UNIVERSITY OF CALIFORNIA
RIVERSIDE

Composing Behaviors of Networks

A Dissertation submitted in partial satisfaction
of the requirements for the degree of

Doctor of Philosophy

in

Mathematics

by

Jade E. Master

June 2021

Dissertation Committee:

Dr. John C. Baez, Chairperson
Dr. Wee Liang Gan
Dr. José González

The Dissertation of Jade E. Master is approved:

Committee Chairperson

University of California, Riverside

Acknowledgments

I wouldn't have made it this far or even been the person I am today without the support of my friends and family. Allison I love you and feel so lucky to have you in my life. Thank you to my parents Eliza and Steve. I love you both so much thank you for everything. Thank you to my grandmother Babette. You have always been my biggest supporter and you inspire me to live my best life. Thank you to my friends for putting up with me. Thank you to Joyo, Mia, Alice, Casey, Maria, Ellie, and everyone on the server. I love you all. Thank you to everyone I know on the internet, I hope we meet in person someday. Thank you to all my haters, you pushed me to be a better person.

Thank you to my advisor John Baez, I've learned so much from you and would not be where I am now without your dedication and hard work. Thank you to Christian Williams and Joe Moeller for sharing camaraderie and inspiration. Thank you to Daniel Cicala, Kenny Courser, Christina Vasilokopolou and Brandon Coya. Thank you to Anastasios Stefanou, Todd Trimble, Joshua Meyers, Sarah Rovner-Frydman, Zans Mihejez, Fabrizio Genovese, Morgan Rogers, Jelle Herold, Valeria De Paiva, Sarah Griffith, Alex Pokorny, Rany Tith, Evan Patterson, Arquimedes Canedo, Mike Shulman, Jules Hedges, and Martha Lewis. Math is created by communities not individuals and this thesis is no exception.

Thank you to all the people who produced the things I need to survive. Thank you to the Tongva tribe, this work was done on their land.

The material from Chapter 3 consists of work from "Petri nets based on Lawvere theories" [Mas20d]. The material from Chapter 4 generalizes the work of "Open Petri nets"

joint with John Baez [BM20]. Chapter 5 and Chapter 6 consist of work from "The open algebraic path problem" [Mas20c].

To Allison, grow your wings and fly.

ABSTRACT OF THE DISSERTATION

Composing Behaviors of Networks

by

Jade E. Master

Doctor of Philosophy, Graduate Program in Mathematics
University of California, Riverside, June 2021
Dr. John C. Baez, Chairperson

This thesis aims to develop a compositional theory for the operational semantics of networks. The networks considered are described by either internal or enriched graphs. In the internal case we focus on Q-nets, a generalization of Petri nets based on a Lawvere theory Q. Q-nets include many known variants of Petri nets including pre-nets, integer nets, elementary net systems, and bounded nets. In the enriched case we focus on graphs enriched in a quantale R regarded as matrices with entries in R. These R-matrices represent distance networks, Markov processes, capacity networks, non-deterministic finite automata, simple graphs, and more. The operational semantics of Q-nets is constructed as an adjunction between Q-nets and categories internal to the category of models of Q. The left adjoint of this adjunction sends a Q-net P to an internal category $F_{\mathsf{Q}}(P)$ whose morphisms represent all possible firing sequences in P. Similarly, the operational semantics of R-matrices is constructed as an adjunction between R-matrices and categories enriched in R. The left adjoint of this adjunction sends an R-matrix M to the R-category $F_R(M)$ whose hom-objects are solutions of the algebraic path problem: a generalization of the shortest path problem to graphs

weighted in R. For both Q-nets and R-matrices we use the theory of structured cospans to study the compositionality of the above operational semantics. For each type of network we construct a double category whose morphisms are “open networks”, i.e. networks with certain vertices designated as input or output. The operational semantics gives a double functor from a double category of open networks to a double category of open enriched or internal categories. These double functors give a compositional framework for computing the operational semantics of Q-nets and R-matrices: their functoriality and coherence give relationships between the operational semantics of a network and the operational semantics of the smaller networks from which it is composed. We introduce the black-boxing of an open network, a profunctor describing the externally observable behavior of an open network. We introduce a class of open networks called “functional open networks” for which black-boxing preserves composition.

Contents

Chapter 1

Introduction

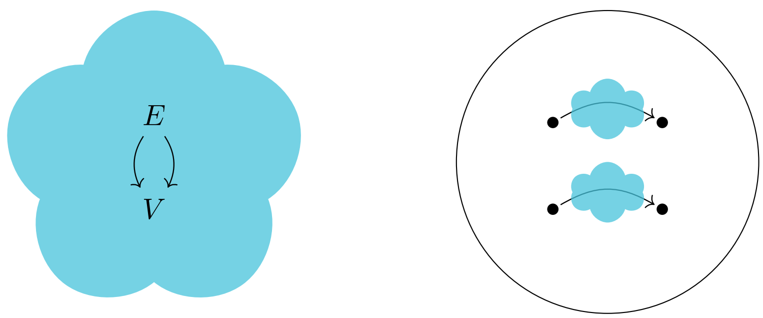
> [...] the whole itself may be viewed as a conceptual construction, hence the question of the ontological status of boundaries becomes of a piece with the more general issue of the conventional status of ordinary objects and events. Cfr. Goodman: “We make a star as we make a constellation, by putting its parts together and marking off its boundaries” (1980: 213) — [Var15]

Underlying much of scientific thought is the assumption that things can be understood by understanding the way their components join together to make the whole. We often find when applying this point of view that the old adage rings true: “things are more than the sum of their parts”. When parts are joined together to form a whole, behavior emerges between the interaction of the parts that was not present before. Therefore, the magic must be in the way that the components are glued together.

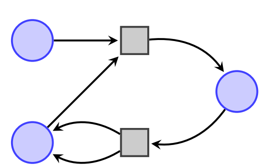
This thesis aims to provide a general setting to study the emergence that occurs when networks are glued together from their components. Here “network” refers to a structure with a discrete set of states and a discrete set of relationships between them. In this thesis, the set of states may be equipped with operations representing the different ways that states are allowed to join together to make new states. The relationships in this thesis may be distances, probabilities, connections, processes, or external input.

Internal and Enriched Graphs

The different types of networks considered in this thesis are unified using enriched and internal graphs. All networks considered in this thesis are either graphs internal to a category C or graphs enriched in a poset R.

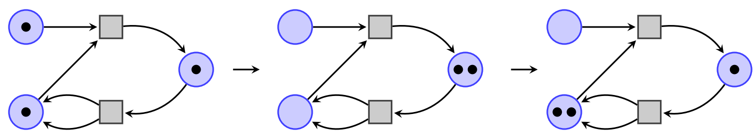


As shown above left, a graph internal to C is entirely in the clouds. In other words, its edges, vertices, source, and target maps are all objects and morphisms of C. An internal graph is equipped with the structure and properties of the category it lives in. If C is the category of commutative monoids, then graphs internal to C are presented by Petri nets. A Petri net is a diagram like this:



The circles, called "places", represent different kinds of resources and the squares, called "transitions", represent processes which take different resources as input and output. A discrete portion of a resource is represented by a "token": a black dot inhabiting a place.

A transition of a Petri net may "fire" if there are enough tokens in the places with arrows going into it. When a transition fires, it removes one token from a place for each arrow going into it from that place. The transition then deposits one token into a place for each arrow coming from the transition. The following diagram represent the firing of the top transition of the above Petri net followed by a firing of the bottom transition:

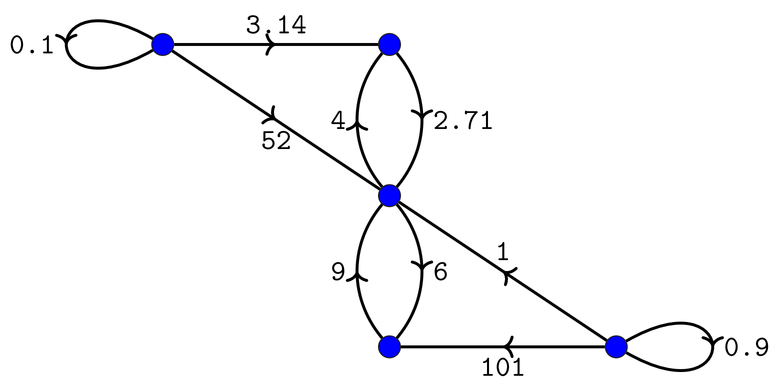


More generally, graphs internal to $\mathsf{Mod}(\mathsf{Q})$, the category of models for a Lawvere theory Q, are presented by Q-nets. We obtain many variants of Petri nets including pre-nets, elementary net systems, k-safe nets, lending nets, and more by generalizing in this way.

On the other hand, a graph enriched in a poset R has only its edges in the clouds: its vertices are elements of sets, but for each pair of vertices x and y, there is an element $R(x, y)$ of R representing the connection between X and Y. In other words, an R-enriched graph on a set X is a square matrix

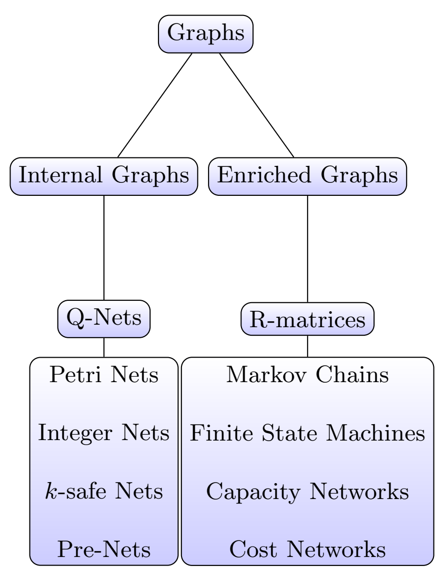
$$M \colon X \times X \to R$$

called an R-matrix. R should be nice enough to enrich in, and it is enough for R to be a commutative "quantale": a monoidal closed poset with all joins. R-matrices are represented graphically as weighted graphs. When R is the quantale $[0, \infty]$, of positive real numbers, the values $R(x, y)$ are regarded as distances between a set of locations. A $[0, \infty]$-matrix can be drawn as a graph where the distance $R(x, y)$ labels an edge from x from y:



where pairs of vertices without an edge between them are assumed to have a distance of ∞. Varying R gives Markov processes, finite state machines, simple graphs, capacity networks, and more as instances of enriched graphs.

The following tree summarizes the networks considered in this thesis. This tree is certainly non-exhaustive as more examples may be derived from the general theory developed here.



Operational Semantics of Networks

Networks are just formal structures until they are equipped with a semantics indicating their real world meaning. The semantics that we equip networks with is the "operational semantics": a mathematical specification of the ways the states of the network may evolve in time. For an ordinary graph G, its operational semantics will be a category $F(G)$ whose objects are the vertices of G and whose morphisms consist of all paths which can be formed from the edges of G. F becomes a functor so that morphisms of graphs extend to behavior preserving functors between their operational semantics. F is the left adjoint of the adjunction

$$\mathsf{Grph} \underset{U}{\overset{F}{\rightleftarrows}} \mathsf{Cat} \quad (\bot)$$

where U is the functor which sends any category to its underlying graph. This perspective allows for a uniform treatment of the operational semantics for internal and enriched graphs. In Section 5.2 we show how graphs internal to $\mathsf{Mod(Q)}$ generate categories internal to $\mathsf{Mod(Q)}$ via the adjunction

$$\mathsf{Q\text{-}Net} \underset{\mathrm{B}^{\bullet}}{\overset{{}^{\bullet}\mathrm{B}}{\rightleftarrows}} \mathsf{Grph(Mod(Q))} \quad (\bot)$$

where $\mathsf{Grph(Mod(Q)}$ is the category of graphs internal to $\mathsf{Mod(Q)}$ and $\mathsf{Cat(Mod(Q))}$ is the category of small categories internal to $\mathsf{Mod(Q)}$. This does not yet give the operational semantics for Q-nets. In Section 3.5 we construct the missing piece, an adjunction

$$\mathsf{Q\text{-}Net} \underset{\mathrm{A}^{\bullet}}{\overset{{}^{\bullet}\mathrm{A}}{\rightleftarrows}} \mathsf{Grph(Mod(Q))} \quad (\bot)$$

making precise the sense in which Q-nets are the generating data for graphs internal to $\mathsf{Mod}(\mathsf{Q})$. In Theorem 45 we combine these to obtain the operational semantics for Q-nets

$$\mathsf{Q}\text{-}\mathsf{Net} \underset{U_{\mathsf{Q}}}{\overset{F_{\mathsf{Q}}}{\rightleftarrows}} \mathsf{Q}\text{-}\mathsf{Cat} \quad (F_{\mathsf{Q}} \dashv U_{\mathsf{Q}})$$

In the case when Q is the Lawvere theory for commutative monoids, this operational semantics is similar to the adjunction developed by Meseguer and Montanari [MM90]. Letting Q be the theory of monoids reproduces the operational semantics for pre-nets introduced in [BMMS01], and setting Q equal to the theory of abelian groups gives the operational semantics for lending nets developed by Genovese and Herold [GH18]. Allowing Q to be other Lawvere theories gives a new categorical characterizations of the operational semantics for many other variants of Petri nets.

In Chapter 5, for each commutative quantale R, we construct an adjunction

$$\mathsf{RMat} \underset{U_R}{\overset{F_R}{\rightleftarrows}} \mathsf{RCat} \quad (F_R \dashv U_R)$$

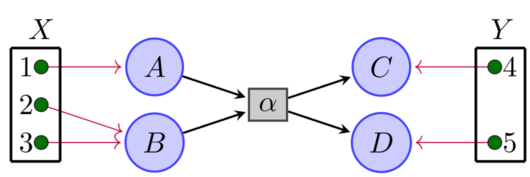
between matrices valued in R and categories enriched in R. The left adjoint of this adjunction is familar: for an R-matrix M, the R-category $F_R(M)$ is a matrix whose entries are solutions to the algebraic path problem. The algebraic path problem is a generalization of the shortest path problem to probability, computing, matrix multiplication, and optimization [Tar81, Foo15]. When R is the quantale of positive real numbers $([0, \infty], \min, +)$, a weighted graph can be regarded as an R-matrix, and the shortest paths of this graph are given by $F_R(M)$. The algebraic path problem allows R to vary, and gets problems of a similar flavor also as the free R-category on an R-matrix. Many popular shortest path algorithms can be extended to compute solutions to the algebraic path problem in a general setting [HM12]. The algebraic path problem can also be implemented generically using functional programming [Dol13]. The above adjunction makes clear the universal property of the solutions to the algebraic path problem.

Open Networks

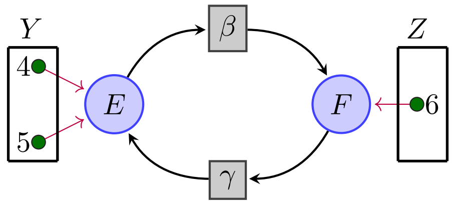
The main goal of this thesis is to study how the operational semantics of networks can be joined together. To compose networks we first need to equip them with boundaries. A network G with vertex set V is made "open" to its surroundings by equipping it with functions $i\colon X \to V$ and $o\colon Y \to V$ designating input and output vertices respectively. This is formalized as a cospan

$$\begin{array}{ccc} & G & \\ \nearrow & & \nwarrow \\ LX & & LY \end{array}$$

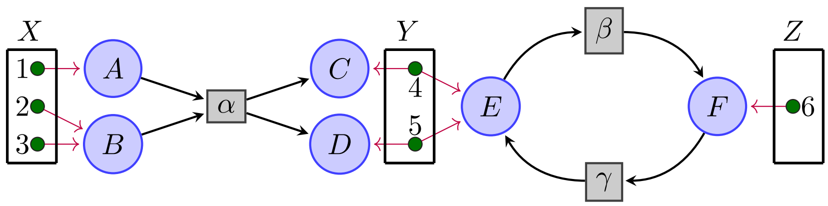
where LX and LY denote the discrete networks on the sets X and Y. Similarly, a Petri net is made open by equipping it with functions from its input and output sets to its places. Open Petri nets will be the running example for the remainder of this introduction. Here is an open Petri net P with input set X and output set Y:



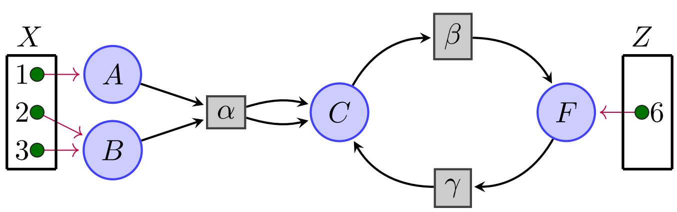
The functions from X and Y into the set of places indicate points at which tokens could flow in or out. We write this open Petri net as $P\colon X \to Y$ for short. There are two fundamental operations on open networks. First, they may be composed along a shared boundary. Given another open Petri net $Q\colon Y \to Z$:



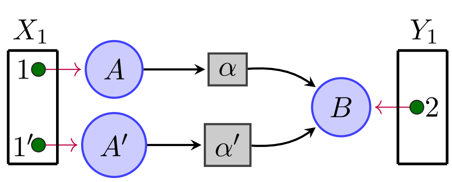
the first step in composing P and Q is to put the pictures together:



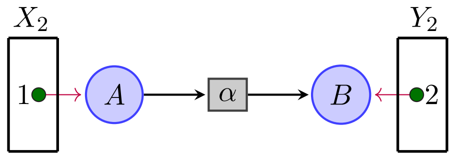
At this point, if we ignore the sets X, Y, Z, we have a new Petri net whose set of places is the disjoint union of those for P and Q. The second step is to identify a place of P with a place of Q whenever both are images of the same point in Y. We can then stop drawing everything involving Y, and get an open Petri net $Q \circ P\colon X \to Z$:



The second fundamental operation of open networks comes from the morphisms between networks. These morphisms represent behavior preserving maps. When extended to open networks, these behavior preserving maps should preserve the inputs and outputs as well. For example, there is a morphism from an open Petri net $G\colon X_1 \to Y_1$:



to an open Petri net $H \colon X_2 \to Y_2$



mapping both primed and unprimed symbols to unprimed ones. More precisely, this morphism of open Petri nets is a commutative diagram

$$\begin{array}{ccccc} LX_1 & \longrightarrow & G & \longleftarrow & LY_1 \\ \downarrow & & \downarrow & & \downarrow \\ LX_2 & \longrightarrow & H & \longleftarrow & LY_2 \end{array}$$

in the category of Petri nets. We denote this morphism with the notation $\beta \colon G \Rightarrow H$. β describes a process of "simplifying" an open Petri net. There are also morphisms that include simple open Petri nets more complicated ones. For example, the above morphism of open Petri nets has a right inverse.

These two operations fit together into the structure of a double category. Double categories were introduced in the 1960s by Ehresmann [Ehr63, Ehr65]. More recently they have been used to study open dynamical systems [Ler18, LS16, Ngo], open electrical circuits and chemical reaction networks [Cou17], open discrete-time Markov chains [CHP17], coarse-graining for open continuous-time Markov chains [BC18], and "tile logic" for concurrency in computer science [BMM02]. Theorem 7 constructs a double category where

- objects are sets X,Y,Z,...
- vertical morphisms are functions $f \colon X \to X'$,
- horizontal morphisms are open networks $G \colon X \to Y$,

- horizontal composition is the composition operation described above, and
- 2-morphisms are the boundary preserving morphisms $\beta\colon G \Rightarrow H$ of open networks described above.

In Theorem 60 we construct a double category $\mathsf{Open}(\mathsf{Q\text{-}Net})$ of the above form whose horizontal morphisms are open Q-nets. In Theorem 91 we construct an analogous double category $\mathsf{Open}(\mathsf{RMat})$ for open R-matrices. Theorem 60 has $\mathsf{Open}(\mathsf{Petri})$, the double category of open Petri nets, as a special case. $\mathsf{Open}(\mathsf{Petri})$ is a double category where objects are sets, vertical morphisms are functions, horizontal morphisms are open Petri nets, and 2-morphisms are morphisms of open Petri nets. The axioms of a double category ensure that morphisms of open networks and composition of open networks are compatible. Besides composing open networks, we can also "tensor" them via disjoint union: this describes networks being run in parallel rather than in series. The result is that the double category described above is upgraded to a symmetric monoidal double category.

Composing Operational Semantics of Networks

The double categories of open networks constructed in this thesis describe a language for gluing smaller open networks into larger ones. The next step is to understand how the operational semantics of these networks can be applied to this language. We may extend the operational semantics functor for Petri nets

$$F\colon \mathsf{Petri} \to \mathsf{CMC}$$

to a symmetric monoidal double functor

$$\mathsf{Open}(F)\colon \mathsf{Open}(\mathsf{Petri}) \to \mathsf{Open}(\mathsf{CMC})$$

where $\mathsf{Open}(\mathsf{CMC})$ is a double category whose horizontal morphisms are "open commutative monoidal categories" $C\colon X \to Y$, i.e. cospans in CMC of the form

$$\mathbb{N}(X) \rightarrow C \leftarrow \mathbb{N}(Y)$$

where $\mathbb{N}(X)$ and $\mathbb{N}(Y)$ are the discrete categories on the free commutative monoids on X and Y. This double functor provides a compositional framework for composing the operational semantics of Petri nets. The key to this double functor is that the functor F preserves pushouts. Suppose a Petri net is decomposed into component open Petri nets

$$LX \rightarrow P \leftarrow LY \rightarrow Q \leftarrow LZ$$

F may either be applied to their composite

$$FLX \rightarrow F(P +_{LY} Q) \leftarrow FLZ$$

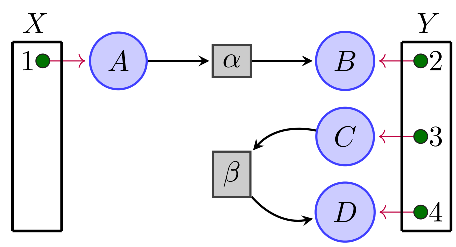
or applied to each component and composed in $\mathsf{Open}(\mathsf{CMC})$

$$FLX \rightarrow F(P) +_{FLY} F(Q) \leftarrow FLZ.$$

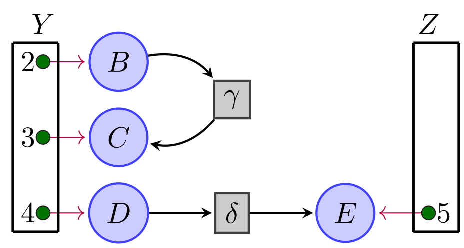
Functoriality of $\mathsf{Open}(F)$ says that these two open commutative monoidal categories must be isomorphic; it therefore provides a compositionality relationship breaking down the operational semantics into smaller pieces. This is not a free lunch, the second pushout is taken in CMC which is constructed in a rather involved way. In general pushouts in CMC may be computed using Kelly's transfinite construction of free algebras [Kel80]. The idea behind this construction is that the pushout first takes the free commutative monoidal category

$$F(UF(P) +_{UFLY} UF(Q))$$

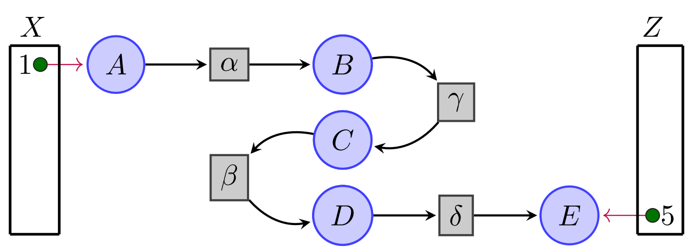
and then quotients away the redundant morphisms. The need for this second application of F is clarified by the following example. Take P to be this open Petri net:



and take Q to be this:



Then their composite, $Q \circ P \colon X \to Z$, looks like this:



This composite contains a morphism starting in A and ending in E which cannot be obtained from a firing sequence starting in P and ending in Q. However, this morphism is contained in the free commutative monoidal category $F(UF(P) +_{UFLY} UF(Q))$ which accounts for all possible feedback loops and zig-zags between P and Q.

Often when studying open networks, we are less concerned with their internal workings than with the relationships they induce between their inputs and outputs. To represent this simplification we introduce the "black-boxing" of an open network. For an open Petri net $P\colon X \to Y$, its black-boxing is a profunctor

$$\blacksquare(P)\colon \mathbb{N}(X) \times \mathbb{N}(Y) \to \mathsf{Set}$$

which sends a pair of markings $(x, y) \in \mathbb{N}(X) \times \mathbb{N}(Y)$ to the set of firing sequences in P which start with x and end with Y. Next we ask about the compositionality of black-boxing, i.e. how the profunctor composite $\blacksquare(Q) \circ \blacksquare(P)$ compares to $\blacksquare(Q \circ P)$. We show that black-boxing gives a lax monoidal double functor

$$\mathsf{Open}(\mathsf{Petri}) \to \mathsf{Prof}$$

where Prof is a double category whose horizontal morphisms are profunctors. This double functor is only lax because the black-boxing $\blacksquare(Q \circ P)$ cannot be entirely reconstructed from the composite $\blacksquare(Q) \circ \blacksquare(P)$. As shown in the above "zig-zag" example, $\blacksquare(Q \circ P)$ may contain morphisms which are not the composite of a morphism in $\blacksquare(P)$ with a morphism in $\blacksquare(Q)$. Next define a class of open networks that do not have this problematic behavior. We introduce "functional open networks" based on functional Petri nets introduced by Zaitsev and Sleptsov [Zai05, ZS97]. These are open networks for which every input is source and every output is a sink. We prove that black-boxing preserves composition on functional open networks. This gives a useful formula for composing operational semantics which can be turned into code as in [Mas20a].

Outline of the thesis

In Chapter 2 we study the operational semantics of graphs and its extension to open graphs. In Section 2.1 we construct the operational semantics of graphs. In Section

2.2 we define the symmetric monoidal double category of open graphs. In Section 2.3 we extend the operational semantics of graphs to a symmetric monoidal double functor from the category of open graphs to the category of open categories. In Section 2.4 we introduce the black-boxing of an open graph and show that it gives a lax double functor into the double category of profunctors. In Theorem 20 we define functional open graphs and show that black-boxing preserves their composition up to isomorphism.

In Chapter 3 we define Q-nets and construct their operational semantics. In Section 3.1 we review some definitions in Petri net theory. In Section 3.2 we define Q-nets and show how many existing variants of Petri nets and their relationships may be derived from this definition. In Section 3.3 we construct an operational semantics adjunction for Petri nets. In Section 3.4 we generalize the previous section to an adjunction between Q-nets and Q-categories. This adjunction is factored into two parts. In Section 3.5 we construct the first part, turning Q-nets into internal graphs, and in Section 3.6 we construct the second part, turning internal graphs into internal categories.

In Chapter 4 we study the compositionality of the operational semantics for Q-nets. In Section 4.1 we define the double category of open Q-nets. In Section 4.2 we extend the operational semantics of Q-nets to a symmetric monoidal double functor from the double category of Q-nets to the double category of Q-categories. In Section 4.3 we define the black-boxing of an open Q-net and show that black-boxing defines a lax double functor from the double category of open Q-categories to a double category of profunctors. In Theorem 68 we show that black-boxing preserves composition of functional open Q-nets up to isomorphism.

In Chapter 5 we construct an operational semantics for matrices valued in a commutative quantale R. In Section 5.1 we review the definition of R-matrices and the algebraic path problem. In Section 5.2 we construct an operational semantics adjunction between the category of R-matrices and the category of categories enriched in R. The left adjoint of this adjunction gives solutions to the algebraic path problem.

In Chapter 6, we explore how solutions of the algebraic path problem behave on open R-matrices. In Section 6.1 we define the symmetric monoidal double category of open R-matrices. In Section 6.2 we show how finding solutions to the algebraic path problem gives a symmetric monoidal double functor from the double category of open R-matrices to the double category of open R-categories. In Section 6.3 we define the black-boxing of an open R-matrix and show that it gives rise to a lax double functor. In Definition 96 we define functional open R-matrices and in Theorem 98 we show that black-boxing preserves their composition strictly.

Lastly, in Appendix A we review the relevant definitions in the theory of double categories and in Appendix B we review Lawvere theories.

Chapter 2

Compositionality of Graphs

This chapter serves as a blueprint for Chapters 3, 4, 5, and 6.2 by outlining the main results of this thesis in the case of ordinary graphs. In particular, in Section 2.1, we construct a well-known operational semantics of graphs in a way that lends itself to generalization to enriched and internal graphs. In Section 2.2, we define "open graphs", i.e. graphs equipped with input and output boundaries, and show that there is a symmetric monoidal double category $\mathsf{Open}(\mathsf{Grph})$ whose horizontal morphisms are open graphs. In Section 2.3 we show how the operational semantics of graphs can be extended a compositional setting, i.e. lifted to a double functor from a double category of open graphs to a double category of open categories. In Section 2.4 we introduce the "black-boxing" of an open graph. The black-boxing of an open graph is a profunctor which records the operational semantics of the open graph when restricted to the input and output boundaries. In Theorem 16 we prove that black-boxing lifts to a lax double functor from open graphs to a double category of profunctors. In Theorem 20 we identify a subclass of open graphs, called "functional", for which this double functor preserves composition up to isomorphism.

2.1 Operational Semantics of Graphs

In this thesis we use the definition of graph preferred by category theorists: the edges have a direction and multiple edges are allowed between pairs of vertices.

Definition 1. A **graph** is a pair of functions

$$E \overset{s}{\underset{t}{\rightrightarrows}} V.$$

A **morphism of graphs** is a pair of functions $f\colon E \to E'$ and $g\colon V \to V'$ such that the following diagrams

$$\begin{array}{ccc} E & \xrightarrow{s} & V \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle g} \\ E' & \xrightarrow[s']{} & V' \end{array} \qquad \begin{array}{ccc} E & \xrightarrow{t} & V \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle g} \\ E' & \xrightarrow[t']{} & V' \end{array}$$

commute. This defines a category Grph of graphs and their morphisms.

Let C be the category generated by the graph

$$\bullet \rightrightarrows \bullet \; .$$

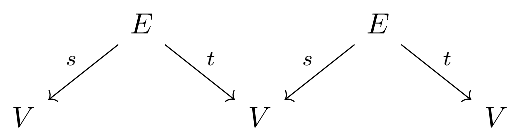
Then Grph is the same as the functor category $[C, \mathsf{Set}]$. This fact implies the following proposition:

Proposition 2. Grph *is complete and cocomplete with limits and colimits given pointwise in* Set.

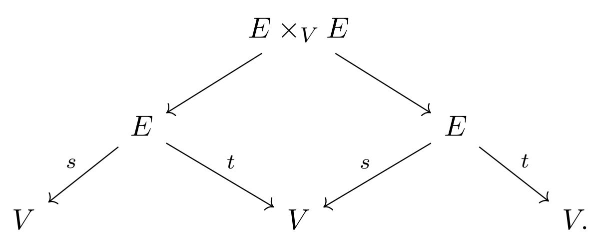
Here a pointwise (co)limit of graphs is given by first taking the (co)limits of their underlying edges and vertices and extending the corresponding source and target maps to these new (co)limits. Paths in a graph can be constructed using pullbacks. Let G be the graph

$$E \underset{t}{\overset{s}{\rightrightarrows}} V.$$

If G is placed next to itself:



then the pullback of the center two functions is computed as



The outermost legs of this diagram form a graph whose edges are described explicitly as

$$\{(e, e') \in E \times E \,|\, t(e) = s(e)\}$$

i.e. the paths of length 2 in G. These pullbacks can be iterated n times by placing n copies of G side by side and taking pullbacks until the outermost functions form a single span. Let G^n be the graph formed by this n-fold pullback. The edges of G^n are given by elements of the set

$$\{(e_1, e_2, \ldots, e_n) \,|\, t(e_1) = s(e_2)\,,\, t(e_2) = s(e_3)\,, \ldots, t(e_{n-1}) = s(e_n)\}$$

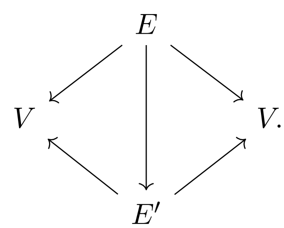
i.e. the set of paths of length n in G. G^0 is defined as the span

$$V \underset{1_V}{\overset{1_V}{\rightrightarrows}} V$$

so that its edges are given by all paths of length 0 in G. These powers are indeed iterated products in the following category:

Definition 3. Let $\mathsf{Span}(V,V)$ be the category where

- objects are spans $V \xleftarrow{s} E \xrightarrow{t} V$ and
- morphisms are commutative diagrams



The product in this category is the pullback of spans defined above. For graphs $V \xleftarrow{s} E \xrightarrow{t} V$ and $V \xleftarrow{s'} E' \xrightarrow{t'} V$, their coproduct in $\mathsf{Span}(V,V)$ is the graph $V \xleftarrow{(s,s')} E + E' \xrightarrow{(t,t')} V$ where (s,s') and (t,t') represent the pairings of the source and target functions in each graph. To account for paths of any length in G, we must combine the graphs G^n for all $n \geq 0$ using coproduct. The following proposition uses this idea to give the well-known free category construction:

Proposition 4. *Let* $U\colon \mathsf{Cat} \to \mathsf{Grph}$ *be the forgetful functor which sends a category to its underlying graph. Then* U *has a left adjoint*

$$F\colon \mathsf{Grph} \to \mathsf{Cat}$$

given by

$$F(G) = \sum_{n \geq 0} G^n$$

where products and sums are taken in the category of spans over the vertices of G*.*

Proof. A proof of this proposition can be found in many textbooks e.g. [ML98]. Alternatively, the result can be proved in a similar way as in Sections 3.6 and 5.2. In these cases as well as the above, the result follows from a general construction of free monoids over the relevant category of spans over a fixed object. Then the dependence on this object is removed using the Grothendieck construction. □

Because the morphisms of $F(G)$ are all paths in G, we borrow terminology from the theory of programming languages to call $F(G)$ the **operational semantics** of G. If G represents a program where nodes are states and edges are ways of changing the state, then $F(G)$ is a category whose morphisms represent all possible runs of your program. Note that with this operational semantics, graphs are non-deterministic, i.e. for a given state there is in general more than one run of the program starting with that state. Non-determinism will be a feature of all the types of networks we consider in this thesis. The operational semantics of a program encapsulates its behavior and is of use for model-checking and formal verification.

