2. every semantically true formula has a formal proof.

$\rightsquigarrow$ e.g. sequent calculus, Gentzen calculus, natural deduction calculus, etc.

Rather, we want to give proof rules that help in practical proving of mathematical statements and checking of given proofs. Differences lie in details.

We distinguish: structural rules, connective rules and quantifier rules.

For every binary logical connective and every standard quantifier, we give at least one rule, where the connective or quantifier occurs as the outermost symbol in the goal or one of the assumptions.

Structural Rules

- If the goal is among the assumptions, the goal can be proved.

$$\text{GoalAssum: } \frac{}{K \ldots, G \vdash G}$$

- Proof by contradiction:

$$\text{A-}\neg\text{: } \frac{K \ldots, \neg G \vdash \bot}{K \ldots \vdash G} \qquad \text{P-}\neg\text{: } \frac{K \ldots \vdash \neg A}{K \ldots, A \vdash \bot}$$

- Add valid assumption:

$$\text{ValidAssum: } \frac{K \ldots, V \vdash G}{K \ldots \vdash G} \quad \text{if } V \text{ is valid}$$

- Drop any assumption:

$$\text{AnyAssum: } \frac{K \ldots \vdash G}{K \ldots, A \vdash G}$$

- Add proved assumption — the cut-rule:

$$\text{Cut: } \frac{K \ldots \vdash A \qquad K \ldots, A \vdash G}{K \ldots \vdash G}$$

Example

$$\text{A-}\neg: \frac{\sqrt{2} \in \mathbb{Q}, \ldots \vdash \bot}{\ldots \vdash \sqrt{2} \notin \mathbb{Q}}$$

Natural language description of this proof step:

We have to prove that $\sqrt{2}$ is not rational. We do a proof by contradiction, hence, we assume that $\sqrt{2}$ was rational and derive a contradiction.

Connective Rules

- Prove parts of a conjunction separately:

$$\text{P-}\wedge\text{:}\ \frac{K \ldots \vdash F_1 \qquad K \ldots \vdash F_2}{K \ldots \vdash F_1 \wedge F_2}$$

- Split conjunction in assumptions:

$$\text{A-}\wedge\text{:}\ \frac{K \ldots, F_1, F_2 \vdash G}{K \ldots, F_1 \wedge F_2 \vdash G}$$

- Prove disjunction:

$$\text{P-}\vee\text{:}\ \frac{K \ldots, \neg F_1 \vdash F_2}{K \ldots \vdash F_1 \vee F_2}$$

- Disjunction in assumptions $\leadsto$ prove by cases:

$$\text{A-}\vee\text{:}\ \frac{K \ldots, F_1 \vdash G \qquad K \ldots, F_2 \vdash G}{K \ldots, F_1 \vee F_2 \vdash G}$$

Connective Rules

- Prove implication ⇝ assume LHS and prove RHS:

$$\text{P-}\rightarrow\text{:}\ \frac{K\ldots, F_1 \vdash F_2}{K\ldots \vdash F_1 \rightarrow F_2}$$

- Implication in assumptions ⇝ "Modus Ponens" (MP):

$$\text{A-}\rightarrow\text{/MP:}\ \frac{K\ldots, F_1, F_1 \rightarrow F_2, F_2 \vdash G}{K\ldots, F_1, F_1 \rightarrow F_2 \vdash G}$$

  An implication alone in the KB is useless, it needs also the LHS!

- Prove equivalence by proving both directions:

$$\text{P-}\leftrightarrow\text{:}\ \frac{K\ldots \vdash F_1 \rightarrow F_2 \qquad K\ldots \vdash F_2 \rightarrow F_1}{K\ldots \vdash F_1 \leftrightarrow F_2}$$

- Equivalence in assumptions ⇝ substitution:

$$\text{A-}\leftrightarrow\text{:}\ \frac{K\ldots[F_2/F_1], F_1 \leftrightarrow F_2 \vdash G}{K\ldots, F_1 \leftrightarrow F_2 \vdash G} \qquad \text{A-}\leftrightarrow\text{:}\ \frac{K\ldots, F_1 \leftrightarrow F_2 \vdash G[F_2/F_1]}{K\ldots, F_1 \leftrightarrow F_2 \vdash G}$$

  $\phi[F_2/F_1]$: replace some occurrences of (sub-)formula F_1 by formula F_2 in formula or sequence of formulas ϕ.

