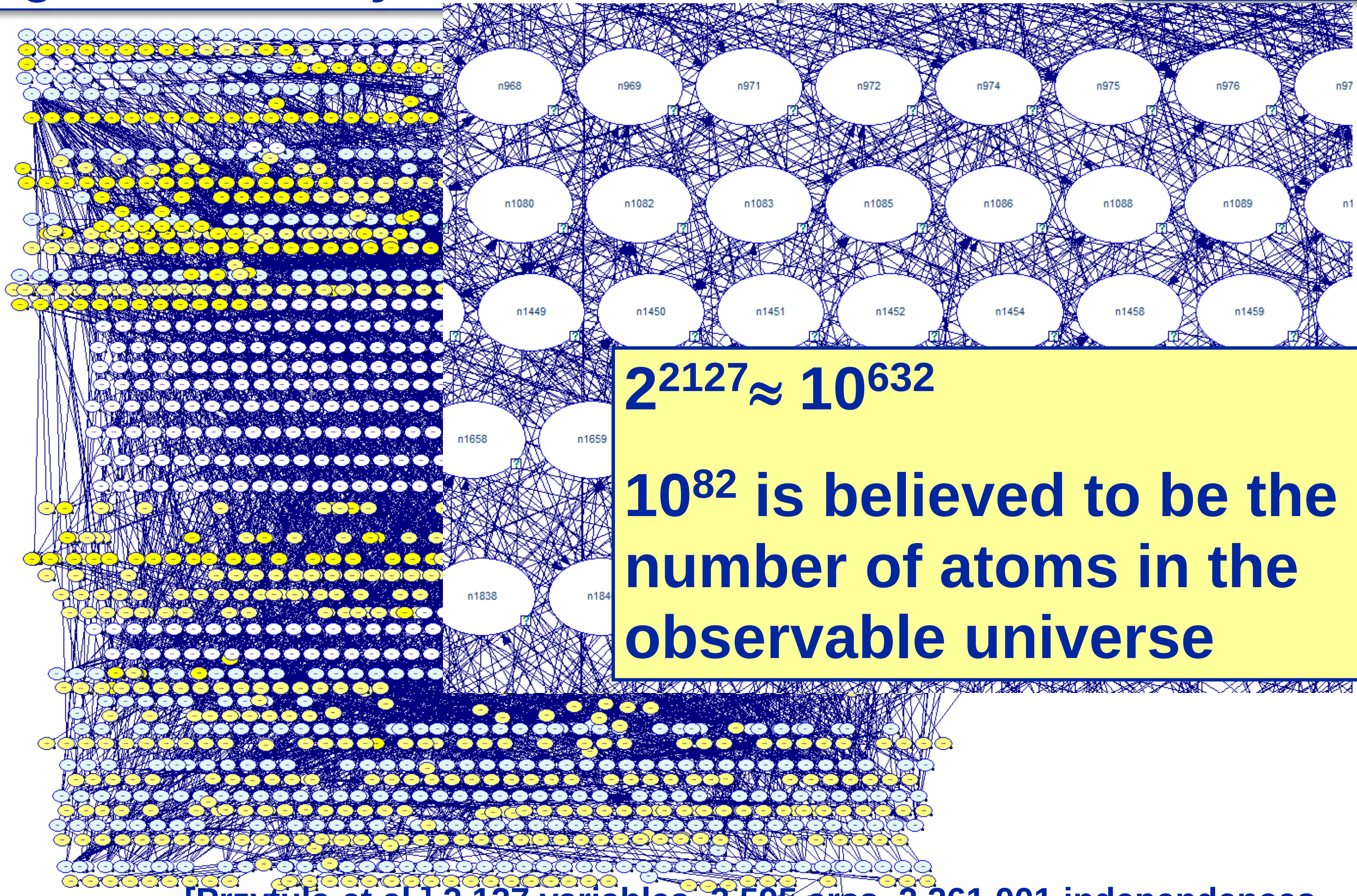
“Canonical” Models

Marek J. Drużdżel

Politechnika Białostocka

m.druzdzel@pb.edu.pl
http://aragorn.wi.pb.bialystok.pl/~druzdzel

Practical BN models can be very large and densely connected



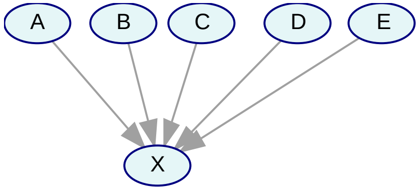
$2^{2127} \approx 10^{632}$

10^{82} is believed to be the number of atoms in the observable universe

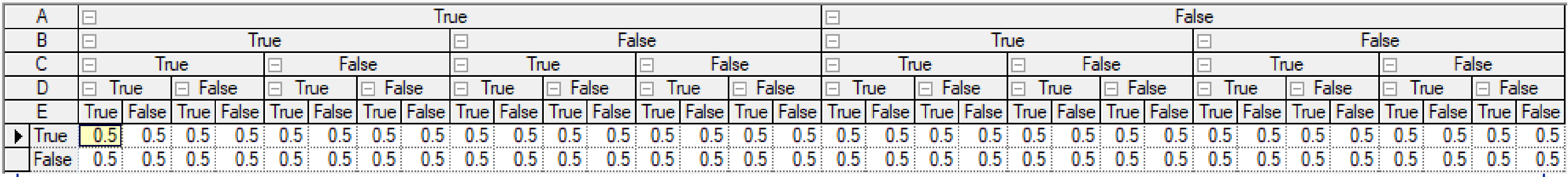
[Przytula et al.] 2,127 variables, 3,595 arcs, 2,261,001 independences, 12,351 numerical parameters (instead of $2^{2,127} \approx 10^{632}$!)

