Graphical Reasoning
in Symmetric Monoidal Categories
for Quantum Information

Lucas Dixon, University of Edinburgh
(Joint work Ross Duncan and Aleks Kissinger)

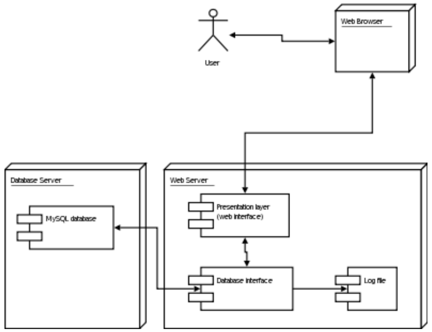
25 Jan 2010

School of informatics

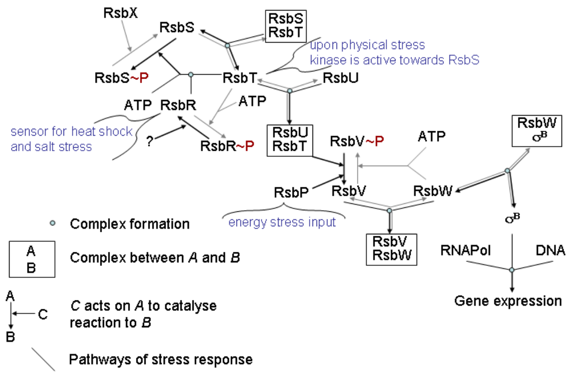
Graphs: ubiquitous and beautiful

- Abstracts over detail to highlight important features
  - Use spacial layout for names and edges for binding
- Visual intuition: (dis)connectedness, paths, and inclusion encode meaning
- A rich and widely used notation: (formally and informally)
  - Specification/Design: UML
  - Modelling/Simulation: Biological Networks, Quantum Information
  - Engineering/Optimisation: Finite State Machines
  - Proof: Quantum Mechanics and Information
- Applications in... Biology ; Physics ; Engineering ; Computer Science

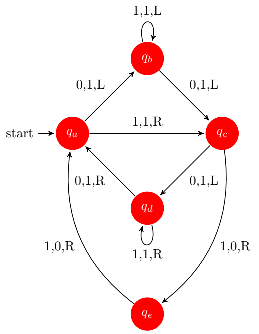
Graphs: UML



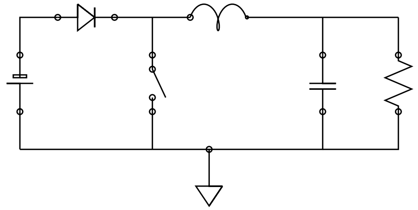
Graphs: Biological Interactions

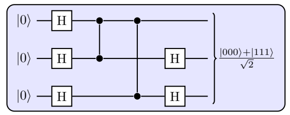


Graphs: State Machines

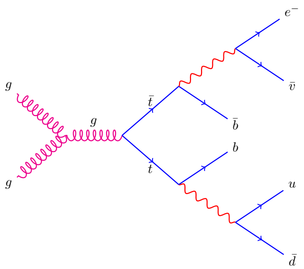


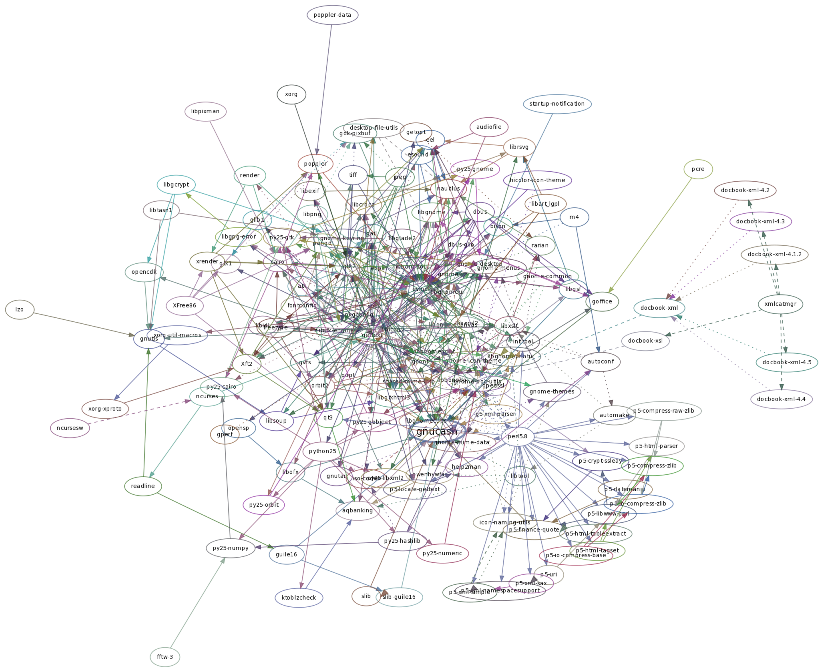
Graphs: circuits, electrical and quantum





Graphs: Feynman Diagrams



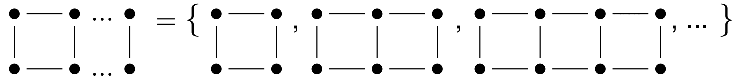


Expressivity breeds complexity!

