Calculational Proofs in Relational Graphical Linear Algebra*

João Paixão[1] and Paweł Sobociński[2]

[1] Universidade Federal do Rio de Janeiro, Brazil
[2] Tallinn University of Technology, Estonia

Abstract. We showcase a modular, graphical language—graphical linear algebra—and use it as high-level language to reason calculationally about linear algebra. We propose a minimal framework of six axioms that highlight the dualities and symmetries of linear algebra, and use the resulting diagrammatic calculus as a convenient tool to prove a number of diverse theorems. Our work develops a relational approach to linear algebra, closely connected to classical relational algebra.

Keywords: Graphical Linear Algebra · Calculational Proofs · Diagrammatic Language · Galois Connections · Relational Mathematics

1 Introduction

This article is an introduction to Graphical Linear Algebra (GLA), a rigorous diagrammatic language for linear algebra. Its equational theory is known as the theory of Interacting Hopf Algebras [10, 25], and it has been applied (and adapted) in a series of papers [3, 8, 9, 4, 7, 5] to the modelling of signal flow graphs, Petri nets and even (non-passive) electrical circuits. In this paper, however, we focus on elucidating its status as an alternative language for linear algebra; that is, external to any specific applications. Some of this development was the focus of the second author's blog [24].

The diagrammatic syntax is extremely simple, yet powerful. Its simplicity is witnessed by the small number of concepts involved. Indeed, diagrams are built from just two structures, which we identify with black-white colouring. The black structure, roughly speaking, is *copying*; while the white structure is *adding*. Crucially, the interpretation of the diagrams is *relational*; instead of vector spaces and linear maps, the semantic universe is vector spaces and linear *relations* [1, 18, 12, 13]. It is somewhat astonishing that *all* other concepts are derived from only the interaction between these two simple structures.

One of our central contributions is the crystallisation of these interactions into six axioms. We show that they suffice to reconstruct the entire equational

* This research was supported by the ESF funded Estonian IT Academy research measure (project 2014-2020.4.05.19-0001). We thank Jason Erbele, Gabriel Ferreira, Lucas Rufino, and Iago Leal for fruitful conversations during various stages of this project.

theory of Interacting Hopf Algebras. A divergence from that work is our focus on *inequations*. Indeed, our axioms are inspired by the notion of *abelian bicategory* of Carboni and Walters [11]. Inequational reasoning leads to shorter and more concise proofs, as known in the relational community; for examples see [15] and [6], the latter with results closely related to ours. Importantly, in spite of having roots in deep mathematical frameworks (functorial semantics [17, 6], cartesian bicategories, abelian bicategories [11]) and its focus on universal properties and the use of high-level axioms, GLA is in fact a very simple system to work with: the rules of soundly manipulating diagrams can be easily taught to a student.

The semantics of diagrams is *compositional* – the meaning of a compound diagram is calculated from the meanings of its sub-diagrams. Here there is a connection with *relational algebra*, the two operations of diagram composition map to standard ways of composing relations: relational composition and cartesian product. In fact, the semantics of diagrams is captured as a monoidal *functor* from the category of diagrams GLA to the category of k-linear relations $\mathbf{LinRel}_{\mathsf{k}}$:

$$\mathsf{GLA} \to \mathbf{LinRel}_{\mathsf{k}}$$

One of our favourite features of GLA is how it makes the underlying symmetries of linear algebra apparent. Two symmetries are immediately built into the syntax of diagrams: *(i) mirror-image*: any diagram can be "flipped around the y-axis", and *(ii) colour-swap*: any diagram can be replaced with its "photographic negative", swapping the black and white colouring. Thus one proof can sometimes result in four theorems. We feel like these important symmetries are sometimes hidden in traditional approaches; moreover, some classical "theorems" become trivial consequences of the diagrammatic syntax.

After introducing the language and the principles of diagrammatic reasoning, our goal is to demonstrate that GLA is naturally suitable for modular constructions. Indeed, we argue that GLA can be considered as **high-level specification language for linear algebra**. While traditional calculational techniques are built around matrix algebra,the diagrammatic language is a higher-level way of expressing concepts and constructions. Of course, the equational theory means that we can "compile" down to low-level matrix representations, when necessary.

The compositional nature of the syntax means that we can express many classical concepts in a straightforward and intuitive fashion. Traditional linear algebra, in spite of its underlying simplicity and elegance, has a tendency for conceptual and notational proliferation: Kernel, $\mathsf{NullSpace}$, Image, sum of matrices, sum and intersection of vector spaces, etc. In GLA, all these concepts are expressible within the diagrammatic language. Moreover, the properties and relationships between the concepts can be derived via calculational proofs within the diagrammatic formalism. In many cases, these 2D proofs are shorter, more concise and more informative than their traditional 1D counterparts.

Finally, we compare the GLA approach with some recent work [20, 22] on blocked linear algebra, which also focussed on calculational proofs.

To quip, we show that "Calculationally Blocked Linear Algebra!" can be done graphically! By doing so, and moving from block matrix algebra to block relational algebra, we extend the expressivity of the approach to deal with subspaces.

Structure of the paper. In Section 2 we introduce the diagrammatic language, its semantics, and the six sound axioms for reasoning about linear relations. We then use those axioms in Section 3 to show that all of the equations of Interacting Hopf Algebras can be derived. In Section 4 we show how the diagrammatic language captures classical linear algebraic concepts, and relate it to recent work on block linear algebra. We conclude in Section 5 with some remarks about ongoing and future work.

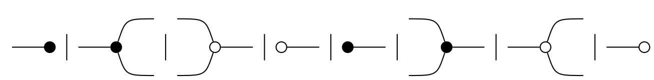
2 Syntax and Semantics of Graphical Linear Algebra

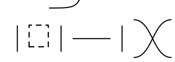
In this section we introduce the diagrammatic language used throughout the paper. We explain how diagrams are constructed and outline basic principles of how to manipulate, and reason with them. Our diagrams are an instance of a particular class of *string diagrams*, which are well-known [23] to characterise the arrows of free strict monoidal categories. They are, therefore, rigorous mathematical objects.

