Contextual Semantics: From Quantum Mechanics to Logic, Databases, Constraints, and Complexity Lecture 2

Samson Abramsky

Department of Computer Science
The University of Oxford

Recap of Basic Mathematical Setting

Recap of Basic Mathematical Setting

An **observational frame** is a structure $(X, \mathcal{U}, O)$ where:

Recap of Basic Mathematical Setting

An **observational frame** is a structure $(X, \mathcal{U}, O)$ where:

- X is a set of “measurement labels” or “variables”

Recap of Basic Mathematical Setting

An **observational frame** is a structure $(X, \mathcal{U}, O)$ where:

- X is a set of "measurement labels" or "variables"
- $\mathcal{U}$ is a family of subsets of X with $\bigcup \mathcal{U} = X$; the "measurement contexts"

Recap of Basic Mathematical Setting

An **observational frame** is a structure $(X, \mathcal{U}, O)$ where:

- X is a set of "measurement labels" or "variables"
- $\mathcal{U}$ is a family of subsets of X with $\bigcup \mathcal{U} = X$; the "measurement contexts"
- O is a set of "outcomes" or "values"

Recap of Basic Mathematical Setting

An **observational frame** is a structure $(X, \mathcal{U}, O)$ where:

- X is a set of "measurement labels" or "variables"
- $\mathcal{U}$ is a family of subsets of X with $\bigcup \mathcal{U} = X$; the "measurement contexts"
- O is a set of "outcomes" or "values"

In addition, we have some commutative semiring R of "weights".

Recap of Basic Mathematical Setting

An **observational frame** is a structure $(X, \mathcal{U}, O)$ where:

- X is a set of "measurement labels" or "variables"
- $\mathcal{U}$ is a family of subsets of X with $\bigcup \mathcal{U} = X$; the "measurement contexts"
- O is a set of "outcomes" or "values"

In addition, we have some commutative semiring R of "weights".

Then we can define the presheaf

$$\mathcal{F} : \mathcal{P}(X)^{\mathsf{op}} \longrightarrow \mathbf{Set} \;::\; U \mapsto \mathcal{D}_R(O^U)$$

Recap of Basic Mathematical Setting

An **observational frame** is a structure $(X, \mathcal{U}, O)$ where:

- X is a set of "measurement labels" or "variables"
- $\mathcal{U}$ is a family of subsets of X with $\bigcup \mathcal{U} = X$; the "measurement contexts"
- O is a set of "outcomes" or "values"

In addition, we have some commutative semiring R of "weights".

Then we can define the presheaf

$$\mathcal{F} : \mathcal{P}(X)^{\mathsf{op}} \longrightarrow \mathbf{Set} \ :: \ U \mapsto \mathcal{D}_R(O^U)$$

A setting for **contextual probability**.

Empirical Models, Global Sections and Contextuality

Empirical Models, Global Sections and Contextuality

Given $(X, \mathcal{U}, O)$, an **empirical model** is a compatible family $\{d_U\}_{U \in \mathcal{U}}$, where $d_U \in \mathcal{F}(U)$, $U \in \mathcal{U}$.

Empirical Models, Global Sections and Contextuality

Given $(X, \mathcal{U}, O)$, an **empirical model** is a compatible family $\{d_U\}_{U \in \mathcal{U}}$, where $d_U \in \mathcal{F}(U)$, $U \in \mathcal{U}$.

Compatibility means that, for all $U.V \in \mathcal{U}$:

$$d_U|_{U \cap V} \ = \ d_V|_{U \cap V}.$$

Empirical Models, Global Sections and Contextuality

Given $(X, \mathcal{U}, O)$, an **empirical model** is a compatible family $\{d_U\}_{U \in \mathcal{U}}$, where $d_U \in \mathcal{F}(U)$, $U \in \mathcal{U}$.

Compatibility means that, for all $U.V \in \mathcal{U}$:

$$d_U|_{U \cap V} \;=\; d_V|_{U \cap V}.$$

A **global section** or **local hidden variable model** or **solution concept** for the model is $d \in \mathcal{F}(X)$ such that:

$$d|_U \;=\; d_U, \quad U \in \mathcal{U}.$$

Empirical Models, Global Sections and Contextuality

Given $(X, \mathcal{U}, O)$, an **empirical model** is a compatible family $\{d_U\}_{U \in \mathcal{U}}$, where $d_U \in \mathcal{F}(U)$, $U \in \mathcal{U}$.

Compatibility means that, for all $U.V \in \mathcal{U}$:

$$d_U|_{U \cap V} \;=\; d_V|_{U \cap V}.$$

A **global section** or **local hidden variable model** or **solution concept** for the model is $d \in \mathcal{F}(X)$ such that:

$$d|_U \;=\; d_U, \quad U \in \mathcal{U}.$$

The model is **contextual** if there is no such global section.

From Probability to Possibility

From Probability to Possibility

There is a semiring homomorphism $\mathbb{R}_{\geq 0} \longrightarrow \mathbb{B}$ which induces a natural transformation

$$\mathcal{D}_{\mathbb{R}_{\geq 0}}\mathcal{E} \stackrel{\cdot}{\longrightarrow} \mathcal{D}_{\mathbb{B}}\mathcal{E}$$

and hence acts on probabilistic models to produce relational or 'possibilistic' models.

From Probability to Possibility

There is a semiring homomorphism $\mathbb{R}_{\geq 0} \longrightarrow \mathbb{B}$ which induces a natural transformation

$$\mathcal{D}_{\mathbb{R}_{\geq 0}}\mathcal{E} \stackrel{\cdot}{\longrightarrow} \mathcal{D}_{\mathbb{B}}\mathcal{E}$$

and hence acts on probabilistic models to produce relational or 'possibilistic' models.

The effect of applying this to a probabilistic model is exactly to produce the boolean model corresponding to its support: for each context C, the probability distribution $e_C \in \mathcal{D}_{\mathbb{R}_{\geq 0}}(O^C)$ is mapped to the finite non-empty subset

$$\mathrm{supp}(e_C) \in \mathcal{D}_{\mathbb{B}}(O^C).$$

From Probability to Possibility

There is a semiring homomorphism $\mathbb{R}_{\geq 0} \longrightarrow \mathbb{B}$ which induces a natural transformation

$$\mathcal{D}_{\mathbb{R}_{\geq 0}}\mathcal{E} \stackrel{\cdot}{\longrightarrow} \mathcal{D}_{\mathbb{B}}\mathcal{E}$$

and hence acts on probabilistic models to produce relational or 'possibilistic' models.

The effect of applying this to a probabilistic model is exactly to produce the boolean model corresponding to its support: for each context C, the probability distribution $e_C \in \mathcal{D}_{\mathbb{R}_{\geq 0}}(O^C)$ is mapped to the finite non-empty subset

$$\mathsf{supp}(e_C) \in \mathcal{D}_{\mathbb{B}}(O^C).$$

Note that this collapse **preserves** global sections, hence **reflects** obstructions to global sections.

From Probability to Possibility

There is a semiring homomorphism $\mathbb{R}_{\geq 0} \longrightarrow \mathbb{B}$ which induces a natural transformation

$$\mathcal{D}_{\mathbb{R}_{\geq 0}}\mathcal{E} \stackrel{\cdot}{\longrightarrow} \mathcal{D}_{\mathbb{B}}\mathcal{E}$$

and hence acts on probabilistic models to produce relational or 'possibilistic' models.

The effect of applying this to a probabilistic model is exactly to produce the boolean model corresponding to its support: for each context C, the probability distribution $e_C \in \mathcal{D}_{\mathbb{R}_{\geq 0}}(O^C)$ is mapped to the finite non-empty subset

$$\mathsf{supp}(e_C) \in \mathcal{D}_{\mathbb{B}}(O^C).$$

Note that this collapse **preserves** global sections, hence **reflects** obstructions to global sections.

This means that no-go theorems proved at the possibilistic level are stronger (in fact, **strictly** stronger) than those proved at the probabilistic level.

Boolean global sections

Boolean global sections

It is then natural to ask what the existence of global sections on the Boolean level means for probabilistic models.

