Displayed Monoidal Categories for the Semantics of Linear Logic

Benedikt Ahrens   Ralph Matthes   **Niels van der Weide**
Kobe Wullaert

January 30, 2024

Linear Logic

Categorical Semantics of Linear Logic

Displayed Categories

Displayed Monoidal Categories

What is Linear Logic?

- Linear logic is "the logic of resources"
- Key feature of linear logic: **assumptions are used exactly once**
- Used for many applications (e.g., quantum physics, separation logic, domain theory)

Logic

- To describe linear logic, we give **connectives** and **derivation rules**
- Among the derivation rules, there are **structural rules**, **introduction rules**, and **elimination rules**
- Note: often sequent calculus is used for linear logic
- This talk: **natural deduction** (following a note by Pfenning[1] and the linear λ-calculus by Benton and Wadler)

[1] https://www.cs.cmu.edu/~fp/courses/15816-f01/handouts/lnd.pdf

Structural Rules

$$\text{Dup}\ \frac{\Gamma, A, A \vdash C}{\Gamma, A \vdash C}$$

$$\text{Weaken}\ \frac{\Gamma \vdash C}{\Gamma, A \vdash C}$$

$$\text{Hyp}\ \frac{}{\Gamma, A \vdash A}$$

$$\text{Sym}\ \frac{\Gamma, A, B \vdash C}{\Gamma, B, A \vdash C}$$

(a) Propositional logic

$$\text{Hyp}\ \frac{}{A \vdash A}$$

$$\text{Sym}\ \frac{\Gamma, A, B \vdash C}{\Gamma, B, A \vdash C}$$

(b) Linear logic

Note the difference between the Hyp rules!

Connectives

We consider a fragment of **intuitionistic linear logic**.
It has the following connectives:

- linear conjunction: $\otimes$
- linear implication: $\multimap$
- bang modality: ! (you can duplicate assumptions under a !)

People also consider other connectives for linear logic.

- why not modality: ?
- linear negation
- quantifiers

But we shall ignore them in this talk

Derivation Rules for $\otimes$

$$\wedge\text{I}\ \frac{\Gamma \vdash A \qquad \Gamma \vdash B}{\Gamma \vdash A \wedge B}$$

$$\otimes\text{E1}\ \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash A}$$

$$\otimes\text{E2}\ \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash B}$$

(a) Conjunction

$$\otimes\text{I}\ \frac{\Gamma \vdash A \qquad \Delta \vdash B}{\Gamma, \Delta \vdash A \otimes B}$$

$$\otimes\text{E}\ \frac{\Gamma \vdash A \otimes B \qquad \Delta, A, B \vdash C}{\Gamma, \Delta \vdash C}$$

(b) Linear conjunction

Derivation Rules for $\multimap$

$$\rightarrow\mathrm{I}\ \frac{\Gamma, A \vdash B}{\Gamma \vdash A \rightarrow B}$$

$$\rightarrow\mathrm{E}\ \frac{\Gamma \vdash A \rightarrow B \qquad \Gamma \vdash A}{\Gamma \vdash B}$$

(a) Implication

$$\multimap\mathrm{I}\ \frac{\Gamma, A \vdash B}{\Gamma \vdash A \multimap B}$$

$$\multimap\mathrm{E}\ \frac{\Gamma \vdash A \multimap B \qquad \Delta \vdash A}{\Gamma, \Delta \vdash B}$$

(b) Linear implication

Derivation Rules for !

$$\frac{\Gamma \vdash !A}{\Gamma \vdash A}$$

$$\frac{\Gamma \vdash !A \qquad \Delta \vdash B}{\Gamma, \Delta \vdash B}$$

$$\frac{\Gamma \vdash !A \qquad \Delta, !A, !A \vdash B}{\Gamma, \Delta \vdash B}$$

$$\frac{\Gamma_1 \vdash !A_1, \ \ldots, \ \Gamma_n \vdash !A_n \qquad !A_1, \ldots, !A_n \vdash B}{\Gamma_1, \ldots, \Gamma_n \vdash !B}$$

We can copy and discard assumptions under the bang modality

The Semantics of Linear Logic

- To relate the syntax to actual applications, we give denotational semantics
- For example, to relate linear logic to quantum mechanics, we interpret formulas as vector spaces
- Our tool for denotational semantics: **category theory**

Category Theory and Semantics

Curry-Howard-Lambek correspondence

Logic	Type theory	Category Theory
Formula	Type	Object
Proof	Term	Morphism
Connective	Type Constructor	Categorical structure

Category Theory and Semantics

Curry-Howard-Lambek correspondence

Logic	Type theory	Category Theory
Formula	Type	Object
Proof	Term	Morphism
Connective	Type Constructor	Categorical structure

Note: categorical structure is described via **universal properties** whereas connective/type constructors are described via **introduction and elimination rules**.