2.2 Open Graphs

To understand the compositionality of the operational semantics of graphs, we first need a paradigm where graphs are equipped with boundaries. These boundaries are represented by discrete graphs.

Proposition 5. *Let $R\colon \mathsf{Grph} \to \mathsf{Set}$ be the forgetful functor which sends a graph to its set of vertices and a function to its vertex component. Then R has a left adjoint*

$$L\colon \mathsf{Set} \to \mathsf{Grph}$$

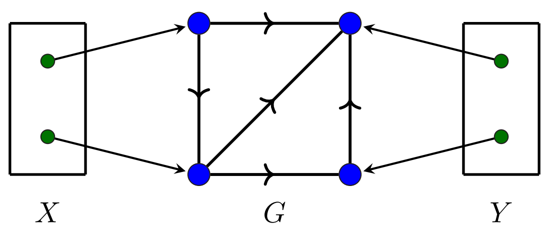
which sends a set X to the graph with no edges and X as its set of vertices.

Definition 6. A **open graph** is a cospan in Grph of the form

$$\begin{array}{ccc} & G & \\ \nearrow & & \nwarrow \\ LX & & LY \end{array}$$

and is denoted by $G\colon X \to Y$.

The idea is that the morphisms of the cospan designate certain vertices of G to be either inputs or outputs. An open graph $G\colon X \to Y$ is represented by a picture like this:



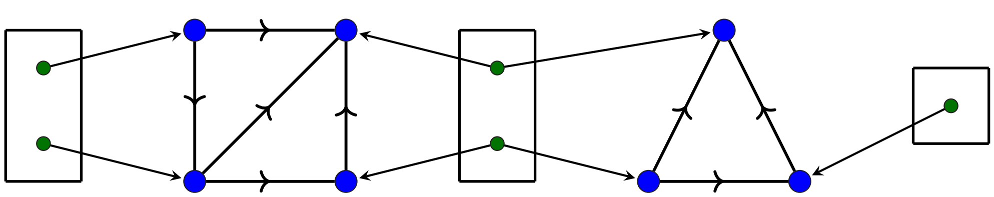
Given two composable open graphs

$$\begin{array}{ccccccc} & G & & & & H & \\ \nearrow & & \nwarrow & & \nearrow & & \nwarrow \\ LX & & & LY & & & LZ \end{array}$$

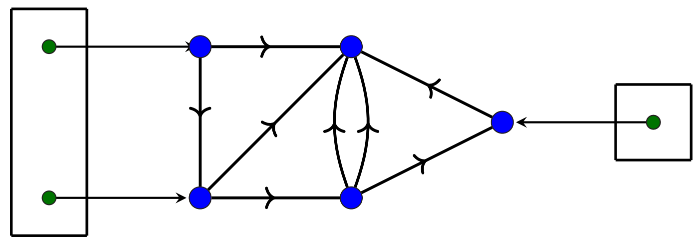
they are joined together using pushout

$$\begin{array}{ccccccc} & & & G +_{LY} H & & & \\ & & \nearrow & & \nwarrow & & \\ & G & & & & H & \\ \nearrow & & \nwarrow & & \nearrow & & \nwarrow \\ LX & & & LY & & & LZ \end{array}$$

to obtain an open graph whose apex is a graph obtained by gluing G and H along their shared boundary. For example, if G and H are these open graphs:



then their pushout is the open graph



obtained by identifying all vertices which are mapped to by a common point in their shared boundary. This gluing operation forms the horizontal composition of a double category. The formalisms developed by Fong [Fon16] and Courser [Cou20] define a syntax for gluing of open systems using cospans. We will use the following result from [Cou20] to construct a syntax for open graphs as well as the other networks considered in this thesis. Since this is a symmetric monoidal double category, it involves quite a lot of structure. The definition of symmetric monoidal double category can be found in Appendix A.

Lemma 7 (Courser). *Let* A *be a category with finite coproducts and* X *be a category with finite colimits. Given a left adjoint* $L\colon \mathsf{A} \to \mathsf{X}$*, there exists a unique symmetric monoidal double category* ${}_L\mathbb{C}\mathbf{sp}(\mathsf{X})$*, such that:*

- *objects are objects of* A*,*
- *vertical 1-morphisms are morphisms of* A*,*
- *a horizontal 1-cell from* $a \in \mathsf{A}$ *to* $b \in \mathsf{A}$ *is a cospan in* X *of this form:*

$$La \longrightarrow x \longleftarrow Lb$$

- *a 2-morphism is a commutative diagram in* X *of this form:*

$$\begin{array}{ccccc} La & \longrightarrow & x & \longleftarrow & Lb \\ {\scriptstyle Lf}\big\downarrow & & {\scriptstyle h}\big\downarrow & & \big\downarrow{\scriptstyle Lg} \\ Lc & \longrightarrow & y & \longleftarrow & Ld. \end{array}$$

Composition of vertical 1-morphisms is composition in A. *Composition of horizontal 1-cells is composition of cospans in* X *via pushout: given horizontal 1-cells*

$$\begin{array}{ccccc} & & x & & \\ & {\scriptstyle i_1}\nearrow & & \nwarrow{\scriptstyle o_1} & \\ La & & & & Lb \end{array} \qquad \begin{array}{ccccc} & & y & & \\ & {\scriptstyle i_2}\nearrow & & \nwarrow{\scriptstyle o_2} & \\ Lb & & & & Lc \end{array}$$

their composite is this cospan from La to Lc:

$$\begin{array}{ccccccccc} & & & & x +_{Lb} y & & & & \\ & & & {\scriptstyle j_x}\nearrow & & \nwarrow{\scriptstyle j_y} & & & \\ & & x & & & & y & & \\ & {\scriptstyle i_1}\nearrow & & \nwarrow{\scriptstyle o_1} & & {\scriptstyle i_2}\nearrow & & \nwarrow{\scriptstyle o_2} & \\ La & & & & Lb & & & & Lc \end{array}$$

where the diamond is a pushout square. The horizontal composite of 2-morphisms

$$\begin{array}{ccccc} La & \xrightarrow{i_1} & x & \xleftarrow{o_1} & Lb \\ {\scriptstyle Lf}\big\downarrow & & {\scriptstyle \alpha}\big\downarrow & & \big\downarrow{\scriptstyle Lg} \\ La' & \xrightarrow[i'^{1}]{} & x' & \xleftarrow[o'^{1}]{} & Lb' \end{array} \qquad \begin{array}{ccccc} Lb & \xrightarrow{i_2} & y & \xleftarrow{o_2} & Lc \\ {\scriptstyle Lg}\big\downarrow & & {\scriptstyle \beta}\big\downarrow & & \big\downarrow{\scriptstyle Lh} \\ Lb' & \xrightarrow[i'^{2}]{} & y' & \xleftarrow[o'^{2}]{} & Lc' \end{array}$$

is given by

$$\begin{array}{ccccc} La & \xrightarrow{\quad j_x i_1 \quad} & x +_{Lb} y & \xleftarrow{\quad j_y o_2 \quad} & Lc \\ {\scriptstyle Lf}\big\downarrow & & {\scriptstyle \alpha +_{Lg} \beta}\big\downarrow & & \big\downarrow{\scriptstyle Lh} \\ La' & \xrightarrow[\quad j_{x'} i'_1 \quad]{} & x' +_{Lb'} y' & \xleftarrow[\quad j_{y'} o'_2 \quad]{} & Lc'. \end{array}$$

The vertical composite of 2-morphisms

$$\begin{array}{ccccc} La & \xrightarrow{i_1} & x & \xleftarrow{o_1} & Lb \\ {\scriptstyle Lf}\downarrow & & {\scriptstyle \alpha}\downarrow & & \downarrow{\scriptstyle Lg} \\ La' & \xrightarrow[i_1']{} & x' & \xleftarrow[o_1']{} & Lb' \end{array}$$

$$\begin{array}{ccccc} La' & \xrightarrow{i_1'} & x' & \xleftarrow{o_1'} & Lb' \\ {\scriptstyle Lf'}\downarrow & & {\scriptstyle \alpha'}\downarrow & & \downarrow{\scriptstyle Lg'} \\ La'' & \xrightarrow[i_1'']{} & x'' & \xleftarrow[o_1'']{} & Lb'' \end{array}$$

is given by

$$\begin{array}{ccccc} La & \xrightarrow{i_1} & x & \xleftarrow{o_1} & Lb \\ {\scriptstyle L(f'f)}\downarrow & & {\scriptstyle \alpha'\alpha}\downarrow & & \downarrow{\scriptstyle L(g'g)} \\ La'' & \xrightarrow[i_1'']{} & x'' & \xleftarrow[o_1'']{} & Lb''. \end{array}$$

The tensor product is defined using chosen coproducts in A *and* X*. Thus, the tensor product of two objects* a_1 *and* a_2 *is* $a_1 + a_2$*, the tensor product of two vertical 1-morphisms*

$$\begin{array}{c} a_1 \\ {\scriptstyle f_1}\downarrow \\ b_1 \end{array} \qquad\qquad \begin{array}{c} a_2 \\ {\scriptstyle f_2}\downarrow \\ b_2 \end{array}$$

is

$$\begin{array}{c} a_1 + a_2 \\ {\scriptstyle f_1+f_2}\downarrow \\ b_1 + b_2, \end{array}$$

the tensor product of two horizontal 1-cells

$$La_1 \xrightarrow{i_1} x_1 \xleftarrow{o_1} Lb_1 \qquad\qquad La_2 \xrightarrow{i_2} x_2 \xleftarrow{o_2} Lb_2$$

is

$$L(a_1 + a_2) \xrightarrow{i_1+i_2} x_1 + x_2 \xleftarrow{o_1+o_2} L(b_1 + b_2),$$

and the tensor product of two 2-morphisms

$$
\begin{array}{ccccc}
La_1 & \xrightarrow{i_1} & x_1 & \xleftarrow{o_1} & Lb_1 \\
{\scriptstyle Lf_1}\downarrow & & \alpha_1 \downarrow & & \downarrow {\scriptstyle Lg_1} \\
La'_1 & \xrightarrow[i'_1]{} & x'_1 & \xleftarrow[o'_1]{} & Lb'_1
\end{array}
\qquad
\begin{array}{ccccc}
La_2 & \xrightarrow{i_2} & x_2 & \xleftarrow{o_2} & Lb_2 \\
{\scriptstyle Lf_2}\downarrow & & \alpha_2 \downarrow & & \downarrow {\scriptstyle Lg_2} \\
La'_2 & \xrightarrow[i'_2]{} & x'_2 & \xleftarrow[o'_2]{} & Lb'_2
\end{array}
$$

is

$$
\begin{array}{ccccc}
L(a_1+a_2) & \xrightarrow{i_1+i_2} & x_1+x_2 & \xleftarrow{o_1+o_2} & L(b_1+b_2) \\
{\scriptstyle L(f_1+f_2)}\downarrow & & \alpha_1+\alpha_2 \downarrow & & \downarrow {\scriptstyle L(g_1+g_2)} \\
L(a'_1+a'_2) & \xrightarrow[i'_1+i'_2]{} & x'_1+x'_2 & \xleftarrow[o'_1+o'_2]{} & L(b'_1+b'_2).
\end{array}
$$

The units for these tensor products are taken to be initial objects, and the symmetry is defined using the canonical isomorphisms $a + b \cong b + a$.

Proof. This was proved by Baez and Courser [BC20, Thm. 3.9]. Note that we are abusing language slightly above. We must choose a specific coproduct for each pair of objects in X and A to give ${}_L\mathbb{C}\mathbf{sp}(X)$ its tensor product. Given morphisms $i_1 \colon La_1 \to x_1$ and $i_2 \colon La_2 \to x_2$, their coproduct is really a morphism $i_1 + i_2 \colon La_1 + La_2 \to x_1 + x_2$ between these chosen coproducts. But since L preserves coproducts, we can compose this morphism with the canonical isomorphism $L(a_1 + a_2) \cong La_1 + La_2$ to obtain the morphism that we call $i_1 + i_2 \colon L(a_1 + a_2) \to x_1 + x_2$ above. □

Now we apply this lemma to the left adjoint L defined above.

Theorem 8. *There is a symmetric monoidal double category* $\mathsf{Open}(\mathsf{Grph})$ *where*

- *objects are sets X,Y,Z . . .*
- *vertical morphisms are functions $f : X \to Y$,*

- *a horizontal morphism $G\colon X \to Y$ is an open graph*

$$\begin{array}{ccc} & G & \\ \nearrow & & \nwarrow \\ LX & & LY \end{array}$$

- *vertical 2-morphisms are commutative rectangles*

$$\begin{array}{ccccc} LX & \longrightarrow & G & \longleftarrow & LY \\ {\scriptstyle Lf}\downarrow & & \downarrow{\scriptstyle g} & & \downarrow{\scriptstyle Lh} \\ LY' & \longrightarrow & H & \longleftarrow & LY' \end{array}$$

- *vertical composition is ordinary composition of functions,*

- *and horizontal composition of an open graph $G\colon X \to Y$ and an open graph $H\colon Y \to Z$ is given by their pushout.*

The symmetric monoidal structure is given by

- *coproducts in* Set *on objects and vertical morphisms,*

- *pointwise coproducts on horizontal morphisms i.e. for open graphs,*

$$\begin{array}{ccc} & G & \\ \nearrow & & \nwarrow \\ LX & & LY \end{array} \qquad \begin{array}{ccc} & G' & \\ \nearrow & & \nwarrow \\ LX' & & LY' \end{array}$$

  their coproduct is

$$\begin{array}{ccc} & G+G' & \\ \nearrow & & \nwarrow \\ LX+X' & & LY+Y' \end{array}$$

  and pointise coproduct for two vertical 2-morphisms i.e. for vertical 2-morphisms,

$$\begin{array}{ccccc} LX & \longrightarrow & G & \longleftarrow & LY \\ {\scriptstyle Lf}\downarrow & & \downarrow{\scriptstyle g} & & \downarrow{\scriptstyle Lh} \\ LZ & \longrightarrow & H & \longleftarrow & LQ \end{array} \quad \begin{array}{ccccc} LX' & \longrightarrow & G' & \longleftarrow & LY' \\ {\scriptstyle Lf'}\downarrow & & \downarrow{\scriptstyle g'} & & \downarrow{\scriptstyle Lh'} \\ LZ' & \longrightarrow & H' & \longleftarrow & LQ' \end{array}$$

their coproduct is

$$\begin{array}{ccccc} LX + X' & \longrightarrow & G + G' & \longleftarrow & LY + Y' \\ \downarrow{\scriptstyle Lf+f'} & & \downarrow{\scriptstyle g+g'} & & \downarrow{\scriptstyle Lh+h'} \\ LZ + Z' & \longrightarrow & H + H' & \longleftarrow & LQ + Q' \end{array}$$

Proof. Lemma 7 constructs this symmetric monoidal double category given that

- Grph has coproducts and pushouts,

- and $L\colon \mathsf{Set} \to \mathsf{Grph}$ preserves pushouts and coproducts.

The first point follows from Proposition 2 and the second point is true because L is a left adjoint. □

2.3 Compositional Operational Semantics of Graphs

In this section we show how the operational semantics of graphs

$$F\colon \mathsf{Grph} \to \mathsf{Cat}$$

defined in Section 2.1 is extended to open graphs. Categories can also be made open in a similar way as graphs.

Definition 9. An **open category** is a cospan in Cat of the form

$$\begin{array}{ccc} & C & \\ \nearrow & & \nwarrow \\ FLX & & FLY \end{array}$$

Categories are also glued together using pushout.

Pushout of open categories also gives the horizontal composition of a symmetric monoidal double category.

Theorem 10. *There is a symmetric monoidal double category* Open(Cat) *where*

- *objects are sets* X,Y,$Z \ldots$
- *vertical morphisms are functions* $f : X \to Y$,
- *a horizontal morphism* $C \colon X \to Y$ *is an open category*

$$\begin{array}{ccc} & C & \\ \nearrow & & \nwarrow \\ FLX & & FLY \end{array}$$

- *vertical 2-morphisms are commutative rectangles*

$$\begin{array}{ccccc} FLX & \longrightarrow & C & \longleftarrow & FLY \\ {\scriptstyle FLf}\downarrow & & \downarrow{\scriptstyle g} & & \downarrow{\scriptstyle FLh} \\ FLY' & \longrightarrow & D & \longleftarrow & FLY' \end{array}$$

- *vertical composition is ordinary composition of functions,*
- *and horizontal composition of an open category* $C \colon X \to Y$ *and an open graph* $D \colon Y \to Z$ *is given by their pushout.*

The symmetric monoidal structure is given by

- *coproducts in* Set *on objects and vertical morphisms,*
- *pointwise coproducts on horizontal morphisms i.e. for open categories,*

$$\begin{array}{ccc} & C & \\ \nearrow & & \nwarrow \\ FLX & & FLY \end{array} \qquad \begin{array}{ccc} & C' & \\ \nearrow & & \nwarrow \\ FLX' & & FLY' \end{array}$$

their coproduct is

$$\begin{array}{ccc} & C + C' & \\ \nearrow & & \nwarrow \\ FL(X + X') & & FL(Y + Y') \end{array}$$

and pointise coproduct for two vertical 2-morphisms i.e. for vertical 2-morphisms,

$$\begin{array}{ccccc} FLX & \longrightarrow & C & \longleftarrow & FLY \\ {\scriptstyle FLf}\downarrow & & \downarrow{\scriptstyle g} & & \downarrow{\scriptstyle FLh} \\ FLZ & \longrightarrow & D & \longleftarrow & FLQ \end{array} \qquad \begin{array}{ccccc} FLX' & \longrightarrow & C' & \longleftarrow & FLY' \\ {\scriptstyle FLf'}\downarrow & & \downarrow{\scriptstyle g'} & & \downarrow{\scriptstyle FLh'} \\ FLZ' & \longrightarrow & D' & \longleftarrow & FLQ' \end{array}$$

their coproduct is

$$\begin{array}{ccccc} FL(X+X') & \longrightarrow & C+C' & \longleftarrow & FLY+Y' \\ {\scriptstyle FLf+f'}\downarrow & & \downarrow{\scriptstyle g+g'} & & \downarrow{\scriptstyle FLh+h'} \\ FLZ+Z' & \longrightarrow & D+D' & \longleftarrow & FLQ+Q' \end{array}$$

Proof. This double category is constructed by applying Lemma 7 to the composite left adjoint

$$\mathsf{Set} \xrightarrow{L} \mathsf{Grph} \xrightarrow{F} \mathsf{Cat}. \qquad \square$$

Theorem 4.3 of [BC20] shows how a commutative square of left adjoint functors lifts to a symmetric monoidal double functor. In this thesis we require a weakening of this result.

Lemma 11. *Let*

$$\begin{array}{ccc} C & \xrightarrow{F} & D \\ {\scriptstyle L}\uparrow & & \uparrow{\scriptstyle L'} \\ A & \xrightarrow[F_0]{} & A' \end{array}$$

be a diagram commuting up to natural isomorphism where L, L', and F_0 preserve finite colimits. Then there is a lax symmetric monoidal lax double functor

$$\mathsf{Open}(F)\colon \mathsf{Open}(C) \to \mathsf{Open}(D)$$

which is given by F_0 on objects and morphisms of A and is given by pointwise application of F on horizontal morphisms and 2-cells. Explicitly, 0-cells and vertical morphisms are *mapped as follows:*

$$\begin{array}{ccc} A & & F_0(A) \\ \downarrow f & \mapsto & \downarrow F_0(f) \\ B & & F_0(B) \end{array}$$

horizontal morphisms are mapped as follows:

$$LX \longrightarrow G \longleftarrow LY \qquad \mapsto \qquad L'F_0X \longrightarrow FG \longleftarrow L'F_0Y$$

and vertical 2-cells are mapped as follows:

$$\begin{array}{ccccc} LX & \longrightarrow & G & \longleftarrow & LY \\ {\scriptstyle Lf}\downarrow & & \downarrow{\scriptstyle g} & & \downarrow{\scriptstyle Lh} \\ LX' & \longrightarrow & H & \longleftarrow & LY' \end{array} \longmapsto \begin{array}{ccccc} L'F_0X & \longrightarrow & FG & \longleftarrow & L'F_0Y \\ {\scriptstyle L'F_0f}\downarrow & & \downarrow{\scriptstyle Fg} & & \downarrow{\scriptstyle L'F_0h} \\ L'F_0X' & \longrightarrow & FG' & \longleftarrow & L'F_0Y' \end{array}$$

Proof. Theorem 4.3 of [BC20] supplies a symmetric monoidal double functor for the above square when all functors preserve finite colimits. However, a lax symmetric monoidal lax double functor (as opposed to pseudo) can nevertheless be constructed when F is an arbitrary functor. The laxator of composition for this double functor

$$\begin{array}{ccccc} & & F(G) +_{L'Y} F(H) & & \\ & \nearrow & & \nwarrow & \\ L'X & & \downarrow{\scriptstyle \phi_{GH}} & & L'Z \\ & \searrow & & \swarrow & \\ & & F(G +_{LY} H) & & \end{array}$$

is induced by the universal property of pushout on the morphisms $F(i)\colon F(G) \to F(G +_{LY} H)$ and $F(j)\colon F(H) \to F(G +_{LY} H)$ where $i\colon G \hookrightarrow G +_{LY} H$ and $j\colon H \hookrightarrow G +_{LY} H$ are the canonical inclusions. Similarly, the monoidal comparison

$$\begin{array}{ccccc} & & F(G) + F(H) & & \\ & \nearrow & & \nwarrow & \\ L'(X + X') & & \downarrow{\scriptstyle \psi_{GH}} & & L'(Y + Y') \\ & \searrow & & \swarrow & \\ & & F(G + H) & & \end{array}$$

is induced by the universal property of coproduct applied to the morphisms $F(i)\colon F(G) \to F(G+H)$ and $F(j)\colon F(H) \to F(G+H)$ where i and j are the canonical inclusions into the coproduct. Verifying that this structure satisfies the axioms of a symmetric monoidal double functor follows the proof of Theorem 4.3 in [BC20] very closely. □

The discrete graph functor and the free category functor assemble into a diagram

$$\begin{array}{ccc} \mathsf{Grph} & \xrightarrow{F} & \mathsf{Cat} \\ {\scriptstyle L}\uparrow & & \uparrow{\scriptstyle FL} \\ \mathsf{Set} & = & \mathsf{Set} \end{array}$$

of commuting left adjoint functors. The following theorem is given by applying Lemma 11 to this diagram. Because F preserves colimits, horizontal composition is preserved up to isomorphism.

Theorem 12. *There is a symmetric monoidal double functor*

$$\mathsf{Open}(F)\colon \mathsf{Open}(\mathsf{Grph}) \to \mathsf{Open}(\mathsf{Cat})$$

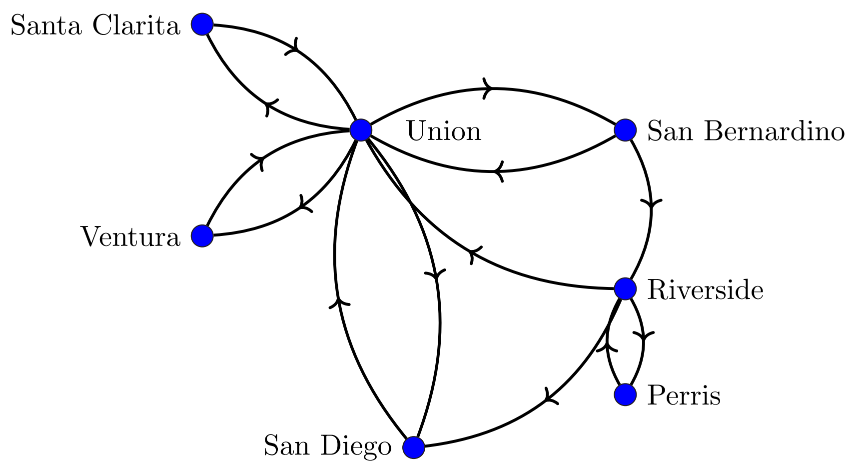
which is the identity on objects and functions and is given by pointwise application of F on horizontal morphisms and 2-morphisms. Explicitly, the horizontal morphisms and two morphisms are mapped as follows:

$$LX \longrightarrow G \longleftarrow LY \qquad \mapsto \qquad FLX \longrightarrow FG \longleftarrow FLY]$$

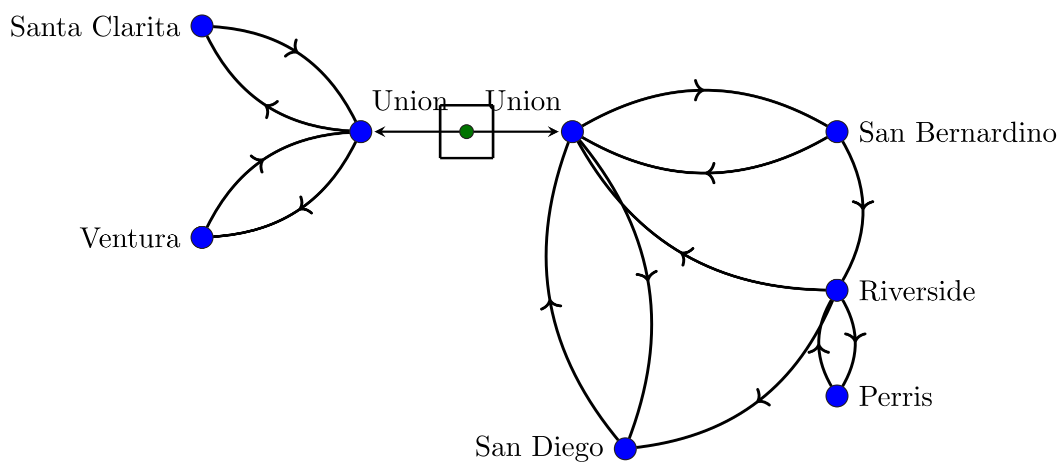
$$\begin{array}{ccccc} LX & \longrightarrow & G & \longleftarrow & LY \\ {\scriptstyle Lf}\downarrow & & \downarrow{\scriptstyle g} & & \downarrow{\scriptstyle Lh} \\ LX' & \longrightarrow & H & \longleftarrow & LY' \end{array} \quad \longmapsto \quad \begin{array}{ccccc} FLX & \longrightarrow & FG & \longleftarrow & FLY \\ {\scriptstyle FLf}\downarrow & & \downarrow{\scriptstyle Fg} & & \downarrow{\scriptstyle FLh} \\ FLX' & \longrightarrow & FG' & \longleftarrow & FLY' \end{array}$$

This symmetric monoidal double functor gives a compositional framework for building the operational semantics of a graph recursively. A graph is decomposed into component open graphs, the operational semantics on each of these components is computed and then they are joined together using pushout to form the operational semantics of the total graph.

Example 13. Consider this non-exhaustive map of train routes in Southern California regarded as an open graph $G\colon \phi \to \phi$:



This graph is decomposed into open graphs $A\colon \phi \to \{*\}$ and $B\colon \{*\} \to \phi$ as follows:



The category $F(A)$ has three objects, Santa Clarita, Union, and Ventura and the morphisms are given by all possible paths on the graph A. For example a morphism is given by the unique path

$$f = \text{Santa Clarita} \to \text{Union} \to \text{Ventura} \to \text{Union}$$

Similarly the category $F(B)$ has stations as objects and paths as morphisms. For example, $F(B)$ contains the unique morphism represented by the path

$$g = \text{Union} \to \text{San Bernardino} \to \text{Riverside} \to \text{Union}$$

Because $\mathsf{Open}(F)$ is a double functor, it is equipped with an isomorphism

$$\begin{array}{ccccc} & & F(A) +_1 F(B) & & \\ & \nearrow & \big\downarrow \wr & \nwarrow & \\ 0 & & & & 0 \\ & \searrow & & \swarrow & \\ & & F(G) & & \end{array}$$

where 0 is the empty category and 1 is the terminal category. This isomorphism builds the operational semantics $F(G)$ using the operational semantics $F(A)$ and $F(B)$. First, paths such as f and g are formed then combined together using the pushout of categories. The pushout of categories requires a closure under all paths which travel back and forth between A and B. In particular, because f and g both start and end at Union, $F(G)$ must contain all words in f and g as morphisms.

The additional axioms of a double functor ensure that building the operational semantics of an open graph from its components is independent of reassociating composition or adding identity horizontal morphisms. The second transitive closure required to join together operational semantics can be very computationally expensive and this leads to a combinatorial explosion. Although $\mathsf{Open}(F)$ provides a useful conceptual framework for building the operational semantics of graphs recursively, the isomorphisms provided are unlikely to provide a speed up without further assumptions or restrictions. In the next section we identify a subclass of open graphs, called functional, which can be joined together without combinatorial explosion.

2.4 Black-boxing and Functional Open Graphs

In this section we define a "black-boxing" functor, which restricts an open category to the data between its boundaries. This matches the traditional meaning of black-boxing in systems theory, i.e. forgetting about the internal workings of a system and concentrating only on the relationship it induces between its inputs and ouputs. In this case, the black-boxing of an open category is a profunctor whose data consists of the morphisms that travel from the input ports to the output port.

Our black-boxing is in general only laxly functorial. However, we define a subclass of open graphs for which the functoriality is strict.

Definition 14. For an open category

$$\begin{array}{ccccc} & & C & & \\ & \nearrow & & \nwarrow & \\ FLX & & & & FLY \end{array}$$

its black-boxing $\blacksquare(C)$ is a profunctor

$$\blacksquare(C)\colon FLX \times FLY \to \mathsf{Set}$$

given by $\blacksquare(C)(x, y) = C(i(x), j(y))$.

This black-boxing operation extends to a lax double functor into the double category of profunctors [Shu08, Ex. 2.6].