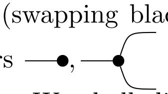
We emphasise a syntactic approach: we generate string diagrams as terms and then quotient them w.r.t. the laws of strict symmetric monoidal categories. This quotienting process is particularly pleasant and corresponds to a topological understanding of diagrams where only the connectivity matters: intuitive deformations of diagrams do not change their meaning. Thus our work does not require extensive familiarity of (monoidal) category theory: the rules of constructing and manipulating the diagrams can be easily explained.

2.1 Syntax

Our starting point is the simple grammar for a language of diagrams, generated from the BNF below.

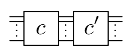
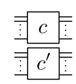
$c, d \quad ::=$ 

 | $c \,;\, d$ | $c \oplus d$ (1)

The first line of (1) consists of eight constants that we refer to as *generators*. Although they are given in diagrammatic form, for now we can consider them as mere symbols. Already here we can see two fundamental symmetries that are present throughout this paper: for every generator there is a "mirror image" generator (inverted in the "y-axis"), and for every generator there is a "colour inverse" generator (swapping black with white). Thus, it suffices to note that there are generators  and that the set of generators is closed w.r.t the two symmetries. We shall discuss the formal semantics of the syntax in

Section 2.2, but it is useful to already provide some intuition at this point. It is helpful to think of , as gates that, respectively, *discard* and *copy* the value on the left, and of , as *zero* and *add* gates. The mirror-image versions have the same intuitions, but from right-to-left.

The second line of (1) contains some structural terms and two binary operations for composing diagrams. The term is the empty diagram, is a wire and allows swapping the order of two wires. The two binary operations ; and $\oplus$ allow us to construct diagrams from smaller diagrams. Indeed, the diagrammatic convention is to draw $c\,;\,c'$ (or as $c' \cdot c$) as series composition , connecting the "dangling" wires on the right of c with those on the left of c', and to draw $c \oplus c'$ as c stacked on top of c', that is:

$c\,;\,c'$ is drawn  and $c \oplus c'$ is drawn  .

The simple sorting discipline of Fig. 1 counts the "dangling" wires and thus ensures that the diagrammatic convention for ; makes sense. A *sort* is a pair of natural numbers (m, n): m counts the dangling wires on the left and n those on the right. We consider only those terms of (1) that have a sort. It is not difficult to show that if a term has a sort, it is unique.

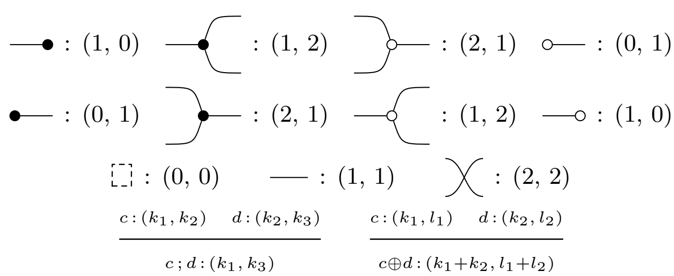
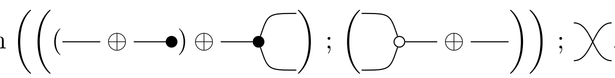
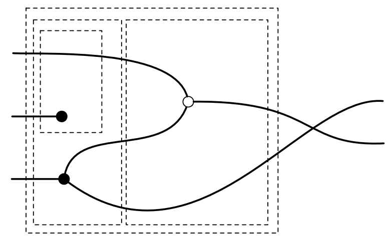


Fig. 1: The sorting discipline.

Example 1. Consider the term 
It has sort (3, 2) and the diagrammatic representation is



where the dashed-line boxes play the role of disambiguating associativity of operations.

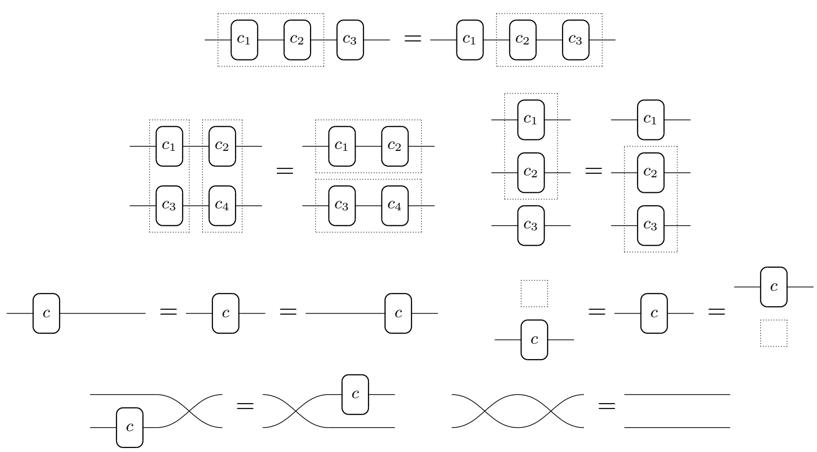


Fig. 2: Laws of Symmetric Monoidal Categories. Sort labels are omitted for readability.

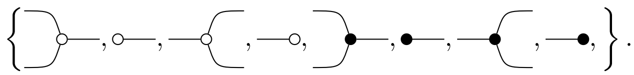
Raw terms are quotiented w.r.t. the laws of symmetric strict monoidal (SSM) categories, summarised in Fig. 2. We omit the (well-known) details [23] here and mention only that this amounts to eschewing the need for "dotted line boxes" and ensuring that diagrams which have the same topological connectivity are equated.

We refer to terms-modulo-SSM-equations as *string diagrams*. The resulting string diagrams are then the arrows of a SSM category known as a prop.

Definition 1. *A prop is a SSM category where the set of objects is the set of natural numbers, and, on objects $m \oplus n := m + n$. String diagrams generated from* (1) *are the arrows of a prop* GLA.

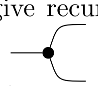
The idea is that the set of arrows from m to n is the set of string diagrams with m "dangling wires" on the left and n on the right.

