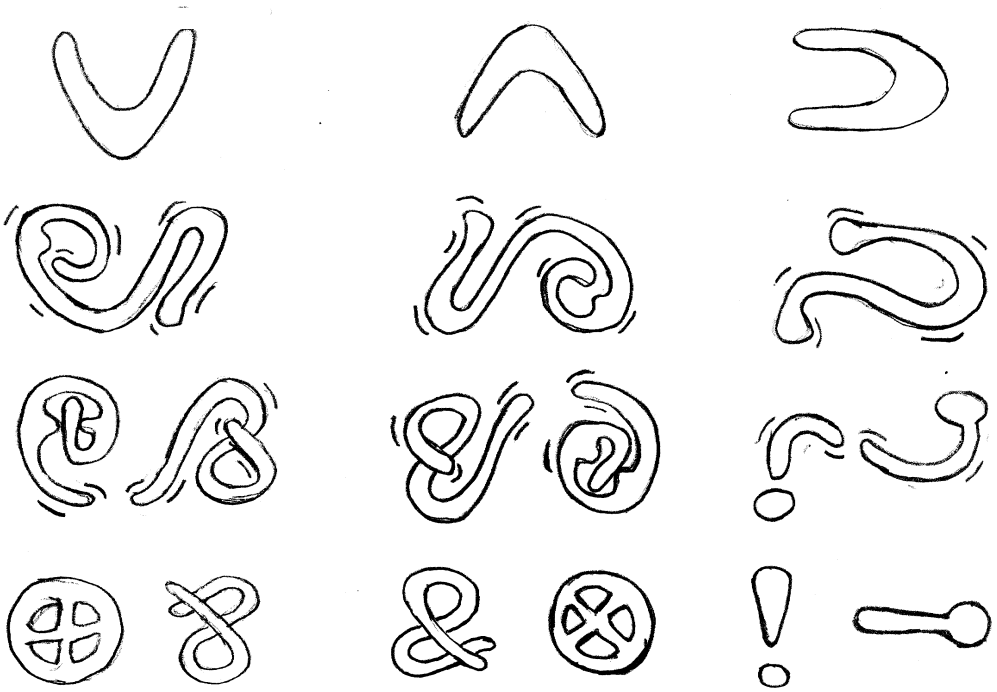
Chapter 2

Logical Embeddings into Linear Logic



2.1 Recovering Intuitionistic Logic

The exponential connectives allow us to recover the full power of Intuitionistic Logic: we can provide a faithful translation * of intuitionistic formulae and proofs into linear ones, in such a way that if a sequent $\Gamma \vdash A$ of LJ is provable, then its translation $\Gamma^\star \vdash A^\star$ is provable.

Notice that, for simplicity, we will give the translation just for LJ *without* the right weakening rule: this fragment is largely sufficient both to encode the λ-calculus without explicit escape constructs and to faithfully encode the full classical sequent calculus, as we will see in the next section. If one really wants to handle also the right weakening, then an LJ-sequent $\Gamma \vdash A$ should be translated into something like $!\Gamma \vdash ?!A$: we do not consider this extra burden worthwhile, and suggest that the interested reader use the translation of LK into LL instead[1].

Translation of formulae.

If A is atomic, then $A^\star = A$.

$$\begin{array}{lcl} (A \Rightarrow B)^\star & = & ?(A^\star)^\perp \parr !B^\star \;=\; (!A^\star) \multimap !B^\star \\ (A \wedge B)^\star & = & !A^\star \& !B^\star \\ (A \vee B)^\star & = & !A^\star \oplus !B^\star \\ (\neg A)^\star & = & ?(A^\star)^\perp \end{array}$$

Remark 2.1.1 *Exponentials are used in the translation; they are in charge of non linear effects. The first equation is remarkable, it is as if linear logic were a looking-glass through which the intuitionistic implication appears as a compound connective. Notice that the translation suggested for the conjunction is for its additive presentation: if you want to take the multiplicative presentation, than you should better try* $!A \otimes !B$.

