Introduction to Linear Logic

Andres Ojamaa

23.02.2006

Outline

Classical Sequent Calculus

Structural rules of weakening and contraction:

- $\dfrac{\Gamma \vdash \Delta}{\Gamma, A \vdash \Delta}$ $\qquad$ $\dfrac{\Gamma \vdash \Delta}{\Gamma \vdash A, \Delta}$
- $\dfrac{\Gamma, A, A \vdash \Delta}{\Gamma, A \vdash \Delta}$ $\qquad$ $\dfrac{\Gamma \vdash A, A, \Delta}{\Gamma \vdash A, \Delta}$

- A fact can be used freely as many times as needed.
- What can be concluded from $A, A => B, A => C$?

Capitalistic Point of View

“A implies B” should be read as “give me *as many* A as I might need and I get you B”.

Linear Logic

- Proposed by Jean-Yves Girard in 1987
- Denies the structural rules of weakening and contraction
- Assumptions as consumable resources

There are other resource-oriented logics:

- Relevance logic
- Lambek calculus

Classical Linear Logic Sequent Calculus

$$\frac{}{A \vdash A}\ \text{init} \qquad \frac{\Gamma_1 \vdash A, \Sigma_1 \quad \Gamma_2, A \vdash \Sigma_2}{\Gamma_1, \Gamma_2 \vdash \Sigma_1, \Sigma_2}\ \text{cut}$$

$$\frac{\Gamma \vdash \Sigma}{\Gamma, !A \vdash \Sigma}\ \text{weak}_L \qquad \frac{\Gamma \vdash \Sigma}{\Gamma \vdash ?A, \Sigma}\ \text{weak}_R \qquad \frac{\Gamma, !A, !A \vdash \Sigma}{\Gamma, !A \vdash \Sigma}\ \text{contr}_L \qquad \frac{\Gamma \vdash ?A, ?A, \Sigma}{\Gamma \vdash ?A, \Sigma}\ \text{contr}_R$$

$$\frac{\Gamma, A, B \vdash \Sigma}{\Gamma, A \otimes B \vdash \Sigma}\ \otimes_L \qquad \frac{\Gamma_1 \vdash A, \Sigma_1 \quad \Gamma_2 \vdash B, \Sigma_2}{\Gamma_1, \Gamma_2 \vdash A \otimes B, \Sigma_1, \Sigma_2}\ \otimes_R \qquad \frac{\Gamma \vdash \Sigma}{\Gamma, \mathbf{1} \vdash \Sigma}\ \mathbf{1}_L \qquad \frac{}{\vdash \mathbf{1}}\ \mathbf{1}_R$$

$$\frac{\Gamma, A \vdash \Sigma}{\Gamma, A \,\&\, B \vdash \Sigma}\ \&_{L1} \qquad \frac{\Gamma, B \vdash \Sigma}{\Gamma, A \,\&\, B \vdash \Sigma}\ \&_{L2} \qquad \frac{\Gamma \vdash A, \Sigma \quad \Gamma \vdash B, \Sigma}{\Gamma \vdash A \,\&\, B, \Sigma}\ \&_R \qquad \text{no } \top_L \qquad \frac{}{\Gamma \vdash \top, \Sigma}\ \top_R$$

$$\frac{\Gamma_1, A \vdash \Sigma_1 \quad \Gamma_2, B \vdash \Sigma_2}{\Gamma_1, \Gamma_2, A \parr B \vdash \Sigma_1, \Sigma_2}\ \parr_L \qquad \frac{\Gamma \vdash A, B, \Sigma}{\Gamma \vdash A \parr B, \Sigma}\ \parr_R \qquad \frac{}{\bot \vdash}\ \bot_L \qquad \frac{\Gamma \vdash \Sigma}{\Gamma \vdash \bot, \Sigma}\ \bot_R$$

$$\frac{\Gamma, A \vdash \Sigma \quad \Gamma, B \vdash \Sigma}{\Gamma, A \oplus B \vdash \Sigma}\ \oplus_L \qquad \frac{\Gamma \vdash A, \Sigma}{\Gamma \vdash A \oplus B, \Sigma}\ \oplus_{R1} \qquad \frac{\Gamma \vdash B, \Sigma}{\Gamma \vdash A \oplus B, \Sigma}\ \oplus_{R2} \qquad \frac{}{\Gamma, \mathbf{0} \vdash \Sigma}\ \mathbf{0}_L \qquad \text{no } \mathbf{0}_R$$

$$\frac{\Gamma, A \vdash \Sigma}{\Gamma, !A \vdash \Sigma}\ !_L \qquad \frac{!\Gamma \vdash A, ?\Sigma}{!\Gamma \vdash !A, ?\Sigma}\ !_R \qquad \frac{!\Gamma, A \vdash ?\Sigma}{!\Gamma, ?A \vdash ?\Sigma}\ ?_L \qquad \frac{\Gamma \vdash A, \Sigma}{\Gamma \vdash ?A, \Sigma}\ ?_R$$

