Math 293A: Strong Normalization for $\mathbf{Nm}_{\rightarrow}$, $\lambda_{\rightarrow}$ and Arithmetic

Ting Zhang (tingz@cs.stanford.edu)
Department of Computer Science
Stanford University
November 12, 2002

1 Introduction

In this report we present strong normalization proofs of $\mathbf{Nm}_{\rightarrow}$ and arithmetic respectively [TS00] [Lei75]. As we shall see the two proofs bear great similarity.

2 Strong Normalization for $\mathbf{Nm}_{\rightarrow}$ and $\lambda_{\rightarrow}$ [TS00]

Definition 2.1. $\mathbf{Nm}_{\rightarrow}$ *is the natural deduction system of minimal implicational logic and* $\lambda_{\rightarrow}$ *is the system of* λ *terms of corresponding deductions.*

Definition 2.2 (Strongly Normalizable)**.** *A* $\lambda_{\rightarrow}$ *term* t *is* ***strongly normalizable (s.n.)*** *if any* β*-conversion sequence beginning with* t *terminates. Let* $\succ$ *denote* 1*-step* β*-conversion and* $\succ\!\!\succ$ *multi-step* β*-conversion respectively.*

Definition 2.3. *A* $\lambda_{\rightarrow}$ *term* t *is* ***non-introduced*** *if* t *is not of the form* $\lambda x.s$*. In other words* t *is non-introduced if the final rule of the corresponding derivation is not* $\rightarrow I$*.*

Example 2.1. $\mathbf{k}_{\lambda}^{A,B} \equiv \lambda x^A y^B.x^A$ *is introduced while terms of the form* st *is non-introduced.*

Definition 2.4. *We define* ***computability predicate*** $Comp_T(t)$ *recursively as follows.*

$$\begin{aligned} Comp_X(t) &:= \mathbf{SN}(t) \\ Comp_{A\rightarrow B}(t) &:= \forall s(Comp_A(s) \rightarrow Comp_B(ts)) \end{aligned}$$

Definition 2.5 (Strong Computability)**.** *A term $t : B$ is **strongly computable** if $FV(t) \subseteq \{x_1 : A_1, \ldots, x_n : A_n\}$ and $Comp_{A_i}(s_i)$ for $i \geq n$, then $Comp_B(t[x_1, \ldots, x_n/s_1, \ldots, s_n])$.*

Lemma 2.1. *Four properties hold for Comp.*

C1 If $Comp_A(t)$, then $\mathbf{SN}(t)$.

C2 If $Comp_A(t)$ and $t \succ\!\succ t'$, then $Comp_A(t')$.

C3 If t is non-introduced and $\forall t' \prec t\ Comp_A(t')$, then $Comp_A(t)$.

C4 If t is non-introduced and normal, then $Comp_A(t)$.

Proof. We show $C1$-$C3$ by induction simultaneously and $C4$ follows outright from $C3$ since if t is normal then $\forall t' \prec t\ Comp_A(t')$ vacuously holds.
Induction Base: $A \equiv X$.
$C1$, $C2$ and $C3$ follow immediately from the definition.
Induction Step: $A \equiv B \to C$.

C1 Suppose that $Comp_{B \to C}(t)$ and let x be a variable of type B. By definition $Comp_C(tx)$ and by IH of C1, $\mathbf{SN}(tx)$. It follows that $\mathbf{SN}(t)$ as any reduction tree of t is embedded in a reduction tree of tx.

C2 Let $t' \prec t$ and $s \in Comp_B$. By defintion $Comp_C(ts)$ and since $ts \succ\!\succ t's$, by IH $Comp_C(t's)$. It follows from definition that $Comp_{B \to C}(t')$.

C3 Let $s \in Comp_B$ and $t'' \prec ts$. As t is non-introduction, either $t'' \equiv t's$ and $t' \prec t$ or $t'' \equiv ts'$ and $s' \prec s$.

- $t'' \equiv t's$ and $t' \prec t$. By assumption $Comp_{B \to C}(t')$ and hence $Comp_C(t's)$, that is $Comp_C(t'')$. By IH of C3, $Comp_C(ts)$ and by definition $Comp_{B \to C}(t)$ as s is arbitrary.
- $t'' \equiv ts'$ and $s' \prec s$. We use subinduction on the length of reduction of s. The base case is trivial as s can not be normal. As the length of reduction of s' is less than that of s, by sub IH, $Comp_C(ts')$ and by IH, $Comp_C(ts)$ and so $Comp_{B \to C}(t)$.

