Centre National de la Recherche Scientifique

INSTITUT POLYTECHNIQUE DE PARIS

Computing with responsable electronics

Laurie E. Calvet

CNRS-Ecole Polytechnique

laurie.calvet@cnrs.fr

ICT sector in the context of climate change

Industry	Emissions 2022/2020 (million tCO_2e)			Change 2022/2020 %	Electricity (TWh)			Change 2022/2020 %
	2020	2021	2022		2020	2021	2022	22/20 %
Telecommunications operators	135	134	133	-1%	239	255	258	8%
Colocation data centers	36	40	43	20%	89	100	109	22%
Cloud & content	22	27	32	46%	54	70	85	63%
Subtotal	**193**	**201**	**208**	**8%**	**382**	**425**	**442**	**18%**
% of world	0.6%	0.6%	0.6%		1.60%	1.70%	1.70%	
ICT Equipment	**154**	**173**	**154**	**0.5%**	**282**	**329**	**311**	**10.6%**
- PCs	62	71	65	4.8%	110	133	124	
- Smartphones	60	64	57	-5.1%	116	131	119	**2.5%**
- Network	32	38	33	2.4%	56	65	69	**22.0%**
Product use	222	215	205	-7.5%	430	442	430	**-0.1%**
- PCs	203	197	187	-7.9%	394	405	392	**-0.5%**
- Smartphones	19	18	18	-3.4%	36	37	38	**4.3%**
Subtotal	**375**	**388**	**359**	-4.2%	**712**	**771**	**741**	**4.1%**
% of world	**1.2%**	**1.1%**	**1.0%**		**3.0%**	**3.0%**		
TOTAL	**568**	**589**	**567**	-0.2%	**1094**	**1196**	**1183**	**8.2%**
% of world	1.8%	1.7%	1.7%		4.6%	4.7%		

Certain electronics consume more energy when fabricated than when used.

Can we improve electronics to make it more sustainable?

From 'Measuring the Emissions & Energy Footprint of the ICT sector
World Bank, 2024

Outline

- Stochastic Computing from a historical perspective
  - Finite State Machine with ‘event inputs’
  - Fundamentals of stochastic computing
  - Stochastic computing for neural networks

- The BAYFLEX project
  - Overview
  - Recall of basic probability
  - Spiking neurons with organic electronics
  - Bayesian inference with OECTs

Timeline of Stochastic Computing

Centre National de la Recherche Scientifique

INSTITUT POLYTECHNIQUE DE PARIS

Hawai
ERC NANOINFER
(D. Querlioz)
BAYFLEX

1950s

Proposal: 1956
Von Neumann

1960s,1970s

Fundamental Ideas
Gaines 1969

1980s,1990s

Artificial neural networks 1980s-1990s, Brown & Card 2001

2000s

Factor Graphs
V. Gaudet 2000

2010s

Bayesian Inference (Droulez, Bessière, Laurent, 2010)
FET-OPEN BAMBI

LPICM

Finite State Machine

A *finite-state machine* is formally defined as a 5-tuple (Q, I, Z, q_0, W) such that:

- Q = finite set of states
- I = finite set of input symbols (alphabet)
- Z = finite set of output symbols
- q_0 = starting state
- W = mapping W of I x Q onto Z, called the state transition function

Very influenced by the McCulloch-Pitts paper on biological neurons, von Neumann explored what the impact of noisy hardware (noisy synapses) on computation, as defined from the logical calculus in McCulloch-Pitts

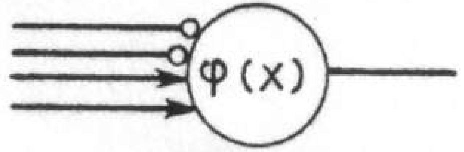


FIG. 1:

Automaton with time delay δ

INPUTS ARE BERNOULLI SEQUENCES

V. Neumann, "Probabilistic logics and the synthesis of reliable organisms from unreliable components," in *Automata Studies*, (Princeton Press), pp. 46–98, 1956.