Definition 15. Let Prof be the double category where

- objects are categories,
- vertical morphisms are functors,
- horizontal morphisms are profunctors, and
- vertical 2-cells are squares

$$\begin{array}{ccc} A & \xrightarrow{P} & B \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle g} \\ C & \xrightarrow[Q]{} & D \end{array}$$

  equipped with a natural transformation

$$\begin{array}{ccc} A \times B^{op} & \xrightarrow{f \times g^{op}} & C \times D^{op} \\ & \overset{\alpha}{\Longrightarrow} & \\ {\scriptstyle P} \searrow & \mathsf{Set} & \swarrow {\scriptstyle Q} \end{array}$$

Theorem 16. *Black-boxing lifts to a double functor*

$$\blacksquare \colon \mathsf{Open(Cat)} \to \mathsf{Prof}$$

Proof. On objects $\blacksquare$ sends a set X to its discrete category 1_X and functions are sent to their unique extensions on these discrete categories. A 2-cell

$$\begin{array}{ccccc} FLX & \xrightarrow{i} & C & \xleftarrow{j} & FLY \\ {\scriptstyle FLf}\downarrow & & \downarrow{\scriptstyle g} & & \downarrow{\scriptstyle FLh} \\ FLX' & \xrightarrow[i']{} & D & \xleftarrow[j']{} & FLY' \end{array}$$

is sent to the natural transformation

$$\begin{array}{ccc} FLX \times FLY & \xrightarrow{FLf \times FLh} & FLX' \times FLY' \\ & \overset{\blacksquare(g)}{\Longrightarrow} & \\ {\scriptstyle \blacksquare(C)} \searrow & \mathsf{Set} & \swarrow {\scriptstyle \blacksquare(D)} \end{array}$$

with components

$$\blacksquare(g)(x,y) \colon \blacksquare(C)(x,y) \to \blacksquare(D)(f(x), h(y))$$

given by the restriction of the functor g to the hom-set $C(i(x), j(y))$. $\blacksquare(g)$ is well-defined because of the commutativity of the 2-cell it comes from. Naturality of $\blacksquare(g)$ is trivial because $FLX \times FLY$ has no non-identity arrows. Functoriality of $\blacksquare$ on object and arrow categories follows immediately from the definitions. An identity 2-cell in $\mathsf{Open}(\mathsf{Cat})$

$$\begin{array}{ccccc} & & FLX & & \\ & /\!\!/ & & \backslash\!\!\backslash & \\ FLX & & & & FLX \end{array}$$

is sent to a profunctor $\blacksquare(FLX)$ given by

$$\blacksquare(FLX)(x, x') = \delta_{xx'}$$

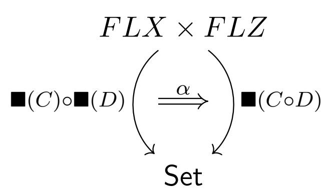
where $\delta_{xx'}$ is the function which returns a one element set when $x = x'$ and the empty set otherwise. $\blacksquare(FLX)$ is clearly the identity profunctor on the category FLX so our double functor $\blacksquare$ preserves identities. Consider a composable pair of horizontal morphisms

$$\begin{array}{ccccccc} & D & & & & C & \\ \nearrow & & \nwarrow & & \nearrow & & \nwarrow \\ FLX & & & FLY & & & FLZ \end{array}$$

in $\mathsf{Open}(\mathsf{Cat})$. Note that for every $y \in Y$ there is a function

$$\circ_y \colon \blacksquare(D)(x, y) \times \blacksquare(C)(y, z) \to \blacksquare(C \circ D)(x, z)$$

sending a pair of morphisms to their composite in $C \circ D$. The composition comparison of $\blacksquare$ is a natural transformation

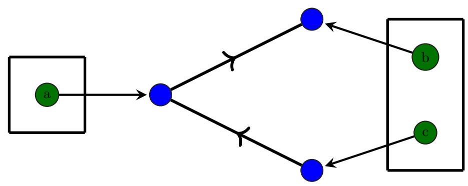


with components

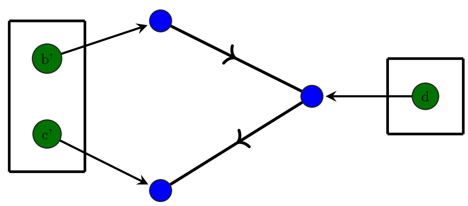
$$\alpha_{x,z} \colon \int_{y \in Y} \blacksquare(D)(x, y) \times \blacksquare(C)(y, z) \to \blacksquare(C \circ D)(x, z)$$

given by stitching together the functions $\circ_y$ with the universal property of coends. $\square$

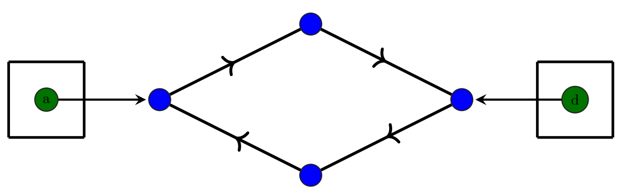
Example 17. Let $G\colon 1 \to 2$ be the open graph



and let $H\colon 2 \to 1$ be the open graph



so that their composite $H \circ G\colon 1 \to 1$ is



The black-boxing $\blacksquare(G)\colon FL(1) \times FL(2) \to \mathsf{Set}$ has the singleton as $\blacksquare(G)(a, b)$ representing the only path from a to b and $\blacksquare(G)(a, c)$ is the empty set. Similarly, $\blacksquare(H)\colon FL(2) \times FL(1) \to \mathsf{Set}$ has the singleton for $\blacksquare(H)(b', d)$ and the empty set for $\blacksquare(c', d)$. Their composite $\blacksquare(H) \circ \blacksquare(G)\colon FL(1) \times FL(1) \to \mathsf{Set}$ has only one value and it's given by the coend formula

$$\blacksquare(H) \circ \blacksquare(G)(a, d) = \int_{x \in 2} \blacksquare(G)(a, x) \times \blacksquare(H)(x, d)$$

In this case the coend is again the singleton, whose unique element represents the path in $H \circ G$ traversing the top two edges. Note that there are many more elements in $\blacksquare(H \circ$

$G)(a,d)$. Paths in $H \circ G$ may circle around any natural number of times before arriving at their destination and this gives an element of $\blacksquare(H \circ G)$ for every finite natural number n. The laxator of composition for $\blacksquare$ sends the unique element of $\blacksquare(H) \circ \blacksquare(G)(a,d)$ to element 0 in $\blacksquare(H \circ D)(a,d)$ representing the path which goes directly from a to d without appending any loops.

The previous example may be somewhat discouraging, in general $\blacksquare(H \circ G)$ will have much larger sets than $\blacksquare(H) \circ \blacksquare(G)$ so any technique for constructing the former from the latter will be rife with difficulty. However, we can define a sort of open graph for which the looping behavior of Example 17 is disallowed.

Definition 18. Let $G = \; E \overset{s}{\underset{t}{\rightrightarrows}} V \;$ be a graph. Then a vertex $x \in V$ is a **source** if it has no incoming edges i.e. $t^{-1}(x) = \phi$. Similarly, x is a **sink** if it has no outgoing edge i.e. $s^{-1}(x) = \phi$. An open graph

$$LX \xrightarrow{i} G \xleftarrow{o} LY$$

is **functional** if for every $x \in X$, $i(x)$ is a source and for $y \in Y$, $j(y)$ is a sink.

The following theorem relies on a lemma.

Lemma 19. *For functional open graphs $G \colon X \to Y$ and $H \colon Y \to Z$,*

$$\blacksquare((H \circ G)^n) \cong \sum_{i+j=n} \blacksquare(H^i) \circ \blacksquare(G^j)$$

where the powers refer to iterated pullbacks of a graph with itself.

Proof. For elements $x \in X$ and $z \in Z$, let f be an element of $\blacksquare((H \circ G)^n)(x,z)$. Then f is a sequence of composable edges $(e_1, e_2, \ldots, e_n)$ such that the source of e_1 is x and the target of e_n is z. Because G and H are functional, if e_i is an edge of H, then e_{i+1} cannot be an element of G. Therefore there is a last occurence $1 \leq i_* \leq n$ such that e_{i_*} is an edrnge of G and e_i is an edge of H for all $i > i_*$. The n-tuple $(e_1, e_2, \ldots, e_n)$ can be split into tuples $(e_1, e_2, \ldots, e_{i_*}) \in \blacksquare(G^{i_*})(x, y)$ and $(e_{i_*+1}, e_{i_*+2}, \ldots, e_n) \in \blacksquare(H^{n-i_*})(y, z)$. The composite profunctor

$$\blacksquare(H^{n-i_*}) \circ \blacksquare(G^{i_*})(x, z) = \int_{y \in Y} \blacksquare(G^{i_*})(x, y) \times \blacksquare(H^{n-i_*})(y, z)$$

accounts for every possible y value that the path can stop in. Because i_* can occur at any value, we need to to take the coproduct of the above composite for every power of H and G summing to n in order to account for every element of $\blacksquare((H \circ G)^n)(x, z)$. □

Theorem 20. *The composite*

$$\blacksquare \circ \mathsf{Open}(F) \colon \mathsf{Open}(\mathsf{Grph}) \to \mathsf{Prof}$$

preserves horizontal composition on functional open graphs up to isomorphism.

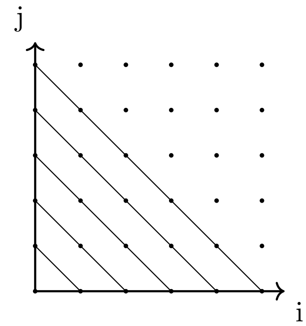
Proof. We show that for functional open graphs $G \colon X \to Y$ and $H \colon Y \to Z$, there is an isomorphism

$$\blacksquare(\sum_{n \geq 0} (H \circ G)^n) \cong \blacksquare(\sum_{i \geq 0} H^i) \circ \blacksquare(\sum_{j \geq 0} G^j)$$

Starting with the right hand side, we pull the $\blacksquare$'s inside the sums and use the distributive law

$$\blacksquare(\sum_{i \geq 0} H^i) \circ \blacksquare(\sum_{j \geq 0} G^j) \cong \sum_{i \geq 0} \sum_{j \geq 0} \blacksquare(H^i) \circ \blacksquare(G^j)$$

The values of i and j in this sum are represented by the grid points on the cartesian axes below



The above double coproduct is arranged so that entries are summed horizontally first and then going up vertically. A rearrangement this sum is represented by the diagonal lines, starting from $(0,0)$ and summing each diagonal before moving on to the next. The n-th diagonal line crosses all coordinates (i,j) with $i+j=n$ so the above sum can be rearranged as the sum going over the diagonals i.e.

$$\sum_{n\geq 0}\sum_{i+j=n} \blacksquare(H^i)\circ \blacksquare(G^j).$$

By Lemma 19 this composite is isomorphic to

$$\sum_{n\geq 0} \blacksquare((H\circ G)^n)$$

Pulling $\blacksquare$ out of the sum gives the desired result. □

This theorem says the operational semantics of a composite of open graphs can be computed using profunctor composition when paths can only go from the first open graph to the second. Lemma 19, says that the paths which occur in exactly n steps on a composite of functional open graphs can be computed compositionally using the given formula. In what follows we will generalize these results to a larger range of networks which can be described by enriched and internal graphs.

Chapter 3

Operational Semantics of Q-Nets

In this chapter we construct the operational semantics for Petri nets and many of their related variants by generalizing the free category construction of Proposition 4 to the case of graphs internal to $\mathsf{Mod}(\mathsf{Q})$, the category of models of a Lawvere theory Q. Petri nets are not internal graphs. However, they are the generating data for graphs internal to $\mathsf{CommMon}$, the category of commutative monoids. In Section 3.1 we review the basic definitions of Petri net theory. If Petri nets are the generating data for graphs internal to $\mathsf{CommMon}$, more generally, what is the generating data for graphs internal to $\mathsf{Mod}(\mathsf{Q})$? This chapter answers this question by introducing Q-nets, a generalization of Petri net based on the operations and axioms of a Lawvere theory Q. In Section 3.2, we define Q-nets and show how many well-known variants of Petri nets and the relationships between them can be understood using this definition. The main goal of this chapter is to construct an adjunction representing the operational semantics of Q-nets. We start with a motivating example: when Q is the Lawvere theory for commutative monoids we obtain Petri nets. In

Section 3.3, we construct an adjunction

$$\mathsf{Petri} \underset{U}{\overset{F}{\rightleftarrows}} \mathsf{CMC}$$

where CMC is the category of commutative monoidal categories. In Section 3.4 we state Theorem 45, the main result of this chapter: for every Lawvere theory Q, there is an adjunction

$$\mathsf{Q}\text{-}\mathsf{Net} \underset{U_{\mathsf{Q}}}{\overset{F_{\mathsf{Q}}}{\rightleftarrows}} \mathsf{Q}\text{-}\mathsf{Cat}$$

between the category of Q-nets and the category of "Q-categories", i.e., models of Q internal to the category of categories. For a Q-net P, the Q-category $F_{\mathsf{Q}}(P)$ is a category whose morphisms represent the execution sequences of P. To construct this adjunction we factor it as

$$\mathsf{Q}\text{-}\mathsf{Net} \underset{\mathrm{A}^{\bullet}_{\mathsf{Q}}}{\overset{{}^{\bullet}\mathrm{A}_{\mathsf{Q}}}{\rightleftarrows}} \!\!\!\perp\; \mathsf{Grph}(\mathsf{Mod}(\mathsf{Q})) \;\perp \underset{\mathrm{B}^{\bullet}_{\mathsf{Q}}}{\overset{{}^{\bullet}\mathrm{B}_{\mathsf{Q}}}{\rightleftarrows}} \mathsf{Mod}(\mathsf{Q}, \mathsf{Cat})$$

where $\mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))$ is the category of graphs internal to $\mathsf{Mod}(\mathsf{Q})$. In Section 3.5 we construct the first part ${}^{\bullet}\mathrm{A} \dashv \mathrm{A}^{\bullet}$ and in Section 3.6 we construct the second part ${}^{\bullet}\mathrm{B} \dashv \mathrm{B}^{\bullet}$.

3.1 Petri Nets and Their Executions

Petri nets are the motivating example for the generalizations considered here. Therefore we review their properties here in depth to provide intuition for the more abstract treatment later on. Petri nets are regarded as a graph, whose source and target land in a free commutative monoid. To describe this graph we need the following adjunction.

Definition 21. Let $L\colon \mathsf{Set} \to \mathsf{CommMon}$ be the free commutative monoid functor, that is, the left adjoint of the functor $R\colon \mathsf{CommMon} \to \mathsf{Set}$ that sends commutative monoids to their underlying sets and monoid homomorphisms to their underlying functions. Let

$$\mathbb{N}\colon \mathsf{Set} \to \mathsf{Set}$$

be the **free commutative monoid monad** given by the composite $R \circ L$.

For any set X, $\mathbb{N}[X]$ is the set of formal finite linear combinations of elements of X with natural number coefficients. The unit of $\mathbb{N}$ is given by the natural inclusion of X into $\mathbb{N}[X]$, and for any function $f\colon X \to Y$, $\mathbb{N}[f]\colon \mathbb{N}[X] \to \mathbb{N}[Y]$ is the unique monoid homomorphism that extends f.

Definition 22. We define a **Petri net** to be a pair of functions of the following form:

$$T \underset{t}{\overset{s}{\rightrightarrows}} \mathbb{N}[S].$$

We call T the set of **transitions**, S the set of **places**, s the **source** function, and t the **target** function.

Definition 23. A **Petri net morphism** from the Petri net $T \underset{t}{\overset{s}{\rightrightarrows}} \mathbb{N}[S]$ to the Petri net $T' \underset{t'}{\overset{s'}{\rightrightarrows}} \mathbb{N}[S']$ is a pair of functions $(f\colon T \to T', g\colon S \to S')$ such that the diagrams

$$\begin{array}{ccc} T & \xrightarrow{s} & \mathbb{N}[S] \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle \mathbb{N}[g]} \\ T' & \xrightarrow[s']{} & \mathbb{N}[S'] \end{array} \qquad \begin{array}{ccc} T & \xrightarrow{t} & \mathbb{N}[S] \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle \mathbb{N}[g]} \\ T' & \xrightarrow[t']{} & \mathbb{N}[S']. \end{array}$$

commute.

Definition 24. Let Petri be the category of Petri nets and Petri net morphisms, with composition defined by

$$(f, g) \circ (f', g') = (f \circ f', g \circ g').$$

Our definition of Petri net morphism differs from the earlier definition used by Degano–Meseguer–Montanari [DMM89] and Sassone [Sas95, Sas96, BMMS01]. The difference is that our definition requires that the homomorphism between free commutative monoids come from a function between the sets of places whereas the above references allow arbitrary commutative monoid homomorphisms. This difference of definition is present in our definition of the category of Q-nets as well. With this change, the categories Petri and $\mathsf{Q\text{-}Net}$ become complete and cocomplete as shown in Proposition 40. This is important for the compositionality results in Chapter 4.

Petri nets have a natural semantics which is described by "the token game". This is a game where each place of a Petri net is equipped with a natural number of tokens. Players are then allowed to shuffle the tokens from place to place using the transitions. The token game is formalized by the notions of marking and firing.

Definition 25. A **marking** of a Petri net $P = \; T \underset{t}{\overset{s}{\rightrightarrows}} \mathbb{N}[S]$ is an element $m \in \mathbb{N}[S]$, or equivalently, a function $m \colon S \to \mathbb{N}$ which is zero on all but a finite number of elements. A **firing** of P is a tuple (τ, x, y), where τ is a transition and x and y are markings of P with $x - s(\tau) \geq 0$ and $x - s(\tau) + t(\tau) = y$.

Firings can be chained together in sequence: for a firing (τ, x, y) and a firing (σ, y, z) we can define their composite as a tuple $(\sigma \circ \tau, x, z)$ where $\circ$ is a formal symbol. Firings can also be performed in parallel: for two firings (τ, x, y) and (τ', x', y') there is a parallelization $(\tau + \tau', x + x', y + y')$. This suggests that firings of a Petri net have the structure of a monoidal category. Meseguer and Montanari were the first to notice this and show how Petri nets can be turned into commutative monoidal categories [MM90].

Definition 26. A **commutative monoidal category** is a commutative monoid object internal to Cat. Explicitly, a commutative monoidal category is a strict monoidal category $(C, \otimes, I)$, such that, for all objects a, b and morphisms f, g in C

$$a \otimes b = b \otimes a \text{ and } f \otimes g = g \otimes f.$$

Note that a commutative monoidal category is the same as a strict symmetric monoidal category where the symmetry isomorphisms $\sigma_{a,b} \colon a \otimes b \xrightarrow{\sim} b \otimes a$ are all identity morphisms. In fact, a commutative monoidal category is precisely a category where the objects and morphisms form commutative monoids and the structure maps are commutative monoid homomorphisms. A commutative monoidal category where the morphisms represent sequences of firings of a Petri net P will be referred to as the "operational semantics" of P. In this chapter we characterize this semantics construction as an adjunction between the category of Petri nets and the following category.

Definition 27. Let CMC be the category whose objects are commutative monoidal categories and whose morphisms are strict monoidal functors.

Note that every monoidal functor between commutative monoidal categories is automatically a strict symmetric monoidal functor, so the adjective symmetric is not included in the above definition.

3.2 Q-Nets

Petri nets need not have a free commutative monoid of places, and this aspect can be generalized using Lawvere theories. A review of the basic definitions and properties of Lawvere theories can be found in in Appendix B. As in Definition 110, let $\mathsf{Mod}(\mathsf{Q})$ be the category of models of Q in Set,

$$\mathsf{Set} \underset{R_{\mathsf{Q}}}{\overset{L_{\mathsf{Q}}}{\rightleftarrows}} \mathsf{Mod}(\mathsf{Q}) \quad (L_{\mathsf{Q}} \dashv R_{\mathsf{Q}})$$

be the adjunction generating free models of Q and let M_{Q} be the composite $R_{\mathsf{Q}} \circ L_{\mathsf{Q}}$.

Definition 28. Let $\mathsf{Q}\text{-}\mathsf{Net}$ be the category where

- objects are **Q-nets**, i.e. pairs of functions of the form

$$T \underset{t}{\overset{s}{\rightrightarrows}} M_{\mathsf{Q}}S$$

- a morphism from the Q-net $T \underset{t}{\overset{s}{\rightrightarrows}} M_{\mathsf{Q}}S$ to the Q-net $T' \underset{t'}{\overset{s'}{\rightrightarrows}} M_{\mathsf{Q}}S'$ is a pair of functions $(f \colon T \to T', g \colon S \to S')$ such that the following diagrams commute:

$$\begin{array}{ccc} T & \xrightarrow{s} & M_{\mathsf{Q}}S \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle M_{\mathsf{Q}}g} \\ T' & \xrightarrow[s']{} & M_{\mathsf{Q}}S' \end{array} \qquad \begin{array}{ccc} T & \xrightarrow{t} & M_{\mathsf{Q}}S \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle M_{\mathsf{Q}}g} \\ T' & \xrightarrow[t']{} & M_{\mathsf{Q}}S'. \end{array}$$

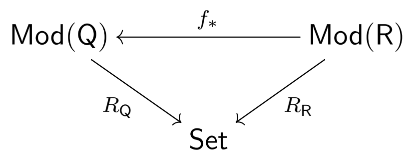
This definition lifts to a functor. Let M_{Q} be as before and let $M_{\mathsf{R}} \colon \mathsf{Set} \to \mathsf{Set}$ be corresponding monad induced by a Lawvere theory R. Every morphism of Lawvere theories $f \colon \mathsf{Q} \to \mathsf{R}$ induces a functor

$$f_* \colon \mathsf{Mod}(\mathsf{R}) \to \mathsf{Mod}(\mathsf{Q})$$

which composes every model of R with f. A left adjoint

$$f^*\colon \mathsf{Mod}(\mathsf{Q}) \to \mathsf{Mod}(\mathsf{R})$$

is given by the left Kan extension of each model along f [BW85, Buc08]. Now, we have the following commutative diagram of functors



all of which have left adjoints. Given this set of assumptions, there is a morphism of monads M^f given by

$$M^f = R_{\mathsf{Q}} \eta L_{\mathsf{Q}} \colon M_{\mathsf{Q}} \Rightarrow M_{\mathsf{R}}$$

where η is the unit of the adjunction $f^* \dashv f_*$. This can either be verified directly, or by using the adjoint triangle theorem [Dub68]. In what follows we will use this morphism of monads to translate between different types of Q-nets.

Definition 29. Let

$$(-)\text{-}\mathsf{Net}\colon \mathsf{Law} \to \mathsf{Cat}$$

be the functor which sends a Lawvere theory Q to the category Q-Net and sends a morphism $f\colon \mathsf{Q} \to \mathsf{R}$ of Lawvere theories to the functor f-$\mathsf{Net}\colon \mathsf{Q}\text{-}\mathsf{Net} \to \mathsf{R}\text{-}\mathsf{Net}$ which sends a Q-net

$$T \underset{t}{\overset{s}{\rightrightarrows}} M_{\mathsf{Q}} S$$

to the R-net

$$T \underset{t}{\overset{s}{\rightrightarrows}} M_{\mathsf{Q}} S \xrightarrow{M^f_S} M_{\mathsf{R}} S$$

For a morphism of Q-nets $(g\colon T \to T', h\colon S \to S')$, $f\text{-}\mathsf{Net}(g,h)$ is (g,h). This is well-defined because of the naturality of M^f.

Varying the Lawvere theory Q gives many known variants of Petri nets.

Example 30. Setting Q equal to CMON, the Lawvere theory for commutative monoids, we obtain the category of Petri nets.

Definition 31. Let $(-)^*\colon \mathsf{Set} \to \mathsf{Set}$ denote the monad that the Lawvere theory MON induces via the correspondence in [EJL66]. For a set X, X^* is given by the underlying set of the free monoid on X. A **pre-net** is a pair of functions of the form

$$T \underset{t}{\overset{s}{\rightrightarrows}} S^*$$

A morphism of pre-nets from a pre-net $(s,t\colon T \to S^*)$ to a pre-net $(s',t'\colon T' \to S'^*)$ is a pair of functions $(f\colon T \to T', g\colon S \to S')$ which preserves the source and target as in Definition 23. This defines a category PreNet.

Example 32. If we take $\mathsf{Q} = \mathsf{MON}$, the Lawvere theory of monoids, we get the category PreNet.

A description of MON can be found in the Appendix. PreNet has the same objects as the category introduced in [BMMS01] but the morphisms are restricted as in Definition 23. Pre-nets are the same as tensor schemes introduced by Joyal and Street in [JS91]. The authors define a notion of free category on a tensor scheme and Bruni, Meseguer, Montanari, and Sassone construct an adjunction between pre-nets and a subcategory of the category of strict symmetric monoidal categories SSMC [BMMS01].

In [Mas20d], a closely related adjunction

$$\mathsf{PreNet} \underset{K}{\overset{Z}{\rightleftarrows}} \mathsf{SSMC} \quad (\bot)$$

is constructed which does not require the restriction to a subcategory of SSMC. Pre-nets are useful because after forming an appropriate quotient, the category $Z(P)$ for a pre-net P is equivalent the category of *strongly concatenable processes* which can be performed on the net. This equivalence is important for realizing the individual token philosophy [BMMS01]. The individual token philosophy, as opposed to the collective token philosophy, gives identities to the individual tokens and keeps track of the causality in the executions of a Petri net.

Example 33. In 2013 Bartoletti, Cimoli, and Pinna introduced lending Petri nets [BCP15]. These are Petri nets where arcs can have a negative multiplicity and tokens can be borrowed in order to fire a transition. Lending nets are also equipped with a partial labeling of the places and transitions so they can be composed and are required to have no transitions which can be fired spontaneously. In 2018 Genovese and Herold introduced integer nets [GH18]. Let ABGRP be the Lawvere theory of abelian groups. This Lawvere theory contains three generating operations

$$e\colon 0 \to 1,\ i\colon 1 \to 1, \text{ and } m\colon 2 \to 1$$

representing the identies, inverses, and multiplication of an abelian group. These generating morphisms are required to satisfy the axioms of an abelian group; associativity, commutativity, and the existence of inverses and an identity. The category of integer nets, modulo a change in the definition of morphisms, can be obtained by taking $\mathsf{Q} = \mathsf{ABGRP}$ in the definition of Q-Net.

Definition 34. Let $\mathbb{Z}\colon \mathsf{Set} \to \mathsf{Set}$ be the free abelian group monad which for a set X generates the free abelian group $\mathbb{Z}[X]$ on the set X. Note that $\mathbb{Z}$ is the monad induced by the Lawvere theory ABGRP via the correspondence in [EJL66]. An **integer net** is a pair of functions of the form

$$T \overset{s}{\underset{t}{\rightrightarrows}} \mathbb{Z}[S].$$

A **morphism of integer nets** is a pair $(f\colon T \to T', g\colon S \to S')$ which makes the diagrams analogous to the definition of Petri net morphism (Definition 23) commute. Let $\mathbb{Z}\text{-}\mathsf{Net}$ be the category where objects are integer nets and morphism are morphisms of integer nets.

Integer nets are useful for modeling the concepts of credit and borrowing. There is a correspondence between lending Petri nets and propositional contract logic; a form of logic useful for ensuring that complex networks of contracts are honored [BCP15]. Genovese and Herold constructed a categorical semantics for integer nets [GH18]. In [Mas20d], the author constructed a variation of this semantics which uses the general framework developed in this chapter.

Example 35. Elementary net systems, introduced by Rozenberg and Thiagarajan in 1986, are are Petri nets with a maximum of one edge between a given place and transition [RT86].

Definition 36. An **elementary net system** is a pair of functions

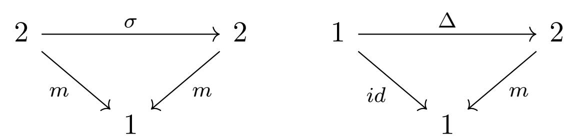
$$T \overset{s}{\underset{t}{\rightrightarrows}} 2^S$$

where 2^S denotes the power set of S.

Elementary net systems can be obtained from our general formalism. Let SLAT be the Lawvere theory for semi-lattices, i.e. commutative idempotent monoids. This Lawvere theory contains morphisms

$$m\colon 2 \to 1 \text{ and } e\colon 0 \to 1$$

as in MON the theory of monoids. Also similar to MON, SLAT is quotiented by the associativity and unitality axioms given in Example 108. In addition, SLAT has the following axioms representing commutativity and idempotence



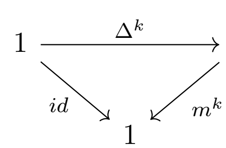
where $\sigma\colon 2 \to 2$ is the braiding of the cartesian product and $\Delta\colon 1 \to 2$ is the diagonal. For models in Set, The first diagram says that you can multiply two elements in either order and the get the same thing. The second diagram says that if you multiply an element by itself you get itself. As in Definition 110, SLAT corresponds to a monad on Set. It is well known that this monad is the covariant power set monad

$$2^{(-)}\colon \mathsf{Set} \to \mathsf{Set}$$

which sends a set X to its set of finite subsets and a function to the mapping which sends subsets of X to their image. This motivates the following:

Definition 37. Let SLAT-Net be the category of elementary net systems obtained as in Definition 3.2 for $\mathsf{Q} = \mathsf{SLAT}$.

Example 38. Generalizing the previous example, for any natural number k we form the Lawvere theory k for k-idempotent monoids. k has the same operations and axioms as SLAT except the idempotency axiom is replaced with the axiom



where Δ^k is the k-fold diagonal map and m^k is the k-fold multiplication map. Via the correspondence of Linton, k gives the k-powerset monad

$$k^{(-)}\colon \mathsf{Set} \to \mathsf{Set}$$

sending a set X to the set of finitely supported functions $\{X \to \{0, 1, 2, \ldots, k-1\} = k\}$ [EJL66]. For a function $f\colon X \to Y$, $k^f\colon k^X \to k^Y$ is defined by

$$k^f(X \xrightarrow{a} k)(y) = \sum_{x \in f^{-1}(y)} a(x)$$

i.e. the analog of direct image for k-multisets.

Definition 39. A ***k*-safe** net is a pair of functions

$$T \underset{t}{\overset{s}{\rightrightarrows}} k^S$$

A morphism of k-safe nets is a pair of functions

$$\begin{array}{ccc} T & \underset{t}{\overset{s}{\rightrightarrows}} & k^S \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle g} \\ T' & \underset{t'}{\overset{s'}{\rightrightarrows}} & k^{S'} \end{array}$$

commuting as in the previous definitions. This defines a category $\mathsf{k\text{-}Net}$ of k-safe nets and their morphisms.

The functoriality of Definition 3.2 can be exploited to generate functors between different categories of Q-nets. There is the diagram in Law

$$\begin{array}{ccc} \mathsf{SLAT} & & \\ {\scriptstyle a}\uparrow & & \\ \mathsf{CMON} & \xrightarrow{b} & \mathsf{ABGRP} \\ {\scriptstyle c}\uparrow & & \uparrow{\scriptstyle e} \\ \mathsf{MON} & \xrightarrow[d]{} & \mathsf{GRP} \end{array}$$

where all the morphisms send their generating operations to their counterparts in the Lawvere theory of their codomain. These target Lawvere theories either have extra axioms or operations making the above functors not necessarily full or faithful:

- a sends the morphism $m \circ \Delta$ in CMON to $id_1 \colon 1 \to 1$ in SLAT to impose the idempotent law. All other other generating components are sent to their natural counterparts.

- b and d send every object and morphism to its natural analog. However, ABGRP and GRP have an extra operation $i \colon 1 \to 1$ representing inverses. This makes the functors b and d faithful but not full.

- c and e add the commutativity law; they send both the multiplication $m \colon 2 \to 1$ and the composite $\sigma \circ m \colon 2 \to 2$ of the braiding $\sigma \colon 2 \to 2$ to the multiplication map $m \colon 2 \to 2$ in the target Lawvere theory. This makes c and e not faithful.

Definition 29 is used to give a network between different flavors of Q-nets. By applying $(-)$-Net to the above diagram we get the diagram of categories

$$\begin{array}{ccc} \mathsf{SLAT\text{-}Net} & & \\ \big\uparrow {\scriptstyle a\text{-}\mathsf{Net}} & & \\ \mathsf{Petri} & \xrightarrow{b\text{-}\mathsf{Net}} & \mathbb{Z}\text{-}\mathsf{Net} \\ \big\uparrow {\scriptstyle c\text{-}\mathsf{Net}} & & \big\uparrow {\scriptstyle e\text{-}\mathsf{Net}} \\ \mathsf{PreNet} & \xrightarrow[d\text{-}\mathsf{Net}]{} & \mathsf{GRP\text{-}Net} \end{array}$$

The functors in this diagram are described as follows:

$$c\text{-}\mathsf{Net} \colon \mathsf{PreNet} \to \mathsf{Petri}$$

is often called "abelianization" because it sends a pre-net to the Petri net which forgets about the ordering on the input and output of each transition. The authors of [BMMS01] use c-Net

to explore the relationship between pre-nets and Petri nets. The functor $e\text{-}\mathsf{Net}\colon \mathsf{GRP}\text{-}\mathsf{Net} \to \mathbb{Z}\text{-}\mathsf{Net}$ gives the analogous relationship for integer nets.

$$b\text{-}\mathsf{Net}\colon \mathsf{Petri} \to \mathbb{Z}\text{-}\mathsf{Net}$$

is the functor which does not change the source and target of a given place. The only difference is that the markings of a $\mathbb{Z}$-net coming from a Petri net are thought of as elements of a free abelian group rather than a free abelian monoid. d-Net is the analogous functor for pre-nets.

$$a\text{-}\mathsf{Net}\colon \mathsf{Petri} \to \mathsf{SLAT}\text{-}\mathsf{Net}$$

is the functor which sends a Petri net to the SLAT-net which forgets about the multiplicity of the edges between a given source and transition. Before moving on to the semantics of Q-nets, we discuss a property of the category Q-Net.