$\square$

Lemma 2.2 (Substitution)**.** *If $\forall s(Comp_A(s) \to Comp_B(t[x/s]))$, then $Comp_{A \to B}(\lambda x.t)$.*

Proof. Let $s \in Comp_A$. We need to show that $Comp_B((\lambda x.t)s)$. Let if $t'' \prec (\lambda x.t)s$. By Lemma 2.1.C3 it suffices to show $Comp_B(t'')$. We do induction on the sum $h_s + h_t$ of reduction heights of s and t.
Base case: $h_s + h_t = 0$.
Then $t'' \equiv t[x/s]$ and by assumption $Comp_B(t'')$.
Induction step.

1. $t'' \equiv (\lambda x.t)s'$ and $s' \prec s$. So $t[x/s'] \succ t[x/s']$. By Lemma 2.1 C2 $Comp_B(t[x/s'])$ and by IH, $Comp_B(t'')$.

2. $t'' \equiv (\lambda x.t')s$ and $t' \prec t$. So $t'[x/s] \prec t[x/s]$. By Lemma 2.1 C2 $Comp_B(t'[x/s])$ and by IH, $Comp_B(t'')$.

3. $t'' \equiv t[x/s]$ and by assumption $Comp_B(t'')$.

□

Theorem 2.1 (Strong Computability)**.** *All terms of $\lambda_\rightarrow$ are strongly computable under substitution.*

Proof. By induction on t. If t is a variable, then t is strongly computable by assumption. Let $t^* = t[x_1/s_1, \dots, x_n/s_n]$.

1. $t^B = t_1^{A\rightarrow B} t_2^A$. Then $t^{*,B} = t_1^{*,A\rightarrow B} t_2^{*,A}$. By IH $Comp_{A\rightarrow B}(t_1^*)$ and $Comp_A(t_2^*)$. By defintion $Comp_B(t_1^* t_2^*)$.

2. $t^{A\rightarrow B} = \lambda x^A.t_1^B$. Let $s \in Comp_A$. Note that

$$t_1^*[x/s] \equiv t_1[x, x_1, \dots, x_n/s, s_1, \dots, s_n].$$

   By IH $Comp_B(t_1^*[x/s])$ and by Lemma 2.2 $Comp_{A\rightarrow B}(\lambda x.t_1^*)$, that is, $Comp_{A\rightarrow B}(t^*)$.

□

Corollary 2.1. *All terms of $\lambda_\rightarrow$ (deductions in* **Nm**$_\rightarrow$*) are strongly normalizable.*

2.1 Reduction of strong normalization between systems

In general if we can map one derivation step in system **S** to finite number of derivation steps in system **S**′, then strong normalization of **S**′ implies strong normalization for **S**.

Example 2.2. *Reduction of strong normalization for $\lambda_{\forall\to}$ to strong normalization for $\lambda_{\to}$.*

Define reduction map φ recursively as follows.

$$\begin{array}{rcl} \varphi(Rt_1 \ldots t_n) & = & R^* \quad (R^* \in \mathcal{PV}) \\ \varphi(A \to B) & = & \varphi(A) \to \varphi(B) \\ \varphi(\forall x A x) & = & (Q \to Q) \to A \quad (Q \in \mathcal{PV} \textit{ distinict from } R^*) \end{array}$$

Every derivation of $\lambda_{\forall\to}$ is mapped under φ to a derivation of $\lambda_{\to}$. More precisely,

1.

$$\dfrac{\begin{array}{c}[A]\\ \Sigma \\ B\end{array}}{A \to B}\,(\to I) \quad \overset{\varphi}{\mapsto} \quad \dfrac{\begin{array}{c}[\varphi(A)]\\ \varphi(\Sigma) \\ \varphi(B)\end{array}}{\varphi(A) \to \varphi(B)}\,(\to I)$$

2.

$$\dfrac{\begin{array}{c}\Sigma(a) \\ A(a)\end{array}}{\forall x A x}\,(\forall I) \quad \overset{\varphi}{\mapsto} \quad \dfrac{\begin{array}{c}\varphi(\Sigma(a)) \\ \varphi(A(a))\end{array}}{(Q \to Q) \to \varphi(A(a))}\,(\forall I)$$