McCulloch, W.S., Pitts, W. A logical calculus of the ideas immanent in nervous activity. Bulletin of Mathematical Biophysics 5, 115–133 (1943)

Information, Error Automatons

Using three basic operations (AND, OR, NOT) automatons can realize propositional calculus: every polynomial P can be realized using:

$$\sum_{i_1=\pm 1} \cdots \sum_{i_n=\pm 1} f_{i_1 \dots i_n} x_1^{i_1} \cdots x_n^{i_n},$$

- Information into the automaton can be represented by the occurrence of an event: appearance of head or tail of a coin toss, each element in has associated to it an error ε
- Given the maximum allowable error of the machine, can one construct the automaton?

➔ One cure for the error is to make multiple inputs.

ADVANTAGE 1: COMPUTATIONAL TECHNIQUE VERY RESILIENT TO HARDWARE ERRORS

V. Neumann, "Probabilistic logics and the synthesis of reliable organisms from unreliable components," in *Automata Studies*, (Princeton Press), pp. 46–98, 1956.

Fundamental Ideas

Stochastic Number: Given a probability p_x: $0 \leq p_x \leq 1$, the corresponding stochastic number *X* is a sequence of **random binary numbers** *X*0,*X*1, . . . for which any *Xj* ∈ {0, 1} may equal 1 with probability *pX*. **An ideal stochastic number** has the properties of a Bernoulli process, wherein the sequence bits are all statistically independent from each other.

1,0,1,1,0,1,1,0,1,1 = 0.7
0,1,1,1,1,1,1,0,0,0 = 0.6

1,-1,1,1,-1,1,1,-1, 1, 1 = 0.7
1, 1,1,1, 1,1,1,-1,-1,-1 = 0.6

Unipolar stochastic numbers: bitstreams of 0,1s : $p_x = \frac{x}{M}$

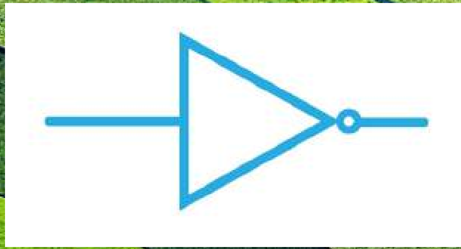
Bipolar stochastic numbers: bitstreams of -1, 1 and $p_x = 0.5\left(1 + \frac{x}{M}\right)$

'Tutorial on Stochastic Computing' C. Winstead in W. J. Gross and V. C. Gaudet (eds.), *Stochastic Computing: Techniques and Applications*, https://doi.org/10.1007/978-3-030-03730-7_3

Stochastic Arithmetic

Unipolar

1,0,1,1,0,1,1,0,1,1 = 0.7



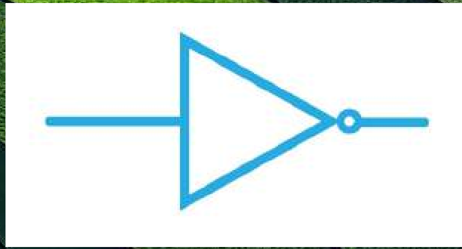
0,1,0,0,1,0,0,1,0,0 = 0.3

$p_x = \frac{x}{M}$

$p_z = 1 - \frac{x}{M}$

Bipolar

1,-1,1,1,-1,1,1,-1,1,1 = 0.7



-1,1,-1,-1,1,-1,-1,1,-1,-1 = 0.3

$p_x = 0.5\left(1 + \frac{x}{M}\right)$

$p_z = 1 - 0.5\left(1 + \frac{x}{M}\right)$

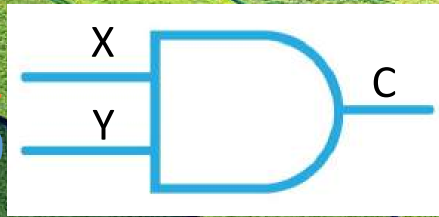
'Tutorial on Stochastic Computing' C. Winstead in W. J. Gross and V. C. Gaudet (eds.), *Stochastic Computing: Techniques and Applications*, https://doi.org/10.1007/978-3-030-03730-7_3