Fundamental problem: (too) many parameters

- **Size of conditional probability tables (CPTs) grows exponentially in the number of parents**
- **This can become quickly unmanageable**



5 parents

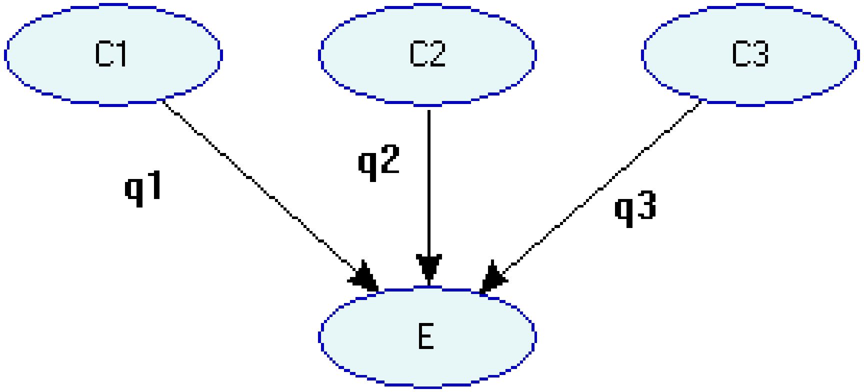


A	True																False															
B	True								False								True								False							
C	True				False				True				False				True				False				True				False			
D	True		False		True		False		True		False		True		False		True		False		True		False		True		False		True		False	
E	True	False	True	False	True	False	True	False	True	False	True	False	True	False	True	False	True	False	True	False	True	False	True	False	True	False	True	False	True	False	True	False
True	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
False	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5

- **Not uncommon to see 10-15 parents (would need between 1,024 and 32,768 parameters).**
- **A lot of work for experts or a lot of data needed.**

Solution: Canonical gates

- Various solutions were proposed, but one of them seems to be most popular and useful: Noisy-OR
- We assume that all nodes are binary {present, absent}
- We specify the interaction between the parents and the child by means of one numerical parameter q_i per parent



Solution: Canonical gates

Conditions that have to be fulfileld in practice for Noisy-OR to be applicable:

- **There should be a causal mechanism for each parent such that the parent is able to impact the child variable in the absence of the other parents.**
- **The causal mechanisms through which each parent influences the child should be independent?**
- **If there are other, unmodeled causes, they should be independent of the modeled causes.**

Noisy-OR: The meaning of q_i?

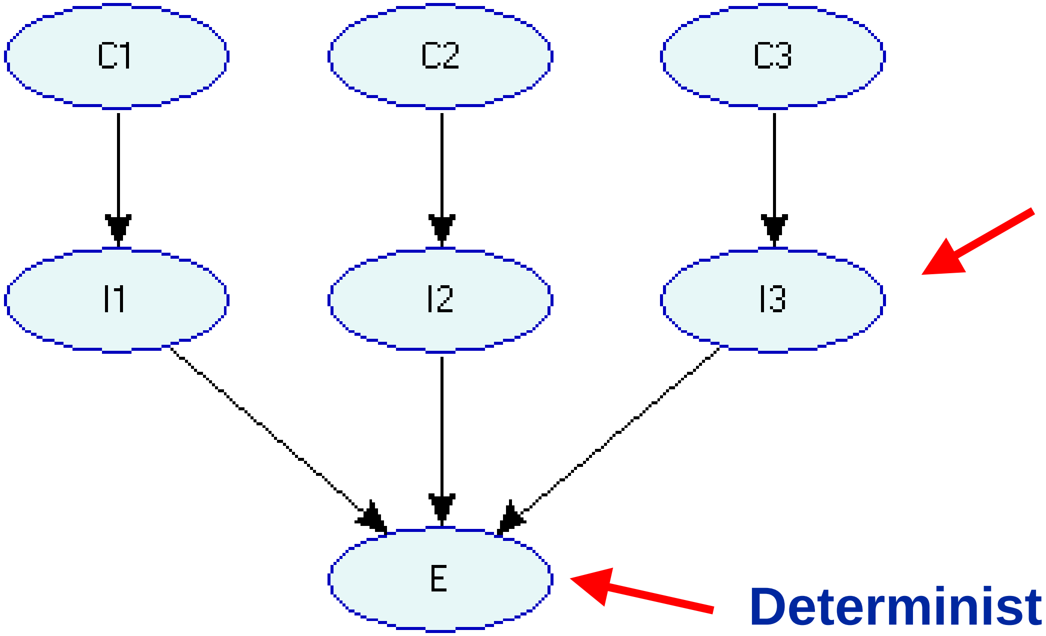
q_i is the probability that E=*present* given C_i=*present* and all other parents $C_{j \neq i}$=*absent*

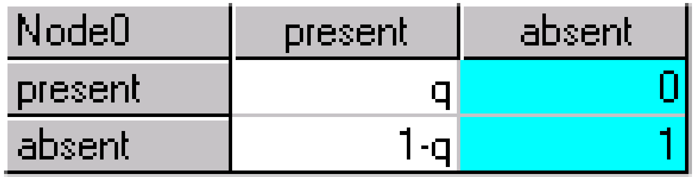
$$q_i = P(E=\textit{present} \mid C_1=\textit{absent}, \ldots, C_i=\textit{present}, \ldots, C_n=\textit{absent})$$

Why is it called Noisy-OR?