- Visualise: Human understanding is modulo graph-layout.
- Families of similar graphs? (e.g. common repeated structures)
- Difficult to reason about: Composition? Equivalences? Induction/recursion principles? Many generic algorithms, such as depth-first search/spanning trees, do not respect the domain's semantics.
- Dynamics? Interesting systems are not static, how do graphs change? proof, evaluation, evolution...

Complexity breeds computer science?

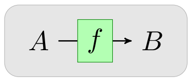
- Visualise: use other people's solutions...
- Families: formal account of ellipses notation



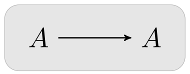
- Formalism that allows computer aided manipulation
- Dynamics captures by graph rewriting (evolution, evaluation)
- **Need: practical tools to represent and reason with graphs**

Tensor ($\otimes$) Graphs

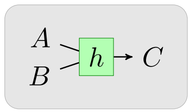
- Graphs where **Vertices** are processes/functions/relations and **Edges** are states/objects and paths describe change/flow of information.
- $f : A \mapsto B$



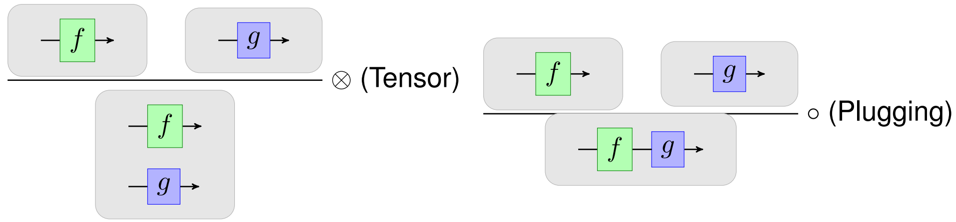
- $I : A \mapsto A$ (identity)



- $h : (A \otimes B) \mapsto C$

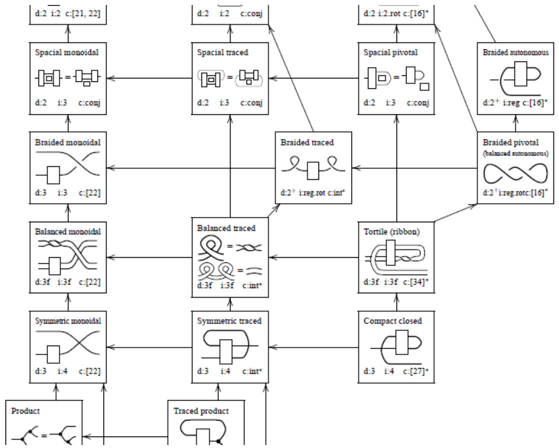


Two forms (de)composition for $\otimes$Graphs



- These are graphical presentations of monoidal categories.
- **Coherence results: prove equivalence between categories and graphs** (Mac Lane, Kelly and Laplaza, Joyal and Street, Selinger)

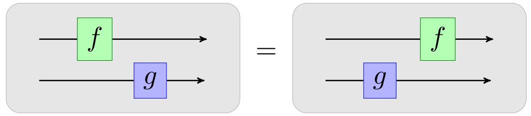
Lots of kinds of monoidal graphs



Symmetric Monoidal Categories

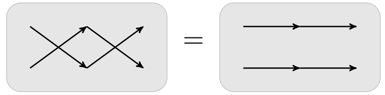
Bifunctoriality of $\otimes$

$$(f \otimes I) \circ (I \otimes g) = (I \otimes f) \circ (g \otimes I)$$



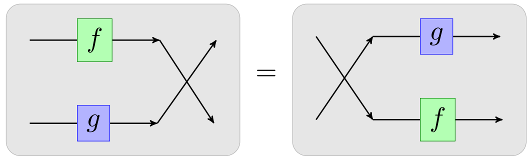
Symmetry of braiding (σ)

