Tracing Monoidal Man in the Middle (with hindsight)

Dusko Pavlovic

Royal Holloway

CMCS, Tallinn
1 April 2012

Outline

Category of protocols

Arities
Clones
Programs
Protocols
Protocol analysis

Trace monad
Future traces

Category of arities

\[ |\text{A}| = \{ n = [0, 1, \ldots, n-1] | n \in \mathbb{N} \} \]

\[ \text{A}(m, n) = \left\{ (x_0, \ldots, x_{m-1}), (x_0(0), \ldots, x_{n-1}(0)) | f : n \to m \right\} / \alpha \]

Remark

\( \text{A} \) is the free strict cartesian category generated by 1.

Algebraic theories

- An algebraic theory is a pair \( T = \langle \Sigma_T, E_T \rangle \) where
  - \( \Sigma = \Sigma_T \) is a signature, and
  - \( E = E_T \) is a set of equations
Algebraic theories

- An algebraic theory is a pair $\mathcal{T} = (\Sigma, E)$ where
  - $\Sigma = \Sigma_T$ is a signature,
  - $E = E_T$ is a set of equations.
- Algebraic operations $\varphi(x_1, \ldots, x_n)$ are generated from $\Sigma$ modulo $E$.
- A $\mathcal{T}$-algebra assigns a map $\varphi : X^n \to X$ for each $n$-ary algebraic operation $\varphi$.

Programming languages

- A programming language $\mathcal{L}$ generates programs $P$ with inputs $(x_0, \ldots, x_{m-1})$ and outputs $(s_0, \ldots, s_{n-1})$.
- Semantics of $\mathcal{L}$ assigns a map $P : \prod_{i=0}^{m} X_i \to \prod_{j=0}^{n} S_j$ for each program $P$ with $m$ inputs and $n$ outputs.

Category of programs (untyped)

$$|\mathcal{C}_\mathcal{T}| = N \quad \mathcal{C}_\mathcal{T}(m, n) = \{ (x_0, \ldots, x_{m-1})|\varphi(\varphi(\ldots, \varphi(x_0), \ldots)) \}/\alpha$$

Category of programs

Remark

- typing $\leftrightarrow$ extended arities
Category of programs

Remark

- typing $\leftrightarrow$ extended arities
- sequential composition $\leftrightarrow$ arrow composition
- parallel composition $\leftrightarrow$ monoidal structure
- program loops $\leftrightarrow$ trace structure

Category of programs: String diagrams

Remark

- typing $\leftrightarrow$ extended arities
- sequential composition $\leftrightarrow$ arrow composition
- parallel composition $\leftrightarrow$ monoidal structure
Category of programs: Composition

\[
\delta_{ij} \rightarrow (P, Q)(\vec{x}, \vec{y}) \rightarrow \delta_{ij}(\vec{x}, \vec{y})
\]

Category of programs: Tensor

\[
\begin{align*}
&X_1 X_2 \ldots X_p \rightarrow P \quad \emptyset \\
&X'_1 X'_2 \ldots X'_p \rightarrow P' \quad \emptyset \\
&X_1 \ldots X_p X'_1 \ldots X'_p \\
= &RP P \quad \emptyset \quad \emptyset
\end{align*}
\]

where \( \vec{x}_i \cap \vec{x}'_i = \emptyset \)

Category of programs: Trace

\[
\begin{align*}
&Tr_{\vec{x}, \vec{y}} \left( \begin{array}{c}
\delta_{ij} \\
\vec{x}_i \\
\vec{y}_j
\end{array} \right) \rightarrow \delta_{ij} \\
&\vec{x}_i \\
\vec{y}_j
\end{align*}
\]

But verifying the trace structure is a lot of work!
Category of programs: Trace

But verifying the trace structure is a lot of work!

If the traces are OK, we model protocols as follows:

\[
\begin{align*}
\mathcal{P} & = \left\{ A = (A_1, A_2) \in \mathcal{P}^2 \right\} \\
\mathcal{J}(A, B) & = \left\{ A \circ B \rightarrow A \circ B \text{ in } \mathcal{P} \right\} \\
\left( \delta_{A_1}, \delta_{A_2} \right) & = \left\{ \delta_1, \delta_2 \rightarrow \delta_1, \delta_2 \right\}
\end{align*}
\]

Composition of protocols

Categorical protocol analysis

A protocol is

- specified by giving
  1. local observations at the interfaces
  2. a (security) requirement on the interactions
- represented as a morphism \( A \xrightarrow{\rho} B \) in \( \mathcal{J} \)
- analyzed by enumerating

**Man-in-the-Middle**: nontrivial decompositions

\( A \rightarrow X \rightarrow B \)

**Chosen Protocol Attack**: leaking compositions

\( A \rightarrow B \rightarrow X \)

Local monoids

**Definition**

Let \( \mathbb{M} \times \mathbb{M} \rightarrow \mathbb{M} \rightarrow 1 \) be a commutative monoid.