Proposition 40. Q-Net *is cocomplete.*

Proof. We can construct Q-Net as the comma category.

$$\mathsf{Set} \downarrow (\times \circ \Delta \circ M_{\mathsf{Q}})$$

where $M_{\mathsf{Q}}\colon \mathsf{Set} \to \mathsf{Set}$ is the monad corresponding to the Lawvere theory Q, $\Delta\colon \mathsf{Set} \to \mathsf{Set} \times \mathsf{Set}$ is the diagonal, and $\times\colon \mathsf{Set} \times \mathsf{Set} \to \mathsf{Set}$ is the cartesian product in Set. An object in this category is a map

$$T \to M_{\mathsf{Q}} S \times M_{\mathsf{Q}} S$$

which corresponds to a pair of maps $s, t\colon T \to M_{\mathsf{Q}} S$ which become the source and target maps of a Q-net. Morphisms in this comma category are commutative squares

$$\begin{array}{ccc} T & \longrightarrow & M_{\mathsf{Q}}S \times M_{\mathsf{Q}}S \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle M_{\mathsf{Q}}g \times M_{\mathsf{Q}}g} \\ T' & \longrightarrow & M_{\mathsf{Q}}S' \times M_{\mathsf{Q}}S' \end{array}$$

giving a map of Q-nets $(f\colon T \to T', g\colon S \to S')$. The commutativity of the above square ensures that this map of Q-nets is well-defined.

Theorem 3, Section 5.2 of *Computational Category Theory* [RB88] says that given $S\colon A \to C$ and $T\colon B \to C$ then the comma category $(S \downarrow T)$ is cocomplete if

- S is cocontinuous, and
- A and B are cocomplete,

Because Set is cocomplete and the identity functor $1_{\mathsf{Set}}\colon \mathsf{Set} \to \mathsf{Set}$ preserves all colimits, we have that $\mathsf{Set} \downarrow (\times \circ \Delta \circ M_{\mathsf{Q}})$ is cocomplete. Because $\mathsf{Q}\text{-}\mathsf{Net}$ is equivalent to this category, it is cocomplete as well. □

3.3 Generating Free Commutative Monoidal Categories

In this section we examine in detail the motivating example for the main result of this chapter, an adjunction generating the semantics of Q-nets for every Lawvere theory Q. This result can feel abstract on its own and the example of Petri nets provides invaluable intuition. A confident reader may skip this section, as it is not strictly necessary for the rest of the chapter.

The operational semantics for Petri nets will take the form of an adjunction

$$\mathsf{Petri} \underset{U}{\overset{F}{\rightleftarrows}} \mathsf{CMC} \quad (F \dashv U)$$

For a given Petri net P, this adjunction will be constructed in two steps: first the transitions of P will be freely closed under a commutative monoidal sum and then freely closed under composition. This will take the form of factoring the adjunction into the composite

$$\mathsf{Petri} \underset{\mathrm{A}^\bullet}{\overset{{}^\bullet\mathrm{A}}{\rightleftarrows}}_{\bot} \mathsf{Grph(CommMon)} \underset{\mathrm{B}^\bullet}{\overset{{}^\bullet\mathrm{B}}{\rightleftarrows}}_{\bot} \mathsf{CMC}.$$

Here a left adjoint is indicated by a bullet on the left and a right adjoint is indicated by a bullet on the right. $\mathsf{Grph(CommMon)}$ is the category of graphs internal to $\mathsf{CommMon}$.

Definition 41. A **commutative monoidal graph** is a graph

$$E \underset{t}{\overset{s}{\rightrightarrows}} V$$

where E and V are commutative monoids and s and t are commutative monoid homomorphisms. A morphism of commutative monoidal graphs is a tuple of commutative monoid homomorphisms $(f\colon E \to E', g\colon V \to V')$ making the diagrams

$$\begin{array}{ccc} E & \xrightarrow{s} & V \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle g} \\ E' & \xrightarrow[s']{} & V' \end{array} \qquad \begin{array}{ccc} E & \xrightarrow{t} & V \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle g} \\ E' & \xrightarrow[t']{} & V' \end{array}$$

commute. This defines a category $\mathsf{Grph(CommMon)}$ where objects are commutative monoidal graphs and morphisms are as above. In short, $\mathsf{Grph(CommMon)}$ is the category of graphs internal to $\mathsf{CommMon}$.

We will now define these adjunctions but omit the proofs that they are indeed well-defined adjunctions, as this follows from the more general results of Section 3.4. The left adjoint ${}^\bullet\mathrm{A}\colon \mathsf{Petri} \to \mathsf{Grph(CommMon)}$ is defined as follows:

Definition 42. Let

$$^{\bullet}\mathrm{A}\colon \mathsf{Petri} \to \mathsf{Grph}(\mathsf{CommMon})$$

be the functor which sends a Petri net

$$P = T \underset{t}{\overset{s}{\rightrightarrows}} \mathbb{N}[S]$$

to the commutative monoidal graph

$$^{\bullet}\mathrm{A}P = LT \underset{\phi^{-1}(t)}{\overset{\phi^{-1}(s)}{\rightrightarrows}} LS$$

where L is the left adjoint of the adjunction in Definition 21 and $\phi\colon \mathrm{Hom}(LT, LS) \xrightarrow{\sim} \mathrm{Hom}(T, RLS)$ is the natural isomorphism of that adjunction. $^{\bullet}\mathrm{A}$ sends a morphism of Petri nets

$$(f\colon T \to T', g\colon S \to S')$$

to the morphism of commutative monoidal graphs given by

$$(Lf\colon LT \to LT', Lg\colon LS \to LS')$$

In words, $^{\bullet}\mathrm{A}$ freely generates a commutative monoidal structure on the transitions of a Petri net and $^{\bullet}\mathrm{A}$ uniquely extends each component of a Petri net morphism to a commutative monoid homomorphism. The right adjoint of this functor is non-trivial:

Definition 43. Let

$$\mathrm{A}^{\bullet}\colon \mathsf{Grph}(\mathsf{CommMon}) \to \mathsf{Petri}$$

be the functor which sends a commutative monoidal graph

$$Q = E \underset{t}{\overset{s}{\rightrightarrows}} V$$

to the Petri net

$$\mathrm{A}^\bullet Q = \bar{E} \underset{\bar{t}}{\overset{\bar{s}}{\rightrightarrows}} \mathbb{N}[RV]$$

$\bar{E}$ is defined as

$$\bar{E} = \{(e, x, y) \in RE \times \mathbb{N}[RV] \times \mathbb{N}[RV] | R\epsilon_V(x) = s(e) \text{ and } R\epsilon_V(y) = t(e)\}$$

where ϵ_V is the counit of the adjunction $L \dashv R$. $\bar{s}$ and $\bar{t}$ are given by the projection of $\bar{E}$ onto its first and second coordinates respectively. $\mathrm{A}^\bullet$ sends a morphism of commutative monoidal graphs

$$(f \colon E \to E', g \colon V \to V')$$

to the morphism of Petri nets

$$(h \colon \bar{E} \to \bar{E}', Rg \colon RV \to RV')$$

where h is the function which makes the assignment

$$(e, x, y) \mapsto (\phi(f)(e), \mathbb{N}[Rg](x), \mathbb{N}[Rg](y))$$

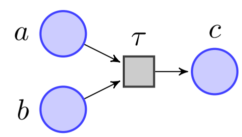
Remark. Petri nets must have a free commutative monoid of places, so it is necessary to regard RV as the set of places for $\mathrm{A}^\bullet Q$ rather than having V be the commutative monoid of places itself. The reader at this point may guess a simpler formula for the right adjoint $\mathrm{A}^\bullet$ which keeps the RE as the set of transitions and uses the unit of $\mathbb{N}$ to construct the source and target maps. Unfortunately this construction is doomed to fail. For a commutative monoidal graph $E \underset{t}{\overset{s}{\rightrightarrows}} V$, suppose that the right adjoint ${}^\bullet\mathrm{A}$ sends this graph to the Petri net

$$RE \underset{\eta \circ Rt}{\overset{\eta \circ Rs}{\rightrightarrows}} \mathbb{N}[RV].$$

A problem arises because this process unnaturally chunks the source and target of each transition. To see this consider the commutative monoidal graph

$$Q = \mathbb{N}[\tau] \rightrightarrows \mathbb{N}[\{a, b, c\}] \cong \mathbb{N}^3$$

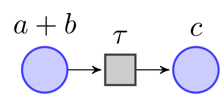
The edge τ in Q can be depicted as



With the above (faulty) description, $\mathrm{A}^\bullet Q$ is given by

$$\mathbb{N}[\tau] \rightrightarrows \mathbb{N}[\mathbb{N}^3]$$

To avoid confusion, we denote the outer sum in $\mathbb{N}[\mathbb{N}^3]$ by $\times$ and the sum in $\mathbb{N}^3$ by $+$. Then, the faulty $\mathrm{A}^\bullet$ would turn τ into the transition



To find a counit for this adjunction we seek a morphism

$$^\bullet\mathrm{A}\mathrm{A}^\bullet Q \to Q$$

A morphism of this sort is defined by its assignment on generators. A natural choice of morphism sends the places $a+b$ to the sum of the places a and b using the counit of $L \dashv R$. However, then the assignment of $\tau \mapsto \tau$ does not respect the source of τ and is therefore not a morphism of commutative monoidal graphs. The problem is that we want the source of τ in $^\bullet\mathrm{A}\mathrm{A}^\bullet Q$ to be $a \times b$ and not $a + b$. To fix this we force the source of τ to be $a \times b$ by upgrading τ to the tuple $(\tau, a \times b, c)$ in $\mathbb{N}[\tau] \times \mathbb{N}[\mathbb{N}^3] \times \mathbb{N}[\mathbb{N}^3]$. Now, the natural choice for the counit which sends $a \times b$ to $a + b$ respects the source of τ.

The next part of the semantics adjunction for Petri nets freely generates the structure of a category on a given commutative monoidal graph. In Section 3.4 this is accomplished by rephrasing this construction in terms of free monoids. Here we provide an explicit description in the case of Petri nets.

Definition 44. Let

$$\mathrm{B}^\bullet \colon \mathsf{CMC} \to \mathsf{Grph}(\mathsf{CommMon})$$

be the forgetful functor which sends a commutative monoidal category to its underlying commutative monoidal graph and a strict monoidal functor to its underlying morphism of commutative monoidal graphs. Then $\mathrm{B}^\bullet$ has a left adjoint

$$^\bullet\mathrm{B} \colon \mathsf{Grph}(\mathsf{CommMon}) \to \mathsf{CMC}$$

which sends a commutative monoidal graph

$$Q = E \overset{s}{\underset{t}{\rightrightarrows}} V$$

to the commutative monoidal category $^\bullet\mathrm{B}Q$ with objects given by V and morphisms generated inductively by the rules:

- for every edge $e \in E$ a morphism $e \colon s(e) \to t(e)$,
- for every pair of morphisms $e \colon x \to y$ and $d \colon y \to z$, a morphism $d \circ e \colon x \to z$,
- for every object $v \in V$ a morphism $1_v \colon v \to v$

This defines an evident composition operation $\circ$ on $^\bullet\mathrm{B}Q$. There is also a sum on the $^\bullet\mathrm{B}Q$ defined using the sum of V on objects. If e and e' are edges of Q then the morphisms

$e\colon x \to y$ and $e'\colon x' \to y'$ already have a sum given by

$$e + e'\colon x + x' \to y + y'$$

The morphisms of ${}^\bullet\mathrm{B}Q$ are quotiented by the relations:

- for all tuples of morphisms $(f\colon a \to b, g\colon b \to c, h\colon c \to d)$

$$(f \circ g) \circ h = f \circ (g \circ h)$$

- for all morphisms $f\colon x \to y$

$$1_y \circ f = f = f \circ 1_x$$

- We require that composition is a commutative monoid homomorphism. For tuples of morphisms $(e\colon x \to y, d\colon y \to z, e'\colon x' \to y', d'\colon y' \to z')$ we can form their sum and composite in two different ways. We quotient the morphisms of ${}^\bullet\mathrm{B}Q$ so that these are equal, i.e.

$$(d \circ e) + (d' \circ e') = (d + d') \circ (e + e')$$

- We require that the assignment of identities is a commutative monoid homomorphism. For objects x and x' in V we set

$$1_{x+x'} = 1_x + 1'_x$$

3.4 Semantics Functors for Generalized Nets

In this section we state our construction of semantics categories for Q-nets; categories whose morphisms represent possible sequences of firings which can be performed using a given Q-net. Let Q be a Lawvere theory and let

$$\mathsf{Set} \underset{R}{\overset{L}{\rightleftarrows}} \mathsf{Mod}(\mathsf{Q}, \mathsf{Set}) \quad (\bot)$$

be the adjunction it induces on Set. In this section we will use this adjunction to construct an adjunction

$$\mathsf{Q}\text{-}\mathsf{Net} \underset{U_{\mathsf{Q}}}{\overset{F_{\mathsf{Q}}}{\rightleftarrows}} \mathsf{Mod}(\mathsf{Q}, \mathsf{Cat}) \quad (\bot)$$

which is analogous to the adjunction in Section 3.3 and where $\mathsf{Mod}(\mathsf{Q}, \mathsf{Cat})$ is the category of models of Q in Cat. This adjunction factors as

$$\mathsf{Q}\text{-}\mathsf{Net} \underset{\mathrm{A}^{\bullet}_{\mathsf{Q}}}{\overset{{}^{\bullet}\mathrm{A}_{\mathsf{Q}}}{\rightleftarrows}} \mathsf{Grph}(\mathsf{Mod}(\mathsf{Q})) \underset{\mathrm{B}^{\bullet}_{\mathsf{Q}}}{\overset{{}^{\bullet}\mathrm{B}_{\mathsf{Q}}}{\rightleftarrows}} \mathsf{Mod}(\mathsf{Q}, \mathsf{Cat})$$

where ${}^{\bullet}\mathrm{A}_{\mathsf{Q}}$ freely generates a model of Q on the transitions of a given Q-Net and ${}^{\bullet}\mathrm{B}_{\mathsf{Q}}$ freely generates the structure of a category on a given Q-graph.

These adjunctions are heavily motivated by the case when $\mathsf{Q} = \mathsf{CMON}$ as this gives Petri nets. The main result of this chapter is as follows:

Theorem 45. *There is an adjunction*

$$\mathsf{Q}\text{-}\mathsf{Net} \underset{U_{\mathsf{Q}}}{\overset{F_{\mathsf{Q}}}{\rightleftarrows}} \mathsf{Mod}(\mathsf{Q}, \mathsf{Cat}). \quad (\bot)$$

The left adjoint can be described using inference rules. Let P be the Q-net

$$T \underset{t}{\overset{s}{\rightrightarrows}} M_{\mathsf{Q}}S$$

The objects of F_{Q} are given by $L_{\mathsf{Q}}S$. That is, for every morphism $o\colon n \to m$ in Q and every tuple of places $x_1, x_2, \ldots, x_n$ there is an object $\mathbf{o}(x_1, x_2, x_3, \ldots, x_n)$. For an equation of morphisms in Q

$$\begin{array}{ccc} & n & \\ {\scriptstyle f}\swarrow & & \searrow{\scriptstyle h} \\ m & \xrightarrow[g]{} & k \end{array}$$

the objects generated by each path must be equal. This means that there are k equations of objects

$$\mathbf{g_j}(\mathbf{f_1}(x_1, x_2, \ldots, x_n), \mathbf{f_2}(x_1, x_2, \ldots, x_n), \ldots, \mathbf{f_m}(x_1, x_2, \ldots, x_n)) = \mathbf{h_j}(x_1, x_2, \ldots, x_n)$$

where the unlabeled index runs over the components of f and the index j runs over the components of g and h. The morphisms of $F_Q P$ are generated inductively by the rules

$$\frac{\tau \in T}{\tau\colon s(\tau) \to t(\tau) \in \operatorname{Mor} F_{\mathsf{Q}}P} \qquad \frac{x \in \operatorname{Ob} F_{\mathsf{Q}}P}{1_x\colon x \to x \in \operatorname{Mor} F_{\mathsf{Q}}P} \qquad \frac{f\colon x \to y \text{ and } g\colon y \to z \in \operatorname{Mor} F_{\mathsf{Q}}P}{g \circ f\colon x \to z \in \operatorname{Mor} F_{\mathsf{Q}}P}$$

$$\frac{o\colon n \to 1 \in \operatorname{Mor}\mathsf{Q} \text{ and } f_1\colon x_1 \to y_1, f_2\colon x_2 \to y_2, \ldots f_n\colon x_n \to y_n \in \operatorname{Mor} F_{\mathsf{Q}}P}{\mathbf{o}(f_1, f_2, \ldots, f_n)\colon \mathbf{o}(x_1, x_2, \ldots, x_n) \to \mathbf{o}(y_1, y_2, \ldots, y_n)}$$

and is quotiented to satisfy the following:

- The morphisms must satisfy the same equations that the objects satisfy. That is, for an equation of morphisms in Q, the objects generated by each path must again be equal.

- $\operatorname{Mor} F_{\mathsf{Q}}P$ is quotiented to satisfy the axioms of a category including the associative and unital laws

$$(f \circ g) \circ h = f \circ (g \circ h) \text{ and } 1_y \circ f = f = f \circ 1_x$$

  for all morphisms f, g and h in $\operatorname{Mor} F_{\mathsf{Q}}P$.

- $\operatorname{Mor} F_{\mathsf{Q}} P$ is quotiented so that the structure maps of a category (source, target, identity and composition) are Q-model homomorphisms.

For a morphism of Q-nets, $(f,g)\colon P \to P'$, the Q-functor

$$F_{\mathsf{Q}}(f,g)\colon F_{\mathsf{Q}}P \to F_{\mathsf{Q}}P'$$

is the unique extension of f and g which respects composition, unitality, and the operations of Q. The proof will require several lemmas. The first step is to show how Q-nets freely generate graphs internal to the category of models of Q.

3.5 Q-nets freely generate Q-graphs

In this section we show how Q-nets freely generate graphs internal to $\mathsf{Mod}(\mathsf{Q})$. In the following proofs we will write the monad M_{Q} as RL and make use of the natural isomorphism

$$\phi\colon \hom(LX, Y) \xrightarrow{\sim} \hom(X, RY)$$

for all sets X and objects Y in $\mathsf{Mod}(\mathsf{Q})$.

Definition 46. Let

$$^{\bullet}\mathrm{A}_{\mathsf{Q}}\colon \mathsf{Q}\text{-}\mathsf{Net} \to \mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))$$

be the functor which makes the assignment

$$\begin{array}{ccc} T & \overset{s}{\underset{t}{\rightrightarrows}} & RLS \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle RLg} \\ T' & \overset{s'}{\underset{t'}{\rightrightarrows}} & RLS' \end{array} \quad\longmapsto\quad \begin{array}{ccc} LT & \overset{\phi^{-1}(s)}{\underset{\phi^{-1}(t)}{\rightrightarrows}} & LS \\ {\scriptstyle Lf}\downarrow & & \downarrow{\scriptstyle Lg} \\ LT' & \overset{\phi^{-1}(s')}{\underset{\phi^{-1}(t')}{\rightrightarrows}} & LS' \end{array}$$

on objects and morphisms.

Lemma 47. ${}^{\bullet}\mathrm{A}_{\mathsf{Q}}$ *is well-defined.*

The next few proofs will make heavy use of the naturality equations for ϕ and its inverse:

$$\phi(a \circ b \circ Lc) = Ra \circ \phi(b) \circ c$$

and

$$\phi^{-1}(Ra \circ b \circ c) = a \circ \phi^{-1}(b) \circ Lc.$$

Proof. First we show that Le commutes with the source of ${}^{\bullet}\mathrm{A}_{\mathsf{Q}}P$. This follows from the chain of equalities:

$$\begin{aligned}
&\phi^{-1}(s') \circ Lf \\
&= \phi^{-1}(s' \circ f) \\
&= \phi^{-1}(RLg \circ s) \\
&= Lg \circ \phi^{-1}(s).
\end{aligned}$$

A similar equation holds for the target maps. □

Let G be the Q-graph

$$E \overset{s}{\underset{t}{\rightrightarrows}} V.$$

Because V is not a free model of Q, there is no obvious forgetful way to turn this into a Q-net. A first guess for the Q-net $\mathrm{A}^{\bullet}_{\mathsf{Q}}(G)$ might be the Q-net

$$RE \overset{\eta_{RV} \circ s}{\underset{\eta_{RV} \circ t}{\rightrightarrows}} M_{\mathsf{Q}}(RV)$$

where η_{RV} is the unit of the monad M_{Q} applied to the set RV. However, as explained in Remark 3.3, this fails to be a right adjoint. An alternative approach was suggested by Mike

Shulman in the comments of an nCafé blog post [Mas19]. This solution was inspired by the construction of the free category on a tensor scheme introduced in *The Geometry of Tensor Calculus I* [JS91]. Instead of using RE as the set of transitions, we use

$$\bar{E} = \{(e, x, y) \in RE \times M_{\mathsf{Q}}RV \times M_{\mathsf{Q}}RV \mid s(e) = R\epsilon_V(x) \text{ and } t(e) = R\epsilon_V(y)\}$$

where ϵ_V is the V component of the counit for M_{Q}. Here and in what follows we are using s to denote Rs and t to denote Rt for notational simplicity. The source and target maps of the resulting Q-net are given by the projections of $\bar{E}$ onto its second and third coordinates. The set $\bar{E}$ can be described formally using pullbacks.

Definition 48. Let

$$\mathrm{A}^{\bullet}_{\mathsf{Q}} \colon \mathsf{Grph}(\mathsf{Q}) \to \mathsf{Q}\text{-}\mathsf{Net}$$

be the functor which makes the assignment on objects and morphisms

$$\begin{array}{ccc} E \overset{s}{\underset{t}{\rightrightarrows}} V & & \bar{E} \overset{\bar{s}}{\underset{\bar{t}}{\rightrightarrows}} M_{\mathsf{Q}}RV \\ f\downarrow \quad\quad \downarrow g & \longmapsto & \bar{f}\downarrow \quad\quad\quad \downarrow M_{\mathsf{Q}}Rg \\ E \overset{s}{\underset{t}{\rightrightarrows}} V & & \bar{E}' \overset{\bar{s}'}{\underset{\bar{t}'}{\rightrightarrows}} M_{\mathsf{Q}}RV' \end{array}$$

where

- $\bar{E}$ is the pullback of sets

$$\begin{array}{ccccc} & & \bar{E} & & \\ & \overset{i}{\swarrow} & & \overset{j}{\searrow} & \\ RE & & & & M_{\mathsf{Q}}RV \times M_{\mathsf{Q}}RV \\ & \underset{(s,t)}{\searrow} & & \underset{R\epsilon_V \times R\epsilon_V}{\swarrow} & \\ & & RV \times RV & & \end{array}$$

  where (s, t) denotes the pairing of s and t, and $\epsilon_{RV} \times \epsilon_{RV}$ denotes the cartesian product of the counits.

- $\bar{s}\colon \bar{E} \to M_{\mathsf{Q}}RV$ is the composite

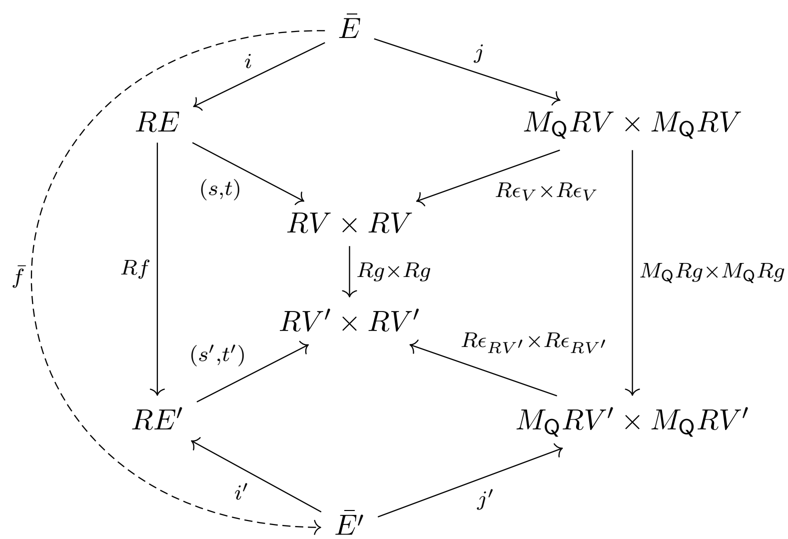
$$\bar{E} \xrightarrow{j} M_{\mathsf{Q}}RV \times M_{\mathsf{Q}}RV \xrightarrow{\pi_1} M_Q RV$$

and $\bar{t}\colon \bar{E} \to M_{\mathsf{Q}}RV$ is the composite

$$\bar{E} \xrightarrow{j} M_{\mathsf{Q}}RV \times M_{\mathsf{Q}}RV \xrightarrow{\pi_2} M_Q RV$$

that is the maps which send an element (e, x, y) of $\bar{E}$ to its second and third coordinates.

- $\bar{f}\colon \bar{E} \to \bar{E}'$ is induced by the universal property of $\bar{E}$ as shown below



More simply, $\bar{f}$ makes the assignment

$$(e, x, y) \mapsto (Rf(e), M_{\mathsf{Q}}Rg(x), M_{\mathsf{Q}}Rg(y))$$

Lemma 49. $\mathrm{A}^{\bullet}_{\mathsf{Q}}$ *is well-defined.*

Proof. We must show that $(\bar{f}, Rg)$ is a well-defined morphism of Q-nets. $\bar{f}$ and Rg commute with the source and target maps. Indeed, using the elementary descriptions we get that

$$\begin{aligned}\bar{s'} \circ \bar{f}(\tau, x, y) &= \bar{s}(Rf(\tau), M_{\mathsf{Q}}Rg(x), M_{\mathsf{Q}}Rg(y)) \\ &= M_{\mathsf{Q}}Rg(x) \\ &= M_{\mathsf{Q}}Rg(\bar{s}(\tau, x, y))\end{aligned}$$

A similar equation holds for the target maps. $(\bar{f}, Rg)$ commutes with the identity maps:

$$\begin{aligned}\bar{f} \circ \bar{e}(x) &= \bar{f}(Re(x), \eta_{RV}(x), \eta_{RV}(x)) \\ &= (Rf \circ Re(x), M_{\mathsf{Q}}Rg \circ \eta_{RV}(x), M_{\mathsf{Q}}Rg \circ \eta_{RV}(x)) \\ &= (Re' \circ Rg(x), M_{\mathsf{Q}}Rg \circ \eta_{RV}(x), M_{\mathsf{Q}}Rg \circ \eta_{RV}(x)) \\ &= (Re' \circ Rg(x), \eta_{RV'} \circ Rg(x), \eta_{RV'} \circ Rg(x)) \\ &= \bar{e'} \circ Rg(x)\end{aligned}$$

where the last two steps follow from naturality of η and (f, g) being a morphism of Q-graphs. □

Lemma 50. $\mathrm{A}^{\bullet}_{\mathsf{Q}}$ *is a right adjoint to* ${}^{\bullet}\mathrm{A}_{\mathsf{Q}}$*.*

Proof. Let P be the Q-net

$$T \underset{t}{\overset{s}{\rightrightarrows}} RLS$$

and Q be the Q-graph

$$E \underset{t'}{\overset{s'}{\rightrightarrows}} V$$

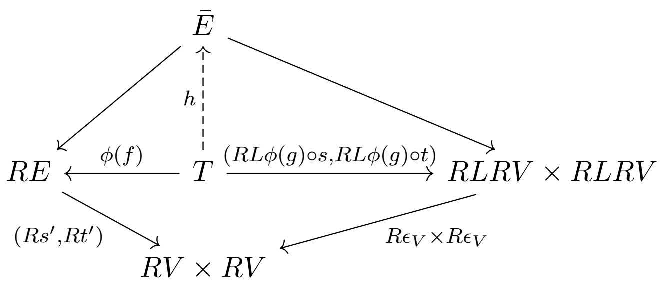
We define a natural isomorphism

$$\Phi \colon \mathrm{Hom}({}^{\bullet}\mathrm{B}P, Q) \xrightarrow{\sim} \mathrm{Hom}(P, \mathrm{B}^{\bullet}_{\mathsf{Q}}Q)$$

by the rule

$$\begin{array}{ccc} LT & \overset{\phi^{-1}(s)}{\underset{\phi^{-1}(t)}{\rightrightarrows}} & LS \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle g} \\ E & \overset{s'}{\underset{t'}{\rightrightarrows}} & V \end{array} \quad \longmapsto \quad \begin{array}{ccc} T & \overset{s}{\underset{t}{\rightrightarrows}} & RLS \\ {\scriptstyle h}\downarrow & & \downarrow{\scriptstyle RL\phi(g)} \\ \bar{E} & \overset{\bar{s'}}{\underset{\bar{t'}}{\rightrightarrows}} & RLRV \end{array}$$

h is defined by the universal property induced by $\bar{E}$ and the diagram



This diagram is well defined because T is a competitor to the pullback $\bar{E}$ i.e. it makes the lowest triangle commute. Checking this amounts to showing that the bottom square commutes and this can be verified componentwise:

$$\begin{aligned} R\epsilon_V \circ RL\phi(g) \circ s &= R(\epsilon_V \circ L\phi(g)) \circ s \\ &= R(\phi^{-1}(1_R V) \circ L\phi(g)) \circ s \\ &= R(\phi^{-1}(1_{RV} \circ \phi(g))) \circ s \\ &= R(\phi^{-1}(\phi(g))) \circ s \\ &= Rg \circ s \\ &= Rg \circ \phi(\phi^{-1}(s)) \\ &= \phi(g \circ \phi^{-1}(s)) \\ &= \phi(s' \circ f) \\ &= Rs' \circ \phi(f) \end{aligned}$$

and similar equations hold for the target maps. Therefore, h is well defined. Explicitly h is the map which makes the assignment on transitions in T

$$\tau \mapsto (\phi(f)(\tau), RL\phi(g) \circ s(\tau), RL\phi(g) \circ t(\tau)).$$

$(h, \phi(g))$ is a well-defined morphism of Q-graphs by construction. The source and target functions map elements to their second and third coordinates so the equation

$$\bar{s'} \circ h = RL\phi(g) \circ s$$

is true.