If all parameters q_i=1, noisy-OR becomes logical OR

Here is an alternative representation of Noisy-OR





Node0	present	absent
present	q	0
absent	1-q	1

Deterministic OR

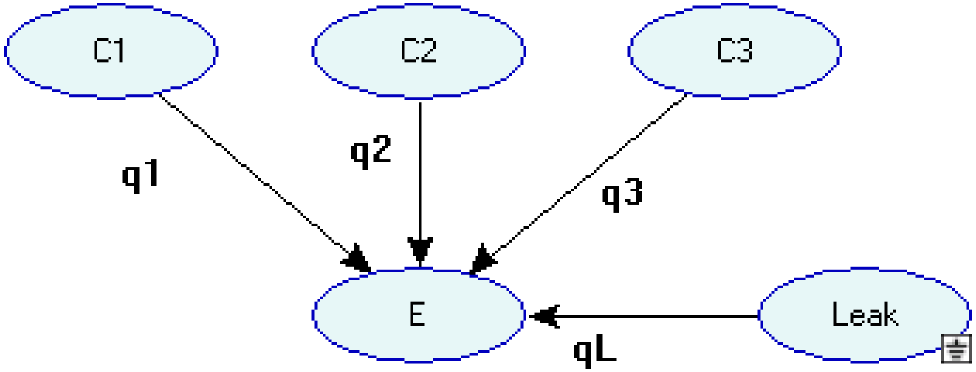
Noisy-OR vs. CPT

Noisy-OR always defines a unique CPT (i.e., you can always calculate the CPT that is defined by a noisy-OR gate)

$$P(E = absent \mid C1,\ldots,Cn) = \prod_{C_i = present} (1 - q_i)$$

Leaky Noisy-OR

- Noisy-OR assumes that the effect will be absent with probability 1 if all the causes are absent. This is not very realistic
- Leak is a special dummy node, that represents the influence of all unmodeled causes on the effect node
- Leak is always present



Leaky Noisy-OR: Parameters

- Leaky Noisy-OR is an extension of the Noisy-OR
- Two parameterizations of leaky Noisy-OR: due to Henrion and Diez (*compound* and *net* parameters)
- They are mathematically equivalent, however they imply different questions in knowledge elicitation

Leaky Noisy-OR: Diez

Leak probability q_L:

$$q_L = P(E = present \mid C1 = absent, .., CN = absent)$$

Link probability q_i:

$$q_i = P(E = present \mid C1 = absent, ..., Ci = present, CN = absent, L = absent)$$

How to calculate the CPT:

$$P(E = absent \mid C1, ..., Cn) = (1 - q_L) \prod_{C_i = present} (1 - q_i)$$

Leaky Noisy-OR: Henrion

- **Leak probability p_L: (same as Diez)**

$$p_L = P(E = present \mid C1 = absent, .., CN = absent)$$

- **Link probability p_i: (no leak term)**

$$p_i = P(E = present \mid C1 = absent, ..., Ci = present, CN = absent)$$

- **How to calculate CPT:**

$$P(E = absent \mid C1, ..., Cn) = (1 - p_L) \prod_{C_i = present} \frac{1 - p_i}{1 - p_L}$$

Henrion vs. Diez

- **They imply different questions to ask of experts:**
- **Henrion:**

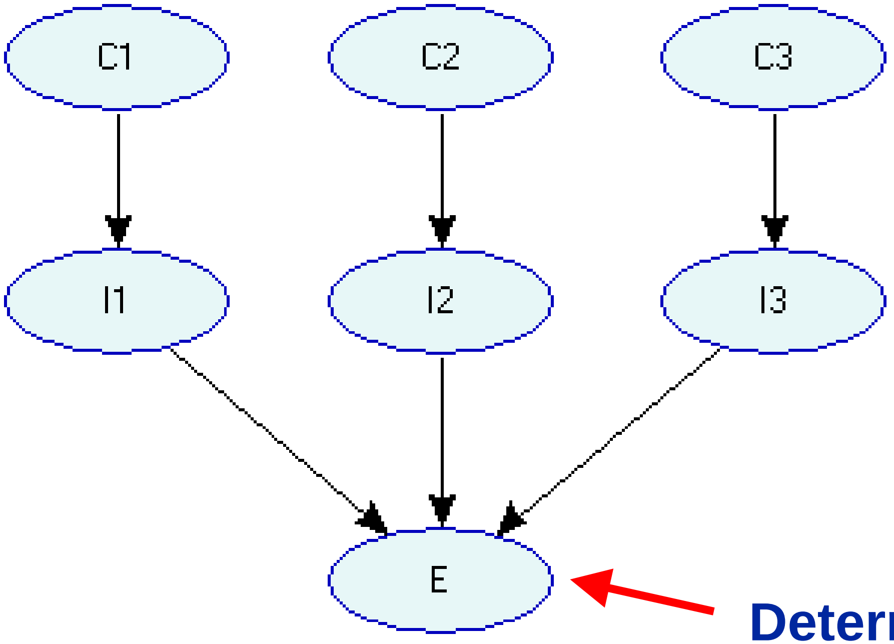
  "What is the probability that E is present given that C_i is present and all other modeled causes are absent?"

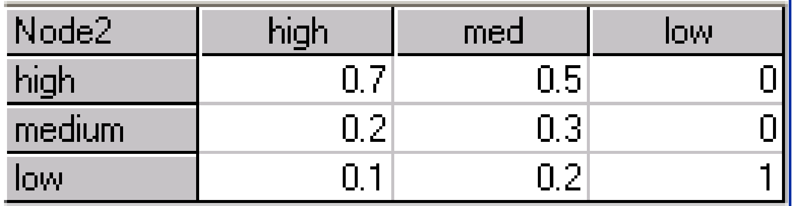
- **Diez:**

  "What is the probability that E is present given that C_i is present and all other modeled and unmodeled causes are absent?"