Example

$$\text{A-}\vee\text{:}\ \frac{\dfrac{P_1}{even(m) \vdash G} \qquad \dfrac{P_2}{odd(m) \vdash G}}{even(m) \vee odd(m) \vdash G}$$

Natural language description of this proof step:

We already know that m is even or m is odd. Thus, we can distinguish the two cases:

1. *m is even: ... (insert proof* P_1 *here)*
2. *m is odd: ... (insert proof* P_2 *here)*

Making our Lives Easier: Derivable Rules

$$
\begin{array}{rc}
 & K \ldots, A, B \vdash G \\ \cline{2-2}
\text{AnyAssum:} & K \ldots, A, A \rightarrow B, B \vdash G \\ \cline{2-2}
\text{MP:} & K \ldots, A, A \rightarrow B \vdash G \\ \cline{2-2}
\text{ValidAssum:} & K \ldots, A \vdash G
\end{array}
$$

if B is a logical consequence of A, i.e. $A \rightarrow B$ is valid

This shows that with a combination of AnyAssum, Modus Ponens, and DropAssum we can always add a logical consequence of an assumption to the knowledge base. We can formulate this as a derivable rule

$$
\text{ConsAssum: } \frac{K \ldots, A, B \vdash G}{K \ldots, A \vdash G}
$$

if B is a logical consequence of A

Making our Lives Easier: Derivable Rules

As soon as we have contradicting assumptions, the proof can be finished:

$$\text{A-}\neg\text{:}\ \dfrac{\text{P-}\neg\text{:}\ \dfrac{\text{GoalAssum:}\ \dfrac{}{K\ldots,\neg A,\neg G \vdash \neg A}}{K\ldots,A,\neg A,\neg G \vdash \bot}}{K\ldots,A,\neg A \vdash G}$$

Derivable rule:

$$\text{ContrAssum:}\ \dfrac{}{K\ldots,A,\neg A \vdash G}$$

Example

Prove $((A \rightarrow (B \vee C)) \wedge \neg C) \rightarrow (A \rightarrow B)$,

where A, B, and C are abbreviations for complex predicate logic formulas.

Develop proof tree top-down with root on top (convenient in practice).

$$
\begin{array}{rc}
 & \vdash ((A \rightarrow (B \vee C)) \wedge \neg C) \rightarrow (A \rightarrow B) \\
\text{P-}\!\rightarrow\text{:} & \overline{(A \rightarrow (B \vee C)) \wedge \neg C \vdash A \rightarrow B} \\
\text{A-}\wedge\text{:} & \overline{A \rightarrow (B \vee C), \neg C \vdash A \rightarrow B} \\
\text{P-}\!\rightarrow\text{:} & \overline{A \rightarrow (B \vee C), \neg C, A \vdash B} \\
\text{MP:} & \overline{A \rightarrow (B \vee C), \neg C, A, B \vee C \vdash B} \\
\text{A-}\vee\text{:} & \overline{\ldots, \neg C, A, B \vdash B \qquad \ldots, \neg C, A, C \vdash B} \\
\text{GoalAssum: / ContrAssum:} & \overline{\qquad\qquad\qquad} \quad \overline{\qquad\qquad\qquad}
\end{array}
\qquad \downarrow
$$

Compare to sequent calculus for propositional logic!

Backward Chaining

Modus ponens: may generate "useless knowledge".

Backward chaining: use implications that "lead to the goal".

Derivable rule:

$$\text{BackChain: } \frac{K \ldots \vdash F}{K \ldots, F \rightarrow G \vdash G}$$

Justified by:

$$\text{Cut: } \dfrac{\text{AnyAssum: } \dfrac{K \ldots \vdash F}{K \ldots, F \rightarrow G \vdash F} \qquad \text{MP: } \dfrac{\text{GoalAssum: } \dfrac{}{K \ldots, F \rightarrow G, F, G \vdash G}}{K \ldots, F \rightarrow G, F \vdash G}}{K \ldots, F \rightarrow G \vdash G}$$

Example

$$\text{BackChain: } \frac{\ldots \vdash \mathit{even}(m)}{\ldots, \mathit{even}(m) \rightarrow \mathit{even}(m^2) \vdash \mathit{even}(m^2)}$$

Natural language description of this proof step:

We know that if m is even then also m^2 is even. Therefore, in order to prove that m^2 is even, it is sufficient to show that m is even.

