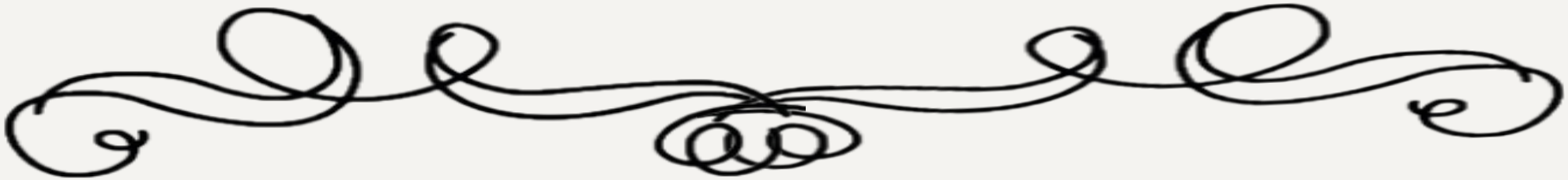
Semantics of CaMPL - Showing the Power Set Functor is Monoidal



Alexanna (Xanna) Little
with Dr. Robin Cockett, University of Calgary
FMCS 2024, Kananaskis

Background/Spoilers

- Categorical Message Passing Language (CaMPL) is a concurrent functional programming language which has its semantics defined in category theory.
- Semantics of non-deterministic programs is given by a sup-lattice enriched category which is created by a change of base from the original semantics.
- Geoff Crutwell's PhD thesis showed that we can define a change of base functor if there is a monoidal functor between the "base" categories.
    - He also describes how monoidal functors can be applied "monoidally" and that applying functors monoidally is "functorial."
- Power set functor gives a monoidal functor that takes sets to sup-lattices.

Change of Base

- We want to define the semantics of non-deterministic programs as a sup-lattice enriched version of the deterministic semantics.
- We also want to ensure the monoidal structures in the semantics, which define some of the other programming features in the language, are preserved.
- So we need a functor that takes the deterministic semantics which are set-enriched to the non-deterministic semantics which are sup-lattice enriched and preserves the monoidal structures in the semantics.

Change of Base

According to results from Geoff Cruttwell's PhD thesis, a change of base functor exists if there is a monoidal functor between the monoidal "base" categories.

- Theorem 5.7.1. Suppose that $N : (V, \bullet, I, \sigma_1) \to (W, \otimes, J, \sigma_2)$ is a braided monoidal functor between braided monoidal categories. Then the change of base 2-functor N_* takes monoidal V-categories to monoidal W-categories.

This means that if we can define a monoidal "N functor" of our own from **Set** to **SupLat**, we also get a change of base functor from the deterministic semantics to the non-deterministic semantics.

Monoidal Functors

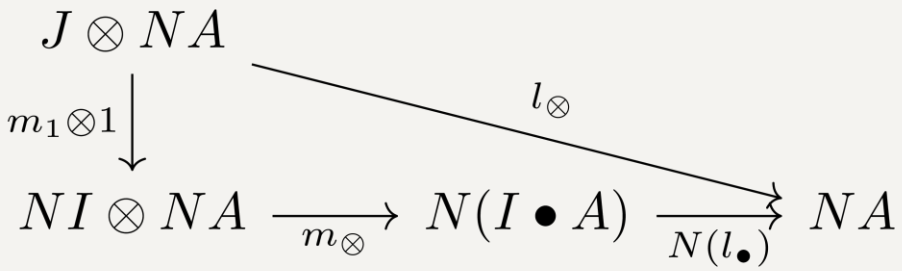
A monoidal functor between monoidal categories $(V, \bullet, I)$ and $(W, \otimes, J)$, is a functor $N : V \to W$, together with natural transformations, called comparison maps:

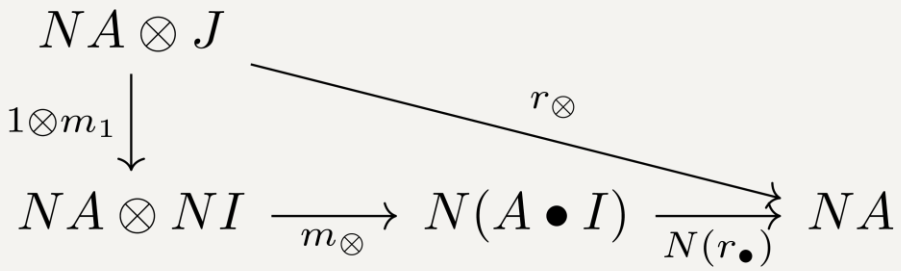
- $m_{\otimes AB} : NA \otimes NB \to N(A \bullet B)$
- $m_1 : J \to NI$

For which the following coherence diagrams commute $\forall A, B, C \in V$.