Categorical Semantics for Linear Logic

We discussed 3 connectives in our logic:

- linear conjunction: $\otimes$
- linear implication: $\multimap$
- bang modality: ! (you can duplicate assumptions under a !)

Categorical Semantics for Linear Logic

We discussed 3 connectives in our logic:

- linear conjunction: $\otimes$
- linear implication: $\multimap$
- bang modality: ! (you can duplicate assumptions under a !)

Let us start by looking at the semantics of linear conjunction and implication.

Monoidal Categories, what are they?

Basically: **Monoid**al **category** = **Monoid** + **category**

A **monoidal category** is a category with a **multiplication** $\otimes$.

- given objects x, y, we have an object $x \otimes y$
- given morphisms $f : x \to x'$ and $g : y \to y'$, we have a morphism $f \otimes g : x \otimes y \to x' \otimes y'$

We require $\otimes$ to be associative and unital in a weak sense.

Monoidal Categories and Linear Logic

$$\frac{x : \mathcal{C} \qquad y : \mathcal{C}}{x \otimes y : \mathcal{C}}$$

$$\frac{f : x \to x' \qquad g : y \to y'}{f \otimes g : x \otimes y \to x' \otimes y'}$$

(a) Monoidal Categories

$$\frac{A : \mathrm{Prop} \qquad B : \mathrm{Prop}}{A \otimes B : \mathrm{Prop}}$$

$$\frac{\Gamma \vdash A \qquad \Delta \vdash B}{\Gamma, \Delta \vdash A \otimes B}$$

(b) Linear Logic

Monoidal Categories and Linear Logic

$$\frac{x : \mathcal{C} \qquad y : \mathcal{C}}{x \otimes y : \mathcal{C}}$$

$$\frac{f : x \to x' \qquad g : y \to y'}{f \otimes g : x \otimes y \to x' \otimes y'}$$

(a) Monoidal Categories

$$\frac{A : \text{Prop} \qquad B : \text{Prop}}{A \otimes B : \text{Prop}}$$

$$\frac{\Gamma \vdash A \qquad \Delta \vdash B}{\Gamma, \Delta \vdash A \otimes B}$$

(b) Linear Logic

For the linear implication, one can use **symmetric monoidal closed categories**.

But what about ! (the bang modality)?

- Giving semantics to the bang modality is more challenging
- There are various options: Lafont categories, Seely-categories, linear categories, linear non-linear models
- This talk: **linear non-linear models**

Linear-non-linear models: Intuition

- We have a linear world where we cannot duplicate assumptions
- We have a cartesian world where we can duplicate assumptions
- The ! modality jumps from the linear world to the cartesian world and back

Linear-non-linear models: Precisely

A linear-non-linear model is a symmetric monoidal adjunction

$$\mathbb{C} \underset{\longleftarrow}{\overset{\longrightarrow}{}} \mathbb{L}$$

where $\mathbb{L}$ is a symmetric monoidal category ($\otimes$ and $\multimap$) and $\mathbb{C}$ is a cartesian category (we can copy and delete hypotheses).
We interpret ! as $\mathbb{L} \to \mathbb{C} \to \mathbb{L}$.

Example of a linear-non-linear model

Lifting of complete partial orders gives a model[2]

$$\omega\,\mathrm{cpo}_{\perp!} \underset{\mathrm{lift}}{\overset{U}{\rightleftarrows}} \omega\,\mathrm{cpo}$$

Here:

- $\omega\,\mathrm{cpo}$: objects are $\omega\,\mathrm{cpos}$, morphisms are continuous maps
- $\omega\,\mathrm{cpo}_{\perp!}$: objects are pointed $\omega\,\mathrm{cpos}$, morphisms are continuous strict maps
- lift: attaches a minimum element to a $\omega\,\mathrm{cpo}$

[2] A Mixed Linear and Non-Linear Logic, Benton

Another axample of a linear-non-linear model

A model from abelian groups[3]

$$\mathrm{Set} \underset{\mathrm{U}}{\overset{F}{\rightleftarrows}} \mathrm{Ab}$$

Here:

- $\omega\,$Ab: objects are abelian groups, morphisms are homomorphisms
- Set: objects are sets, morphisms are functions
- F: free abelian group functor

[3] A Mixed Linear and Non-Linear Logic, Benton

HOWEVER.........

- There are several methods to give models of linear logic
- Those make use complicated monoidal categories
- The relation model by Lafont uses **comonoids**
- Other models uses comonads and **Eilenberg-Moore categories**

Challenge: how do we formalize such monoidal categories in a modular way?

