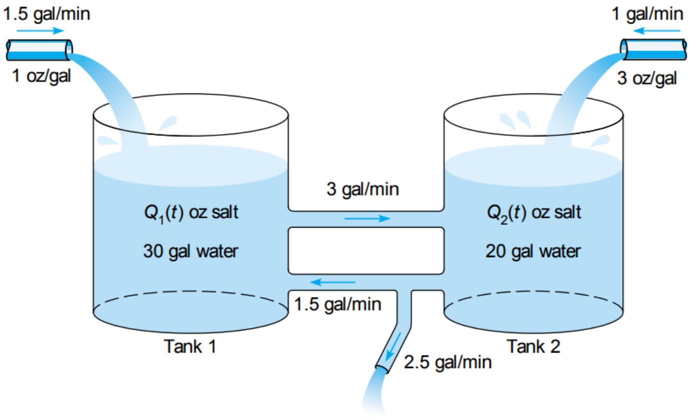
Algebras of Open Systems on the Operad of Wiring Diagrams

Dmitry Vagner

Joint work with David I. Spivak and Eugene Lerman

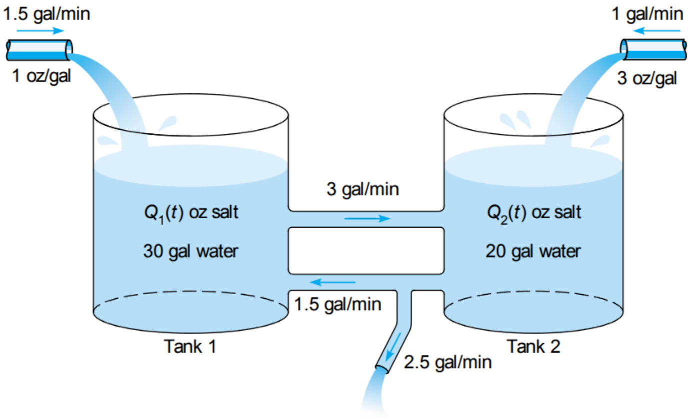
Remember these?

Remember these?



source: Figure 7.1.6 in Boyce & DiPrima

Remember these?

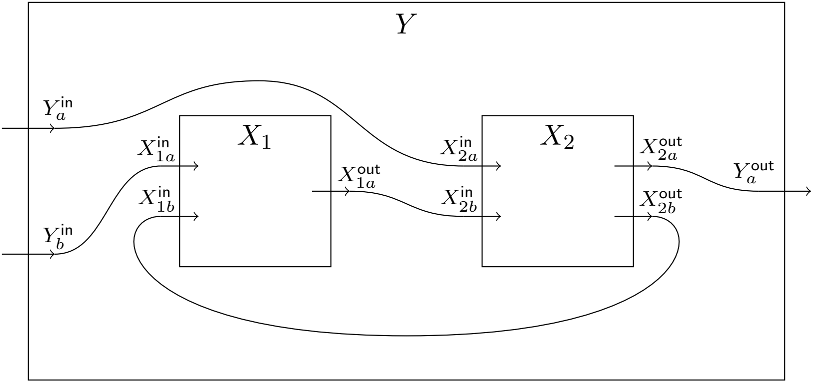


source: Figure 7.1.6 in Boyce & DiPrima

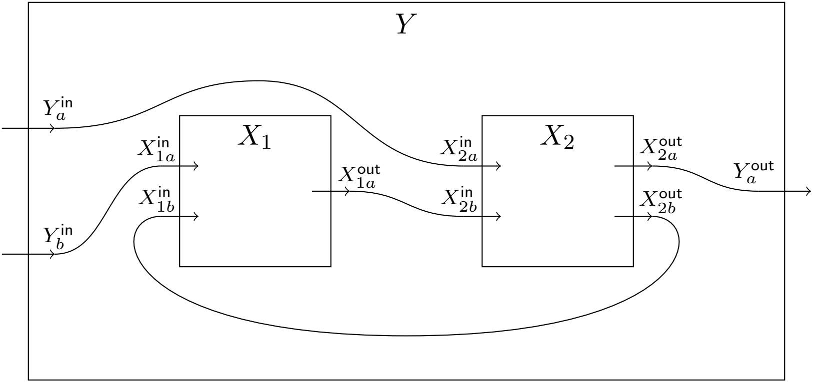
$$\begin{cases} \dot{Q}_1 = -.1Q_1 + .075Q_2 + 1.5 \\ \dot{Q}_2 = .1Q_1 - .2Q_2 + 3 \end{cases} \tag{1}$$

