Lecture Notes on Quantification

15-816: Linear Logic
Frank Pfenning

Lecture 6
February 6, 2012

We introduce quantification into linear logic. The domains of the quantifiers are external to linear logic, which requires that we have a new external typing judgment. We analyze the quantifiers within the Curry-Howard interpretation of linear propositions as session types. The universal quantified represents data input, while the existential quantifier represents data output. More details can be found in a recent paper [CPT12].

1 Universal Quantification

We write $\forall n{:}\tau.A$, universally quantifying over objects of type τ. We will also refer to objects as *terms* (taking a syntactic point of view) or *values* (to contrast them with channels). We will specify little about the types τ, essentially giving a presentation that does not depend on how the new types and their inhabiting objects are chosen. However, we will make some requirements of this language so that the quantifiers indeed make sense.

We start with the left rule: how to we use a resource $\forall n{:}\tau.A$? Since it means that A is true for all objects of type τ, we can instantiate the quantifier with any particular object M. This suggests:

$$\frac{M : \tau \quad \Gamma\,;\,\Delta, A\{M/n\} \vdash C}{\Gamma\,;\,\Delta, \forall n{:}\tau.A \vdash C}\ \forall L$$

We will need to slightly generalize this presently.

The right rule answers the questions how we can prove that $\forall n{:}\tau.A$ is true. We proceed by showing $A\{m/n\}$ for a new parameter m of type τ. But where do we record the type of m? To be explicit about the type, and also

the parameters currently available in a proof, we introduce a new context Ψ which contains typings for term variables. We write:

$$\underbrace{m_1{:}\tau_1, \ldots, m_k{:}\tau_k}_{\Psi} \; ; \; \underbrace{u_1{:}B_1, \ldots, u_j{:}B_j}_{\Gamma} \; ; \; \underbrace{x_1{:}A_1, \ldots, u_i{:}A_i}_{\Delta} \vdash P :: z : C$$

where all variables are distinct, m are term variables, u are shared channels, and x are linear channels. Term variables stand for data that are *not* considered resources. They can be used arbitrarily often, including zero times. With this notation, the right rule becomes

$$\frac{\Psi, m{:}\tau \; ; \; \Gamma \; ; \; \Delta \vdash A\{m/n\}}{\Psi \; ; \; \Gamma \; ; \; \Delta \vdash \forall n{:}\tau.A} \; \forall R$$

The freshness condition is implicit, because we assume the rule can only be applied if m is not already declared in Ψ. We can always choose a fresh name. Note also that we currently do not have any way for n to actually appear in A, unless we allow some atomic propositions to depend on term variables. We will see an example of this in Section 7.

With this notation we can now generalize the left rule.

$$\frac{\Psi \vdash M : \tau \quad \Psi \; ; \; \Gamma \; ; \; \Delta, A\{M/n\} \vdash C}{\Psi \; ; \; \Gamma \; ; \; \Delta, \forall n{:}\tau.A \vdash C} \; \forall L$$

Here, the typing of M does not depend Γ or Δ, because we stipulate that terms cannot depend on channels of processes.

2 Harmony for Universal Quantification

We are, of course, obligated to check that the left and right rules are in harmony. This will impose some constraints on the judgment $\Psi \vdash M : \tau$. First reduction. The situation we must consider for reduction is

$$\dfrac{\dfrac{\Psi, m{:}\tau \; ; \; \Gamma \; ; \; \Delta \vdash A\{m/n\}}{\Psi \; ; \; \Gamma \; ; \; \Delta \vdash \forall n{:}\tau.A} \; \forall R \quad \dfrac{\Psi \vdash M : \tau \quad \Psi \; ; \; \Gamma \; ; \; \Delta', A\{M/n\} \vdash C}{\Psi \; ; \; \Gamma \; ; \; \Delta', \forall n{:}\tau.A \vdash C} \; \forall L}{\Psi \; ; \; \Gamma \; ; \; \Delta, \Delta' \vdash C} \; \mathsf{cut}_{\forall n:\tau.A}$$

We would like to reduce this to a cut of the two premises, but the formulas do not match: it is $A\{m/n\}$ in the first premise, and $A\{M/n\}$ in the second premise. How do we get one from the other? The solution is to *substitute*

M for m throughout the sequent $\Psi, m{:}\tau\ ;\ \Gamma\ ;\ \Delta \vdash A\{m/n\}$. Since m was chosen fresh (so it does not occur in Γ, Δ or A), we obtain $\Psi\ ;\ \Gamma\ ;\ \Delta \vdash A\{M/n\}$. We therefore need to postulate:

> **Substitution Principle.** If $\Psi \vdash M : \tau$ and $\Psi, m{:}\tau\ ;\ \Gamma\ ;\ \Delta \vdash A$ then $\Psi\ ;\ \Gamma\ ;\ \Delta \vdash A\{M/m\}$, provided Γ and Δ do not depend on m.