3.

$$\dfrac{\begin{array}{c}\Sigma \\ \forall x A x\end{array}}{At}\,(\forall E) \quad \overset{\varphi}{\mapsto} \quad \dfrac{\begin{array}{c}\varphi(\Sigma) \\ (Q \to Q) \to \varphi(A)\end{array} \quad \dfrac{Q}{Q \to Q}}{\varphi(At)}\,(\to E)$$

*It is easily seen that this reduction is sound, i.e., one derivation step in $\lambda_{\forall\to}$ corresponds to one or two derivation steps in $\lambda_{\to}$. As $\lambda_{\to}$ is **s.n.**, so is $\lambda_{\forall\to}$.*

3 Strong Normalization for Arithmetic [Lei75]

Definition 3.1 (Complexity Measure)**.** *The measure μ on formulas is defined recursively on their structures:*

$$\begin{array}{r} \mu(A) := 0 \textit{ for } A \textit{ atomic}, \\ \mu(A \,\&\, B) := \mu(A \vee B) := max(\mu(A), \mu(B)), \\ \mu(\forall x A x) := \mu(\exists x A x) = \mu(A\bar{0}), \\ \mu(A \to B) := max(\mu(A) + 1, \mu(B)). \end{array}$$

We define $\mu(\Delta)$ to be $\mu(A)$ where A is derived formula of Δ.

Definition 3.2 (Detour Reduction)**.** *There are five detour reductions, denoted by* $\succ_\odot$ *for* $\odot \equiv \&, \vee, \to, \forall, \exists$.

$\succ_\&$

$$\dfrac{\dfrac{\begin{matrix}\Sigma_0\\ A_0\end{matrix}\quad\begin{matrix}\Sigma_1\\ A_1\end{matrix}}{A_0 \,\&\, A_1}}{A_i} \qquad \succ_\& \qquad \begin{matrix}\Sigma_i\\ A_i\end{matrix} \qquad (i=0,1)$$

$\succ_\vee$

$$\dfrac{\dfrac{\begin{matrix}\Sigma\\ A_i\end{matrix}}{A_0 \vee A_1}\quad\begin{matrix}[A_0]\\ \Delta_0\\ C\end{matrix}\quad\begin{matrix}[A_1]\\ \Delta_1\\ C\end{matrix}}{C} \qquad \succ_\vee \qquad \begin{matrix}\Sigma\\ [A_i]\\ \Delta_i\\ C\end{matrix} \qquad (i=0,1)$$

$\succ_\to$

$$\dfrac{\dfrac{\begin{matrix}[A]\\ \Sigma\\ B\end{matrix}}{A \to B}\quad\begin{matrix}\Delta\\ A\end{matrix}}{B} \qquad \succ_\to \qquad \begin{matrix}\Delta\\ [A]\\ \Sigma\\ B\end{matrix}$$

$\succ_\forall$

$$\dfrac{\dfrac{\begin{matrix}\Sigma(a)\\ Aa\end{matrix}}{\forall x A x}}{At} \qquad \succ_\forall \qquad \begin{matrix}\Sigma(t)\\ At\end{matrix}$$

$\succ_\exists$

$$\dfrac{\dfrac{\begin{matrix}\Sigma\\ At\end{matrix}}{\exists x A x}\quad\begin{matrix}[Aa]\\ \Delta(a)\\ B\end{matrix}}{B} \qquad \succ_\exists \qquad \begin{matrix}\Sigma\\ [At]\\ \Delta(t)\\ B\end{matrix}$$

Definition 3.3 (Strongly Normalizable)**.** *A derivation* Δ *is* ***strongly normalizable (s.n.)*** *if it is impossible to have the infinitely desceding chain as follows:*

$$\Delta \succ \Delta_1 \succ \dots$$

If Δ *is* ***s.n.****, we write* $\nu(\Delta)$ *for the maximum* n *such that*

$$\Delta \succ \Delta_1 \succ \dots \Delta_n$$

$\nu(\Delta)$ *is well-defined by König's Lemma and the fact that every derivation has only finite number of reducts.*

Definition 3.4 (Improper Reduction)**.** *There are five corresponding improper reductions (written* $\succsim_\odot$ *for* $\odot \equiv \&, \vee, \to, \forall, \exists$*) which are only used in the proof.*