Boolean global sections

It is then natural to ask what the existence of global sections on the Boolean level means for probabilistic models.

In fact, we have already answered this question.

Proposition

The following are equivalent for a probabilistic empirical model:

1. *Its boolean image has a global section.*
2. *The model is* **logically contextual***.*

Boolean global sections

It is then natural to ask what the existence of global sections on the Boolean level means for probabilistic models.

In fact, we have already answered this question.

Proposition

The following are equivalent for a probabilistic empirical model:

1. *Its boolean image has a global section.*
2. *The model is* **logically contextual**.

What is the significance of Boolean global sections in their own right, independently of being derived from probabilistic models?

Boolean global sections

It is then natural to ask what the existence of global sections on the Boolean level means for probabilistic models.

In fact, we have already answered this question.

Proposition

The following are equivalent for a probabilistic empirical model:

1. *Its boolean image has a global section.*
2. *The model is* **logically contextual**.

What is the significance of Boolean global sections in their own right, independently of being derived from probabilistic models?

As we shall now see, they arise very directly in a number of familiar CS settings.

Relational databases

Relational databases

Samson Abramsky, 'Relational databases and Bell's theorem', **Festschrift for Peter Buneman**,Val Tannen (ed), 2013, to appear. Available as CoRR, abs/1208.6416.

Relational databases

Samson Abramsky, 'Relational databases and Bell's theorem', **Festschrift for Peter Buneman**,Val Tannen (ed), 2013, to appear. Available as CoRR, abs/1208.6416.

branch-name	**account-no**	**customer-name**	**balance**
Cambridge	10991-06284	Newton	£2,567.53
Hanover	10992-35671	Leibniz	€11,245.75
...	...	...	...

Anatomy of a table in a relational database

Anatomy of a table in a relational database

The columns are determined by a set A of **attributes**. Assume $A \subset \mathcal{A}$ for some global set $\mathcal{A}$ specified by the database schema.

Anatomy of a table in a relational database

The columns are determined by a set A of **attributes**. Assume $A \subset \mathcal{A}$ for some global set $\mathcal{A}$ specified by the database schema.

For each attribute a, there is a possible set of **data values** D_a. For simplicity, we collect these into a global set $D = \bigsqcup_{a \in \mathcal{A}} D_a$.

Anatomy of a table in a relational database

The columns are determined by a set A of **attributes**. Assume $A \subset \mathcal{A}$ for some global set $\mathcal{A}$ specified by the database schema.

For each attribute a, there is a possible set of **data values** D_a. For simplicity, we collect these into a global set $D = \bigsqcup_{a \in \mathcal{A}} D_a$.

An **A-tuple** is specified by a function $t : A \to D$.

Anatomy of a table in a relational database

The columns are determined by a set A of **attributes**. Assume $A \subset \mathcal{A}$ for some global set $\mathcal{A}$ specified by the database schema.

For each attribute a, there is a possible set of **data values** D_a. For simplicity, we collect these into a global set $D = \bigsqcup_{a \in \mathcal{A}} D_a$.

An **A-tuple** is specified by a function $t : A \to D$.

A **relation instance** or **table** of schema A is a set of A-tuples.

Anatomy of a table in a relational database

The columns are determined by a set A of **attributes**. Assume $A \subset \mathcal{A}$ for some global set $\mathcal{A}$ specified by the database schema.

For each attribute a, there is a possible set of **data values** D_a. For simplicity, we collect these into a global set $D = \bigsqcup_{a \in \mathcal{A}} D_a$.

An **A-tuple** is specified by a function $t : A \to D$.

A **relation instance** or **table** of schema A is a set of A-tuples.

A **database schema** is given by a family $\Sigma = \{A_1, \ldots, A_k\}$ of finite subsets of $\mathcal{A}$.

Anatomy of a table in a relational database

The columns are determined by a set A of **attributes**. Assume $A \subset \mathcal{A}$ for some global set $\mathcal{A}$ specified by the database schema.

For each attribute a, there is a possible set of **data values** D_a. For simplicity, we collect these into a global set $D = \bigsqcup_{a \in \mathcal{A}} D_a$.

An **A-tuple** is specified by a function $t : A \to D$.

A **relation instance** or **table** of schema A is a set of A-tuples.

A **database schema** is given by a family $\Sigma = \{A_1, \ldots, A_k\}$ of finite subsets of $\mathcal{A}$.

A database **instance** of schema Σ is given by a family of relation instances $\{R_i\}$ where R_i is of schema A_i.

Anatomy of a table in a relational database

The columns are determined by a set A of **attributes**. Assume $A \subset \mathcal{A}$ for some global set $\mathcal{A}$ specified by the database schema.

For each attribute a, there is a possible set of **data values** D_a. For simplicity, we collect these into a global set $D = \bigsqcup_{a \in \mathcal{A}} D_a$.

An **A-tuple** is specified by a function $t : A \to D$.

A **relation instance** or **table** of schema A is a set of A-tuples.

A **database schema** is given by a family $\Sigma = \{A_1, \ldots, A_k\}$ of finite subsets of $\mathcal{A}$.

A database **instance** of schema Σ is given by a family of relation instances $\{R_i\}$ where R_i is of schema A_i.

Does this look familiar?

Databases in the language of presheaves

Databases in the language of presheaves

An A-tuple t is just a local section over A: $t \in D^A = \mathcal{E}(A)$.

Databases in the language of presheaves

An A-tuple t is just a local section over A: $t \in D^A = \mathcal{E}(A)$.

A relation table R of schema A is a boolean distribution on A-tuples:

$$R \in \mathcal{D}_{\mathsf{B}}\mathcal{E}(A).$$

Databases in the language of presheaves

An A-tuple t is just a local section over A: $t \in D^A = \mathcal{E}(A)$.

A relation table R of schema A is a boolean distribution on A-tuples:

$$R \in \mathcal{D}_{\mathrm{B}}\mathcal{E}(A).$$

Note that if $A \subseteq B$, then restriction is just **projection**. For $R \in \mathcal{D}_{\mathrm{B}}\mathcal{E}(B)$

$$R|_A := \{t|_A \,:\, t \in R\}.$$

Databases in the language of presheaves

An A-tuple t is just a local section over A: $t \in D^A = \mathcal{E}(A)$.

A relation table R of schema A is a boolean distribution on A-tuples:

$$R \in \mathcal{D}_{\mathrm{B}}\mathcal{E}(A).$$

Note that if $A \subseteq B$, then restriction is just **projection**. For $R \in \mathcal{D}_{\mathrm{B}}\mathcal{E}(B)$

$$R|_A := \{t|_A \, : \, t \in R\}.$$

We can regard a schema Σ as a cover of $\mathcal{A}$.

Databases in the language of presheaves

An A-tuple t is just a local section over A: $t \in D^A = \mathcal{E}(A)$.

A relation table R of schema A is a boolean distribution on A-tuples:

$$R \in \mathcal{D}_{\mathrm{B}}\mathcal{E}(A).$$

Note that if $A \subseteq B$, then restriction is just **projection**. For $R \in \mathcal{D}_{\mathrm{B}}\mathcal{E}(B)$

$$R|_A := \{t|_A \,:\, t \in R\}.$$

We can regard a schema Σ as a cover of $\mathcal{A}$.

A database instance of schema Σ is a family of elements $\{R_A\}_{A \in \Sigma}$.

Databases in the language of presheaves

An A-tuple t is just a local section over A: $t \in D^A = \mathcal{E}(A)$.

A relation table R of schema A is a boolean distribution on A-tuples:

$$R \in \mathcal{D}_{\mathrm{B}}\mathcal{E}(A).$$

Note that if $A \subseteq B$, then restriction is just **projection**. For $R \in \mathcal{D}_{\mathrm{B}}\mathcal{E}(B)$

$$R|_A := \{t|_A \,:\, t \in R\}.$$

We can regard a schema Σ as a cover of $\mathcal{A}$.

A database instance of schema Σ is a family of elements $\{R_A\}_{A \in \Sigma}$.