We will need to generalize this in order to verify it for our sequent calculus. For example, if A is a linear implication, after the $\multimap R$ rule the type of some channels may now depend on m even if the proviso is satisfied in the conclusion. We return to this when discussing functional interpretations.

As a notational device, we use a dashed line to indicate that we are applying some principle or theorem about proofs, rather than a rule of inference. For example, the dashed line in

$$\dfrac{\Psi \vdash M : \tau \quad \Psi, m{:}\tau\ ;\ \Gamma\ ;\ \Delta \vdash A}{\Psi\ ;\ \Gamma\{M/m\}\ ;\ \Delta\{M/m\} \vdash A\{M/m\}}\ \mathsf{subst}$$

indicates the application of the substitution principle, which is an assumption about how our typing judgment $\Psi \vdash M : \tau$ interacts with sequents. This is also called an *admissible rule* in the sense that whenever we have proofs of the premises then we can construct a proof of the conclusion. Using this notation, the evidence for cut reduction is:

$$\dfrac{\dfrac{\Psi, m{:}\tau\ ;\ \Gamma\ ;\ \Delta \vdash A\{m/n\}}{\Psi\ ;\ \Gamma\ ;\ \Delta \vdash \forall n{:}\tau.A}\ \forall R \qquad \dfrac{\Psi \vdash M : \tau \quad \Psi\ ;\ \Gamma\ ;\ \Delta', A\{M/n\} \vdash C}{\Psi\ ;\ \Gamma\ ;\ \Delta', \forall n{:}\tau.A \vdash C}\ \forall L}{\Psi\ ;\ \Gamma\ ;\ \Delta, \Delta' \vdash C}\ \mathsf{cut}_{\forall n:\tau.A}$$

$$\longrightarrow_R$$

$$\dfrac{\dfrac{\Psi \vdash M : \tau \quad \Psi, m{:}\tau\ ;\ \Gamma\ ;\ \Delta \vdash A}{\Psi\ ;\ \Gamma\ ;\ \Delta \vdash A\{M/n\}}\ \mathsf{subst} \qquad \Psi\ ;\ \Gamma\ ;\ \Delta', A\{M/n\} \vdash C}{\Psi\ ;\ \Gamma\ ;\ \Delta, \Delta' \vdash C}\ \mathsf{cut}_{A\{M/n\}}$$

Here $\Gamma\{M/n\} = \Gamma$ and $\Delta\{M/n\} = \Delta$ by the freshness of m in the $\forall R$ rule.

For the identity expansion, we have

$$\dfrac{}{\cdot\ ;\ \cdot\ ;\ \forall n{:}\tau.A \vdash \forall n{:}\tau.A}\ \mathsf{id}_{\forall n:\tau.A} \longrightarrow_E \dfrac{\dfrac{\dfrac{}{m{:}\tau \vdash m : \tau}\ \mathsf{hyp} \qquad \dfrac{}{m{:}\tau\ ;\ \cdot\ ;\ A\{m/n\} \vdash A\{m/n\}}\ \mathsf{id}_{A\{m/n\}}}{m{:}\tau\ ;\ \cdot\ ;\ \forall n{:}\tau.A \vdash A\{m/n\}}\ \forall L}{\cdot\ ;\ \cdot\ ;\ \forall n{:}\tau.A \vdash \forall n{:}\tau.A}\ \forall R$$

We see, that we need to assume a hypothesis rule or principle for the typing judgments of terms. Usually, this is simply a rule, but we do not want to fix this, since it is external to the sequent calculus.

Hypothesis. $m{:}\tau \vdash m : \tau$ for any variable m and type τ

We also need to allow hypotheses in Ψ to appear in applications of the identity rule in sequent calculus. In general, we can *weaken* any judgment with new typing assumptions $m{:}\tau$, because they do not need to be used. We apply this principle silently.