Distill Into Underlying Network

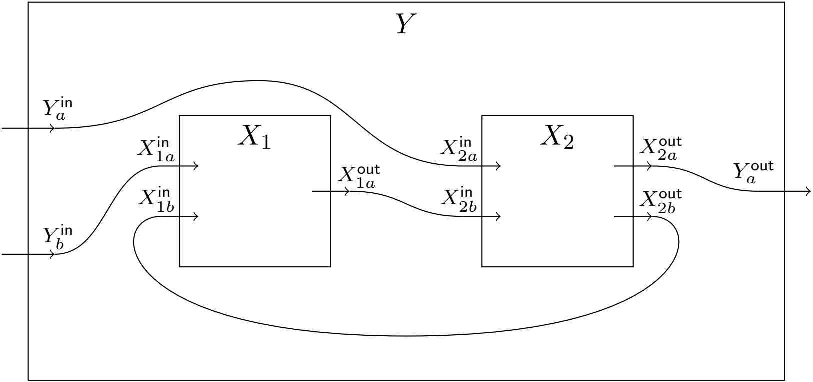
Distill Into Underlying Network



Distill Into Underlying Network (the operad)

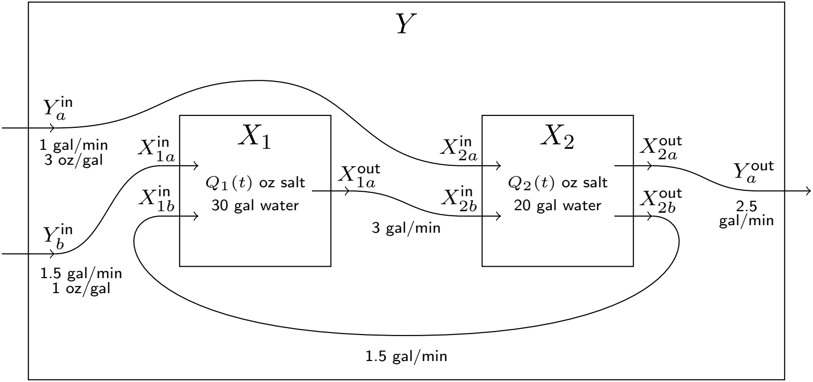


Distill Into Underlying Network (the operad)



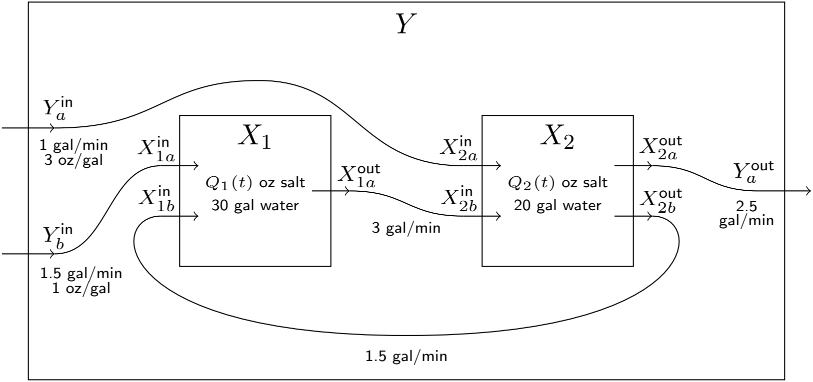
wiring diagram $\Phi : X_1, X_2 \to Y$

Add Content



Y
X_1
X_2
Y_a^{in}
1 gal/min
3 oz/gal
Y_b^{in}
1.5 gal/min
1 oz/gal
X_{1a}^{in}
X_{1b}^{in}
$Q_1(t)$ oz salt
30 gal water
X_{1a}^{out}
3 gal/min
X_{2a}^{in}
X_{2b}^{in}
$Q_2(t)$ oz salt
20 gal water
X_{2a}^{out}
X_{2b}^{out}
Y_a^{out}
2.5
gal/min
1.5 gal/min

Add Content (the algebra)



We will use SMC language

We will use SMC language

Can always recover an operad via **underlying operad** functor

We will use SMC language

Can always recover an operad via **underlying operad** functor

$$\mathcal{O}\colon \mathbf{SMC} \to \mathbf{Opd}.$$

We will use SMC language

Can always recover an operad via **underlying operad** functor

$$\mathcal{O}\colon \mathbf{SMC} \to \mathbf{Opd}.$$

Given $(\mathcal{M}, \otimes, I)$, its underlying operad $\mathcal{O}\mathcal{M}$ is defined by