Equality Rules

- $t = t$ can be proved:

$$\text{P-=: } \frac{}{K\ldots \vdash t = t}$$

- Equality in assumptions $\rightsquigarrow$ substitution:

$$\text{A-=: } \frac{K\ldots[t_2/t_1], t_1 = t_2 \vdash G}{K\ldots, t_1 = t_2 \vdash G} \qquad \text{A-=: } \frac{K\ldots, t_1 = t_2 \vdash G[t_2/t_1]}{K\ldots, t_1 = t_2 \vdash G}$$

$\Gamma[t_2/t_1]$: replace some occurrences of term t_1 by term t_2 in formula or sequence of formulas Γ. If t_1 is a variable, then replace only free occurrences!

The rules A-$\leftrightarrow$ and A-= allow to use all known logical equivalences (e.g. De-Morgan rules, etc.) and arithmetic laws (e.g. distributivity, etc.) for rewriting anywhere in a proof. Typically, not all known rules will be listed explicitly in the assumptions. They may be added through the rule ValidAssum.

Example

$$\text{A-=:}\ \frac{\ldots, even(m), n = m^2 \vdash even(m^2)}{\ldots, even(m), n = m^2 \vdash even(n)}$$

Natural language description of this proof step:

We have to prove that n is even. Since we know $n = m^2$, it suffices to prove that m^2 is even.

Quantifier Rules: Universal Quantifier

- Prove for all $x \rightsquigarrow$ choose $\bar{x}$ "arbitrary but fixed" (skolemize):

$$\text{P-}\forall\text{:}\ \frac{K\ldots \vdash F[\bar{x}/x]}{K\ldots \vdash \forall x : F} \quad \text{if } \bar{x} \text{ does not occur in } K\ldots, F$$

  - What is "arbitrary but fixed"?
  - fixed: $\bar{x}$ is constant in contrast to x, which is a variable.
  - arbitrary: nothing is known about $\bar{x}$, it is a completely new symbol, which does not occur in the current proof situation. It is arbitrary in the sense that we could have taken any other one as well.
  - Justification: for all assignments for x we see that F is true by the argument that works for $\bar{x}$.

- Instantiate universal assumption:

$$\text{A-}\forall\text{:}\ \frac{K\ldots, \forall x : F, F[t/x] \vdash G}{K\ldots, \forall x : F \vdash G}$$

  - $\forall x : F$ stays in the assumptions $\rightsquigarrow$ multiple instantiations.
  - Knowledge generating rule.

Example

$$\text{P-}\forall\text{:}\ \frac{\ldots \vdash \mathit{even}(\bar{n}) \rightarrow \mathit{even}(\bar{n}^2)}{\ldots \vdash \forall n : \mathit{even}(n) \rightarrow \mathit{even}(n^2)}$$

Natural language description of this proof step:

In order to prove that the square of any even number n is again even, we take an arbitrary but fixed natural number $\bar{n}$ and show $\mathit{even}(\bar{n}) \rightarrow \mathit{even}(\bar{n}^2)$.

$$\text{A-}\forall\text{:}\ \frac{\ldots, \forall n : \mathit{even}(n) \rightarrow \mathit{even}(n^2), \mathit{even}(m) \rightarrow \mathit{even}(m^2) \vdash \ldots}{\ldots, \forall n : \mathit{even}(n) \rightarrow \mathit{even}(n^2) \vdash \ldots}$$

Natural language description of this proof step:

We know that the square of any even number is again even. Hence, this holds for a particular number m also, i.e. if m is even then also m^2 must be even.

Quantifier Rules: Existential Quantifier

- Prove there exists $x \rightsquigarrow$ find a witness t (instantiate):

$$\text{P-}\exists\text{:}\ \frac{K \ldots \vdash F[t/x]}{K \ldots \vdash \exists x : F}$$

  - How to find the witness term t?

- Skolemize existential assumption:

$$\text{A-}\exists\text{:}\ \frac{K \ldots, F[\bar{x}/x] \vdash G}{K \ldots, \exists x : F \vdash G} \quad \text{if } \bar{x} \text{ does not occur in } K \ldots, F, G$$

  - $\bar{x}$ is "arbitrary but fixed".