The compatibility condition for an instance is **projection consistency**:

$$R_A|_{A \cap B} = R_B|_{A \cap B}$$

means that the two relations have the same projections onto their common set of attributes.

Universal Relations

Universal Relations

A **universal relation** for an instance $\{R_A : A \in \Sigma\}$ of a schema Σ is a relation $R \in \mathcal{D}_{\mathrm{B}}(D^{\mathcal{A}})$ such that, for all $A \in \Sigma$:

$$R|_A = R_A.$$

Universal Relations

A **universal relation** for an instance $\{R_A : A \in \Sigma\}$ of a schema Σ is a relation $R \in \mathcal{D}_{\mathrm{B}}(D^{\mathcal{A}})$ such that, for all $A \in \Sigma$:

$$R|_A = R_A.$$

Thus it is a relation defined on the whole set of attributes from which each of the relations in the instance can be recovered by projection.

Universal Relations

A **universal relation** for an instance $\{R_A : A \in \Sigma\}$ of a schema Σ is a relation $R \in \mathcal{D}_{\mathrm{B}}(D^{\mathcal{A}})$ such that, for all $A \in \Sigma$:

$$R|_A = R_A.$$

Thus it is a relation defined on the whole set of attributes from which each of the relations in the instance can be recovered by projection.

This notion, and related ideas, played an important rôle in early developments in relational database theory in the 1980's.

Universal Relations

A **universal relation** for an instance $\{R_A : A \in \Sigma\}$ of a schema Σ is a relation $R \in \mathcal{D}_{\mathrm{B}}(D^{\mathcal{A}})$ such that, for all $A \in \Sigma$:

$$R|_A = R_A.$$

Thus it is a relation defined on the whole set of attributes from which each of the relations in the instance can be recovered by projection.

This notion, and related ideas, played an important rôle in early developments in relational database theory in the 1980's.

Note that a universal relation instance corresponds exactly to the notion of **global section** for the DB instance viewed as a compatible family. (Compatibility is obviously a necessary condition for such an instance to exist).

Universal Relations

A **universal relation** for an instance $\{R_A : A \in \Sigma\}$ of a schema Σ is a relation $R \in \mathcal{D}_{\mathrm{B}}(D^{\mathcal{A}})$ such that, for all $A \in \Sigma$:

$$R|_A = R_A.$$

Thus it is a relation defined on the whole set of attributes from which each of the relations in the instance can be recovered by projection.

This notion, and related ideas, played an important rôle in early developments in relational database theory in the 1980's.

Note that a universal relation instance corresponds exactly to the notion of **global section** for the DB instance viewed as a compatible family. (Compatibility is obviously a necessary condition for such an instance to exist).

Proposition

Let $(R_1, \ldots, R_k)$ be an instance for the schema $\Sigma = \{A_1, \ldots, A_k\}$. Define $R := \bowtie_{i=1}^{k} R_i$. Then a universal relation for the instance exists if and only if $R|_{A_i} = R_i$, $i = 1, \ldots, k$, and in this case R is the largest relation in $\mathcal{R}(\bigcup_i A_i)$ satisfying the condition for a global section.

A dictionary

A dictionary

Relational databases	measurement scenarios
attribute	measurement
set of attributes defining a relation table	compatible set of measurements
database schema	measurement cover
tuple	local section (joint outcome)
relation/set of tuples	boolean distribution on joint outcomes
universal relation instance	global section/hidden variable model
acyclicity	Vorob'ev condition

A dictionary

Relational databases	measurement scenarios
attribute	measurement
set of attributes defining a relation table	compatible set of measurements
database schema	measurement cover
tuple	local section (joint outcome)
relation/set of tuples	boolean distribution on joint outcomes
universal relation instance	global section/hidden variable model
acyclicity	Vorob'ev condition

We can also consider probabilistic databases and other generalisations;
cf. provenance semirings.

Constraint satisfaction

Constraint satisfaction

Given $U \in \mathcal{U}$, an **assignment** on U is a function $s \in O^U$.

Constraint satisfaction

Given $U \in \mathcal{U}$, an **assignment** on U is a function $s \in O^U$.

Boolean assignments can be described by formulas ('state descriptions')

$$\phi_s := \bigwedge_{x \in U} l_x$$

where $l_x = x$ if $s(x) = \mathbf{tt}$, $l_x = \neg x$ if $s(x) = \mathbf{ff}$.

Constraint satisfaction

Given $U \in \mathcal{U}$, an **assignment** on U is a function $s \in O^U$.

Boolean assignments can be described by formulas ('state descriptions')

$$\phi_s := \bigwedge_{x \in U} l_x$$

where $l_x = x$ if $s(x) = \mathbf{tt}$, $l_x = \neg x$ if $s(x) = \mathbf{ff}$.
Note that s is the unique assignment which makes ϕ_s true.

Constraint satisfaction

Given $U \in \mathcal{U}$, an **assignment** on U is a function $s \in O^U$.

Boolean assignments can be described by formulas ('state descriptions')

$$\phi_s := \bigwedge_{x \in U} l_x$$

where $l_x = x$ if $s(x) = \mathbf{tt}$, $l_x = \neg x$ if $s(x) = \mathbf{ff}$.
Note that s is the unique assignment which makes ϕ_s true.

This extends to general assignments into a set O using suitable predicates P_o.

Constraint satisfaction

Given $U \in \mathcal{U}$, an **assignment** on U is a function $s \in O^U$.

Boolean assignments can be described by formulas ('state descriptions')

$$\phi_s := \bigwedge_{x \in U} l_x$$

where $l_x = x$ if $s(x) = \mathbf{tt}$, $l_x = \neg x$ if $s(x) = \mathbf{ff}$.
Note that s is the unique assignment which makes ϕ_s true.

This extends to general assignments into a set O using suitable predicates P_o.

Now suppose we are given a set of assignments on U, $\{s_i\}_{i \in I}$. This is described by a formula

$$\phi_U = \bigvee_{i \in I} \phi_{s_i}.$$

Constraint satisfaction

Given $U \in \mathcal{U}$, an **assignment** on U is a function $s \in O^U$.

Boolean assignments can be described by formulas ('state descriptions')

$$\phi_s := \bigwedge_{x \in U} l_x$$

where $l_x = x$ if $s(x) = \mathbf{tt}$, $l_x = \neg x$ if $s(x) = \mathbf{ff}$.
Note that s is the unique assignment which makes ϕ_s true.

This extends to general assignments into a set O using suitable predicates P_o.

Now suppose we are given a set of assignments on U, $\{s_i\}_{i \in I}$. This is described by a formula

$$\phi_U = \bigvee_{i \in I} \phi_{s_i}.$$

We shall consider formulas $\phi = \bigwedge_{U \in \mathcal{U}} \phi_U$, where ϕ_U specifies a team on U.

Constraint satisfaction

Given $U \in \mathcal{U}$, an **assignment** on U is a function $s \in O^U$.

Boolean assignments can be described by formulas ('state descriptions')

$$\phi_s := \bigwedge_{x \in U} l_x$$

where $l_x = x$ if $s(x) = \mathbf{tt}$, $l_x = \neg x$ if $s(x) = \mathbf{ff}$.
Note that s is the unique assignment which makes ϕ_s true.

This extends to general assignments into a set O using suitable predicates P_o.

Now suppose we are given a set of assignments on U, $\{s_i\}_{i \in I}$. This is described by a formula

$$\phi_U = \bigvee_{i \in I} \phi_{s_i}.$$

We shall consider formulas $\phi = \bigwedge_{U \in \mathcal{U}} \phi_U$, where ϕ_U specifies a team on U.

These formulas define possibilistic models; and their satisfying assignments

$$v : \mathcal{X} \longrightarrow O$$

correspond exactly to **global sections** of these models.

LHV is in NP

LHV is in NP

In order to simplify notation, we shall consider relational models of the form (U, e), where $e \subseteq U^n \times U^n$. Thus we use the same underlying set U for both measurements and outcomes at each site.