We will use SMC language

Can always recover an operad via **underlying operad** functor

$$\mathcal{O}\colon \mathbf{SMC} \to \mathbf{Opd}.$$

Given $(\mathcal{M}, \otimes, I)$, its underlying operad $\mathcal{O}\mathcal{M}$ is defined by

$$\operatorname{Ob}\mathcal{O}\mathcal{M} := \operatorname{Ob}\mathcal{M}$$

We will use SMC language

Can always recover an operad via **underlying operad** functor

$$\mathcal{O}\colon \mathbf{SMC} \to \mathbf{Opd}.$$

Given $(\mathcal{M}, \otimes, I)$, its underlying operad $\mathcal{OM}$ is defined by

$$\operatorname{Ob}\mathcal{OM} \coloneqq \operatorname{Ob}\mathcal{M}$$

$$\hom_{\mathcal{OM}}(X_1, \ldots, X_n; Y) \coloneqq \hom_{\mathcal{M}}(X_1 \otimes \cdots \otimes X_n, Y)$$

We will use SMC language

Can always recover an operad via **underlying operad** functor

$$\mathcal{O}\colon \mathbf{SMC} \to \mathbf{Opd}.$$

Given $(\mathcal{M}, \otimes, I)$, its underlying operad $\mathcal{O}\mathcal{M}$ is defined by

$$\operatorname{Ob} \mathcal{O}\mathcal{M} \coloneqq \operatorname{Ob} \mathcal{M}$$

$$\hom_{\mathcal{O}\mathcal{M}}(X_1, \ldots, X_n; Y) \coloneqq \hom_{\mathcal{M}}(X_1 \otimes \cdots \otimes X_n, Y)$$

This allows us to avoid distracting subscripts.

The Wiring Diagram Category $\mathbb{W}$

The Wiring Diagram Category $\mathbf{W}$

Objects are ordered pairs $X = (X^{\mathsf{in}}, X^{\mathsf{out}})$ of *typed finite sets*

The Wiring Diagram Category $\mathbb{W}$

Objects are ordered pairs $X = (X^{\text{in}}, X^{\text{out}})$ of *typed finite sets*

A $\mathcal{C}$-typed finite set is just a map $A \to \mathrm{Ob}\,\mathcal{C}$ for finite set A

The Wiring Diagram Category $\mathbb{W}$

Objects are ordered pairs $X = (X^{\text{in}}, X^{\text{out}})$ of *typed finite sets*

A $\mathcal{C}$-typed finite set is just a map $A \to \mathrm{Ob}\,\mathcal{C}$ for finite set A

Since we're talking about ODE's, we'll let $\mathcal{C}$ be smooth manifolds

The Wiring Diagram Category $\mathbb{W}$

Objects are ordered pairs $X = (X^{\mathsf{in}}, X^{\mathsf{out}})$ of *typed finite sets*

A $\mathcal{C}$-typed finite set is just a map $A \to \mathrm{Ob}\,\mathcal{C}$ for finite set A

Since we're talking about ODE's, we'll let $\mathcal{C}$ be smooth manifolds

We call objects *boxes* and depict them as such

The Wiring Diagram Category $\mathbf{W}$

Objects are ordered pairs $X = (X^{\text{in}}, X^{\text{out}})$ of *typed finite sets*

A $\mathcal{C}$-typed finite set is just a map $A \to \mathrm{Ob}\,\mathcal{C}$ for finite set A

Since we're talking about ODE's, we'll let $\mathcal{C}$ be smooth manifolds

We call objects *boxes* and depict them as such

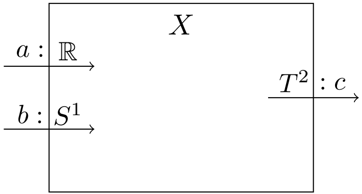


Figure: A box $X = (\{a, b\}, \{c\})$ with two input ports, $a : \mathbb{R}$ and b:S^1, and one output port, $c : T^2$.