In fact, GLA is the *free* prop on the set of generators

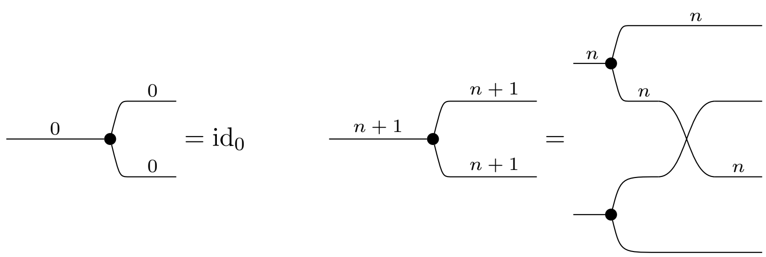


(2)

This perspective makes clear the status of GLA as a *syntax*: for example to define a morphism $\mathsf{GLA} \to \mathbb{X}$ to some prop $\mathbb{X}$, it suffices to define its action on the generators (2): the rest follows by induction.

We can also give recursive definitions, for example, the following generalisation of copying  will be useful for us in subsequent sections. Similar constructions can be given for the other generators in (2).

Definition 2.



2.2 Semantics

We use GLA as a diagrammatic language for linear algebra. Differently from traditional developments, the graphical syntax has a relational meaning. Indeed, the central mathematical concept is that of *linear relation.* Linear relations are also sometimes called *additive* relations [19] in the literature.

Definition 3. *Fix a field* k *and* k*-vector spaces* V*,* W*. A linear relation from* V *to* W *is a set* $R \subseteq V \times W$ *that is a subspace of* $V \times W$*, considered as a* k*-vector space. Explicitly this means that* $(0,0) \in R$*, given* $k \in \mathsf{k}$ *and* $(v,w) \in R$*,* $(kv, kw) \in R$*, and given* $(v,w), (v',w') \in R$*,* $(v+v', w+w') \in R$*.*

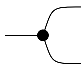
Given a field k, the prop $\mathbf{LinRel}_\mathsf{k}$ is of particular interest.

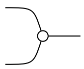
Definition 4. *The prop* $\mathbf{LinRel}_\mathsf{k}$ *has as arrows* $m \to n$ *linear relations* $R \subseteq \mathsf{k}^m \times \mathsf{k}^n$*. Composition is relational composition, and monoidal product is cartesian product of relations.*

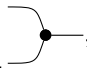
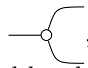
We now give the interpretation of the string diagrams of our language GLA as linear relations.

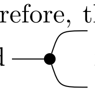
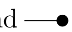
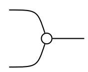
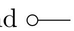
$$\mathsf{GLA} \to \mathbf{LinRel}_\mathsf{k} \tag{3}$$

Given that GLA is a free prop, it suffices to define the action on the generators.

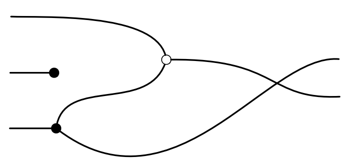
 $\longmapsto \left\{ \left(x, \begin{pmatrix} x \\ x \end{pmatrix} \right) \mid x \in \mathsf{k} \right\} \subseteq \mathsf{k} \times \mathsf{k}^2$ $\longmapsto \{(x, \star) \mid x \in \mathsf{k}\} \subseteq k \times k^0$

 $\longmapsto \left\{ \left(\begin{pmatrix} x \\ y \end{pmatrix}, x + y \right) \mid x, y \in \mathsf{k} \right\} \subseteq \mathsf{k}^2 \times \mathsf{k}$ $\longmapsto \{(\star, 0)\} \subseteq \mathsf{k}^0 \times \mathsf{k}$

The generators , , , map, respectively, to the opposite relations of the above, as hinted by the symmetric diagrammatic notation. As we shall see in the following, this semantic interpretation is canonical.

Intuitively, therefore, the *black structure* in diagrams refers to copying and discarding – indeed  is *copy*, and  is *discard*. Instead, the *white structure* refers to addition in k – indeed  is *add* and  is *zero*.

Example 2. Consider the term of Example 1, reproduced below without the unnecessary dashed-line box annotations.

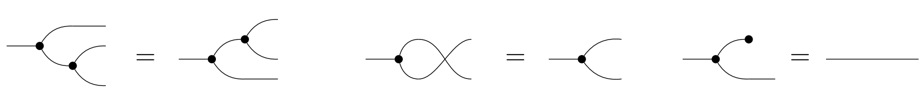


Its semantics is the (functional) relation $\left\{ \left(\begin{pmatrix} x \\ y \\ z \end{pmatrix}, \begin{pmatrix} z \\ x+z \end{pmatrix} \right) \mid x, y, z \in \mathsf{k} \right\}$.

2.3 Diagrammatic reasoning

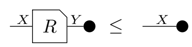
In this section we identify the (in)equational theory of GLA. It is common to define algebraic structure in a prop as a *monoidal theory* – roughly speaking, the prop version of an equational algebraic theory. Our goal, instead, is to arrive at a powerful calculus for linear algebra, with few high level axioms. These are inspired by the notion of *abelian bicategory* of Carboni and Walters [11].

Axiom 1 (Commutative comonoid) The copy structure satisfies the equations of commutative comonoids, that is, *associativity*, *commutativity* and *unitality*, as listed below:

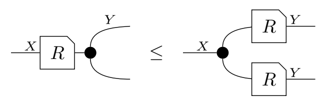


Inequalities. Instead of a purely equational axiomatisation, we find the use of inequalities very convenient in proofs. From a technical point of view this means the extension of the semantic mapping (3) to a (2-)functor between poset-enriched monoidal categories. Indeed, $\mathbf{LinRel}_{\mathsf{k}}$ has a natural poset-enrichment given by (set-theoretical) *inclusion* of relations.

We state three axioms below. Here R ranges over arbitrary string diagrams of GLA. Given that R can have arbitrary sort, the axioms make use of Definition 2.