Conventions for sequents in LL.

Let's write $\Gamma \vdash \Delta$ for $\vdash \Gamma^\perp, \Delta$.
The idea behind this translation from LJ to LL is as follows:

$$\begin{array}{ccc} \Pi & & \Pi^\star \\ \vdots & & \vdots \\ \Gamma \vdash A & \rightsquigarrow^\star & !\Gamma^\star \vdash !A^\star \end{array}$$

Now we define a translation of LJ-proofs by induction on the structure of proofs in LJ.

[1] It is also possible to give a translation where ? is used only for weakening, see [Jac94].

axiom

$$\frac{}{A \vdash A} \quad \overset{\star}{\rightsquigarrow} \quad \frac{}{!A^\star \vdash !A^\star}$$

cut

$$\frac{\begin{array}{cc}\Pi_1 & \Pi_2 \\ \vdots & \vdots \\ \Gamma, A \vdash C & \Gamma' \vdash A\end{array}}{\Gamma, \Gamma' \vdash C} \quad \overset{\star}{\rightsquigarrow} \quad \frac{\begin{array}{cc}\Pi_1^\star & \Pi_2^\star \\ \vdots & \vdots \\ !\Gamma^\star, !A^\star \vdash !C^\star & !\Gamma'^\star \vdash !A^\star\end{array}}{!\Gamma, !\Gamma'^\star \vdash !C^\star}$$

weakening

$$\frac{\begin{array}{c}\Pi \\ \vdots \\ \Gamma \vdash C\end{array}}{\Gamma, A \vdash C} \quad \overset{\star}{\rightsquigarrow} \quad \frac{\begin{array}{c}\Pi^\star \\ \vdots \\ !\Gamma^\star \vdash !C^\star\end{array}}{!\Gamma^\star, !A^\star \vdash !C^\star}$$

contraction

$$\frac{\begin{array}{c}\Pi \\ \vdots \\ \Gamma, A, A \vdash C\end{array}}{\Gamma, A \vdash C} \quad \overset{\star}{\rightsquigarrow} \quad \frac{\begin{array}{c}\Pi^\star \\ \vdots \\ !\Gamma^\star, !A^\star, !A^\star \vdash !C^\star\end{array}}{!\Gamma^\star, !A^\star \vdash !C^\star}$$

left arrow

$$\frac{\begin{array}{cc}\Pi_1 & \Pi_2 \\ \vdots & \vdots \\ \Gamma, B \vdash C & \Gamma' \vdash A\end{array}}{\Gamma, \Gamma', A \Rightarrow B \vdash C} \quad \overset{\star}{\rightsquigarrow} \quad \frac{\dfrac{\begin{array}{cc}\Pi_1^\star & \Pi_2^\star \\ \vdots & \vdots \\ !\Gamma^\star, !B^\star \vdash !C^\star & !\Gamma^\star \vdash !A^\star\end{array}}{!\Gamma^\star, !\Gamma'^\star, (!A^\star) \multimap (!B^\star) \vdash !C^\star}}{!\Gamma^\star, !\Gamma'^\star, !((!A^\star) \multimap (!B^\star)) \vdash C^\star}$$

right arrow

$$\frac{\begin{array}{c}\Pi \\ \vdots \\ \Gamma, A \vdash C\end{array}}{\Gamma \vdash A \Rightarrow C} \quad \overset{\star}{\rightsquigarrow} \quad \frac{\dfrac{\dfrac{\begin{array}{c}\Pi^\star \\ \vdots \\ !\Gamma^\star, !A^\star \vdash !C^\star\end{array}}{!\Gamma^\star \vdash (!A^\star) \multimap (!C^\star)}}{!\Gamma^\star \vdash !((!A^\star) \multimap (!C^\star))}}{}$$

Exercise 2.1.2 *Complete the translation of proofs treating the case of $\wedge$, $\vee$ and $\neg$.*

