# A Canonical Algebra of Open Transition Systems.

Elena Di Lavore, Alessandro Gianola, Nicoletta Sabadini, Mario Román, Paweł Sobociński.

arXiv: 2010.10069

October 28th, 2021

Formal Aspects of Component Software (FACS '21)

Supported by the European Union through the ESF Estonian IT Academy Research Measure.

# PART 1:

## SPANS OF GRAPHS

# Span(Graph), Algebra of Open Transition Systems

Many systems do not communicate by I/O message passing, but by synchronization on a common boundary.

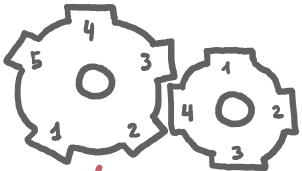
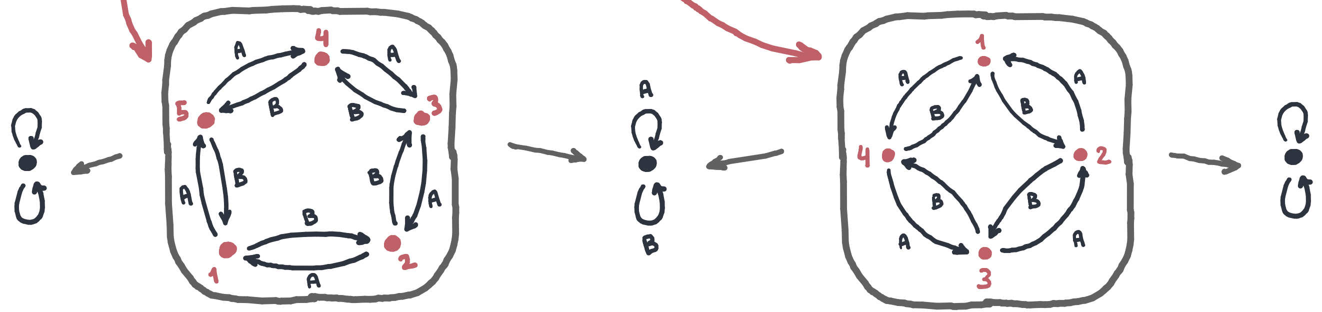



FIG 1. Two gears move together.

How to model this situation? We use spans of graphs.



# Span(Graph), Algebra of Open Transition Systems

**DEFINITION.** A span of functions from a set $A$ to a set $B$ is a pair of functions with a common domain and codomain on $A$ and $B$.

$$A \xleftarrow{f} X \xrightarrow{g} B$$

Something happens in the apex that gets reflected on the boundaries.

Spans can be composed by pullback: synchronization along a boundary.

$$\left(A \xleftarrow{f} X \xrightarrow{g} B\right) ⨾ \left(B \xleftarrow{h} Y \xrightarrow{k} C\right) =$$

$$\left(A \xleftarrow{f} X \times_B Y \xrightarrow{k} C\right)$$

where $X \times_B Y = \{(x,y) \mid x \in X, y \in Y, g(x) = h(y)\}$.

# Span(Graph), Algebra of Open Transition Systems

Each gear can be modelled as a finite state machine graph: states are vertices, and transitions are edges.

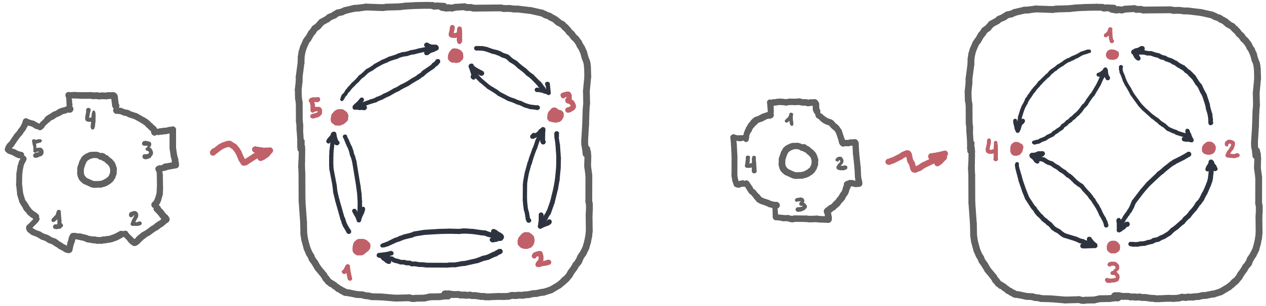



Fig 1. Two gears with 5 states and 4 states, respectively.

