Handout for Lecture 5
Dale Miller, 16 April 2014

Note: Most of the the text in Sections 1 and 2 are from [4], and in Sections 3 and 4 are from [3].

1 Macro inference rules

Focused proof systems such as LKF allow us to change the size of inference rules with which we work. Let us call individual introduction rules "micro-rules". An entire phase within a focused proof can be seen as a "macro-rule". In particular, consider the following derivation.

$$\dfrac{\dfrac{\vdash \Theta, D \Uparrow N_1 \quad \cdots \quad \vdash \Theta, D \Uparrow N_n}{\vdash \Theta, D \Downarrow D}}{\vdash \Theta, D \Uparrow \cdot}$$

Here, the selection of the formula D for the focus can be taken as selecting among several macro-rules: this derivation illustrates one such macro-rule: the inference rule with conclusion $\vdash \Theta, D \Uparrow \cdot$ and with $n \geq 0$ premises $\vdash \Theta, D \Uparrow N_1, \ldots, \vdash \Theta, D \Uparrow N_n$ (where $N_1, \ldots, N_n$ are negative formulas). We shall say that this macro-rule is positive.

Similarly, there is a corresponding negative macro-rule with conclusion, say, $\vdash \Theta, D \Uparrow N_i$, and with $m \geq 0$ premises of the form $\vdash \Theta, D, C \Uparrow \cdot$, where C is a multiset of positive formulas or negative literals.

In this way, focused proofs allow us to view the construction of proofs from conclusions of the form $\vdash \Theta \Uparrow \cdot$ as first attaching a positive macro rule (by focusing on some formula in Θ) and then attaching negative inference rules to the resulting premises until one is again to sequents of the form $\vdash \Theta' \Uparrow \cdot$. Such a combination of a positive macro rule below negative macro rules is often called a *bipole* [1].

Focusing can be broken at any point via *delays*. Within LKF, we can define the delaying operators

$$\partial^+(B) = B \wedge^+ t^+ \quad \text{and} \quad \partial^-(B) = B \wedge^- t^-.$$

Clearly, B, $\partial^-(B)$, and $\partial^+(B)$ are all logically equivalent but $\partial^-(B)$ is always negative and $\partial^+(B)$ is always positive. If one wishes to break a positive macro rule resulting from focusing on a given positive formula into smaller pieces, then one can insert $\partial^-(\cdot)$ into that formula. Similarly, inserting $\partial^+(\cdot)$ can limit the size of a negative macro rule. By inserting many delay operators, a focused proof can be made to emulate an unfocused proof.

2 Fixed points and equality

In order for capture some interesting computational problems, the logic of propositional connectives and first-order quantifiers can be augmented with equality and fixed point operators. Consider the left and right introduction rules for $=$ and μ given in Figure 1. Notice that since the left and right introduction rules for μ are the same, μ is *self-dual*: that is, the De Morgan dual of μ is μ. It is possible to have a more expressive proof theory for fixed points that provides also for least and greatest fixed points (see, for example, [3, 2]): in that case, the De Morgan dual of the least fixed point is the greatest fixed point.

$$\frac{\Gamma, B(\mu B)\bar{t} \vdash \Delta}{\Gamma, \mu B\bar{t} \vdash \Delta} \qquad \frac{\Gamma \vdash \Delta, B(\mu B)\bar{t}}{\Gamma \vdash \Delta, \mu B\bar{t}}$$

$$\frac{\Gamma\sigma \vdash \Delta\sigma}{\Gamma, s = t \vdash \Delta}\ \dagger \qquad \frac{}{\Gamma, s = t \vdash \Delta}\ \ddagger \qquad \frac{}{\Gamma \vdash \Delta, t = t}$$

Figure 1: Introduction rules for = and μ. B is a formula with $n \geq 0$ variables abstracted and $\bar{t}$ is a list of n terms. The † proviso requires the terms s and t to be unifiable and σ to be their most general unifier. The ‡ proviso requires that the terms s and t are not unifiable.

$$\frac{\vdash \Theta\sigma \Uparrow \Gamma\sigma}{\vdash \Theta \Uparrow \Gamma, s \neq t}\ \dagger \qquad \frac{}{\vdash \Theta \Uparrow \Gamma, s \neq t}\ \ddagger \qquad \frac{}{\vdash \Theta \Downarrow t = t}$$

$$\frac{\vdash \Theta \Uparrow \Gamma, B(\mu B)\bar{t}}{\vdash \Theta \Uparrow \Gamma, \mu B\bar{t}} \qquad \frac{\vdash \Theta \Downarrow B(\mu B)\bar{t}}{\vdash \Theta \Downarrow \mu B\bar{t}}$$

Figure 2: Focused inference rules for = and μ. The proviso † and ‡ and the definition of σ are the same as above.