LHV is in NP

In order to simplify notation, we shall consider relational models of the form (U, e), where $e \subseteq U^n \times U^n$. Thus we use the same underlying set U for both measurements and outcomes at each site.

We shall write $\mathrm{HV}(n)$ for the class of models of this form which has a local hidden variable realisation (*i.e.* a boolean global section). We are interested in the algorithmic problem of determining if a structure (U, e) of arity n is in $\mathrm{HV}(n)$.

LHV is in NP

In order to simplify notation, we shall consider relational models of the form (U, e), where $e \subseteq U^n \times U^n$. Thus we use the same underlying set U for both measurements and outcomes at each site.

We shall write $\mathsf{HV}(n)$ for the class of models of this form which has a local hidden variable realisation (*i.e.* a boolean global section). We are interested in the algorithmic problem of determining if a structure (U, e) of arity n is in $\mathsf{HV}(n)$.

Proposition

For each n, $\mathsf{HV}(n)$ is in NP.

LHV is in NP

In order to simplify notation, we shall consider relational models of the form (U, e), where $e \subseteq U^n \times U^n$. Thus we use the same underlying set U for both measurements and outcomes at each site.

We shall write HV(n) for the class of models of this form which has a local hidden variable realisation (*i.e.* a boolean global section). We are interested in the algorithmic problem of determining if a structure (U, e) of arity n is in HV(n).

Proposition

For each n, HV(n) *is in* NP*.*

Proof

From the previous Proposition, it is clear that HV(n) is defined by the following second-order formula interpreted over finite structures (U, e):

$$\forall \vec{x}.\exists \vec{y}.R(\vec{x}, \vec{y}) \ \wedge \ \forall \vec{x}, \vec{y}.\, R(\vec{x}, \vec{y}) \ \rightarrow \ \exists f_1, \ldots, f_n.\, \textstyle\bigwedge_i f_i(x_i) = y_i \ \wedge \ \forall \vec{v}.R(\vec{v}, f(\vec{v})).$$

LHV is in NP

In order to simplify notation, we shall consider relational models of the form (U, e), where $e \subseteq U^n \times U^n$. Thus we use the same underlying set U for both measurements and outcomes at each site.

We shall write $\mathsf{HV}(n)$ for the class of models of this form which has a local hidden variable realisation (*i.e.* a boolean global section). We are interested in the algorithmic problem of determining if a structure (U, e) of arity n is in $\mathsf{HV}(n)$.

Proposition

For each n, $\mathsf{HV}(n)$ is in NP.

Proof

From the previous Proposition, it is clear that $\mathsf{HV}(n)$ is defined by the following second-order formula interpreted over finite structures (U, e):

$$\forall \vec{x}.\exists \vec{y}.R(\vec{x}, \vec{y}) \ \wedge \ \forall \vec{x}, \vec{y}.\, R(\vec{x}, \vec{y}) \ \rightarrow \ \exists f_1, \ldots, f_n.\, \bigwedge_i f_i(x_i) = y_i \ \wedge \ \forall \vec{v}.R(\vec{v}, f(\vec{v})).$$

By standard quantifier manipulations, this can be brought into an equivalent Σ^1_1 form, and hence $\mathsf{HV}(n)$ is in NP. □

Robust CSP

Robust CSP

Samson Abramsky, Georg Gottlob and Phokion Kolaitis, 'Robust Constraint Satisfaction and Local Hidden Variables in Quantum Mechanics', in Proceedings of IJCAI 2013.

Robust CSP

Samson Abramsky, Georg Gottlob and Phokion Kolaitis, 'Robust Constraint Satisfaction and Local Hidden Variables in Quantum Mechanics', in Proceedings of IJCAI 2013.

- Robust CSP: can every **consistent partial assignment** of a certain length be extended to a solution?

Robust CSP

Samson Abramsky, Georg Gottlob and Phokion Kolaitis, ‘Robust Constraint Satisfaction and Local Hidden Variables in Quantum Mechanics’, in Proceedings of IJCAI 2013.

- Robust CSP: can every **consistent partial assignment** of a certain length be extended to a solution?
- Special cases studied previously by Beacham and Gottlob.

Robust CSP

Samson Abramsky, Georg Gottlob and Phokion Kolaitis, ‘Robust Constraint Satisfaction and Local Hidden Variables in Quantum Mechanics’, in Proceedings of IJCAI 2013.

- Robust CSP: can every **consistent partial assignment** of a certain length be extended to a solution?
- Special cases studied previously by Beacham and Gottlob.
- Main results: Robust 3-colourability and Robust 2-sat are NP-complete.

Robust CSP

Samson Abramsky, Georg Gottlob and Phokion Kolaitis, 'Robust Constraint Satisfaction and Local Hidden Variables in Quantum Mechanics', in Proceedings of IJCAI 2013.

- Robust CSP: can every **consistent partial assignment** of a certain length be extended to a solution?
- Special cases studied previously by Beacham and Gottlob.
- Main results: ROBUST 3-COLOURABILITY and ROBUST 2-SAT are NP-complete.
- These are used to show that HV(n), $n > 2$, is NP-complete; smaller instances are in PTIME.

Robust CSP

Samson Abramsky, Georg Gottlob and Phokion Kolaitis, 'Robust Constraint Satisfaction and Local Hidden Variables in Quantum Mechanics', in Proceedings of IJCAI 2013.

- Robust CSP: can every **consistent partial assignment** of a certain length be extended to a solution?
- Special cases studied previously by Beacham and Gottlob.
- Main results: ROBUST 3-COLOURABILITY and ROBUST 2-SAT are NP-complete.
- These are used to show that HV(n), $n > 2$, is NP-complete; smaller instances are in PTIME.
- The robust paradigm is an interesting and non-trivial extension of current theory, and worthy of further study.

Contextual Semantics

Contextual Semantics

Why do such similar structures arise in such apparently different settings?

Contextual Semantics

Why do such similar structures arise in such apparently different settings?

The phenomenon of contextuality is pervasive. Once we start looking for it, we can find it everywhere!

Contextual Semantics

Why do such similar structures arise in such apparently different settings?

The phenomenon of contextuality is pervasive. Once we start looking for it, we can find it everywhere!

Physics, computation, logic, natural language, . . . biology, economics, . . .

Contextual Semantics

Why do such similar structures arise in such apparently different settings?

The phenomenon of contextuality is pervasive. Once we start looking for it, we can find it everywhere!

Physics, computation, logic, natural language, . . . biology, economics, . . .

The **Contextual semantics hypothesis**: we can find common mathematical structure in all these diverse manifestations, and develop a widely applicable theory.

Valuation Algebras

Valuation Algebras

Introduced by G. Shafer and P. Shenoy c. 1988 as an algebraic structure providing a common basis for **local computation** in Bayesian and belief-propagation networks.

Valuation Algebras

Introduced by G. Shafer and P. Shenoy c. 1988 as an algebraic structure providing a common basis for **local computation** in Bayesian and belief-propagation networks.

Subsequently understood to be applicable to a very wide range of inference problems. E.g. *Generic Inference* by Marc Pouly and Jürg Kohlas, Wiley 2011;

> *The inference task can then be described for general valuation algebras, which leads to a single computational problem that abstracts numerous important and seemingly different applications in computer science as for example query answering in databases, the evaluation of Bayesian and Gaussian networks, the solution of constraint, equation and inequality systems, satisfiability and theorem proving in logics, smoothing and filtering in linear dynamic systems and hidden Markov chains, the computation of discrete Fourier and Cosine transforms, various applications of path problems and coding schemes, sparse matrix techniques or numerical and symbolic partial differentiation. The properties of valuation algebras enable the efficient solution of all these problems with a single generic algorithm that exploits so-called tree-decomposition techniques.*

Valuation Algebras

Valuation Algebras

Carriers: Φ (valuations), D (domains, forming a lattice).

In practice, take $D \subseteq \mathcal{P}(\mathrm{Var})$, where Var is a set of variables.

Valuation Algebras

Carriers: Φ (valuations), D (domains, forming a lattice).
In practice, take $D \subseteq \mathcal{P}(\mathsf{Var})$, where Var is a set of variables.