The next step is to give interfaces to the different components.

# Span(Graph), Algebra of Open Transition Systems

Each gear can be modelled as a finite state machine graph: states are vertices, and transitions are edges. Each graph has a left and a right interface labelling the transitions.

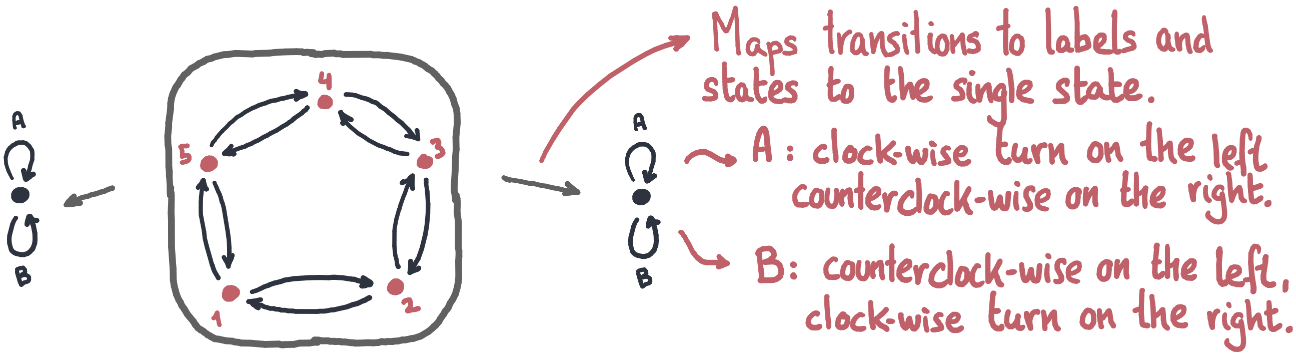
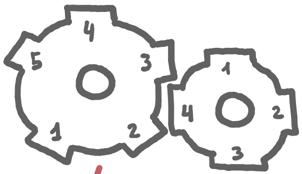


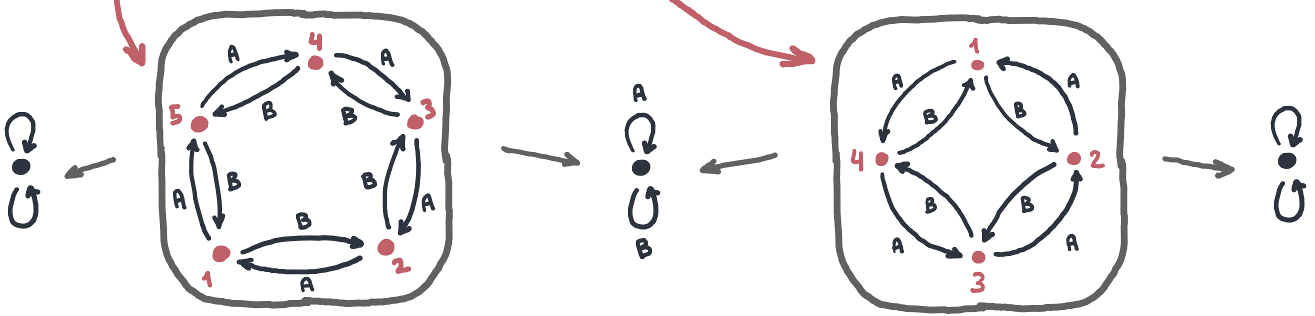

Fig 1. Gear with left and right interfaces.

# Span(Graph), Algebra of Open Transition Systems

Synchronous composition along a boundary is achieved by only allowing transitions whose image coincide on the interface.




Synchronous composition of the two gears in Span(Graph).



# Span(Graph), Algebra of Open Transition Systems


- Gianola, Kasangian, Manicardi, Sabadini, Tini. Compositional modeling of biological systems in CospanSpan(Graph). Procs. of ICTCS 2020.
- Katis, Sabadini, Walters. A formalization of the IWIM model. International Conference on Coordination Languages and Models. Springer, 2000.
- Fong. Decorated Cospans. Theory and Applications of Categories, 2015.
- Baez, Courser. Structured Cospans. Theory and Applications of Categories, 2020.
- Gianola, Kasangian, Sabadini. Cospan/Span(Graph): an Algebra of Open, Reconfigurable Automata Networks. 7th Conference on Algebra and Coalgebra in Comp. Sci. CALCO 2017.