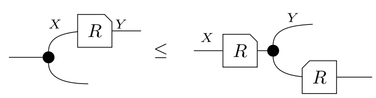
Axiom 2 (Discard) 

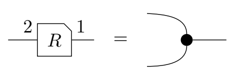
The axiom is sound; the left-hand-side denotes $LHS = \{(x, \star) \mid \exists y.\, xRy\}$, while right hand side denotes the relation $RHS = \{(x, \star) \mid \top\}$. Clearly $LHS \subseteq RHS$.

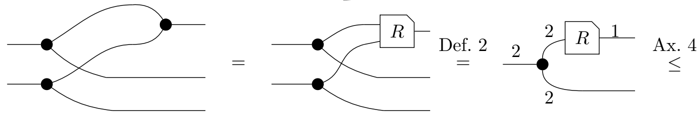
Axiom 3 (Copy) 

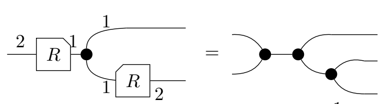
Soundness is again easy: the left-hand-side is the relation $LHS = \{(x, \binom{y}{y}) \mid xRy\}$, while the right-hand-side is $RHS = \{(x, \binom{y}{y'}) \mid xRy \wedge xRy'\}$. Again $LHS \subseteq RHS$.

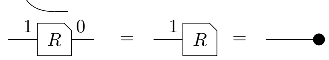
The diagrammatic notation for R allows us to represent the mirror image symmetry in an intuitive way. In the following diagram, the bottom right occurence of R is the reflected diagram—which we write R^o in linear syntax—and which we have seen denotes the opposite relation under the mapping (3).

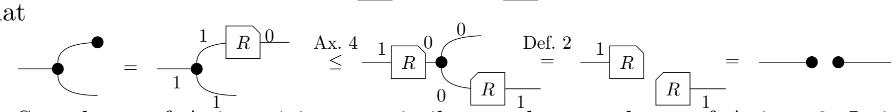
Axiom 4 (Wrong Way) 

For example, let  , then Axiom 4 says that





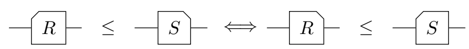
For another example, let  , then Axiom 4 says that



Soundness of Axiom 4 is very similar to the soundness of Axiom 3. It is worthwhile to explain the relationship between Axioms 3 and 4. In tandem, they capture a topological intuition: a relation that is adjacent to black node can "commute" with the node, resulting in a potentially larger relation. In Axiom 3 the relation approaches from the left. In Axiom 4 the relation comes from the "wrong side", it can still commute with the node to obtain a larger relation, but one must take care of the lower right wire that "curves around". It is this "curving around" that results in the R^o.

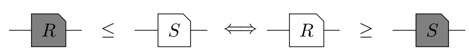
Symmetries In Section 2.1, we saw that the generators of GLA are closed under two symmetries: the "mirror-image" $(-)^o$ and the "colour-swap". Henceforward we denote the colour swap symmetry by $(-)^\dagger$. Clearly, for any R, we have $R^{oo} = R$ and $R^{\dagger\dagger} = R$. We are now ready to state our final two axioms.

Axiom 5 (Converse) *$R^o \leq S \iff R \leq S^o$, or in diagrams:*



In particular, the mirror-image symmetry is *covariant*: if $R \leq S$ then $R^o \leq S^o$. Soundness is immediate.

Axiom 6 (Colour inverse) *$R^\dagger \leq S \iff R \geq S^\dagger$, or in diagrams:*



Thus the colour-swap symmetry is *contravariant*: if $R \leq S$ then $S^\dagger \leq R^\dagger$. Here soundness is not as obvious: one way is to show that the inequalities of Axioms 2, 3 and 4 reverse when it is the white structure that is under consideration. First it is easy to see that the two operations commute.

Theorem 1. $(R^o)^\dagger = (R^\dagger)^o$

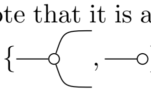
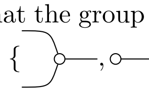
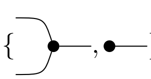
By the above Theorem we can define the transpose operation.

Definition 5. $R^t := (R^o)^\dagger = (R^\dagger)^o$.

Prove one, get three theorems for free principle. We will explore the symmetries of Axioms 5-6 throughout the paper. When we prove a result, we will usually assume we can use any of the other three theorems (its converse theorem, its colour inverse theorem, and its transpose theorem) afterwards just by referencing the original result. For example when we use, in a proof, the colour inverse of Axiom 3 we will simply denote Ax. 3.

3 Algebraic structure

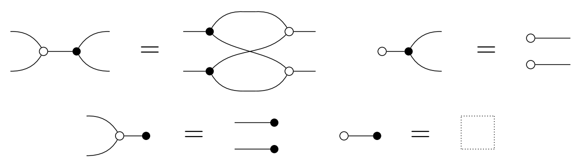
In this section we identify some of the algebraic structure that is a consequence of our six axioms. The two structures that play an important role are *bialgebras* and *special Frobenius algebras*. As Lack explains in [16], these are two canonical ways in which monoids and comonoids can interact. See *loc. cit.* for additional information on the provenance and importance of these two algebraic structures.

First note that it is an easy consequence of Axioms 5 and 6 that the group of generators {} is a commutative comonoid, and that {}, {} are both commutative monoids.

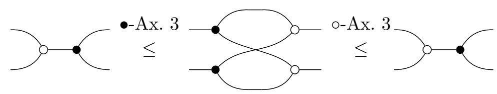
3.1 Bialgebra Structure

White and black structures act as bialgebras when they interact. This can be summarised by the following equations.

Theorem 2 (Bialgebra).

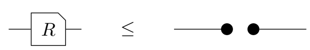


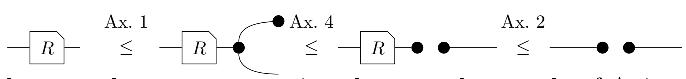
Proof.



We use Axiom 3 in the first inequality and its color inverse in the second inequality. Note that while using Axiom 3 in the proof, we must use Definition 2. The other derivations are similar.