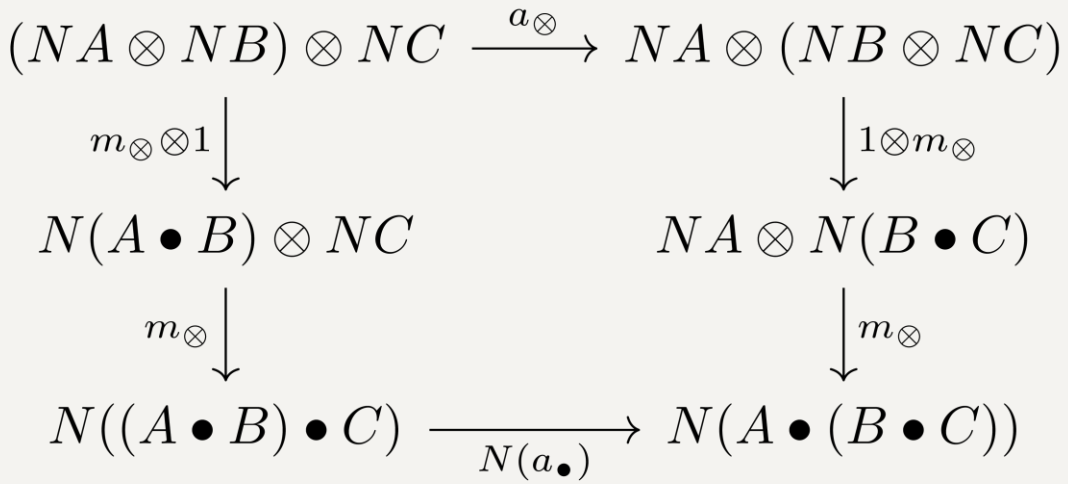
Monoidal Functors

- Left and right unitors:





- Associativity:



What is a Sup-Lattice?

- A **sup-lattice** $\mathcal{L} = (L, \bot, \vee)$ has carrier L, bottom element $\bot \in L$, and join operation $\vee$.
- A partial ordering is given by $\vee$ which can be expressed as $x \leq y \iff x \vee y = y$ and $\bot \leq x$.
- Additionally, sup is a unique map from elements to their **least upper bound** (supremum) element if it exists, so $x \leq sup(x, y)$, $y \leq sup(x, y)$ and $x \leq z,\ y \leq z \implies sup(x, y) \leq z$.
- We can also express the notion of arbitrary sups as joins: $x_i \leq \bigvee_{i \in I} x_i$ and $x_i \leq z,\ \forall i \in I \implies \bigvee_{i \in I} x_i \leq z$

Power Set Functor

- Let $\mathscr{P}$: **Set** $\to$ **Set** denote the power set functor. We can write $\mathscr{P} = \mathcal{PU}$ where $\mathcal{P}$: **Set** $\to$ **SupLat** lifts a set to a sup-lattice where the carrier is its power set and $\bot := \emptyset, \vee := \cup$ and $\mathcal{U}$: **SupLat** $\to$ **Set** is the underlying functor that maps a sup-lattice back to the set that is its carrier.
- These functors form an adjunction $(\eta, \epsilon) : \mathcal{P} \dashv \mathcal{U} :$ **Set** $\to$ **SupLat** with $\eta : 1_{\textbf{Set}} \Rightarrow \mathcal{PU}$ which sends elements to singletons.
  An abstract component map is $\eta_X : X \to \mathcal{PU}(X)\ ;\ x \mapsto \{x\}$.
  The power set is formed by the sup-lattice with bottom element $\emptyset$ and the singletons are joined with $\cup$.

