Using Bayes Nets for Quantum Inferring of Acausal Systems of Events

Marcin Przewięźlikowski[1], Michał Grabowski[1], Katarzyna Rycerz[1], Dariusz Kurzyk[2], Piotr Gawron[2]

[1] Department of Computer Science, AGH, Krakow, Poland

[2] Polish Academy of Sciences, Institute of Theoretical and Applied Informatics, Gliwice, Poland

Agenda

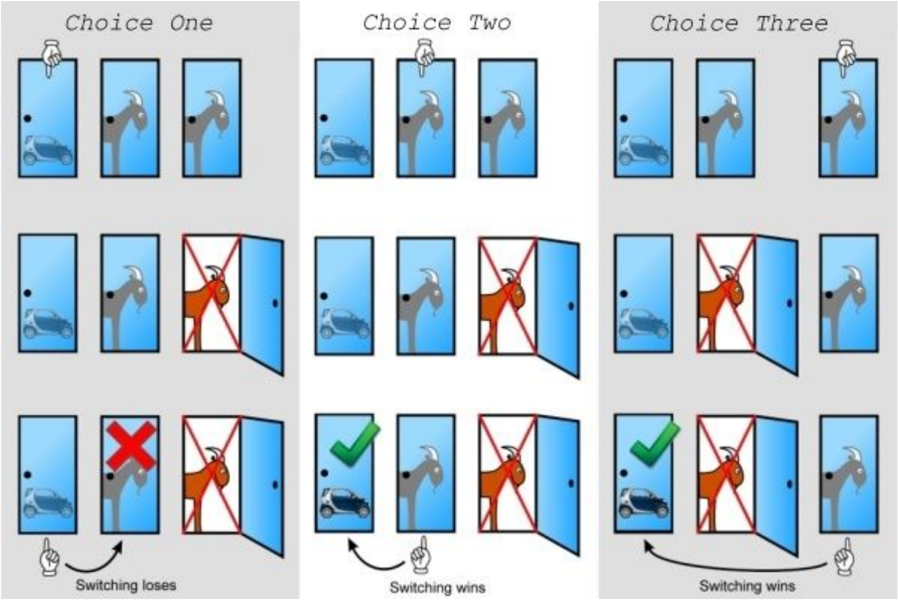
- Non-cooperative games
  - Monty Hall
  - Prisoner’s Dilemma
- Quantum superpositions and strategies in games
- Bayesian Networks
- AcausalNets.jl and how it differs from other Bayesian Network frameworks
- Modifications to inference algorithm
- Research on non-cooperative games
- Bibliography

Non-cooperative games

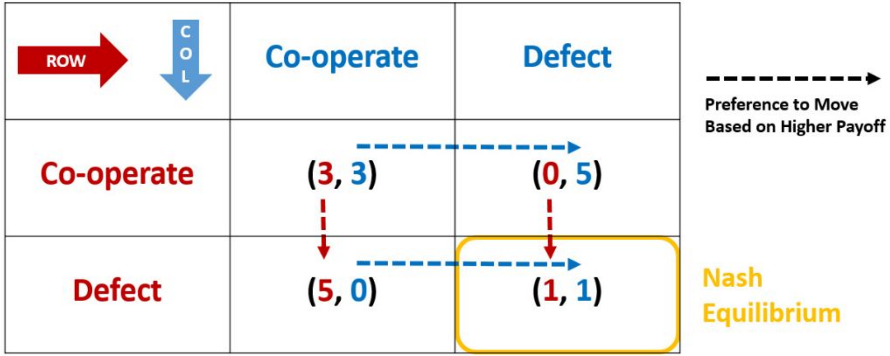
- there are multiple players who are competing against one another.
- players must employ various strategies
- those strategies must account for
  - the rules of the game
  - most importantly, the strategies of the opponents.