Remark 2.1.3 (Decoration) *Indeed, the translation of proofs we have exhibited has a very nice structural property (if you properly treat the cases of disjunction and conjunction in the previous*

exercise): if we forget about the exponentials and the eventually duplicated sequents, the underlying tree of the derivation (often called a skeleton*) is the same in the* LJ*-proof and in the translated* LL*-proof. Indeed, one can see the whole translation as a process of* decoration *of the intuitionistic proof with a bunch of exponentials. This is not the case, for example, of the more economical and well-known translation that we will present later.*

The translation of proofs we have just exhibited actually *proves* the following

Theorem 2.1.4 (Soundness) *Let A be an intuitionistic propositional formula of* LJ*. If A is provable in* LJ*, then $A^\star$ is provable in* LL*.*

2.2 More economical embeddings

The translation we just presented is often referred to as the "plethoric" translation , as it uses much more exponentials than what is strictly necessary (though it is indeed minimal if one asks for a uniform *decoration* procedure as defined above). Here we present another, more economical embedding, which is better known in the linear logic literature, but it should be noted that it is possible to give an embedding which is the most economical among all embeddings, as shown in [DJS93, DJS95a], and could be useful in compiling into a linear machine, as suggested in [Rov92]. Unlike the plethoric translation, though, this new translation will not respect the skeleton of a proof, most notably in the case of the arrow, which we will detail here.

Translation of formulae.

If A is atomic, then $A^* = A$.

$$\begin{array}{lcl} (A \Rightarrow B)^* & = & ?(A^*)^\perp \parr B^* \;=\; (!A^*) \multimap B^* \\ (A \wedge B)^* & = & A^* \& B^* \\ (A \vee B)^* & = & !A^* \oplus !B^* \\ (\neg A)^* & = & ?(A^*)^\perp \end{array}$$

The idea behind this translation from LJ to LL is as follows:

$$\begin{array}{ccc} \Pi & & \Pi^* \\ \vdots & & \vdots \\ \Gamma \vdash A & \overset{*}{\rightsquigarrow} & !\Gamma^* \vdash A^* \end{array}$$

Exercise 2.2.1 *Find a proof $\Phi(X,Y)$ in* LL *of $!((!X) \multimap Y) \vdash (!X) \multimap (!Y)$ and of $!(X \& Y) \vdash (!X) \& (!Y)$.*

Remark 2.2.2 (Savings balance) *It is interesting to remark that only on the connectives $\multimap$ and $\wedge$ we could save on exponentials, while you can show that such a savings is not really possible on the (notoriously proof-theoretically ugly) $\vee$ and $\neg$ connectives (that would imply, for example, to be able to prove something like $!(A \oplus B) \vdash !(!A \oplus !B)$, which is not likely). Indeed, one can really show that such savings are only possible on the connectives that are right-invertibles (see* **??** *for more details).*

Now we define the translation by induction on the structure of proofs in LJ.

axiom

$$\frac{}{A \vdash A} \quad \overset{*}{\rightsquigarrow} \quad \frac{\dfrac{}{A^* \vdash A^*}}{!A^* \vdash A^*}$$

cut

$$\frac{\begin{array}{cc}\Pi_1 & \Pi_2 \\ \vdots & \vdots \\ \Gamma, A \vdash C & \Gamma' \vdash A\end{array}}{\Gamma, \Gamma' \vdash C} \quad \overset{*}{\rightsquigarrow} \quad \frac{\begin{array}{cc}\Pi_1^* & \Pi_2^* \\ \vdots & \vdots \\ !\Gamma^*, !A^* \vdash C^* & !\Gamma'^* \vdash A^*\end{array}}{!\Gamma, !\Gamma'^* \vdash C^*}$$