Operations:

- **Domain** $d : \Phi \to D$; $\phi \mapsto d(\phi)$.
- **Projection** $\Phi \times D \to \Phi$; $(\phi, x) \mapsto \phi^{\downarrow x}$, for $x \subseteq d(\phi)$.
- **Combination** $\Phi \times \Phi \to \Phi$; $(\phi, \psi) \mapsto \phi \otimes \psi$.

Valuation Algebras

Carriers: Φ (valuations), D (domains, forming a lattice).
In practice, take $D \subseteq \mathcal{P}(\mathsf{Var})$, where Var is a set of variables.

Operations:

- Domain $d : \Phi \to D$; $\phi \mapsto d(\phi)$.
- Projection $\Phi \times D \to \Phi$; $(\phi, x) \mapsto \phi^{\downarrow x}$, for $x \subseteq d(\phi)$.
- Combination $\Phi \times \Phi \to \Phi$; $(\phi, \psi) \mapsto \phi \otimes \psi$.

Axioms:

- (A1) $(\Phi, \otimes)$ forms a commutative semigroup.
- (A2) $d(\phi \otimes \psi) = d(\phi) \cup d(\psi)$.
- (A3) $d(\phi^{\downarrow x}) = x$.
- (A4) $(\phi^{\downarrow y})^{\downarrow x} = \phi^{\downarrow x}, \quad (x \subseteq y \subseteq d(\phi))$.
- (A5) $(\phi \otimes \psi)^{\downarrow z} = \phi \otimes \psi^{\downarrow z \cap d(\psi)}, \quad (d(\phi) \subseteq z \subseteq d(\phi) \cup d(\psi))$.
- (A6) $\phi^{\downarrow d(\phi)} = \phi$.

Valuation Algebras

Carriers: Φ (valuations), D (domains, forming a lattice).
In practice, take $D \subseteq \mathcal{P}(\mathsf{Var})$, where Var is a set of variables.

Operations:

Domain $d : \Phi \to D$; $\phi \mapsto d(\phi)$.

Projection $\Phi \times D \to \Phi$; $(\phi, x) \mapsto \phi^{\downarrow x}$, for $x \subseteq d(\phi)$.

Combination $\Phi \times \Phi \to \Phi$; $(\phi, \psi) \mapsto \phi \otimes \psi$.

Axioms:

(A1) $(\Phi, \otimes)$ forms a commutative semigroup.

(A2) $d(\phi \otimes \psi) = d(\phi) \cup d(\psi)$.

(A3) $d(\phi^{\downarrow x}) = x$.

(A4) $(\phi^{\downarrow y})^{\downarrow x} = \phi^{\downarrow x}, \quad (x \subseteq y \subseteq d(\phi))$.

(A5) $(\phi \otimes \psi)^{\downarrow z} = \phi \otimes \psi^{\downarrow z \cap d(\psi)}, \quad (d(\phi) \subseteq z \subseteq d(\phi) \cup d(\psi))$.

(A6) $\phi^{\downarrow d(\phi)} = \phi$.

In most applications, valuations with domain x are distributions on assignments $\Pi_{X \in x} V_X$ valued in a semiring R. Projection is marginalisation.

Variable Elimination and Fusion

Variable Elimination and Fusion

Queries have the form $(\phi_1 \otimes \cdots \otimes \phi_n)^{\downarrow x}$.

Variable Elimination and Fusion

Queries have the form $(\phi_1 \otimes \cdots \otimes \phi_n)^{\downarrow x}$.

Marginalization over the full domain is prohibitively expensive. The key to tractable inference is to **localise the computation** using (A5).

Variable Elimination and Fusion

Queries have the form $(\phi_1 \otimes \cdots \otimes \phi_n)^{\downarrow x}$.

Marginalization over the full domain is prohibitively expensive. The key to tractable inference is to **localise the computation** using (A5).

A basic form of this localisation is **variable elimination**:

$$\phi^{-Y} \;=\; \psi^{-Y} \otimes \left(\bigotimes_{Y \notin d(\phi_i)} \phi_i \right),$$

where

$$\psi \;=\; \bigotimes_{Y \in d(\phi_i)} \phi_i.$$

Variable Elimination and Fusion

Queries have the form $(\phi_1 \otimes \cdots \otimes \phi_n)^{\downarrow x}$.

Marginalization over the full domain is prohibitively expensive. The key to tractable inference is to **localise the computation** using (A5).

A basic form of this localisation is **variable elimination**:

$$\phi^{-Y} \;=\; \psi^{-Y} \otimes \left(\bigotimes_{Y \notin d(\phi_i)} \phi_i \right),$$

where

$$\psi \;=\; \bigotimes_{Y \in d(\phi_i)} \phi_i .$$

This leads to the fusion rule: if $x = \{X_1, \ldots, X_n\}$, then

$$(\phi_1 \otimes \cdots \otimes \phi_n)^{\downarrow x} \;=\; \bigotimes \mathsf{Fus}_{X_n}(\cdots(\mathsf{Fus}_{X_1}(\{\phi_1, \ldots, \phi_n\}))\cdots)$$

where

$$\mathsf{Fus}_Y(\{\phi_1, \ldots, \phi_n\}) \;=\; \{\psi^{-Y}\} \cup \{\phi_i \, : \, Y \notin d(\phi_i)\}.$$

Local Computation

Local Computation

This idea is extended to performing multiple queries on **covering join trees** using **message passing architectures**. A number of these architectures have been widely used:

Local Computation

This idea is extended to performing multiple queries on **covering join trees** using **message passing architectures**. A number of these architectures have been widely used:

- The Shafer-Shenoy architecture

Local Computation

This idea is extended to performing multiple queries on **covering join trees** using **message passing architectures**. A number of these architectures have been widely used:

- The Shafer-Shenoy architecture
- The Lauritzen-Spiegelhalter architecture

Local Computation

This idea is extended to performing multiple queries on **covering join trees** using **message passing architectures**. A number of these architectures have been widely used:

- The Shafer-Shenoy architecture
- The Lauritzen-Spiegelhalter architecture
- The HUGIN architecture

Local Computation

This idea is extended to performing multiple queries on **covering join trees** using **message passing architectures**. A number of these architectures have been widely used:

- The Shafer-Shenoy architecture
- The Lauritzen-Spiegelhalter architecture
- The HUGIN architecture
- The idempotent architecture

Local Computation

This idea is extended to performing multiple queries on **covering join trees** using **message passing architectures**. A number of these architectures have been widely used:

- The Shafer-Shenoy architecture
- The Lauritzen-Spiegelhalter architecture
- The HUGIN architecture
- The idempotent architecture

The Shafer-Shenoy architecture is the most basic, and works for any valuation algebra. The L-S and HUGIN architectures achieve more efficiency, assuming additional properties of the valuation algebra (divisibility), while logical inference can be performed if we assume idempotence.

Valuation Algebras are Symmetric Monoidal Presheaves

Valuation Algebras are Symmetric Monoidal Presheaves

Valuation algebras fit perfectly in our contextual semantics framework:

Valuation Algebras are Symmetric Monoidal Presheaves

Valuation algebras fit perfectly in our contextual semantics framework:

- The map $d : \Phi \to D$ fibres the valuations over sets of variables, and the axioms (A3), (A4) and (A5) say this forms a presheaf

$$\mathcal{F} : D^{\mathrm{op}} \to \mathbf{Set} \;::\; x \mapsto \{\phi \in \Phi \,:\, d(\phi) = x\}.$$

Valuation Algebras are Symmetric Monoidal Presheaves

Valuation algebras fit perfectly in our contextual semantics framework:

- The map $d : \Phi \to D$ fibres the valuations over sets of variables, and the axioms (A3), (A4) and (A5) say this forms a presheaf

$$\mathcal{F} : D^{\mathsf{op}} \to \mathbf{Set} \; :: \; x \mapsto \{\phi \in \Phi \, : \, d(\phi) = x\}.$$

- For $x, y \in D$, combination gives a map (using (A2))

$$\otimes_{x,y} : \mathcal{F}(x) \times \mathcal{F}(y) \to \mathcal{F}(x \cup y).$$

  The properties of this map given by (A1) and (A5) say that it is a natural transformation making $\mathcal{F}$ into a **symmetric monoidal functor**, with respect to the monoidal structure on D as a join semilattice, and $(\mathbf{Set}, \times)$ as a monoidal category.

