Graphical Linear Algebra
a specification language for linear algebra
Pawel Sobocinski

based on joint work with Filippo Bonchi and Fabio Zanasi
CONCUR

• My first CS conference: Concur 2001, Aalborg

Bigraphical Reactive Systems

Robin Milner

University of Cambridge Computer Laboratory
New Museums Site, Pembroke Street, Cambridge CB2 3QG, UK

Abstract. A notion of biograph is introduced as a model of mobile interaction. A bigraph consists of two independent structures: a topograph rep-

• My first talk: EXPRESS 2002, Brno (a Concur 2002, satellite workshop

• My first paper at Concur 2003, Marseille (with Bartek Klin)
Concur
(process algebra)

(string diagrams)
(linear algebra)

(Jean-Raymond Abrial: how theorem provers make you treat mathematics as a branch of software engineering)

This talk: how to see linear algebra as a branch of process algebra
implementations $\subseteq$ specifications

- runnable, completely specified
- non-deterministic, partially specified behaviour

functions $\subseteq$ relations

- single-valued, total
- non single-valued, non total
string diagrams

- showing up in a growing number of recent CS papers
  - Abramsky, Duncan, Coecke, … - Categorical Quantum Foundations and Quantum Computation
  - Mellies, … - Logic, Game Semantics
  - Ghica, Jung, … - Digital Circuits
  - Baez, Bonchi, Erbele, Fong, S., Zanasi, … - Signal Flow Graphs, Control and Systems Theory
  - Coecke, Sadrzadeh, … - Computational Linguistics
  - …

1st Workshop on String Diagrams in Computation Logic and Physics
Jericho Tavern, Oxford
8-9 September, 2017
(satellite of FSCD, next year satellite of CSL)
linear algebra

- the most practical mathematical theory?

- the engine room of systems and control theory, quantum computing, network theory, …

- mathematical physics and engineering relies on it: systems of nonlinear differential equations are solved with linear approximations

- shows up in surprising places (Petri net invariants, PageRank is an eigenvector, SVD in data science and learning, …)

- Graphical Linear Algebra - linear algebra with string diagrams

  - focus on linear relations rather than on linear maps

  - GraphicalLinearAlgebra.net
Plan

• String diagrams & diagrammatic reasoning
  • what is it?
  • why is it relevant for cs?
• Graphical Linear Algebra
• Fun stuff
A prop is a strict symmetric monoidal category with

- strict means: \( \otimes \) is associative on the nose
- objects = natural numbers
- \( m \otimes n := m + n \) (I will usually write \( m \oplus n \))

Simple examples:

- permutations of finite sets
- functions between finite sets

prop homomorphism = identity on objects symmetric monoidal functor
A string diagram
Synchronising Composition

\[ C : k \rightarrow l \quad D : l \rightarrow m \]

\[ C ; D : k \rightarrow m \]

“C and D synchronise on l”
Parallel composition

\[
\begin{align*}
C: k \rightarrow l & \quad D: m \rightarrow n \\
\therefore C \oplus D: k \oplus m \rightarrow l \oplus n
\end{align*}
\]

“C and D in parallel”
Perks of the notation

\[
C : k \to l \quad D : l \to m \quad E : m \to n \\
(C ; D) ; E = C ; (D ; E) : k \to n
\]

\[
C : m \to n \quad D : m' \to n' \quad E : m'' \to n'' \\
(C \oplus D) \oplus E = C \oplus (D \oplus E) : m \oplus m' \oplus m'' \to n \oplus n' \oplus n''
\]
More perks

\[(A \ ; \ B) \oplus (C \ ; \ D) = (A \oplus C) \ ; (B \oplus D)\]
Diagrammatic reasoning

\[ C : m \rightarrow n \]

\[ I_m ; C = C = C ; I_n \]

\[ (A \oplus I_r) ; (l_q \oplus B) = A \oplus B = (l_p \oplus B) ; (A \oplus I_s) \]
Symmetries