The Wiring Diagram Category $\mathbb{W}$

The Wiring Diagram Category $\mathbb{W}$

Morphisms $\Phi : X \to Y$ are typed bijections

The Wiring Diagram Category $\mathbf{W}$

Morphisms $\Phi : X \to Y$ are typed bijections

$$\varphi : X^{\text{in}} + Y^{\text{out}} \to X^{\text{out}} + Y^{\text{in}}$$

The Wiring Diagram Category $\mathbb{W}$

Morphisms $\Phi : X \to Y$ are typed bijections

$$\varphi : X^{\mathsf{in}} + Y^{\mathsf{out}} \to X^{\mathsf{out}} + Y^{\mathsf{in}}$$

We call these *wiring diagrams* and depict them as such

The Wiring Diagram Category $\mathbf{W}$

Morphisms $\Phi : X \to Y$ are typed bijections

$$\varphi : X^{\text{in}} + Y^{\text{out}} \to X^{\text{out}} + Y^{\text{in}}$$

We call these *wiring diagrams* and depict them as such

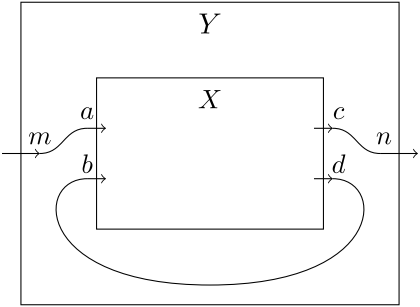


Figure: A Wiring Diagram $\Phi : X \to Y$.

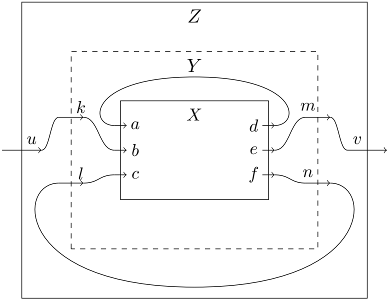
The Wiring Diagram Category $\mathbb{W}$

The Wiring Diagram Category $\mathbf{W}$

Composition defined via nesting

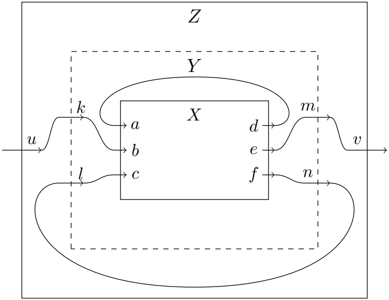
The Wiring Diagram Category $\mathbb{W}$

Composition defined via nesting



The Wiring Diagram Category $\mathbb{W}$

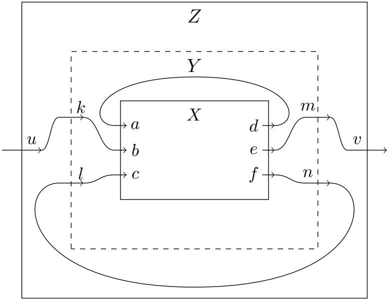
Composition defined via nesting



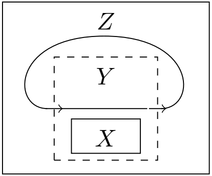
enforce **no passing wires** condition: $\varphi(Y^{\mathsf{out}}) \cap Y^{\mathsf{in}} = \varnothing$, to avoid

The Wiring Diagram Category $\mathbb{W}$

Composition defined via nesting



enforce **no passing wires** condition: $\varphi(Y^{\mathsf{out}}) \cap Y^{\mathsf{in}} = \varnothing$, to avoid



The Wiring Diagram Category $\mathbb{W}$

The Wiring Diagram Category $\mathbf{W}$

Tensor of boxes X, Y defined by

The Wiring Diagram Category $\mathbf{W}$

Tensor of boxes X, Y defined by

$$X \oplus Y = (X^{\mathsf{in}} + Y^{\mathsf{in}}, X^{\mathsf{out}} + Y^{\mathsf{out}})$$

The Wiring Diagram Category $\mathbf{W}$

Tensor of boxes X, Y defined by

$$X \oplus Y = (X^{\mathsf{in}} + Y^{\mathsf{in}}, X^{\mathsf{out}} + Y^{\mathsf{out}})$$

depict by stacking

The Wiring Diagram Category $\mathbb{W}$

Tensor of boxes X, Y defined by

$$X \oplus Y = (X^{\mathsf{in}} + Y^{\mathsf{in}}, X^{\mathsf{out}} + Y^{\mathsf{out}})$$

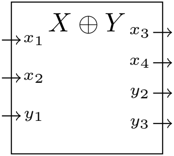
depict by stacking

The Wiring Diagram Category W

Tensor of boxes X, Y defined by

$$X \oplus Y = (X^{\mathsf{in}} + Y^{\mathsf{in}}, X^{\mathsf{out}} + Y^{\mathsf{out}})$$

depict by stacking

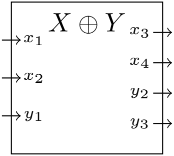


The Wiring Diagram Category $\mathbb{W}$

Tensor of boxes X, Y defined by

$$X \oplus Y = (X^{\mathsf{in}} + Y^{\mathsf{in}}, X^{\mathsf{out}} + Y^{\mathsf{out}})$$