Noisy-MAX

Noisy-MAX is a version of Noisy-OR for multi-valued nodes.



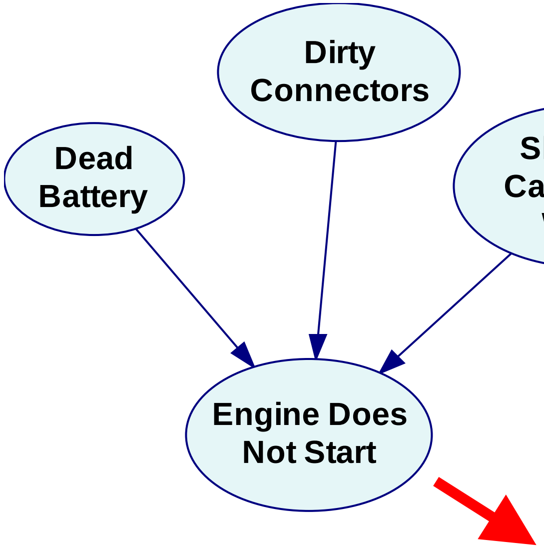


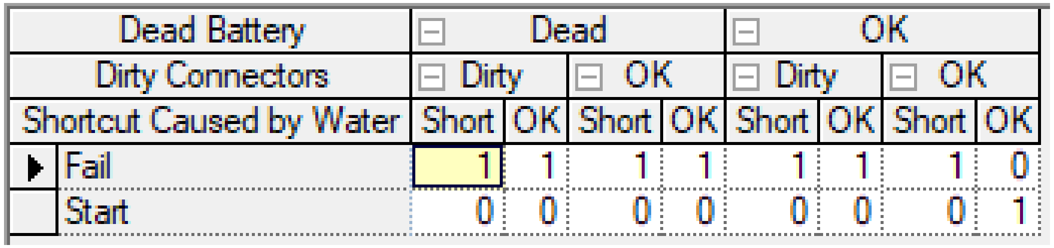
Node2	high	med	low
high	0.7	0.5	0
medium	0.2	0.3	0
low	0.1	0.2	1

Deterministic MAX

Example

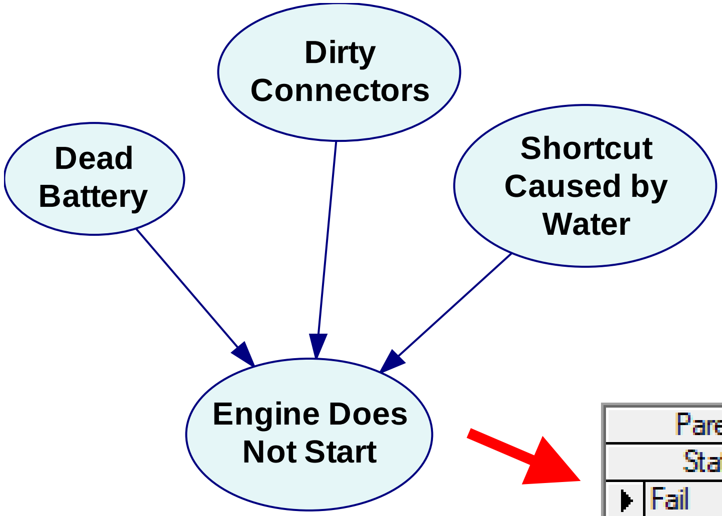
Deterministic OR



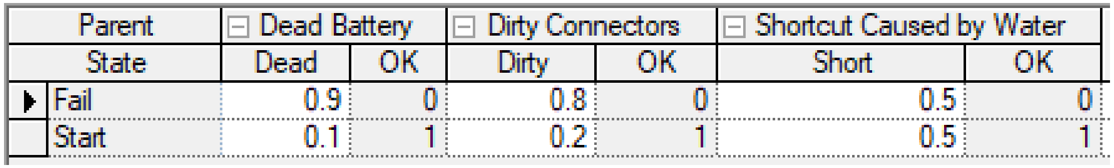


Dead Battery	Dead				OK			
Dirty Connectors	Dirty		OK		Dirty		OK	
Shortcut Caused by Water	Short	OK	Short	OK	Short	OK	Short	OK
Fail	1	1	1	1	1	1	1	0
Start	0	0	0	0	0	0	0	1