$\sigma_{m,n}: m \oplus n \rightarrow n \oplus m$

\[
\begin{align*}
C_n & C_p q \\
C_p q & C_m
\end{align*}
\]
Plan

• String diagrams & diagrammatic reasoning
  • what is it?
  • why is it relevant for cs?
• Graphical Linear Algebra
• Fun stuff
(commutative) monoids and groups
a la 1930s universal algebra - syntax

- (presentation of) algebraic theory
  - pair $T = (\Sigma, \mathcal{E})$ of finite sets

- for commutative monoids:
  - signature $\Sigma$, arity: $\Sigma \rightarrow \mathbb{N}$
    - $\cdot : 2$
    - $e : 0$
  - equations $\mathcal{E}$ (pairs of typed terms)
    - $x \cdot (y \cdot z) = (x \cdot y) \cdot z$
    - $x \cdot y = y \cdot x$
    - $x \cdot e = x$

- For abelian groups, additionally
  - signature: $(-)^{-1} : 1$
  - equations: $x \cdot x^{-1} = e$
(commutative) monoids and groups
a la universal algebra - semantics

- To give a model
  - Pick carrier set $X$
    - $\cdot : 2 \rightarrow \cdot : X^2 \rightarrow X$
    - $e : 0 \rightarrow e : X^0 \rightarrow X$
    - $(\cdot)^{-1} : 1 \rightarrow (\cdot)^{-1} : X^1 \rightarrow X$
  - For every evaluation of variables $\sigma : \text{Var} \rightarrow X$, each equation must hold

- So, e.g. a **model of the algebraic theory of monoids** is the same thing as a **monoid**, in the classical sense
functorial semantics, 1960s

- Lawvere was not happy with universal algebra
  - too set theory specific
    - (e.g. topological groups morally should be a model)
  - too much ad hoc extraneous machinery
    - (e.g. countable set of variables, variable evaluation, etc.)
- Lawvere’s 1963 doctoral thesis “Functorial semantics of algebraic theories” - universal algebra categorically
Lawvere theories

- Given algebraic theory \((\Sigma, E)\), a category \(L(\Sigma, E)\) with
  - objects: the natural numbers
  - arrows from \(m\) to \(n\):
    - \(n\)-tuples of terms that (possibly) use variables \(x_1, x_2, \ldots x_m\) modulo equations \(E\)
    - composition is substitution
  - e.g.
    
    \[
    \begin{align*}
    2 & \xrightarrow{(x_1 \cdot x_2)} 1 \\
    2 & \xrightarrow{(x_2 \cdot x_1)} 1 \\
    1 & \xrightarrow{(e, x_1)} 2 \\
    1 & \xrightarrow{(x_2 \cdot x_1)} 1 = 1 \\
    1 & \xrightarrow{(x_1 \cdot e)} 1
    \end{align*}
    \]
  - More concisely - “free category with products on the data of an algebraic theory”
  - any \(L(\Sigma, E)\) is a prop!

**classical model** \(\Rightarrow\) **cartesian functor** \(L \rightarrow \textbf{Set}\)
products in a Lawvere theory

\[
\begin{aligned}
&m &\xrightarrow{(x_1,x_2, \ldots, x_m)} &m+n \\
&k &\xleftarrow{(f_1,f_2,\ldots,f_m)} &m \\
&m+n &\xrightarrow{(x_{m+1},x_{m+2}, \ldots, x_{m+n})} &n \\
& &\xleftarrow{(g_1,g_2,\ldots,g_n)} &k \\
\end{aligned}
\]
limitations of algebraic theories

- Copying and discarding **built in**

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

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

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

- But in quantum mechanics, computer science, and elsewhere we often need to be more careful with resources
symmetric monoidal theories