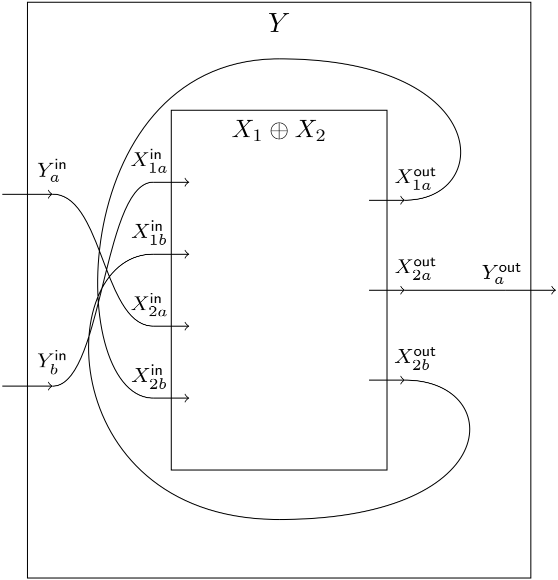
depict by stacking



Monoidal unit, denoted '0', is the closed (without wires) box.

We still use operadic pictures though...

We still use operadic pictures though...



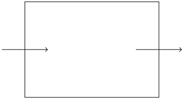
The Algebra of Open Systems

The Algebra of Open Systems

We want to assign an *open dynamical system* to a box

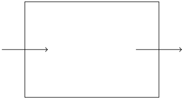
The Algebra of Open Systems

We want to assign an *open dynamical system* to a box



The Algebra of Open Systems

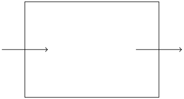
We want to assign an *open dynamical system* to a box



This is an ODE, given input, for the box's state Q and its output

The Algebra of Open Systems

We want to assign an *open dynamical system* to a box

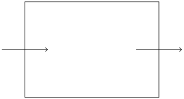


This is an ODE, given input, for the box's state Q and its output

$$\begin{cases} \dot{Q} = f^{\mathsf{in}}(Q, input) \\ output = f^{\mathsf{out}}(Q) \end{cases} \tag{2}$$

The Algebra of Open Systems

We want to assign an *open dynamical system* to a box



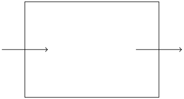
This is an ODE, given input, for the box's state Q and its output

$$\begin{cases} \dot{Q} = f^{\mathsf{in}}(Q, input) \\ output = f^{\mathsf{out}}(Q) \end{cases} \tag{2}$$

Equivalently, if $Q \in M$, a pair of smooth maps

The Algebra of Open Systems

We want to assign an *open dynamical system* to a box



This is an ODE, given input, for the box's state Q and its output

$$\begin{cases} \dot{Q} = f^{\mathsf{in}}(Q, input) \\ output = f^{\mathsf{out}}(Q) \end{cases} \tag{2}$$

Equivalently, if $Q \in M$, a pair of smooth maps

$$\begin{cases} f^{\mathsf{in}} \colon M \times U^{\mathsf{in}} \to TM \\ f^{\mathsf{out}} \colon M \to U^{\mathsf{out}} \end{cases}$$

The Algebra of Open Systems

The Algebra of Open Systems

Define an algebra, i.e. lax functor $\mathcal{G} : (\mathbf{W}, \oplus, 0) \to (\mathbf{Set}, \times, 1)$

The Algebra of Open Systems

Define an algebra, i.e. lax functor $\mathcal{G} : (\mathbf{W}, \oplus, 0) \to (\mathbf{Set}, \times, 1)$

Assign to a box X the set of pairs (S, f), where

The Algebra of Open Systems

Define an algebra, i.e. lax functor $\mathcal{G} : (\mathbf{W}, \oplus, 0) \to (\mathbf{Set}, \times, 1)$

Assign to a box X the set of pairs (S, f), where

- a typed finite set S

The Algebra of Open Systems

Define an algebra, i.e. lax functor $\mathcal{G} : (\mathbf{W}, \oplus, 0) \to (\mathbf{Set}, \times, 1)$