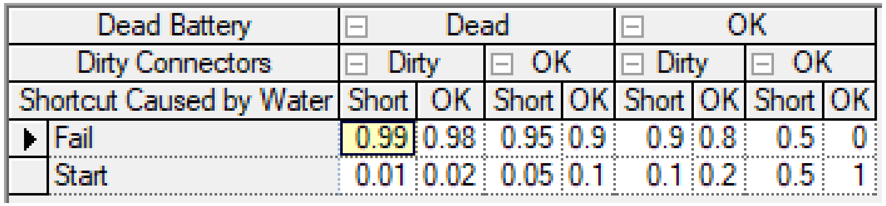
Noisy-OR



$$P(E = absent \mid C1,\ldots,Cn) = \prod_{C_i = present} (1 - q_i)$$



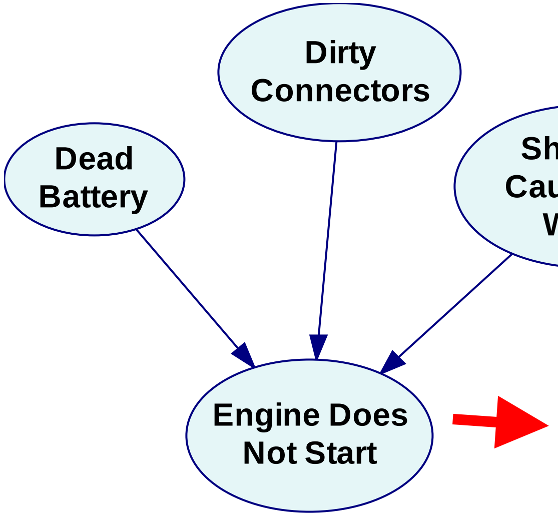
Parent	Dead Battery		Dirty Connectors		Shortcut Caused by Water	
State	Dead	OK	Dirty	OK	Short	OK
Fail	0.9	0	0.8	0	0.5	0
Start	0.1	1	0.2	1	0.5	1

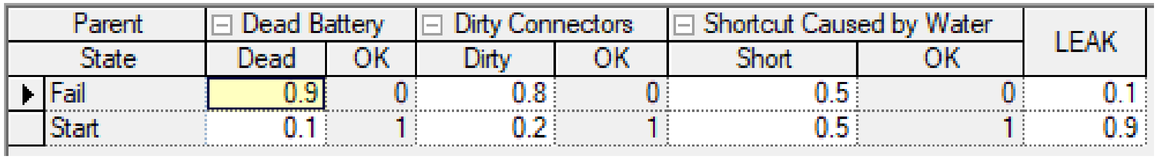


Dead Battery	Dead				OK			
Dirty Connectors	Dirty		OK		Dirty		OK	
Shortcut Caused by Water	Short	OK	Short	OK	Short	OK	Short	OK
Fail	0.99	0.98	0.95	0.9	0.9	0.8	0.5	0
Start	0.01	0.02	0.05	0.1	0.1	0.2	0.5	1

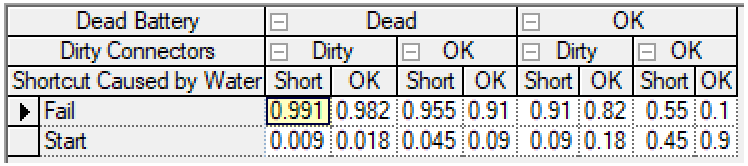
Leaky Noisy-OR

We use a “leak” or “background” probability to model all unmodeled causes





Parent	Dead Battery		Dirty Connectors		Shortcut Caused by Water		LEAK
State	Dead	OK	Dirty	OK	Short	OK	
Fail	0.9	0	0.8	0	0.5	0	0.1
Start	0.1	1	0.2	1	0.5	1	0.9



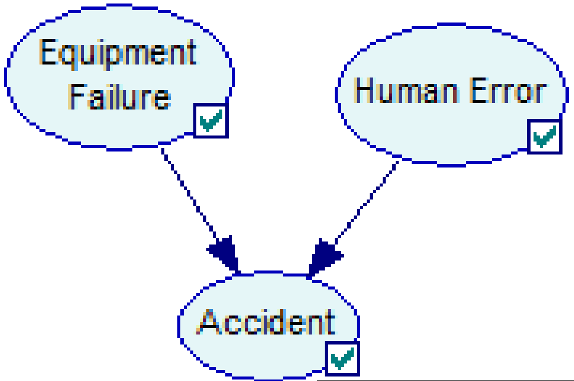
Dead Battery	Dead				OK			
Dirty Connectors	Dirty		OK		Dirty		OK	
Shortcut Caused by Water	Short	OK	Short	OK	Short	OK	Short	OK
Fail	0.991	0.982	0.955	0.91	0.91	0.82	0.55	0.1
Start	0.009	0.018	0.045	0.09	0.09	0.18	0.45	0.9

$$P(E = absent \mid C1,\ldots,Cn) = (1-q_L)\prod_{C_i = present}\frac{1-q_i}{1-q_L}$$

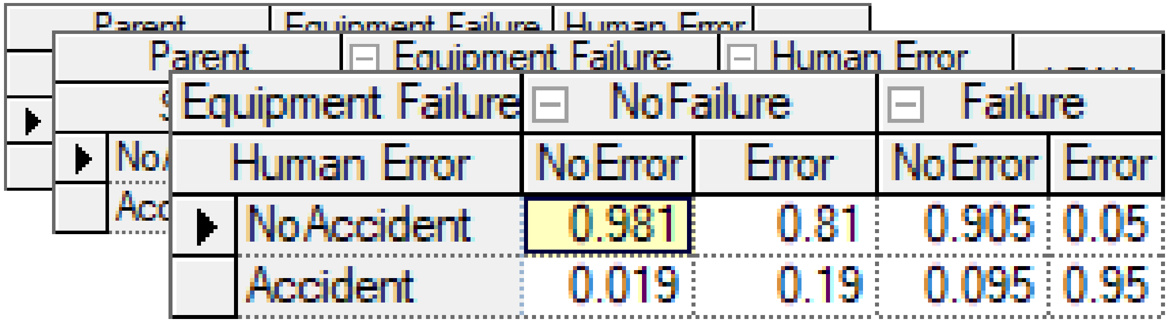
Noisy-AND/MIN

Based on the DeMorgan's law:

