Assignment 5

15-816: Linear Logic
Frank Pfenning

Due
Monday, April 9, 2012

This assignment consists of several, somewhat open-ended problems. **You should pick one of them to do.** If you would like to do a half-semester project instead, please submit your project proposal instead of the assignment. A project proposal should be 2–5 pages and map out background, motivation, technical approach, and expected outcome of your project.

You may do these assignments **by yourself or in pairs**. They are somewhat open-ended, so you have to use your judgment as to when you consider the homework completed. Feel free to contact the instructor when you have questions about the extent of a problem.

As usual, you are allowed and encouraged to use *all* resources (papers, lecture notes, technical reports) that you can find, but you must properly cite and acknowledge any resources you use.

Please submit this assignment as a PDF by email. LaTeX templates and macros that may be helpful are available on the course web pages, but you are not required to use them.

Exercise 1 (Substructural Operational Semantics) Specify, in SSOS form, fragments of a functional language with the following features. Your semantics should *not* use substitution, except of parameters (destinations) for variables.

1. Call-by-value functions (evaluated left-to-right)
2. Call-by-name functions
3. Call-by-need functions
4. Futures (function body and argument evaluate in parallel)

5. Eager pairs, evaluated left-to-right
6. Lazy pairs
7. Parallel pairs
8. Unit type
9. Disjoint sums
10. Empty type
11. Recursion
12. Recursive types
13. Mutable store (creating, reading, and updating cells)
14. Another feature of your choice

You are encouraged to implement and test your SSOS specification in CLF or Ollibot, but it is not required.

Exercise 2 (Rule Permutations) An inference rule can be permuted up in a proof past another rule application with a different principal formula if the inferences can take place in either order. For example, $\otimes L$ can be permuted upwards past $\multimap R$:

$$\dfrac{\dfrac{\Gamma \,;\, \Delta, A, B, C \vdash D}{\Gamma \,;\, \Delta, A, B \vdash C \multimap D}\ {\multimap}R}{\Gamma \,;\, \Delta, A \otimes B \vdash C \multimap D}\ {\otimes}L \qquad \longleftrightarrow \qquad \dfrac{\dfrac{\Gamma \,;\, \Delta, A, B, C \vdash D}{\Gamma \,;\, \Delta, A \otimes B, C \vdash D}\ {\otimes}L}{\Gamma \,;\, \Delta, A \otimes B \vdash C \multimap D}\ {\multimap}R$$

It is important to consider this only for different formulas. For example, the right-to-left direction of the conversion above is deemed to hold in general, even though in the sequent $\vdash (A \otimes B) \multimap C$ an application of $\multimap R$ always has to precede $\otimes L$. For the case above, we write $\otimes L \backslash {\multimap} R$ and (because this one can be interpreted also from right to left), ${\multimap} R \backslash \otimes L$. In some cases, such as $\otimes L \backslash \top R$, the lower inference disappears altogether, or may be duplicated (as in $\otimes L \backslash \& R$).

In the cut-free sequent calculus, identify all pairs of rules $X \backslash Y$ such that X can *not* be permuted upward over Y in general and provide a counterexample. You do not need to prove that all other pairs commute.

Can you conjecture or establish a connection to the classification of connectives as positive or negative? To focusing?

Exercise 3 (Confluence of Inversion) In our (full) focusing system, the focusing rules can only be applied to *stable* sequents, where linear antecedents must be negative propositions or positive atoms and the succedent must be a positive proposition or a negative atom. This means that in an arbitrary sequent without focus, all inversion steps must be applied before we can focus.

From a practical perspective it is important to observe that the order in which these inversion steps are applied is irrelevant: no matter how we decompose an arbitrary goal sequent, (a) inversion always terminates, and (b) we always arrive at the same collection of stable subgoal sequents.

Prove these two properties.

Exercise 4 (Identity for Focused Ordered Logic) Develop the proof of the identity property for focused ordered logic.

We suggest the following outline. For the proof of the identity property only, introduce a new form of antecedent $\langle A^+ \rangle$ and a new form of succedent $\langle A^- \rangle$. The defining property is identity:

$$\frac{}{\Gamma \,;\, \cdot \,;\, \langle A^+ \rangle \rightarrow [A^+]}\ \mathsf{id}\langle\,\rangle[\,] \qquad \frac{}{\Gamma \,;\, \cdot \,;\, [A^-] \rightarrow \langle A^- \rangle}\ \mathsf{id}[\,]\langle\,\rangle$$

In other words, the antecedent $\langle A^+ \rangle$ promises that right focus on A^+ will succeed, and symmetrically for $\langle A^- \rangle$.