Hom posets Every hom-set, i.e. the set of (m, n) string diagrams for $m, n \in \mathbb{N}$, has a top and bottom element. Indeed:

Theorem 3 (Top Element). 

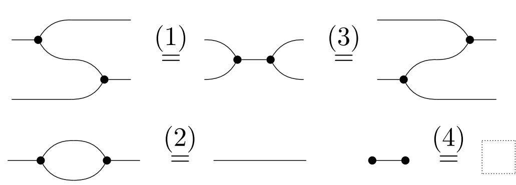
Proof. 

In the second step, we are using the second example of Axiom 4.

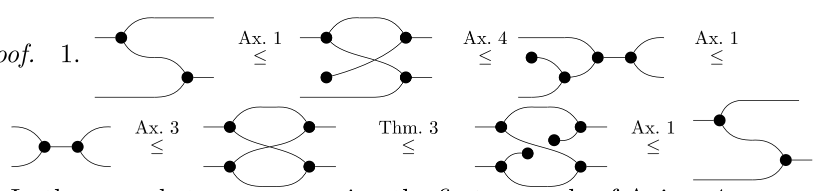
Using Axiom 6, we obtain, dually that is the bottom element.

Frobenius structure We have seen that the white and black structure interact according to the rules of bialgebras. On the other hand, individually the white and black structures interact as (extraspecial) Frobenius algebras.

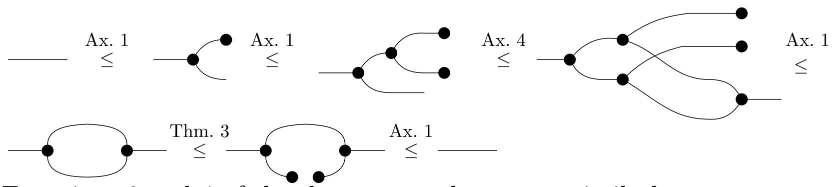
Theorem 4 (Frobenius). *The following equations hold.*



Proof. 1.



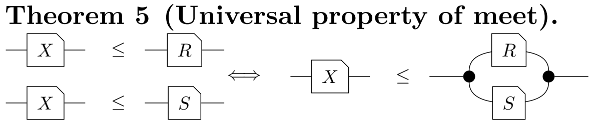
In the second step, we are using the first example of Axiom 4.

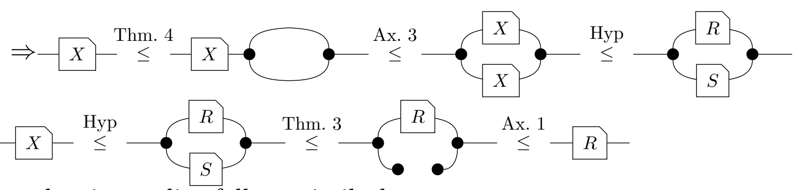
2. 

Equations 3 and 4 of the theorem can be proven similarly.

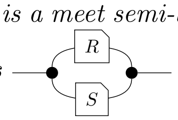
With the previous theorem we can show now that the set of (m, n) string diagrams for $m, n \in \mathbb{N}$, has also a meet and join operation, expressible within the GLA structure, as we show below.

Theorem 5 (Universal property of meet).



Proof. 

The other inequality follows similarly.

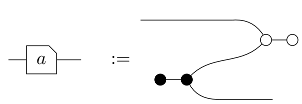
Corollary 1. *The partial order $\leq$ is a meet semi-lattice with top, where meet of R and S, written here as $R \cap S$, is*  *and the top element is given by Theorem 3.*

By duality of Axiom 6, we obtain the join and the bottom element just switching the colours. Therefore we have:

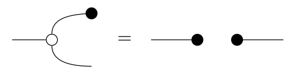
Corollary 2. *The partial order $\leq$ is a lattice with top, bottom, meet and join.*

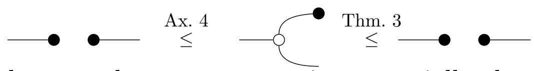
In the semantics, meet is intersection $R \cap S$, while join is the smallest subspace containing R and S: i.e. the subspace closure of the union $R \cup S$, usually denoted $R + S$.

Antipode One of the most surprising facts about this presentation of linear algebra is the plethora of concepts derivable from the basic components of copying and adding (black and white) structures. This includes the notion of *antipode* a, i.e. -1. The following proofs were inspired by [14].

Definition 6 (Antipode). 

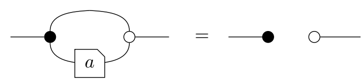
Before proving properties of the antipode, we prove an useful lemma.

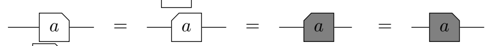
Lemma 1. 

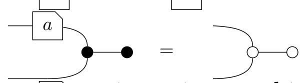
Proof. 

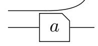
In the second step, we are using essentially the same idea from the second example of Axiom 4.

Theorem 6 (Antipode properties).

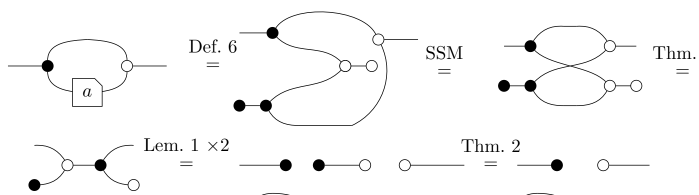
1. *(Hopf)* 

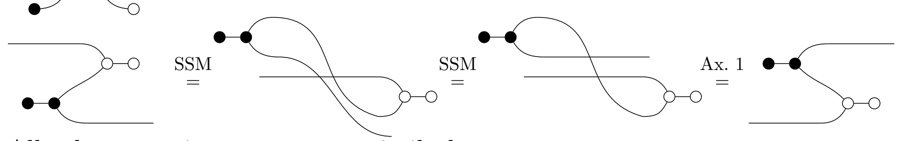
2.  $(a = a^o = a^\dagger = a^t)$

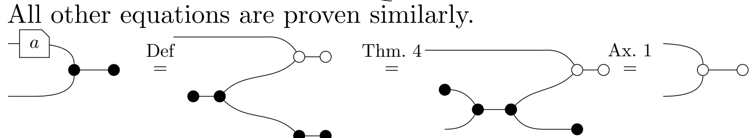
3. 