Valuation Algebras are Symmetric Monoidal Presheaves

Valuation algebras fit perfectly in our contextual semantics framework:

- The map $d : \Phi \rightarrow D$ fibres the valuations over sets of variables, and the axioms (A3), (A4) and (A5) say this forms a presheaf

$$\mathcal{F} : D^{\mathsf{op}} \rightarrow \mathbf{Set} \ :: \ x \mapsto \{\phi \in \Phi \ : \ d(\phi) = x\}.$$

- For $x, y \in D$, combination gives a map (using (A2))

$$\otimes_{x,y} : \mathcal{F}(x) \times \mathcal{F}(y) \rightarrow \mathcal{F}(x \cup y).$$

  The properties of this map given by (A1) and (A5) say that it is a natural transformation making $\mathcal{F}$ into a **symmetric monoidal functor**, with respect to the monoidal structure on D as a join semilattice, and $(\mathbf{Set}, \times)$ as a monoidal category.

- In particular, the crucial axiom (A5) is (essentially) **naturality**.

Valuation Algebras and Independence

Valuation Algebras and Independence

Conditional independence has been axiomatised by Philip Dawid in the notion of **separoid**. This is essentially the notion of independence studied in DL.

Valuation Algebras and Independence

Conditional independence has been axiomatised by Philip Dawid in the notion of **separoid**. This is essentially the notion of independence studied in DL.

Notation: $s \perp t | u \; [\phi]$.

Valuation Algebras and Independence

Conditional independence has been axiomatised by Philip Dawid in the notion of **separoid**. This is essentially the notion of independence studied in DL.

Notation: $s \perp t | u \; [\phi]$.

As pointed out by Shenoy and Dawid and Studeny, valuation algebras give rise to separoid structures.

Valuation Algebras and Independence

Conditional independence has been axiomatised by Philip Dawid in the notion of **separoid**. This is essentially the notion of independence studied in DL.

Notation: $s \perp t | u \; [\phi]$.

As pointed out by Shenoy and Dawid and Studeny, valuation algebras give rise to separoid structures.

We can define $s \perp t | u \; [\phi]$ if there exist $\psi_1, \psi_2 \in \Phi$ with $d(\psi_1) = s \cup u$, $d(\psi_2) = t \cup u$, and

$$\phi^{\downarrow s \cup t \cup u} \; = \; \psi_1 \otimes \psi_2 .$$

Valuation Algebras and Independence

Conditional independence has been axiomatised by Philip Dawid in the notion of **separoid**. This is essentially the notion of independence studied in DL.

Notation: $s \perp t | u \; [\phi]$.

As pointed out by Shenoy and Dawid and Studeny, valuation algebras give rise to separoid structures.

We can define $s \perp t | u \; [\phi]$ if there exist $\psi_1, \psi_2 \in \Phi$ with $d(\psi_1) = s \cup u$, $d(\psi_2) = t \cup u$, and

$$\phi^{\downarrow s \cup t \cup u} \; = \; \psi_1 \otimes \psi_2 .$$

This generalises the two equivalent ways of defining conditional independence in probability theory:

$$p(x, y | z) \; = \; p(x|z) p(y|z) \quad \text{vs.} \quad p(x|y, z) = p(x|z).$$

Valuation Algebras and Independence

Conditional independence has been axiomatised by Philip Dawid in the notion of **separoid**. This is essentially the notion of independence studied in DL.

Notation: $s \perp t | u \; [\phi]$.

As pointed out by Shenoy and Dawid and Studeny, valuation algebras give rise to separoid structures.

We can define $s \perp t | u \; [\phi]$ if there exist $\psi_1, \psi_2 \in \Phi$ with $d(\psi_1) = s \cup u$, $d(\psi_2) = t \cup u$, and

$$\phi^{\downarrow s \cup t \cup u} \; = \; \psi_1 \otimes \psi_2 .$$

This generalises the two equivalent ways of defining conditional independence in probability theory:

$$p(x, y | z) \; = \; p(x|z) p(y|z) \quad \text{vs.} \quad p(x|y, z) = p(x|z).$$

This may suggest a novel extension of DL with combination "built in", which may be well adapted to studying generic inference as captured by valuation algebras.

Applications to Natural Language Semantics

Applications to Natural Language Semantics

Preliminary work with Mehrnoosh Sadrzadeh.

Applications to Natural Language Semantics

Preliminary work with Mehrnoosh Sadrzadeh.

- A presheaf of 'Basic DRS'.

Applications to Natural Language Semantics

Preliminary work with Mehrnoosh Sadrzadeh.

- A presheaf of 'Basic DRS'.
- Need **Grothendieck topology**: not just inclusions of sets of variables, but maps allowing for relabelling.

Applications to Natural Language Semantics

Preliminary work with Mehrnoosh Sadrzadeh.

- A presheaf of 'Basic DRS'.
- Need **Grothendieck topology**: not just inclusions of sets of variables, but maps allowing for relabelling.
- Gluing local sections into global ones as **semantic unification**.

Applications to Natural Language Semantics

Preliminary work with Mehrnoosh Sadrzadeh.

- A presheaf of 'Basic DRS'.
- Need **Grothendieck topology**: not just inclusions of sets of variables, but maps allowing for relabelling.
- Gluing local sections into global ones as **semantic unification**.
- This is used to express resolution of anaphoric references.

Applications to Natural Language Semantics

Preliminary work with Mehrnoosh Sadrzadeh.

- A presheaf of 'Basic DRS'.
- Need **Grothendieck topology**: not just inclusions of sets of variables, but maps allowing for relabelling.
- Gluing local sections into global ones as **semantic unification**.
- This is used to express resolution of anaphoric references.

Example: 'John owns a donkey. It is grey.'

$$s_1 = \{John(x), Man(x)\}, \quad s_2 = \{donkey(y), \neg Man(y)\}, \quad s_3 = \{grey(z)\}\}.$$

Applications to Natural Language Semantics

Preliminary work with Mehrnoosh Sadrzadeh.

- A presheaf of 'Basic DRS'.
- Need **Grothendieck topology**: not just inclusions of sets of variables, but maps allowing for relabelling.
- Gluing local sections into global ones as **semantic unification**.
- This is used to express resolution of anaphoric references.

Example: 'John owns a donkey. It is grey.'

$$s_1 = \{John(x), Man(x)\}, \quad s_2 = \{donkey(y), \neg Man(y)\}, \quad s_3 = \{grey(z)\}\}.$$

Note that a cover which merged x and y would not have a gluing, since the consistency condition would be violated.

Applications to Natural Language Semantics

Preliminary work with Mehrnoosh Sadrzadeh.

- A presheaf of 'Basic DRS'.
- Need **Grothendieck topology**: not just inclusions of sets of variables, but maps allowing for relabelling.
- Gluing local sections into global ones as **semantic unification**.
- This is used to express resolution of anaphoric references.

Example: 'John owns a donkey. It is grey.'

$$s_1 = \{John(x), Man(x)\}, \quad s_2 = \{donkey(y), \neg Man(y)\}, \quad s_3 = \{grey(z)\}.$$

Note that a cover which merged x and y would not have a gluing, since the consistency condition would be violated.

However, using the cover

$$f_1 : x \mapsto a, \quad f_2 : y \mapsto b, \quad f_3 : z \mapsto b$$

we do have a gluing:

$$s = \{John(a), Man(a), donkey(b), \neg Man(b), grey(b)\}.$$

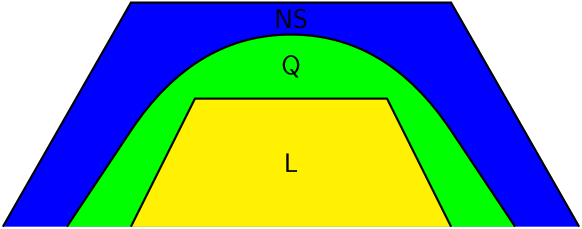
The Quantum Set

The Quantum Set

A subtle convex set sandwiched between two polytopes.

