Plan

• **Lecture 1** - String diagrams and symmetric monoidal categories

• **Lecture 2** - Resource-sensitive algebraic theories

• **Lecture 3** - Interacting Hopf monoids and graphical linear algebra

• **Lecture 4** - Signal Flow Graphs and recurrence relations
Lecture 2
Resource sensitive algebraic theories
Plan

• **algebraic theories**

• symmetric monoidal theories (resource sensitive algebraic theories)

• props

• bimonoids and matrices of natural numbers

• Hopf monoids and matrices of integers
Algebraic theories

Universal Algebra

- A (presentation of) *algebraic theory* is a pair \((\Sigma, E)\) where
  - \(\Sigma\) is a set of *generators* (or *operations*), each with an *arity*, a natural number
  - \(E\) is a set of *equations* (or *relations*), between \(\Sigma\)-*terms* built up from generators and *variables*

**Example 1 - monoids**

\[\Sigma_M = \{ \cdot:2, e:0 \} \]
\[E_M = \{ \cdot(\cdot(x, y), z) = \cdot(x, \cdot(y, z)), \\
\cdot(x, e) = x, \cdot(e, x) = x \} \]

**Example 2 - abelian groups**

\[\Sigma_G = \Sigma_M \cup \{ i:1 \} \]
\[E_G = E_M \cup \{ \cdot(x, y) = \cdot(y, x), \\
\cdot(x, i(x)) = e \} \]
$\Sigma$ - terms (cartesian)

\[
\begin{array}{c}
\text{x } \in \text{Var} \\
\hline
\text{x}
\end{array}
\quad
\begin{array}{c}
t_1 \ t_2 \ \ldots \ t_m \\
\sigma \in \Sigma \\
\text{ar}(\sigma) = m
\end{array}
\quad
\begin{array}{c}
\sigma(t_1, t_2, \ldots, t_m)
\end{array}
\]

i.e. terms a *trees* with internal nodes labelled by the *generators* and the leaves labelled by *variables* and *constants* (*generators with arity 0*)
Models - classically

• To give a model of an algebraic theory $(\Sigma, E)$, choose a set $X$
  • for each operation $\sigma : k$ in $\Sigma$, choose a function $[[\sigma]] : X^k \to X$
  • now for each term $t$, given an assignment of variables $\alpha$, we can recursively compute the element of $[[t]]_{\alpha} \in X$ which is the “meaning” of $t$
  • need to ensure that for every assignment of variables $\alpha$, and every equation $t_1 = t_2$ in $E$, we have $[[t_1]]_{\alpha} = [[t_2]]_{\alpha}$ as elements of $X$

• Example 1: to give a model of the algebraic theory of monoids is to give a monoid

• Example 2: to give a model of the theory of abelian groups is to give an abelian group
Algebraic theories, categorically

• There is a nice way to think of algebraic theories categorically, due to Lawvere in the 1960s

  • get rid of “countably infinite set of variables”, “variable assignments” etc.

  • generalise - models don’t need to be sets (e.g. topological groups)

• relies on the notion of *categorical product*
Categorical product

• Suppose that \( X, Y \) are objects in a category \( \mathbf{C} \). Then \( X \) and \( Y \) have a product if \( \exists \) object \( X \times Y \) and arrows \( \pi_1 : X \times Y \to X, \pi_2 : X \times Y \to Y \) so that the following universal property holds:

\[
\begin{array}{ccc}
X & \xleftarrow{\pi_1} & X \times Y \\
& \downarrow{f} & \uparrow{h} & \downarrow{g} \\
Z & \xrightarrow{\pi_2} & Y
\end{array}
\]

for any object \( Z \) and arrows \( f : Z \to X, g : Z \to Y, \) there exists a unique \( h : Z \to X \times Y \) such that:

\[
h \circ \pi_1 = f \quad \text{and} \quad h \circ \pi_2 = g
\]

• \textit{Example}: in the category \( \mathbf{Set} \) of sets and functions, the cartesian product satisfies the universal property.

• Any category with (binary) categorical products is monoidal, with the categorical product as monoidal product.
Exercise

- If $X$ is a preorder, considered as a category, what does it mean if $X$ has (binary) categorical products?

- In $\textbf{Set}$, the categorical product is the cartesian product
  
  - What is the product in the category of categories and functors?
  