$$\sigma \circ \sigma = I$$

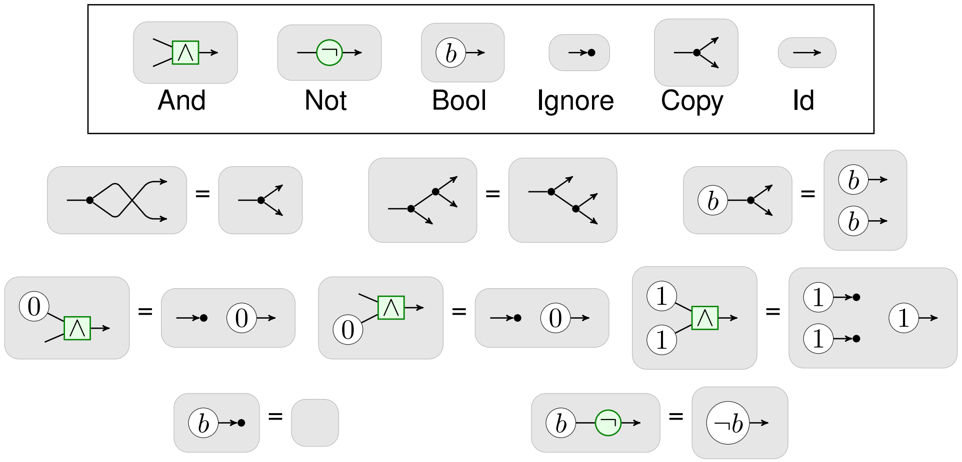


Sliding Boxes: tensor/sigma commutativity

$$(f \otimes g) \circ \sigma = \sigma \circ (g \otimes f)$$

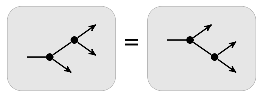


Example: Boolean Circuit Graphical Equations

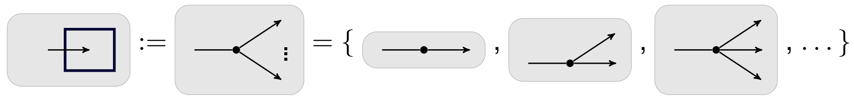


Expressiveness

- Symmetries suggest changing representation:



$\Downarrow$

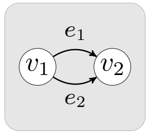


- A formal (and finitely expressible) ellipses notation?
- Traditional graph transformation machinery lacks: graphical derived equations, interfaces for graphs, graphical ellipsis notation.

Formalism and Implementation

- **Need a formal account of graphs, half-edges, tensor, plugging, ellipses notation, matching, and finally rewriting.**

- Graph: $(s : E \mapsto V, t : E \mapsto V)$



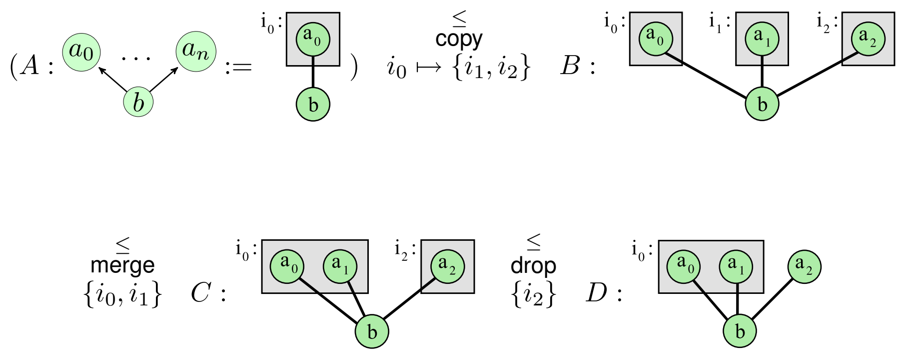
$(s := \{e_1 \mapsto v_1, e_2 \mapsto v_1\},\ t := \{e_1 \mapsto v_1, e_2 \mapsto v_1\})$
(not a binary relation!)

- **Exterior vertices** ('half'-edges) := a subset of the vertices;
  These define the graph's *interface*; can have many incident edges!

- Tensor := disjoint union

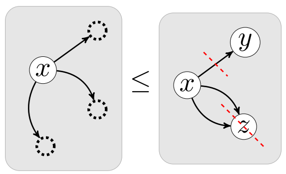
!-Box Graphs

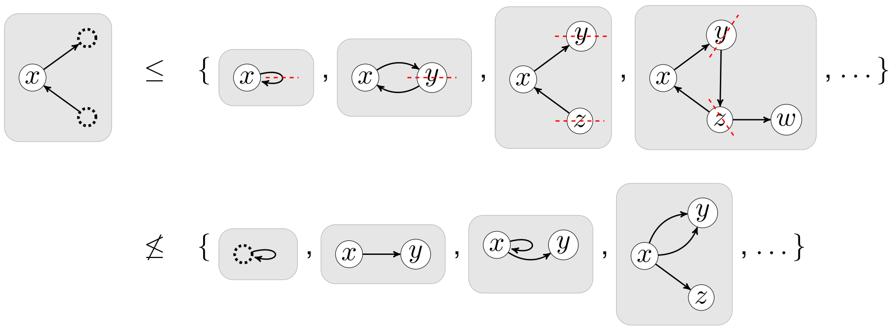
The interpretation of !-box graph is the family of ground expansions