# Span(Graph), Algebra of Open Transition Systems

Span(Graph) is a good algebra of transition systems:

1. Describes compositional, stateful transition systems.
2. Synchronization is plain composition.
3. Transition systems are encoded as graphs.
4. Boundaries may be single vertex graphs without losing any expressivity: Span(Graph)*.

**Problem:** how to justify Span(Graph)*? how is it canonical?

> "It is too easy to cook yet another variant process algebra."
>
> – S. Abramsky

# Span(Graph), Algebra of Open Transition Systems

## Plan for this talk:

1. Show that Span(Graph)* arises from adding **state** to the category of spans.
2. Show that adding state is the same as freely adding **feedback**.
3. Thus, Span(Graph)* is **canonically** the free category with feedback over spans.

"Stateful synchronization is spans of graphs."

4. Conclusions and **variants** of feedback.

# PART 2:

## STATEFUL MORPHISMS

# THE ST(•) CONSTRUCTION

**DEFINITION.** Let $(\mathbb{C}, \otimes, I)$ form a symmetric monoidal category. The category $\mathrm{St}(\mathbb{C})$ has pairs $(S, \varphi)$ as morphisms, where $S \in \mathrm{obj}\,\mathbb{C}$ is the "state space" and $\varphi : S \otimes A \to S \otimes B$ is a "stateful morphism" from A to B.

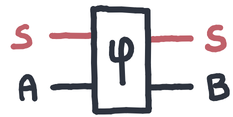



FIG 1. Stateful morphism.

Two stateful morphisms

$$\varphi : S \otimes A \to S \otimes B \quad \text{and} \quad \psi : T \otimes A \to T \otimes B$$

are equal if there exists an isomorphism of state spaces $\gamma : S \cong T$ such that $(\gamma^{-1} \otimes id_A) \circ \varphi \circ (\gamma \otimes id_B) = \psi$.

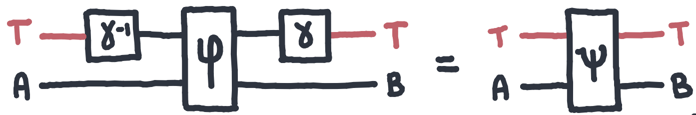



FIG 2. Equivalence relation on stateful morphisms.

# The St(•) Construction

**PROPOSITION** (Katis, Sabadini, Walters, 1997). Stateful morphisms over $\mathbb{C}, \otimes$ form a symmetric monoidal category $\mathrm{St}(\mathbb{C})$, with the same objects as $\mathbb{C}$.

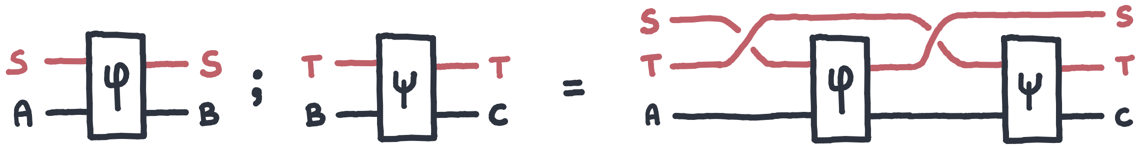



FIG 1. Sequential composition of stateful morphisms.

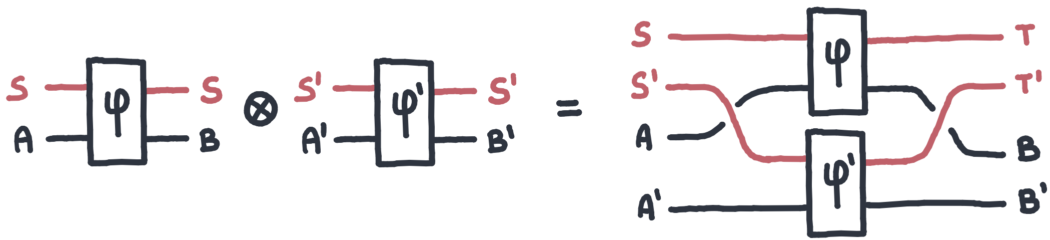



FIG 2. Parallel composition of stateful morphisms.