Weakening. If $\Psi \,;\, \Gamma \,;\, \Delta \vdash C$ then $\Psi, m{:}\tau \,;\, \Gamma \,;\, \Delta \vdash C$.

Here we suppose that m is not already declared in Ψ so that our presupposition about the judgments (no variable is declared more than once) remains satisfied. Analogous substitution and weakening principles also have to apply internally to the term typing judgment itself.

3 Existential Quantification

Existential quantification is somehow dual. In order to prove it, we have to supply some term of the correct type. In order to use it, we have to suppose some new parameter.

$$\frac{\Psi \vdash M : \tau \quad \Psi \,;\, \Gamma \,;\, \Delta \vdash A\{M/n\}}{\Psi \,;\, \Gamma \,;\, \Delta \vdash \exists n{:}\tau.A}\ \exists R \qquad \frac{\Psi, m{:}\tau \,;\, \Gamma \,;\, \Delta, A\{m/n\} \vdash C}{\Psi \,;\, \Gamma \,;\, \Delta, \exists n{:}\tau.A \vdash C}\ \exists L$$

Here, m must not already be declared in Ψ (and therefore not used in Γ, Δ, A, or C). We leave it to Exercise 3 to check the cut reduction and identity expansion properties.

4 Term Passing

So far, we have stayed very close to the π-calculus, establishing an interpretation of linear propositions as session types, sequent proofs as processes, and cut reduction as process reduction. Structural congruence arises from structural equivalences between sequent proofs. Next we incorporate the passing of data, or terms, rather than channels, by giving a process assignment for universal and existential quantification. The result is somewhat reminiscent of the applied π-calculus [AF01].

We now generalize all the sequent judgments so far by adding new hypotheses Ψ, assigning types to term variables, written as

$$\Psi \,;\, \Gamma \,;\, \Delta \vdash P :: x : A$$

Like the shared names in Γ, the typing assumptions in Ψ are propagated to all premises in all rules we have presented so far. For example, the identity and cut rules are now

$$\frac{}{\Psi \,;\, \Gamma \,;\, x{:}A \vdash [x \leftrightarrow z] :: z : A}\ \mathsf{id}_A$$

$$\frac{\Psi \,;\, \Gamma \,;\, \Delta \vdash P :: x : A \quad \Psi \,;\, \Gamma \,;\, \Delta', x{:}A \vdash Q :: z : C}{\Psi \,;\, \Gamma \,;\, \Delta, \Delta' \vdash (\nu x)(P \mid Q) :: z : C}\ \mathsf{cut}_A$$

5 Term Input

Input of terms is modeled simply by universal quantification.

$$\frac{\Psi, m{:}\tau \,;\, \Gamma \,;\, \Delta \vdash P\{m/n\} :: x : A\{m/n\}}{\Psi \,;\, \Gamma \,;\, \Delta \vdash x(n).P :: x : \forall n{:}\tau.A}\ \forall R$$

As before, the type of channel x evolves through interaction. In order for cut reduction to work correctly, the $\forall L$ rule must provide a matching output.

$$\frac{\Psi \vdash M : \tau \quad \Psi \,;\, \Gamma \,;\, \Delta', x{:}A\{M/n\} \vdash Q :: z : C}{\Psi \,;\, \Gamma \,;\, \Delta', x{:}\forall n{:}\tau.A \vdash \overline{x}\langle M\rangle.Q :: z : C}\ \forall L$$

Again, as before, we reuse the name x in the premise without conflict since x is linear. Note that m must be chosen fresh so that the new context $\Psi, m{:}\tau$ is well-formed in the $\forall R$ rule.

Applying cut to the right and left rules as formulated above yields the conclusion

$$\Psi \,;\, \Gamma \,;\, \Delta, \Delta' \vdash (\nu x)(x(n).P \mid \overline{x}\langle M\rangle.Q) :: z : C$$

To applying the usual reduction step from the sequent calculus, we must substitute M for n in the premise of the $\forall R$. We see that we need the substitution property for hypotheses in Ψ to justify reduction. After that we

obtain the following cut:

$$\frac{\begin{array}{l}\Psi\,;\,\Gamma\,;\,\Delta \vdash P\{M/n\} :: x : A\{M/n\} \\ \Psi\,;\,\Gamma\,;\,\Delta', x{:}A\{M/n\} \vdash Q :: z : C\end{array}}{\Psi\,;\,\Gamma\,;\,\Delta, \Delta' \vdash (\nu x)(P\{M/n\} \mid Q) :: z : C}\ \mathsf{cut}$$

from which we read off the reduction

$$(\nu x)(x(n).P \mid \overline{x}\langle M\rangle.Q) \longrightarrow (\nu x)(P\{M/n\} \mid Q)$$

In other words, we just use term passing instead of name passing in the π-calculus.