$\gtrsim_{\&}$

$$\dfrac{\begin{matrix}\Sigma_0 & \Sigma_1\\ A_0 & A_1\end{matrix}}{A_0 \,\&\, A_1} \quad \gtrsim_{\&} \quad \begin{matrix}\Sigma_i\\ A_i\end{matrix} \qquad (i=0,1)$$

$\gtrsim_{\vee}$

$$\dfrac{\begin{matrix}\Sigma\\ A_i\end{matrix}}{A_0 \vee A_1} \quad \gtrsim_{\vee} \quad \begin{matrix}\Sigma\\ A_i\end{matrix} \qquad (i=0,1)$$

$\gtrsim_{\rightarrow}$

$$\dfrac{\begin{matrix}[A]\\ \Sigma\\ B\end{matrix}}{A \rightarrow B} \quad \gtrsim_{\rightarrow} \quad \begin{matrix}\Delta\\ [A]\\ \Sigma\\ B\end{matrix} \qquad \left(\begin{matrix}\Delta\\ A\end{matrix} \textit{ is stable}\right)$$

$\gtrsim_{\forall}$

$$\dfrac{\begin{matrix}\Sigma(a)\\ Aa\end{matrix}}{\forall x A x} \quad \gtrsim_{\forall} \quad \begin{matrix}\Sigma(t)\\ At\end{matrix}$$

$\gtrsim_{\exists}$

$$\dfrac{\begin{matrix}\Sigma\\ At\end{matrix}}{\exists x A x} \quad \gtrsim_{\exists} \quad \begin{matrix}\Sigma\\ At\end{matrix}$$

Definition 3.5 (Stability)**.** *We write* $\Delta > \Delta'$ *if* $\Delta \succ \Delta'$ *or* $\Delta \gtrsim \Delta'$ *and* $\Delta \gg \Delta'$ *if for some* $n \geq 0$ *we have as a sequence*

$$\Delta \equiv \Delta_0 > \Delta_1 > \ldots > \Delta_n \equiv \Delta'$$

$\Delta < \Delta'$ *and* $\Delta \ll \Delta'$ *are defined similarly.*

We say Δ *is* ***stable*** *if for any* Δ' $\Delta \gg \Delta'$ *implies* Δ' *is* ***s.n.****.*

Definition 3.6 (Substitution Stability)**.** *We write* $\Delta \mapsto \Delta^*$ *if* Δ^* *is obtained by substituting any terms for parameters in* Δ *and then substituting stable derivations for some open assumptions in* Δ*. We say* Δ *is* ***stable under substitution (s.s.)*** *if for any* Δ^* $\Delta \mapsto \Delta^*$ *implies* Δ^* *is stable.*

3.1 Easy Lemmas

Lemma 3.1. Δ *is stable if and only if* $\Delta > \Delta'$ *implies* Δ' *is stable.*

Lemma 3.2. Δ *is* ***s.s.****implies* Δ *is stable and* Δ *is stable implies* Δ *is* ***s.n.****.*

Lemma 3.3. *Let*

$$\Delta \quad \equiv \quad \frac{\Delta_0 \quad (\Delta_1)}{A}\ (\rho)$$

where ρ is an introduction rule other than $\vee I$ or $\forall I$. If Δ_0 (and Δ_1) are stable, then Δ is stable.

Lemma 3.4. *If a is free in*

$$\begin{matrix}\Delta_0(a)\\ Aa\end{matrix}$$

and $\Delta_0(t)$ is stable for every t, then

$$\Delta \quad \equiv \quad \frac{\begin{matrix}\Delta(a)\\ Aa\end{matrix}}{\forall x A x}$$

is stable.

Proof. We show the result by induction on $\nu(\Delta_0)$. Suppose that

$$\Delta \quad \succ \quad \Delta' \quad \equiv \quad \frac{\begin{matrix}\Delta_0'(a)\\ Aa\end{matrix}}{\forall x A x}$$

Then $\nu(\Delta_0') < \nu(\Delta_0)$. By IH Δ' is stable. Suppose

$$\Delta \quad \succsim \quad \Delta' \quad \equiv \quad \begin{matrix}\Delta_0(t)\\ At\end{matrix}$$

Then Δ' is stable by assumption. So By Lemma 3.1 Δ is stable. ☐

Lemma 3.5. *If*

$$\begin{matrix}\Delta_0\\ At\end{matrix}$$

is stable, then

$$\Delta \quad \equiv \quad \frac{\begin{matrix}\Delta_0\\ At\end{matrix}}{\exists x A x}$$

is stable.