Our paper

- We introduce **displayed monoidal categories**
- We use them to construct complicated monoidal categories in a **modular** way
- Nice application of dependent types to category theory
- Formalized using **Coq** and the **UniMath library**

Linear Logic

Categorical Semantics of Linear Logic

Displayed Categories

Displayed Monoidal Categories

Example of Linear-non-linear models

The relation model of linear logic

$$\text{Comonoid(Rel)} \underset{C}{\overset{U}{\rightleftarrows}} \text{Rel}$$

Here:

- Rel: objects are sets, morphisms are relations
- Comonoid(Rel): objects are comonoids
- The functor C is given by **finite multisets** (i.e., free comonoid)

Complicated Monoidal Categories: Comonoids

A **comonoid** (x, ε, δ) in a monoidal category $\mathcal{C}$ consists of

- an object $x : \mathcal{C}$
- a comultiplication $\varepsilon : x \to x \otimes x$
- a counit $\delta : x \to \mathbf{1}$
- Laws: coassociativity and counitality.

Complicated Monoidal Categories: Comonoids

A **comonoid** (x, ε, δ) in a monoidal category $\mathcal{C}$ consists of

- an object $x : \mathcal{C}$
- a comultiplication $\varepsilon : x \to x \otimes x$
- a counit $\delta : x \to \mathbf{1}$
- Laws: coassociativity and counitality.

For the tensor, we need to consider comonoids as a whole
This does not allow for code reuse (i.e., complicated structures of which comonoids form substructure)

Interlude: Group Structures

We can use the following strategy to define the notion of groups.

1. Given a set X, define the type of **group structures** over X
2. A group is a set together with a group structure

This means we define the notion of groups in **2 steps**.

Interlude: Group Structures

We can use the following strategy to define the notion of groups.

1. Given a set X, define the type of **group structures** over X
2. A group is a set together with a group structure

This means we define the notion of groups in **2 steps**.

Displayed categories formalize this idea for categories

Displayed Categories

A displayed category over a category $\mathcal{C}$ consists of

- For every object $x : \mathcal{C}$, a type of structures over x
- For all morphisms $f : x \to y$ and structures S_x and S_y for x and y respectively, a type of structure-preserving maps

Displayed Categories: Example

The displayed category of groups over sets:

- For every set X, a type of group structures for X
- For all functions $f : X \to Y$ and group structures G_X and G_Y, a type expressing that f is a homomorphism

Building Complicated Structures from Simpler Ones

Displayed categories give **modularity**, because we can **untangle** and **stratify** structures.
Basically: build up complicated structures from simpler structures

Building Complicated Structures from Simpler Ones

Displayed categories give **modularity**, because we can **untangle** and **stratify** structures.
Basically: build up complicated structures from simpler structures
For example:

- Product of displayed categories (combines structures)

$$\frac{f : X \to \mathsf{Type} \qquad g : X \to \mathsf{Type}}{h(x) = f(x) \times g(x)}$$

- Adding a destructor (i.e. coalgebra structure)

$$f(x) = x \to x^n$$

We can reason about the these parts **independently**, and we can reuse the results in larger proofs.

Displayed Monoidal Categories, but what are they?

Displayed monoidal categories
=
Displayed categories + monoidal categories

Displayed Monoidal Categories, but what are they?

Displayed monoidal categories

$=$

Displayed categories $+$ monoidal categories

Note: there also needs to be a suitable interaction between the two concepts

Displayed Monoidal Categories, but *what* are they?

Let S be a displayed category over $\mathcal{C}$.

$$\frac{x : \mathcal{C} \qquad y : \mathcal{C}}{x \otimes y : \mathcal{C}}$$

(a) Monoidal Categories

$$\frac{x : \mathcal{C} \qquad \overline{x} : S_x \qquad y : \mathcal{C} \qquad \overline{y} : S_y}{\overline{x} \otimes \overline{y} : S_{x \otimes y}}$$

(b) Displayed Monoidal Categories

Comonoids using displayed monoidal categories

Main idea:

- We define a displayed monoidal category that adds a destructor $x \to F(x)$ for a lax monoidal functor F
- This way we acquire the counit ε and the comultiplication δ
- We define the full subcategory via a displayed monoidal category, and that gives us the laws

So: we build up the category of comonoids via smaller pieces and we reason about those smaller parts

Conclusion

- Main take-away: displayed monoidal categories are a technique to modularly build monoidal categories
- In the paper, we define and study displayed monoidal categories
- We apply it to a case study arising from linear logic
- They make the formalization of complicated monoidal categories more convenient and nicer
- Key examples: category of comonoids, Eilenberg-Moore category

Check our paper:
`https://dl.acm.org/doi/abs/10.1145/3636501.3636956`.