Example Identify the natural numbers as terms involving 0 for zero and s for successor. The following simple logic program defines two predicates on natural numbers.

$$\begin{aligned} \mathit{nat}\ 0 &\subset \mathit{true}. \\ \mathit{nat}\ (s\ X) &\subset \mathit{nat}\ X. \\ \mathit{leq}\ 0\ Y &\subset \mathit{true}. \\ \mathit{leq}\ (s\ X)\ (s\ Y) &\subset \mathit{leq}\ X\ Y. \end{aligned}$$

The predicate *nat* can be written as the fixed point

$$\mu(\lambda p \lambda x.(x = 0) \vee \exists y.(s\ y) = x \wedge p\ y)$$

and binary predicate *leq* (less-than-or-equal) can be written as the fixed point

$$\mu(\lambda q \lambda x \lambda y.(x = 0) \vee \exists u \exists v.(s\ u) = x \wedge (s\ v) = y \wedge q\ u\ v).$$

In a similar fashion, any Horn clause specification can be made into fixed point specifications (mutual recursions requires standard encoding techniques).

These two logical connectives can be added to LKF as follows. First, we classify both = and μ as positive connectives (this choice is forced for equality while μ can be polarized either way). The (one-sided) focused versions of the introduction rules above are given in Figure 2.

Example Consider proving the positive focused sequent

$$\vdash \Theta \Downarrow (\mathit{leq}\ m\ n \wedge^+ N_1) \vee^+ (\mathit{leq}\ n\ m \wedge^+ N_2),$$

where m and n are natural numbers and *leq* is the fixed point expression displayed above but this time with all occurrences of $\wedge$ and $\vee$ polarized with their positive variants. If both N_1 and N_2 are negative formulas,

then there are exactly two possible macro rules: one with premise $\vdash \Theta \Uparrow N_1$ when $m \leq n$ and one with premise $\vdash \Theta \Uparrow N_2$ when $n \leq m$ (thus, if $m = n$, either premise is possible). In this sense, a macro inference rule can contain an entire Prolog-style computation.

Example Macro rules can be built to match many computational situations. Consider, for example, defining simulation as the (greatest) fixed point of the equivalence

$$sim\ P\ Q \equiv \forall P' \forall A[P \xrightarrow{A} P' \supset \exists Q'[Q \xrightarrow{A} Q' \wedge sim\ P'\ Q']].$$

Although the right-hand-side of this definition looks complex, we show how it is possible to see proof search with this formula as being *exactly two* macro inference rules. First, the expression $P \xrightarrow{A} P'$ is, presumably, given via some SOS (structured operational semantic) specifications. Such specifications are simple, syntax-directed inference rules that can be captured as a least fixed point expression. As above, we will view such fixed point expressions as purely positive formulas. Thus, the expression $\forall P' \forall A[P \xrightarrow{A} P' \supset \cdot]$ is a negative macro rule: since all possible actions A and continuations P' must be computed, there are no choices to be made in building a proof for this expression. (Here, we are assuming that the implication $B \supset C$ is rendered as $\neg B \vee^{-} C$ in the polarized setting.) On the other hand, focusing on the expression $\exists Q'[Q \xrightarrow{A} Q' \wedge^{+} \cdot]$ yields a non-invertible, positive macro rule. In this way, the focused proof system is aligned directly with the structure of the actual (model-checking) problem. Notice that if one wishes to communicate a proof of a simulation to a proof checker, no information regarding the use of the negative macro rule needs to be communicated since the proof checker can also perform the computation behind that inference rule (*i.e.*, enumerating all possible transitions of a given process P).