An inverse to Φ,

$$\Phi^{-1} \colon \mathrm{Hom}(P, \mathrm{B}^{\bullet}_{\mathsf{Q}} Q) \to \mathrm{Hom}({}^{\bullet}\mathrm{B}_{\mathsf{Q}} P, Q),$$

is defined as follows

$$\begin{array}{ccc} T & \overset{s}{\underset{t}{\rightrightarrows}} & RLS \\ {\scriptstyle h}\downarrow & & \downarrow{\scriptstyle RLg} \\ \bar{E} & \overset{\bar{s'}}{\underset{\bar{t'}}{\rightrightarrows}} & RLRV \end{array} \quad \longmapsto \quad \begin{array}{ccc} LT & \overset{\phi^{-1}(s)}{\underset{\phi^{-1}(t)}{\rightrightarrows}} & LS \\ {\scriptstyle \phi^{-1}(a)}\downarrow & & \downarrow{\scriptstyle \phi^{-1}(g)} \\ E & \overset{s'}{\underset{t'}{\rightrightarrows}} & V \end{array}$$

a is defined by the universal property of $\bar{E}$ and the diagram

$$\begin{array}{ccccc} & \overset{i}{\swarrow} & \bar{E} & \overset{j}{\searrow} & \\ & & {\scriptstyle h}\uparrow & & \\ RE & \overset{a}{\dashleftarrow} & T & \overset{b}{\dashrightarrow} & RLRV \times RLRV \\ & {\scriptstyle (Rs',Rt')}\searrow & & \swarrow{\scriptstyle R\epsilon_V \times R\epsilon_V} & \\ & & RV \times RV & & \end{array}$$

To show that $(\phi^{-1}(a), \phi^{-1}(g))$ is a well defined morphism of Q-graphs we perform the computation:

$$\begin{aligned} s' \circ \phi^{-1}(a) &= \phi^{-1}(Rs' \circ a) \\ &= \phi^{-1}(R\epsilon_V \circ \pi_1 \circ b) \\ &= \phi^{-1}(R\epsilon_V \circ \bar{s'} \circ h) \end{aligned}$$

where $\pi_1 \colon RLRV \times RLRV \to RLRV$ is the projection and the last two steps follow from the definition of $\bar{s'}$ and commutativity of the above diagram. This can be reduced using the fact that h commutes with the source and target of P and $\mathrm{B}^{\bullet}_{\mathsf{Q}}Q$ and naturality of ϕ^{-1}. Indeed,

$$
\begin{aligned}
\phi^{-1}(R\epsilon_V \circ \bar{s'} \circ h) &= \phi^{-1}(R\epsilon_V \circ RLg \circ s) \\
&= \phi^{-1}(R(\epsilon_V \circ Lg) \circ s) \\
&= \epsilon_V \circ Lg \circ \phi^{-1}(s) \\
&= \phi^{-1}(1_{RV}) \circ Lg \circ \phi^{-1}(s) \\
&= \phi^{-1}(1_{RV} \circ g) \circ \phi^{-1}(s) \\
&= \phi^{-1}(g) \circ \phi^{-1}(s)
\end{aligned}
$$

A similar equation holds for target so this is a well-defined morphism of Q-graphs. Φ is a natural isomorphism if it is a natural and a bijection in the places component and the transitions component. The places component is only an application of ϕ so it is both natural and a bijection. For the transition component let $D \colon C \to \mathsf{Set}$ be the diagram

$$
RE \xrightarrow{(Rs,Rt)} RV \times RV \xleftarrow{R\epsilon_V \times R\epsilon_V} M_{\mathsf{Q}}RV \times M_{\mathsf{Q}}RV
$$

where C is the walking cospan. Let $\Delta_T \colon C \to \mathsf{Set}$ be the constant diagram which sends every object to T and every morphism to 1_T. Then the universal property of $\bar{E}$ can be expressed as the natural isomorphism

$$
\Psi \colon \mathrm{Nat}(\Delta_T, D) \xrightarrow{\sim} \mathrm{Hom}(T, \bar{E})
$$

where $\mathrm{Nat}(\Delta_T, D)$ denotes the set of natural transformations from Δ_T to D. With this description, the transition component of Φ can be described as follows

$$\Phi\colon\ f \ \mapsto\ \Psi(\langle\phi(f), (RL\phi(g)\circ s, RL\phi(g)\circ t)\rangle)$$

where the angle brackets encase the components of a natural transformation. Similarly, the transition component of Φ^{-1} can be described as

$$\Phi^{-1}\colon\ h \ \mapsto\ \phi^{-1}(\Psi^{-1}(h)_{RE})$$

where the subscript RE indicates that we take the RE component of the natural transformation. With this description, we can verify that they are inverses on the transition component:

$$\begin{aligned}
f &\mapsto \Psi(\langle\phi(f), (RL\phi(g)\circ s, RL\phi(g)\circ t)\rangle) \\
&\mapsto \phi^{-1}(\Psi^{-1}(\Psi(\langle\phi(f), (RL\phi(g)\circ s, RL\phi(g)\circ t)\rangle))_{RE}) \\
&= \phi^{-1}(\langle\phi(f), (RL\phi(g)\circ s, RL\phi(g)\circ t)\rangle_{RE}) \\
&= \phi^{-1}(\phi(f)) \\
&= f
\end{aligned}$$

and the other direction:

$$\begin{aligned}
h &\mapsto \phi^{-1}(\Psi^{-1}(h)_{RE}) \\
&\mapsto \Psi(\langle\phi(\phi^{-1}(\Psi^{-1}(h)_{RE})), (RL\phi(g)\circ s, RL\phi(g)\circ t)\rangle) \\
&= \Psi\Psi^{-1}\langle h, (RL\phi(g)\circ s, RL\phi(g)\circ t\rangle_{RE} \\
&= \langle h, (RL\phi(g)\circ s, RL\phi(g)\circ t\rangle_{RE} \\
&= h.
\end{aligned}$$

The transition component of Φ and Φ^{-1} are natural because they are made up of components which are individually natural transformations. $\square$

The next step in the proof of Theorem 45 is to construct an adjunction between $\mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))$ and $\mathsf{Mod}(\mathsf{Q}, \mathsf{Cat})$, i.e. the free graph construction internal to $\mathsf{Mod}(\mathsf{Q})$.

3.6 Free Categories Internal to $\mathsf{Mod}(\mathsf{Q})$

In this section we will construct an adjunction

$$\mathsf{Mod}(\mathsf{Q}, \mathsf{Grph}) \underset{U_{\mathsf{Q}}}{\overset{F_{\mathsf{Q}}}{\rightleftarrows}} \mathsf{Mod}(\mathsf{Q}, \mathsf{Cat})$$

to complete the proof of Theorem 45. A general property of algebraic theories P and Q is that models of P in the category of models of Q are the same as models of Q in the category of models of P. In particular for a Lawvere theory Q, a model of Q in Cat is the same as a category internal to $\mathsf{Mod}(\mathsf{Q})$ and a model of Q in Grph is the same as a graph internal to $\mathsf{Mod}(\mathsf{Q})$. This extends to an equivalences of categories

$$\mathsf{Mod}(\mathsf{Q}, \mathsf{Cat}) \cong \mathsf{Cat}(\mathsf{Mod}(\mathsf{Q})) \text{ and } \mathsf{Mod}(\mathsf{Q}, \mathsf{Grph}) \cong \mathsf{Grph}(\mathsf{Mod}(\mathsf{Q})).$$

Therefore, in this section we instead construct an adjunction

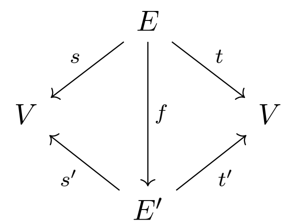
$$\mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}) \underset{U_{\mathsf{Q}}}{\overset{F_{\mathsf{Q}}}{\rightleftarrows}} \mathsf{Cat}(\mathsf{Mod}(\mathsf{Q}))$$

i.e. we construct free categories internal to $\mathsf{Mod}(\mathsf{Q})$. This adjunction is not new, and was first given in [BJT97]. In this section we obtain it by applying a construction of Lack [Lac10] to the following monoidal category:

Definition 51. Let $\mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))(V)$ be the monoidal category where

- objects are given by graphs $s, t\colon E \to V$ in $\mathsf{Mod}(\mathsf{Q})$,

- morphisms are given by maps $f\colon E \to E'$ making the diagram

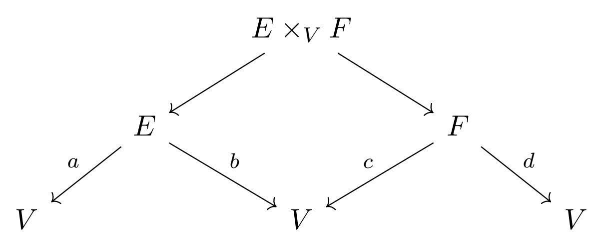
  

  commute.

- monoidal product is given by chosen pullbacks. That is, for spans

  their monoidal product is the chosen pullback

  On morphisms $f\colon E \to E'$ and $g\colon F \to F'$ is the unique map

  $$(f, g)\colon E \times_V F \to E' \times_V F'$$

  induced by the universal property of $E' \times_V F'$.

A monoid in this monoidal category is a span $s, t\colon E \to V$ along with multiplication and unit maps

$$\circ\colon E \times_V E \to E \text{ and } e\colon V \to E$$

satisfying associativity and unitality. Interpreting $\circ$ as composition and e as the map assigning identity morphisms, the monoid becomes a category internal to $\mathsf{Mod}(\mathsf{Q})$. Indeed, a category with object model V is exactly a monoid in the category $\mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))(V)$ [Bet96]. Therefore it suffices to show that $\mathsf{Grph}(\mathsf{Mod}(\mathsf{Q})(V)$ admits a free monoid construction. In Proposition 4 we handled the case when $\mathsf{Mod}(\mathsf{Q})$ is Set using the geometric series formula

$$F(G) = 1 + G + G^2 + \ldots = \sum_{n\geq 0} G^n$$

in the category of graphs $\mathsf{Grph}(V)$ over a fixed vertex set. The multiplication of $F(G)$ has the type

$$\sum_{n\geq 0} G^n \times \sum_{m\geq 0} G^m \to \sum_{n\geq 0} G^n.$$

Because products distribute over coproducts in $\mathsf{Grph}(V)$, we can factor the multiplication as

$$\sum_{n\geq 0} G^n \times \sum_{m\geq 0} G^m \xrightarrow{\sim} \sum_{m,n\geq 0} G^m \times G^n \xrightarrow{m} \sum_{n\geq 0} G^n$$

where m is the unique map induced by the natural concatenation morphisms $G^m \times G^n \to G^{m+n}$. In the general case, because products may not distribute over coproducts in the category $\mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))(V)$, this multiplication won't in general exist and a different construction of free monoids is necessary. Luckily Lack offers an alternative construction which replaces the coproducts of the previous approach with filtered colimits and reflexive coequalizers [Lac10]. These colimits are sifted so they commute with finite products and a multiplication map based on concatenation can be naturally defined.

Theorem 52. [Lack] *Let $(C, \otimes)$ be a monoidal category with*

- *finite limits,*

- *countable colimits, and*

- *the functors* $- \otimes A$ *and* $A \otimes -$ *preserve reflexive coequalizers and colimits of countable chains.*

Then C *admits a free monoid construction, that is, a left adjoint to the forgetful functor*

$$\mathrm{Mon}(C) \to C$$

that sends every monoid to its underlying object of C*.*

We now apply this Theorem to $\mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))(V)$.

Proposition 53. *For each object* V *in* $\mathsf{Mod}(\mathsf{Q})$*, there is an adjunction*

$$\mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))(V) \underset{\mathrm{B}^{\bullet}_V}{\overset{{}^{\bullet}\mathrm{B}_V}{\rightleftarrows}} \mathsf{Cat}(\mathsf{Mod}(\mathsf{Q}))(V)$$

where $\mathsf{Cat}(\mathsf{Mod}(\mathsf{Q}))(V)$ *is the category of categories internal to* $\mathsf{Mod}(\mathsf{Q})$ *whose model of objects is* V*.*

Proof. The hypotheses of Theorem 52 require that the following conditions hold:

- $\mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))(V)$ has finite limits and countable colimits. $\mathsf{Mod}(\mathsf{Q})$ has these limits and colimits as shown in Theorem 3.4.5 of [Bor94b]. The corresponding limits and colimits in $\mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))(V)$ are computed on the edges of each graph.

- The product of $\mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))(V)$ preserves colimits of countable chains and reflexive coequalizers. This is true because colimits of countable chains and reflexive coequalizers are sifted colimits so they commute with finite products.

Applying Theorem 52 to the category $\mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))(V)$ gives the desired result. $\square$

To complete the adjunction exhibiting the operational semantics of Q-nets, we need to remove the dependence on the model of vertices V. To accomplish this, we use the Grothendieck construction [Bor94b].

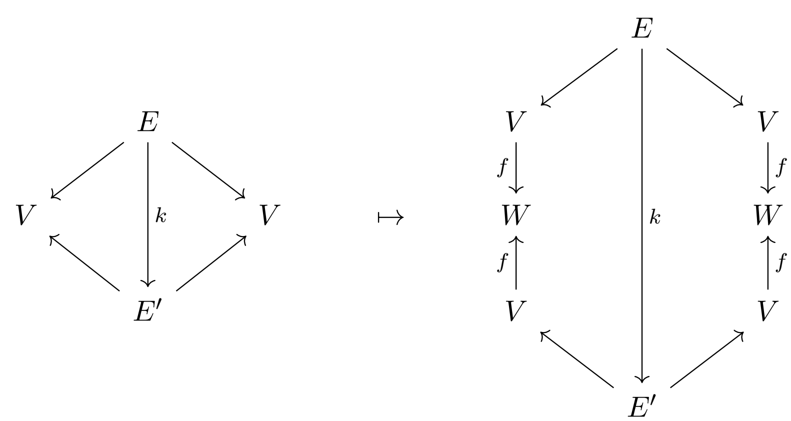
Definition 54. Let

$$\mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))(-)\colon \mathsf{Mod}(\mathsf{Q}) \to \mathsf{CAT}$$

be the functor which sends an object V to the category $\mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))(V)$ of graphs over V. For a morphism $f\colon V \to W$ in $\mathsf{Mod}(\mathsf{Q})$, let

$$\mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))(f)\colon \mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))(V) \to \mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))(W)$$

be the functor which makes the assignment



on objects and morphisms. Let

$$\mathsf{Cat}(\mathsf{Mod}(\mathsf{Q}))(-)\colon \mathsf{Mod}(\mathsf{Q}) \to \mathsf{CAT}$$

be the functor which sends an object V to the category of small categories internal to $\mathsf{Mod}(\mathsf{Q})$ with object model of Q given by V. For a morphism $f\colon V \to W$, let

$$\mathsf{Cat}(\mathsf{Mod}(\mathsf{Q}))(f)\colon \mathsf{Cat}(\mathsf{Mod}(\mathsf{Q}))(V) \to \mathsf{Cat}(\mathsf{Mod}(\mathsf{Q}))(W)$$

be the functor which which makes the assignment

$$\begin{array}{ccc} \operatorname{Mor} C & \overset{s}{\underset{t}{\rightrightarrows}} & V \\ {\scriptstyle k}\downarrow & & \downarrow{\scriptstyle \mathrm{id}} \\ \operatorname{Mor} C' & \overset{s'}{\underset{t'}{\rightrightarrows}} & V \end{array} \longmapsto \begin{array}{ccccc} \operatorname{Mor} C & \overset{s}{\underset{t}{\rightrightarrows}} & V & \xrightarrow{f} & W \\ {\scriptstyle k}\downarrow & & \downarrow{\scriptstyle \mathrm{id}} & & \downarrow{\scriptstyle \mathrm{id}} \\ \operatorname{Mor} C' & \overset{s'}{\underset{t'}{\rightrightarrows}} & V & \xrightarrow[f]{} & W \end{array}$$

on the underlying graphs of objects and morphisms.

Proposition 53 is reframed in this context.

Proposition 55. *The family of adjunctions*

$$\mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))(V) \underset{\mathrm{B}^{\bullet}{}_{V}}{\overset{{}^{\bullet}\mathrm{B}_{V}}{\rightleftarrows}} \mathsf{Cat}(\mathsf{Mod}(\mathsf{Q}))(V)$$

form components of natural transformations

$$C \colon \mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))(-) \Rightarrow \mathsf{Cat}(\mathsf{Mod}(\mathsf{Q}))(-) \text{ and } \mathrm{B}^{\bullet} \colon \mathsf{Cat}(\mathsf{Mod}(\mathsf{Q}))(-) \Rightarrow \mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))(-)$$

Furthermore, C and $\mathrm{B}^{\bullet}$ form an adjoint pair in the 2-category $[\mathsf{Mod}(\mathsf{Q}), \mathsf{Cat}]$ where

- *objects are functors $F \colon \mathsf{Mod}(\mathsf{Q}) \to \mathsf{Cat}$,*
- *morphisms are natural transformations $\alpha \colon F \Rightarrow G$ whose components $\alpha_c \colon F(c) \to G(c)$ are functors and,*
- *2-morphisms are modifications $\gamma \colon \alpha \to \beta$. That is, for every object c in $\mathsf{Mod}(\mathsf{Q})$ a natural transformation of the type*

$$F(c) \underset{\beta_c}{\overset{\alpha_c}{\rightrightarrows}} G(c), \qquad \gamma_c \colon \alpha_c \Rightarrow \beta_c.$$

Proof. For naturality, it suffices to show that the squares

$$\begin{array}{ccc}
\mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))(V) & \xrightarrow{{}^\bullet\mathrm{B}_V} & \mathsf{Cat}(\mathsf{Mod}(\mathsf{Q}))(V) \\
{\scriptstyle \mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))(f)}\downarrow & & \downarrow{\scriptstyle \mathsf{Cat}(\mathsf{Mod}(\mathsf{Q}))(f)} \\
\mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))(W) & \xrightarrow[{}^\bullet\mathrm{B}_W]{} & \mathsf{Cat}(\mathsf{Mod}(\mathsf{Q}))(W)
\end{array}$$

and

$$\begin{array}{ccc}
\mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))(V) & \xleftarrow{\mathrm{B}^\bullet{}_V} & \mathsf{Cat}(\mathsf{Mod}(\mathsf{Q}))(V) \\
{\scriptstyle \mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))(f)}\downarrow & & \downarrow{\scriptstyle \mathsf{Cat}(\mathsf{Mod}(\mathsf{Q}))(f)} \\
\mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))(W) & \xleftarrow[\mathrm{B}^\bullet{}_V]{} & \mathsf{Cat}(\mathsf{Mod}(\mathsf{Q}))(W)
\end{array}$$

commute. This is verified by direct computation. To show that ${}^\bullet\mathrm{B}$ and $\mathrm{B}^\bullet$ are an adjoint pair we need the following fact: ${}^\bullet\mathrm{B}$ is a left adjoint to $\mathrm{B}^\bullet$ in $[\mathsf{Mod}(\mathsf{Q}), \mathsf{Cat}]$ if and only if the components

$$\mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))(V) \underset{\mathrm{B}^\bullet{}_V}{\overset{{}^\bullet\mathrm{B}_V}{\rightleftarrows}} \mathsf{Cat}(\mathsf{Mod}(\mathsf{Q}))(V)$$

form an adjoint pair in Cat. The counit-unit definition of adjunction requires that we have modifications $\epsilon\colon {}^\bullet\mathrm{B} \circ \mathrm{B}^\bullet \to 1_{\mathsf{Cat}(\mathsf{Mod}(\mathsf{Q}))(-)}$ and $\eta\colon 1_{\mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))(-)} \to \mathrm{B}^\bullet \circ {}^\bullet\mathrm{B}$ satisfying the snake equations. Unpacking this gives components $\epsilon_V : {}^\bullet\mathrm{B}_V \circ \mathrm{B}^\bullet_V \Rightarrow 1_{\mathsf{Cat}(\mathsf{Mod}(\mathsf{Q}))(V)}$ and $\eta_V : \mathrm{B}^\bullet_V \circ {}^\bullet\mathrm{B}_V \Rightarrow 1_{\mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))(V)}$ satisfying the snake equations. This is equivalent to each component being an adjunction. However, Theorem 52 says that each component is an adjunction so the claim is shown. ∎

So far we have the diagram

$$\mathsf{Mod}(\mathsf{Q}) \underset{\mathsf{Cat}(\mathsf{Mod}(\mathsf{Q}))(-)}{\overset{\mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))(-)}{\rightrightarrows}} \mathsf{Cat} \qquad {}^\bullet\mathrm{B} \Downarrow \quad \Uparrow \mathrm{B}^\bullet$$

of adjoint 1-cells in $[\mathsf{Mod}(\mathsf{Q}), \mathsf{Cat}]$. We apply the Grothendieck construction to this diagram to get

$$\int \mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))(-) \underset{\int \mathrm{B}^{\bullet}}{\overset{\int^{\bullet} \mathrm{B}}{\rightleftarrows}} \int \mathsf{Cat}(\mathsf{Mod}(\mathsf{Q}))(-)$$

The Grothendieck construction is a 2-functor $\int : [\mathsf{Mod}(\mathsf{Q}), \mathsf{CAT}] \to \mathsf{CAT}/\mathsf{Mod}(\mathsf{Q})$ where CAT denotes the 2-category of large categories, functors, and natural transformations. When composed with the forgetful 2-functor $\mathsf{CAT}/\mathsf{Mod}(\mathsf{Q}) \to \mathsf{CAT}$ which remembers only the domain of each functor, we obtain the composite

$$\int : [\mathsf{Mod}(\mathsf{Q}), \mathsf{CAT}] \to \mathsf{CAT}$$

which we denote as $\int$ in an abuse of notation.

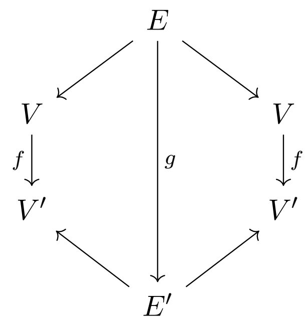
A fundamental fact is that every 2-functor preserves adjunctions. Therefore the above diagram is an adjunction. Moreover, the following proposition shows that it is the adjunction we are looking for.

Proposition 56. *The category* $\int \mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))(-)$ *is equivalent to* $\mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))$ *and the category* $\int \mathsf{Cat}(\mathsf{Mod}(\mathsf{Q}))(-)$ *is equivalent to* $\mathsf{Mod}(\mathsf{Q}, \mathsf{Cat})$.

Proof. $\int \mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))(-)$ has

- pairs $(V, V \leftarrow E \to V)$ as objects and,

- pairs $(f\colon V \to V', g\colon E \to E')$ such that the diagram



  in $\mathsf{Mod}(\mathsf{Q})$ commutes as morphisms.

An equivalence $\int \mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))(-) \xrightarrow{\sim} \mathsf{Grph}(\mathsf{Mod}(\mathsf{Q}))$ sends $(V, V \leftarrow E \rightarrow V)$ to the graph $E \rightrightarrows V$ and a morphism (f, g) to the evident morphism of graphs $(f\colon E \to E', g\colon V \to V')$. $\int \mathsf{Cat}(\mathsf{Mod}(\mathsf{Q}))(-)$ has

- pairs (V, C) where C is a category over V as objects and,

- pairs $(f\colon V \to V', g\colon C \to C')$ where g is an object fixing functor from $\mathsf{Cat}(\mathsf{Mod}(\mathsf{Q}))(f)(C)$ to C' as morphisms.

An equivalence $\int \mathsf{Cat}(\mathsf{Mod}(\mathsf{Q}))(-) \xrightarrow{\sim} \mathsf{Cat}(\mathsf{Mod}(\mathsf{Q})$ is given by sending objects (V, C) to their second component and morphisms (f, g) to the functor whose object component is f and whose morphism component is the morphism component of g. $\square$

We denote the compositions of $\int^{\bullet} \mathrm{B}$ and $\int \mathrm{B}^{\bullet}$ with the above equivalences by ${}^{\bullet}\mathrm{B}_{\mathsf{Q}}$ and $\mathrm{B}^{\bullet}_{\mathsf{Q}}$ respectively.

Proof of Theorem 45. The composite adjunction $F_{\mathsf{Q}} \dashv U_{\mathsf{Q}}$ is constructed by setting $F_{\mathsf{Q}} =^{\bullet} \mathrm{B}_{\mathsf{Q}} \circ^{\bullet} \mathrm{A}_{\mathsf{Q}}$ and $U_{\mathsf{Q}} = \mathrm{B}^{\bullet}_{\mathsf{Q}} \circ \mathrm{A}^{\bullet}_{\mathsf{Q}}$. $\square$

Chapter 4

Compositionality of Q-Nets

In this chapter we use the operational semantics developed in the previous chapter to develop a compositional theory of the behavior of Q-nets. In Section 4.1 we define "open" Q-nets, i.e. Q-nets equipped with input and output ports. Open Q-nets are glued together via pushout. There is a symmetric monoidal double category called $\mathsf{Open}(\mathsf{Q\text{-}Net})$ where the horizontal morphisms are open Q-nets and horizontal composition is pushout. In Section 4.2 we show how the operational semantics functor

$$F_{\mathsf{Q}} \colon \mathsf{Q\text{-}Net} \to \mathsf{Q\text{-}Cat}$$

of Theorem 45 lifts to a symmetric monoidal double functor

$$\mathsf{Open}(F) \colon \mathsf{Open}(\mathsf{Q\text{-}Net}) \to \mathsf{Open}(\mathsf{Q\text{-}Cat}).$$

Because composition in $\mathsf{Open}(\mathsf{Q\text{-}Net})$ is gluing, functoriality of this double functor gives relationships between the behavior of a Q-net and the behaviors of its components. In Section 4.3 we define the black-boxing of an open Q-category: a profunctor that encapsulates the morphisms from the input ports to the output ports. In Theorem 66 we show that black-boxing lifts to a lax double functor

$$\blacksquare \colon \mathsf{Open}(\mathsf{Q}\text{-}\mathsf{Cat}) \to \mathsf{Prof}.$$

We define functional open Q-nets, based off of the functional Petri nets of Zaitsev and Sleptsov [ZS97], as open Q-nets for which every input port is a source and every output port is a sink. Functional open Q-nets generalize the functional open graphs of Definition 18. In Theorem 68, we show that $\blacksquare \circ \mathsf{Open}(F)$ is strictly functorial on functional open Q-nets. This gives a straightforward expression of the compositionality of functional open Q-nets which does not suffer from combinatorial explosion.

4.1 Open Q-nets

In this section we define open Q-nets and construct a double category Open(Q-Net) with open Q-nets as horizontal 1-morphisms. As in Chapter 2, to define open Q-nets, we require a functor $L \colon \mathsf{Set} \to \mathsf{Q}\text{-}\mathsf{Net}$ that maps any set S to a Q-net with S as its set of places, and we need L to be a left adjoint.

Definition 57. Let $L \colon \mathsf{Set} \to \mathsf{Q}\text{-}\mathsf{Net}$ be the functor defined on sets and functions as follows:

$$\begin{array}{ccc} X & & \emptyset \rightrightarrows M_{\mathsf{Q}}[X] \\ f \downarrow & \mapsto & \downarrow \qquad\qquad \downarrow M_{\mathsf{Q}}[f] \\ Y & & \emptyset \rightrightarrows M_{\mathsf{Q}}[Y] \end{array}$$

where the unlabeled maps are the unique maps of their type.

Lemma 58. *The functor L has a right adjoint $R\colon$ Q-Net $\to$ Set that acts as follows on* Q*-nets and their morphisms:*

$$\begin{array}{ccc} T \overset{s}{\underset{t}{\rightrightarrows}} M_{\mathsf{Q}}[S] & & S \\ f\downarrow \qquad\quad \downarrow M_{\mathsf{Q}}[g] & \mapsto & \downarrow g \\ T' \overset{s'}{\underset{t'}{\rightrightarrows}} M_{\mathsf{Q}}[S] & & S'. \end{array}$$

Proof. For any set X and Q-net $P = (s,t\colon T \to M_{\mathsf{Q}}[S])$ we have natural isomorphisms

$$\begin{aligned} \hom_{\mathsf{Q\text{-}Net}}\big(L(X),\ T \overset{s}{\underset{t}{\rightrightarrows}} M_{\mathsf{Q}}[S]\,\big) &\cong \hom_{\mathsf{Q\text{-}Net}}\big(\,\emptyset \rightrightarrows M_{\mathsf{Q}}[X]\ ,\ T \overset{s}{\underset{t}{\rightrightarrows}} M_{\mathsf{Q}}[S]\,\big) \\ &\cong \hom_{\mathsf{Set}}(X,S) \\ &\cong \hom_{\mathsf{Set}}\big(X, R(T \overset{s}{\underset{t}{\rightrightarrows}} M_{\mathsf{Q}}[S])\big). \qquad \square \end{aligned}$$

An "open" Q-net is a Q-net P equipped with maps from two sets X and Y into its set of places, RP. We can write this as a cospan in Set of the form

$$\begin{array}{ccc} & RP & \\ \nearrow & & \nwarrow \\ X & & Y. \end{array}$$

Using the left adjoint L we can reexpress this as a cospan in Q-Net, and this gives our official definition:

Definition 59. An **open Q-net** is a diagram in Q-Net of the form

$$\begin{array}{ccc} & P & \\ {}^{i}\nearrow & & \nwarrow^{o} \\ LX & & LY \end{array}$$

for some sets X and Y. We sometimes write this as $P\colon X \nrightarrow Y$ for short.

We now introduce the main object of study for this section: the double category Open(Q-Net), which has open Q-nets as its horizontal 1-cells.

Theorem 60. *There is a symmetric monoidal double category* Open(Q-Net) *for which:*

- *objects are sets*
- *vertical 1-morphisms are functions*
- *horizontal 1-cells from a set X to a set Y are open* Q*-nets*

$$\begin{array}{ccccc} & & P & & \\ & \overset{i}{\nearrow} & & \overset{o}{\nwarrow} & \\ LX & & & & LY \end{array}$$

- *2-morphisms $\alpha\colon P \Rightarrow P'$ are commutative diagrams*

$$\begin{array}{ccccc} LX & \xrightarrow{i} & P & \xleftarrow{o} & LY \\ {\scriptstyle Lf}\downarrow & & {\scriptstyle \alpha}\downarrow & & \downarrow{\scriptstyle Lg} \\ LX' & \xrightarrow[i']{} & P' & \xleftarrow[o']{} & LY'. \end{array}$$

  in Q-Net*.*

Composition of vertical 1-morphisms is the usual composition of functions. Composition of horizontal 1-cells is composition of cospans via pushout: given two horizontal 1-cells

$$\begin{array}{ccccc} & & P & & \\ & \overset{i_1}{\nearrow} & & \overset{o_1}{\nwarrow} & \\ LX & & & & LY \end{array} \qquad \begin{array}{ccccc} & & Q & & \\ & \overset{i_2}{\nearrow} & & \overset{o_2}{\nwarrow} & \\ LY & & & & LZ \end{array}$$

their composite is given by this cospan from LX to LZ:

$$\begin{array}{ccccccccc} & & & & P +_{LY} Q & & & & \\ & & & \overset{j_P}{\nearrow} & & \overset{j_Q}{\nwarrow} & & & \\ & & P & & & & Q & & \\ & \overset{i_1}{\nearrow} & & \overset{o_1}{\nwarrow} & & \overset{i_2}{\nearrow} & & \overset{o_2}{\nwarrow} & \\ LX & & & & LY & & & & LZ \end{array}$$

where the diamond is a pushout square. The horizontal composite of 2-morphisms

$$
\begin{CD}
LX @>{i_1}>> P @<{o_1}<< LY \\
@V{Lf}VV @V{\alpha}VV @VV{Lg}V \\
LX' @>>{i'^1}> P' @<<{o'^1}< LY'
\end{CD}
\qquad\qquad
\begin{CD}
LY @>{i_2}>> Q @<{o_2}<< LZ \\
@V{Lg}VV @V{\beta}VV @VV{Lh}V \\
LY' @>>{i'^2}> Q' @<<{o'^2}< LZ'
\end{CD}
$$

is given by

$$
\begin{CD}
LX @>{j_P i_1}>> P +_{LY} Q @<{j_Q o_1}<< LZ \\
@V{Lf}VV @V{\alpha +_{Lg} \beta}VV @VV{Lh}V \\
LX' @>>{j_{P'} i'_1}> P' +_{LY'} Q' @<<{j_{Q'} o'_2}< LZ'.
\end{CD}
$$

Vertical composition of 2-morphisms is done using composition of functions. The symmetric monoidal structure comes from coproducts in Set *and* Q-Net*.*

Proof. We construct this symmetric monoidal double category using Lemma 7. This lemma requires that Q-Net has all colimits and this was proved in Proposition 40. □

Example 61. Setting Q equal to the theory of commutative monoids gives open Petri nets. This double category and its properties are explored in detail in the paper [BM20] by the author and Baez.

In the remarks following Example 38, we exploited the functoriality of Definition 29 to describe functorial relationships between different categories of Q-nets. This functoriality can be extended to produce symmetric monoidal double functors between double categories of open Q-nets.