# The $\mathsf{St}(\bullet)$ Construction

Stateful morphisms over well-known monoidal categories form some well-known categories of transition systems.

- Let $\mathbb{C} = \mathbf{Set}^{\times}$. Stateful functions $S \times A \to S \times B$ are Mealy transition systems, $\mathsf{St}(\mathbf{Set}^{\times}) \cong \mathbf{Mealy}$.
- Let $\mathbb{C} = \mathbf{Set}^{+}$. Stateful functions $S + A \to S + B$ are Elgot transition systems, $\mathsf{St}(\mathbf{Set}^{+}) \cong \mathbf{Elgot}$.
- Let $\mathbb{C} = \mathbf{Rel}$. Stateful relations $S \times A \nrightarrow S \times B$ are Non-deterministic transition systems, $\mathsf{St}(\mathbf{Rel}) \cong \mathbf{Ndts}$.
- Let $\mathbb{C} = \mathbf{Stoch}$. Stateful stochastic functions $S \times A \to D(S \times B)$ are probabilistic transition systems, $\mathsf{St}(\mathbf{Stoch}) \cong \mathbf{Pts}$.

# The St(•) Construction

Consider the monoidal category of spans of functions with the cartesian product, $\mathsf{Span}(\mathsf{Set})^{\times}$. A morphism $A \nrightarrow B$ is a span

$$A \xleftarrow{a} E \xrightarrow{b} B.$$

What is a stateful span $S \times A \nrightarrow S \times B$? It is a span of sets, but it can also be seen as a span of graphs:

$$S \times A \xleftarrow{s \times a} E \xrightarrow{t \times b} S \times B,$$ or also

$$\begin{tikzcd}
A \arrow[d, bend right, "!"'] \arrow[d, bend left, "!"] & E \arrow[l, "a"'] \arrow[r, "b"] \arrow[d, bend right, "s"'] \arrow[d, bend left, "t"] & B \arrow[d, bend right, "!"'] \arrow[d, bend left, "!"] \\
1 & S \arrow[l, "!"] \arrow[r, "!"'] & 1
\end{tikzcd}$$

**THEOREM** (DGRSS, 2020). There exists a monoidal isomorphism

$$\mathsf{St}(\mathsf{Span}(\mathsf{Set})^{\times}) \cong \mathsf{Span}(\mathsf{Graph})^{*}.$$

"Stateful synchronization is spans of graphs."

# The St(•) Construction

Feedback and state are closely interrelated concepts. A remarkable fact from electronic circuit design is how data-storing components can be constructed from stateless components and feedback.

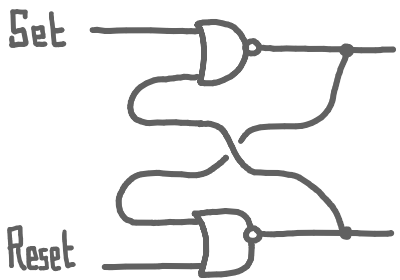



FIG 1. The Set/Reset NOR latch.

Activating any of the inputs sets the output permanently.

We will describe a construction that adds "state" to a monoidal category. The same construction will add "feedback".

# Part 3:

## Feedback

# CATEGORIES WITH FEEDBACK

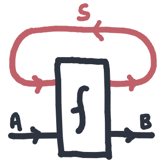



FIG 1. Notation.

**DEFINITION** (Katis, Sabadini, Walters, 1997). A category with feedback is a symmetric monoidal category endowed with an operator

$$\mathsf{fbk}_S : \hom(S \otimes A, S \otimes B) \to \hom(A, B),$$ such that

## AXIOMS.

1. $u \circ \mathsf{fbk}_S(f) \circ v = \mathsf{fbk}_S((u \otimes id) \circ f \circ (v \otimes id))$,
2. $\mathsf{fbk}_I(f) = f$,
3. $\mathsf{fbk}_S(\mathsf{fbk}_S(f)) = \mathsf{fbk}_{S \otimes T}(f)$,
4. $\mathsf{fbk}_S(f) \otimes g = \mathsf{fbk}_S(f \otimes g)$,
5. $\mathsf{fbk}_S(f \circ (h \otimes id)) = \mathsf{fbk}_T((h \otimes id) \circ f)$,
   when $h$ is an isomorphism.