  - What is the product in the category of monoids and homomorphisms?
Lawvere categories

- Suppose that \((\Sigma, E)\) is an algebraic theory

- Define a category \(L_{(\Sigma, E)}\) with
  - **Objects**: natural numbers
  - **Arrows** from \(m\) to \(n\): \(n\) tuples of \(\Sigma\)-terms, each using possibly \(m\) variables \(x_1, x_2, \ldots, x_m\), modulo the equations of \(E\)

- Composition is *substitution*

Examples in the theory of monoids

\[
\begin{align*}
2 \xrightarrow{(x_1 \cdot x_2)} 1 & \quad 2 \xrightarrow{(x_2 \cdot x_1)} 1 \\
1 \xrightarrow{(x_1 \cdot e)} 1 & = 1 \xrightarrow{(x_1)} 1
\end{align*}
\]

It is also possible (and elegant) to view \(L_{(\Sigma, E)}\) as the free category with *products* on the data specified in \((\Sigma, E)\)
Exercise

• Lawvere categories have (binary) categorial products: \( m \times n := m + n \).

  \textbf{Q1.} What are the projections?

• In any category with binary products there is a canonical arrow \( \Delta: X \to X \times X \) called the \textit{diagonal}.

  \textbf{Q2.} How is it defined?

\textbf{Q3.} What is \( L(\emptyset, \emptyset) \)? Can you find a simple way of describing it?
Models categorically
(Functorial semantics)

• A functor $F: C \rightarrow D$ is product-preserving if

$$F(X \times Y) = F(X) \times F(Y)$$

• **Theorem.** To give a model of $(\Sigma, E)$ is to give a product-preserving functor $F: L(\Sigma, E) \rightarrow \text{Set}$

*Proof idea:* since $m = 1+1+\ldots+1$ (m times), to give a product preserving functor $F$ from $L(\Sigma, E)$ it is enough to say what $F(1)$ is.

• By changing $\text{Set}$ to other categories, we obtain a nice generalisation of classical universal algebra, with examples such as topological groups, etc.
Limitations of algebraic theories

• Copying and discarding built in

\[
\begin{align*}
2 & \xrightarrow{(x_1)} 1 \\
2 & \xrightarrow{(x_2)} 1 \\
1 & \xrightarrow{(x_1, x_1)} 2
\end{align*}
\]

• But in computer science (and elsewhere), we often need to be more careful with resources

• Consequently, there are also no bona fide operations with coarities other than one

\[
\begin{align*}
1 & \xrightarrow{c} 2 \\
1 & \xrightarrow{(c_1, c_2)} 2
\end{align*}
\]
Plan

• algebraic theories

• symmetric monoidal theories (resource sensitive algebraic theories)

• props

• bimonoids and matrices of natural numbers

• Hopf monoids and matrices of integers
Symmetric monoidal theories

- symmetric monoidal theories (SMTs) give rise to special kinds of symmetric monoidal categories called props

- Symmetric monoidal theories generalise algebraic theories, a classical concept of universal algebra, but
  - no built in copying and discarding
  - can consider operations with coarities other than 1
Symmetric monoidal theories

- A symmetric monoidal theory is a pair \((\Sigma, E)\) where
  - \(\Sigma\) is a set of generators (or operations), each with an arity, and coarity, both natural numbers
  - \(E\) is a set of equations (or relations), between compatible \(\Sigma\)-terms

- Since generators can have coarities, and since we need to be careful with resources, we can’t use the standard notion of term (tree).

- Instead, terms are arrows in a certain symmetric monoidal category, which we will construct a la magic Lego
Generators and terms

Running example: the SMT of commutative monoids

we always have the following “basic tiles” around

\[
\begin{align*}
\text{--o} & : (2, 1) & \circ \longrightarrow & : (0, 1) \\
\hline
\text{----} & : (1, 1) & \text{C} & : (2, 2)
\end{align*}
\]
Some string diagrams

String diagrams: constructions built up from the generators and basic tiles, with the two operations of magic Lego.
Recall: diagrammatic reasoning

- diagrams can slide along wires

\[
\begin{align*}
\begin{array}{c}
\begin{array}{c}
\text{k} \quad \text{A} \\
\text{m} \quad \text{k}
\end{array}
\end{array}
& \quad = \quad
\begin{array}{c}
\begin{array}{c}
\text{A} \\
\text{m}
\end{array}
\end{array}
\end{align*}
\]

functoriality

\[
\begin{align*}
\begin{array}{c}
\begin{array}{c}
\text{k} \quad \text{n} \\
\text{l} \quad \text{l}
\end{array}
\end{array}
& \quad = \quad
\begin{array}{c}
\begin{array}{c}
\text{m} \\
\text{m}
\end{array}
\end{array}
\end{align*}
\]
naturality

- wires don’t tangle, i.e.