- \( o \in \mathbb{M} \) is zero if \( au = o \) for all \( u \in \mathbb{M} \)
- \( s \in \mathbb{M} \) is a nilpotent if \( s^n = o \) for some \( n \in \mathbb{N} \)
- \( s \in \mathbb{M} \) is regular if it is not nilpotent.
- \( \mathbb{M} \) is local if all regular elements are invertible
Local monoidal categories

Definition
Let \( \mathcal{C} \times \mathcal{C} \xrightarrow{\otimes} \mathcal{C} \xleftarrow{1} \mathcal{I} \) be a small symmetric monoidal category. Then

- \( \mathcal{I} = \mathcal{C}(I, I) \) is a commutative monoid
- \( \mathcal{I} \times \mathcal{C}(A, B) \to \mathcal{C}(A, B) \) is an \( I \)-action
- \( A \xrightarrow{\circ_{AB}} B \) is zero if \( \circ \cdot \circ_{AB} = \circ_{AB} \)
- \( \mathcal{C} \) is a local monoidal category if \( \mathcal{I} \) is a local monoid.

Graded monoidal categories

Definition
A small strict symmetric monoidal category \( \mathcal{C} \times \mathcal{C} \xrightarrow{\otimes} \mathcal{C} \xleftarrow{1} \mathcal{I} \) is graded by a monoid homomorphism

\[
(\mathcal{C}, \otimes, \mathcal{I}) \xrightarrow{| \cdot |} (\mathcal{I}, \circ, \text{id})
\]

where \( \mathcal{I} = \mathcal{C}(I, I) \).

Remark
Every traced monoidal category is graded by

\[
|U| = T^{U}(\text{id}_U)
\]

Normal traces

Definition
A normal trace structure over a monoidal category \( \mathcal{C} \) is a family of operators \( f : A \otimes_U B \to B \otimes_U B \) satisfying the Joyal-Street-Verity axioms and also:

\[
T^{U}_{AB}(f) = \begin{cases} 
A \xrightarrow{\circ} B & \text{if } U \text{ is regular} \\
A \xrightarrow{0} B & \text{otherwise}
\end{cases}
\]

Trace decomposition

Proposition 1
Any trace structure over a local monoidal category \( \mathcal{C} \) decomposes:

\[
T^{U}(f) = |U| \cdot T^{U}(f)
\]

Free normal traces

Proposition 2
Normal traces are monadic over local monoidal categories.
Loop categories

Definition

The loop category over a small local monoidal category $C$ is

$$\mathcal{C}^\perp(A, B) = \int_{U \in \mathcal{C}} C(A \otimes U, B \otimes U)$$

...where $\sim$ extends the coend equivalence...

$$\begin{array}{ccc}
A \otimes U \otimes U & \sim & A \otimes U \\
B \otimes U \otimes U & \sim & B \otimes U \\
\end{array}$$

...by

$$\begin{array}{ccc}
f \otimes_{C_U} s & \sim & f \otimes s \\
A \otimes U \otimes U & \sim & A \otimes U \\
B \otimes U \otimes U & \sim & B \otimes U \\
\end{array}$$

...and

$$\begin{array}{ccc}
f \otimes s & \sim & g \otimes t \\
\exists u, v \in \mathcal{C}. u \circ f = v \circ g \land u \circ s = v \circ t
\end{array}$$
Composition

Given

- \( f \in \mathbb{C}^{\mathbb{C}}(A, B) \) as \( A \otimes U \xrightarrow{h} B \otimes U \), and
- \( g \in \mathbb{C}^{\mathbb{C}}(B, C) \) as \( B \otimes V \xrightarrow{h/g} C \otimes V \),

the composite

- \( f \circ g \in \mathbb{C}^{\mathbb{C}}(A, C) \) can be viewed as

\[
\begin{align*}
A \otimes U \otimes V & \xrightarrow{f} B \otimes U \otimes V \\
& \xrightarrow{g} C \otimes U \otimes V \\
& \xrightarrow{h} C \otimes V \otimes U
\end{align*}
\]