# Categories with Feedback

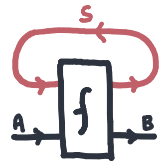


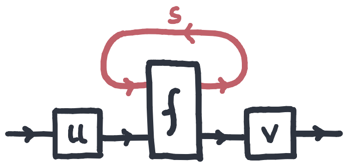

Fig 1. Notation.

**Definition** (Katis, Sabadini, Walters, 1997). A category with feedback is a symmetric monoidal category endowed with an operator

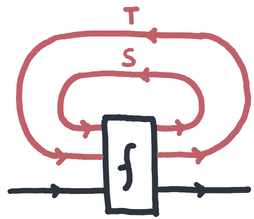
$$\mathrm{fbk}_S : \hom(S \otimes A, S \otimes B) \to \hom(A, B), \quad \text{such that}$$

## Axioms.

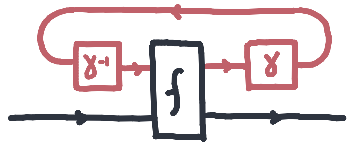
1.



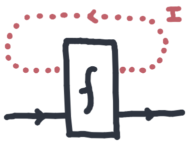

3.



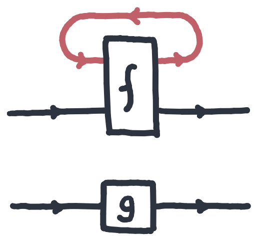

5.




2.




4.



# CATEGORIES WITH FEEDBACK

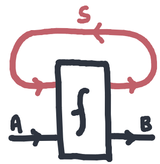


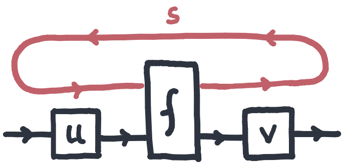

FIG 1. Notation.

**DEFINITION** (Katis, Sabadini, Walters, 1997). A category with feedback is a symmetric monoidal category endowed with an operator

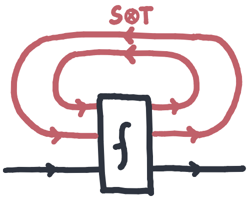
$$\mathrm{fbk}_S : \hom(S \otimes A, S \otimes B) \to \hom(A, B), \quad \text{such that}$$

## AXIOMS.

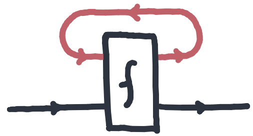
1.



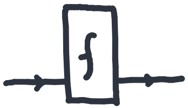

3.



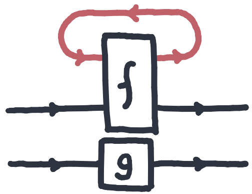

5.




2.

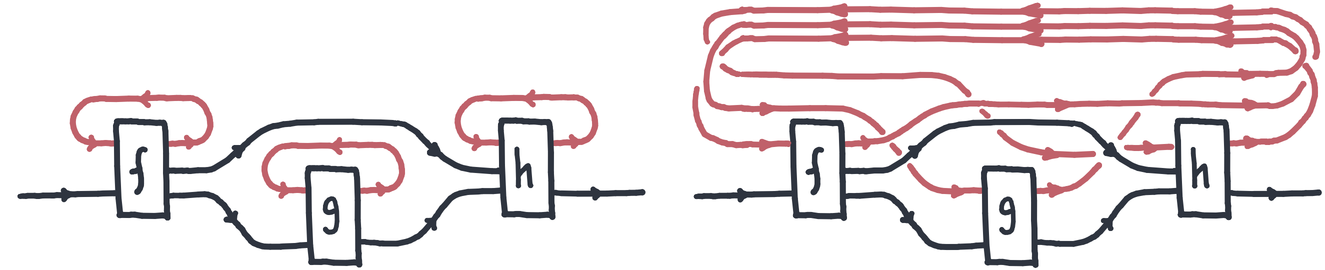



4.



# Categories with Feedback

**THEOREM** (Katis, Sabadini, Walters, 2002). Diagrams in a category with feedback have a normal form with a single feedback loop.