The Monty Hall problem

- Three doors
- One prize
- After the Player’s first choice of the door, the Host opens one of the remaining two door is open, to reveal that it is empty
- The Player can then alter their original choice



Prisoner’s Dilemma (and its’ iterative version)



- Two prisoners
- Each prisoner either cooperates with or tells on the other one
- Based on their actions, each prisoner receives an appropriate reward
- In the iterative version of the problem, prisoners know each other’s previous choices and can act on their basis

Quantum game theory

- *Nash Equlibrium* - a set of strategies choices, where all players have chosen a strategy and none of the players has anything to gain by changing his strategy, while other players stick by their original strategies.
- Extends the classical game theory with an option of quantum superpositions between events occuring in a game
- Given quantum effects, new features of such games can be discovered, such as previously unknown Nash Equlibria or more optimal strategies for all players
- The classical version of a game must be a special case of a quantum version of that game

How does quantum superposition probability matrix look?

An example of quantum superposition between two 3-dimensional variables

$$AB_{entangled} = \frac{1}{3}(|00\rangle + |11\rangle + |22\rangle)(\langle 00| + \langle 11| + \langle 22|)$$

In [4]:

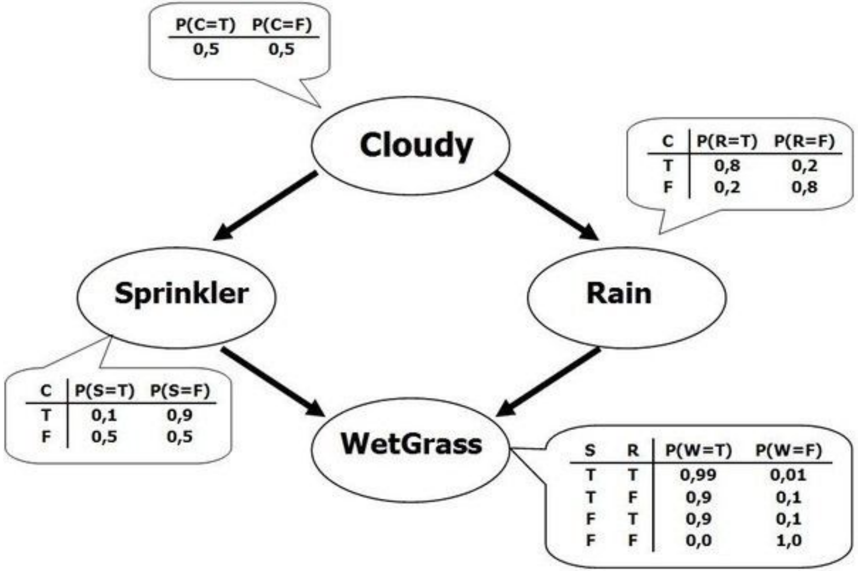
```
AB_entangled=1/3*(ket(0,9)+ket(4,9)+ket(8,9))* (bra(0,9)+bra(4,9)+bra(8,9))
```

```
9×9 Array{Complex{Float64},2}:
 0.333333+0.0im  0.0+0.0im  0.0+0.0im  …  0.0+0.0im  0.333333+0.0im
      0.0+0.0im  0.0+0.0im  0.0+0.0im     0.0+0.0im       0.0+0.0im
      0.0+0.0im  0.0+0.0im  0.0+0.0im     0.0+0.0im       0.0+0.0im
      0.0+0.0im  0.0+0.0im  0.0+0.0im     0.0+0.0im       0.0+0.0im
 0.333333+0.0im  0.0+0.0im  0.0+0.0im     0.0+0.0im  0.333333+0.0im
      0.0+0.0im  0.0+0.0im  0.0+0.0im  …  0.0+0.0im       0.0+0.0im
      0.0+0.0im  0.0+0.0im  0.0+0.0im     0.0+0.0im       0.0+0.0im
      0.0+0.0im  0.0+0.0im  0.0+0.0im     0.0+0.0im       0.0+0.0im
 0.333333+0.0im  0.0+0.0im  0.0+0.0im     0.0+0.0im  0.333333+0.0im
```

Generated with QI.jl library - https://github.com/ZKSI/QI.jl

Bayesian Networks

- probabilistic models
- they represent sets of variables (or events), which are conditionally dependent on each other
- inferring probability distributions of variables through belief propagation
- inference algorithms can take certain observations into account and infer probability distributions of unobserved variables