\[
\begin{align*}
\begin{array}{c}
\begin{array}{c}
\text{k} \quad \text{m} \\
\text{m} \quad \text{m}
\end{array}
\end{array}
& \quad = \quad
\begin{array}{c}
\begin{array}{c}
\text{k} \\
\text{k}
\end{array}
\end{array}
\end{align*}
\]
i.e. pure wiring obeys the same equations as permutations

- sub-diagrams can be replaced with equal diagrams (compositionality)
$\Sigma$ - Terms (monoidal)

• Are thus the arrows of the free symmetric monoidal category $S_\Sigma$ on $\Sigma$

• Objects: natural numbers

• Arrows from $m$ to $n$: string diagrams constructed from generators, identity and twist, modulo diagrammatic reasoning

• Monoidal product, on objects: $m \oplus n := m + n$
Note that all equations are of the form $t_1 = t_2 : (m, n)$, that is, $t_1$ and $t_2$ must agree on domain and codomain.
The SMT of commutative monoids

Generators

Equations

Let’s call this SMT \( M \), for monoid
Diagrammatic reasoning example

\[ \text{Diagramatic reasoning example} \]

\[ \text{Diagramatic reasoning example} \]

\[ \text{Diagramatic reasoning example} \]

\[ \text{Diagramatic reasoning example} \]

\[ \text{Diagramatic reasoning example} \]

\[ \text{Diagramatic reasoning example} \]
Another SMT: commutative comonoids

Generators

Equations
Plan

• algebraic theories

• symmetric monoidal theories (resource sensitive algebraic theories)

• props

• bimonoids and matrices of natural numbers

• Hopf monoids and matrices of integers
From SMTs to symmetric monoidal categories

- Every symmetric monoidal theory \((\Sigma, E)\) yields a free strict symmetric monoidal category \(S_{(\Sigma, E)}\)
  - Object: natural numbers
  - Arrows: monoidal \(\Sigma\)-terms, taken modulo equations in \(E\)
- Such categories are an instance of \textit{props} (product and permutation categories)
props

• A prop (product and permutation category) is
  • strict symmetric monoidal
  • objects = natural numbers
  • monoidal product on objects = addition
    • i.e. $m \oplus n = m + n$
Examples

1. Any symmetric monoidal theory gives us a prop

2. The strict symmetric monoidal category $F$

   - arrows from $m$ to $n$ are all functions from the $m$ element set $\{0, \ldots, m-1\}$ to the $n$ element set $\{0, \ldots, n-1\}$

3. The free strict symmetric monoidal category on one object, the category $P$ of permutations

4. The category $I$ with precisely one arrow from any $m$ to $n$ is a prop
Morphisms of props

• A morphism of props $F: \mathbf{X} \rightarrow \mathbf{Y}$ is an identity on objects symmetric monoidal functor

  • identity-on-objects: $F(m) = m$

  • strict: $F(C \oplus D) = F(C) \oplus F(D)$

  • symmetric monoidal: $F(\text{tw}_{m,n}) = \text{tw}_{m,n}$

  • functor $F(l_m) = l_m$, $F(C ; D) = F(C) ; F(D)$

• In other words, all the structure is simply preserved on the nose — easy peasy
Models

• Recall: models of algebraic theories are finite product preserving functors, often to Set

• We can define models of an SMT to be symmetric monoidal functors, a generalisation of the notion of finite product preserving

• Some computer science intuitions:
  • SMTs, like $M$, are a syntax
  • props like $F$ are a semantics
  • homomorphisms map syntax to semantics
  • when the map is an isomorphisms, we have an equational characterisation, and a sound and fully complete proof system to reason about things in $F$
Example

As props, $\mathbf{M}$ is *isomorphic* to $\mathbf{F}$

- So $\mathbf{M}$ is an equational characterisation of $\mathbf{F}$
- or the “commutative monoids is the theory of functions”
Morphisms from (props obtained from) SMTs

• Let us define a morphism \([-\] : \(M \rightarrow F\)

  • \(M\) is obtained from a symmetric monoidal theory \((\Sigma, E)\), thus its arrows are constructed inductively

• To define \([-\] it thus suffices to

  • say where the generators in \(\Sigma\) are mapped

  • check that the equations in hold in \(F\)

• This is a general pattern when defining morphisms from a prop obtained from an SMT
Simple exercise: check the following hold in $F$

$$[[ - ]] : M \rightarrow F$$

\[
\begin{align*}
\{1,2\} &\rightarrow \{1\} \\
\{\} &\rightarrow \{1\}
\end{align*}
\]

$$= (\text{Assoc})$$

$$= (\text{Unit})$$

$$= (\text{Comm})$$
Soundness

- Simple observation: the fact that we have a homomorphism $[\cdot] : M \to F$ means that diagrammatic reasoning in $M$ is sound for $F$