Assign to a box X the set of pairs (S, f), where

- a typed finite set S
- an open dynamical system f given by

$$\begin{cases} f^{\mathsf{in}} \colon \overline{S} \times \overline{X^{\mathsf{in}}} \to T\overline{S} \\ f^{\mathsf{out}} \colon \overline{S} \to \overline{X^{\mathsf{out}}} \end{cases}$$

The Algebra of Open Systems

Define an algebra, i.e. lax functor $\mathcal{G} : (\mathbf{W}, \oplus, 0) \to (\mathbf{Set}, \times, 1)$

Assign to a box X the set of pairs (S, f), where

- a typed finite set S
- an open dynamical system f given by

$$\begin{cases} f^{\mathsf{in}} \colon \overline{S} \times \overline{X^{\mathsf{in}}} \to T\overline{S} \\ f^{\mathsf{out}} \colon \overline{S} \to \overline{X^{\mathsf{out}}} \end{cases}$$

where $\overline{\cdot}$ is the *dependent product* of a TFS $X : A \to \operatorname{Ob} \mathcal{C}$

The Algebra of Open Systems

Define an algebra, i.e. lax functor $\mathcal{G} : (\mathbf{W}, \oplus, 0) \to (\mathbf{Set}, \times, 1)$

Assign to a box X the set of pairs (S, f), where

- a typed finite set S
- an open dynamical system f given by

$$\begin{cases} f^{\text{in}} \colon \overline{S} \times \overline{X^{\text{in}}} \to T\overline{S} \\ f^{\text{out}} \colon \overline{S} \to \overline{X^{\text{out}}} \end{cases}$$

where $\overline{\cdot}$ is the *dependent product* of a TFS $X : A \to \mathrm{Ob}\,\mathcal{C}$

$$\overline{X} = \prod_{a \in A} X(a)$$

The Sub-Algebra of Open Linear Systems

The Sub-Algebra of Open Linear Systems

Lax functor $\mathcal{L} : (\mathbf{W}, \oplus, 0) \to (\mathbf{Set}, \times, 1)$

The Sub-Algebra of Open Linear Systems

Lax functor $\mathcal{L} : (\mathbf{W}, \oplus, 0) \to (\mathbf{Set}, \times, 1)$

Now our sets are typed in Euclidean Spaces, so we have

The Sub-Algebra of Open Linear Systems

Lax functor $\mathcal{L} : (\mathbf{W}, \oplus, 0) \to (\mathbf{Set}, \times, 1)$

Now our sets are typed in Euclidean Spaces, so we have

- $TM \cong M$

The Sub-Algebra of Open Linear Systems

Lax functor $\mathcal{L} : (\mathbf{W}, \oplus, 0) \to (\mathbf{Set}, \times, 1)$

Now our sets are typed in Euclidean Spaces, so we have

- $TM \cong M$
- a matrix form of f!

The Sub-Algebra of Open Linear Systems

Lax functor $\mathcal{L} : (\mathbf{W}, \oplus, 0) \to (\mathbf{Set}, \times, 1)$

Now our sets are typed in Euclidean Spaces, so we have

- $TM \cong M$
- a matrix form of f!

Our open systems now look like

The Sub-Algebra of Open Linear Systems

Lax functor $\mathcal{L} : (\mathbf{W}, \oplus, 0) \to (\mathbf{Set}, \times, 1)$

Now our sets are typed in Euclidean Spaces, so we have

- $TM \cong M$
- a matrix form of f!

Our open systems now look like

$$\begin{bmatrix} \dot{m} \\ u^{\mathsf{out}} \end{bmatrix} = \begin{bmatrix} f^{M,M} & f^{M,U} \\ f^{U,M} & 0 \end{bmatrix} \begin{bmatrix} m \\ u^{\mathsf{in}} \end{bmatrix}$$

The Sub-Algebra of Open Linear Systems

Lax functor $\mathcal{L} : (\mathbf{W}, \oplus, 0) \to (\mathbf{Set}, \times, 1)$

Now our sets are typed in Euclidean Spaces, so we have

- $TM \cong M$
- a matrix form of f!

Our open systems now look like

$$\begin{bmatrix} \dot{m} \\ u^{\text{out}} \end{bmatrix} = \begin{bmatrix} f^{M,M} & f^{M,U} \\ f^{U,M} & 0 \end{bmatrix} \begin{bmatrix} m \\ u^{\text{in}} \end{bmatrix}$$

given wiring diagram φ, transport linear system by

The Sub-Algebra of Open Linear Systems

Lax functor $\mathcal{L} : (\mathbf{W}, \oplus, 0) \to (\mathbf{Set}, \times, 1)$

Now our sets are typed in Euclidean Spaces, so we have

- $TM \cong M$
- a matrix form of f!