3 Induction and co-induction

Proposition. The following inference rules are derivable:

$$\frac{}{\vdash P, P^{\perp}}\ init \qquad \frac{\vdash \Gamma, B(\nu B)\vec{t}}{\vdash \Gamma, \nu B\vec{t}}\ \nu R$$

MALL rules

$$\frac{}{\vdash \mathbf{1}} \qquad \frac{\vdash \Gamma, P \quad \vdash \Delta, Q}{\vdash \Gamma, \Delta, P \otimes Q} \qquad \frac{\vdash \Gamma, P, Q}{\vdash \Gamma, P \parr Q} \qquad \frac{\vdash \Gamma}{\vdash \Gamma, \perp}$$

$$\frac{}{\vdash \Delta, \top} \qquad \frac{\vdash \Gamma, P \quad \vdash \Gamma, Q}{\vdash \Gamma, P\ \&\ Q} \qquad \frac{\vdash \Gamma, P_i}{\vdash \Gamma, P_0 \oplus P_1}$$

First-order structure

$$\frac{\vdash \Gamma, Pt}{\vdash \Gamma, \exists x.Px} \qquad \frac{\vdash \Gamma, Pc}{\vdash \Gamma, \forall x.Px}\ c \text{ new}$$

$$\frac{}{\vdash t = t} \qquad \frac{\{\vdash \Gamma\theta : \theta \in csu(s \doteq t)\}}{\vdash \Gamma, s \neq t}$$

Fixed points (where S is closed, $\vec{x}$ is new)

$$\frac{\vdash \Gamma, B(\mu B)\vec{t}}{\vdash \Gamma, \mu B\vec{t}}\ \mu \qquad \frac{\vdash \Gamma, S\vec{t} \quad \vdash BS\vec{x}, (S\vec{x})^{\perp}}{\vdash \Gamma, \nu B\vec{t}}\ \nu \qquad \frac{}{\vdash \mu B\vec{t}, \nu \overline{B}\vec{t}}\ \mu\nu$$

Figure 3: Inference rules for μMALL$^{=}$

These results are standard, cf. [5]. The proof of the second one relies on monotonicity and is obtained by applying the ν rule with $B(\nu B)$ as the co-invariant.

Definition We classify as *asynchronous* (resp. *synchronous*) the connectives $\parr$, $\bot$, $\&$, $\top$, $\forall$, $\neq$, ν (resp. $\otimes$, $\mathbf{1}$, $\oplus$, $\mathbf{0}$, $\exists$, $=$, μ). A formula is said to be asynchronous (resp. synchronous) when its top-level connective is asynchronous (resp. synchronous). A formula is said to be *fully asynchronous* (resp. *fully synchronous*) when all of its connectives are asynchronous (resp. synchronous). Finally, a body $\lambda p \lambda \vec{x}.B p \vec{x}$ is said to be fully asynchronous (resp. fully synchronous) when the formula $Bp\vec{x}$ is fully asynchronous (resp. fully synchronous).

Notice, for example, that $\lambda p \lambda \vec{x}.p\vec{x}$ is fully asynchronous and fully synchronous.

Proposition The following structural rules are admissible provided that B is fully asynchronous:

$$\frac{\vdash \Gamma, \nu B\vec{t}, \nu B\vec{t}}{\vdash \Gamma, \nu B\vec{t}}\ \nu C \qquad \frac{\vdash \Gamma}{\vdash \Gamma, \nu B\vec{t}}\ \nu W$$

Hence, the following structural rules hold for any fully asynchronous formula P:

$$\frac{\vdash \Gamma, P, P}{\vdash \Gamma, P}\ C \qquad \frac{\vdash \Gamma}{\vdash \Gamma, P}\ W$$

Proposition. The following structural rules are admissible provided that B is fully asynchronous:

$$\frac{\vdash \Gamma, \nu B\vec{t}, \nu B\vec{t}}{\vdash \Gamma, \nu B\vec{t}}\ \nu C \qquad \frac{\vdash \Gamma}{\vdash \Gamma, \nu B\vec{t}}\ \nu W$$

Hence, the following structural rules hold for any fully asynchronous formula P:

$$\frac{\vdash \Gamma, P, P}{\vdash \Gamma, P}\ C \qquad \frac{\vdash \Gamma}{\vdash \Gamma, P}\ W$$

The rules for equality are not surprising. The main novelty here is the treatment of fixed points. Depending on the body, both μ and ν rules can be applied any number of times — but not with any co-invariant concerning ν. Notice for example that an instance of $\mu\nu$ can be η-expanded into a larger derivation, unfolding both fixed points to apply $\mu\nu$ on the recursive occurrences. As a result, each of the fixed point connectives has two rules in the focused system: one treats it as "an atom" and the other one as an expression with "internal structure."