**Q1.** What property of $[\cdot]$ do we need to ensure completeness?

**Q2.** If we have soundness and completeness, is this enough for $[\cdot]$ to be an *isomorphism*? (i.e. invertible)
Full and faithful

- To show that a morphism of props $F: X \to Y$ is an isomorphism it suffices to show that it is full and faithful

  - **full**: for every arrow $g$ of $Y$ there exists an arrow $f$ of $X$ such that $F(f) = g$

  - **faithful**: given arrows $f, f'$ in $X$, if $F(f) = F(f')$ then $f = f'$

So full and faithful functor from a (free PROP on an) SMT = sound and fully complete equational characterisation
\[
[[\cdot]] : M \rightarrow F
\]

- **full**: every function between finite sets can be constructed from the two basic building blocks together with permutations

- **faithful**: every diagram in M can be written as multiplications followed by units, which corresponds to a factorisation of a function as an surjection followed by an injection. This factorisation is unique “up-to-permutation”.
Free things

- A free “something on $X$” is one that satisfies a universal property — it’s the “smallest” thing that contains $X$ which satisfies the properties of “something”

- e.g. free “monoid on a set $\Sigma$” is the set of finite words $\Sigma^*$
Free strict symmetric monoidal category on one object

• Any ideas?
  
  • Recall: there is a category $\mathbf{1}$ with one object and one arrow
  
  • Let $\mathbf{X}$ be the free symmetric monoidal category on $\mathbf{1}$
  
  • There should be a functor from $\mathbf{1}$ to $\mathbf{X}$
  
  • For any functor to a strict symmetric monoidal category $\mathbf{Y}$, there should be a strict symmetric monoidal functor $\mathbf{X}$ to $\mathbf{Y}$ such that the diagram below commutes
Plan

• algebraic theories

• symmetric monoidal theories (resource sensitive algebraic theories)

• props

• bimonoids and matrices of natural numbers

• Hopf monoids and matrices of integers
The SMT of bimonoids

- Combines generators and equations of the SMTs of monoids and comonoids

- *Intuition*: “numbers” travel on wires from left to right

The monoid structure acts as addition/zero

The comonoid structure acts as copying/discardng
The SMT of bimonoids

- all the generators we have seen so far
- monoid and comonoid equations

```
\begin{align*}
\text{Bimonoid equations:} \\
\text{monoid equations:} \\
\text{comonoid equations:}
\end{align*}
```

- “adding meets copying” - equations compatible with intuition
Mat

- A PROP where arrows m to n are n×m matrices of natural numbers
  - e.g. \( \begin{pmatrix} 0 & 5 \end{pmatrix} : 2 \rightarrow 1 \quad \begin{pmatrix} 3 \\ 15 \end{pmatrix} : 1 \rightarrow 2 \quad \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} : 2 \rightarrow 2 \)
- Composition is matrix multiplication
- Monoidal product is direct sum

\[
A_1 \oplus A_2 = \begin{pmatrix} A_1 & 0 \\ 0 & A_2 \end{pmatrix}
\]
- Symmetries are permutation matrices
Theorem. $B$ is isomorphic to the Mat

- ie. bimonoids is the theory of natural number matrices

- natural numbers themselves can be seen as certain $(1,1)$ diagrams, with the recursive definition below

- as we will see, the algebra (rig) of natural numbers follows

\[
\begin{align*}
0 &:= \quad \bullet \quad \circ \\
\text{k+1} &:= \quad \bullet \quad \circ
\end{align*}
\]

+1 is “add one path”
Exercise

Given

\[0 := \bullet - \circ\]

\[k+1 := \bullet - k - \circ\]

1. \[\quad m \quad = \quad m \quad m\]
2. \[\quad m \quad m \quad = \quad m\]
3. \[\quad m \quad n \quad = \quad m + n\]
4. \[\quad m \quad n \quad = \quad nm\]

Given, prove
**Proof** $B \cong \text{Mat}$

*Recall:* Since $B$ is an SMT, suffices to say where generators go (and check that equations hold in the codomain)

\[
\begin{align*}
\begin{array}{c}
\text{Diagram} \\
\text{Symbol} \\
\text{Expression} \\
\text{Function} \\
\end{array}
\end{align*}
\]

\begin{align*}
\begin{array}{c}
\begin{array}{c}
\text{Diagram} \\
\text{Symbol} \\
\text{Expression} \\
\text{Function} \\
\end{array}
\end{align*}
\]

**Full** - easy!
Recursively define a syntactic sugar for matrices

**Faithful** - harder
Use the fact that equations are a presentation of a *distributive law*, obtain factorisation of diagrams as comonoid structure followed by monoid structure - **normal form**
Normal form for B

• Every diagram can be put in the form
  • comonoid ; monoid
• Centipedes
Matrices

• To get the ijth entry in the matrix, count the paths from the jth port on the left to the ith port on the right

• Example:
Q1. Show that the monoidal product in \( B=\text{Mat} \) is the categorical product.