left arrow

$$\frac{\begin{array}{cc}\Pi_1 & \Pi_2 \\ \vdots & \vdots \\ \Gamma, B \vdash C & \Gamma' \vdash A\end{array}}{\Gamma, \Gamma', A \Rightarrow B \vdash C} \quad \overset{*}{\rightsquigarrow} \quad \frac{\begin{array}{ccc} & & \Pi_2^* \\ & \Pi_1^* & \vdots \\ \Phi(A^*, B^*) & \vdots & !\Gamma^* \vdash A^* \\ \vdots & !\Gamma^*, !B^* \vdash C^* & \overline{!\Gamma^* \vdash\ !A^*} \\ !((!A^*) \multimap B^*) \vdash (!A^*) \multimap (!B^*) & !\Gamma^*, !\Gamma'^*, (!A^*) \multimap (!B^*) \vdash C^* & \end{array}}{!\Gamma^*, !\Gamma'^*, !((!A^*) \multimap B^*) \vdash C^*}$$

right arrow

$$\frac{\begin{array}{c}\Pi \\ \vdots \\ \Gamma, A \vdash C\end{array}}{\Gamma \vdash A \Rightarrow C} \quad \overset{*}{\rightsquigarrow} \quad \frac{\begin{array}{c}\Pi^* \\ \vdots \\ !\Gamma^*, !A^* \vdash C^*\end{array}}{!\Gamma^* \vdash\ !A^* \multimap C^*}$$

Exercise 2.2.3 *Complete the translation treating the case of* $\wedge$ *and* $\vee$.

The translation of proofs we have just exhibited actually proves the following

Theorem 2.2.4 (Soundness) *Let* A *be an intuitionistic propositional formula of* LJ*. If* A *is provable in* LJ*, then* A^* *is provable in* LL*.*

Exercise 2.2.5 *Try the translation:* $(A \wedge B)^* = A^* \otimes B^*$.

2.2.1 Completeness of the translation

We discuss here a simple proof of completeness of the translation for the propositional fragment of LJ with the connectives $\Rightarrow$ $\vee$ and $\wedge$, but no constants (like true and false) and no negation.

1. Recall that ILL, the Intuitionistic Linear Logic, is obtained from the symmetric presentation of LL by restricting the formulae on the right hand side of a sequent to at most one. Furthermore, we need only consider here the connectives: $\multimap$, $\&$, $\oplus$, and $!$. (Why?).

2. Show that if Π is a cut-free proof in LL of $\Gamma \vdash A$ where Γ and A are without $^{\perp}$, $\parr$ or constants, then it stays wholly in ILL. (hint: inspect each rule that can be performed on such a sequent and show they all are in ILL; in particular, this means checking that any rule that can be applied to derive in LL a sequent with exactly one consequent $\Gamma \vdash A$ can use as premisses only sequents with one consequent, hence still in ILL. In particular, notice that if a premiss of a left $\multimap$ rule has an empty succedent, then this premiss cannot be derived in the system without linear negation and constants).

3. Deduce from these facts the

   Theorem 2.2.6 (Completeness) *Let A be an intuitionistic propositional formula of LJ, If $A^\star$ is provable in LL, then it is provable in LJ.*

   Proof. Hint: show that the following read-back procedure $^\circ$ from ILL to LJ is *correct*.
   If A is atomic, then $A^\circ = A$.

$$\begin{array}{lcl} (!A)^\circ & = & A^\circ \\ (A \multimap B)^\circ & = & A^\circ \Rightarrow B^\circ \\ (A \& B)^\circ & = & A^\circ \wedge B^\circ \\ (A \oplus B)^\circ & = & A^\circ \vee B^\circ \end{array}$$

   □

Remark 2.2.7 (Schellinx) *Notice, though, that if we extend the translation to the full LJ system with the $\perp$ constant, and hence negation defined in the usual way, it is then rather more complex to show that this translation is complete. Schellinx observed that in the presence of the rules for 0, a derivation of an ILL sequent using only the connectives of ILL could have more than one consequent, as in the following case:*

$$\dfrac{\dfrac{\dfrac{A \vdash A \quad 0 \vdash A, A \multimap 0}{A, A \multimap 0 \vdash A, A \multimap 0} \quad 0 \vdash}{A, A \multimap 0, A \multimap 0 \vdash A \multimap 0}}{!A, !(A \multimap 0), !(A \multimap 0) \vdash A \multimap 0}}{!A, !(A \multimap 0) \vdash A \multimap 0}$$

Hence the proof we conducted above is no longer correct. The theorem is nonetheless true, and the interested reader is referred to [Sch91] for details on a correct, but more complex proof. It is possible to generalize the theorem to the full calculus with (first or second order) quantifiers.