4.  *is an isomorphism.*

Proof.

1. 

2. 

All other equations are proven similarly.
3. 

4. We postpone this proof to Section 4.

The complete set of equations is known as the theory of Interacting Hopf (IH) Algebras [10]. In fact, given the results of this section (Theorems 2, 4, and 6) we have our main result.

Theorem 7. *Axioms 1-6 suffice to derive all of the algebraic structure of Interacting Hopf (IH) algebras. In fact, they are also sound in that theory. In particular, the prop obtained from* GLA*, quotiented by the axioms, is* isomorphic *to* $\mathbf{LinRel}_{\mathbb{Q}}$*, the prop of linear relations over the rationals.*

The above result justifies our claims about the canonicity of the semantics (3). While the base language (of adding and copying) is powerful enough to express any rational number, for other fields (e.g. the set of real numbers) we can add additional generators to our base language in a principled way [10]. We omit the details here.

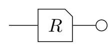
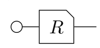
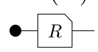
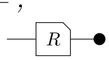
4 Applications

In this section, we will show how the graphical language, developed over the last few sections, can be used to reason about classical concepts and results in linear algebra.

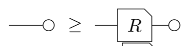
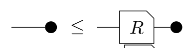
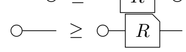
4.1 Dictionary

The goal of this subsection is to provide a dictionary, showing how familiar linear algebraic concepts manifest in the graphical language. Given a diagram R, we first define some derived diagrams from R directly in GLA but which are inspired by classical relations and subspaces in linear algebra and relational algebra.

Definition 7 (Derived diagrams).

1. *the nullspace of R: $N(R) :=$*  ,
2. *the multivalued part of R: $Mul(R) :=$*  ,
3. *the range of R: $Ran(R) :=$*  ,
4. *and the domain of R: $Dom(R) :=$*  .

Definition 8 (Dictionary). *Let R be a diagram. We call R*

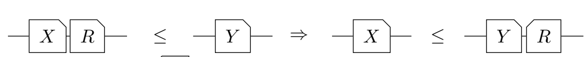
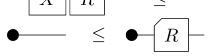
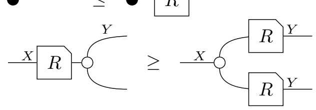
injective (INJ) if  ; *total (TOT) if*  ;
single-valued (SV) if  ; *surjective (SUR) if*  .

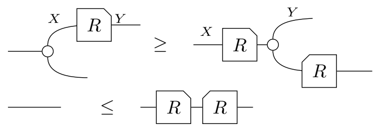
Moreover, since the converses of these inequalities hold in GLA, the four inequalities are actually equalities. We define a *map* to be a diagram that is both single-valued and total, $co - map$ is a diagram that is both surjective and injective, and finally an *Isomorphism* is a diagram that is map and a co-map. With the definition above we clearly obtain a nice symmetry.

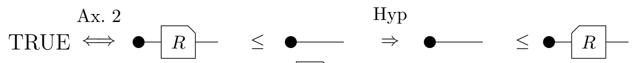
Theorem 8 (Dictionary symmetry). *R is total $\iff$ R^o is surjective $\iff$ $R^\dagger$ is injective $\iff$ $(R^\dagger)^o$ is single-valued.*

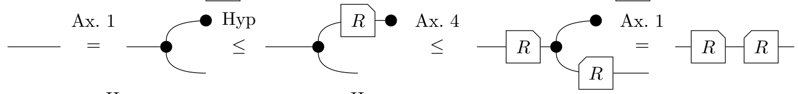
The next theorem shows that the notions in Definition 8 that were described through the converse inequalities of Axiom 2, can also be characterized by universal properties (item 1 from Theorem 9), the converse inequalities of Axioms 3 (item 3 from Theorem 9) and 4 (item 4 from Theorem 9), or by comparing their kernel and image relation with the identity (item 5 from Theorem 9).

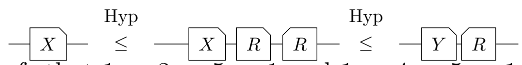
Theorem 9 (Total). *All the following statements are equivalent.*

1. 
2. 
3. 

4.–5. 

Proof. (1 ⇒ 2) 

(2 ⇒ 5) 

(5 ⇒ 1) 

The proofs that $1 \Rightarrow 3 \Rightarrow 5 \Rightarrow 1$ and $1 \Rightarrow 4 \Rightarrow 5 \Rightarrow 1$ are similar.

With the symmetries described in Theorem 8, we are able to obtain the same characterizations for injective, single-valued and surjective.

Observation When using Theorem 9 and its 3 other symmetrical variants in proofs, we refer to the initials TOT (Theorem 9), SV, INJ, and SUR.

Theorem 10. *1. R is Total* $\iff XR \subseteq Y \implies X \subseteq YR^o$
2. R is Single Valued $\iff XR \subseteq Y \impliedby X \subseteq YR^o$
3. R is Surjective $\iff XR \supseteq Y \implies X \supseteq YR^o$
4. R is Injective $\iff XR \supseteq Y \impliedby X \supseteq YR^o$
5. R is function (total and single valued) $\iff XR \subseteq Y \iff X \subseteq YR^o$
6. R is Co-function (injective and surjective) $\iff RX \subseteq Y \iff X \subseteq R^oY$
7. R is Injective and total $\iff XR = Y \implies X = YR^o$.
8. R is Isomorphism $\iff XR = Y \iff X = YR^o$.

These properties are essentially "shunting rules" inspired from relational algebra and its Galois connections [21, 2] which you can derive generic properties from them and allow you to reason effectively. We can use them in the proof of the following useful corollaries in GLA.

