

art.1 Derivations in LK

inc:art:plk:
sec

The definition of Deriv is not actually primitive recursive and needs to be rewritten.

In order to arithmetize **derivations**, we must represent **derivations** as numbers. explanation
 Since **derivations** are trees of sequents where each inference carries also a label, a recursive representation is the most obvious approach: we represent a **derivation** as a tuple, the components of which are the end-sequent, the label, and the representations of the sub-**derivations** leading to the premises of the last inference.

Definition art.1. If Γ is a finite sequence of **sentences**, $\Gamma = \langle \varphi_1, \dots, \varphi_n \rangle$, then $\# \Gamma \# = \langle \# \varphi_1 \#, \dots, \# \varphi_n \# \rangle$.

If $\Gamma \Rightarrow \Delta$ is a sequent, then a Gödel number of $\Gamma \Rightarrow \Delta$ is

$$\# \Gamma \Rightarrow \Delta \# = \langle \# \Gamma \#, \# \Delta \# \rangle$$

If π is a **derivation** in **LK**, then $\# \pi \#$ is

1. $\langle 0, \# \Gamma \Rightarrow \Delta \# \rangle$ if π consists only of the initial sequent $\Gamma \Rightarrow \Delta$.
2. $\langle 1, \# \Gamma \Rightarrow \Delta \#, k, \# \pi' \# \rangle$ if π ends in an inference with one premise, k is given by the following table according to which rule was used in the last inference, and π' is the immediate subproof ending in the premise of the last inference.

Rule:	WL	WR	CL	CR	XL	XR
k :	1	2	3	4	5	6

Rule:	\neg L	\neg R	\wedge L	\vee R	\rightarrow R
k :	7	8	9	10	11

Rule:	\forall L	\forall R	\exists L	\exists R	=
k :	12	13	14	15	16

3. $\langle 2, \# \Gamma \Rightarrow \Delta \#, k, \# \pi' \#, \# \pi'' \# \rangle$ if π ends in an inference with two premises, k is given by the following table according to which rule was used in the last inference, and π' , π'' are the immediate subproof ending in the left and right premise of the last inference, respectively.

Rule:	Cut	\wedge R	\vee L	\rightarrow L
k :	1	2	3	4

Having settled on a representation of **derivations**, we must also show that we can manipulate such derivations primitive recursively, and express their essential properties and relations so. explanation
 Some operations are simple: e.g., given

a Gödel number d of a **derivation**, $(s)_1$ gives us the Gödel number of its end-sequent. Some are much harder. We'll at least sketch how to do this. The goal is to show that the relation “ π is a **derivation** of φ from Γ ” is a primitive recursive relation of the Gödel numbers of π and φ .

Proposition art.2. *The following relations are primitive recursive:*

*inc:art:plk:
prop:followsby*

1. $\Gamma \Rightarrow \Delta$ is an initial sequent.
2. $\Gamma \Rightarrow \Delta$ follows from $\Gamma' \Rightarrow \Delta'$ (and $\Gamma'' \Rightarrow \Delta''$) by a rule of **LK**.
3. π is a correct **LK-derivation**.

Proof. We have to show that the corresponding relations between Gödel numbers of **formulas**, sequences of Gödel numbers of **formulas** (which code sequences of **formulas**), and Gödel numbers of sequents, are primitive recursive.

1. $\Gamma \Rightarrow \Delta$ is an initial sequent if either there is a **sentence** φ such that $\Gamma \Rightarrow \Delta$ is $\varphi \Rightarrow \varphi$, or there is a term t such that $\Gamma \Rightarrow \Delta$ is $\emptyset \Rightarrow t = t$. In terms of Gödel numbers, $\text{InitSeq}(s)$ holds iff

$$\begin{aligned} & (\exists x < s) (\text{Sent}(x) \wedge s = \langle \langle x \rangle, \langle x \rangle \rangle) \vee \\ & (\exists t < s) (\text{Term}(t) \wedge s = \langle 0, \langle \# = (\# \frown t \frown \#, \# \frown t \frown \#) \rangle \rangle). \end{aligned}$$

2. Here we have to show that for each rule of inference R the relation $\text{FollowsBy}_R(s, s')$ which holds if s and s' are the Gödel numbers of conclusion and premise of a correct application of R is primitive recursive. If R has two premises, FollowsBy_R of course has three arguments.

For instance, $\Gamma \Rightarrow \Delta$ follows correctly from $\Gamma' \Rightarrow \Delta'$ by $\exists R$ iff $\Gamma = \Gamma'$ and there is a sequence of **formulas** Δ'' , a **formula** φ , a variable x and a closed term t such that $\Delta' = \Delta'', \varphi[t/x]$ and $\Delta = \Delta'', \exists x \varphi$. We just have to translate this into Gödel numbers. If $s = \# \Gamma \Rightarrow \Delta \#$ then $(s)_0 = \# \Gamma \#$ and $(s)_1 = \# \Delta \#$. So, $\text{FollowsBy}_{\exists R}(s, s')$ holds iff

$$\begin{aligned} & (s)_0 = (s')_0 \wedge \\ & (\exists d < s) (\exists f < s) (\exists x < s) (\exists t < s') (\text{Frm}(f) \wedge \text{Var}(x) \wedge \text{Term}(t) \wedge \\ & \quad (s')_1 = d \frown \langle \text{Subst}(f, t, x) \rangle \wedge \\ & \quad (s)_1 = d \frown \langle \#(\exists) \frown x \frown f \rangle \end{aligned}$$