- algebraic theory in the symmetric monoidal settings

- a symmetric monoidal theory is a pair of finite sets $(\Sigma, E)$
  - $\Sigma$ signature, arity : $\Sigma \rightarrow \mathbb{N}$, coarity : $\Sigma \rightarrow \mathbb{N}$
  - $E$ equations, pairs of string diagrams constructed from $\Sigma$, identity and symmetries
symmetric monoidal theory of commutative monoids

\[
\begin{align*}
\begin{tikzpicture}
  \node (a) at (0,0) [circle, draw] {};
  \node (b) at (1,0) [circle, draw] {};
  \draw (a) to (b);
\end{tikzpicture} & : (2,1) & \begin{tikzpicture}
  \node (a) at (0,0) [circle, draw] {};
  \end{tikzpicture} & : (0,1)
\end{align*}
\]

\[
\begin{align*}
\begin{tikzpicture}
  \node (a) at (0,0) [circle, draw] {};
  \node (b) at (1,0) [circle, draw] {};
  \node (c) at (1,1) [circle, draw] {};
  \draw (a) to (b);
  \draw (c) to (b);
\end{tikzpicture} & = & \begin{tikzpicture}
  \node (a) at (0,0) [circle, draw] {};
  \node (b) at (1,0) [circle, draw] {};
  \node (c) at (1,1) [circle, draw] {};
  \draw (a) to (b);
  \draw (c) to (b);
\end{tikzpicture}
\end{align*}
\]

\[
\begin{align*}
\begin{tikzpicture}
  \node (a) at (0,0) [circle, draw] {};
  \node (b) at (0.5,0.5) [circle, draw] {};
  \node (c) at (1,0) [circle, draw] {};
  \draw (a) to (b);
  \draw (b) to (c);
\end{tikzpicture} & = & \begin{tikzpicture}
  \node (a) at (0,0) [circle, draw] {};
  \node (b) at (0.5,0.5) [circle, draw] {};
  \node (c) at (1,0) [circle, draw] {};
  \draw (a) to (b);
  \draw (b) to (c);
\end{tikzpicture}
\end{align*}
\]

\[
\begin{align*}
\begin{tikzpicture}
  \node (a) at (0,0) [circle, draw] {};
  \node (b) at (1,0) [circle, draw] {};
  \end{tikzpicture} & = & \begin{tikzpicture}
  \node (a) at (0,0) [circle, draw] {};
  \node (b) at (1,0) [circle, draw] {};
\end{tikzpicture}
\end{align*}
\]

\[
\begin{align*}
\begin{tikzpicture}
  \node (a) at (0,0) [circle, draw] {};
  \node (b) at (1,0) [circle, draw] {};
  \end{tikzpicture} & = & \begin{tikzpicture}
  \node (a) at (0,0) [circle, draw] {};
  \node (b) at (1,0) [circle, draw] {};
  \node (c) at (1,1) [circle, draw] {};
  \draw (a) to (b);
  \end{tikzpicture}
\end{align*}
\]

\[
\begin{align*}
\begin{tikzpicture}
  \node (a) at (0,0) [circle, draw] {};
  \node (b) at (1,0) [circle, draw] {};
  \node (c) at (1,1) [circle, draw] {};
  \draw (a) to (b);
\end{tikzpicture} & = & \begin{tikzpicture}
  \node (a) at (0,0) [circle, draw] {};
  \node (b) at (1,0) [circle, draw] {};
  \node (c) at (1,1) [circle, draw] {};
  \draw (a) to (b);
\end{tikzpicture}
\end{align*}
\]
commutative monoid facts

• the following are isomorphic as props
  • prop of commutative monoids
  • prop of functions between finite sets

• not isomorphic to the Lawvere theory of commutative monoids
folk theorem

- A symmetric monoidal category $\mathcal{C}$ is cartesian iff
  - every object $C \in \mathcal{C}$ has a commutative comonoid $\Delta: C \to C \otimes C$, $c: C \to I$

```
\xymatrix{  & C \ar[dl]_{c} \ar[dr]^{c} & \ar[d] C \\
C \otimes C & & C \otimes C }
```