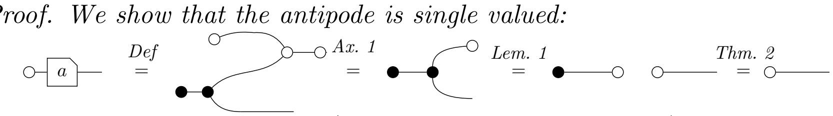
Corollary 3.

1. *A function f is an isomorphism iff its converse f^o is a function.*
2. *Smaller than injective (single valued) is injective (single valued).*
3. *Larger than total (surjective) is total (surjective).*

To end this section, we present an antipode property that we postponed in an earlier section.

Theorem 11. *The antipode is an isomorphism.*

Proof. We show that the antipode is single valued:



The other three properties (surjective, injective and total) have similar proofs.

4.2 Classical Existence Theorems

In this next subsection, we demonstrate how GLA can also prove classical *existence* theorems of linear algebra which connect subspaces, maps, and inverses. We begin with a common theorem in any undergraduate linear algebra class which gives sufficient conditions for the existence of a solution of a linear system $AX = B$. In linear algebra textbooks, this theorem is usually proven with Gaussian Elimination, LU decomposition, or RREF (Reduced Row Echelon Form). Here, we prove it completely in the graphical language.

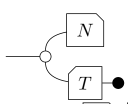
Theorem 12. *If B is a map, A is an injective map, and* $\bullet\!-\boxed{B}- \;\leq\; \bullet\!-\boxed{A}-$, *then there exists a map X such that* $-\boxed{X}\boxed{A}- \;=\; -\boxed{B}-$.

Proof. First let $-\boxed{X}- \;:=\; -\boxed{B}\boxed{A}-$, we will first show, in the first two items, that X is a map (single-valued and total) and then prove the equality in the last two items using Theorem 9 and its variants.

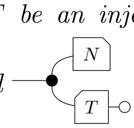
1. (X is SV) $\circ\!-\boxed{X}- \overset{\text{Def}}{=} \circ\!-\boxed{B}\boxed{A}- \overset{\text{B SV}}{\leq} \circ\!-\boxed{A}- \overset{\text{A INJ}}{\leq} \circ\!-\!-$
2. (X is TOT) $-\boxed{X}\!-\bullet \overset{\text{Def}}{=} -\boxed{B}\boxed{A}\!-\bullet \overset{\text{Hyp}}{\geq} -\boxed{B}\boxed{B}\!-\bullet \overset{\text{B TOT}}{\geq} -\!-\bullet$
3. $-\boxed{X}\boxed{A}- \overset{\text{Def}}{=} -\boxed{B}\boxed{A}\boxed{A}- \overset{\text{A SV}}{\leq} -\boxed{B}-$
4. $-\boxed{X}\boxed{A}- \overset{\text{B SV}}{\geq} -\boxed{X}\boxed{A}\boxed{B}\boxed{B}- \overset{\text{Item 3}}{\geq} -\boxed{X}\boxed{A}\boxed{A}\boxed{X}\boxed{B}- \overset{\text{A TOT}}{\geq}$
   $-\boxed{X}\boxed{X}\boxed{B}- \overset{\text{X TOT}}{\geq} -\boxed{B}-$

The last proof demonstrate the usefulness of Theorem 9 and its variants. Also we are able to define the solution X even though A might not be invertible, which shows the "high level expressivity" of GLA. The next theorem shows the connection between existence of complementary subspaces and existence of left inverses.

Theorem 13. *Let T be an injective map, there exists a* $\boxed{N}-$ *such that*

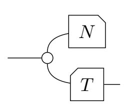


$\geq -\!-\bullet$ *and*

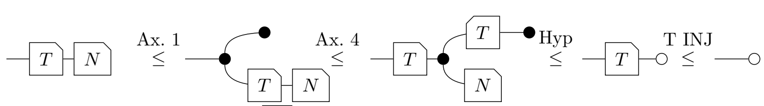


$\leq -\!-\circ \iff$ *there exists a map* $-\boxed{X}-$ *such that* $-\boxed{T}\boxed{X}- \;=\; -\!-$.

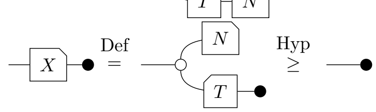
Proof. $\Rightarrow$ Let $-\boxed{X}- \;:=$



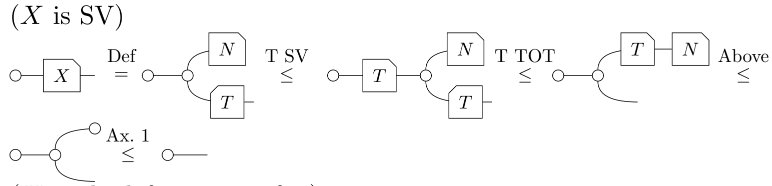
1. Claim:



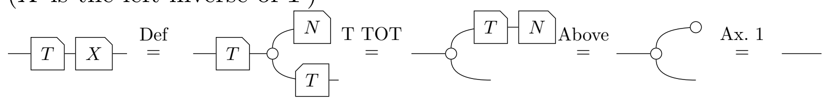
2. (X is TOT)



3. (X is SV)

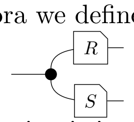
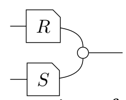


4. (X is the left inverse of T)



4.3 Dotted line associativity in Block Linear Algebra

In this short section we present the power of the dotted line associativity mentioned in the beginning of the paper. To make one final connection with classical linear algebra we define:

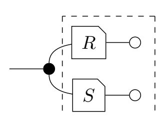
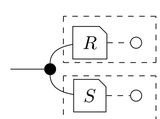
$\begin{bmatrix} R \\ \hline S \end{bmatrix} :=$  $\qquad [R|\, S] :=$ 

The notation is inspired by the semantics of the copy and add generators in the case that R and S are matrices (linear functions), which are horizontal or vertical block matrices R and S [24].

In GLA, we are able to immediately get the following properties by dotted line associativity, by looking and "grouping" the 2D syntax in two different ways such as, we did in example in Figure 2.