$$X \wedge Y = \neg(\neg X \vee \neg Y))$$



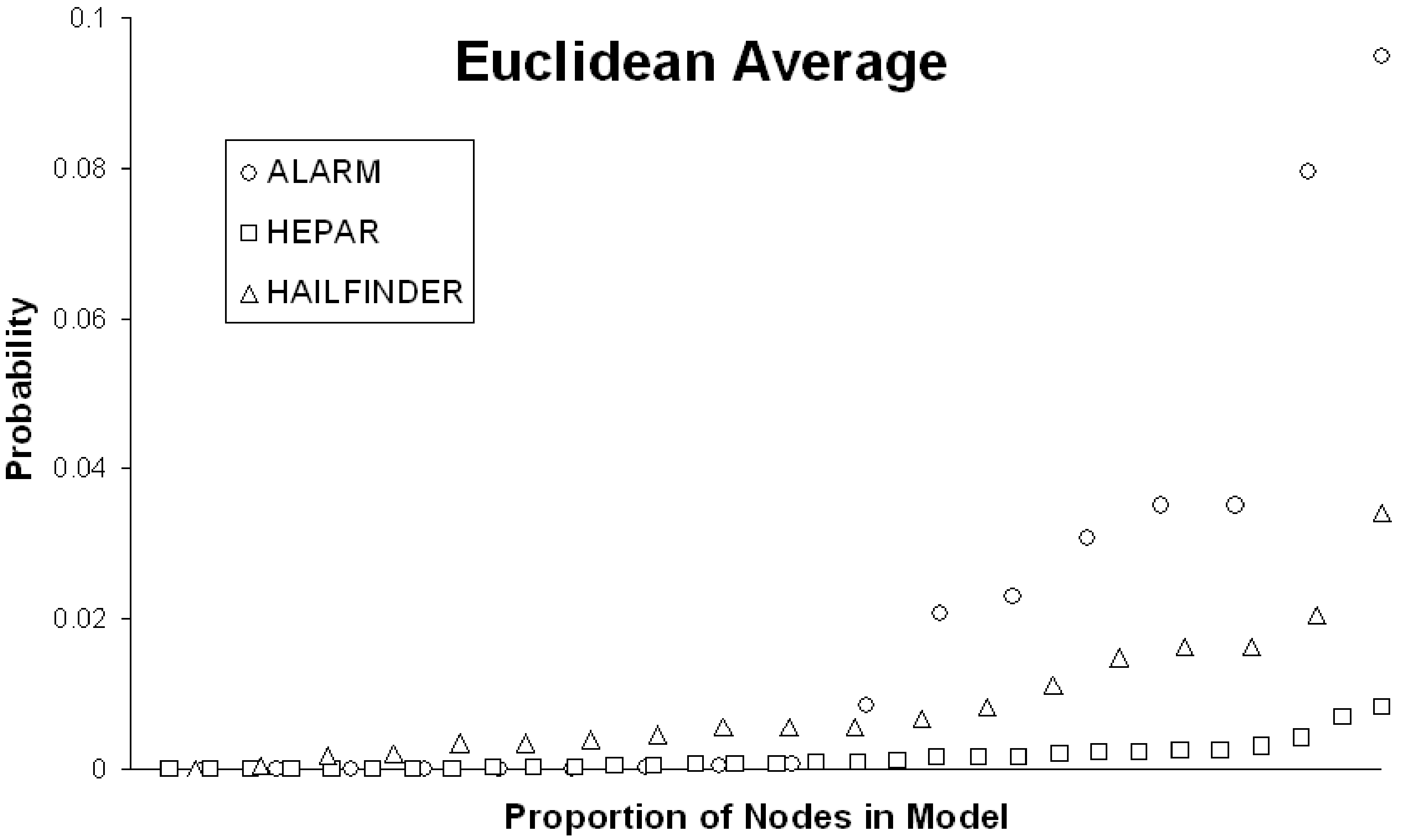
We negate all nodes by reversing the order of their states.



Equipment Failure	NoFailure		Failure	
Human Error	NoError	Error	NoError	Error
NoAccident	0.981	0.81	0.905	0.05
Accident	0.019	0.19	0.095	0.95