Example

$$\text{P-}\exists\text{:}\ \frac{\ldots \vdash 2 \cdot 2a = 4a}{\ldots \vdash \exists m : 2m = 4a}$$

Natural language description of this proof step:

We have to prove that there exists an m *with* $2m = 4a$*. Let now* $m := 2a$*, thus, we have to show* $2 \cdot 2a = 4a$*.*

$$\text{A-}\exists\text{:}\ \frac{\ldots, \frac{\bar{m}^2}{\bar{n}^2} = 2 \vdash \ldots}{\ldots, \exists m, n : \frac{m^2}{n^2} = 2 \vdash \ldots}$$

Natural language description of this proof step:

We know there exist m *and* n *such that* $\frac{m^2}{n^2} = 2$*. Thus, we may choose* $\bar{m}$ *and* $\bar{n}$ *arbitrary but fixed with* $\frac{\bar{m}^2}{\bar{n}^2} = 2$*.*

Rules for Expanding Definitions

Typically, we assume that definitions are available in a "global context" $\rightsquigarrow$ they are not explicit assumptions in the knowledge base.

Moreover, we assume that the validity conditions have been verified for each definition $\rightsquigarrow$ each defintion corresponds to a valid formula $\rightsquigarrow$ add this formula to the knowledge base and and use available proof rules.

Example: derivable rule for expanding explicit predicate definition.

$$\text{ExpandDef: } \frac{K\ldots[F[z/x]/p(z)] \vdash G}{K\ldots \vdash G} \qquad \begin{array}{l} p(x) :\Leftrightarrow F \\ p(z) \text{ occurs in } K\ldots \end{array}$$

Justified by:

$$\text{ValidAssum: } \cfrac{\text{A-}\forall\text{: } \cfrac{\text{A-}\leftrightarrow\ \cfrac{\text{AnyAssum: } \cfrac{K\ldots[F[z/x]/p(z)] \vdash G}{K\ldots[F[z/x]/p(z)], p(z) \leftrightarrow F[z/x] \vdash G}}{K\ldots, p(z) \leftrightarrow F[z/x] \vdash G}}{K\ldots, \forall x : p(x) \leftrightarrow F \vdash G}}{K\ldots \vdash G} \qquad \begin{array}{l} p(x) :\Leftrightarrow F \\ p(z) \text{ occurs in } K\ldots \end{array}$$

Rules for Expanding Definitions

Using analogous justifications we can derive rules for applying predicate definitions in the goal and for applying explicit function definitions in goal and knowledge base.

$$\text{ExpandDef: } \frac{K \ldots \vdash G[F[z/x]/p(z)]}{K \ldots \vdash G} \quad \begin{array}{l} p(x) :\Leftrightarrow F \\ p(z) \text{ occurs in } G \end{array}$$

$$\text{ExpandDef: } \frac{K \ldots [t[z/x]/f(z)] \vdash G}{K \ldots \vdash G} \quad \begin{array}{l} f(x) := t \\ f(z) \text{ occurs in } K \ldots \end{array}$$

$$\text{ExpandDef: } \frac{K \ldots \vdash G[t[z/x]/f(z)]}{K \ldots \vdash G} \quad \begin{array}{l} f(x) := t \\ f(z) \text{ occurs in } G \end{array}$$

Analogous: Rules for definitions in more than one variable.

Example

If a divides b then it also divides every multiple of b.

<u>Definition:</u> a divides $b :\Leftrightarrow \exists t \in \mathbb{N} : b = t \cdot a$

$\downarrow$

$$\vdash \forall a, b, s \in \mathbb{N} : a \text{ divides } b \rightarrow a \text{ divides } s \cdot b$$

P-$\forall$:

$$\vdash \bar{a}, \bar{b}, \bar{s} \in \mathbb{N} \rightarrow (\bar{a} \text{ divides } \bar{b} \rightarrow \bar{a} \text{ divides } \bar{s} \cdot \bar{b})$$

P-$\rightarrow$:

$$\bar{a}, \bar{b}, \bar{s} \in \mathbb{N} \vdash \bar{a} \text{ divides } \bar{b} \rightarrow \bar{a} \text{ divides } \bar{s} \cdot \bar{b}$$