Q2. The categorical coproduct of \( X, Y \), if it exists satisfies the following universal property:

For any object \( Z \) and arrows \( f: X \to Z, \ g: Y \to Z \),

\[
\exists \text{ unique } h: X+Y \to Z \text{ s.t. } i_1 \circ h = f \text{ and } i_2 \circ h = g
\]

show that the monoidal product in \( B=\text{Mat} \) is the categorical coproduct.

When a monoidal product satisfies both the universal properties of products and coproducts, we say that it is a \textit{biproduct}.

In fact \( B=\text{Mat} \) is the free category with biproducts on one object.

Q3 (challenging). Given a category \( \mathbf{C} \), describe the free category with biproducts on \( \mathbf{C} \).
Lawvere categories with string diagrams
(i.e. how ordinary syntax looks, with string diagrams)

\[ \sigma \in \Sigma \]

\[ \begin{array}{c}
\bullet \\
\hspace{1cm} \sigma \\
\hspace{1cm} \sigma \\
\end{array} \quad \begin{array}{c}
\bullet \\
\hspace{1cm} \bullet \\
\hspace{1cm} \bullet \\
\end{array} \quad \begin{array}{c}
\bullet \\
\bullet \\
\end{array} \]

and what else?
In particular, notice that $B$ is isomorphic (as a symmetric monoidal category) to the Lawvere category of commutative monoids!

**Exercise**: show that the monoidal product now becomes a *categorical* product

In particular, notice that $B$ is isomorphic (as a symmetric monoidal category) to the Lawvere category of commutative monoids!
Plan

• algebraic theories

• symmetric monoidal theories (resource sensitive algebraic theories)

• props

• bimonoids and matrices of natural numbers

• **Hopf monoids and matrices of integers**
Putting the n in ring: Hopf monoids

• generators of bimonoids + **antipode**

• think of this as acting as -1

• equations of bimonoids and the following

\[ \begin{align*}
\includegraphics{eq1} &= \includegraphics{eq2} \\
\includegraphics{eq3} &= \includegraphics{eq4} \\
\includegraphics{eq5} &= \includegraphics{eq6} \\
\includegraphics{eq7} &= \includegraphics{eq8}
\end{align*} \]
-1 \cdot -1 = 1
The ring of integers

- Simple induction:  \[ \boxed{n} = \boxed{n} \]

- Recall: in \( B \), the arrows \( 1 \rightarrow 1 \) were in one-to-one correspondence with natural numbers

- In \( H \), the arrows \( 1 \rightarrow 1 \) are in one-to-one correspondence with the integers

\[
\begin{align*}
0 & := -
\end{align*}
\]

\[
\begin{align*}
k+1 & := \boxed{k}
\end{align*}
\]

\[
\begin{align*}
-n & := \boxed{n}
\end{align*}
\]
Exercise

- Verify that, in $\mathbb{H}$, for all integers $m, n$ we have

\[ m \odot n = m + n \]

\[ m \odot n = mn \]
Mat\(_z\)

- Arrows m to n are n×m matrices of integers
  - composition is matrix multiplication
  - monoidal product is direct sum
- \(\text{Mat}_z\) is equivalent to the category of finite dimensional free \(\mathbb{Z}\)-modules

- SMT \(H\) is isomorphic to the PROP \(\text{Mat}_z\)
Path counting in MatZ

- To get the $ij$th entry in the matrix, count the
  - positive paths from the $j$th port on the left to the $i$th port on the right (where antipode appears an even number of times)
  - negative paths between these two ports (where antipode appears an odd number of times)
  - subtract the negative paths from the positive paths

- Example:

\[
\begin{pmatrix}
0 & -1 \\
1 & 0
\end{pmatrix}
\]
Proof $H \cong \text{Mat}_\mathbb{Z}$

- Fullness easy

- Faithfulness more challenging: put diagrams in the form

  - copying; antipode; adding
We saw that $\mathbf{B}$ is the isomorphic, as a symmetric monoidal category, to the Lawvere category of commutative monoids.

Which Lawvere category is $\mathbf{H}$ isomorphic to?