Proposition 62. *Every morphism of Lawvere theories $f\colon Q \to R$ induces a lax symmetric monoidal lax double functor*

$$\mathsf{Open}(f\text{-}\mathsf{Net})\colon \mathsf{Open}(\mathsf{Q}\text{-}\mathsf{Net}) \to \mathsf{Open}(\mathsf{R}\text{-}\mathsf{Net})$$

Proof. From Definition 29 and Proposition 58 we have the following diagram

$$\begin{array}{ccc} \mathsf{Q\text{-}Net} & \xrightarrow{f\text{-}\mathsf{Net}} & \mathsf{R\text{-}Net} \\ {\scriptstyle L_{\mathsf{R}}}\uparrow & & \uparrow{\scriptstyle L_{\mathsf{Q}}} \\ \mathsf{Set} & = & \mathsf{Set} \end{array}$$

and it is straightforward to verify that this diagram commutes up to natural isomorphism. The desired double functor is obtained by applying Lemma 11 to this diagram. □

In Section 3.2 we constructed the following diagram of functors

$$\begin{array}{ccc} \mathsf{SLAT\text{-}Net} & & \\ {\scriptstyle a\text{-}\mathsf{Net}}\uparrow & & \\ \mathsf{Petri} & \xrightarrow{b\text{-}\mathsf{Net}} & \mathbb{Z}\text{-}\mathsf{Net} \\ {\scriptstyle c\text{-}\mathsf{Net}}\uparrow & & \uparrow{\scriptstyle e\text{-}\mathsf{Net}} \\ \mathsf{PreNet} & \xrightarrow[d\text{-}\mathsf{Net}]{} & \mathsf{GRP\text{-}Net} \end{array}$$

This diagram of functors can be continued one step further to categories of open Q-nets via Proposition 62:

$$\begin{array}{ccc} \mathsf{Open}(\mathsf{SLAT\text{-}Net}) & & \\ {\scriptstyle \mathsf{Open}(a\text{-}\mathsf{Net})}\uparrow & & \\ \mathsf{Open}(\mathsf{Petri}) & \xrightarrow{\mathsf{Open}(b\text{-}\mathsf{Net})} & \mathsf{Open}(\mathbb{Z}\text{-}\mathsf{Net}) \\ {\scriptstyle \mathsf{Open}(c\text{-}\mathsf{Net})}\uparrow & & \uparrow{\scriptstyle \mathsf{Open}(e\text{-}\mathsf{Net})} \\ \mathsf{Open}(\mathsf{PreNet}) & \xrightarrow[\mathsf{Open}(d\text{-}\mathsf{Net})]{} & \mathsf{Open}(\mathsf{GRP\text{-}Net}) \end{array}$$

This diagram says that the above functors between Q-nets can be extended in a coherent way to open Q-nets.

4.2 Compositionality of the Operational Semantics for Q-nets

In Theorem 45 we saw how a Q-net P gives a Q-category $F_{\mathsf{Q}}(P)$, and in Theorem 60 we constructed a double category $\mathsf{Open}(\mathsf{Q\text{-}Net})$ of open Q-nets. Now we construct a double category $\mathsf{Open}(\mathsf{Q}\text{-}\mathsf{Cat})$ of "open Q-categories" and a map

$$\mathsf{Open}(F_{\mathsf{Q}})\colon \mathsf{Open}(\mathsf{Q}\text{-}\mathsf{Net}) \to \mathsf{Open}(\mathsf{Q}\text{-}\mathsf{Cat}).$$

This can be seen as providing an operational semantics for open Q-nets in which any open Q-net is mapped to the Q-category it presents. The key is this commutative diagram of left adjoint functors:

$$\begin{array}{ccc} \mathsf{Set} & \xrightarrow{L} & \mathsf{Q}\text{-}\mathsf{Net} \\ & \searrow^{FL} & \downarrow F \\ & & \mathsf{Q}\text{-}\mathsf{Cat} \end{array}$$

where FL sends any set to the free Q-category on this set: FLX has $M_{\mathsf{Q}}[X]$, the free Q-model on X, as its set of objects and only identity morphisms. Using Lemma 7, we can produce two symmetric monoidal double categories from this diagram. We have already seen one: $\mathsf{Open}(\mathsf{Q}\text{-}\mathsf{Net})$ obtained from the left adjoint L. We now obtain $\mathsf{Open}(\mathsf{Q}\text{-}\mathsf{Cat})$ from the left adjoint FL.

Theorem 63. *There is a symmetric monoidal double category* $\mathsf{Open}(\mathsf{Q}\text{-}\mathsf{Cat})$ *for which:*

- *objects are sets*
- *vertical 1-morphisms are functions*
- *horizontal 1-cells from a set X to a set Y are* **open Q-categories** $C\colon X \nrightarrow Y$, *that is, cospans in* $\mathsf{Q}\text{-}\mathsf{Cat}$ *of the form*

$$\begin{array}{ccccc} & & C & & \\ & {}^{i}\nearrow & & \nwarrow^{o} & \\ L'X & & & & L'Y \end{array}$$

where C is a Q*-category and i, o are strict* Q*-functors,*

- *2-morphisms $\alpha\colon C \Rightarrow C'$ are commutative diagrams in* $\mathsf{Q\text{-}Cat}$ *of the form*

$$\begin{array}{ccccc} L'X & \xrightarrow{i} & C & \xleftarrow{o} & L'Y \\ {\scriptstyle L'f}\downarrow & & {\scriptstyle \alpha}\downarrow & & \downarrow{\scriptstyle L'g} \\ L'X' & \xrightarrow[i']{} & C' & \xleftarrow[o']{} & L'Y'. \end{array}$$

and the rest of the structure is given as in Lemma 7.

Proof. To apply Lemma 7 to the functor $FL\colon \mathsf{Set} \to \mathsf{Q\text{-}Cat}$ we just need to check that $\mathsf{Q\text{-}Cat}$ has finite colimits. First note that

$$\mathsf{Q\text{-}Cat} \simeq \mathsf{Mod}(\mathsf{Q}, \mathsf{Cat}).$$

The cocompleteness of this category then follows from various classical results, some listed in the introduction of a paper by Freyd and Kelly [FK72]. More recently, Trimble [Tri14, Prop. 3.1] showed that for any Lawvere theory Q and any cocomplete cartesian category X with finite products distributing over colimits, the category of finite-product-preserving functors $\mathsf{Mod}(\mathsf{Q}, \mathsf{X})$ is cocomplete. $\square$

The functor $F\colon \mathsf{Q\text{-}Net} \to \mathsf{Q\text{-}Cat}$ induces a map sending open Q-nets to open Q-categories. This map is actually part of a symmetric monoidal double functor.

Theorem 64. *There is a symmetric monoidal double functor*

$$\mathsf{Open}(F)\colon \mathsf{Open}(\mathsf{Q\text{-}Net}) \to \mathsf{Open}(\mathsf{Q\text{-}Cat})$$

that is the identity on objects and vertical 1-morphisms, and makes the following assignments on horizontal 1-cells and 2-morphisms:

$$\begin{array}{ccccc} LX & \xrightarrow{i} & P & \xleftarrow{o} & LY \\ {\scriptstyle Lf}\downarrow & & {\scriptstyle \alpha}\downarrow & & \downarrow{\scriptstyle Lg} \\ LX' & \xrightarrow[i']{} & P' & \xleftarrow[o']{} & LY' \end{array} \quad \mapsto \quad \begin{array}{ccccc} L'X & \xrightarrow{Fi} & FP & \xleftarrow{Fo} & L'Y \\ {\scriptstyle L'f}\downarrow & & {\scriptstyle F\alpha}\downarrow & & \downarrow{\scriptstyle L'g} \\ L'X' & \xrightarrow[Fi']{} & FP' & \xleftarrow[Fo']{} & L'Y'. \end{array}$$

Proof. We apply Lemma 11 to the commutative square

$$\begin{array}{ccc} \mathsf{Q\text{-}Net} & \xrightarrow{F} & \mathsf{Q\text{-}Cat} \\ {\scriptstyle L}\Big\uparrow & & \Big\uparrow{\scriptstyle F\circ L} \\ \mathsf{Set} & = & \mathsf{Set} \end{array}$$

of left adjoint functors. ☐

We can think of the Q-category FP as providing an operational semantics for the Q-net P: morphisms in this category are processes allowed by the Q-net. The above theorem says that this semantics is compositional. That is, if we write P as a composite (or tensor product) of smaller open Q-nets, FP will be the composite (or tensor product) of the corresponding open Q-categories.

4.3 Black-boxing and Functional Open Q-nets

In this section we show how the black-boxing functor introduced in Theorem 16 can be applied to the categorical operational semantics of open Q-nets. We also introduce functional open Q-nets and prove that black-boxing preserves composition of functional open Q-nets up to isomorphism. To define black-boxing for open Q-nets we make use of the following functor.

Definition 65. Let

$$U\colon \mathsf{Q\text{-}Cat} \to \mathsf{Cat}$$

be the forgetful functor that regards every Q-category as an ordinary category and every Q-functor as an ordinary functor.

Theorem 66. *There is a lax double functor*

$$\blacksquare\colon \mathsf{Open(Q\text{-}Cat)} \to \mathsf{Prof}$$

that

- *sends sets* X,Y *and functions* $f\colon X \to Y$ *to the discrete categories and functors between them.*

- *An open* Q*-category*

$$\begin{array}{ccc} & C & \\ {\scriptstyle i}\nearrow & & \nwarrow{\scriptstyle j} \\ FLX & & FLY \end{array}$$

  to the profunctor

$$\blacksquare(C)\colon UFLX \times UFLY \to \mathsf{Set}$$

  given by $\blacksquare(C)(x, y) = UFP(i(x), j(y))$.

- *A 2-cell of open* Q*-nets*

$$\begin{array}{ccccc} FLX & \xrightarrow{i} & C & \xleftarrow{o} & FLY \\ {\scriptstyle Lf}\downarrow & & \downarrow{\scriptstyle g} & & \downarrow{\scriptstyle Lh} \\ FLX' & \xrightarrow[i']{} & D & \xleftarrow[o']{} & FLY' \end{array}$$

  is sent to the 2-cell of profunctors

$$\begin{array}{ccc} UFLX \times UFLY & \xrightarrow{UFLf \times UFLh} & UFLX' \times UFLY' \\ & \overset{\alpha}{\Longrightarrow} & \\ {\scriptstyle \blacksquare(C)}\searrow & \mathsf{Set} & \swarrow{\scriptstyle \blacksquare(D)} \end{array}$$

  where the components of α

$$\alpha_{x,y}\colon UC(ix, oy) \to UD(i'fx, o'hy)$$

  are given by pointwise application of the functor Ug.

Proof. We factor $\blacksquare$ into three parts

$$\mathsf{Open}(\mathsf{Q\text{-}Cat}) \hookrightarrow \mathsf{Csp}(\mathsf{Q\text{-}Cat}) \xrightarrow{\mathsf{Open}(U)} \mathsf{Csp}(\mathsf{Cat}) \xrightarrow{\blacksquare_g} \mathsf{Prof}$$

where

- the inclusion $\mathsf{Open}(\mathsf{Q\text{-}Cat}) \hookrightarrow \mathsf{Csp}(\mathsf{Q\text{-}Cat})$ is given by pointwise application of $FL\colon \mathsf{Set} \to \mathsf{Q\text{-}Cat}$ on sets and functions and is given by the identity on horizontal morphisms and 2-cells.

- The functor $\mathsf{Open}(U)$ is given by applying Lemma 11 to the square

$$\begin{array}{ccc} \mathsf{Q\text{-}Cat} & \xrightarrow{U} & \mathsf{Cat} \\ \| & & \| \\ \mathsf{Q\text{-}Cat} & \xrightarrow[U]{} & \mathsf{Cat} \end{array}$$

  Explicitly it is given by pointwise application of U everywhere. Note that because U does not preserve finite colimits, this double functor will preserve horizontal composition laxly.

- $\blacksquare_g$ is an extension of the black-boxing functor

$$\blacksquare\colon \mathsf{Open}(\mathsf{Cat}) \to \mathsf{Prof}$$

  of Theorem 16 to the domain $\mathsf{Csp}(\mathsf{Cat})$. Explicitly, $\blacksquare_g$ is the identity on categories and functors and the black-boxing of horizontal morphisms and 2-cells is exactly as in Theorem 16.

The desired double functor is obtained by composing the three double functors above. □

The phenomenon of Example 17 persists for the black-boxing of open Q-nets: the double functor $\blacksquare$ is lax rather than strict because profunctor composition only accounts for firing sequences which go from the first component to the second and do not come back. Functional Q-nets are a class of Q-nets for which firing sequences on a composite open Petri net only flow from the the first component to the second. Functional Petri nets were first introduced by Zaitsev and Sleptsov [ZS97]. Their definition generalizes straightforwardly to Q-nets.

Definition 67. Let $P\colon X \to Y$ be the open Q-net

$$LX \xrightarrow{i} P \xleftarrow{j} LY$$

An element $x \in X$ is a **source** if $i(x)$ is not the target of any transition in P and an element $y \in Y$ is a **sink** if $j(y)$ is not the source of any transition in P. $P\colon X \to Y$ is **functional** if every $x \in X$ is a source and every $y \in Y$ is a sink.

For functional open Q-nets profunctor composition accounts for all firing sequences on a composite. Therefore black-boxing preserves horizontal composition up to isomorphism when the component Q-nets are functional.

Theorem 68. *The composite double functor*

$$\blacksquare \circ \mathsf{Open}(F)\colon \mathsf{Open}(\mathsf{Q\text{-}Net}) \to \mathsf{Prof}$$

preserves horizontal composition of functional Q*-nets up to isomorphism.*

Proof. Let $P\colon X \to Y$ and $\mathsf{Q}\colon Y \to Z$ be functional open Q-nets. Then their black-boxings are equipped with a composition comparison

$$\alpha\colon \blacksquare \circ \mathsf{Open}(F)(P) \circ \blacksquare \circ \mathsf{Open}(F)(Q) \to \blacksquare \circ \mathsf{Open}(F)(P \circ Q)$$

with components

$$\alpha_{x,z}\colon \int_{y \in M_{\mathsf{Q}}(Y)} \blacksquare(\mathsf{Open}(P))(x,y) \times \blacksquare(\mathsf{Open}(Q))(y,z) \to \blacksquare(\mathsf{Open}(P) \circ \mathsf{Open}(Q))(x,z)$$

given by sending a pair of morphisms (g, f) to their composite $g \circ f$ in $\mathsf{Open}(P) \circ \mathsf{Open}(Q)$. Because P and Q are functional, every morphism from x to z in $\mathsf{Open}(P) \circ \mathsf{Open}(Q)$ is of this form. Therefore the composition comparison is an isomorphism. □

Remark. Let $\mathbb{R}\mathbf{el}$ be the double category where objects are sets, vertical morphisms are functions, horizontal morphisms are relations, and 2-cells are rectangles

$$\begin{array}{ccc} A & \overset{R}{\rightsquigarrow} & B \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle g} \\ A' & \underset{R'}{\rightsquigarrow} & B \end{array}$$

such that there is an inclusion

$$(f \times g) \circ R \subseteq R'.$$

In [BM20], the author and Baez construct a different black-boxing functor

$$\blacksquare\colon \mathsf{Open}(\mathsf{Petri}) \to \mathbb{R}\mathbf{el}$$

that sends an open Petri net $LX \xrightarrow{i} P \xleftarrow{o} LY$ to its reachability relation i.e. the relation $\blacksquare(P) \subseteq \mathbb{N}[X] \times \mathbb{N}[Y]$ given by

$$\blacksquare(P) = \{(x,y) \in \mathbb{N}[X] \times \mathbb{N}[Y] \mid \text{ there exists a morphism } f\colon x \to y \text{ in } FP(ix, oy)\}.$$

The reachability semantics functor is the decategorification of the black-boxing functor of Theorem 66. Any profunctor $P\colon C \times D^{\mathrm{op}} \to \mathsf{Set}$ can be turned into a relation $R \subseteq \operatorname{Ob} C \times \operatorname{Ob} D$ given by

$$R = \{(x,y) \mid P(x,y) \text{ is nonempty.}\}$$

When the profunctor $\blacksquare_{\mathsf{CMON}}(FLX \xrightarrow{Fi} FP \xleftarrow{o} FLY)$ is turned into a relation in this way it becomes the reachability relation for P. This process may be called decategorification as it arises from the change of enrichment $\mathsf{Set} \to \{0,1\}$ where $\{0,1\}$ is boolean monoid regarded as a monoidal category. The change of enrichment is a functor which sends a set to 1 if it is non-empty and extends to a double functor

$$\mathsf{Prof} \to \mathbb{R}\mathbf{el}.$$

To obtain the black-boxing of Open Petri nets, the black-boxing of this thesis is composed with the above decategorification to obtain the reachability semantics double functor of [BM20].

In [Zai05], Zaitsev gives a polynomial time algorithm for decomposing a Petri net into functional open Petri nets. This decomposition provides a speedup for computing invariants of Petri nets. This chapter offers a formal language and general language to understand this strategy of decomposition in a larger context.

Chapter 5

Operational Semantics of Enriched Graphs

The algebraic path problem is a generalization of the shortest path problem to probability, computing, matrix multiplication, and optimization [Tar81, Foo15]. Let R be the quantale of positive real numbers $([0, \infty], \min, +)$. A weighted graph is regarded as an R-matrix, and the shortest paths of this graph are computed as the operational semantics studied in this chapter. The algebraic path problem allows R to vary, and gets solutions to other problems of a similar flavor within the same framework. In Section 5.1 we review the relevant definitions for quantales and the algebraic path problem. In Section 5.2 we generalize the free category construction of Proposition 4 to graphs enriched in R. In Theorem 3.6 we obtain for every quantale R an adjunction

$$\mathsf{Mat}_R \underset{U}{\overset{F}{\rightleftarrows}} \mathsf{RCat} \quad (F \dashv U)$$

between R-matrices and categories enriched in R. Remarkably, the free R-category $F(M)$ on an R-matrix M is the solution to its algebraic path problem. The adjunction above encapsulates the universal property of this solution. Note that the functors in the above adjunction are referred to as F_R and U_R in the introduction to disambiguate from the other functors in this thesis. However, because this chapter is self contained we drop the subscript to reduce notational clutter.

5.1 The Algebraic Path Problem

The networks considered in this chapter will be parameterized by commutative quantales.

Definition 69. A **quantale** is a monoidal closed poset with all joins. Explicitly, a quantale is a a poset R with an associative, unital, and monotone multiplication $\cdot\colon R \times R \to R$ such that

- all joins, $\sum_{i \in I} x_i$, exist for arbitrary index set I and
- $\cdot$ preserves all joins, i.e.

$$a \cdot \sum_{i \in I} x_i = \sum_{i \in I} a \cdot x_i$$

  for all joins over an arbitrary index set I.

A quantale is commutative if its multiplication operation, $\cdot$, is commutative.

A motivating example of such a quantale is the poset $[0, \infty]$ with $+$ as its monoidal product and with join given by infimum. Note that this poset is equipped with the reverse of the usual ordering on $[0, \infty]$. Fong and Spivak show how the shortest path problem on this quantale computes the shortest paths between all pairs of vertices in a given $[0,\infty]$-weighted graph [FS19, §2.5.3]. Other motivating examples include the rig $([0,1],\sup,\times)$ (whose algebraic path problem corresponds to most likely path in a Markov chain) and the powerset of the language generated by an alphabet (whose algebraic path problem corresponds to the language decided by a nondeterministic finite automata (NFA))[Foo15].

Definition 70. For a commutative quantale R and sets X and Y, an ***R*-matrix** $M\colon X\to Y$ is a function $M\colon X\times Y\to R$. For R-matrices $M\colon X\to Y$ and $N\colon Y\to Z$, their matrix product MN is defined by the rule

$$MN(i,k)=\sum_{j\in Y}M(i,j)N(j,k)$$

If R is a commutative quantale, R-matrices form a quantale as well.

Definition 71. Let $\mathsf{RMat}(X)$ be the set of X by X matrices $M\colon X\times X\to R$. $\mathsf{RMat}(X)$ is equipped with the partial order where $M\le N$ if and only if $M(i,j)\le N(i,j)$ for all $i,j\in X$.

Proposition 72. $\mathsf{RMat}(X)$ *is a quantale with*

- *join given by pointwise sum of matrices,*
- *and multiplication given by matrix product.*

The proof of this proposition is left to the reader. All the required properties of $\mathsf{RMat}(X)$ follow from the analogous properties in R.

A square matrix $M\colon X\times X\to R$ represents a complete R-weighted graph whose vertex set is given by X.

Definition 73. Let $M\colon X \times X \to R$ be a square matrix. A **vertex** of M is an element $i \in X$. An **edge** of M is a tuple of vertices $(a, b) \in X \times X$. A **path** in M from a_0 to a_n is a list of adjacent edges $p = ((a_0, a_1), (a_1, a_2), \ldots, (a_{n-1}, a_n),)$. The **weight** of p is defined as the product

$$l(p) = \Pi_{i=0}^{n-1} M(a_i, a_{i+1})$$

in R. For vertices $i, j \in X$, let

$$P_{ij}^M = \{ \text{ paths in } M \text{ from } i \text{ to } j \ \}$$

Let i and j be vertices of a square matrix $M\colon X \times X \to R$. The algebraic path problem asks to compute the quantity

$$\sum_{p \in P_{ij}^M} l(p)$$

in the quantale R. If R is the quantale $([0, \infty], \inf, +)$ then the weight of an edge M_{ij} represents the distance between vertex i and vertex j and the weight of a path $l(p)$ represents the total distance traversed by p. Summing the weights of all paths between a pair of vertices corresponds to finding the path with the minimum weight. For example, the algebraic path problem asks to compute the length of the shortest path in the case when R is $([0, \infty], \inf, +)$.

A more tractable framing of the algebraic path problem can be found by considering matrix powers. The entries of M^2 are given by

$$M^2(i, j) = \sum_{l \in X} M(i, l) M(l, j) = \inf_{l \in X} \{M(i, l) + M(l, j)\}.$$

Because $M(i, l)$ and $M(l, j)$ represent the distance from i to l and from l to j, this infimum computes the cheapest way to travel from i to j while stopping at some l in between. More generally, the entries of M^n for $n \geq 0$ represent the shortest paths between nodes of your graph that occur in exactly n steps. To compute the shortest paths which can occur in any number of steps, we must take the infimum of the matrices M^n over all $n \geq 0$. This pattern replicates for other choices of quantale. Therefore, the **algebraic path problem** seeks to compute

$$F(M) = \sum_{n \geq 0} M^n \tag{5.1}$$

where M is an R-matrix. The following table summarizes some instances of the algebraic path problem for different choices of R. Fink provides an explanation of the algebraic path problems for $([0, \infty], \leq)$ and $\{T, F\}$ and Foote provides an explanation for the quantales $([0, 1], \leq)$ and $(\mathcal{P}(\Sigma), \subseteq)$ [Fin92, Foo15].

poset	**join**	**multiplication**	**solution of path problem**
$([0, \infty], \geq)$	inf	$+$	shortest paths in a weighted graph
$([0, \infty], \leq)$	sup	inf	maximum capacity in the tunnel problem
$([0, 1], \leq)$	sup	$\times$	most likely paths in a Markov process
$\{T, F\}$	OR	AND	transitive closure of a directed graph
$(\mathcal{P}(\Sigma^*), \subseteq)$	$\bigcup$	concatenation	decidable language of a NFA

Note that in this table, $\mathcal{P}(\Sigma^*)$ denotes the power set of the language generated by an alphabet Σ.

5.2 The Algebraic Path Problem Functor

Equation 5.1 is known to category theorists by a different name: the free monoid on M. Framing it in this way gives a categorical proof of existence and uniqueness of $F(M)$. A classic result from [ML98, §V11] gives a construction of free monoids. MacLane's construction is defined as an adjunction into a category of internal monoids.

Definition 74. Let $(C, \otimes, I)$ be a monoidal category. A **monoid internal to *C*** is an object A of C equipped with morphisms

$$m \colon A \otimes A \to A \text{ and } i \colon I \to M$$

satisfying the axioms of associativity and unitality expressed as commutative diagrams. A **monoid homomorphism** from a monoid A to a monoid B is a morphism $f : A \to B$ in C which commutes with the maps m and i of each monoid. Let $\mathsf{Mon}(C)$ be the category where objects are monoids internal to C and morphisms are their homomorphisms.

Proposition 75 (MacLane)**.** *Let $(C, \otimes, I)$ be a monoidal category with countable coproducts such that tensoring on both sides preserves these coproducts. Then there is an adjunction*

$$C \underset{U}{\overset{F}{\rightleftarrows}} \mathsf{Mon}(C) \quad (F \dashv U)$$

whose left adjoint is given by the countable coproduct

$$F(X) = \sum_{n \geq 0} X^n. \tag{5.2}$$

The poset $\mathsf{RMat}(X)$ when viewed as a category satisfies the hypotheses of Proposition 75 and admits a free monoid construction.

Proposition 76. *There is an adjoint pair*

$$\mathsf{RMat}(X) \underset{U_X}{\overset{F_X}{\rightleftarrows}} \mathsf{Mon}(\mathsf{RMat}(X))$$

where F_X is the monotone map which produces the solution to the algebraic path problem on a matrix and U_X is the natural forgetful map.

Proof. Because $\mathsf{RMat}(X)$ is a quantale, it can be regarded as a monoidal category with all coproducts such that tensoring distributes over these coproducts. The result follows from applying Proposition 75 and noticing that Equation 5.2 matches Equation 5.1 in the case when $C = \mathsf{RMat}$. □

Monoids internal to $\mathsf{RMat}(X)$ are R-enriched categories.

Definition 77. An ***R*-enriched category** C with object set X consists of an element $C(x, y)$ in R for every $x, y \in X$ such that

- $1 \leq C(x, x)$ (the identity law),
- and $C(x, y)C(y, z) \leq C(x, z)$ (the composition law).

R-enriched categories will be referred to as R-categories. Let $\mathsf{RCat}(X)$ be the poset whose elements are R-categories with object set X. For R-categories C and D in $\mathsf{RCat}(X)$,

$$C \leq D \leftrightarrow C(i, j) \leq D(i, j) \quad \forall i, j \in X.$$

Proposition 78. $\mathsf{Mon}(\mathsf{RMat}(X))$ *is isomorphic to* $\mathsf{RCat}(X)$*, the poset of categories enriched in R with object set X.*

Proof. The isomorphism in question assigns a matrix $M \colon X \times X \to R$ to the R-category with $\hom(x, y) = M(x, y)$. The identity law follows from the inequality $1 \leq M$ and the inequality $M^2 \leq M$ implies that for all $y \in X$,

$$\sum_{y \in X} M(x, y)M(y, z) \leq M(x, z)$$

The composition law follows from the fact that any element of R is less than a join which contains it. □

Proposition 76 says that each matrix valued in R has a unique, universally characterized solution to the algebraic path problem: namely the free R-category on that matrix. This adjunction can be extended to matrices over an arbitrary set.

Definition 79. Let $f : X \to Y$ be a function and let $M : X \times X \to R$ be an R-matrix. Then the **pushforward** of M along f is the matrix $f_*(M) \colon Y \times Y \to R$ defined by

$$f_*(M)(y, y') = \sum_{(x,x') \in (f \times f)^{-1}(y,y')} M(x, x').$$

Definition 80. Let RMat be the category where objects are square matrices $M \colon X \times X \to R$ on a set X and where a morphism from $M \colon X \times X \to R$ to $N \colon Y \times Y \to R$ is a function $f \colon X \to Y$ satisfying

$$f_*(M) \leq N.$$

Let RCat be the full subcategory of RMat consisting of matrices satisfying the axioms of an R-category.

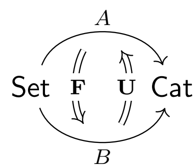
Theorem 81. *The free monoid construction of Proposition 76 extends to an adjunction*

$$\mathsf{RMat} \underset{U}{\overset{F}{\rightleftarrows}} \mathsf{RCat}, \quad F \dashv U.$$

Proof. Let $A \colon \mathsf{Set} \to \mathsf{Cat}$ be the functor which sends a set X to the poset $\mathsf{RMat}(X)$ regarded as a category and sends a function $f \colon X \to Y$ to the pushforward functor

$$f_* \colon \mathsf{RMat}(X) \to \mathsf{RMat}(Y).$$

Analogously, let $B \colon \mathsf{Set} \to \mathsf{Cat}$ be the functor which sends a set X to the poset $\mathsf{RCat}(X)$ and sends a function f to its pushforward functor. The functors F_X form the components of a natural transformation $\mathbf{F} \colon A \Rightarrow B$ and the functors U_X form the components of a natural transformation $\mathbf{U}\colon B \Rightarrow A$. Furthermore, these natural transformations form an adjoint pair in the 2-category $[\mathsf{Set}, \mathsf{Cat}]$ of functors $\mathsf{Set} \to \mathsf{Cat}$, natural transformations between them, and modifications. $\mathbf{F}$ and $\mathbf{U}$ are adjoint because an adjoint pair in $[\mathsf{Set}, \mathsf{Cat}]$ is a pair of natural transformations which are adjoint in each component. To summarize, we have a pair of adjoint natural transformations



A restriction of the Grothendieck construction [Bor94a] defines a 2-functor

$$\int \colon [\mathsf{Set}, \mathsf{Cat}] \to \mathsf{CAT}$$

where CAT is the 2-category of large categories. Because every 2-functor preserves adjunctions, the above diagram maps to an adjunction

$$\int A \underset{\int \mathbf{U}}{\overset{\int \mathbf{F}}{\rightleftarrows}} \int B.$$

The result follows from the equivalences $\int A \cong \mathsf{RMat}$ and $\int B \cong \mathsf{RCat}$. The desired functors F and U are obtained by composing $\int \mathbf{F}$ and $\int \mathbf{U}$ with these equivalences. $\square$

We conclude this chapter with a property of the above adjunction which will be useful in the next chapter.

Proposition 82.

$$\mathsf{RMat} \underset{U}{\overset{F}{\rightleftarrows}} \mathsf{RCat}. \quad (\bot)$$

is an idempotent adjunction.

Proof. Every adjunction between posets is idempotent. Therefore the smaller adjunctions $F_X \dashv U_X$ are idempotent. Because F and U are stitched together using these adjunctions, it is idempotent as well. □

Chapter 6

Compositionality of the Algebraic Path Problem

In this chapter we show how the algebraic path problem extends to the syntax of open R-matrices, i.e., R-matrices equipped with input and output nodes. In Section 6.1, we define open R-matrices and construct a symmetric monoidal double category $\mathsf{Open}(\mathsf{RMat})$ whose horizontal composition is gluing of open R-matrices. In Section 6.2, we show that the algebraic path problem functor

$$F\colon \mathsf{RMat} \to \mathsf{RCat}$$

of Theorem 3.6 lifts to a symmetric monoidal double functor

$$\mathsf{Open}(F)\colon \mathsf{Open}(\mathsf{RMat}) \to \mathsf{Open}(\mathsf{RCat}).$$

This double functor describes the compositionality of solutions to the algebraic path problem with respect to gluing of open R-matrices. In Section 6.3, we define the black-boxing $\blacksquare(C)$

of an open R-category $C\colon X \to Y$. $\blacksquare(C)$ is a R-matrix whose values contain only the entries of C which go from input nodes to output nodes. In Theorem 16 we show that the black-boxing functor is in general laxly functorial. However, in Theorem 6.3 we show that black-boxing is strictly functorial on functional open R-matrices, i.e. R-matrices for which every input is a source and every output is a sink. This gives a useful expression for solving the algebraic path problem compositionally.

6.1 Open R-Matrices

R-matrices are made open by designating some of their vertices to be either inputs or outputs. In this section we show how these open R-matrices are composed by gluing the output vertices of one to the input vertices of another and adding the R-matrices on the overlap. To define open R-matrices, we need a notion of a discrete weighted matrix on a set. The map sending a set to its discrete R-matrix is a functor and a left adjoint.

Proposition 83. *Let $R\colon \mathsf{RMat} \to \mathsf{Set}$ be the functor which sends a weighted graph to its underlying set of vertices and sends a morphism to its underlying function. Then R has a left adjoint*

$$0\colon \mathsf{Set} \to \mathsf{RMat}$$

which sends a set X to the R-weighted graph

$$0_X\colon X \times X \to Y$$

defined by $0_X(i,j) = 0$ for all i and j in X. F sends a function $f\colon X \to Y$ to the morphism of R-matrices which has f as its underlying function between vertices.

Proof. The natural isomorphism

$$\phi \colon \mathsf{RMat}(0_X, G) \cong \mathsf{Set}(X, R(G))$$

is formed by noting that a morphism $0_X \to R(G)$ is uniquely determined by its underlying function on vertices and every such function obeys the inequality in Definition 80. □

A weighted graph can be opened up to its environment by equipping it with inputs and outputs.

Definition 84. Let $M : A \times A \to R$ be an R-matrix. An **open R-matrix** $M \colon X \to Y$ is a cospan in RMat of the form

$$\begin{array}{ccccc} & & M & & \\ & \nearrow & & \nwarrow & \\ 0_X & & & & 0_Y . \end{array}$$

The idea is that the maps of this cospan point to input and output nodes of the matrix M.