Here, μ is treated during the synchronous phase and ν during the asynchronous phase. (Other choices are possible.) Roughly, what the focused system implies is that if a proof involving a ν-expression proceeds by co-induction on it, then this co-induction can be done at the beginning; otherwise that formula can be ignored in the whole derivation, except for the $\mu\nu$ rule. Focusing on a μ-expression yields two choices: unfolding or applying the initial rule for fixed points. If the body is fully synchronous, the focusing will never be lost. For example, if *nat* is the (fully synchronous) expression $\mu(\lambda nat.\lambda x.\ x = 0 \oplus \exists y.x = s\ y \otimes nat\ y)$, then focusing puts a lot of structure on a proof of $\Gamma \Downarrow nat\ t$: either t is a ground term representing a natural number and Γ is empty, or $t = s^n x$ for some $n \geq 0$ and Γ is $\{(nat\ x)^\perp\}$.

Theorem. The focused system is sound and complete with respect to $\mu\mathrm{MALL}^=$.

Asynchronous phase

$$\frac{\vdash \Gamma \Uparrow P, Q, \Delta}{\vdash \Gamma \Uparrow P \parr Q, \Delta} \quad \frac{\vdash \Gamma \Uparrow P, \Delta \quad \vdash \Gamma \Uparrow Q, \Delta}{\vdash \Gamma \Uparrow P \,\&\, Q, \Delta}$$

$$\frac{\vdash \Gamma \Uparrow \Delta}{\vdash \Gamma \Uparrow \bot, \Delta} \quad \frac{}{\vdash \Gamma \Uparrow \top, \Delta} \quad \frac{\{\vdash \Gamma\theta \Uparrow \Delta\theta : \theta \in csu(s \doteq t)\}}{\vdash \Gamma \Uparrow s \neq t, \Delta}$$

$$\frac{\vdash \Gamma \Uparrow Pc, \Delta}{\vdash \Gamma \Uparrow \forall x.Px, \Delta}\ c \text{ new}$$

$$\frac{\vdash \Gamma \Uparrow S\vec{t}, \Delta \quad \vdash\Uparrow BS\vec{x}, S\vec{x}^{\perp}}{\vdash \Gamma \Uparrow \nu B\vec{t}, \Delta}\ \vec{x} \text{ new} \quad \frac{\vdash \Gamma, \nu B\vec{t} \Uparrow \Delta}{\vdash \Gamma \Uparrow \nu B\vec{t}, \Delta}$$

Synchronous phase

$$\frac{\vdash \Gamma \Downarrow P \quad \vdash \Gamma' \Downarrow Q}{\vdash \Gamma, \Gamma' \Downarrow P \otimes Q} \quad \frac{\vdash \Gamma \Downarrow P_i}{\vdash \Gamma \Downarrow P_0 \oplus P_1}$$

$$\frac{}{\vdash\Downarrow \mathbf{1}} \quad \frac{}{\vdash\Downarrow t = t}$$

$$\frac{\vdash \Gamma \Downarrow Pt}{\vdash \Gamma \Downarrow \exists x.Px}$$

$$\frac{\vdash \Gamma \Downarrow B(\mu B)\vec{x}}{\vdash \Gamma \Downarrow \mu B\vec{x}} \quad \frac{}{\vdash \nu\overline{B}\vec{x} \Downarrow \mu B\vec{x}}$$

Switching (where P is synchronous, Q asynchronous)

$$\frac{\vdash \Gamma, P \Uparrow \Delta}{\vdash \Gamma \Uparrow P, \Delta} \quad \frac{\vdash \Gamma \Downarrow P}{\vdash \Gamma, P \Uparrow} \quad \frac{\vdash \Gamma \Uparrow Q}{\vdash \Gamma \Downarrow Q}$$

Figure 4: A focused proof-system for μMALL$^{=}$

4 Examples

We shall now give a few theorems in μMALL$^{=}$. Although we do not give their derivations here, we stress that all of these examples are proved naturally in the focused proof system. The reader will also note that although μMALL$^{=}$ is linear, these derivations are intuitive and their structure resemble that of proofs in intuitionistic logic.