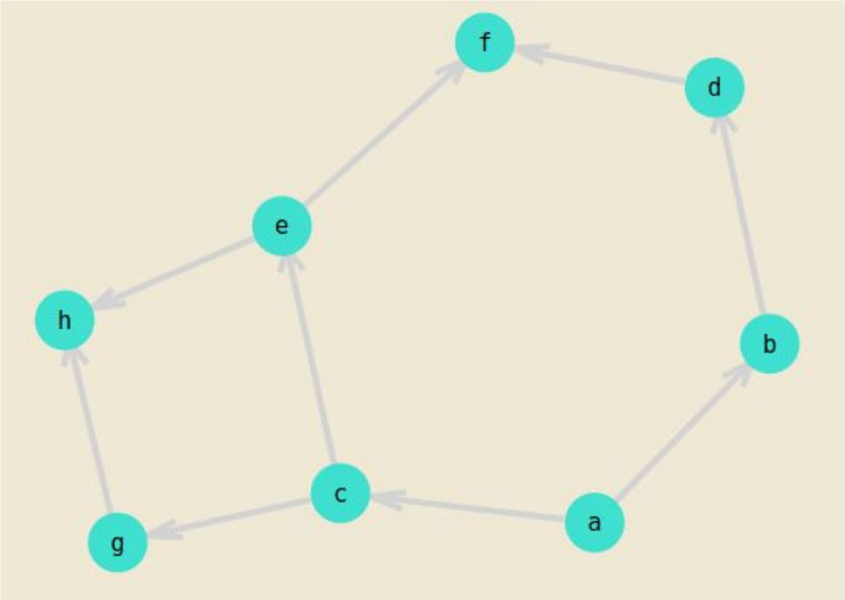


AcausalNets.jl

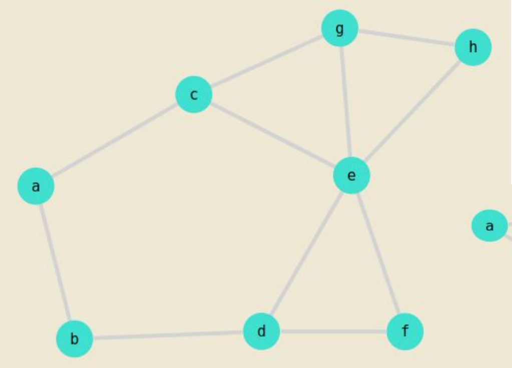
- library for building Discrete Bayesian Networks and inferring belief through them
- two inference algorithms:
  - naive (calculate the joint probability of the entire BN and call it a day)
  - junction tree algorithm
- as opposed to similar software:
  - written with Quantum (Acausal) Bayesian Networks in mind
  - thanks to Julia's type system, heavily generalizable
  - builds networks from systems of variables (as opposed to variables)
- implemented in Julia (0.7)
- https://github.com/mikegpl/AcausalNets.jl

Junction Tree algorithm for belief propagation

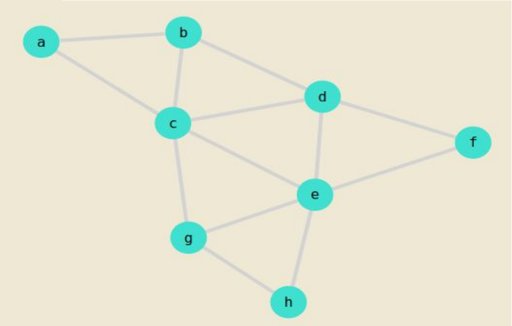
Given a Bayesian Network:



1. Moralization of the graph.
2. Triangulation of the graph.
3. Obtaining cliques of the triangulated graph



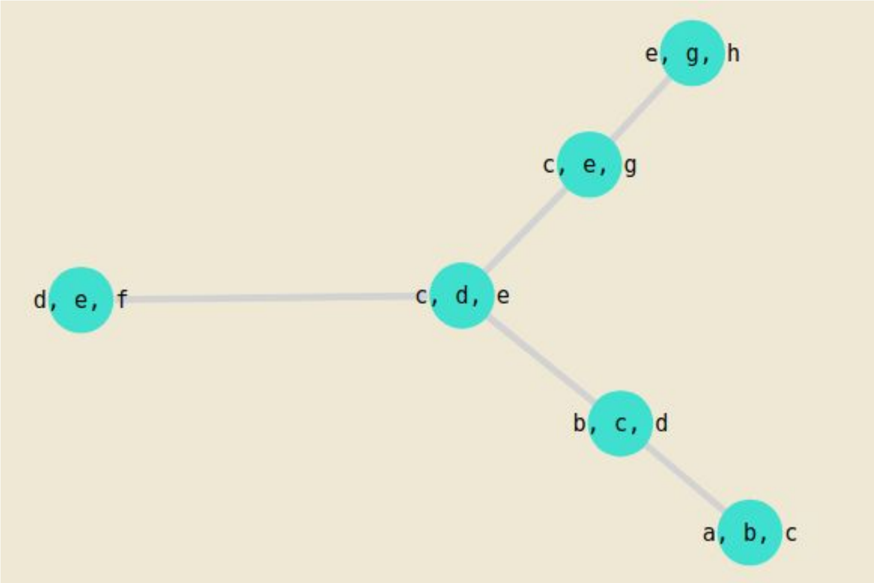
(1)



(2)

Junction Tree algorithm for belief propagation (2)

4. Construction of a Junction Tree from the obtained cliques



5. Each system from the original DBN chooses its’ parent clique

6. Initialization of cliques’ probability distributions

6. Evidence application

7. Belief propagation by passing messages between adjacent vertices of the tree

Our modifications to the Junction Tree algorithm

- the building blocks of Junction Trees are (merged) systems from the DBN
- enforcement of creating a clique with all the variables one wants to infer
  - performed between steps (1) and (2) of the original algorithm
  - this may potentially make the Junction Tree less optimal, but it’s the only way to obtain the joint system we are looking for
- ordering and dimensionality matters!
  - it is imperative to keep the variables sorted in topological order, so that no operation gets mixed up
  - when multiplying systems, one must also multiply the factors by an appropriate identity (and then permute)
- merging systems and message passing is done using modified multiplication operators
  - $\star$ - a modified matrix multiplication operator
  - $\otimes$ - Kronecker product

A strange case of the ⋆ operator

- quantum probabilities are expressed with matrices
- multiplication of probability matrices is performed with a ⋆**(1)** operator
- ⋆**(n)** is parametrized and introduces a family of probability of operators
- ⋆**(1)** is non-commutative and and non-associative, which makes it imperative to perform the multiplication of probability matrices in the right order
- the ⋆ operator is defined variously across different sources. Its definition is tightly associated with the order in which probability matrices should be multiplied
- The order of multiplication doesn’t matter in classical case
- We are still researching literature on Quantum Belief Propagation in order to make sure our implementation is correct

$$A \star^{(n)} B = \left(A^{\frac{1}{2n}} B^{\frac{1}{n}} A^{\frac{1}{2n}}\right)^{n}$$

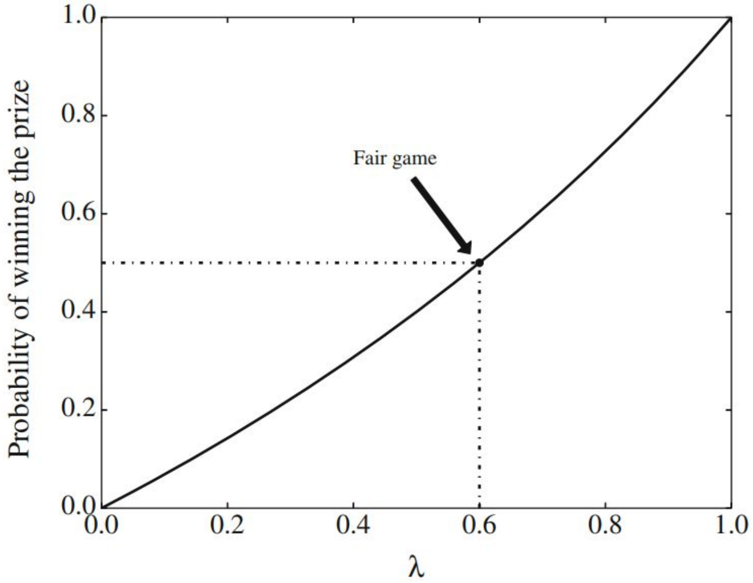
$$\rho_V = \left(\star^{(n)}\right)_{j=1}^{N} \rho^{(n)}_{v_j|v_{j-1}v_{j-2}\ldots v_1}$$

$$\left(\star^{(n)}\right)_{j=1}^{N} A_j = \left(\left(\left(A_1 \star^{(n)} A_2\right) \star^{(n)} A_3\right) \ldots\right) \star^{(n)} A_N$$