6 Term Output

A channel $x : \exists y{:}\tau.A$ offers to output a term M of type τ along x and then offer $A\{M/y\}$. This is symmetric to term input as described for $\forall y{:}\tau.A$. So even though our logic is intuitionistic, we obtain a strong duality between universal and existential quantification.

$$\frac{\Psi \vdash M : \tau \quad \Psi\,;\,\Gamma\,;\,\Delta \vdash P :: x : A\{M/n\}}{\Psi\,;\,\Gamma\,;\,\Delta \vdash \overline{x}\langle M\rangle.P :: x : \exists n{:}\tau.A}\ \exists R$$

$$\frac{\Psi, m{:}\tau\,;\,\Gamma\,;\,\Delta', x{:}A\{m/n\} \vdash Q\{m/n\} :: z : C}{\Psi\,;\,\Gamma\,;\,\Delta', x{:}\exists n{:}\tau.A \vdash x(n).Q :: z : C}\ \exists L$$

Applying cut to these two rules yields the conclusion

$$\Psi\,;\,\Gamma\,;\,\Delta, \Delta' \vdash (\nu x)(\overline{x}\langle M\rangle.P \mid x(n).Q) :: z : C$$

which is reduced with the same term-passing communication as for the $\forall R/\forall L$ pair:

$$(\nu x)(\overline{x}\langle M\rangle.P \mid x(n).Q) \longrightarrow (\nu x)(P \mid Q\{M/n\})$$

7 Example: An ATM

We now exercise our interpretation by building a model of a very simple ATM. Just being able to do a balance inquiry could be

$$\mathsf{Atm}_1 \triangleq \forall k{:}\mathsf{userid}.\ \exists n{:}\mathsf{val}.\ \mathbf{1}$$

Assuming we have a shared channel u:Atm_1, we can build a simple client that punches in a user id $'\mathsf{fp}'$ and obtains the balance.

$$u{:}\mathsf{Atm}_1\ ;\ \cdot \vdash (\nu a)\overline{u}\langle a\rangle.\ \overline{a}\langle'\mathsf{fp}'\rangle.\ a(n).a().\ \overline{z}\langle\rangle.\ \mathbf{0} :: z : \mathbf{1}$$

A slightly more complicated example would be to also offer a withdrawal of n dollars and then indicate failure (due, for example, to insufficient funds) or return the cash. Since cash is a resource that way data (like account balances) are not, we represent it as an ephemeral proposition $\mathsf{cash}(n)$ to indicate n dollars. This is an example where a quantified variable can actually appear.

$$\mathsf{Atm}_2 \triangleq \forall k{:}\mathsf{userid}.\ ((\exists n{:}\mathsf{val}.\ \mathbf{1})\ \&\ \forall n{:}\mathsf{val}.\ \mathbf{1} \oplus (\mathsf{cash}(n) \otimes \mathbf{1}))$$

A client that withdraws \$100 cash and puts into a wallet w, might look like this:

$$\begin{aligned} u{:}\mathsf{Atm}_2\ ;\ \cdot \vdash\ & (\nu a)\overline{u}\langle a\rangle.\ \overline{a}\langle'\mathsf{fp}'\rangle.\ a.\mathsf{inr}; \overline{a}\langle\$100\rangle. \\ & \quad a.\mathsf{case}(??, a(c).a().[c \leftrightarrow w]) :: w : \mathsf{cash}(\$100) \end{aligned}$$

However, we notice one problem: in case there are insufficient funds, we will actually not be able to full the goal of putting \$100 into our wallet. So we need to hedge our bets and say we are either putting \$100 into our wallet or nothing, where "nothing" represent by the logical constant $\mathbf{1}$.

$$\begin{aligned} u{:}\mathsf{Atm}_2\ ;\ \cdot\ \vdash\ & (\nu a)\overline{u}\langle a\rangle.\ \overline{a}\langle'\mathsf{fp}'\rangle.\ a.\mathsf{inr}; \overline{a}\langle\$100\rangle. \\ & \quad a.\mathsf{case}(w.\mathsf{inr}; a().\overline{w}\langle\rangle.\mathbf{0}, \\ & \qquad w.\mathsf{inl}; a(c).a().[c \leftrightarrow w]) \\ & :: w : \mathsf{cash}(\$100) \oplus \mathbf{1} \end{aligned}$$

The type of the channel c, input in the second to last line along channel a, will be $\mathsf{cash}(\$100)$. This is because we instantiated the universal quantifier on n with $\$100$. We leave the implementation of the ATM itself to Exercise 4.