We first define a few least fixed points expressing basic properties of natural numbers. We assume two constants z and s of respective types n and $n \to n$. Note that all these definitions are fully synchronous.

$$\begin{aligned}
nat &\stackrel{def}{=} \mu(\lambda nat \lambda x.\ x = z \oplus \exists y.\ x = s\ y \otimes nat\ y) \\
even &\stackrel{def}{=} \mu(\lambda even \lambda x.\ x = z \oplus \exists y.\ x = s\ (s\ y) \otimes even\ y) \\
plus &\stackrel{def}{=} \mu(\lambda plus \lambda a \lambda b \lambda c.\ a = z \otimes b = c \\
&\qquad\qquad \oplus \exists a' \exists c'.a = s\ a' \otimes c = s\ c' \otimes plus\ a'\ b\ c') \\
leq &\stackrel{def}{=} \mu(\lambda leq \lambda x \lambda y.\ x = y \oplus \exists y'.\ y = s\ y' \otimes leq\ x\ y') \\
half &\stackrel{def}{=} \mu(\lambda half \lambda x \lambda h.\ (x = z \oplus x = s\ z) \otimes h = z \\
&\qquad\qquad \oplus \exists x' \exists h'.\ x = s\ (s\ x') \otimes h = s\ h' \otimes half\ x'\ h')
\end{aligned}$$

The following statements are theorems, all of which can be proved by induction. The main insights required for proving these theorems involve deciding which fixed point expression should be introduced by induction: the proper invariant is not the difficult choice here since the context itself is adequate in these cases.

$$\begin{aligned}
&\vdash \forall x.\ nat\ x \multimap even\ x \oplus even\ (s\ x) \\
&\vdash \forall x.\ nat\ x \multimap \forall y \exists z.\ plus\ x\ y\ z \\
&\vdash \forall x.\ nat\ x \multimap plus\ x\ z\ x \\
&\vdash \forall x.\ nat\ x \multimap \forall y.\ nat\ y \multimap \forall z.\ plus\ x\ y\ z \multimap nat\ z
\end{aligned}$$

In the last theorem, the assumption $(nat\ x)^\perp$ is not needed and can be weakened (see earlier Proposition). In order to prove $(\forall x.\ nat\ x \multimap \exists h.\ half\ x\ h)$ one has to use a complete induction, *i.e.*, use the strengthened invariant $(\lambda x.\ nat\ x \otimes \forall y.\ leq\ y\ x \multimap \exists h.\ half\ y\ h)$.

A typical example of co-induction involves the simulation relation. Assume that $step : state \to label \to state \to o$ is an inductively defined relation encoding a labeled transition system. Simulation can be defined using the definition

$$sim \stackrel{def}{=} \nu(\lambda sim \lambda p \lambda q.\ \forall a \forall p'.\ step\ p\ a\ p' \multimap \exists q'.\ step\ q\ a\ q' \otimes sim\ p'\ q').$$

Reflexivity of simulation $(\forall p.\ sim\ p\ p)$ is proved easily by co-induction with the co-invariant $(\lambda p \lambda q.\ p = q)$. Instances of *step* are not subject to induction but are treated "as atoms". Proving transitivity, that is,

$$\forall p \forall q \forall r.\ sim\ p\ q \multimap sim\ q\ r \multimap sim\ p\ r$$

is done by co-induction on $(sim\ p\ r)$ with the co-invariant $(\lambda p \lambda r.\ \exists q.\ sim\ p\ q \otimes sim\ q\ r)$. The focus is first put on $(sim\ p\ q)^\perp$, then on $(sim\ q\ r)^\perp$. The fixed points $(sim\ p'\ q')$ and $(sim\ q'\ r')$ appearing later in the proof are treated "as atoms", as are all negative instances of *step*.

Except for the totality of *half*, all these theorems seem simple to prove using a limited number of heuristics. For example, one could first try to treat fixed points "as atoms", an approach that would likely fail quickly if inappropriate. Second, depending on the "rigid" structure of the arguments to a fixed point expression, one might choose to either unfold the fixed point or attempt to use the surrounding context to generate an invariant.

References

[1] J.-M. Andreoli. Focussing and proof construction. *Annals of Pure and Applied Logic*, 107(1):131–163, 2001.

[2] D. Baelde. *A linear approach to the proof-theory of least and greatest fixed points*. PhD thesis, Ecole Polytechnique, Dec. 2008.

[3] D. Baelde and D. Miller. Least and greatest fixed points in linear logic. In N. Dershowitz and A. Voronkov, editors, *International Conference on Logic for Programming and Automated Reasoning (LPAR)*, volume 4790 of *LNCS*, pages 92–106, 2007.

[4] D. Miller. Finding unity in computational logic. In *Proceedings of the 2010 ACM-BCS Visions of Computer Science Conference*, ACM-BCS '10, pages 3:1–3:13. British Computer Society, Apr. 2010.

[5] A. Tiu. *A Logical Framework for Reasoning about Logical Specifications*. PhD thesis, Pennsylvania State University, May 2004.