We can compose open R-matrices using the theory of structured cospans; concretely the composition has the following more elementary description. Let $M \colon X \to Y$ and $N \colon Y \to Z$

$$\begin{array}{ccccccccc} & & M & & & & N & & \\ & \nearrow & & \nwarrow & & \nearrow & & \nwarrow & \\ 0_X & & & & 0_Y & & & & 0_Z \end{array}$$

be open R-matrices. The underlying sets of M and N form a diagram

$$\begin{array}{ccccccccc} & & R(M) & & & & R(N) & & \\ & \overset{l}{\nearrow} & & \overset{m}{\nwarrow} & & \overset{n}{\nearrow} & & \overset{o}{\nwarrow} & \\ X & & & & Y & & & & Z \end{array}$$

which generate a pushout

$$\begin{array}{ccccc} & & R(M)+_Y R(N) & & \\ & \overset{a}{\nearrow} & & \overset{b}{\nwarrow} & \\ R(M) & & & & R(N) \\ & \underset{m}{\nwarrow} & & \underset{n}{\nearrow} & \\ & & Y & & \end{array}$$

The functions a and b of this pushout allow the matrices M and N to be compared on equal footing: the pushforwards $a_*(M)$ and $b_*(N)$ both have $R(M) +_Y R(N)$ as their underlying set. The matrices $a_*(M)$ and $b_*(N)$ are combined using pointwise sum.

Definition 85. For open R-matrices $M : X \to Y$ and $N : Y \to Z$ as defined above, their **composite** is defined by

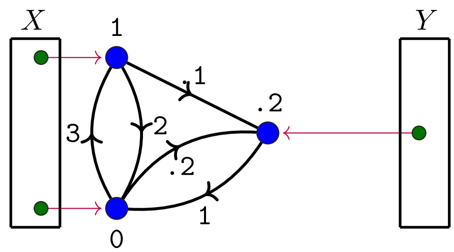
$$N \circ M : X \to Z = \begin{array}{ccccc} & & a_*(M) + b_*(N) & & \\ & \overset{\phi^{-1}(a \circ l)}{\nearrow} & & \overset{\phi^{-1}(b \circ r)}{\nwarrow} & \\ LX & & & & LZ \end{array}$$

where ϕ^{-1} gives the unique morphism out of a discrete R-matrix defined by a function on its underlying set.

An R-matrix $M : X \times X \to R$ can represent a graph with vertex set X weighted in R. Similarly, an open R-matrix, represents an R-weighted graph equipped with inputs and outputs. For example, let R be the quantale $[0, \infty]$ where the addition is infimum and the multiplication is ordinary addition. Then the $[0, \infty]$-matrix

$$\begin{bmatrix} 1 & 2 & .1 \\ 3 & 0 & .2 \\ \infty & 1 & .2 \end{bmatrix}$$

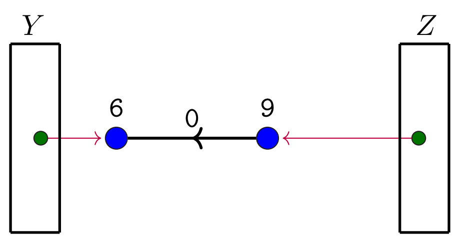
on the set $\{a, b, c\}$ can be regarded as on open $[0, \infty]$-matrix with left input set $\{1, 2\}$ and right input set $\{3\}$. The mappings of the cospan are given by $1 \mapsto a$, $2 \mapsto b$ and $3 \mapsto c$. This can be drawn as an open weighted graph



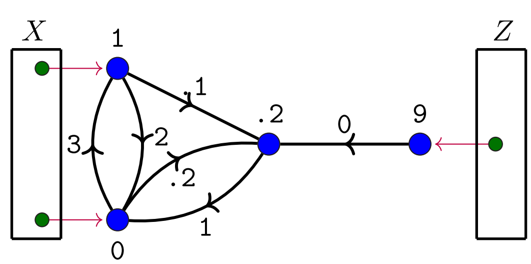
where unlabeled edges are assumed to have a value of ∞. Similarly, we define an open $[0, \infty]$-matrix on $\{d, e\}$

$$\begin{bmatrix} 6 & \infty \\ 0 & 9 \end{bmatrix}$$

with left input set given by $\{3\}$ and right input set given by $\{4\}$. The mappings in the cospan for this open $[0, \infty]$-matrix are given by the assignments $3 \mapsto d$ and $4 \mapsto e$. This open $[0, \infty]$-matrix is drawn as



The composite of these two $[0, \infty]$-matrices is represented by



where edges are omitted if their weight is infinite. The matrix on the apex of this composite is computed by pushing each component matrix forward to the pushout of their underlying sets and adding them together i.e.

$$\begin{bmatrix} 1 & 2 & .1 & \infty \\ 3 & 0 & .2 & \infty \\ \infty & 1 & .2 & \infty \\ \infty & \infty & \infty & \infty \end{bmatrix} + \begin{bmatrix} \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty \\ \infty & \infty & 6 & \infty \\ \infty & \infty & 0 & 9 \end{bmatrix} = \begin{bmatrix} 1 & 2 & .1 & \infty \\ 3 & 0 & .2 & \infty \\ \infty & 1 & .2 & \infty \\ \infty & \infty & 0 & 9 \end{bmatrix}$$

The entries of this matrix represent the shortest distance between pairs of vertices. Next we construct a double category where horizontal morphisms are open R-matrices.

Theorem 86. *For a quantale R, there is a symmetric monoidal double category* $\mathsf{Open}(\mathsf{RMat})$ *where*

- *objects are sets X,Y,$Z\ldots$*
- *vertical morphisms are functions $f\colon X \to Y$,*
- *a horizontal morphism $M\colon X \to Y$ is an open R-matrix*

$$\begin{array}{ccc} & M & \\ \nearrow & & \nwarrow \\ 0_X & & 0_Y \end{array}$$

- *vertical 2-morphisms are commutative rectangles*

$$\begin{array}{ccccc} 0_X & \longrightarrow & M & \longleftarrow & 0_Y \\ {\scriptstyle 0_f}\downarrow & & \downarrow{\scriptstyle g} & & \downarrow{\scriptstyle 0_h} \\ 0'_Y & \longrightarrow & N & \longleftarrow & 0'_Y \end{array}$$

- *vertical composition is ordinary composition of functions,*
- *and horizontal composition is given by the composite operation defined above.*

The symmetric monoidal structure is given by

- *coproducts in* Set *on objects and vertical morphisms,*

- *pointwise coproducts on horizontal morphisms i.e. for open R-matrices,*

$$
\begin{array}{ccccc}
 & M & & & M' & \\
 \nearrow & & \nwarrow & \nearrow & & \nwarrow \\
0_X & & 0_Y \quad 0'_X & & & 0'_Y
\end{array}
$$

their coproduct is

$$
\begin{array}{ccc}
 & M \sqcup M' & \\
 \nearrow & & \nwarrow \\
0_{X \sqcup X'} & & 0_{Y \sqcup Y'}
\end{array}
$$

where $\sqcup$ indicates the coproduct in RMat*. For vertical 2-morphisms,*

$$
\begin{array}{ccccc}
0_X & \longrightarrow & M & \longleftarrow & 0_Y \\
{\scriptstyle 0_f}\downarrow & & \downarrow{\scriptstyle g} & & \downarrow{\scriptstyle 0_h} \\
0_Z & \longrightarrow & N & \longleftarrow & 0_Q
\end{array}
\qquad
\begin{array}{ccccc}
0'_X & \longrightarrow & M' & \longleftarrow & 0'_Y \\
{\scriptstyle 0'_f}\downarrow & & \downarrow{\scriptstyle g'} & & \downarrow{\scriptstyle 0'_h} \\
0'_Z & \longrightarrow & N' & \longleftarrow & 0'_Q
\end{array}
$$

their coproduct is

$$
\begin{array}{ccccc}
0_{X \sqcup X'} & \longrightarrow & M \sqcup M' & \longleftarrow & 0_{Y \sqcup Y'} \\
{\scriptstyle 0_{f \sqcup f'}}\downarrow & & \downarrow{\scriptstyle g \sqcup g'} & & \downarrow{\scriptstyle 0_{h \sqcup h'}} \\
0_{Z \sqcup Z'} & \longrightarrow & N \sqcup N' & \longleftarrow & 0_{Q \sqcup Q'}.
\end{array}
$$

Proof. Lemma 7 constructs this symmetric monoidal double category as long as

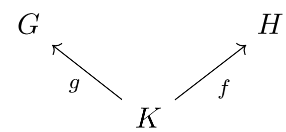
- RMat has finite coproducts and pushouts,

- and $0\colon \mathsf{Set} \to \mathsf{RMat}$ preserves pushouts and coproducts.

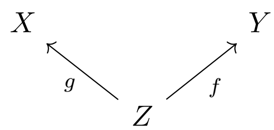
Because 0 is a left adjoint (Proposition 83) it preserves pushouts and coproducts when they exist so it suffices to prove the following lemma. □

Lemma 87. RMat *has coproducts and pushouts.*

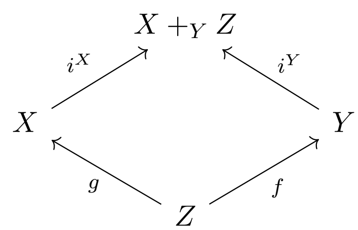
Proof. This is a consequence of Proposition 2.4 of [Wol74] after noting that RMat is the category of R-graphs, the generating data for R-enriched categories. For concreteness and practicality, we offer an explicit construction of pushouts and coproducts here. Let



be a diagram in RMat with



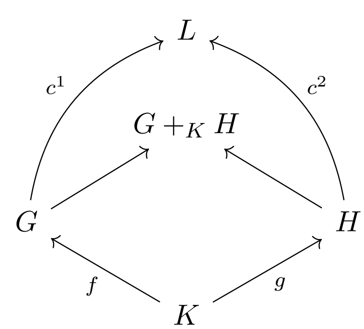
as the underlying diagram of sets. To compute the pushout $G +_K H$ first we take the pushout of sets



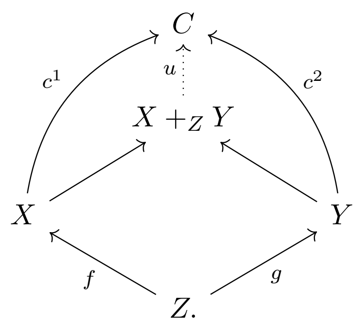
push them forward to get matrices $i^X_*(G)$ and $i^Y_*(H)$ and add them together to get

$$G +_Y H\colon (X +_Y Z) \times (X +_Y Z) \to R = i^X_*(G) + i^Y_*(H).$$

This does indeed define a pushout in RMat. Suppose we have a commutative diagram of R-matrices as follows:



then the underlying diagram of sets induces a unique function u



commuting suitably with c^1 and c^2. The map u is certainly unique, it remains to show that it is well-defined i.e. it satisfies the inequality

$$u_*(G +_K H) \leq L.$$

Indeed, for $(x, y) \in C \times C$,

$$\begin{aligned}
u_*(G +_K H)(x, y) &= \sum_{(a,b)\in(u\times u)^{-1}(x,y)} (G +_K H)(a, b) \\
&= \sum_{(a,b)\in(u\times u)^{-1}(x,y)} i^X_*(G)(a, b) + i^Y_*(H)(a, b) \\
&= \sum_{(a,b)\in(u\times u)^{-1}(x,y)} i^X_*(G)(a, b) + \sum_{(a,b)\in(u\times u)^{-1}(x,y)} i^Y_*(H)(a, b) \\
&= u_*(i^X_*(G))(x, y) + u_*(i^Y_*(H))(x, y)
\end{aligned}$$

However, because

$$u_*(i^X(G)) = c^1_*(G) \text{ and } u_*(i^Y(H)) = c^2_*(G)$$

the above expression is equal to

$$c^1_*(G)(x, y) + c^2_*(H)(x, y)$$

which is less than or equal to $L(x, y)$ because each term is and $+$ is the least upper bound.

For R-matrices $G\colon X \times X \to R$ and $H\colon Y \times Y \to R$, their coproduct is given by the pushout

$$\begin{array}{ccccc} & & G +_{\phi} H & & \\ & \nearrow & & \nwarrow & \\ G & & & & H \\ & \nwarrow^{!_G} & & \nearrow^{!_H} & \\ & & \phi & & \end{array}$$

where ϕ is the initial object of RMat, i.e., the unique R-matrix on the empty set, and $!_G$ and $!_H$ are the unique morphisms into G and H respectively. $\square$

6.2 Compositional Semantics of the Algebraic Path Problem

Computing the solution to the algebraic path problem on an R-matrix G suffers from combinatorial explosion when the size of G grows very large. Therefore, efficient strategies to compute the algebraic path problem must break down large matrices into small pieces, compute the algebraic path problem on each piece, and then combine those solutions together. This strategy can be understood using structured cospans. Suppose that an R-matrix G is divided into open R-matrices

$$\begin{array}{ccccccc} & & M & & & N & & \\ & \nearrow & & \nwarrow \quad \nearrow & & & \nwarrow & \\ 0_X & & & 0_Y & & & & 0_Z \end{array}$$

sharing a common boundary Y. We may then apply the algebraic path problem functor F to get two composable cospans of R-categories

$$\begin{array}{ccccccc} & & F(M) & & & F(N) & & \\ & \nearrow & & \nwarrow \quad \nearrow & & & \nwarrow & \\ 1_X & & & 1_Y & & & & 1_Z. \end{array}$$

The pushout in RMat, $UF(M) +_{1_Y} UF(N)$, is not equal to the solution $F(M +_{0_Y} N)$. The former optimizes over only paths that are the composite of a path in M and a path in N. On the other hand, $F(M +_{0_Y} N)$ optimizes over paths that may zig-zag back and forth between M and N as many times as they like before arriving at their destination. Therefore, to construct $F(M +_{0_Y} N)$ from its components we turn to the pushout in RCat.

Proposition 88. RCat *has pushouts and coproducts.*

Proof. More generally, RCat has all colimits by Corollary 2.14 of [Wol74]. These colimits are constructed via the transfinite construction of free algebras [Kel80]. The idea behind the transfinite construction is that colimits in a category of monoids can be constructed by first taking the colimit of their underlying objects, taking the free monoid on that colimit, and then quotienting out by the equations in your original monoids. □

Next we provide an explicit description of colimits in RCat.

Proposition 89. *For a diagram* $D\colon C \to \mathsf{RCat}$*, its colimit is given by the formula*

$$\mathrm{colim}_{c\in C} D(c) \cong F(\mathrm{colim}_{c\in C} U(D(c)))$$

Proof. It suffices to show that $F(\mathrm{colim}_{c\in C}\, U(D(c)))$ satisfies the universal property of $\mathrm{colim}_{c\in C}\, D(c)$. Let $\alpha\colon \Delta_d \Rightarrow D$ be a cocone from an object $d \in \mathsf{RCat}$ to our diagram D. Because α can be regarded as a cocone in RMat, the universal property of colimits induces a unique map

$$\mathrm{colim}_{c\in C}\, U(D(c)) \to U(d)$$

of R-matrices. Applying F to this morphism gives a map

$$F(\mathrm{colim}_{c\in C}\, U(D(c))) \to FU(d) = d$$

where the last equality follows either from elementary considerations or from the adjunction $F \dashv U$ being idempotent. The above map is a unique morphism satsifying the universal property for $\operatorname{colim}_{c \in C} D(c)$. $\square$

Corollary 90. *For a diagram*

$$M \leftarrow K \rightarrow N$$

in RCat, *the pushout is given by*

$$M +_K N \cong F(U(M) +_{U(K)} U(N))$$

and the coproduct of R-categories is given by their coproduct in RMat *i.e.*

$$M \sqcup N \cong U(M) \sqcup U(N).$$

This pushout forms the horizontal composition of a double category of open R-categories.

Theorem 91. *There is a symmetric monoidal double category* $\mathsf{Open}(\mathsf{RCat})$ *where*

- *objects are sets,*
- *vertical morphisms are functions,*
- *horizontal morphisms are cospans*

$$1_X \rightarrow M \leftarrow 1_Y$$

where the apex M satisfies the axioms of an R-category,

- *and vertical 2-morphisms are commuting rectangles*

$$\begin{array}{ccccc} 1_X & \longrightarrow & M & \longleftarrow & 1_Y \\ {\scriptstyle 1_f}\downarrow & & \downarrow{\scriptstyle g} & & \downarrow{\scriptstyle 1_h} \\ 1'_X & \longrightarrow & N & \longleftarrow & 1'_Y \end{array}$$

- *The horizontal composition is given by pushout of open R-categories i.e. for open R-categories*

$$\begin{array}{ccccccccc} & & M & & & & N & & \\ & \nearrow & & \nwarrow & & \nearrow & & \nwarrow & \\ 1_X & & & & 1_Y & & & & 1_Z \end{array}$$

their pushout is the cospan

$$\begin{array}{ccccc} & & F(U(M) +_{U(K)} U(N)) & & \\ & \nearrow & & \nwarrow & \\ 1_X & & & & 1_Y \end{array}$$

The symmetric monoidal structure of $\mathsf{Open}(\mathsf{RCat})$ *is given by*

- *coproduct of sets and functions,*

- *pointwise coproduct on horizontal morphisms,*

- *and pointiwise coproduct on vertical 2-morphisms.*

Proof. To construct the desired symmetric monoidal double category, we apply Lemma 7 to the composite left adjoint

$$\mathsf{Set} \xrightarrow{0} \mathsf{RMat} \xrightarrow{F} \mathsf{RCat}.$$

We write the above composite as

$$1 \colon \mathsf{Set} \to \mathsf{RCat}$$

as it sends a set X to the identity matrix 1_X on X. □

So far we have the commutative diagram of functors

$$\begin{array}{ccc} \mathsf{RMat} & \xrightarrow{F} & \mathsf{RCat} \\ & {}_{0}\nwarrow \quad \nearrow_{1} & \\ & \mathsf{Set}. & \end{array}$$

The definition of Open is functorial with respect to this sort of diagram, i.e. it induces a symmetric monoidal double functor between the relevant double categories.

Theorem 92. *There is a symmetric monoidal double functor*

$$\bigstar\colon \mathsf{Open}(\mathsf{RMat}) \to \mathsf{Open}(\mathsf{RCat})$$

which is

- *the identity on objects and vertical morphisms,*

- *an open R-matrix*

$$M\colon X \to Y = \begin{array}{ccc} & M & \\ \nearrow & & \nwarrow \\ 0_X & & 0_Y \end{array}$$

  is sent to the solution of its algebraic path problem

$$\bigstar(M)\colon X \to Y = \begin{array}{ccc} & FM & \\ \nearrow & & \nwarrow \\ 1_X & & 1_Y, \end{array}$$

  and

- *a vertical 2-morphism of open R-matrices*

$$\alpha\colon M \Rightarrow N = \begin{array}{ccccc} 0_X & \longrightarrow & M & \longleftarrow & 0_Y \\ {\scriptstyle 0_f}\downarrow & & \downarrow{\scriptstyle g} & & \downarrow{\scriptstyle 0_h} \\ 0_Z & \longrightarrow & N & \longleftarrow & 0_Q \end{array}$$

  is sent to the 2-morphism given by pointwise application of F

$$\bigstar(\alpha)\colon M \Rightarrow N = \begin{array}{ccccc} 1_X & \longrightarrow & FM & \longleftarrow & 1_Y \\ {\scriptstyle 1_f}\downarrow & & \downarrow{\scriptstyle Fg} & & \downarrow{\scriptstyle 1_h} \\ 1'_X & \longrightarrow & FN & \longleftarrow & 1'_Y. \end{array}$$

Proof. We apply Lemma 11 to the square

$$\begin{array}{ccc} \mathsf{RMat} & \xrightarrow{F} & \mathsf{RCat} \\ {\scriptstyle 0}\uparrow & & \uparrow{\scriptstyle 1} \\ \mathsf{Set} & = & \mathsf{Set} \end{array}$$

to obtain the desired double functor. Because F preserves pushouts, this double functor preserves horizontal composition and monoidal product up to isomorphism. □

The definition of symmetric monoidal double functor packages up a lot of information very succinctly. In particular, it contains a coherent comparison isomorphism relating the solution of the algebraic path problem on a composite matrix to the solution on its components. For open R-matrices $M : X \to Y$ and $N : Y \to Z$, there is a composition comparison

$$\phi_{MN} \colon \bigstar(M) \circ \bigstar(N) \xrightarrow{\sim} \bigstar(M \circ N) \tag{6.1}$$

and monoidal comparison

$$\psi_{MM'} \colon \bigstar(M + M') \xrightarrow{\sim} \bigstar(M) + \bigstar(M') \tag{6.2}$$

giving recipes to break solutions to the algebraic path problem into their components. In other words, the left-hand side of each comparison is computed to determine the right-hand side

Pouly and Kohlas present a similar relationship in the context of valuation algebras [PK12, §6.7]. For matrices M and N representing weighted graphs on vertex sets s and t respectively, the solution to the algebraic path problem on the union of their vertex sets is given by

$$F(M) \otimes F(N) = F\left(F(M)^{\uparrow s \cup t} + F(N)^{\uparrow s \cup t}\right)$$

In this formula, $\uparrow s \cup t$ indicates that the matrix is trivially extended to the union of the vertex sets. This formula is less general than comparison 6.1: it corresponds to the special case when the legs of the open R-matrices are inclusions.

A typical algorithm for the algebraic path problem has spatial complexity $\Theta(n^3)$ where n is the number of vertices in your weighted graph [HM12]. The comparisons 6.1 and 6.2 suggests a faulty strategy for computing the solution to the algebraic path problem which reduces this complexity. First cut your weighted graph into smaller chunks, compute the solution to the algebraic path problem on those chunks, then combine their solutions using 6.1 and 6.2. Unfortunately, this strategy will in general to take *more* time to compute the solution to the algebraic path problem on a composite because the right hand side of comparison 6.1 requires three applications of the functor F. However, the situation improves if the open R-matrices are functional.

6.3 Functional Open Matrices

In this section we define functional open R-matrices, a class of open R-matrices for which the composition comparison

$$\phi_{MN} \colon \bigstar(M) \circ \bigstar(N) \xrightarrow{\sim} \bigstar(M \circ N)$$

can be expressed in terms of matrix multiplication. The one caveat is that this expression requires that the open matrices be restricted to their inputs and outputs as follows.

Definition 93. Let $M \colon X \to Y$ be the open R-category

$$1_X \xrightarrow{i} M \xleftarrow{o} 1_Y.$$

Then the **black-boxing** of M is the matrix

$$\blacksquare(M) \colon X \times Y \to R$$

given by

$$\blacksquare(M)(x,y) = M(i(x), o(y)).$$

At first the relationship between this black-boxing and the black-boxing of Theorem 2.4 and Theorem 66 may be opaque. In these theorems we considered black-boxings as profunctors

$$P \colon A \times B \to \mathsf{Set}$$

where A and B are discrete categories containing only identity morphisms. When enriching in the quantale R, we replace Set with R to obtain a function

$$P \colon A \times B \to R.$$

In general R-enriched profunctors must satisfy axioms expressing compatibility with R-category structure of A and B. However, because A and B are discrete, these axioms become trivial and the above R-enriched profunctor is exactly the same as a matrix valued in R. Furthermore, as shown in [Mas20b], this correspondence embeds the category of R-matrices into the category R-enriched profunctors. In this thesis we require a double category of R-matrices to match the double categories defined earlier in this chapter.

Definition 94. Let Mat_R be the double category where

- an object is a set X,Y,Z,...
- a vertical morphism is a function $f \colon X \to Y$,

- a horizontal morphism $M\colon X \to Y$ is a matrix $M\colon X \times Y \to R$,

- a vertical 2-morphism from $M\colon X \to Y$ to $N\colon X' \to Y'$ is a square

$$\begin{array}{ccc} X & \xrightarrow{M} & Y \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle g} \\ X' & \xrightarrow[N]{} & Y' \end{array}$$

  such that

$$\sum_{x \in f^{-1}(x'),\, y \in g^{-1}(y')} M(x,y) \leq N(x',y')$$

  for all $x' \in X'$ and $y' \in Y'$,

- vertical composition is function composition, and

- horizontal composition is given by matrix multiplication.

In this double category, the composite of matrices M and N is written as the juxtaposition MN. Black-boxing is extended to the double category of open R-categories.

Theorem 95. *There is a lax double functor*

$$\blacksquare\colon \mathsf{Open}(\mathsf{RCat}) \to \mathsf{Mat}_R$$

which

- *is the identity on objects,*

- *sends an open R-category $M\colon X \to Y$ to its black-boxing $\blacksquare(M)$, and*

- *sends a vertical 2-cell*

$$\begin{array}{ccccc} 1_X & \longrightarrow & M & \longleftarrow & 1_Y \\ {\scriptstyle 1_f}\downarrow & & \downarrow{\scriptstyle g} & & \downarrow{\scriptstyle 1_h} \\ 1_{X'} & \longrightarrow & N & \longleftarrow & 1_{Y'} \end{array}$$

to the vertical 2-cell

$$\begin{array}{ccc} X & \xrightarrow{\blacksquare(M)} & Y \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle g} \\ X' & \xrightarrow[\blacksquare(N)]{} & Y'. \end{array}$$

Proof. First observe that this lax double functor is well-defined on 2-cells. This amounts to showing that the inequality

$$\sum_{x \in f^{-1}(x'),\, y \in h^{-1}(y')} M(i(x), j(y)) \leq N(i'(x'), j'(y')) \tag{6.3}$$

holds. Because g is a morphism of R-matrices, we have that

$$\sum_{a \in g^{-1}(i'(x')),\, b \in g^{-1}(j'(y'))} M(a, b) \leq N(i'(x'), j'(y')) \tag{6.4}$$

Let $M(i(x), j(y))$ be a term on the left hand side of inequality 6.3. Then by definition, $x' = f(x)$ and $y' = h(y)$ so $a \in g^{-1}(i'(f(x))$ and $b \in g^{-1}(j'(h(y))$. However, because we started with a 2-cell in $\mathsf{Open}(\mathsf{RCat})$, $i' \circ f = g \circ i$ and $j' \circ h = g \circ j$ so we can rewrite inequality 6.4 as

$$\sum_{a \in g^{-1}(g \circ i(x)),\, b \in g^{-1}(g \circ j(y))} M(a, b) \leq N(i'(x'), j'(y'))$$

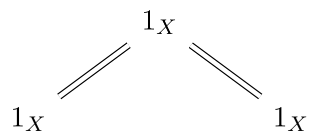
The term $M(i(x), j(y))$ of the left hand side of inequality 6.3 is also a term of the left hand side of inequality 6.4 so we have that

$$M(i(x), i(y)) \leq \sum_{a \in g^{-1}(g \circ i(x)),\, b \in g^{-1}(g \circ j(y))} M(a, b) \leq N(i'(x'), j'(y'))$$

Because each term on the left hand side of 6.3 is less than the desired quantity, the join of all the terms will be as well. Therefore the lax double functor is well-defined on 2-cells. Note that Mat_R is locally posetal, i.e. for every square

$$\begin{array}{ccc} X & \xrightarrow{M} & Y \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle g} \\ X' & \xrightarrow[N]{} & Y' \end{array}$$

there is at most one 2-cell filling it. This property makes it so many of the axioms in the definition of lax double functor are satisfied trivially. It suffices to show that the globular composition and identity comparisons exist. The identity morphism in $\mathsf{Open}(\mathsf{RCat})$ on a set X is the cospan



The black-box of this cospan is equal to the identity matrix on X, so the identity comparison is the identity. The composition comparison

$$\blacksquare(M)\blacksquare(N) \leq \blacksquare(M \circ N)$$

follows from the chain of inequalities

$$\begin{aligned}
\blacksquare(M)\blacksquare(N) &= \sum_{y \in Y} \blacksquare(M)(x,y)\blacksquare(N)(y,z) \\
&= \sum_{y \in Y} M(i(x),j(y))N(i'(y),j'(z)) \\
&= (M +_{1_Y} N)^2 \\
&\leq \sum_{n geq 0} (M +_{1_Y} N)^n(i(x),j'(z)) \\
&= \blacksquare(M \circ N)(x,z).
\end{aligned}$$

□

We can compose the black-boxing operation with the algebraic path problem functor to get a lax symmetric monoidal double functor

$$\mathsf{Open}(\mathsf{RMat}) \xrightarrow{\bigstar} \mathsf{Open}(\mathsf{RCat}) \xrightarrow{\blacksquare} \mathsf{Mat}_R$$

This lax symmetric monoidal double functor gives the solution to the algebraic path problem on an open R-matrix *restricted to its boundaries*. It is natural to ask when this mapping is strictly functorial, as this yields a very simple compositional formula for the algebraic path problem:

$$\blacksquare(\bigstar(M \circ N)) = \blacksquare(\bigstar(M))\blacksquare(\bigstar(N)).$$

The double functor $\blacksquare \circ \bigstar$ is strictly functorial on "functional" open R-matrices.

Definition 96. Let $M \colon A \times A \to R$ be an R-matrix. An element $a \in X$ is a **source** if for every $b \in X$, $M(b, a) = 0$ and a **sink** if $M(a, b) = 0$. A **functional open *R*-matrix** is an open R-matrix

$$\begin{array}{ccccc} & & M & & \\ & {}^{l}\nearrow & & \nwarrow^{r} & \\ 0_X & & & & 0_Y \end{array}$$

such that for every $x \in X$, $l(x)$ is a source and for every $y \in Y$, $r(y)$ is a sink.

Because the composite of functional open R-matrices is also functional, we can form the following sub-double category.

Definition 97. Let $\mathsf{Open}(\mathsf{RMat})_{fxn}$ be the full sub-symmetric monoidal double category generated by the open R-matrices which are functional.

Theorem 98. *The composite* $\blacksquare \circ \bigstar$ *restricts to a strict double functor*

$$\blacksquare \circ \bigstar_{fxn} \colon \mathsf{Open}(\mathsf{RMat})_{fxn} \to \mathsf{Mat}_R$$

The proof of this theorem relies on a lemma which resembles the the binomial expansion of $(a + b)^n$ in the case when $ba = 0$. If a and b represent black-boxes of functional open matrices, then the identity $ba = 0$ indicates that there are no paths which go backwards.

Lemma 99. *For functional open R-matrices $M\colon X \to Y$ and $N\colon Y \to Z$ we have that*

$$\blacksquare(M +_{1_Y} N)^n = \sum_{i+j=n} \blacksquare(M^i)\blacksquare(N^j).$$

Proof. The entries of the left hand side are expanded as

$$\blacksquare((M+_{1_Y}N)^n)(a_0, a_n) = \sum_{a_1,a_2,\dots,a_{n-1}} (M+_{1_Y}N)(a_0, a_1)(M+_{1_Y}N)(a_1, a_2)\dots(M+_{1_Y}N)(a_{n-1}, a_n)$$

where the a_i are equivalence classes in $RM +_Y RN$. For a particular term of this sum, let $1 \le k \le n$ be the first natural number such that a_k contains an element of RN. Because M and N are functional, for $k \le i \le n$ the equivalence classes a_i must also contain an element of RN if our term is nonzero. Therefore for a fixed k the contribution to the above sum is given by

$$\sum M(a_0, a_1)\dots M(a_{k-1}, a_k)N(a_k, a_{k+1})\dots N(a_{n-1}, a_n)$$

which simplifies to

$$\blacksquare(M^k)\blacksquare(N^{n-k})(a_0, a_n).$$

Because k can occur in any entry we have that

$$\begin{aligned}\blacksquare((M +_{1_Y} N)^n) &= \sum_{k\le n} \blacksquare(M^k)\blacksquare(N^{n-k}) \\ &= \sum_{i+j=n} \blacksquare(M^i)\blacksquare(N^j)\end{aligned}$$

□

Proof of Theorem 98: It suffices to prove that for functional open matrices

$$0_X \longrightarrow M \longleftarrow 0_Y$$

and

$$0_Y \longrightarrow N \longleftarrow 0_Z$$

the equation

$$\blacksquare(\bigstar(M \circ N)) = \blacksquare(\bigstar(M))\blacksquare(\bigstar(N))$$

holds. Consider the left-hand side:

$$\begin{aligned}
\blacksquare(\bigstar(M \circ N)) &= \blacksquare \sum_{n \geq 0} (M \circ N)^n \\
&= \sum_{n \geq 0} \blacksquare(M \circ N)^n \\
&= \sum_{n \geq 0} \sum_{i+j=n} \blacksquare(M^i)\blacksquare(N^j).
\end{aligned}$$

where the third step uses Lemma 99. On the other hand,

$$\begin{aligned}
\blacksquare(\bigstar(M))\blacksquare(\bigstar(N)) &= \sum_{i \geq 0} \blacksquare(M^i) \sum_{j \geq 0} \blacksquare(N^j) \\
&= \sum_{i,j \geq 0} \blacksquare(M^i)\blacksquare(M^j)
\end{aligned}$$

Both sums contain the term $\blacksquare(M^i)\blacksquare(N^j)$ for every value of i and j, but the left hand side may contain repeated terms. However, because addition is idempotent, repeated terms don't contribute to the sum and the two sides are the same. □

The functoriality of Theorem 98 might not be surprising. It says that if your open matrices are joined together directionally along bottlenecks, then the computation of the algebraic path problem can be reduced to a computation on components. This strategy has already proven sucessful. In [STV95], Sairam, Tamassia, and Vitter show how choosing *one* *way separators* as cuts in a graph, allow for an efficient divide and conquer parallel algorithm for computing shortest paths. In [RSS14] Rathke, Sobocinksi, and Stephens show how the reachability problem on a 1-safe Petri net can be computed more efficiently by cutting it up into more manageable pieces. Theorem 92 provides a framework for compositional formulas of this type. In future work we plan on extending the construction of this theorem to many other sorts of discrete event dynamic systems.