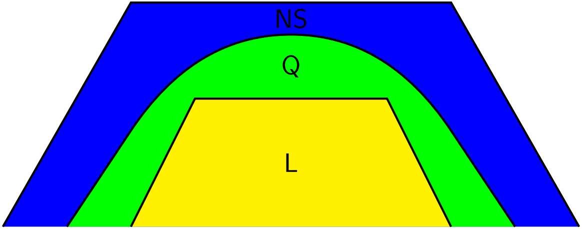
The Quantum Set

A subtle convex set sandwiched between two polytopes.



The Quantum Set

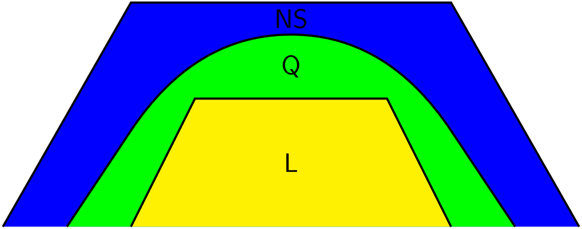
A subtle convex set sandwiched between two polytopes.



Key question: find compelling principles to explain why Nature picks out the quantum set.

The Quantum Set

A subtle convex set sandwiched between two polytopes.

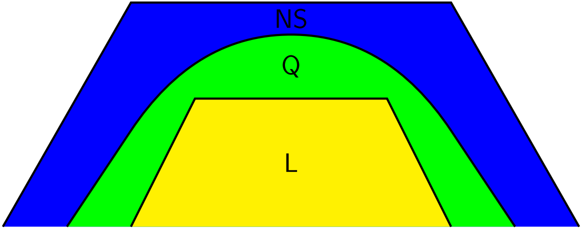


Key question: find compelling principles to explain why Nature picks out the quantum set.

- Probability distributions on global assignments give local models.

The Quantum Set

A subtle convex set sandwiched between two polytopes.

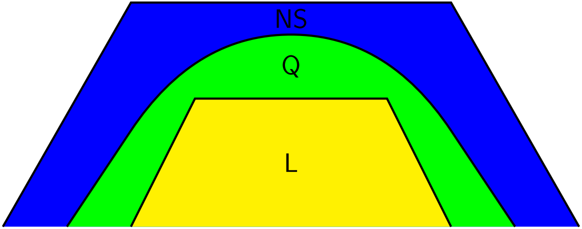


Key question: find compelling principles to explain why Nature picks out the quantum set.

- Probability distributions on global assignments give local models.
- Signed measures on global assignments give no-signalling models (SA + Adam Brandenburger).

The Quantum Set

A subtle convex set sandwiched between two polytopes.



Key question: find compelling principles to explain why Nature picks out the quantum set.

- Probability distributions on global assignments give local models.
- Signed measures on global assignments give no-signalling models (SA + Adam Brandenburger).
- Quantum measures on global assignments give quantum models?

Quantum Realizations of Relational Models

Quantum Realizations of Relational Models

A **quantum realization** of the system type (M, O) of arity n is given by:

Quantum Realizations of Relational Models

A **quantum realization** of the system type (M, O) of arity n is given by:

- Finite dimensional Hilbert spaces $\mathcal{H}_1, \ldots, \mathcal{H}_n$.

Quantum Realizations of Relational Models

A **quantum realization** of the system type (M, O) of arity n is given by:

- Finite dimensional Hilbert spaces $\mathcal{H}_1, \ldots, \mathcal{H}_n$.
- For each $i \in \mathbf{n}$, $m \in M_i$, and $o \in O_i$, a unit vector $\psi_{m,o}$ in $\mathcal{H}_i$, subject to the condition that the vectors $\{\psi_{m,o} : o \in O_i\}$ form an orthonormal basis of $\mathcal{H}_i$.

Quantum Realizations of Relational Models

A **quantum realization** of the system type (M, O) of arity n is given by:

- Finite dimensional Hilbert spaces $\mathcal{H}_1, \ldots, \mathcal{H}_n$.
- For each $i \in \mathbf{n}$, $m \in M_i$, and $o \in O_i$, a unit vector $\psi_{m,o}$ in $\mathcal{H}_i$, subject to the condition that the vectors $\{\psi_{m,o} \, : \, o \in O_i\}$ form an orthonormal basis of $\mathcal{H}_i$.
- A state ψ, *i.e.* a unit vector in $\mathcal{H}_1 \otimes \cdots \otimes H_n$.

Quantum Realizations of Relational Models

A **quantum realization** of the system type (M, O) of arity n is given by:

- Finite dimensional Hilbert spaces $\mathcal{H}_1, \ldots, \mathcal{H}_n$.
- For each $i \in \mathbf{n}$, $m \in M_i$, and $o \in O_i$, a unit vector $\psi_{m,o}$ in $\mathcal{H}_i$, subject to the condition that the vectors $\{\psi_{m,o} \, : \, o \in O_i\}$ form an orthonormal basis of $\mathcal{H}_i$.
- A state ψ, *i.e.* a unit vector in $\mathcal{H}_1 \otimes \cdots \otimes H_n$.

For each choice of measurement $\overline{m} \in M$, and outcome $\overline{o} \in O$, the usual 'statistical algorithm' of quantum mechanics defines a probability $p_{\overline{m}}(\overline{o})$ for obtaining outcome $\overline{o}$ from performing the measurement $\overline{m}$ on ρ:

$$p_{\overline{m}}(\overline{o}) = |\langle \psi \, : \, \psi_{\overline{m},\overline{o}} \rangle|^2,$$

where $\psi_{\overline{m},\overline{o}} = \psi_{\overline{m}_1,\overline{o}_1} \otimes \cdots \otimes \psi_{\overline{m}_n,\overline{o}_n}$.

Quantum Realizations of Relational Models

A **quantum realization** of the system type (M, O) of arity n is given by:

- Finite dimensional Hilbert spaces $\mathcal{H}_1, \ldots, \mathcal{H}_n$.
- For each $i \in \mathbf{n}$, $m \in M_i$, and $o \in O_i$, a unit vector $\psi_{m,o}$ in $\mathcal{H}_i$, subject to the condition that the vectors $\{\psi_{m,o} \, : \, o \in O_i\}$ form an orthonormal basis of $\mathcal{H}_i$.
- A state ψ, *i.e.* a unit vector in $\mathcal{H}_1 \otimes \cdots \otimes H_n$.

For each choice of measurement $\overline{m} \in M$, and outcome $\overline{o} \in O$, the usual 'statistical algorithm' of quantum mechanics defines a probability $p_{\overline{m}}(\overline{o})$ for obtaining outcome $\overline{o}$ from performing the measurement $\overline{m}$ on ρ:

$$p_{\overline{m}}(\overline{o}) = |\langle \psi \, : \, \psi_{\overline{m},\overline{o}} \rangle|^2,$$

where $\psi_{\overline{m},\overline{o}} = \psi_{\overline{m}_1,\overline{o}_1} \otimes \cdots \otimes \psi_{\overline{m}_n,\overline{o}_n}$.

To compute the tensor product of vectors:

$$\sum_i a_i |i\rangle \otimes \sum_j b_j |j\rangle = \sum_{i,j} a_i b_j |ij\rangle.$$

Quantum Realizations of Relational Models

A **quantum realization** of the system type (M, O) of arity n is given by:

- Finite dimensional Hilbert spaces $\mathcal{H}_1, \ldots, \mathcal{H}_n$.
- For each $i \in \mathbf{n}$, $m \in M_i$, and $o \in O_i$, a unit vector $\psi_{m,o}$ in $\mathcal{H}_i$, subject to the condition that the vectors $\{\psi_{m,o} : o \in O_i\}$ form an orthonormal basis of $\mathcal{H}_i$.
- A state ψ, *i.e.* a unit vector in $\mathcal{H}_1 \otimes \cdots \otimes H_n$.