⋆ operator, and the right order of matrix multiplication as defined in [6]

Research on quantum strategies

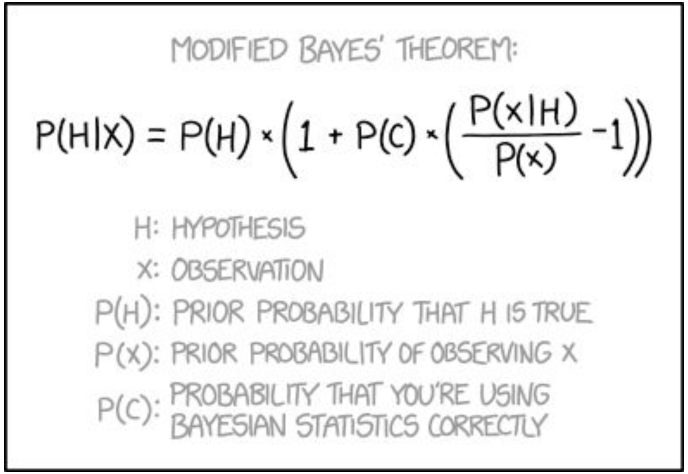
- we’ve managed to reproduce results research on the Monty Hall Game conducted in **[3]**
- given a specific quantum superposition between the placement of the prize and the initial choice of the player, a Nash Equilibrium can be found
- the player has 0.5 chance of winning, regardless of their strategy



Probability of winning the prize based on the combination of two specific quantum entanglements of A and B.
Source: [3]

Future plans

- We are still researching the consequences of the ⋆ operator on the Junction Tree algorithm, as this is a wide research subject
  - reversing the order of the ⋆ multiplication yields results of uncertain meaning, which at a first glance seem to be more intuitive (with regards to the classical game)
  - our implementation of message passing in the Quantum Belief Propagation algorithm is still occasionally prone to errors - we still need to work on that
- We plan on performing calculations on the iterative Prisoner's Dilemma in order to find interesting quantum strategies



Bibliography

[1] D. Heckerman, M. P. Wellman. Bayesian networks. Communications of the ACM 38.3 (1995): 27-31.

[2] C. Huang and A. Darwiche. Inference in belief networks: A procedural guide. International Journal of Approximate Reasoning, 15(3):225 – 263, 1996. ISSN 0888-613X. doi: https://doi.org/10.1016/S0888-613X(96)00069-2. URL http://www.sciencedirect.com/science/article/pii/ S0888613X96000692.

[3] D. Kurzyk and A. Glos. Quantum inferring acausal structures and the monty hall problem. Quantum Information Processing, 15(12):4927–4937, Dec 2016. ISSN 1573-1332. doi: 10.1007/s11128-016-1431-8. URL https://doi.org/10.1007/s11128-016-1431-8.

[4] J. Eisert, M. Wilkens and M. Lewenstein (1998). Quantum Games and Quantum Strategies. Physical Review Letters. 83. 10.1103/PhysRevLett.83.3077.

[5] J. Bezanson, A. Edelman, S. Karpinski, and V. Shah. Julia: A fresh approach to numerical computing. SIAM Review, 59(1):65–98, 2017. doi: 10.1137/ 141000671. URL https://doi.org/10.1137/141000671.

[6] M.S. Leifer, D. Poulin, Quantum Graphical Models and Belief Propagation, Annals of Physics, Volume 323, Issue 8, 2008, Pages 1899-1946, ISSN 0003-4916, https://doi.org/10.1016/j.aop.2007.10.001.