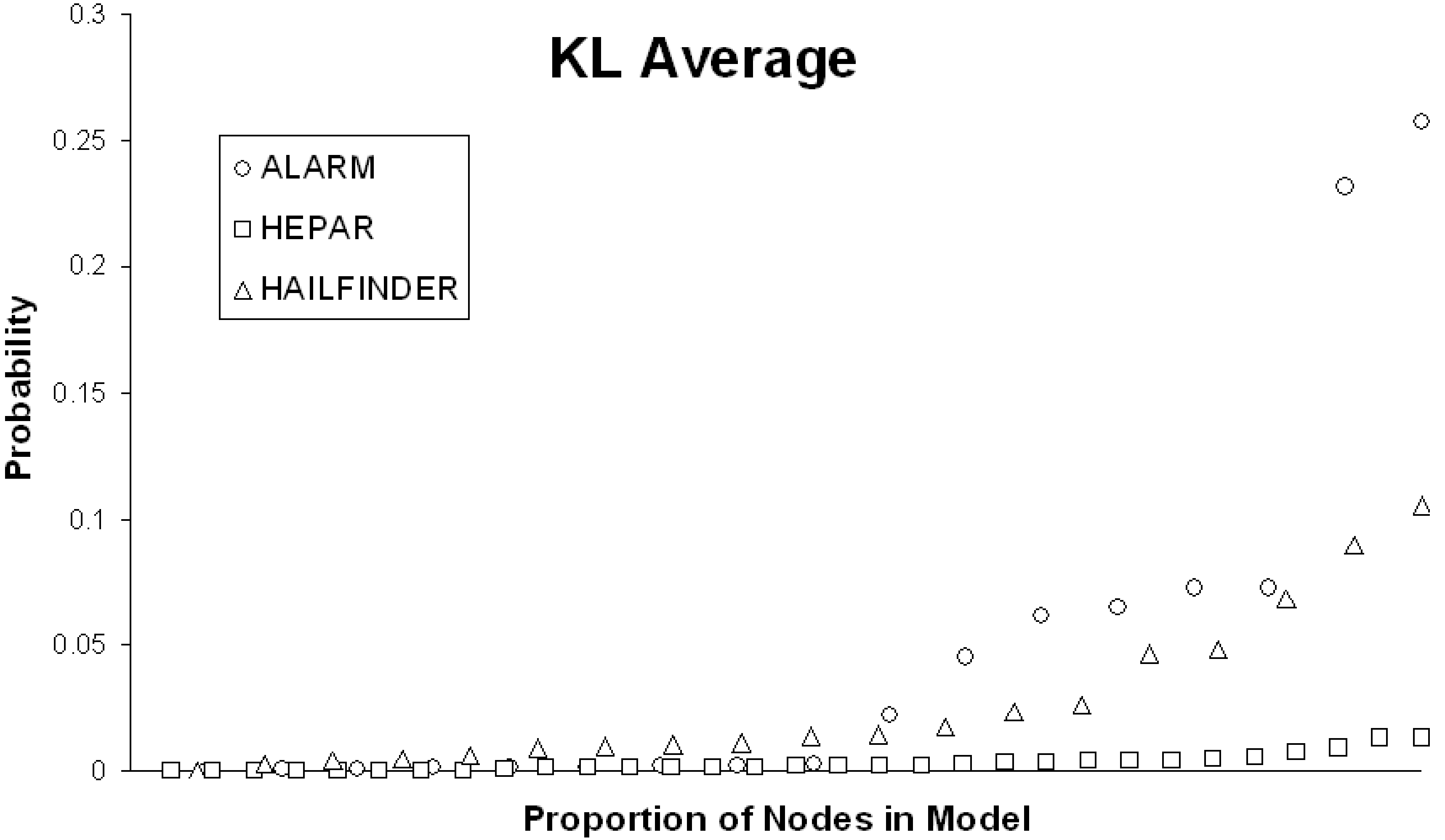
Canonical Gates in Practical Models

Noisy MAX in practical models



[Zagorecki & Druzdzel 2011]

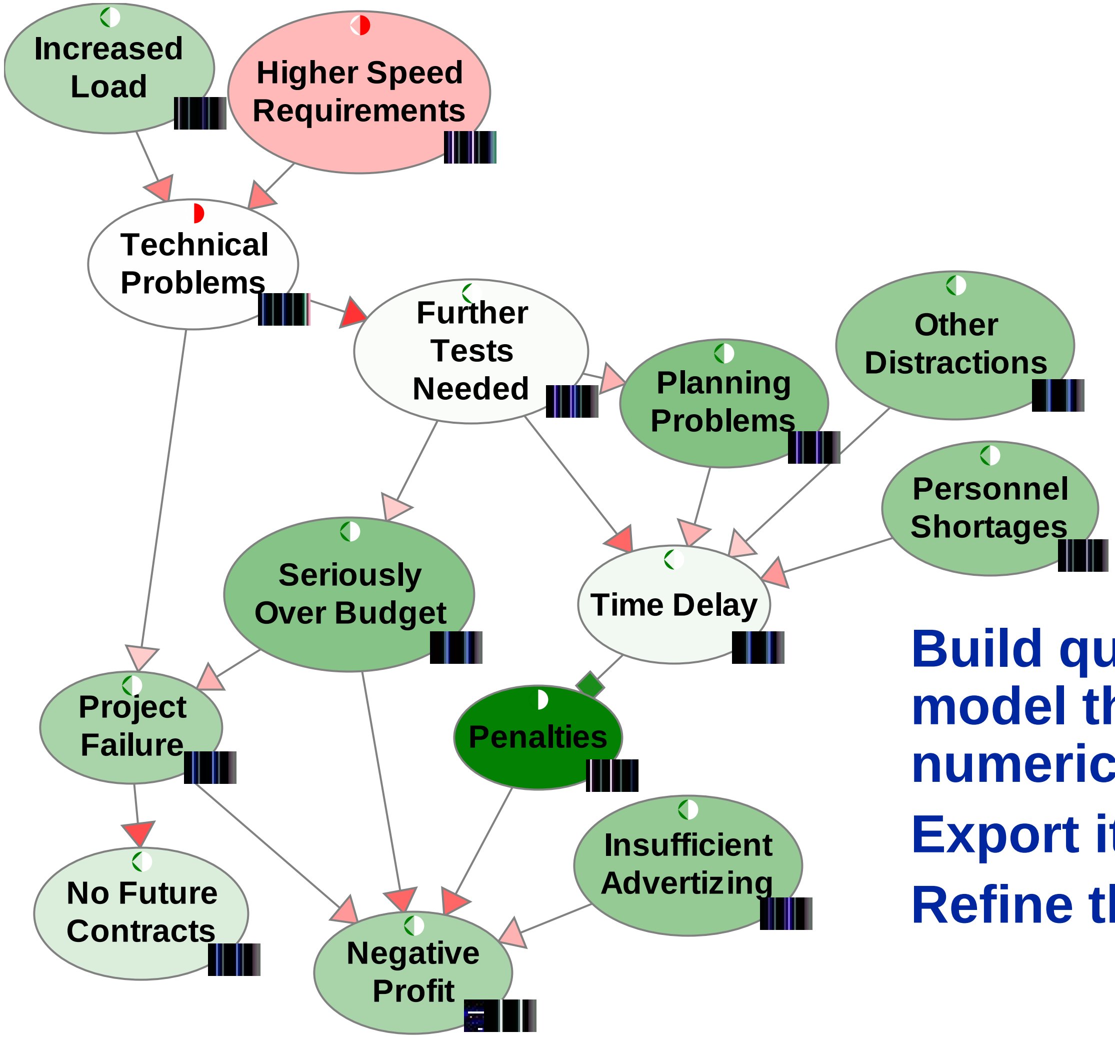
Noisy MAX in practical models



[Zagorecki & Druzdzel 2011]

DeMorgan Gate

Elicitation of probabilities: QGeNIe



Build quickly a QGeNIe model that requires n+m numerical parameters.