Proof. We show the result by induction on $\nu(\Delta_0)$. Suppose

$$\Delta \quad \succ \quad \Delta' \quad \equiv \quad \frac{\begin{matrix}\Delta_0'\\ At\end{matrix}}{\exists x A x}$$

Then $\nu(\Delta_0') < \nu(\Delta_0)$. By IH Δ' is stable. Suppose

$$\Delta \quad \succsim \quad \Delta' \quad \equiv \quad \begin{matrix}\Delta_0\\ At\end{matrix}$$

Then Δ' is stable by assumption. So By Lemma 3.1 Δ is stable. ☐

Lemma 3.6. *Let*

$$\Pi \quad \equiv \quad \frac{\Pi_0 \quad \Pi_1}{A}\ (\rho)$$

where ρ is an elimination rule other than $\exists E$ or $\vee E$. If Π_0 and Π_1 are stable, then Π is stable.

Proof. 1. Inner Reduction.

By induction on $\nu(\Pi_0) + \nu(\Pi_1)$. Note that $<$ is exactly $\succ$ for all derivation ending with elimination rules. Let

$$\Pi \quad \succ \quad \Pi' \quad \equiv \quad \frac{\Pi_0' \quad \Pi_1'}{A}$$

where $\Pi_0 \succ \Pi_0'$ and $\Pi_1 \succ \Pi_1'$. By assumption Π_0' and Π_1' are stable and since $\nu(\Pi_0') + \nu(\Pi_1') < \nu(\Pi_0) + \nu(\Pi_1)$, by IH Π' is stable. By Lemma 3.1 Π is stable.

2. Detour Reduction.

Take the case $\rho = \forall E$. Let

$$\Pi \quad \equiv \quad \cfrac{\cfrac{\begin{matrix}\Sigma(a)\\ Aa\end{matrix}}{\forall x A x}}{At} \quad \succ \quad \begin{matrix}\Sigma(t)\\ At\end{matrix}$$

Since

$$\cfrac{\begin{matrix}\Sigma(a)\\ Aa\end{matrix}}{\forall x A x} \quad \succsim \quad \begin{matrix}\Sigma(t)\\ At\end{matrix}$$

and

$$\cfrac{\begin{matrix}\Sigma(a)\\ Aa\end{matrix}}{\forall x A x}$$

is stable by assumption, $\Sigma(t)$ is stable by Lemma 3.1. By Lemma 3.1 again Π is stable.

□

Lemma 3.7. *Let*

$$\Pi \quad \equiv \quad \frac{\Pi_0 \quad \Pi_1 \quad \Pi_2}{A}\ (\vee E)$$

If Π_0, Π_1 and Π_2 are stable, then Π is stable.

Proof. Analogous to Lemma 3.6. □

Lemma 3.8. *Let*

$$\Pi \quad \equiv \quad \cfrac{\begin{array}{c}\Pi_0\\ \exists x A x\end{array} \quad \begin{array}{c}[A(a)]\\ \Pi_1(a)\\ B\end{array}}{B}\,(\exists E)$$

If Π_0 *is stable and for every* t $\Pi_1(t)$ *is stable under* Π_0 *at* $[At]$*, then* Π *is stable.*

Proof. c.f. [Lei75] □

3.2 Main Theorems

Theorem 3.1. *Let*

$$\Delta \quad \equiv \quad \frac{\Delta_0 \quad (\Delta_1)}{A}\,(\rho)$$

where ρ *is an introduction rule. If* Δ_0 *(and* Δ_1*) are* ***s.s.****, then* Δ *is* ***s.s.****.*

Proof. For ρ is $\vee I$, $\& I$ or $\rightarrow I$ it is an outright application of defintion. We show that case where ρ is $\exists I$. In this case Δ_1 is empty. Let

$$\Delta \quad \equiv \quad \cfrac{\begin{array}{c}\Delta_0\\ At\end{array}}{\exists x A x}\,(\exists I)$$

where

$$\begin{array}{c}\Delta_0\\ At\end{array}$$

is **s.s.**, and

$$\Delta \quad \mapsto \quad \Delta^* \quad \equiv \quad \cfrac{\begin{array}{c}\Delta_0^*\\ A^* t\end{array}}{\exists x A^* x}\,(\exists I)$$