For each choice of measurement $\overline{m} \in M$, and outcome $\overline{o} \in O$, the usual 'statistical algorithm' of quantum mechanics defines a probability $p_{\overline{m}}(\overline{o})$ for obtaining outcome $\overline{o}$ from performing the measurement $\overline{m}$ on ρ:

$$p_{\overline{m}}(\overline{o}) = |\langle \psi : \psi_{\overline{m},\overline{o}} \rangle|^2,$$

where $\psi_{\overline{m},\overline{o}} = \psi_{\overline{m}_1,\overline{o}_1} \otimes \cdots \otimes \psi_{\overline{m}_n,\overline{o}_n}$.

To compute the tensor product of vectors:

$$\sum_i a_i |i\rangle \otimes \sum_j b_j |j\rangle = \sum_{i,j} a_i b_j |ij\rangle.$$

We define a relational empirical model $e \subseteq M \times O$ by

$$e(\overline{m}, \overline{o}) \;\equiv\; p_{\overline{m}}(\overline{o}) > 0.$$

Thus e arises as the 'possibilistic collapse' of the usual quantum formalism.

Quantum Realization of the Hardy Model

Quantum Realization of the Hardy Model

We consider the two-qubit system, with X_2 and Y_2 measurement in the computational basis. We take $R = 0$, $G = 1$. The eigenvectors for X_1 are taken to be

$$\sqrt{\frac{3}{5}}|0\rangle + \sqrt{\frac{2}{5}}|1\rangle, \qquad -\sqrt{\frac{2}{5}}|0\rangle + \sqrt{\frac{3}{5}}|1\rangle$$

and similarly for Y_1. The state is taken to be

$$\sqrt{\frac{3}{8}}|10\rangle \; + \; \sqrt{\frac{3}{8}}|01\rangle \; - \; \frac{1}{2}|00\rangle.$$

Quantum Realization of the Hardy Model

We consider the two-qubit system, with X_2 and Y_2 measurement in the computational basis. We take $R = 0$, $G = 1$. The eigenvectors for X_1 are taken to be

$$\sqrt{\frac{3}{5}}|0\rangle + \sqrt{\frac{2}{5}}|1\rangle, \qquad -\sqrt{\frac{2}{5}}|0\rangle + \sqrt{\frac{3}{5}}|1\rangle$$

and similarly for Y_1. The state is taken to be

$$\sqrt{\frac{3}{8}}|10\rangle \; + \; \sqrt{\frac{3}{8}}|01\rangle \; - \; \frac{1}{2}|00\rangle.$$

One can then calculate the probabilities to be

$$p_{X_1 Y_2}(RR) \; = \; p_{X_2 Y_1}(RR) \; = \; p_{X_2 Y_2}(GG) = 0,$$

and $p_{X_1 Y_1}(RR) = 0.09$, which is very near the maximum attainable value.

Quantum Realization of the Hardy Model

We consider the two-qubit system, with X_2 and Y_2 measurement in the computational basis. We take $R = 0$, $G = 1$. The eigenvectors for X_1 are taken to be

$$\sqrt{\frac{3}{5}}|0\rangle + \sqrt{\frac{2}{5}}|1\rangle, \qquad -\sqrt{\frac{2}{5}}|0\rangle + \sqrt{\frac{3}{5}}|1\rangle$$

and similarly for Y_1. The state is taken to be

$$\sqrt{\frac{3}{8}}|10\rangle \; + \; \sqrt{\frac{3}{8}}|01\rangle \; - \; \frac{1}{2}|00\rangle.$$

One can then calculate the probabilities to be

$$p_{X_1 Y_2}(RR) \; = \; p_{X_2 Y_1}(RR) \; = \; p_{X_2 Y_2}(GG) = 0,$$

and $p_{X_1 Y_1}(RR) = 0.09$, which is very near the maximum attainable value.

The possibilistic collapse of this model is thus a Hardy model.

QM

QM

Proposition

The class QM(d) *is in* PSPACE*. That is, there is a* PSPACE *algorithm to decide, given an empirical model, if it arises from a quantum system of dimension* d*.*

QM

Proposition

The class QM(d) *is in* PSPACE*. That is, there is a* PSPACE *algorithm to decide, given an empirical model, if it arises from a quantum system of dimension* d*.*

Proof Outline

The condition for quantum realization of a relational model can be written as the existence of a list of complex matrices satisfying some algebraic conditions. These can be written in terms of the entries of the matrices, and we can use the standard representation of complex numbers as pairs of reals.

QM

Proposition

The class QM(d) *is in* PSPACE*. That is, there is a* PSPACE *algorithm to decide, given an empirical model, if it arises from a quantum system of dimension* d*.*

Proof Outline

The condition for quantum realization of a relational model can be written as the existence of a list of complex matrices satisfying some algebraic conditions. These can be written in terms of the entries of the matrices, and we can use the standard representation of complex numbers as pairs of reals.

The parameter d allows us to bound the dimensions of the matrices which need to be considered.

QM

Proposition

The class QM(d) *is in* PSPACE*. That is, there is a* PSPACE *algorithm to decide, given an empirical model, if it arises from a quantum system of dimension d.*

Proof Outline

The condition for quantum realization of a relational model can be written as the existence of a list of complex matrices satisfying some algebraic conditions. These can be written in terms of the entries of the matrices, and we can use the standard representation of complex numbers as pairs of reals.

The parameter d allows us to bound the dimensions of the matrices which need to be considered.

The whole condition can be written as an existential sentence $\exists v_1 \ldots \exists v_k . \psi$, where ψ is a conjunction of atomic formulas in the signature $(+, 0, \times, 1, <)$, interpreted over the reals.

QM

Proposition

The class QM(d) *is in* PSPACE*. That is, there is a* PSPACE *algorithm to decide, given an empirical model, if it arises from a quantum system of dimension d.*

Proof Outline

The condition for quantum realization of a relational model can be written as the existence of a list of complex matrices satisfying some algebraic conditions. These can be written in terms of the entries of the matrices, and we can use the standard representation of complex numbers as pairs of reals.

The parameter d allows us to bound the dimensions of the matrices which need to be considered.

The whole condition can be written as an existential sentence $\exists v_1 \ldots \exists v_k . \psi$, where ψ is a conjunction of atomic formulas in the signature $(+, 0, \times, 1, <)$, interpreted over the reals.

This fragment has PSPACE complexity (Canny). Moreover, the sentence can be constructed in polynomial time from the given relational empirical model. Hence membership of QM is in PSPACE.

A Decision Problem

A Decision Problem

Can we bound the dimension d effectively, so that QM itself is decidable?

A Decision Problem

Can we bound the dimension d effectively, so that QM itself is decidable?

Seems hard . . .

References

Papers (available on arXiv):

- S. Abramsky and A. Brandenburger. The sheaf-theoretic structure of non-locality and contextuality. *New Journal of Physics*, 13(2011):113036, 2011.
- S. Abramsky, S. Mansfield and R. Soares Barbosa, The Cohomology of Non-Locality and Contextuality, in *Proceedings of QPL 2011*, EPTCS 2011.
- S. Abramsky and L. Hardy. Logical Bell Inequalities. *Phys. Rev. A* 85, 062114 (2012).
- S. Abramsky, Relational Hidden Variables and Non-Locality. *Studia Logica* 101(2), 411–452, 2013.
- S. Abramsky, G. Gottlob and P. Kolaitis, Robust Constraint Satisfaction and Local Hidden Variables in Quantum Mechanics, Proceedings IJCAI 2013.
- S. Abramsky, Relational Databases and Bell's Theorem, In *In Search of Elegance in the Theory and Practice of Computation: Essays Dedicated to Peter Buneman*, Springer 2013.
- S. Abramsky and M. Sadrzadeh, Semantic Unification: a sheaf theoretic approach to natural language, in *Categories and Types in Logic, Language and Physics: Essays Dedicated to Jim Lambek*, Eds. Claudia Casadio, Bob Coecke, Michael Moortgat, Philip Scott, Springer 2014.