We then have two new versions of cut that should be admissible:

$$\frac{\Gamma \,;\, \Delta \,;\, \Omega \rightarrow [A^+] \quad \Gamma \,;\, \Delta' \,;\, \Omega_L, \langle A^+ \rangle, \Omega_R \rightarrow C}{\Gamma \,;\, \Delta, \Delta' \,;\, \Omega_L, \Omega, \Omega_R \rightarrow C}\ \mathsf{cut}[\,]\langle\,\rangle$$

$$\frac{\Gamma \,;\, \Delta \,;\, \Omega \rightarrow \langle A^- \rangle \quad \Gamma \,;\, \Delta' \,;\, \Omega_L, [A^-], \Omega_R \rightarrow C}{\Gamma \,;\, \Delta, \Delta' \,;\, \Omega_L, \Omega, \Omega_R \rightarrow C}\ \mathsf{cut}\langle\,\rangle[\,]$$

Key are two lemmas which show that permitting focus to succeed is admissible. This is encapsulated in the following two rules you'll have to show admissible:

$$\frac{\Gamma \,;\, \Delta \,;\, \Omega_L, \langle A^+ \rangle, \Omega_R \rightarrow C}{\Gamma \,;\, \Delta \,;\, \Omega_L, A^+, \Omega_R \rightarrow C}\ \langle\,\rangle L \qquad \frac{\Gamma \,;\, \Delta \,;\, \Omega \rightarrow \langle A^- \rangle}{\Gamma \,;\, \Delta \,;\, \Omega \rightarrow A^-}\ \langle\,\rangle R$$

The identity for positive propositions, for example, then is proven as follows:

$$\dfrac{\dfrac{\dfrac{}{\Gamma \,;\, \cdot \,;\, \langle A^+\rangle \to [A^+]}\ \mathsf{id}\langle\,\rangle[\,]}{\Gamma \,;\, \cdot \,;\, \langle A^+\rangle \to A^+}\ \mathsf{focus}R}{\Gamma \,;\, \cdot \,;\, A^+ \to A^+}\ \langle\,\rangle L$$

If you follow this outline, sketch the proof of admissibility of cut$[\,]\langle\,\rangle$ and cut$\langle\,\rangle[\,]$ and show the admissibility of $\langle\,\rangle L$ and $\langle\,\rangle R$ for the cases of positive and negative atoms, fuse ($A \bullet B$), unit ($\mathbf{1}$), right implication ($A \twoheadrightarrow B$), mobility ($¡A$), persistence ($!A$), and disjunction ($A \oplus B$).

Exercise 5 (Cut and Completeness for Focused Ordered Logic) Develop the cut and completeness properties for focusing in ordered logic. You only need to show some representative cases in the proofs—important is to sufficiently generalize the induction hypothesis and convey a sense of the overall structure of the proof. For the critical cases, concentrate on positive and negative atoms, fuse ($A \bullet B$), right implication ($A \twoheadrightarrow B$), mobility ($¡A$), and disjunction ($A \oplus B$).

One would expect the admissibility of cut to proceed as in the case of linear logic, as should the general form of the completeness argument. You will need to refer to the identity property as outlined in Exercise 4, but you don't need to prove that.

Exercise 6 (Strict Logic) Develop a proof theory for *strict logic* from first principles, that is, without embedding it in linear logic.

In a sequent calculus for strict logic we have three kinds of antecedents: A is unrestricted, A is *strict* (must be used at least once), and A is *ignored* (cannot be used). Ignored antecedents can only play a role in parts of a proof that are themselves ignored.

Provide a sequent calculus for strict logic and prove cut and identity properties. Classify the connectives and modal operators with respect to their polarity (positive or negative).

Informally compare strict logic to linear logic, in terms of available connectives, modal operators, and expressive power.

Exercise 7 (Strict λ-Calculus) Strict logic (see Exercise 6) is connected to functional programming where strict functions are those that definitely use their arguments, and ignored terms represent dead code. As a continuation or alternative to Exercise 6, provide a natural deduction system with proof terms for strict logic. State the necessary substitution properties for this system to make sense (you don't need to prove them).

Then give a substructural operational semantics for strict λ-calculus that does not use substitution, but binds variables to values or terms as appropriate. Your are encouraged, but now required, to implement your semantics in CLF or Ollibot.

Exercise 8 (Ordered Computation) Develop a Curry-Howard interpretation of ordered logic, where cut reduction corresponds to some form of computation. How does your interpretation and the mobility modality $¡A$ relate to the process interpretation of the linear sequent calculus?