- compatible with $\otimes$ in the obvious way

- and every arrow $f : m \to n$ of $\mathcal{C}$ is a comonoid homomorphism, i.e.

```
\xymatrix{  & m \ar[r]^{f} & n \\
m \ar[ur]_{f} & & n \ar[ul]^{f} }
```
Lawvere theories as SMTs

\( (\sigma \in \Sigma) \)

\[ E^+ \]
Lawvere theory of commutative monoids as SMT
Lawvere theory of abelian groups as an SMT
• e.g. the Hopf equation

\[ \square \quad = \quad - \quad - \]

is simply the SMT version of \( x \cdot x^{-1} = e \)

• Lawvere theory of commutative monoids = Symmetric monoidal theory of (co)commutative bialgebra

• Lawvere theory of abelian groups = Symmetric monoidal theory of (co)commutative Hopf algebras

So bialgebras and Hopf algebras are, respectively, monoids and groups in a resource sensitive universe.
Plan

• String diagrams & diagrammatic reasoning
  • what is it?
  • why is it relevant for cs?

• Graphical Linear Algebra

• Fun stuff
Linear relation

- Definition. Suppose $V, W$ are $k$-vector spaces. A linear relation $R$ from $V$ to $W$ is a linear subspace of $V \times W$

- i.e.

- $(0_V, 0_W) \in R$

- if $(v, w), (v', w') \in R$ then $(v + v', w + w') \in R$

- if $(v, w) \in R$ and $\lambda \in k$ then $(\lambda v, \lambda w) \in R$
Why linear relations?

- any $m \times n$ matrix $A$ gives linear relation $\{ (x, Ax) \mid x \in k^n \} \subseteq k^n \times k^m$

- the singleton $(0, *)$ is a linear relation $\subseteq k^m \times k^0$

- composing gives the kernel of $A$

- the set $\{ (*, x) \mid x \in k^n \}$ is a linear relation $\subseteq k^0 \times k^n$

- composing gives the image of $A$
Graphical linear algebra

String diagrammatic syntax for linear relations with a sound and fully complete axiomatisation called Interacting Hopf Algebra
The signature, pt 1

\[
\begin{align*}
\left\{ \begin{pmatrix} x \\ y \end{pmatrix}, x + y \right\} & \subseteq k^2 \times k \\
\{*, 0\} & \subseteq k^0 \times k^1
\end{align*}
\]

+ mirror images
The signature, pt 2

\[
\left\{ x, \begin{pmatrix} x \\ x \end{pmatrix} \right\} \subseteq k \times k^2
\]

\[
\left\{ x, * \right\} \subseteq k \times k^0
\]

+ mirror images
interacting Hopf algebras
Bonchi, S., Zanasi, JPAA 2017

special Frobenius
Hopf
special Frobenius

$p = (p \neq 0)$

cf. Coecke, Duncan. Interacting quantum observables, NJP 2011
(special) Frobenius monoids
Bonchi, Holland, Pavlovic, S. Refinement for signal flow graphs, CONCUR’17
Plan

- String diagrams & diagrammatic reasoning
  - what is it?
  - why is it relevant for cs?
- Graphical Linear Algebra
- Fun stuff
naturals as string diagrams

• naturals as syntactic sugar

```
0 := \bullet - \circ
```

```
k+1 := \bullet - \circ
```

• some easy lemmas

```
\begin{align*}
 m + m &= m + m \\
 m \cdot n &= m \cdot n
\end{align*}
```

```
 m + n &= m + n \\
 mn &= mn
```

correspondence with matrices

- in general, the $ij$th entry is the number of paths from the $j$th port on the left to the $i$th port on the right.
rational numbers

\[ p/q := \begin{array}{c}
\begin{array}{c}
p \\
q
\end{array}
\end{array} \quad \text{e.g. 2/3 is} \begin{array}{c}
\begin{array}{c}
\text{Diagram}
\end{array}
\end{array} \]

\[
\begin{array}{c}
p \quad q \quad r \quad s = \begin{array}{c}
p \\
r \\
q \\
s
\end{array}
\end{array} = \begin{array}{c}
p \\
r \\
q \\
s
\end{array} = \begin{array}{c}
rp \\
sq
\end{array}
\]

\[
\begin{array}{c}
p \quad q \\
r \\
s
\end{array} = \begin{array}{c}
p \\
q \\
q \\
q
\end{array} = \begin{array}{c}
sp \\
qs
\end{array} = \begin{array}{c}
sp \\
qr
\end{array} = \begin{array}{c}
sp+qr
\end{array}
\]
division by 0