Theorem 14 (Dotted line associativity).

1. $N(\begin{bmatrix} R \\ \hline S \end{bmatrix}) = N(R) \cap N(S)$
2. $Ran(\begin{bmatrix} R | S \end{bmatrix}) = Ran(R) + Ran(S)$.
3. *(Absorption)* $(T \oplus U) \cdot \begin{bmatrix} R \\ \hline S \end{bmatrix} = \begin{bmatrix} T \cdot R \\ \hline U \cdot S \end{bmatrix}$
4. $\begin{bmatrix} R_1 \\ \hline R_2 \end{bmatrix}^o \cdot \begin{bmatrix} S_1 \\ \hline S_2 \end{bmatrix} = R_1^o \cdot S_1 \cap R_2^o \cdot S_2$
5. *(Exchange Law)* $\begin{bmatrix} \begin{bmatrix} R \\ \hline S \end{bmatrix} & \begin{bmatrix} T \\ \hline U \end{bmatrix} \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} R|T \\ \hline S|U \end{bmatrix} \end{bmatrix}$

Proof. 1)$N(R) \cap N(S) =$  $=$  $= N(\begin{bmatrix} R \\ \hline S \end{bmatrix})$

All the others items are also proven with dotted line associativity.

5 Conclusions and Future Work

We showcased *Graphical Linear Algebra*, a simple bichromatic diagrammatic language for linear algebra. We introduced a high level axiomatisation—which we argue is especially convenient for use in calculational proofs—and showed that it suffices to derive all of the algebraic structure of the theory of Interacting Hopf Algebras. We also focused on the modular nature of the language and how it captures many of the classical concepts of linear algebra.

A direction where future work will be especially fruitful is diagrammatic descriptions of various normal form theorems and matrix factorisations (Smith Normal Form, singular value decomposition, etc.); both in elucidating the classical theory, as well as obtaining useful relational generalisations.

References

1. Arens, R., et al.: Operational calculus of linear relations. Pacific Journal of Mathematics **11**(1), 9–23 (1961)
2. Backhouse, R.: Galois connections and fixed point calculus. In: Algebraic and coalgebraic methods in the mathematics of program construction, pp. 89–150. Springer (2002)
3. Baez, J., Erbele, J.: Categories in control. Theory and Application of Categories **30**, 836–881 (2015)
4. Bonchi, F., Holland, J., Pavlovic, D., Sobocinski, P.: Refinement for signal flow graphs. In: 28th International Conference on Concurrency Theory (CONCUR 2017). Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik (2017)
5. Bonchi, F., Holland, J., Piedeleu, R., Sobociński, P., Zanasi, F.: Diagrammatic algebra: from linear to concurrent systems. Proceedings of the ACM on Programming Languages **3**(POPL), 1–28 (2019)
6. Bonchi, F., Pavlovic, D., Sobocinski, P.: Functorial semantics for relational theories. arXiv preprint arXiv:1711.08699 (2017)
7. Bonchi, F., Piedeleu, R., annd Fabio Zanasi, P.S.: Graphical affine algebra. In: ACM/IEEE Symposium on Logic and Computer Science (LiCS '19) (2019)
8. Bonchi, F., Sobociński, P., Zanasi, F.: A categorical semantics of signal flow graphs. In: International Conference on Concurrency Theory. pp. 435–450. Springer (2014)
9. Bonchi, F., Sobocinski, P., Zanasi, F.: Full abstraction for signal flow graphs. In: POPL 2015. pp. 515–526. ACM (2015)
10. Bonchi, F., Sobociński, P., Zanasi, F.: Interacting Hopf algebras. Journal of Pure and Applied Algebra **221**(1), 144–184 (2017)
11. Carboni, A., Walters, R.F.: Cartesian bicategories I. Journal of pure and applied algebra **49**(1-2), 11–32 (1987)
12. Coddington, E.A.: Extension theory of formally normal and symmetric subspaces, vol. 134. American Mathematical Soc. (1973)
13. Cross, R.: Multivalued linear operators, vol. 213. CRC Press (1998)
14. Erbele, J.: Redundancy and zebra snakes, https://graphicallinearalgebra.net/2017/10/23/episode-r1-redundancy-and-zebra-snakes/
15. Freyd, P.J., Scedrov, A.: Categories, allegories, vol. 39. Elsevier (1990)
16. Lack, S.: Composing props. Theory and Applications of Categories **13**(9), 147–163 (2004)
17. Lawvere, F.W.: Functorial semantics of algebraic theories. Proceedings of the National Academy of Sciences **50**(5), 869–872 (1963)
18. Mac Lane, S.: An algebra of additive relations. Proceedings of the National Academy of Sciences of the United States of America **47**(7), 1043 (1961)

19. Mac Lane, S.: An algebra of additive relations. Proc Natl Acad Sci USA **47**(7), 1043–1051 (1961)
20. Macedo, H.D., Oliveira, J.N.: Typing linear algebra: A biproduct-oriented approach. Science of Computer Programming **78**(11), 2160–2191 (2013)
21. Mu, S.C., Oliveira, J.N.: Programming from galois connections. The Journal of Logic and Algebraic Programming **81**(6), 680–704 (2012)
22. Santos, A., Oliveira, J.N.: Type your matrices for great good: A haskell library of typed matrices and applications (functional pearl). In: Proceedings of the 13th ACM SIGPLAN International Symposium on Haskell. p. 54–66. Haskell 2020, Association for Computing Machinery, New York, NY, USA (2020). https://doi.org/10.1145/3406088.3409019, https://doi.org/10.1145/3406088.3409019
23. Selinger, P.: A survey of graphical languages for monoidal categories. In: New structures for physics, pp. 289–355. Springer (2010)
24. Sobociński, P.: Graphical linear algebra. mathematical blog.[Online]. Available: https://graphicallinearalgebra.net
25. Zanasi, F.: Interacting Hopf Algebras: the theory of linear systems. Ph.D. thesis, Ecole Normale Supérieure de Lyon (2015)