SupLat is a Monoidal Category

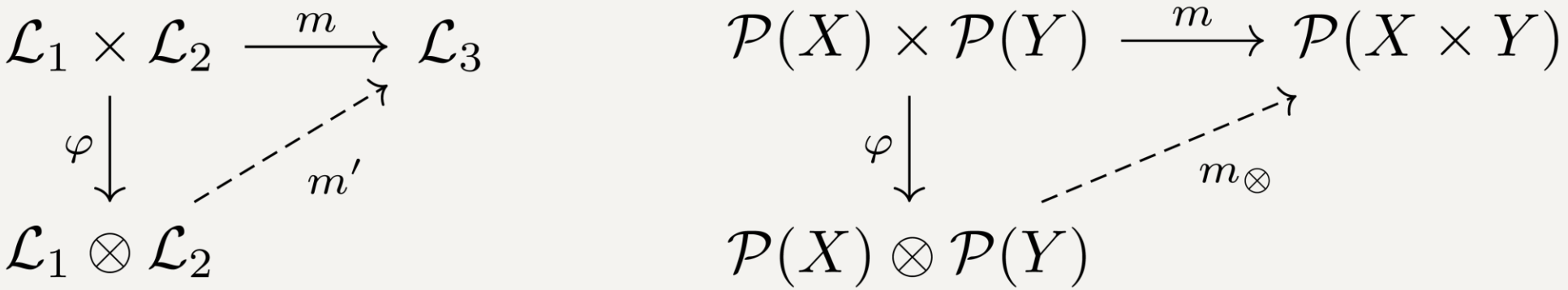
- **SupLat** is the category of sup-lattices where the objects are sup-lattices and the morphisms are sup-preserving maps.
- A sup-preserving map ensures the sup for a set of elements is mapped to the sup for the mappings of those elements: $f(\bigvee_{i \in I} x_i) = \bigvee_{i \in I} f(x_i)$
- The unit sup-lattice I has carrier $\{\top, \bot\}$.
- The tensor of sup-lattices, $\mathcal{L}_1 \otimes \mathcal{L}_2$, creates a sup-lattice with carrier $\{\bigvee(x_1 \otimes x_2) \mid x_1 \in L_1, x_2 \in L_2\}$, where $\bot \otimes x = \bot = x \otimes \bot$.
  Maps out of the tensor are sup-preserving and bi-sup-preserving:
  $(\bigvee_{i \in I} x_i) \otimes y = \bigvee_{i \in I} (x_i \otimes y) \quad x \otimes (\bigvee_{j \in J} y_j) = \bigvee_{j \in J} (x \otimes y_j)$

Power Set Functor is a Monoidal Functor

- We want to show that $\mathcal{P}$: **Set** $\to$ **SupLat** is a monoidal functor; that is, the monoidal structure in **Set** $(\times, *, a_\times, l_\times, r_\times)$ is lifted to the monoidal structure in **SupLat** $(\otimes, I, a_\otimes, l_\otimes, r_\otimes)$.
- To show that $\mathcal{P}$ is a monoidal functor, we must define comparison maps and show that the necessary coherence diagrams commute.
- $m_1 : I \to \mathcal{P}(*)$ is given by $\top \mapsto *, \bot \mapsto \emptyset$.
- $m_\otimes : \mathcal{P}(X) \otimes \mathcal{P}(Y) \to \mathcal{P}(X \times Y)$ is given by the universal property of the tensor of sup-lattices.

Power Set Functor is a Monoidal Functor

- The diagram on the left is the universal property of the tensor which says, if we have a bi-sup-preserving map m, there exists the map m':



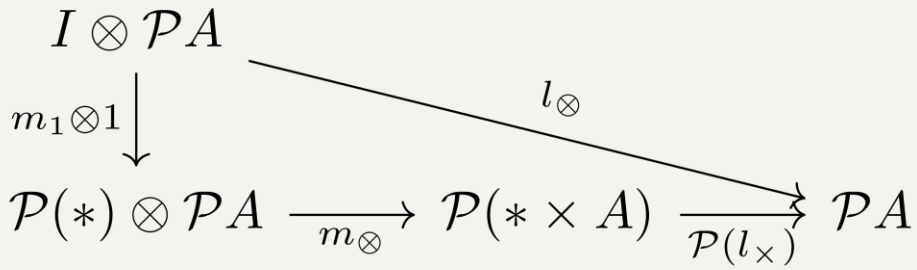
- If we set $\mathcal{L}_3 := \mathcal{P}(X \times Y)$ and show the map m exists, the m' map is the $m_\otimes$ map we want, as shown in the diagram on the right.