P-$\rightarrow$:

$$\bar{a}, \bar{b}, \bar{s} \in \mathbb{N}, \bar{a} \text{ divides } \bar{b} \vdash \bar{a} \text{ divides } \bar{s} \cdot \bar{b}$$

$(*)$:

$$\bar{a}, \bar{b}, \bar{s} \in \mathbb{N}, \bar{a} \text{ divides } \bar{b} \vdash \exists t \in \mathbb{N} : \bar{s} \cdot \bar{b} = t \cdot \bar{a}$$

$(**)$:

$$\bar{a}, \bar{b}, \bar{s} \in \mathbb{N}, \exists t \in \mathbb{N} : \bar{b} = t \cdot \bar{a} \vdash \exists t \in \mathbb{N} : \bar{s} \cdot \bar{b} = t \cdot \bar{a}$$

A-$\exists$, A-$\wedge$:

$$\bar{a}, \bar{b}, \bar{s}, \bar{t} \in \mathbb{N}, \bar{b} = \bar{t} \cdot \bar{a} \vdash \exists t \in \mathbb{N} : \bar{s} \cdot \bar{b} = t \cdot \bar{a}$$

A-$=$:

$$\bar{a}, \bar{b}, \bar{s}, \bar{t} \in \mathbb{N}, \bar{b} = \bar{t} \cdot \bar{a} \vdash \exists t \in \mathbb{N} : \bar{s} \cdot \bar{t} \cdot \bar{a} = t \cdot \bar{a}$$

P-$\exists$:

$$\bar{a}, \bar{b}, \bar{s}, \bar{t} \in \mathbb{N}, \bar{b} = \bar{t} \cdot \bar{a} \vdash \bar{s} \cdot \bar{t} \in \mathbb{N} \wedge \bar{s} \cdot \bar{t} \cdot \bar{a} = \bar{s} \cdot \bar{t} \cdot \bar{a}$$

P-$\wedge$:

$$\frac{\ldots, \bar{s}, \bar{t} \in \mathbb{N} \vdash \bar{s} \cdot \bar{t} \in \mathbb{N}}{} \qquad \text{P-}=:\ \frac{\ldots \vdash \bar{s} \cdot \bar{t} \cdot \bar{a} = \bar{s} \cdot \bar{t} \cdot \bar{a}}{}$$

Example: Explanation

In the example: apply definition of "divides"

$$\forall a, b : a \text{ divides } b \leftrightarrow \exists t \in \mathbb{N} : b = t \cdot a \tag{1}$$

to the goal "$\bar{a}$ divides $\bar{s} \cdot \bar{b}$" (instantiate $[a \mapsto \bar{a}, b \mapsto \bar{s} \cdot \bar{b}]$).

$$(*): \frac{\bar{a}, \bar{b}, \bar{s} \in \mathbb{N}, \bar{a} \text{ divides } \bar{b} \vdash \bar{a} \text{ divides } \bar{s} \cdot \bar{b}}{\bar{a}, \bar{b}, \bar{s} \in \mathbb{N}, \bar{a} \text{ divides } \bar{b} \vdash \exists t \in \mathbb{N} : \bar{s} \cdot \bar{b} = t \cdot \bar{a}} \quad \downarrow$$

Apply (1) to the assumption "$\bar{a}$ divides $\bar{b}$" (instantiate $[a \mapsto \bar{a}, b \mapsto \bar{b}]$):

$$(**): \frac{\bar{a}, \bar{b}, \bar{s} \in \mathbb{N}, \bar{a} \text{ divides } \bar{b} \vdash \exists t \in \mathbb{N} : \bar{s} \cdot \bar{b} = t \cdot \bar{a}}{\bar{a}, \bar{b}, \bar{s} \in \mathbb{N}, \exists t \in \mathbb{N} : \bar{b} = t \cdot \bar{a} \vdash \exists t \in \mathbb{N} : \bar{s} \cdot \bar{b} = t \cdot \bar{a}} \quad \downarrow$$

Rules for Implicit Function Definitions

Implicit definitions are slightly more tricky ...

$$\text{ImpDef: } \frac{\begin{array}{l} K\ldots[\bar{y}/f(z)], \\ F[z/x][\bar{y}/y], \bar{y} \in T \end{array} \quad \vdash G[\bar{y}/f(z)]}{K\ldots, \exists y \in T : F[z/x] \vdash G} \qquad \begin{array}{l} f(x) := \textbf{such } y \in T : F \\ f(z) \text{ occurs in } K\ldots, G \end{array}$$

Note, that $\bar{y}$ must not occur in $K\ldots, F, G$.

