From monoidal automata to string diagrams for effects

Matt Earnshaw

including work in progress
with Chad Nester and Mario Román

TSEM Seminar
21st March 2024

Trace monoids

Words record the behaviour of sequential machines with atomic actions.

Mazurkiewicz traces are words in which specified pairs of actions can commute [DR95].

Given,

- Σ, a finite set of atomic actions
- I, a symmetric relation on Σ specifying independent pairs of actions,

the trace monoid $T(\Sigma, I) := \langle \Sigma \mid ab = ba \rangle$, for all independent pairs (a, b).

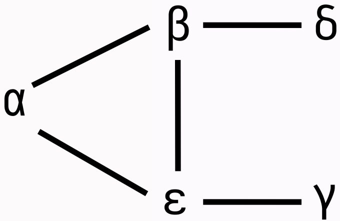
Traces are then just sequences of actions up to commutation of independent actions.

$$[wabv] = [wbav]$$

Mazurkiewicz traces are a non-interleaving semantics for the behaviour of systems with concurrent atomic actions.

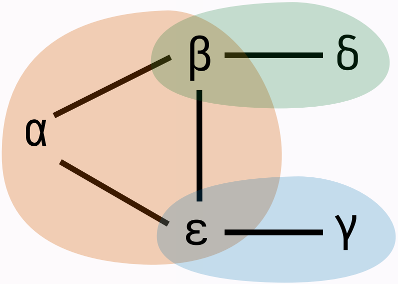
Dependence and locations

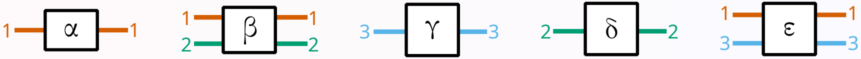
Trace monoids can also be presented graphically, following the idea of *dependence graphs* [DR95].



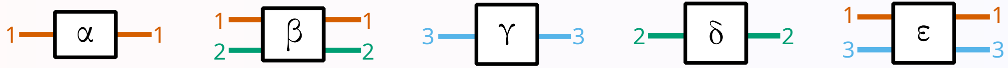
Dependence and locations

Trace monoids can also be presented graphically, following the idea of *dependence graphs* [DR95].

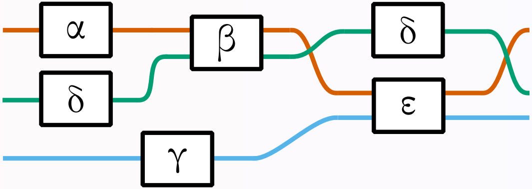




Traces graphically



Traces are string diagrams over this polygraph, from the set of locations to itself [ES23]:



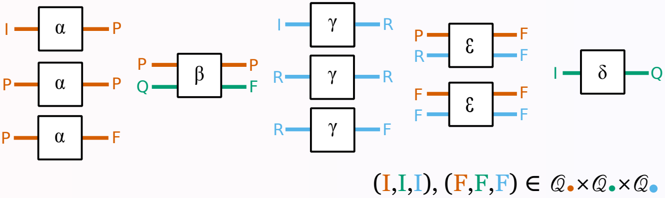
$$[\alpha\delta\gamma\beta\delta\epsilon] = [\delta\alpha\beta\gamma\epsilon\delta] = \ldots$$

Monoidal automata

A *non-deterministic monoidal automaton* over a finite polygraph Γ has dynamics:

- an family of finite state sets $\{Q_c\}_{c \in S_\Gamma}$ indexed by the sorts of Γ,
- for each $\gamma : c_1 \ldots c_n \to c'_1 \ldots c'_m$ in Γ, transitions $\delta_\gamma : \prod_{i=0}^{n} Q_{c_i} \to \mathscr{P}(\prod_{j=0}^{m} Q_{c'_j})$.
- initial and final finite languages over $\prod Q_c$.

For single-sorted polygraphs, transitions are of the form $\delta_\gamma : Q^n \to \mathscr{P}(Q^m)$.