Tensor

Given

- \( f \in \mathbb{C}^{\mathbb{C}}(A, B) \) as \( A \otimes U \xrightarrow{h} B \otimes U \), and
- \( h \in \mathbb{C}^{\mathbb{C}}(C, D) \) as \( C \otimes V \xrightarrow{h/g} D \otimes V \),

the tensor product

- \( f \otimes h \in \mathbb{C}^{\mathbb{C}}(A \otimes C, B \otimes D) \) can be viewed as

\[
\begin{align*}
A \otimes C \otimes U \otimes V & \xrightarrow{f \otimes h} B \otimes D \otimes U \otimes V \\
& \xrightarrow{h} B \otimes D \otimes V
\end{align*}
\]

Free normal traces

Proposition 2 (continuation)

The free normal traces are given by the loop monad:

- 2-functor \( \Theta : \mathcal{L}M \rightarrow \mathcal{L}M \)
- unit functors

\[
\eta_C : C \rightarrow \mathbb{C}^{\mathbb{C}}(A^{1 \otimes B}, B^{1 \otimes B})
\]
- evaluation functors

\[
\mu_C : \mathbb{C}^{\mathbb{C}}(A^{1 \otimes B}, B^{1 \otimes B}) \rightarrow A^{1 \otimes B} \otimes B^{1 \otimes B}
\]

Normal trace

- \( f \in \mathbb{C}^{\mathbb{C}}(A, B) \) as \( A \otimes U \xrightarrow{h} B \otimes U \), and
- \( g \in \mathbb{C}^{\mathbb{C}}(B, C) \) as \( B \otimes V \xrightarrow{h/g} C \otimes V \),

the composite

- \( f \circ g \in \mathbb{C}^{\mathbb{C}}(A, C) \) can be viewed as

\[
\begin{align*}
A \otimes U \otimes V & \xrightarrow{f} B \otimes U \otimes V \\
& \xrightarrow{g} C \otimes U \otimes V \\
& \xrightarrow{h} C \otimes V \otimes U
\end{align*}
\]

Free normal traces

Proposition 2 (continuation)

The free normal traces are given by the loop monad:

- 2-functor \( \Theta : \mathcal{L}M \rightarrow \mathcal{L}M \)
- unit functors

\[
\eta_C : C \rightarrow \mathbb{C}^{\mathbb{C}}(A^{1 \otimes B}, B^{1 \otimes B})
\]
- evaluation functors

\[
\mu_C : \mathbb{C}^{\mathbb{C}}(A^{1 \otimes B}, B^{1 \otimes B}) \rightarrow A^{1 \otimes B} \otimes B^{1 \otimes B}
\]
Free normal traces

Corollary
\[ \mathcal{P}_\mathcal{C}(m, n) = \{ (s_0, \ldots, s_{m-1}) | [P](s_0, \ldots, s_{m-1}) \} / \sim \]

is a normal traced monoidal category.

Outline

Category of protocols

Trace monad

Future traces

Weak distributivity

Weak traces

Weak interactions

Weakly distributive categories

Definition

A weakly distributive category \( C \) carries two (symmetric!) monoidal structures

\[ C \times C \xrightarrow{T} \mathbb{1} \]

\[ C \times C \xrightarrow{\bot} \mathbb{1} \]

together with

\[ A \otimes (B \oplus C) \xrightarrow{d_r} (A \otimes B) \oplus C \]

\[ m \xrightarrow{\top} \]

cohently...

Weak traces

Definition

The normal trace operator over a weakly distributive category \( C \) is a family of operators

\[ f : A \otimes U \rightarrow B \oplus U \]

\[ T_{nU}^A f : A \rightarrow B \]

satisfying the well-typed versions of the familiar laws.\(^1\)

\(^1\)Vanishing can be weakened!

Weak interactions

Definition

Given a weakly distributive traced category \( C \), the induced interaction category \( W_{\mathcal{C}} = \text{Int}(C) \) consists of

\[ \mathcal{I}(W_{\mathcal{C}}) = \{ [A] = (A_+, A_-) \in \mathbb{K}^2 \} \]

\[ \mathcal{I}(A, B) = \{ [A_+, B_-] \rightarrow A_+ \otimes B_- \text{ in } C \} \]

where

\[ A_+ \otimes B_- \xrightarrow{g} A_+ \otimes B_- \]

\[ B_+ \otimes C_- \xrightarrow{f} B_+ \otimes C_- \]

\[ g \circ f = T_{nU}^B f(A_+ \otimes B_- \oplus C_-) \]

\[ \text{id}_A = (A_+ \otimes A_- \xrightarrow{\text{id}_A} A_+ \otimes A_-) \]

Weak interactions

Conjecture

\( C \rightarrow \text{Int}(C) \) is the free \( \ast \)-autonomous weakly traced category.
Weak interactions

Conjecture

\[ C \to \text{Int}(C) \] is the free \( \ast \)-autonomous weakly traced category.

Question

What does this mean logically? Geometrically?