In words: if $f(z)$ is defined, then we can introduce a $\bar{y}$ for $f(z)$ and $\bar{y}$ has the characteristic property from the definition for $f(z)$. We may replace $f(z)$ by $\bar{y}$ anywhere in the proof.

$$\text{ImpDefUn: } \frac{\begin{array}{l} K\ldots[\bar{y}/f(z)], \\ F[z/x][\bar{y}/y], \bar{y} \in T \end{array} \quad \vdash G[\bar{y}/f(z)]}{K\ldots \vdash G} \qquad \begin{array}{l} f(x) := \textbf{the } y \in T : F \\ f(z) \text{ occurs in } K\ldots, G \end{array}$$

$$\text{ImpDefUnEq: } \frac{K\ldots \vdash F[z/x][t/y]}{K\ldots, t \in T \vdash f(z) = t} \qquad f(x) := \textbf{the } y \in T : F$$

Example

Prove that for every bijective function $f : A \to B$ we have $(f^{-1})^{-1} = f$.

Inverse function exists and is unique (bijective!) $\rightsquigarrow$ implicit definition:

$$f^{-1} := \textbf{the } g : B \to A : (f \circ g = \mathrm{id}_B) \wedge (g \circ f = \mathrm{id}_A)$$

$$\text{P-}\forall\text{, P-}\to\text{:}\ \frac{\vdash \forall A, B, f : A \to B : (f^{-1})^{-1} = f}{\bar{f} : \bar{A} \to \bar{B} \vdash (\bar{f}^{-1})^{-1} = \bar{f}} \qquad \downarrow$$

$$\text{ImpDefUn, A-}\wedge\text{:}\ \frac{\bar{f} : \bar{A} \to \bar{B} \vdash (\bar{f}^{-1})^{-1} = \bar{f}}{\bar{f} : \bar{A} \to \bar{B}, \bar{g} : \bar{B} \to \bar{A}, \bar{f} \circ \bar{g} = \mathrm{id}_{\bar{B}}, \bar{g} \circ \bar{f} = \mathrm{id}_{\bar{A}} \vdash \bar{g}^{-1} = \bar{f}}$$

$$\text{ImpDefUnEq:}\ \frac{\bar{f} : \bar{A} \to \bar{B}, \bar{g} : \bar{B} \to \bar{A}, \bar{f} \circ \bar{g} = \mathrm{id}_{\bar{B}}, \bar{g} \circ \bar{f} = \mathrm{id}_{\bar{A}} \vdash \bar{g}^{-1} = \bar{f}}{\begin{array}{l}\bar{f} : \bar{A} \to \bar{B}, \bar{g} : \bar{B} \to \bar{A},\\ \bar{f} \circ \bar{g} = \mathrm{id}_{\bar{B}}, \bar{g} \circ \bar{f} = \mathrm{id}_{\bar{A}}\end{array} \vdash (\bar{g} \circ \bar{f} = \mathrm{id}_{\bar{A}}) \wedge (\bar{f} \circ \bar{g} = \mathrm{id}_{\bar{B}})}$$

$$\text{P-}\wedge\text{:}\ \frac{\cdots \vdash (\bar{g} \circ \bar{f} = \mathrm{id}_{\bar{A}}) \wedge (\bar{f} \circ \bar{g} = \mathrm{id}_{\bar{B}})}{\dfrac{K \vdash \bar{g} \circ \bar{f} = \mathrm{id}_{\bar{A}}}{} \qquad \dfrac{K \vdash \bar{f} \circ \bar{g} = \mathrm{id}_{\bar{B}}}{}}$$

In both cases the knowledge base K contains the goal to be proved.

Be careful with instantiation in second application of (ImpDef).

Natural Language Presentation of Proofs

1. Do not mention all steps,
2. combine several steps into one (derivable rules!),
3. use same names for arbitrary but fixed constants, etc.

Theorem: If a divides b then it also divides every multiple of b.