**COROLLARY** (DGRSS, 2020). The category $\mathsf{St}(\mathbb{C})$ of stateful morphisms is the free category with feedback over $\mathbb{C}$. There exists an adjunction

$$\begin{tikzcd} \mathsf{SymMon} \arrow[r, bend left, "\mathsf{St}"] \arrow[r, phantom, "\perp"] & \mathsf{Feedback} \arrow[l, bend left, "\mathsf{U}"] \end{tikzcd}$$

$$\mathsf{St}(\mathsf{Span}(\mathsf{Set})^{\times}) \cong \mathsf{Span}(\mathsf{Graph})^{*}.$$

"Feedback & synchronization is spans of graphs."

# Part 4:

## Generalizing $St(\cdot)$

# Conclusion

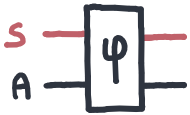
- **Main Result:** a monoidal isomorphism $\mathsf{St}(\mathsf{Span}(\mathsf{Set})^{\times}) \cong \mathsf{Span}(\mathsf{Graph})^{*}$.
- **Consequence:** a universal property for $\mathsf{Span}(\mathsf{Graph})^{*}$.
- **More Generally:** the $\mathsf{St}(\bullet)$ construction from the literature is the free category with feedback.
- Categories with **Feedback** are a weakening of traces that still have a normal form theorem.
- The $\mathsf{St}(\bullet)$ construction can be generalized to other variants of feedback.

## Future Work.

- Variants of feedback, how to categorically describe them.
- Applications to stream-based programming.

# GENERALIZING THE ST(•) CONSTRUCTION

What is the most general form of state? What is the most general form of feedback?




S ranges over arbitrary monoidal categories.

S acts on A and B with an arbitrary monoidal action.

FIG 1. Stateful morphism.

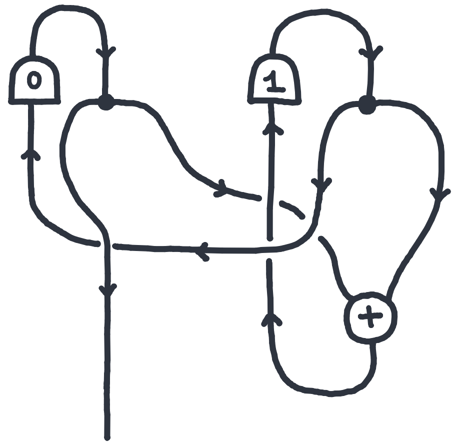



FIG 1. Fibonacci, using initialized feedback.

We recover multiple variants of feedback.

1. Guarded feedback.
2. Initialized feedback.
3. Delayed feedback.

# References


Katis, Sabadini, Walters. Feedback, Trace, and Fixed-Point Semantics.
RAIRO-Theoretical Informatics and Applications. Tome 36, 2002.

Katis, Sabadini, Walters. A Categorial Algebra of Transition Systems.
AMAST Proceedings. Ed: Michael Johnson. Springer, 1997.

Bonchi, Holland, Piedeleu, Sobociński, Zanasi. Diagrammatic Algebra: from Linear to Concurrent Systems.
Proc. ACM Program. Lang. (POPL '19).

Katsumata, Sprunger. Differentiable Causal Computations via Delayed Trace.
Annual ACM/IEEE Symposium on Logic in Computer Science (LICS), 2019.

Hoshino, Muroya, Hasuo. Memoryful Geometry of Interaction.
Proc. Computer Science Logic (CSL), 2014.

Román. Open Diagrams via Coend Calculus.
Applied Category Theory, 2020.

# CATEGORIES WITH FEEDBACK

What are the differences with traced monoidal categories?

1. Only isomorphisms can be slided.


2. The yanking equation does not hold.


More technically,

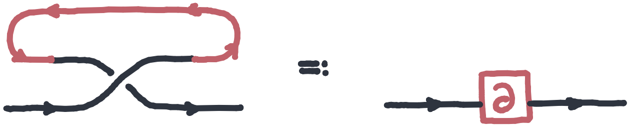
- feedback is weaker than trace, and
- feedback coincides with guarded trace in compact closed categories.

# CATEGORIES WITH FEEDBACK

What are the differences with traced monoidal categories?

1. Only isomorphisms can be slided.


2. The yanking equation does not hold. There is a delay.




More technically,

- feedback is weaker than trace, and
- feedback coincides with guarded trace in compact closed categories.