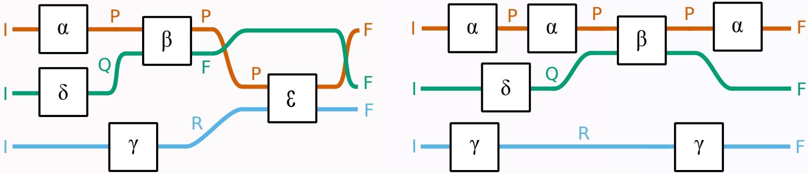


Examples: classical non-deterministic finite automata, tree automata, branching automata, asynchronous automata.

Monoidal automata

Inductively define transitions over string diagrams in the free prop over Γ,

- For a generator $\gamma \in \Gamma$, $\hat{\delta}(\vec{q}, \gamma) := \delta_\gamma(\vec{q})$,
- for identities, $\hat{\delta}(\vec{q}, \mathrm{id}) := \vec{q}$,
- for symmetries, $\hat{\delta}(ab, \sigma) := ba$,
- for $s_1 \otimes s_2$, $\hat{\delta}(\vec{q}, s_1 \otimes s_2) := \hat{\delta}(\vec{q_1}, s_1) + \hat{\delta}(\vec{q_2}, s_2)$, where $\vec{q} = \vec{q_1} + \vec{q_2}$,
- for $s \mathbin{;} s'$, $\hat{\delta}(\vec{q}, s \mathbin{;} s') := \hat{\delta}(\hat{\delta}(\vec{q}, s), s')$.



Regular monoidal languages over Γ are the sets of string diagrams in $\mathcal{F}_\otimes\Gamma$ accepted by a monoidal automaton.

Asynchronous automata are monoidal automata

Regular trace languages are defined in the literature by an algebraic criterion.

(Zielonka 1987[1]) introduced *asynchronous automata* and proved that these accept exactly the regular trace languages.

Non-deterministic monoidal automata over distributed alphabets are precisely Zielonka's asynchronous automata.

Consequently, regular trace languages are exactly regular (symmetric) monoidal languages over distributed alphabets.

[1] *Notes on Finite Asynchronous Automata*, Informatique théorique et applications

From trace monoids to effectful categories

There are some unsatisfactory aspects of the foregoing, but also some intriguing connections.

- We generated a monoidal category but we only cared about one hom set.
- We gave an arbitrary order to the set of locations.

The idea of using strings to prevent global interchange reminds us of string diagrams for premonoidal categories.

Let's pursue this idea.

Effectful categories for effectful processes

As we have seen, processes in computer science are not always independent:

Premonoidal categories[2] or more generally *effectful categories*[3] refine monoidal categories: interchange does not hold globally.

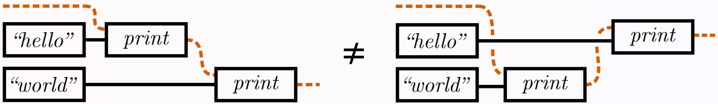
Key example: Kleisli categories of strong monads, or more generally strong promonads.

Interchange holds just when the monad is *commutative*.

[2] Power and Robinson [PR97]
[3] Levy, Power, and Thielecke [LPT03], Garner and López Franco [GF16], Román [Rom23]

String diagrams for premonoidal categories

String diagrams with a distinguished wire present free effectful categories.[4]



In practice, this global effect limits topological reasoning:

[4] Jeffrey, Román [Jef97, Rom23].

String diagrams with devices

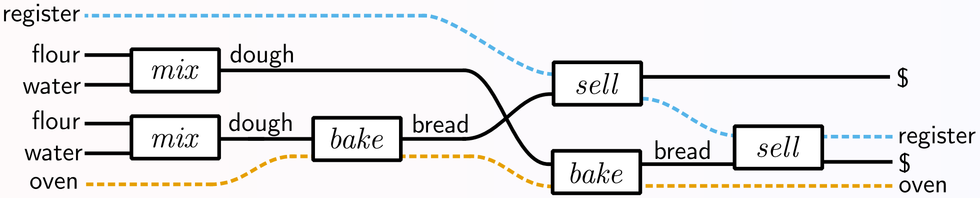
In practice, this global effect limits topological reasoning:

By introducing multiple *device* wires,

many natural equations are now topological again.