The client's goal is not very expressive, in this and many other examples. Just consider the following typing:

$$u{:}\mathsf{Atm}_2\ ;\ \cdot \vdash w.\ \mathsf{inr}; \overline{w}\langle\rangle.\ \mathbf{0} :: w : \mathsf{cash}(\$100) \oplus \mathbf{1}$$

which represents a client that blithely decides to ignore the ATM and walk past it without carrying out any transaction at all. Of course, in that case there will also be no cash in the client's pocket.

This illustrates that the information in the process is critical. The type, while important, clearly does not fully specify the behavior. In particular, it doesn't prescribe the client's internal choices, which includes the trival inaction in this case.

Exercises

Exercise 1 Clearly, a service $x{:}\exists n{:}\tau.A$ should not be sufficient to satisfy goal $z{:}\forall m{:}\tau.A$. Illustrate how this fails because parameters must be chosen to be fresh.

Exercise 2 Under which conditions (if any) can a service $x : (\forall k{:}\tau.\exists n{:}\sigma.\mathbf{1})$ be used to provide service $y : (\exists n{:}\sigma.\forall k{:}\tau.\mathbf{1})$? How about the other direction? All services should be assumed to be linear.

Exercise 3 We explore here the existential quantifier from Section 3

(i) Write out the cut reduction.

(ii) Write out the identity expansion.

(iii) Check that the cut reduction is appropriately modeled by a term-passing communication.

Exercise 4 In this exercise we explore the ATM example from Section 7. You may assume the following persistent services offered by the bank:

$$\begin{array}{lcl} \mathsf{Balance} & \triangleq & \forall k{:}\mathsf{userid}.\,\exists n{:}\mathsf{val}.\,\mathbf{1} \\ \mathsf{Withdraw} & \triangleq & \forall k{:}\mathsf{userid}.\,\forall n{:}\mathsf{val}.\,\mathbf{1} \oplus \mathbf{1} \\ \mathsf{Deposit} & \triangleq & \forall k{:}\mathsf{userid}.\,\forall n{:}\mathsf{val}.\,\mathbf{1} \end{array}$$

In the type of Withdraw, the left alternative of $\oplus$ means there were insufficient funds and the transaction failed, while the right alternative of $\oplus$ means that the transaction succeeded. In that case, the ATM should only close the channel once the transaction with the client has completed.

Provide an implementation of a process P such that it offers the service Atm along channel x, using persistent services bal : Balance, wd : Withdraw, and dep : Deposit from the bank. In other words,

$$bal{:}\mathsf{Balance}, wd{:}\mathsf{Withdraw}, dep{:}\mathsf{Deposit}\ ;\ \cdot \vdash P :: x : \mathsf{Atm}_2$$

The process P should withdraw an additional \$2 service charge from the user account and deposit it under the userid 'atm' at the bank.

Exercise 5 This is a continuation of Exercise 4. Use cut! to compose your implementation of atm_2 with the client from Section 7 and an unspecified implementation of the banks services, and show the evolution of the composition until no further communication is possible. Assume that the bank

processes Q, Q' and Q'' communicate correctly, and that there are sufficient funds to withdraw $\$102$ ($\$100$ for the client, and $\$2$ for the ATM's service charge).

References

[AF01] Martín Abadi and Cédric Fournet. Mobile values, new names, and secure communication. In *28th Symposium on Principles of Programming Languages*, POPL'01, pages 104–115, London, United Kingdom, 2001. ACM.

[CPT12] Luís Caires, Frank Pfenning, and Bernardo Toninho. Towards concurrent type theory. In B. Pierce, editor, *Proceedings of the Workshop for Types in Language Design and Implementation*, TLDI'12, pages 1–12, Philadelphia, Pennsylvania, January 2012. ACM. Notes for an invited talk.