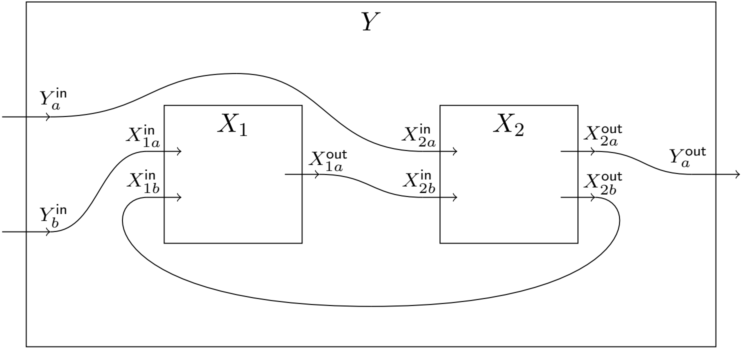
Our open systems now look like

$$\begin{bmatrix} \dot{m} \\ u^{\text{out}} \end{bmatrix} = \begin{bmatrix} f^{M,M} & f^{M,U} \\ f^{U,M} & 0 \end{bmatrix} \begin{bmatrix} m \\ u^{\text{in}} \end{bmatrix}$$

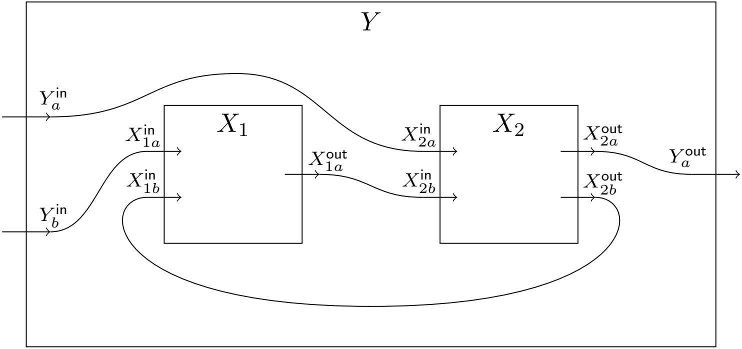
given wiring diagram φ, transport linear system by

$$g = \begin{bmatrix} g^{S,S} & g^{S,X} \\ g^{X,S} & g^{X,X} \end{bmatrix} = \begin{bmatrix} f^{S,X} & 0 \\ 0 & I \end{bmatrix} \overline{\varphi} \begin{bmatrix} f^{X,S} & 0 \\ 0 & I \end{bmatrix} + \begin{bmatrix} f^{S,S} & 0 \\ 0 & 0 \end{bmatrix}$$