String diagrams with devices

Devices are definite noun phrases: if we only have one oven, we cannot *bake* in parallel.



We introduce a convenient presentation for premonoidal categories based on this idea.

Device signature

A *device signature* $\mathcal{S}$ is given by:

- sets R, P, D of *resources, processes* and *devices*,
- functions $s, t : P \to R^*$ assigning source and target *words* of resources,
- a function $d : P \to \mathscr{P}(D)$ specifying a set of devices used by each process.

Device presentations

A device presentation further specifies some equations $\mathcal{E}$ between string diagrams over the signature:

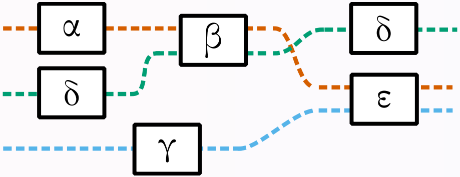
Given a device presentation $(\mathcal{S}, \mathcal{E})$, we construct a category $\mathscr{F}(\mathcal{S}, \mathcal{E})$ whose morphisms $X \to Y$ are string diagrams

$$\bullet \otimes \ldots \otimes \bullet \otimes X \to \bullet \otimes \ldots \otimes \bullet \otimes Y$$

in the free monoidal category presented by the generators and equations.

$\mathscr{F}(\mathcal{S}, \mathcal{E})$ has a canonical premonoidal structure, given by whiskering with resources.

Mazurkiewicz traces by devices



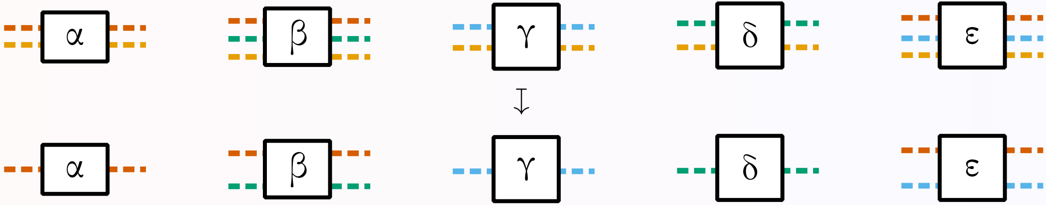
Mazurkiewicz traces arise as the morphisms of premonoidal categories freely generated by device signatures with no resource wires.

These devices may be conceived of as shared memory locations.

Serialization of traces

Often useful to consider the possible serializations of a trace.

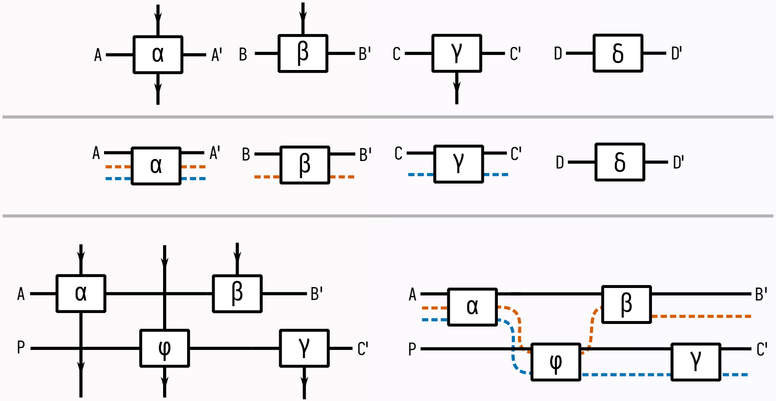
For any distributed alphabet, we can force a global order by introducing a new device,



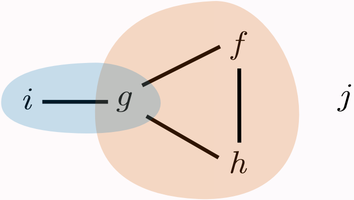
Forgetting this device determines a map between premonoidal categories, and the preimage of a trace language under this map is its serialization.