AND

Unipolar

0.7 = 1011011011

0.6 = 0111111000



0011011000 = 0.4 ≈ 0.42

$p_z = \frac{x}{M} + \frac{y}{M} \; if \; x = y$

$$p_z = p_x \times p_y$$

Bipolar

$$p_z = \frac{p_x \times p_y}{1 + p_x + p_y}$$

Need XNOR for Multiplication

ADVANTAGE 2: REDUCED HARDWARE IMPLEMENTATION COMPARED TO CONVENTIONAL COMPUTING

ADVANTAGE 3: TRADEOFF BETWEEN COMPUTATION TIME AND ACCURACY

‘Tutorial on Stochastic Computing’ C. Winstead in W. J. Gross and V. C. Gaudet (eds.), *Stochastic Computing: Techniques and Applications*, https://doi.org/10.1007/978-3-030-03730-7_3

Beware of correlations

1/3 = 0010 0110 0100

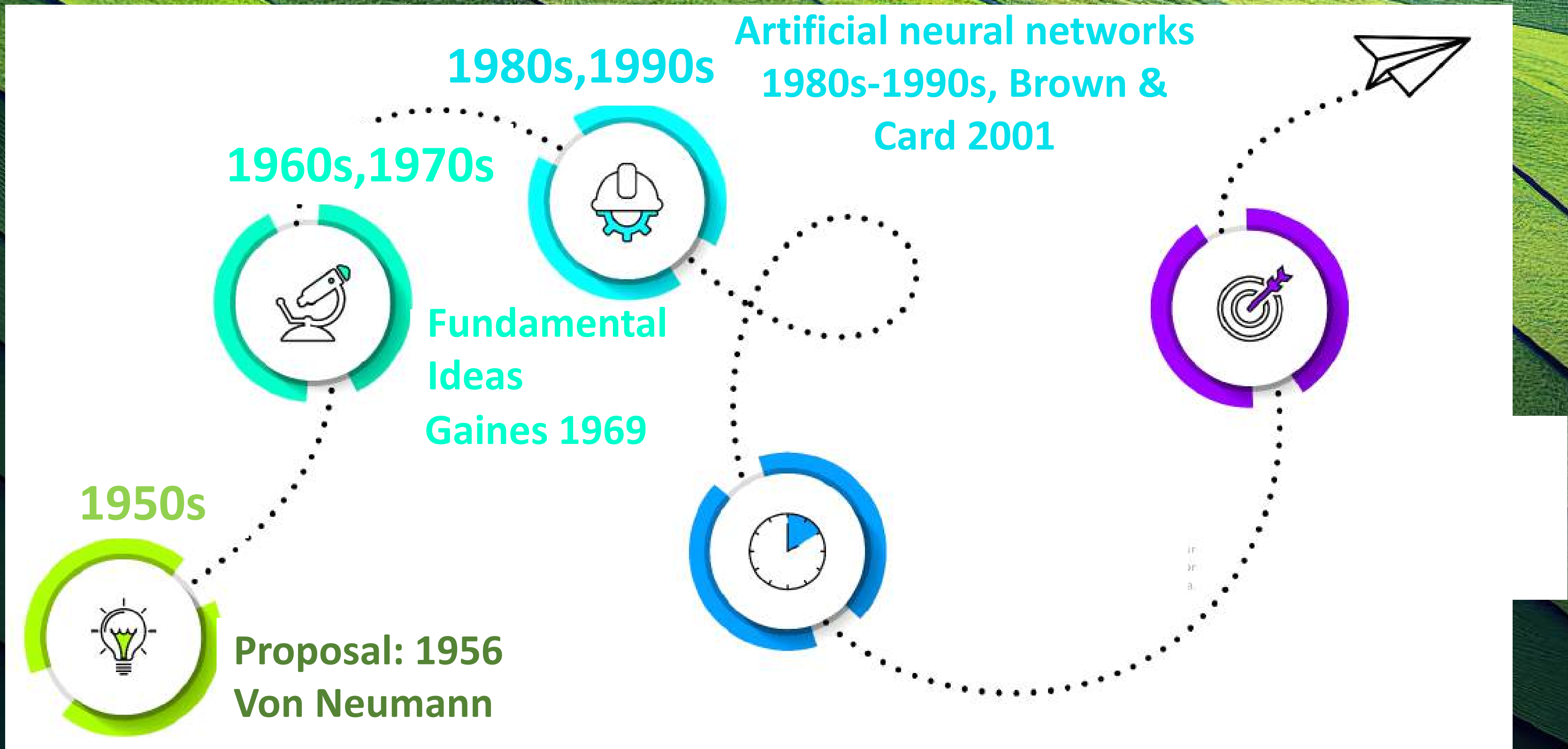
1/3 = 0010 0110 0100

0010 0110 0100 = 1/3

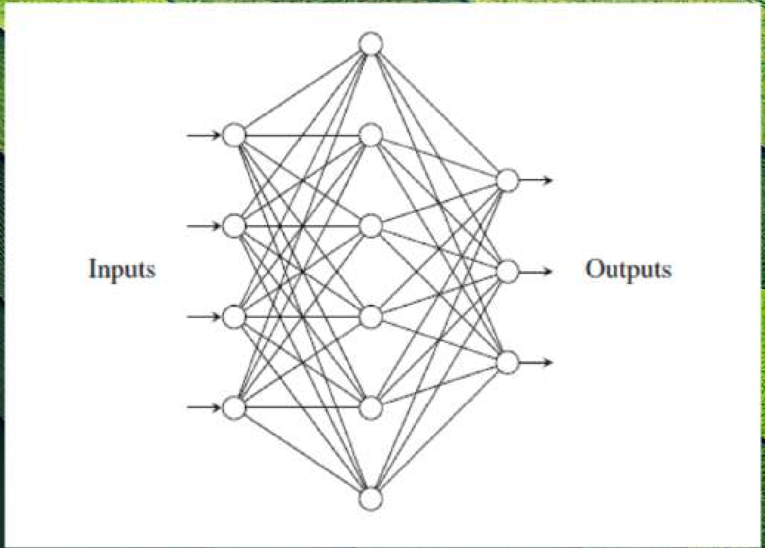
Bitstreams need to be uncorrelated. Generally done with random number generators