2.3 Recovering Classical Logic

In this section, we will show how classical logic can be full and faithfully embedded into LL. To do so, we choose not to directly give the usual translation, but rather to obtain it via an embedding of LK into LJ which is also well behaved w.r.t. to the skeleton of a classical proof. As the reader will immediately notice, this does not mean that the skeleton is *exactly* the same, but only up to some intervening negations that are instrumental to keep the translation inside LJ.

2.3.1 The traditional embeddings: Gödel's translation

Definition et discussion du fait que cela ne marche pas trop bien parce que pas substitutif et pas preserve cut-free et modifie l'arbre (pas decoratif).

2.3.2 A decorative embedding of LK into LJ

We define a translation $\circ$ from classical formulae into intuitionistic formulae as follows:
If A is atomic, then $A^\circ = A$.

$$\begin{array}{lcl}(A \to B)^\circ & = & (\neg B^\circ) \Rightarrow (\neg A^\circ) \\ (A \wedge B)^\circ & = & (\neg\neg A^\circ \wedge \neg\neg B^\circ) \\ (A \vee B)^\circ & = & (\neg\neg A^\circ \vee \neg\neg B^\circ) \\ (\neg A)^\circ & = & \neg A^\circ\end{array}$$

The idea behind this translation from LK to LJ is as follows:

$$\begin{array}{ccc}\Pi & & \Pi^\circ \\ \vdots & & \vdots \\ \Gamma \vdash \Delta & \overset{\circ}{\rightsquigarrow} & \Gamma^\circ, \neg\Delta^\circ \vdash\end{array}$$

Remark 2.3.1 *This actually embeds LK in a fragment of LJ where no right weakening is used, and where all right introduction rule is immediately followed by a negation rule that moves the introduced formula to the left. We will note this fragment* LJ^-.

Now we define a translation of *derivations* by induction on the structure of proofs in LK.

axiom

$$\frac{}{A \vdash A} \quad \overset{\circ}{\rightsquigarrow} \quad \frac{\dfrac{}{A^\circ \vdash A^\circ}}{A^\circ, \neg A^\circ \vdash}$$

cut

$$\frac{\begin{array}{cc}\Pi_1 & \Pi_2 \\ \vdots & \vdots \\ \Gamma, A \vdash \Delta & \Gamma' \vdash A, \Delta'\end{array}}{\Gamma, \Gamma' \vdash \Delta, \Delta'} \quad \overset{\circ}{\rightsquigarrow} \quad \frac{\dfrac{\begin{array}{c}\Pi_1^\circ \\ \vdots \\ \Gamma^\circ, A^\circ, \neg\Delta^\circ \vdash\end{array}}{\Gamma^\circ, \neg\Delta^\circ \vdash \neg A^\circ} \quad \begin{array}{c}\Pi_2^\circ \\ \vdots \\ \Gamma'^\circ, \neg A^\circ, \neg\Delta'^\circ \vdash\end{array}}{\Gamma^\circ, \Gamma'^\circ, \neg\Delta^\circ, \neg\Delta'^\circ \vdash}$$

right weakening

$$\dfrac{\begin{array}{c}\Pi\\ \vdots\\ \Gamma \vdash \Delta\end{array}}{\Gamma \vdash A, \Delta} \quad \overset{\circ}{\rightsquigarrow} \quad \dfrac{\begin{array}{c}\Pi^\circ\\ \vdots\\ \Gamma^\circ, \neg\Delta^\circ \vdash\end{array}}{\Gamma^\circ, \neg A^\circ, \neg\Delta^\circ \vdash}$$