Since

$$\begin{array}{c}\Delta_0\\ At\end{array} \quad \mapsto \quad \begin{array}{c}\Delta_0^*\\ A^* t\end{array}$$

it follows that

$$\begin{array}{c}\Delta_0^*\\ A^* t\end{array}$$

is stable and then by Lemma 3.5 Δ^* is stable and so Δ is **s.s.**. □

Theorem 3.2. *Let*

$$\Pi \quad \equiv \quad \frac{\Pi_0 \quad \Pi_1 \quad \Pi_2}{A}\,(\rho)$$

where ρ *is an elimination rule. If* Π_0*,* Π_1 *and* Π_2 *are* ***s.s.****, then* Π *is* ***s.s.****.*

Proof. We show the case where ρ is $\exists E$. Let

$$\Pi \quad \equiv \quad \dfrac{\dfrac{\begin{array}{c}\Sigma\\ Bt\end{array}}{\exists x Bx} \qquad \begin{array}{c}[B(a)]\\ \Pi_1(a)\\ A\end{array}}{A}\,(\exists E)$$

$$\Pi \quad \mapsto \quad \Pi^* \quad \equiv \quad \dfrac{\begin{array}{c}\Pi_0^*\\ \exists x Bx\end{array} \qquad \begin{array}{c}[B(a)]\\ \Pi_1^*(a)\\ A\end{array}}{A}\,(\exists E)$$

where

$$\Pi_0 \quad \equiv \quad \dfrac{\begin{array}{c}\Sigma\\ Bt\end{array}}{\exists x Bx} \quad \mapsto \quad \Pi_0^*$$

$$\Pi_1 \quad \mapsto \quad \begin{array}{c}[B(a)]\\ \Pi_1^*(a)\\ A\end{array}$$

Since a appears free in any assumptions of Π, we have

$$\Pi_1 \quad \mapsto \quad \begin{array}{c}\Sigma\\ {[B(t)]}\\ \Pi_1^*(t)\\ A\end{array}$$

for any term t and stable derivation

$$\begin{array}{c}\Sigma\\ Bt\end{array}$$

Without loss assume that

$$\Pi_0^* \quad \succ \quad \dots \quad \succ \quad \begin{array}{c}\Theta\\ Bt\\ \exists x Bx\end{array} \quad \succsim \quad \begin{array}{c}\Theta\\ Bt\end{array}$$

By assumption Π_0 is **s.s.**, so Π_0^* is stable and then

$$\begin{array}{c}\Theta\\ Bt\end{array}$$

is stable. It follows that

$$\begin{array}{c}\Theta\\ {[Bt]}\\ \Pi_1^*(t)\\ A\end{array}$$

is stable. So $\Pi_1^*(t)$ is stable under Π_0^* at $[Bt]$ for any term t. By Lemma 3.8

$$\Pi^* \quad \equiv \quad \dfrac{\begin{matrix} \\ \\ \Pi_0 \\ \exists x B x \end{matrix} \qquad \begin{matrix} \Sigma \\ [B(t)] \\ \Pi_1(t) \\ A \end{matrix}}{A}\ (\exists E)$$

is stable and hence Π is **s.s.**. □

Theorem 3.3. *Every derivation* Π *is* ***s.s.****.*

Proof. By induction on derivation length $\lambda(\Pi)$. If $\lambda(\Pi) = 1$ the theorem follows from the definition of **s.s.**. If Π ends with an introduction rule, it follows from Theorem 3.1 and if Π ends with an elimination rule, it follows from Theorem 3.2. □

Corollary 3.1 (Strong Normalization)**.** *Every derivation is* ***s.n.****.*

Proof. By Theorem 3.3 and Lemma 3.2 □

References

[Lei75] Daniel Leivant. Strong normalization for arithmetic (variations on a theme of prawitz). *Lecture Notes in Mathematics*, 500, 1975.

[TS00] A.S. Troelstra and H. Schwichtenberg. *Basic Proof Theory.* Cambridge University Press, 2000.