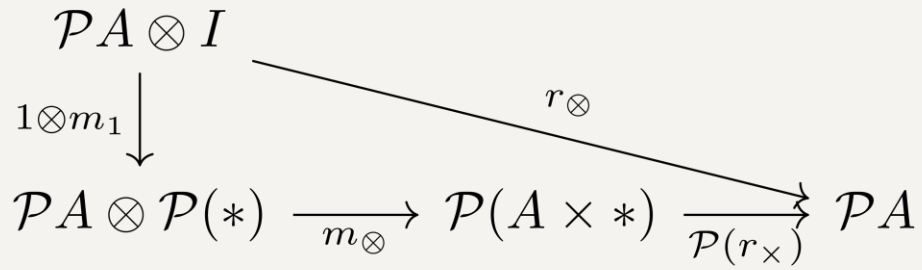
Power Set Functor is a Monoidal Functor

- Suppose (U, V) is an element of the carrier of $\mathcal{P}(X) \times \mathcal{P}(Y)$.
- Define $m : (U, V) \mapsto \{(u, v) \mid u \in U, v \in V\}$.
- Since m is a map out of the product of sup-lattices, it must be bi-sup-preserving, so we want to show that $m(\bigvee_i U_i, V) = \bigvee_i m(U_i, V)$ as well as the symmetric case.
- By definition, $m(\bigvee_i U_i, V) \mapsto \{(u, v) \mid u \in \bigvee_i U_i, v \in V\}$, and $\bigvee_i m(U_i, V) \mapsto \bigvee_i \{(u, v) \mid u \in U_i, v \in V\}$.
- These are equivalent as arbitrary elements from each are included in the other, so we have defined our m map and therefore $m_\otimes$.

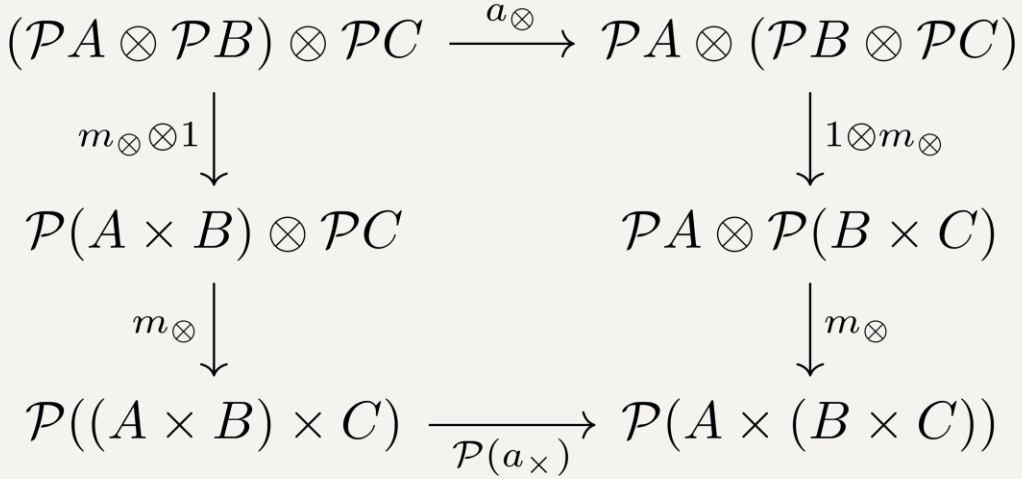
Power Set Functor is a Monoidal Functor

- Left and right unitors:





- Associativity:



Change of Base

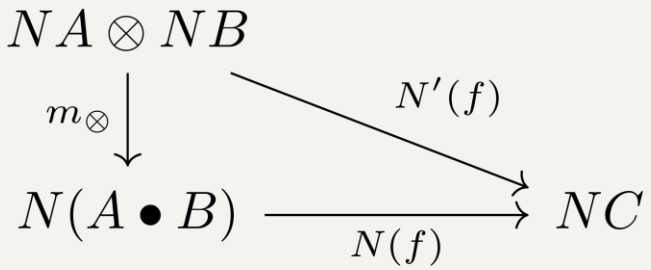
Again, we use results from Geoff's PhD thesis to define the category produced by a change of base.

- Proposition 4.2.1. Let N be as defined, and X a V-category. Then the following defines a W-category N_*X:
  - N_*X has the same objects as those of X,
  - hom-objects are defined by $(N_*X)(x, y) := N(X(x, y))$,
  - composition is N applied monoidally to the composition in X,
  - identities are N applied monoidally to the identities in X.

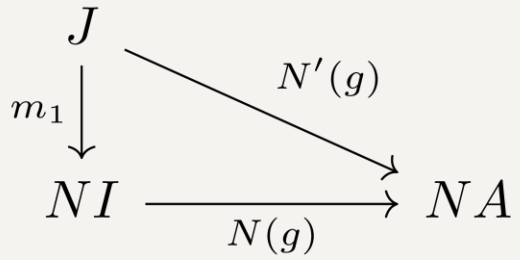
Applying Monoidal Functors Monoidally

There are two ways to apply a monoidal functor N "monoidally":

- Given an arrow in V,
  $f : A \bullet B \to C$
  we apply N monoidally to get an arrow in W,
  $N'(f) : NA \otimes NB \to NC$
  defined as the composite:



- Given an arrow in V,
  $g : I \to A$
  we apply N monoidally to get an arrow in W,
  $N'(g) : J \to NA$
  defined as the composite:



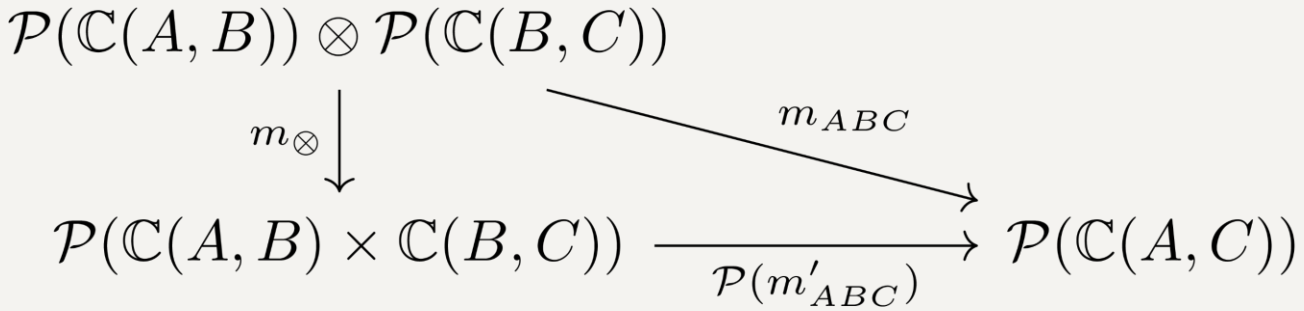
Sup-Lattice Enrichment

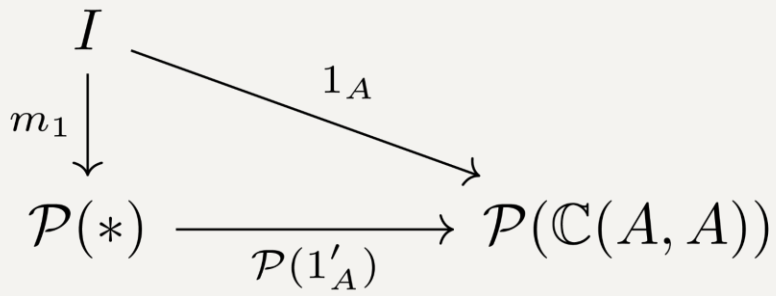
Let $\mathbb{C}$ be the set-enriched deterministic semantics category, then define $\mathcal{P}_*\mathbb{C}$ as the sup-lattice enriched non-deterministic semantics category produced by the change of base:

- Objects : Obj $(\mathbb{C})$
- Hom-objects : $(\mathcal{P}_*\mathbb{C})(A, B) := \mathcal{P}(\mathbb{C}(A, B))$
- Identity : $\mathcal{P}$ applied monoidally to the identities in $\mathbb{C}$ to produce: $1_A : I \to \mathcal{P}(\mathbb{C}(A, A))$
- Composition : $\mathcal{P}$ applied monoidally to the composition in $\mathbb{C}$ to produce: $m_{ABC} : \mathcal{P}(\mathbb{C}(A, B)) \otimes \mathcal{P}(\mathbb{C}(B, C)) \to \mathcal{P}(\mathbb{C}(A, C))$

Sup-Lattice Enrichment

- Composite diagrams for composition and identity maps:





References

1. J. R. B. Cockett and Craig Pastro. The Logic of Message Passing. Science of Computer Programming, 74(8):498–533, 2009. (Determinsitic semantics for CaMPL.)

2. G. S. H. Cruttwell. Normed Spaces and the Change of Base for Enriched Categories. PhD thesis, Dalhousie University, Halifax, Nova Scotia, December 2008.
(Enriched categories, change of base, monoidal functors.)

3. Andre Joyal and Myles Tierney. An extension of the Galois theory of Grothendieck, volume 309 of Memoirs of the American Mathematical Society. American Mathematical Society, Providence, R.I, 1984. (Sup-lattices.)

4. Reginald Lybbert. Progress for the Message Passing Logic. Undergraduate thesis, University of Calgary, April 2018. Provided by the author. (Progress for deterministic CaMPL programs.)

5. Jared Pon. Implementation Status of CMPL. Undergraduate thesis Interim Report, University of Calgary, December 2021. Provided by the author. (Races in CaMPL.)

6. Masuka Yeasin. Linear Functors and their Fixed Points. Master's thesis, University of Calgary, Calgary, Alberta, December 2012. Provided by the author.
(Deterministic protocols and coprotocols.)