(fixing the deficiencies of the usual syntax)

Two ways of interpreting “0/0”
projective arithmetic ++

- projective arithmetic identifies rationals with 1-dim spaces (lines) of $\mathbb{Q}^2$
  - $p \rightarrow \{ (x, px) \mid x \in \mathbb{Q} \}$
  - $\infty : \{ (0, x) \mid x \in \mathbb{Q} \}$
- The extended system includes all the subspaces of $\mathbb{Q}^2$, in particular:
  - the unique zero dimensional space $\{ (0, 0) \}$
  - the unique two dimensional space $\{ (x, y) \mid x, y \in \mathbb{Q} \}$
Linear subspaces

• **Observation.** Linear subspaces of $k^n$ are in 1-1 correspondence with string diagrams.
Some examples

\[ \left\{ \left( \left( \begin{array}{c} x \\ y \end{array} \right) , \ast \right) \mid x + 2y = 0 \right\} \subseteq k^2 \times k^0 \]

\[ \left\{ \left( a \left( \begin{array}{c} 1 \\ 2 \end{array} \right) , \ast \right) \mid a \in k \right\} \subseteq k^2 \times k^0 \]
Intersection and sum of spaces

intersection

sum
linear independence

decomposition into linearly independent subspaces $R$ and $S$
Eigenvalues & eigenspaces

• $V$ is an eigenspace of $A$: $k^n \rightarrow k^n$ with eigenvalue $a \in k$ when:

$A^n V = \alpha^n V$
A has a spectral decomposition when we can find a decomposition of $k^n$ into eigenspaces $V_1, V_2, \ldots, V_m$ and eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_m$. 

$$A = \lambda_1 V_1 = \lambda_2 V_2 = \cdots = \lambda_m V_m$$
\[
\begin{pmatrix}
5 & -1 \\
-\frac{5}{2} & -2 \\
\end{pmatrix}
\begin{pmatrix}
1 & 1 \\
-\frac{1}{2} & 2 \\
\end{pmatrix}^{-1}
= \frac{1}{5}
\begin{pmatrix}
19 & -12 \\
-12 & 1 \\
\end{pmatrix}
\]
Bibliography

• Bonchi, S., Zanasi - Interacting bialgebras are Frobenius, FoSSaCS '14
• Bonchi, S., Zanasi - Interacting Hopf algebras, J Pure Applied Algebra 221:144–184, 2017
• Bonchi, S., Zanasi - The calculus of signal flow diagrams I: Linear Relations on Streams, Inf Comput 252:2–29, 2017
• Bonchi, S., Zanasi - A categorical semantics of signal flow graphs, CONCUR ’14
• Bonchi, S., Zanasi - Full abstraction for signal flow graphs, PoPL ’16
• Bonchi, S., Zanasi - Lawvere Theories as composed PROPs, CMCS ’16
• Fong, Rapisarda, S. - A categorical approach to open and interconnected dynamical systems, LiCS ’16
• Bonchi, Gadducci, Kissinger, S. - Rewriting modulo symmetric monoidal structure, LiCS ’16
• Bonchi, Gadducci, Kissinger, S. - Confluence of graph rewriting with interfaces, ESOP ’17
• Bonchi, Holland, Pavlovic, S - Refinement for signal flow graphs, CONCUR ’17
• Bonchi, Pavlovic, S. - Functorial semantics of Frobenius theories (in preparation)

GraphicalLinearAlgebra.net