Export it to GeNIe.

Refine the full CPTs.

DeMorgan Gate: The Theory

The name comes from “De Morgan’s Laws”.

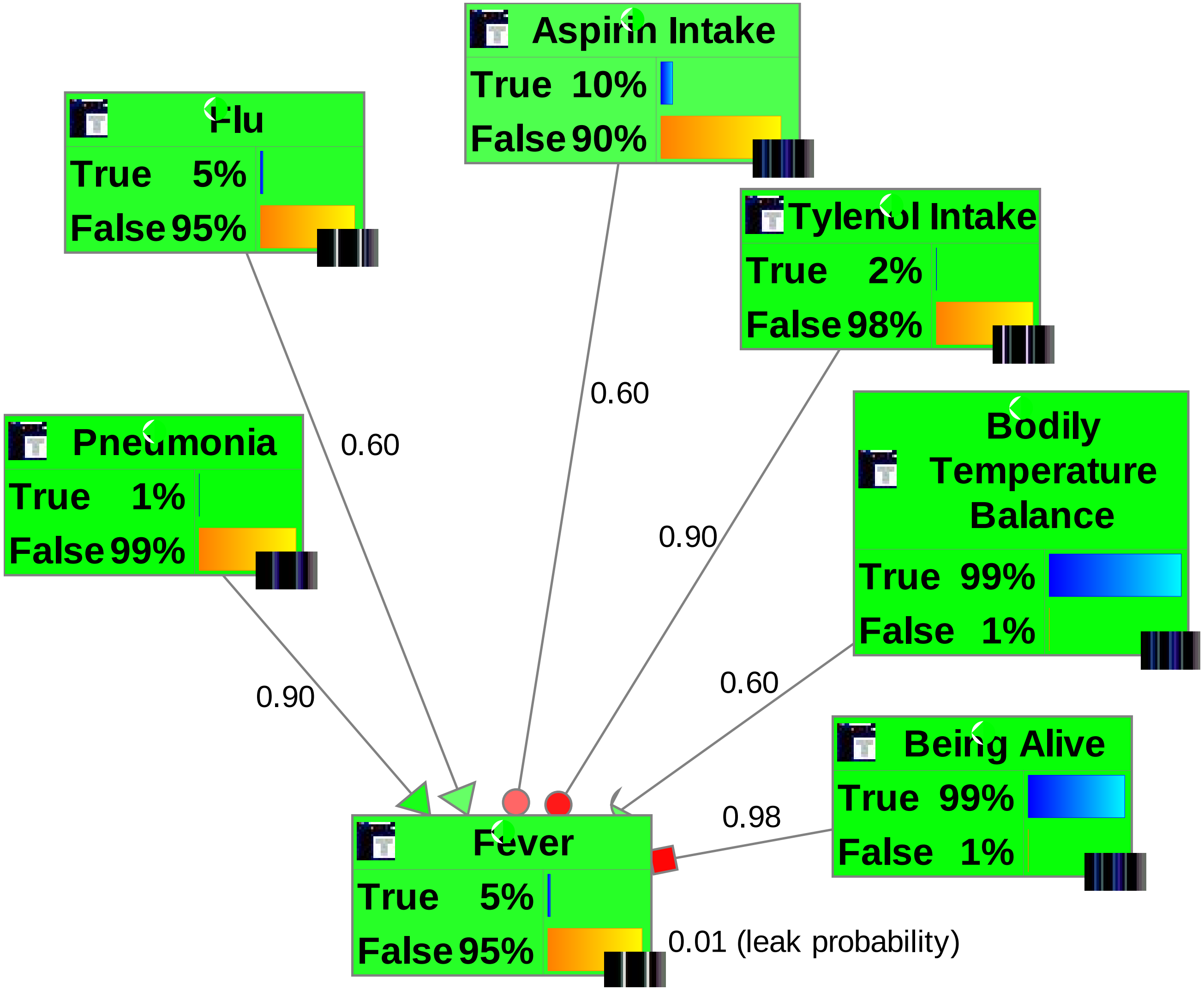
Four types of causes that interact with their effect through the following logical formula:

$$e = (c_1 \lor c_2 \lor \neg b_1 \lor \neg b_2) \land r_1 \land r_2 \land \neg i_1 \land \neg i_2 \;,$$

where:

- c_i**s stand for *Causes***
- b_i**s stand for *Barriers***
- r_i**s stand for *Requirements***
- i_i**s stand for *Inhibitors***

DeMorgan Gate: Example



Aspirin Intake
True 10%
False 90%

Flu
True 5%
False 95%

Tylenol Intake
True 2%
False 98%

Pneumonia
True 1%
False 99%

Bodily Temperature Balance
True 99%
False 1%

Being Alive
True 99%
False 1%

Fever
True 5%
False 95%

Concluding remarks

- In practical models, canonical gates are the only way to go
- There are significant computational advantages that stem from canonical gates