right contraction

$$\dfrac{\begin{array}{c}\Pi\\ \vdots\\ \Gamma \vdash A, A\Delta\end{array}}{\Gamma \vdash A, \Delta} \quad \overset{\circ}{\rightsquigarrow} \quad \dfrac{\begin{array}{c}\Pi^\circ\\ \vdots\\ \Gamma^\circ, \neg A^\circ, \neg A^\circ, \neg\Delta^\circ \vdash\end{array}}{\Gamma^\circ, \neg A^\circ, \neg\Delta^\circ \vdash}$$

left arrow

$$\dfrac{\begin{array}{c}\Pi_1\\ \vdots\\ \Gamma, B \vdash \Delta\end{array} \quad \begin{array}{c}\Pi_2\\ \vdots\\ \Gamma' \vdash A, \Delta'\end{array}}{\Gamma, \Gamma', A \to B \vdash \Delta, \Delta'} \quad \overset{\circ}{\rightsquigarrow} \quad \dfrac{\dfrac{\begin{array}{c}\Pi_1^\circ\\ \vdots\\ \Gamma^\circ, B^\circ, \neg\Delta^\circ \vdash\end{array}}{\Gamma^\circ, \neg\Delta^\circ \vdash \neg B^\circ} \quad \begin{array}{c}\Pi_2^\circ\\ \vdots\\ \Gamma^\circ, \neg A^\circ, \neg\Delta'^\circ \vdash\end{array}}{\Gamma^\circ, \Gamma'^\circ, (\neg B^\circ) \Rightarrow (\neg A^\circ), \neg\Delta^\circ, \neg\Delta'^\circ \vdash}$$

right arrow

$$\dfrac{\begin{array}{c}\Pi\\ \vdots\\ \Gamma, A \vdash B, \Delta\end{array}}{\Gamma \vdash A \to B, \Delta} \quad \overset{\circ}{\rightsquigarrow} \quad \dfrac{\dfrac{\dfrac{\begin{array}{c}\Pi^\circ\\ \vdots\\ \Gamma^\circ, A^\circ, \neg B^\circ, \neg\Delta^\circ \vdash\end{array}}{\Gamma^\circ, \neg B^\circ, \neg\Delta^\circ \vdash \neg A^\circ}}{\Gamma^\circ, \neg\Delta^\circ \vdash\ (\neg B^\circ) \Rightarrow (\neg A^\circ)}}{\Gamma^\circ, \neg((\neg B^\circ) \Rightarrow (\neg A^\circ)), \neg\Delta^\circ \vdash}$$

Exercise 2.3.2 *Complete the translation of proofs treating the trivial case of left contraction and weakening and the case of $\wedge$, $\vee$ and $\neg$.*

The translation of proofs we have just exhibited actually proves the following

Theorem 2.3.3 (Soundness) *Let A be a classical propositional formula of LK. If A is provable in LK, then A° is provable in LJ.*

The converse is also true, but unlike the case of the embedding of LJ into LL, the proof is trivial.

Theorem 2.3.4 (Completeness) *Let A be a classical propositional formula of LK. If A° is provable in LJ, then A is provable in LK.*

Proof. A derivation of A° in LJ is also a derivation in LK, since LJ is a subsystem of LK. Now any formula A is classically equivalent to A°, and the result trivially follows. □

Remark 2.3.5 (Preservation of cut-free proofs) *Notice that a cut-free proof is mapped by the translation into a cut-free proof, as it is clear from the soundness proof for the embedding. Actually, we will be able to say much more than this: it is possible to define a notion of cut-elimination in* LK *that is preserved by the translation, in such a way that the cut-elimination theorem for* LK *can be inferred from the one of* LJ*. This uses in an essential way the fact that the translation respects the skeleton.*