Proof: Assume $a, b, s \in \mathbb{N}$ arbitrary but fixed such that a divides b. We have to show that a divides $s \cdot b$, i.e. $\exists t \in \mathbb{N} : s \cdot b = t \cdot a$. Since a divides b, we know that $b = \bar{t} \cdot a$ for some $\bar{t} \in \mathbb{N}$, thus, we have to find $t \in \mathbb{N}$ s.t. $s \cdot \bar{t} \cdot a = t \cdot a$. Let now $t := s \cdot \bar{t} \in \mathbb{N}$, we have to show $s \cdot \bar{t} \cdot a = s \cdot \bar{t} \cdot a$. q.e.d.

Every sentence in the proof is justified by one or more proof rules. Trivial steps (e.g. split conjunction in knowledge base) not mentioned explicitly.

Example

Every even natural number is the sum of two odd numbers with a difference less or equal than 2, i.e.

$$\forall even(n) : \exists odd(k), odd(l) : n = k + l \wedge k - l \leq 2$$

Let n be arbitrary but fixed and assume n is even.	P-$\forall$, P-$\rightarrow$
Hence, $n = 2m$.	ExpandDef, A-$\exists$
	$\forall n : odd(n) \vee even(n)$, A-$\forall$
Case m is odd:	$odd(m) \vee even(m)$, A-$\vee$
Let $k = l := m$. Then $k + l = 2m = n$,	P-$\exists$
thus, n is the sum of two odd numbers k and l s.t. $k - l = 0 \leq 2$.	GoalAssum
Case m is even:	A-$\vee$
Let $k := m + 1$ and $l := m - 1$.	P-$\exists$
Then $k + l = m + 1 + m - 1 = 2m = n$,	GoalAssum
thus, n is the sum of two odd numbers k and l s.t. $k - l = 2 \leq 2$.	

Drinker's Paradox

In every non-empty bar there is one person such that if (s)he drinks, then everybody drinks.

$$\exists x : (D(x) \rightarrow \forall y : D(y)) \tag{2}$$

Apply P-$\exists$: no chance.

Apply proof by contradiction, assume $\neg\exists x : (D(x) \rightarrow \forall y : D(y))$, i.e.

$$\forall x : (D(x) \wedge \exists y : \neg D(y)) \tag{3}$$

Since the bar is not empty, there is at least one person in the bar, call her/him p. Since (3) holds for all x, it must also hold for p (instantiation!), thus $D(p)$ and also $\exists y : \neg D(y)$. So there exists a person, call her/him q, such that

$$\neg D(q). \tag{4}$$

But (3) must hold for q also, i.e. $D(q) \wedge \neg\forall y : D(y)$, thus

$$D(q). \tag{5}$$

(5) contradicts (4), so the original statement (2) is proven.

Example

Prove over the domain $\mathbb{N}$: $\forall n: 0+n=n$.

$$\forall n: n+0=n. \tag{P3}$$

$$\forall m,n: n+(m+1)=(n+m)+1. \tag{P4}$$

$$(A[0/n] \wedge (\forall m: A[m/n] \rightarrow A[m+1/n])) \rightarrow \forall n: A \tag{P5}$$

In this case for $A \equiv 0+n=n$: By (BackChain), in order to prove $\forall n: 0+n=n$, it is sufficient to prove

$$A[0/n] \wedge (\forall m: A[m/n] \rightarrow A[m+1/n]).$$

Using (P-$\wedge$) we have to

1. Prove $A[0/n]$, i.e. $0+0=0$. Instantiation of (P3) by $[n \mapsto 0]$ yields $0+0=0$, hence we are done (GoalAssum).
2. Prove $\forall m: A[m/n] \rightarrow A[m+1/n]$, i.e. for arbitrary but fixed m, we assume $0+m=m$ $(*)$ and show $0+(m+1)=m+1$. Now,

$$0+(m+1) \overset{(P4)}{=} (0+m)+1 \overset{(*)}{=} m+1.$$

Summary

- Proof rules are purely syntactic $\rightsquigarrow$ proving can be viewed as a syntactic process.
- When doing "real mathematical proofs":
  - Obey the syntactic structure of the involved formulas.
  - Apply rules "matching" the current proof situation.
  - Think of the proof as a tree and try to "close" all branches.
  - Instead of "waiting for the brilliant idea" that solves a proof problem, better "stupidly" apply the rules.
- You will be surprised, in how many proofs you will succeed this way!