DISADVANTAGE: RANDOMIZATION OF BITSTREAMS

Timeline of Stochastic Computing



Neural networks (NN)



Typical feed-forward neural network (multilayer perceptron)

B. Brown, H. C. Card IEEE Trans on Comp 50 891 2001

Each 'neuron' implements a summation and an activation:

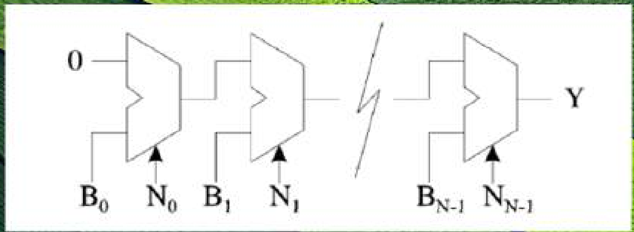
$$y = \sum_{j=0}^{K-1} w_j x_j$$

where w_j is a weight connecting in put j to output neuron y and x_j is the value arriving on the synapse

$$f_A(y) = \tanh(y)$$

$$f_A(y) = \begin{cases} x \text{ If } x > 0 \\ 0 \text{ Otherwise} \end{cases}$$ relu

Stochastic computing for NN



Digital to stochastic converter: Chain of weighted adders (2-input multiplexers) with the select line driven by a noise bit with probability 1/2