CLL sequent calculus

Linear Connective Soup

Multiplicative conjunction

- Operator: $\otimes$
- Denotes simultaneous occurrence of resources
- Unit: 1 ($1 \otimes A = A = A \otimes 1$)

Multiplicative disjunction

- Operator: $⅋$
- Represents simultaneous goals that must be reached
- Unit: $\perp$

Linear Connective Soup

Additive conjunction

- Operator: $\&$
- Internal choice, represents alternative occurrence of resources
- Unit: $\top$

Additive disjunction

- Operator: $\oplus$
- External choice, represents a choice over which one has no control
- Unit: 0

Linear Implication and Exponentials

The proposition $A \multimap B$ consumes resource A to reach resource B.

- Reuse is allowed for propositions using “of course” operator: !. (contraction)
- A fact can be weakened by additional conclusion $?A$ (“why not” operator).

Negation

Atomic formula:

- Negation of A is $A^\perp$
- Negation of $A^{\perp\perp}$ is A

Negation of non-atomic formulae is defined using the De Morgan rule:

- $(A \otimes B)^\perp = A^\perp \parr B^\perp$
- $(A \parr B)^\perp = A^\perp \otimes B^\perp$
- $(A \& B)^\perp = A^\perp \oplus B^\perp$
- $(A \oplus B)^\perp = A^\perp \& B^\perp$

Linear implication $A \multimap B$ is defined as a shorthand for $A^\perp \parr B$

The Linear Menu

Menu a 75 Frs

Entree:

- quiche lorraine ou
- saumon fume

et Plat:

- pot-au-feu ou
- filet de canard

et

- Fruit selon saison (banane ou raisin ou oranges ou ananas) ou
- Dessert au choix (mistere, glace, tarte aux pommes)

$$75FF^{\perp} \parr (Q \& S) \otimes (P \& F) \otimes ((B \oplus R \oplus O \oplus A) \& (M \& G \& T))$$

Common Variants of Linear Logic

- MLL - Multiplicative LL
  - Only $\otimes$ and $⅋$ are allowed
  - Decidable, NP-complete (Max I. Kanovich)
- MALL - Multiplicative Additive LL
  - Adds additive connectives ($\oplus$, $\&$) to MLL
  - Decidable, PSPACE-complete (P. Lincoln, J. Mitchell, A. Scedrov, N. Shankar)
- MELL - Multiplicative Exponential LL
  - Adds exponential operators to MLL
  - The decision problem is open
- MAELL - Multiplicative Additive Exponential LL
  - Undecidable

There are also first- and higher-order extensions of LL.

Synthesis of Web Services

- How to find solutions effectively?
- General description of a service:

$$resources \otimes constraints \otimes precontitions \otimes !inputs \multimap (effects \otimes !outputs) \oplus exception$$

- Example:

$$have\_processing\_time \otimes !x\_is\_known \otimes !y\_is\_known \multimap !z\_is\_known \oplus exception$$

Synthesis of Web Services

Using admissible rules

The extralogical axiom describing a service:

$$a \otimes !i \multimap (f \otimes !o) \oplus e$$

Where:

- a - multiplicative conjunction of resources, constraints and preconditions
- i - multiplicative conjunction of inputs
- f - multiplicative conjunction of effects
- o - multiplicative conjunction of outputs
- e - exception

Synthesis of Web Services

Using admissible rules

Admissible derivation rule:

$$\frac{\vdash a \otimes !i \multimap f \otimes !o \qquad \Gamma \vdash a \qquad \Sigma \vdash !i}{\Gamma, \Sigma \vdash f \otimes !u}$$

Where u consists of something from i, o.

- What about the complexity of the proof-search?

Theorem Provers: linprove

- Searches a cut-free proof of the given two-sided sequent of first-order linear logic
- Written in SICStus Prolog ($\approx$ 1400 LOC)
- Online demo: http://bach.istc.kobe-u.ac.jp/llprover/
- Author: Naoyuki Tamura

Example proof:

```
------- Ax
a --> a
---------- L/\1
a/\b --> a
```

Linear Logic Theorem Provers

- linprove - prover for propositional linear logic
  - Written in Scheme ($\approx$ 4000 LOC)
  - Proof Strategies in Linear Logic. 1994
  - Author: Tanel Tammet
- Forum
  - Based on intuitionistic linear logic
  - Designed by Dale Miller
- RAPS - Resource-Aware Planning System
  - LL planner to support reasoning over Web service composition problems in propositional and first-order LL
  - Written in Java
  - Peep Küngas

QML - Quantum Meta Language

QML is a functional language for quantum computations developed by T. Altenkirch and J. Grattage.

- Based on strict linear logic
  - SLL is an extension of LL with structural rule of contraction.
- Quantum control and quantum data
- Important issue: control of decoherence

Summary

- “I’m *not* a linear logician.” – Girard
- Linear Logic provides useful tools for different applications
- Expressive power vs complexity