Matching

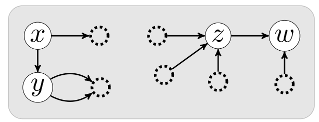
Cut-out the source graph from the target graph
e.g. cut edge x-y, and cut vertex z:
(not unique)

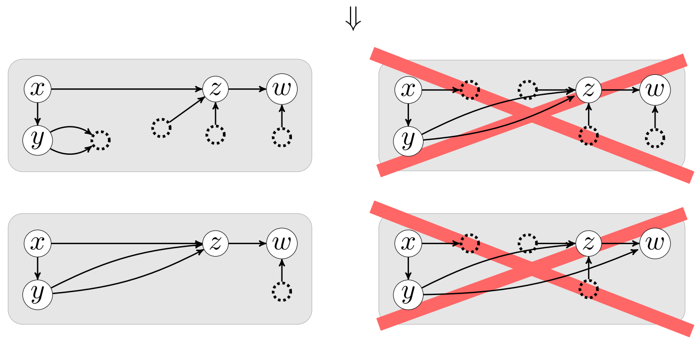




Plugging

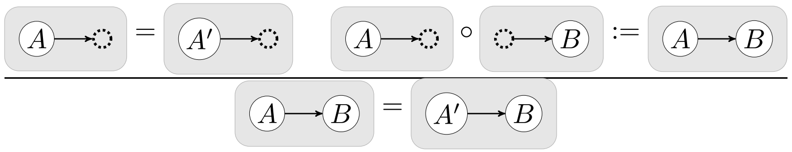
Unify edges to exterior, respect the vertices, e.g:
≈ a minimal graph w.r.t. shared *joining* graph
(Defined as a pushout)



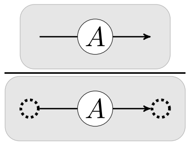


Properties

Plugging and rewriting commute:



Adequacy:



Matching forms a partial order.

Quantum Information: why is it important

- Quantum Money: unforgable (still blue-sky)
- Quantum Computation: speedup? polynomial-time factorisation (significant experimental work under way)
- Quantum Protocols: Secure communication - built-in physically (Have working systems, claims to be in mobile devices in a few years)
- Complexity Results: Alternative formulations of complexity classes. e.g. quantum accounts of PSPACE (Jain,Ji,Upadhyay.Watrous 2009) and EXP (Kobayashi, Matsumoto, 2003)

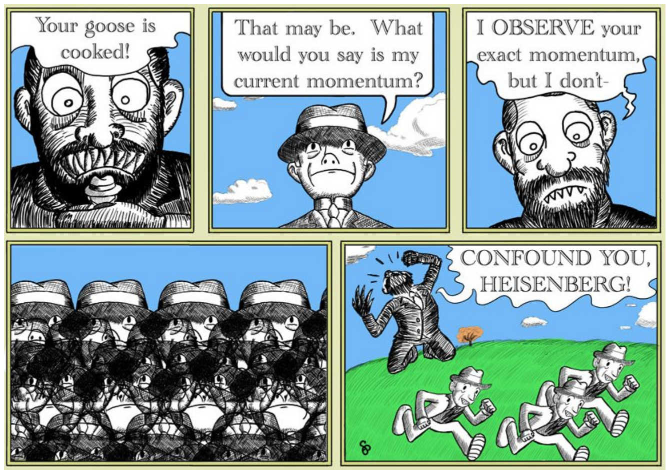
Quantum Information

- Quantum computation as circuit-like graphs (Abramsky, Duncan, Coecke, ...)

$$\frac{\text{graphical calculi}}{\text{quantum mechanics}} = \frac{\lambda\text{-calculus}}{\text{Turing machines}}$$

  - Easier to understand and manipulate.
  - Certain properties have a natural graphical representation (e.g. disjointness in graph $\Rightarrow$ separable state)
  - Abstract algebra of graphs has other applications
- Hard to reason with manually $\Rightarrow$ develop tool support: Quantomatic
  - Generalised formalism for graph rewriting (includes some ellipses notation).

Quantumness: Complementary Observables



http://dresdencodak.com/

Demo

Summary

- Graphical presentations of monoidal categories as tensor-graphs
- Formalised and implemented tensor-graphs with interfaces, some ellipses notation, and compositional matching/rewriting
- Supports reasoning about quantum information
- Graphical-equations as a tool for representation and reasoning
- Closely related formalisms: Bi-graphs (+hierarchical nodes define spaces, -no ellipses notation); Graph transformations (not for tensor graphs)
- Thanks!