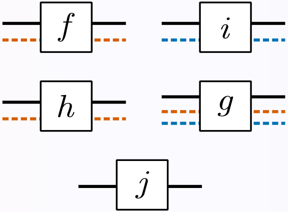
Devices from cornerings

For the free cornering of a monoidal category, we can form a premonoidal category whose morphisms $A \to B$ are cells with boundaries A and B in the resource direction [Nes22].



Interference graph of a premonoidal category



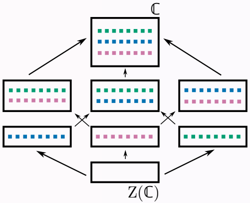


The interference graph of a sufficiently nice premonoidal category $\mathbb{C}$ determines a device presentation of $\mathbb{C}$ that contains a device for each non-trivial maximal clique.

Centralizers

The *centralizer* of a set S of morphisms in a premonoidal category contains all morphisms interchanging with elements of S.

Centralizers are premonoidal subcategories.



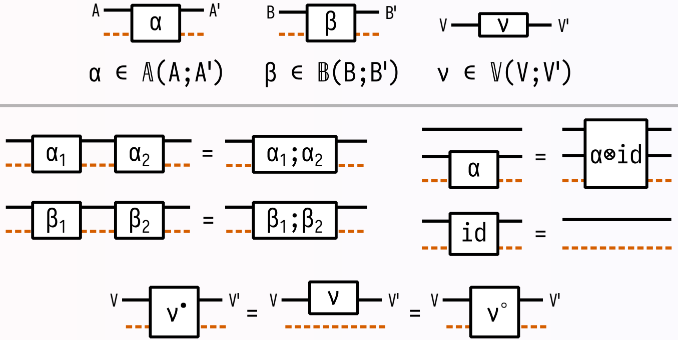
A premonoidal category $\mathbb{C}$ *admits a lattice of premonoidal subcategories, each corresponding to a subset of the devices of* $\mathbb{C}$*, bounded below by its centre* $Z(\mathbb{C})$*, and above by* $\mathbb{C}$*.*

For disjoint sets of devices D, D'*, the inclusions* $\mathbb{C}_D, \mathbb{C}_{D'} \rightrightarrows \mathbb{C}$ *form a commuting cospan in the sense of Garner and López Franco [GF16].*

Funny tensor product of effectful categories

Let $(-)^\circ : \mathbb{V} \to \mathbb{A}$ and $(-)^\bullet : \mathbb{V} \to \mathbb{B}$ be effectful categories over $\mathbb{V}$.

Their funny tensor product $\mathbb{V} \to \mathbb{A} \,\square\, \mathbb{B}$ has objects those of $\mathbb{V}$ and morphisms $X \to Y$ the morphisms $\bullet \otimes X \to \bullet \otimes Y$ in the monoidal category presented by:



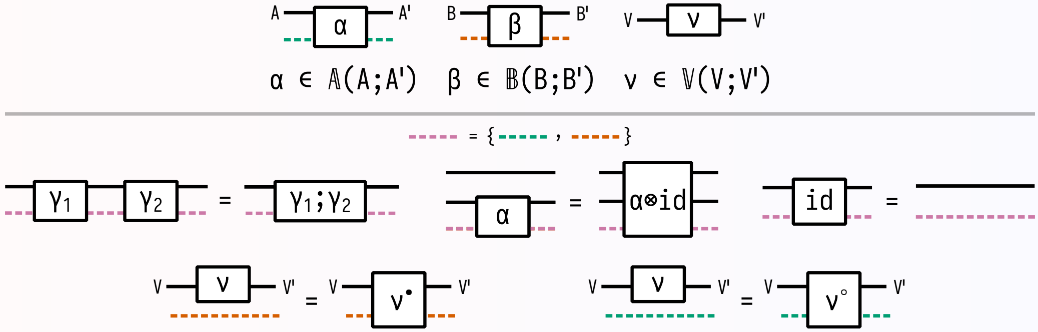
This is a special case of Román's *pure tensor product* of promonads [Rom23].