2.3.3 Embedding LK into LL

If we now take the composition $(_\circ)\star$ of the translations, we get back a translation from LK into LL that acts on formulae as follows:

$$(A \to B)^{\circ\star} \;=\; (\neg B^{\circ} \Rightarrow \neg A^{\circ})^{\star} \;= !(\neg B^{\circ})^{\star} \multimap !(\neg A^{\circ})^{\star} \;= !?(B^{\circ\star})^{\perp} \multimap !?(A^{\circ\star})^{\perp} \;= ?!A^{\circ\star} \multimap ?!B^{\circ\star}$$

and translates sequents $\Gamma \vdash \Delta$ as $!\Gamma^{\circ\star} \vdash ?!\Delta^{\circ\star}$.
Again this is a plethoric translation: it is of course possible to obtain a more economical one (indeed, even two, one of them allows to recover then the economical intuitionistic translation; you can find all this in [DJS95b]).

2.4 A normalization on LK

2.4.1 The q-protocol

In this subsection we define a normalization-scheme for LK, The q-protocol, so that the embeddings just defined will prove to be homomorphisms with respect to normalizations. That is this scheme will be simulable by LJ's one through the $\circ$ embedding, as well as by LL's one through the composition of both embeddings.
Consider a cut: $\frac{\vdash A \quad A \vdash}{\vdash}$, reduction proceeds via two possible types of steps, 'structural' ones, S1 and S2, and 'logical' ones, L ('key-steps'):
— An L-step applies when both A's are main in a logical rule, in which case we obtain as descendants one or two cuts on the immediate subformula(s) of the cutformula. (The order in which these cuts are applied is determined by the *orientation* of A's main connective.)
— In case no L-step is applicable, necessarily an S-step applies, which consists in 'transporting' *one* of the cut's subproofs up the tree of the cutformula's ancestors in the other one, duplicating it and contracting the context whenever passing an instance of contraction (or via the context of a binary additive rule); this process ends when reaching instances of introduction in an axiom, in which case the resulting 'axiom-cuts' are reduced *immediately*, when reaching an introduction by weakening, which are replaced by weakenings of the context formulas, or when reaching instances of introduction of the main connective of the cutformula.
Of course, now one needs to know *which* of the two subproofs has to move: move $A \vdash$ if A is not main in $\vdash A$ (S1); move $\vdash A$ if A is main in $\vdash A$ but not in $A \vdash$ (S2).
A symmetric protocol, the t-protocol, can be obtained just by asking $\vdash A$ to move first.
Let's define the *energy* of a cut c as the pair $(d(A), s(A))$ where $d(A)$ is the number of symbols in A and $s(A)$ is the type of the q-reduction-step applicable to c: S1, S2 or L. By definition of those steps, when applied to c, the energy of (the descendants of) c decreases (energies are lexicographically ordered, with S1 $>$ S2 $>$ L). (Except, of course, in a second order quantifier L-step, where the possible increase of $d(A)$ is out of control.)
At this point the reader might try his left hand at proving that the cut rule is redundant in first order LK. In fact the induction argument can be kept surprisingly simple: suppose π is a proof and c a cut in π with cutformula A. Now, to deduce from the remark above the usual key-lemma in a cut-elimination proof 'à la Gentzen', it suffices to show that the energy of (descendants of) any cut other than c in π never increases. This simple weak normalization proof is a foretaste of the 'good quality' of the q-reduction scheme.
A remark as to why we ask the *immediate* reduction of axiom-cuts appearing in the course of an S-step: this in fact is a 'sine qua non' for strong normalization. Consider the following example:

$$\dfrac{\dfrac{A \vdash A \quad A \vdash A}{A \vdash A} \qquad A \vdash A}{A \vdash A}$$

where not absorbing the axiom gives rise to a cyclic reduction.

2.4.2 Weak q-normalization proof.

We are now in a position to state a few interesting facts, but let's start by introducing some useful terminology.