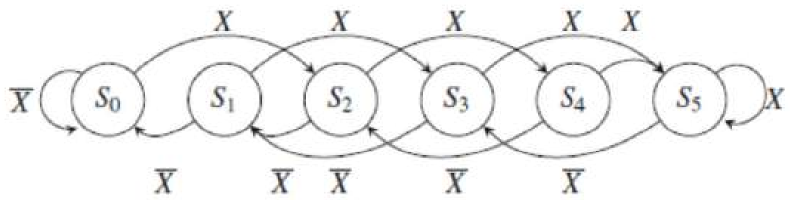
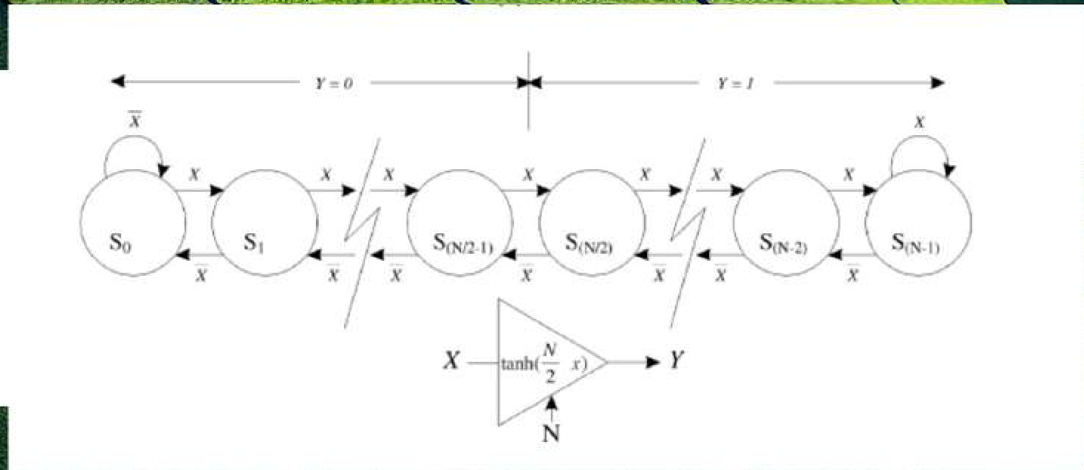


Fig. 44 Stochastic finite state machine design for the rectified linear unit function



B. Brown, H. C. Card IEEE Trans on Comp 50 891 2001

'Tutorial on Stochastic Computing' C. Winstead in W. J. Gross and V. C. Gaudet (eds.), *Stochastic Computing: Techniques and Applications*,