The individual lines express, respectively, “ $\Gamma = \Gamma'$,” “there is a sequence (Δ'') with Gödel number d , a **formula** (φ) with Gödel number f , a variable with Gödel number x , and a term with Gödel number t ,” “ $\Delta' = \Delta'', \varphi[t/x]$,” and “ $\Delta = \Delta'', \exists x \varphi$ ”. (Remember that $\# \Delta \#$ is the number of a sequence of Gödel numbers of **formulas** in Δ .)

3. We first define a helper relation $\text{hDeriv}(s, n)$ which holds if s codes a correct derivation to at least n inferences up from the end sequent. If $n =$

0 we let the relation be satisfied by default. Otherwise, $\text{hDeriv}(s, n+1)$ iff either s consists just of an initial sequent, or it ends in a correct inference and the codes of the immediate subderivations satisfy $\text{hDeriv}(s, n)$.

$$\begin{aligned}
& \text{hDeriv}(s, 0) \Leftrightarrow \text{true} \\
& \text{hDeriv}(s, n+1) \Leftrightarrow \\
& \quad ((s)_0 = 0 \wedge \text{InitialSeq}((s)_1)) \vee \\
& \quad ((s)_0 = 1 \wedge \\
& \quad \quad ((s)_2 = 1 \wedge \text{FollowsBy}_{\text{CL}}((s)_1, ((s)_3)_1)) \vee \\
& \quad \quad \vdots \\
& \quad \quad ((s)_2 = 16 \wedge \text{FollowsBy}_{=}((s)_1, ((s)_3)_1)) \wedge \\
& \quad \quad \text{hDeriv}((s)_3, n)) \vee \\
& \quad ((s)_0 = 2 \wedge \\
& \quad \quad ((s)_2 = 1 \wedge \text{FollowsBy}_{\text{Cut}}((s)_1, ((s)_3)_1, ((s)_4)_1)) \vee \\
& \quad \quad \vdots \\
& \quad \quad ((s)_2 = 4 \wedge \text{FollowsBy}_{\rightarrow\text{L}}((s)_1, ((s)_3)_1, ((s)_4)_1)) \wedge \\
& \quad \quad \text{hDeriv}((s)_3, n) \wedge \text{hDeriv}((s)_4, n))
\end{aligned}$$

This is a primitive recursive definition. If the number n is large enough, e.g., larger than the maximum number of inferences between an initial sequent and the end sequent in s , it holds of s iff s is the Gödel number of a correct **derivation**. The number s itself is larger than that maximum number of inferences. So we can now define $\text{Deriv}(s)$ by $\text{hDeriv}(s, s)$.

□

Problem art.1. Define the following relations as in [Proposition art.2](#):

1. $\text{FollowsBy}_{\wedge\text{R}}(s, s', s'')$,
2. $\text{FollowsBy}_{=}(s, s')$,
3. $\text{FollowsBy}_{\vee\text{R}}(s, s')$.

Proposition art.3. *Suppose Γ is a primitive recursive set of **sentences**. Then the relation $\text{Prf}_{\Gamma}(x, y)$ expressing “ x is the code of a **derivation** π of $\Gamma_0 \Rightarrow \varphi$ for some finite $\Gamma_0 \subseteq \Gamma$ and x is the Gödel number of φ ” is primitive recursive.*

Proof. Suppose “ $y \in \Gamma$ ” is given by the primitive recursive predicate $R_{\Gamma}(y)$. We have to show that $\text{Prf}_{\Gamma}(x, y)$ which holds iff y is the Gödel number of a sentence φ and x is the code of an **LK-derivation** with end sequent $\Gamma_0 \Rightarrow \varphi$ is primitive recursive.

By the previous proposition, the property $\text{Deriv}(x)$ which holds iff x is the code of a correct derivation π in **LK** is primitive recursive. If x is such a code,

then $(x)_1$ is the code of the end sequent of π , and so $((x)_1)_0$ is the code of the left side of the end sequent and $((x)_1)_1$ the right side. So we can express “the right side of the end sequent of π is φ ” as $\text{len}(((x)_1)_1) = 1 \wedge (((x)_1)_1)_0 = x$. The left side of the end sequent of π is of course automatically finite, we just have to express that every sentence in it is in Γ . Thus we can define $\text{Prf}_\Gamma(x, y)$ by

$$\begin{aligned} \text{Prf}_\Gamma(x, y) \Leftrightarrow & \text{Sent}(y) \wedge \text{Deriv}(x) \wedge \\ & (\forall i < \text{len}(((x)_1)_0)) R_\Gamma(((x)_1)_0)_i) \wedge \\ & \text{len}(((x)_1)_1) = 1 \wedge (((x)_1)_1)_0 = x \end{aligned}$$

□

Photo Credits

Bibliography