Definition 2.4.1 (Main formula) *One says an occurrence of formula A in a proof π is* main *iff the rule above A is a logical rule introducing A.*

Definition 2.4.2 (Energy levels of a cut) *Let c be a cut in a derivation π and A be its cut-formula. The* potential energy *of c is the number of connectives and quantifiers in A.*
The kinetic energy *of c is:*

- *2 if none of its active formulas are main;*
- *1 if its attractive active formula is main and its non-attractive is not;*
- *0 if both active formulas are main.*

Kinetic energy of c is a local concept, only depending on the rules above c in π (potential energy is even more local). When a cut has kinetic energy zero it can't move anymore, whence the name. The *main active interspace* of A, an active formula, in π is the subtree of π that contains the tree of A. It is said to be *flat* when it contains no other rule than the one where A is main on top of the one where it is active.

Lemma 2.4.3 (Flatness stability) *A flat main-active interspace stays so by tq-normalization.*

Proof. Points where proofs are grafted (by cut) during normalization are leaves of formula-trees. But the sequent contained in a flat main-active interspace is a logical leaf of A's tree by definition, which is by flatness reduced to that leaf. □

Lemma 2.4.4 (Energy decreasing) *The kinetic energy level of a cut can only decrease under q-normalization.*

From this it easy to deduce that:

Theorem 2.4.5 (cut elimination) *The cut-rule is redundant in LK without second order quantifiers.*

Proof. Say the total energy of a cut is the couple (potential energy, kinetic energy) with the lexicographic ordering. Say the energy of a proof is that of the more energetic cuts. Then if chosen so that no equally energetic cut stands above it, a cut of maximal energy will be such that the corresponding reduction step will decrease the energy of the proof. Because new cuts (only appearing in case of a logical step) have less potential energy, and old cuts have less energy by the energy decreasing lemma. □

2.4.3 Strong normalization for cut-elimination in LK-q

Let's take our translation $_^\circ$ that maps LK into the special fragment LJ^- of LJ discussed above, and make the following little adjustment on the translation of the **cut** rules whose energy is ≤ 1:

$$\dfrac{\dfrac{\vdash A^\star}{\neg A^\star \vdash} \qquad \dfrac{A^\star \vdash}{\vdash \neg A^\star}}{\vdash} \quad \longrightarrow \quad \dfrac{\vdash A^\star \qquad A^\star \vdash}{\vdash}$$

In other words, when we are translating a cut whose right formula is the main formula of a logical rule, we adjust the translation in such a way that the **cut** in $\mathsf{LJ}^{-}$is still done on the main formula and not on its negation.
For this slightly modified translation, it is possible to show the following

Theorem 2.4.6 (Simulation) *Any step of q-cut-elimination in LK mapping a derivation π to π' can be simulated by a sequence of cut-elimination steps of LJ^{-} from π° to π'°.*

Notice that, as a consequence of 2.4.3, we have that

Corollary 2.4.7 (LJ^{-} stable under q-reduction) *A q-step performed on a LJ^{-} proof produces an LJ^{-} proof.*

and this together with the previous theorem allows us to prove the following

Corollary 2.4.8 (Reduction of strong normalization) *The q-cut-elimination protocol is strongly normalizing if the cut-elimination procedure in the fragment LJ^{-} (where proofs are mapped by $_^{\circ}$) is strongly normalizing.*

A similar reduction of normalization can be performed along the translatio $\star$ from LJ^{-} to LL, so that strong normalization for LK under the q-protocl can be reduced to strong normalization of cut-elimination in LL, which we will not prove now on the sequents, but rather on the underlying structure of LL, the *proof nets*, which are the subject of our next chapter. Nevertheless, we can state here the theorem:

Theorem 2.4.9 (Strong normalization of cut-elimination for LK) *The q-protocol for cut-elimination in LK is strongly normalizing.*

References: Parigot's $\lambda\mu$, Girard's **LC**, computational isos.