Lemma 99 also holds independent computational interest. The equation given there gives a novel compositional formula for computing the solution to the algebraic path problem. The author has implemented this formula for the special case of Markov processes [Mas20a]. We hope that this is the start of a more extensive library, made faster and more reliable by the mathematics developed in this chapter.

Chapter 7

Conclusion

There are a few directions of research which would make this thesis more complete:

- Enriched graphs are only considered when the enriching category is a quantale. More generally, the theory of this thesis could be developed for graphs enriched in a monoidal closed category $(V, \otimes)$ with all colimits. Quantales are a particularly simple example of these, and we are excited about the possibility of enriching in categories that are not posets. For example, we may enrich in the category $(\mathsf{Grph}, \times)$. In this case a graph enriched in Grph with vertex set X may be regarded as a function

  $$M \colon X \times X \to \mathsf{Grph}.$$

  For a pair of vertices $x, y \in X$, $M(x, y)$ is a graph whose vertices represent different ways of turning x into y. The edges of $M(x, y)$ may represent higher order relationships between the vertices. An operational semantics and compositional theory may be developed for Grph-enriched graphs which is similar the theory developed in this thesis.

- To define Q-nets we used the finitary monad Q induces on Set. This could be made more general by considering an arbitrary monad on an arbitrary category. For example, let Meas be the category where objects are sets equipped with a σ-algebra and morphisms are σ-algebra preserving functions. Let $G\colon \mathsf{Meas} \to \mathsf{Meas}$ be the Giry monad defined in [Gir82]. For a measurable set (X, Σ), $G((X, \Sigma))$ is the measurable space of probability measures $\mu\colon \Sigma \to [0, 1]$ on (X, Σ). G is not a finitary monad and therefore does not come from a Lawvere theory. However, we may still define a G-net to be a pair of functions

$$T \underset{t}{\overset{s}{\rightrightarrows}} G(S)$$

where T and S are measurable spaces. For a transition $\tau \in T$, the probability distributions $s(\tau)$ and $t(\tau)$ may represent uncertainty about the pre- and post-conditions of the event represented by τ. We may attempt to develop an operational semantics for G-nets by turning them into free categories internal to the category of G-algebras. However, it remains to be seen whether or not this operational semantics is relevant to their natural interpretation.

- In Section 3.2, the functoriality of the definition of Q-net is explored in detail to understand relationships between different variants of Q-nets. The same could be done for R-matrices. A morphism of quantales $f : R \to S$ lifts to a functor between categories $\mathsf{Mat}_R \to \mathsf{Mat}_S$ giving a functorial way to translate between matrices with different weights. There is work to be done to understand how these change of enrichment functors relate their corresponding solutions to the algebraic path problem.

- The black-boxing double functors of Theorems 16, 66, and 95 may be upgraded to symmetric monoidal double functors. This reflects the fact that black-boxing commutes with placing open networks in parallel.

In general, this thesis aims to provide a setting for reasoning about the compositionality of networks, but leaves most of that reasoning to future work. As shown in Example 17, an infinitely large operational operational semantics may arise from the composite of very small networks. We believe that the best way to study this sort of emergence is to build up to it slowly. Functional open networks are intended to start at the bottom, i.e. they are open networks for which the behavior on a composite may be entirely derived from the behavior on its components as shown in Theorems 20, 68, and 98. We may attempt to prove similar theorems for networks which do exhibit emergent behavior when composed. This may be easier when we choose an operational semantics which is bounded or restricted in some way. For example, we may consider an operational semantics of networks containing paths with a length at most n. It is an open question whether or not this operational semantics gives a double functor whose domain is a structured cospan double category of open networks.

Appendix A

Double Categories

What follows is a brief introduction to double categories. A more detailed exposition can be found in the work of Grandis and Paré [GP99, GP04], and for monoidal double categories the work of Shulman [SH19]. We use 'double category' to mean what earlier authors called a 'pseudo double category'.

Definition 100. A **double category** is a category weakly internal to Cat. More explicitly, a double category $\mathbb{D}$ consists of:

- a **category of objects** $\mathbb{D}_0$ and a **category of arrows** $\mathbb{D}_1$,
- **source** and **target** functors

  $$S, T \colon \mathbb{D}_1 \to \mathbb{D}_0,$$

  an **identity-assigning** functor

  $$U \colon \mathbb{D}_0 \to \mathbb{D}_1,$$

  and a **composition** functor

  $$\circ \colon \mathbb{D}_1 \times_{\mathbb{D}_0} \mathbb{D}_1 \to \mathbb{D}_1$$

where the pullback is taken over $\mathbb{D}_1 \xrightarrow{T} \mathbb{D}_0 \xleftarrow{S} \mathbb{D}_1$, such that

$$S(U_A) = A = T(U_A), \quad S(M \circ N) = SN, \quad T(M \circ N) = TM,$$

- natural isomorphisms called the **associator**

$$\alpha_{N,N',N''}\colon (N \circ N') \circ N'' \xrightarrow{\sim} N \circ (N' \circ N''),$$

the **left unitor**

$$\lambda_N\colon U_{T(N)} \circ N \xrightarrow{\sim} N,$$

and the **right unitor**

$$\rho_N\colon N \circ U_{S(N)} \xrightarrow{\sim} N$$

such that $S(\alpha), S(\lambda), S(\rho), T(\alpha), T(\lambda)$ and $T(\rho)$ are all identities and such that the standard coherence axioms hold: the pentagon identity for the associator and the triangle identity for the left and right unitor [ML98, Sec. VII.1].

If α, λ and ρ are identities, we call $\mathbb{D}$ a **strict** double category.

Objects of $\mathbb{D}_0$ are called **objects** and morphisms in $\mathbb{D}_0$ are called **vertical 1-morphisms**. Objects of $\mathbb{D}_1$ are called **horizontal 1-cells** of $\mathbb{D}$ and morphisms in $\mathbb{D}_1$ are called **2-morphisms**. A morphism $\alpha\colon M \to N$ in $\mathbb{D}_1$ can be drawn as a square:

$$\begin{array}{ccc} A & \xrightarrow{M} & B \\ {\scriptstyle f}\downarrow & \Downarrow \alpha & \downarrow{\scriptstyle g} \\ C & \xrightarrow[N]{} & D \end{array}$$

where $f = S\alpha$ and $g = T\alpha$. If f and g are identities we call α a **globular 2-morphism**. These give rise to a bicategory:

Definition 101. Let $\mathbb{D}$ be a double category. Then the **horizontal bicategory** of $\mathbb{D}$, denoted $H(\mathbb{D})$, is the bicategory consisting of objects, horizontal 1-cells and globular 2-morphisms of $\mathbb{D}$.

We have maps between double categories, and also transformations between maps:

Definition 102. Let $\mathbb{A}$ and $\mathbb{B}$ be double categories. A **double functor** $F\colon \mathbb{A} \to \mathbb{B}$ consists of:

- functors $F_0\colon \mathbb{A}_0 \to \mathbb{B}_0$ and $F_1\colon \mathbb{A}_1 \to \mathbb{B}_1$ obeying the following equations:

$$S \circ F_1 = F_0 \circ S, \qquad T \circ F_1 = F_0 \circ T,$$

- natural isomorphisms called the **composition comparison**:

$$\phi(N, N')\colon F_1(N) \circ F_1(N') \xrightarrow{\sim} F_1(N \circ N')$$

  and the **identity comparison**:

$$\phi_A\colon U_{F_0(A)} \xrightarrow{\sim} F_1(U_A)$$

  whose components are globular 2-morphisms,

such that the following diagram commmute:

- a diagram expressing compatibility with the associator:

$$\begin{array}{ccc}
(F_1(N) \circ F_1(N')) \circ F_1(N'') & \xrightarrow{\alpha} & F_1(N) \circ (F_1(N') \circ F_1(N'')) \\
\downarrow{\scriptstyle \phi(N,N')\circ 1} & & \downarrow{\scriptstyle 1\circ\phi(N',N'')} \\
F_1(N \circ N') \circ F_1(N'') & & F_1(N) \circ F_1(N' \circ N'') \\
\downarrow{\scriptstyle \phi(N\circ N',N'')} & & \downarrow{\scriptstyle \phi(N,N'\circ N'')} \\
F_1((N \circ N') \circ N'') & \xrightarrow{F_1(\alpha)} & F_1(N \circ (N' \circ N''))
\end{array}$$

- two diagrams expressing compatibility with the left and right unitors:

$$\begin{array}{ccc} F_1(N) \circ U_{F_0(A)} & \xrightarrow{\rho_{F_1(N)}} & F_1(N) \\ {\scriptstyle 1 \circ \phi_A}\downarrow & & \uparrow{\scriptstyle F_1(\rho_N)} \\ F_1(N) \circ F_1(U_A) & \xrightarrow{\phi(N, U_A)} & F_1(N \circ U_A) \end{array}$$

$$\begin{array}{ccc} U_{F_0(B)} \circ F_1(N) & \xrightarrow{\lambda_{F_1(N)}} & F_1(N) \\ {\scriptstyle \phi_B \circ 1}\downarrow & & \uparrow{\scriptstyle F_1(\lambda_N)} \\ F_1(U_B) \circ F_1(N) & \xrightarrow{\phi(U_B, N)} & F_1(U_B \circ N). \end{array}$$

If the 2-morphisms $\phi(N, N')$ and ϕ_A are identities for all $N, N' \in \mathbb{A}_1$ and $A \in \mathbb{A}_0$, we say $F\colon \mathbb{A} \to \mathbb{B}$ is a **strict** double functor. If on the other hand we drop the requirement that these 2-morphisms be invertible, we call F a **lax** double functor.

Definition 103. Let $F\colon \mathbb{A} \to \mathbb{B}$ and $G\colon \mathbb{A} \to \mathbb{B}$ be lax double functors. A **transformation** $\beta\colon F \Rightarrow G$ consists of natural transformations $\beta_0\colon F_0 \Rightarrow G_0$ and $\beta_1\colon F_1 \Rightarrow G_1$ (both usually written as β) such that

- $S(\beta_M) = \beta_{SM}$ and $T(\beta_M) = \beta_{TM}$ for any object $M \in \mathbb{A}_1$,
- β commutes with the composition comparison, and
- β commutes with the identity comparison.

Shulman defines a 2-category **Dbl** of double categories, double functors, and transformations [SH19]. This has finite products. In any 2-category with finite products we can define a pseudomonoid [DS97], which is a categorification of the concept of monoid. For example, a pseudomonoid in Cat is a monoidal category.

Definition 104. A **monoidal double category** is a pseudomonoid in **Dbl**. Explicitly, a monoidal double category is a double category equipped with double functors $\otimes\colon \mathbb{D}\times\mathbb{D}\to\mathbb{D}$ and $I\colon * \to \mathbb{D}$ where $*$ is the terminal double category, along with invertible transformations called the **associator**:

$$A\colon \otimes \circ (1_{\mathbb{D}} \times \otimes) \Rightarrow \otimes \circ (\otimes \times 1_{\mathbb{D}}),$$

left unitor:

$$L\colon \otimes \circ (1_{\mathbb{D}} \times I) \Rightarrow 1_{\mathbb{D}},$$

and **right unitor**:

$$R\colon \otimes \circ (I \times 1_{\mathbb{D}}) \Rightarrow 1_{\mathbb{D}}$$

satisfying the pentagon axiom and triangle axioms.

This definition neatly packages a large quantity of information. Namely:

- $\mathbb{D}_0$ and $\mathbb{D}_1$ are both monoidal categories.
- If I is the monoidal unit of $\mathbb{D}_0$, then U_I is the monoidal unit of $\mathbb{D}_1$.
- The functors S and T are strict monoidal.
- $\otimes$ is equipped with composition and identity comparisons

$$\chi\colon (M_1 \otimes N_1) \circ (M_2 \otimes N_2) \xrightarrow{\sim} (M_1 \circ M_2) \otimes (N_1 \circ N_2)$$

$$\mu\colon U_{A\otimes B} \xrightarrow{\sim} (U_A \otimes U_B)$$

  making three diagrams commute as in Def. 102.

- The associativity isomorphism for $\otimes$ is a transformation between double functors.

- The unit isomorphisms are transformations between double functors.

Definition 105. A **braided monoidal double category** is a monoidal double category equipped with an invertible transformation

$$\beta\colon \otimes \Rightarrow \otimes \circ \tau$$

called the **braiding**, where $\tau\colon \mathbb{D} \times \mathbb{D} \to \mathbb{D} \times \mathbb{D}$ is the twist double functor sending pairs in the object and arrow categories to the same pairs in the opposite order. The braiding is required to satisfy the usual two hexagon identities [ML98, Sec. XI.1]. If the braiding is self-inverse we say that $\mathbb{D}$ is a **symmetric monoidal double category**.

In other words:

- $\mathbb{D}_0$ and $\mathbb{D}_1$ are braided (resp. symmetric) monoidal categories,
- the functors S and T are strict braided monoidal functors, and
- the braiding is a transformation between double functors.

Definition 106. A **monoidal lax double functor** $F\colon \mathbb{C} \to \mathbb{D}$ between monoidal double categories $\mathbb{C}$ and $\mathbb{D}$ is a lax double functor $F\colon \mathbb{C} \to \mathbb{D}$ such that

- F_0 and F_1 are monoidal functors,
- $SF_1 = F_0 S$ and $TF_1 = F_0 T$ are equations between monoidal functors, and
- the composition and unit comparisons $\phi(N_1, N_2)\colon F_1(N_1) \circ F_1(N_2) \to F_1(N_1 \circ N_2)$ and $\phi_A\colon U_{F_0(A)} \to F_1(U_A)$ are monoidal natural transformations.

The monoidal lax double functor is **braided** if F_0 and F_1 are braided monoidal functors and **symmetric** if they are symmetric monoidal functors.

Appendix B

Lawvere Theories

Introduced by Lawvere in his landmark thesis [Law63], Lawvere theories are a general framework for reasoning about algebraic structures [BW85, Buc08].

Definition 107. A Lawvere theory Q is a small category with finite products such that every object is isomorphic to the iterated finite product $x^n = x \times \ldots \times x$ for a **generic object** x and natural number n. Equivalently, Lawvere theories can be thought of as categories whose objects are given by natural numbers $n \in \mathbb{N}$ and with cartesian product given by $+$. The morphisms in a Lawvere theory are called **operations**.

The idea is that a Lawvere theory represents the platonic embodiment of an algebraic gadget.

Example 108. A canonical example is the Lawvere theory MON of monoids. Like all Lawvere theories, the objects of MON are given by natural numbers. In addition MON contains the morphisms

$$m\colon 2 \to 1 \text{ and } e\colon 0 \to 1$$

For a monoid M, this represents the multiplication map

$$\cdot\colon M \times M \to M$$

and the map

$$e\colon \{*\} \to M$$

which picks out the identity element of M. These maps are required to satisfy the associative law

$$\begin{array}{ccc} 3 & \xrightarrow{\mathrm{id}\times m} & 2 \\ {\scriptstyle m\times\mathrm{id}}\downarrow & & \downarrow{\scriptstyle m} \\ 2 & \xrightarrow[m]{} & 1 \end{array}$$

and the unital laws for monoids.

$$\begin{array}{ccccc} 1 & \xrightarrow{\mathrm{id}\times e} & 2 & \xleftarrow{e\times\mathrm{id}} & 1 \\ & {\scriptstyle \mathrm{id}}\searrow & \downarrow{\scriptstyle m} & \swarrow{\scriptstyle \mathrm{id}} & \\ & & 1 & & \end{array}$$

MON also contains all composites, tensor products, and maps necessary to make n into the product x^n induced by the maps m and e.

Like all good things, Lawvere theories form a category.

Definition 109. Let *Law*-Cat be the category where objects are Lawvere theories and morphisms are product preserving functors.

Note that because morphisms of Lawvere theories preserve products, they must send the generic object of their source to the generic object of their target. Therefore to specify a morphism of Lawvere theories, it suffices to make an assignment of the morphisms which are not part of the product structure.

Let Q be a Lawvere theory and C a category with finite products. We can impose the axioms and operations of Q onto an object in C via a product preserving functor

$F\colon \mathsf{Q} \to C$. The image $F(1)$ of the generating object 1 gives the underlying object of F and for an operation $o\colon n \to k$ in Q, $F(o)\colon F(x)^n \to F(x)^k$ gives a specific instance of the algebraic operation represented by o. There is a natural way to make a category of these functors.

Definition 110. Let Q be a Lawvere theory and C a category with finite products. Then there is a category $\mathsf{Mod}(\mathsf{Q}, C)$ where

- objects are product preserving functors $F\colon \mathsf{Q} \to C$ and,
- morphisms are natural transformations between these functors.

When $\mathsf{Mod}(\mathsf{Q})$ is written without the second argument, it is assumed to be Set. We will refer to objects in $\mathsf{Mod}(\mathsf{Q})$ as **Q-models** and morphisms in $\mathsf{Mod}(\mathsf{Q})$ as **Q-model homomorphisms**. When $C = \mathsf{Cat}$, we will refer to these objects as **Q-categories**.

When the category of models is Set then there is a forgetful functor

$$R_{\mathsf{Q}}\colon \mathsf{Mod}(\mathsf{Q}) \to \mathsf{Set}$$

which sends a product preserving functor $F\colon \mathsf{Q} \to \mathsf{Set}$ to image on the generating object $F(1)$ and a natural transformation to component on the object 1. A classical result says that R_{Q} *always* has a left adjoint

$$L_{\mathsf{Q}}\colon \mathsf{Set} \to \mathsf{Mod}(\mathsf{Q})$$

which for a set X, $L_{\mathsf{Q}}X$ is referred to as the **free model of Q on X**. In fact, this construction extends to fully faithful functor

$$\mathsf{Law} \to \mathsf{Mnd}$$

which sends a Lawvere theory Q to the monad $R_{\mathsf{Q}} \circ L_{\mathsf{Q}} \colon \mathsf{Set} \to \mathsf{Set}$ and where Mnd is the category of monads on Set [EJL66]. For a Lawvere theory Q we will denote the monad it induces via this functor by $M_{\mathsf{Q}} \colon \mathsf{Set} \to \mathsf{Set}$.

For $\mathsf{Q} = \mathsf{MON}$, $\mathsf{Mod}(\mathsf{MON}, \mathsf{Set})$ is equivalent to the category Mon-Cat of monoids and monoid homomorphisms. In this case the functor $R_{\mathsf{MON}} \colon Mon\text{-}\mathsf{Cat} \to \mathsf{Set}$ turns monoids and monoid homomorphisms into their underlying sets and functions. R_{MON} has a left adjoint

$$L_{\mathsf{MON}} \colon \mathsf{Set} \to \mathsf{Mon}$$

which sends a set X to the free monoid $L_{\mathsf{MON}}X$. For a function $f \colon X \to Y$, $L_{\mathsf{MON}}f$ is the unique multiplication preserving extension of f to $L_{\mathsf{MON}}X$.

Bibliography

[BC18] John C. Baez and Kenny Courser. Coarse-graining open Markov processes. *Theory and Applications of Categories*, 33(39):1223–1268, 2018. Available at http://www.tac.mta.ca/tac/volumes/33/39/33-39abs.html. (Referred to on page 9.)

[BC20] John C. Baez and Kenny Courser. Structured cospans. *Theory and Applications of Categories*, 35(48):1771–1822, 2020. Available at http://www.tac.mta.ca/tac/volumes/35/48/35-48abs.html. (Referred to on page 25, 29, 30, 31.)

[BCP15] Massimo Bartoletti, Tiziana Cimoli, and G. Michele Pinna. Lending Petri nets. *Science of Computer Programming*, 112:75–101, 2015. Available at https://www.sciencedirect.com/science/article/pii/S0167642315001021. (Referred to on page 49, 50.)

[Bet96] Renato Betti. Formal theory of internal categories. *Le Matematiche*, 51(3):35–52, 1996. Available at https://lematematiche.dmi.unict.it/index.php/lematematiche/article/view/456. (Referred to on page 75.)

[BJT97] Hans-Joachim Baues, Mamuka Jibladze, and Andy Tonks. Cohomology of monoids in monoidal categories. In *Operads: Proceedings of Renaissance Conferences*, pages 137—-165. AMS, Providence, 1997. Available at http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.25.3891. (Referred to on page 73.)

[BM20] John C. Baez and Jade Master. Open Petri nets. *Mathematical Structures in Computer Science*, 30(3):314–341, 2020. Available at arXiv:1808.05415 . (Referred to on page v, 86, 94, 95.)

[BMM02] Roberto Bruni, José Meseguer, and Ugo Montanari. Symmetric monoidal and cartesian double categories as a semantic framework for tile logic. *Mathematical Structures in Computer Science*, 12(1):53–90, 2002. Available at http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.47.3417. (Referred to on page 9.)

[BMMS01] Roberto Bruni, José Meseguer, Ugo Montanari, and Vladimiro Sassone. Functorial models for Petri nets. *Information and Computa-*

tion, 170(2):207–236, 2001. Available at http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.96.6481. (Referred to on page 6, 44, 48, 49, 53.)

[Bor94a] Francis Borceux. *Handbook of Categorical Algebra: Volume 1, Basic Category Theory*. Cambridge University Press, Cambridge, 1994. (Referred to on page 104.)

[Bor94b] Francis Borceux. *Handbook of Categorical Algebra: Volume 2, Categories and Structures*. Cambridge University Press, Cambrdge, 1994. (Referred to on page 76, 77.)

[Buc08] Mitchell Buckley. Lawvere theories. 2008. Available at http://web.science.mq.edu.au/~street/MitchB.pdf. (Referred to on page 47, 139.)

[BW85] Michael Barr and Charles Wells. *Toposes, Triples and Theories*. Springer, 1985. Available at http://www.tac.mta.ca/tac/reprints/articles/12/tr12abs.html. (Referred to on page 47, 139.)

[CHP17] Florence Clerc, Harrison Humphrey, and Prakash Panangaden. Bicategories of Markov processes. In *Models, Algorithms, Logics and Tools*, pages 112–124. Springer, Berlin, 2017. Available at https://www.cs.mcgill.ca/~prakash/Pubs/bicats_final.pdf. (Referred to on page 9.)

[Cou17] Kenny Courser. A bicategory of decorated cospans. *Theory Appl. Categ.*, 32:995–1027, 2017. Available at arXiv:1605.08100. (Referred to on page 9.)

[Cou20] Kenny Courser. *Open Systems: a Double Categorical Perspective*. PhD thesis, University of California Riverside, 2020. Available at arXiv:2008.02394. (Referred to on page 22.)

[DMM89] Pierpaolo Degano, José Meseguer, and Ugo Montanari. Axiomatizing net computations and processes. In *Proceedings. Fourth Annual Symposium on Logic in Computer Science*, pages 175–185. IEEE, New York, 1989. (Referred to on page 44.)

[Dol13] Stephen Dolan. Fun with semirings: a functional pearl on the abuse of linear algebra. In *Proceedings of the 18th ACM SIGPLAN International Conference on Functional Programming*, pages 101–110. ACM, New York, 2013. (Referred to on page 7.)

[DS97] Brian Day and Ross Street. Monoidal bicategories and Hopf algebroids. *Advances in Mathematics*, 129:99–157, 1997. (Referred to on page 136.)

[Dub68] Eduardo Dubuc. Adjoint triangles. In *Reports of the Midwest Category Seminar II*, pages 69–91. Springer, Berlin, 1968. (Referred to on page 47.)

[Ehr63] Charles Ehresmann. Catégories structurées III: quintettes et applications covariantes. *Cahiers de Topologie et Géométrie Différentielle Catégoriques*, 5:1–22, 1963. (Referred to on page 9.)

[Ehr65] Charles Ehresmann. *Catégories et Structures*. Dunod, Paris, 1965. (Referred to on page 9.)

[EJL66] Fred E. J. Linton. Some aspects of equational categories. In *Proceedings of the Conference on Categorical Algebra*, pages 84–94. Springer, Berlin, 1966. (Referred to on page 48, 50, 52, 142.)

[Fin92] Eugene Fink. A survey of sequential and systolic algorithms for the algebraic path problem. Technical report, University of Waterloo, 1992. Available at https://www.cs.cmu.edu/ eugene/research/full/algebraic-path.pdf. (Referred to on page 100.)

[FK72] Peter J. Freyd and Gregory M. Kelly. Categories of continuous functors I. *Journal of Pure and Applied Algebra*, 2:169–191, 1972. (Referred to on page 89.)

[Fon16] Brendan Fong. *The Algebra of Open and Interconnected Systems*. PhD thesis, University of Oxford, 2016. Available at arXiv:1609.05382. (Referred to on page 22.)

[Foo15] Davis Foote. Kleene algebras and algebraic path problems. 2015. Available at https://edge.edx.org/asset-v1:BerkeleyX+CS188-SU16+SU16+type@asset+block/kleene_algebras_path_problems.pdf. (Referred to on page 6, 96, 98, 100.)

[FS19] Brendan Fong and David I. Spivak. *An Invitation to Applied Category Theory: Seven Sketches in Compositionality*. Cambridge University Press, Cambridge, 2019. (Referred to on page 98.)

[GH18] Fabrizio Genovese and Jelle Herold. Executions in (semi-)integer Petri nets are compact closed categories. In *Proceedings of the 15th International Conference on Quantum Physics and Logic*, pages 127–144. OASPA, The Hague, 2018. Available at arXiv:1805.05988. (Referred to on page 6, 49, 50.)

[Gir82] Michèle Giry. A categorical approach to probability theory. In *Categorical Aspects of Topology and Analysis*, pages 68–85. Springer, Berlin, 1982. (Referred to on page 131.)

[GP99] Marco Grandis and Robert Paré. Limits in double categories. *Cahiers de Topologie et Géométrie Différentielle Catégoriques*, 40:162–220, 1999. (Referred to on page 133.)

[GP04] Marco Grandis and Robert Paré. Adjoints for double categories. *Cahiers de Topologie et Géométrie Différentielle Catégoriques*, 45:193–240, 2004. (Referred to on page 133.)

[HM12] Peter Höfner and Bernhard Möller. Dijkstra, Floyd and Warshall meet Kleene. *Formal Aspects of Computing*, 24(4-6):459–476, 2012. (Referred to on page 7, 121.)

[JS91] André Joyal and Ross Street. The geometry of tensor calculus I. *Advances in Mathematics*, 88(1):55–112, 1991. (Referred to on page 48, 66.)

[Kel80] Gregory M. Kelly. A unified treatment of transfinite constructions for free algebras, free monoids, colimits, associated sheaves, and so on. *Bulletin of the Australian Mathematical Society*, 22(1):1–83, 1980. (Referred to on page 11, 116.)

[Lac10] Stephen Lack. Note on the construction of free monoids. *Applied Categorical Structures*, 18(1):17–29, 2010. (Referred to on page 73, 75.)

[Law63] William F. Lawvere. *Functorial semantics of algebraic theories*. PhD thesis, Columbia University, 1963. (Referred to on page 139.)

[Ler18] Eugene Lerman. Networks of open systems. *Journal of Geometry and Physics*, 130:81–112, 2018. Available at arXiv:1705.04814. (Referred to on page 9.)

[LS16] Eugene Lerman and David Spivak. An algebra of open continuous time dynamical systems and networks. 2016. Available at arXiv:1602.01017. (Referred to on page 9.)

[Mas19] Jade Master. Generalized Petri nets. Blog post on *The n-Category Café*, 2019. Available at https://golem.ph.utexas.edu/category/2019/04/generalized_petri_nets.html. (Referred to on page 66.)

[Mas20a] Jade Master. Compositional Markov. Code repository, 2020. Available at https://github.com/Jademaster/compositionalmarkov. (Referred to on page 13, 129.)

[Mas20b] Jade Master. Matrices are enriched profunctors. Available at jadeedenstarmaster.wordpress.com/2020/10/10/matrices-are-enriched-profunctors/, 2020. Blog post on *Cat Blog*. (Referred to on page 122.)

[Mas20c] Jade Master. The open algebraic path problem. 2020. Available at arXiv:2005.06682. (Referred to on page v.)

[Mas20d] Jade Master. Petri nets based on Lawvere theories. *Mathematical Structures in Computer Science*, 30(7):833–864, 2020. Available at arXiv1904.09091. (Referred to on page iv, 48, 50.)

[ML98] Saunders Mac Lane. *Categories for the Working Mathematician*. Springer, Berlin, 1998. (Referred to on page 19, 100, 134, 138.)

[MM90] José Meseguer and Ugo Montanari. Petri nets are monoids. *Information and omputation*, 88(2):105–155, 1990. (Referred to on page 6, 45.)

[Ngo] Timothy Ngotiaoco. Compositionality of the Runge–Kutta method. Available at arXiv:1707.02804. (Referred to on page 9.)

[PK12] Marc Pouly and Jürg Kohlas. *Generic Inference: a Unifying Theory for Automated Reasoning*. Wiley, Hoboken, 2012. (Referred to on page 120.)

[RB88] David E. Rydeheard and Rod M. Burstall. *Computational Category Theory*. Prentice Hall, Hoboken, 1988. (Referred to on page 55.)

[RSS14] Julian Rathke, Paweł Sobociński, and Owen Stephens. Compositional reachability in Petri nets. In *International Workshop on Reachability Problems*, pages 230–243. Springer, Berlin, 2014. (Referred to on page 129.)

[RT86] Grzegorz Rozenberg and P.S. Thiagarajan. Petri nets: basic notions, structure, behaviour. In *Current Trends in Concurrency*, pages 585–668. Springer, Berlin, 1986. (Referred to on page 50.)

[Sas95] Vladimiro Sassone. On the category of Petri net computations. In *Colloquium on Trees in Algebra and Programming*, pages 334–348. Springer, Berlin, 1995. (Referred to on page 44.)

[Sas96] Vladimiro Sassone. An axiomatization of the algebra of Petri net concatenable processes. *Theoretical Computer Science*, 170(1-2):277–296, 1996. (Referred to on page 44.)

[SH19] Mike Shulman and Linde Wester Hansen. Constructing symmetric monoidal bicategories functorially. 2019. Available at arXiv:1910.09240. (Referred to on page 133, 136.)

[Shu08] Michael Shulman. Framed bicategories and monoidal fibrations. *Theory and Applications of Categories*, 20(18):650–738, 2008. (Referred to on page 34.)

[STV95] Sairam Subramanian, Roberto Tamassia, and Jeffrey Scott Vitter. An efficient parallel algorithm for shortest paths in planar layered digraphs. *Algorithmica*, 14(4):322–339, 1995. (Referred to on page 128.)

[Tar81] Robert Endre Tarjan. A unified approach to path problems. *Journal of the Association for Computing Machinery*, 28(3):577–593, 1981. (Referred to on page 6, 96.)

[Tri14] Todd Trimble. multisorted Lawvere theories. Web resource on nLab, 2014. Available at https://ncatlab.org/toddtrimble/published/multisorted+Lawvere+theories. (Referred to on page 89.)

[Var15] Achille Varzi. Boundary. In *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford, 2015. Available at https://plato.stanford.edu/archives/win2015/entries/boundary/. (Referred to on page 1.)

[Wol74] Harvey Wolff. V-cat and V-graph. *Journal of Pure and Applied Algebra*, 4(2):123–135, 1974. (Referred to on page 113, 116.)

[Zai05] Dmitry Zaitsev. Functional Petri nets. Technical report, Université Paris–Dauphine, 2005. Available at https://www.researchgate.net/publication/265731959_Functional_Petri_Nets. (Referred to on page 13, 95.)

[ZS97] Dmitry Zaitsev and Anatoly Ilyich Sleptsov. State equations and equivalent transformations for timed Petri nets. *Cybernetics and Systems Analysis*, 33(5):659–672, 1997. (Referred to on page 13, 83, 93.)