Commuting tensor product of effectful categories

Let $(-)^\circ : \mathbb{V} \to \mathbb{A}$ and $(-)^\bullet : \mathbb{V} \to \mathbb{B}$ be effectful categories over $\mathbb{V}$.

We can construct their *commuting tensor product* $\mathbb{A} \odot \mathbb{B}$, whose existence was inferred by Garner and López Franco [GF16].

Morphisms $X \to Y$ are string diagrams $\bullet \otimes \bullet \otimes X \to \bullet \otimes \bullet \otimes Y$ generated by



Commuting tensor product of effectful categories

This gives us a graphical calculus for combining effects, and recovers the string diagrams of the *functional machine calculus*.

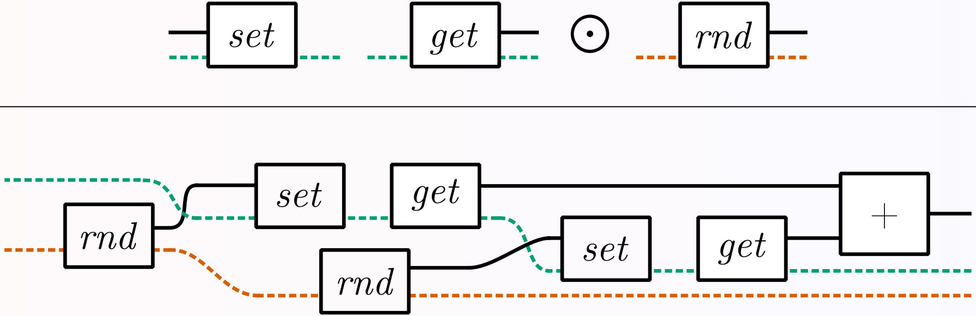
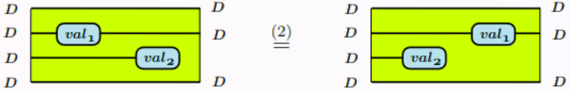


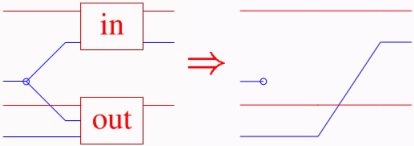
Figure: Example adapted from Barrett, Heijltjes & McCusker [BHM].

Directions and questions

- Characterization of the effectful categories with device presentations?
- Natural examples of the tensor of effectfuls over different bases?
- Relation to Melliès string diagrams for local state [Mel14]?



- Jeffrey's string diagrams for the π-calculus [Jef97]?



References

[BHM] Chris Barrett, Willem Heijltjes, and Guy McCusker, *The Functional Machine Calculus II: Semantics*, 31st EACSL Annual Conference on Computer Science Logic (CSL 2023).

[DR95] V Diekert and G Rozenberg, *The book of traces*, World Scientific, 1995.

[ENR23] Matt Earnshaw, Chad Nester, and Mario Román, *Presentations of premonoidal categories by devices*, 2023, Extended Abstract. Presented at NWPT 2023.

[ES23] Matt Earnshaw and Pawel Sobociński, *String Diagrammatic Trace Theory*, MFCS, 2023.

[GF16] Richard Garner and Ignacio López Franco, *Commutativity*, Journal of Pure and Applied Algebra **220** (2016), no. 5, 1707–1751.

[Jef97] Alan Jeffrey, *Premonoidal categories and a graphical view of programs*, Preprint (1997).

[LPT03] Paul Blain Levy, John Power, and Hayo Thielecke, *Modelling environments in call-by-value programming languages*, Information and Computation **185** (2003), no. 2, 182–210.

[Mel14] Paul-André Mellies, *Local states in string diagrams*, Rewriting and Typed Lambda Calculi: Joint International Conference, Springer, 2014, pp. 334–348.

[Nes22] Chad Nester, *Situated transition systems*, Applied Category Theory, 2022.

[PR97] John Power and Edmund Robinson, *Premonoidal categories and notions of computation*, Math. Struct. Comput. Sci. **7** (1997), no. 5, 453–468.

[Rom23] Mario Román, *Promonads and string diagrams for effectful categories*, Applied Category Theory, 2023.