back to our example

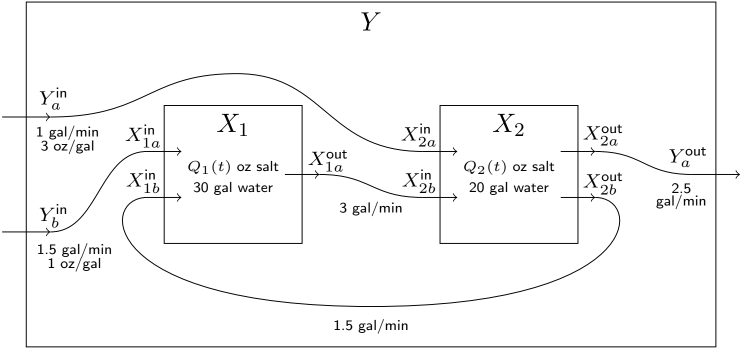


back to our example

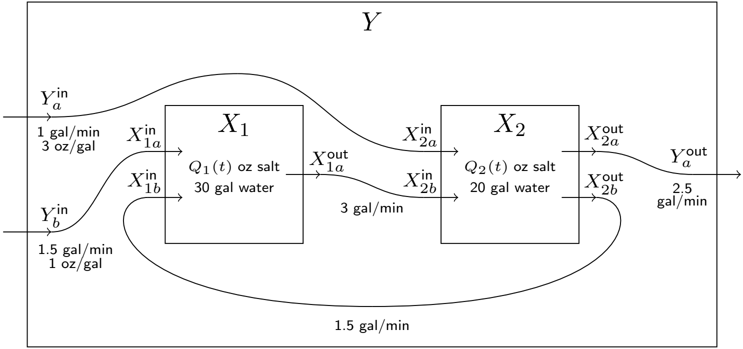


$\overline{\varphi}$ given by
$$\begin{bmatrix} \overline{X_{1a}^{\mathrm{out}}} \\ \overline{X_{2a}^{\mathrm{out}}} \\ \overline{X_{2b}^{\mathrm{out}}} \\ \overline{Y_a^{\mathrm{in}}} \\ \overline{Y_b^{\mathrm{in}}} \end{bmatrix} = \begin{bmatrix} 0 & 0 & I & 0 & 0 \\ 0 & 0 & 0 & 0 & I \\ 0 & I & 0 & 0 & 0 \\ I & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & I & 0 \end{bmatrix} \begin{bmatrix} \overline{X_{1a}^{\mathrm{in}}} \\ \overline{X_{1b}^{\mathrm{in}}} \\ \overline{X_{2a}^{\mathrm{in}}} \\ \overline{X_{2b}^{\mathrm{in}}} \\ \overline{Y_a^{\mathrm{out}}} \end{bmatrix}$$

back to our example



back to our example



ODS for $X_1 \oplus X_2$ is
$$\begin{bmatrix} \dot{Q}_1 \\ \dot{Q}_2 \\ X_{1a}^{\text{out}} \\ X_{2a}^{\text{out}} \\ X_{2b}^{\text{out}} \end{bmatrix} = \begin{bmatrix} -.1 & 0 & 1 & 1 & 0 & 0 \\ 0 & -.2 & 0 & 0 & 1 & 1 \\ .1 & 0 & 0 & 0 & 0 & 0 \\ 0 & .125 & 0 & 0 & 0 & 0 \\ 0 & .075 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} Q_1 \\ Q_2 \\ X_{1a}^{\text{in}} \\ X_{1b}^{\text{in}} \\ X_{2a}^{\text{in}} \\ X_{2ba}^{\text{in}} \end{bmatrix}$$

back to our example

back to our example

Transporting this across the wiring diagram yields

back to our example

Transporting this across the wiring diagram yields

$$\begin{bmatrix} \dot{Q}_1 \\ \dot{Q}_2 \\ Y^{\mathsf{out}} \end{bmatrix} = \begin{bmatrix} -.1 & .075 & 0 & 1 \\ .1 & -.2 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} Q_1 \\ Q_2 \\ Y_a^{\mathsf{in}} \\ Y_b^{\mathsf{in}} \end{bmatrix}$$

back to our example

Transporting this across the wiring diagram yields

$$\begin{bmatrix} \dot{Q}_1 \\ \dot{Q}_2 \\ Y^{\mathsf{out}} \end{bmatrix} = \begin{bmatrix} -.1 & .075 & 0 & 1 \\ .1 & -.2 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} Q_1 \\ Q_2 \\ Y_a^{\mathsf{in}} \\ Y_b^{\mathsf{in}} \end{bmatrix}$$

when forgetting outputs and substituting inputs, yields

back to our example

Transporting this across the wiring diagram yields

$$\begin{bmatrix} \dot{Q}_1 \\ \dot{Q}_2 \\ Y^{\text{out}} \end{bmatrix} = \begin{bmatrix} -.1 & .075 & 0 & 1 \\ .1 & -.2 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} Q_1 \\ Q_2 \\ Y_a^{\text{in}} \\ Y_b^{\text{in}} \end{bmatrix}$$

when forgetting outputs and substituting inputs, yields

$$\begin{cases} \dot{Q}_1 = -.1Q_1 + .075Q_2 + 1.5 \\ \dot{Q}_2 = .1Q_1 - .2Q_2 + 3 \end{cases} \tag{3}$$

back to our example

Transporting this across the wiring diagram yields

$$\begin{bmatrix} \dot{Q}_1 \\ \dot{Q}_2 \\ Y^{\text{out}} \end{bmatrix} = \begin{bmatrix} -.1 & .075 & 0 & 1 \\ .1 & -.2 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} Q_1 \\ Q_2 \\ Y_a^{\text{in}} \\ Y_b^{\text{in}} \end{bmatrix}$$

when forgetting outputs and substituting inputs, yields

$$\begin{cases} \dot{Q}_1 = -.1Q_1 + .075Q_2 + 1.5 \\ \dot{Q}_2 = .1Q_1 - .2Q_2 + 3 \end{cases} \tag{3}$$

this is what we wanted!