Timeline of Stochastic Computing

1950s

Proposal: 1956
Von Neumann

1960s,1970s

Fundamental
Ideas
Gaines 1969

1980s,1990s

Artificial neural networks
1980s-1990s, Brown &
Card 2001

2000s

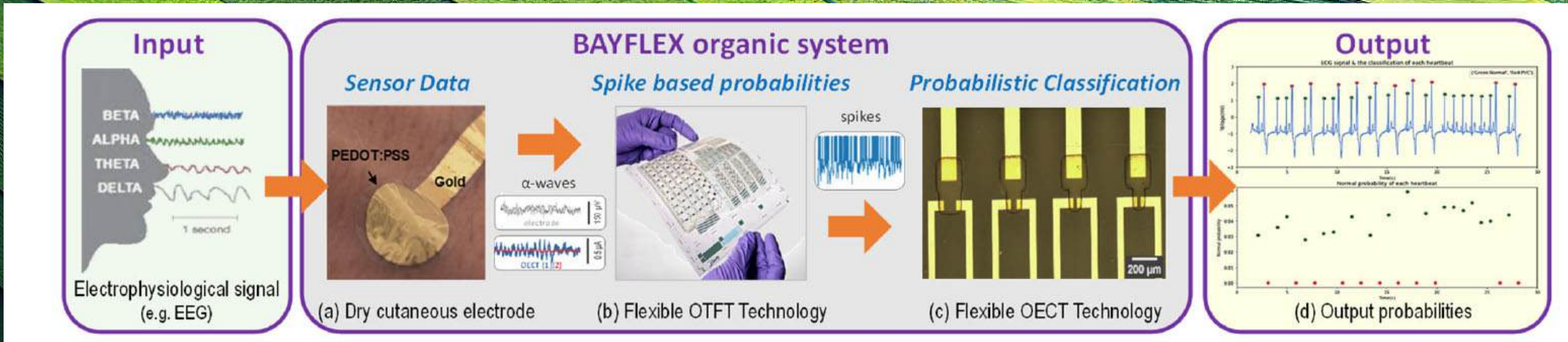
Factor Graphs
V. Gaudet 2000

2010s

Bayesian Inference
(Droulez, Bessière,
Laurent, 2010)
FET-OPEN BAMBI

Hawai
ERC NANOINFER
(D. Querlioz)
BAYFLEX

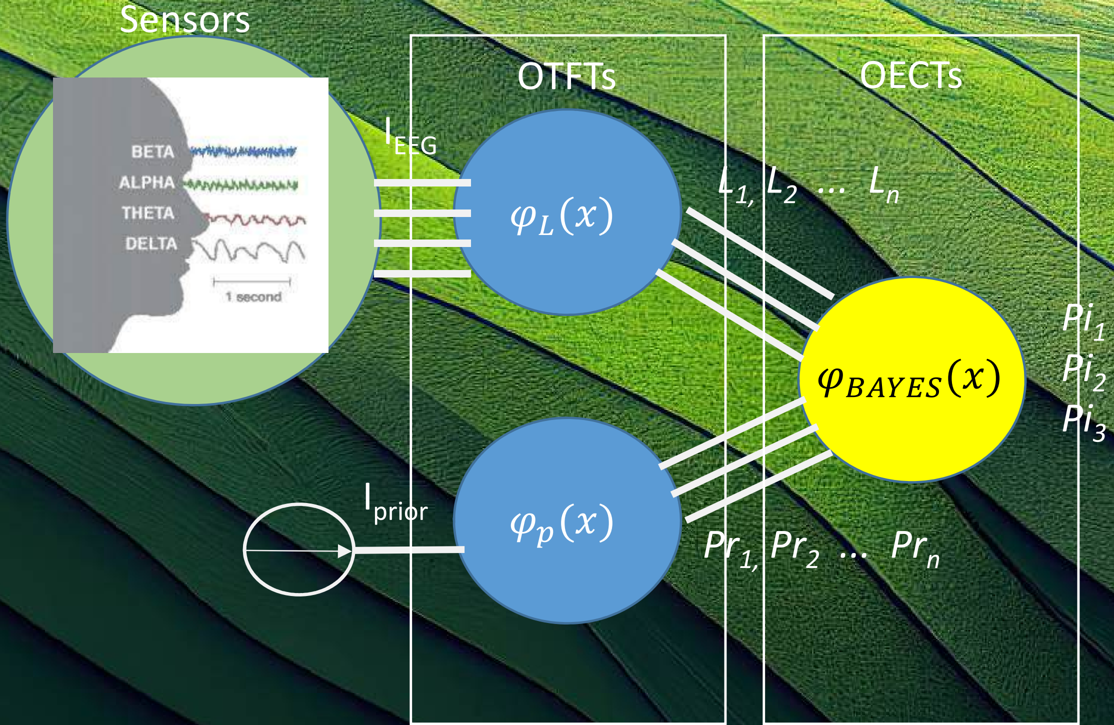
The BAYLFEX Project



Use fault-tolerant stochastic computing with all-organic technologies on flexible substrates to detect electrophysiological signals and classify them into categories.

VISION: Stochastic computing is a natural paradigm for organic electronics

BAYFLEX as a finite automaton



Bayesian inference in organic electronics

BAYFLEX Consortium

Centre National de la Recherche Scientifique

BAY FLEX

INSTITUT POLYTECHNIQUE DE PARIS

Model and simulate

L. Calvet (coordinator)
laurie.calvet@cnrs.fr
CNRS-Ecole Polytechnique
Palaiseau, France

B. Iniguez,
Universitat Rovira i Virgili
Tarragona, Spain

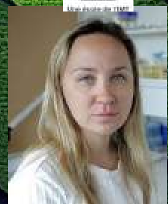
Benchmark

Bitbrain

L. Montesano
Bit & Brain
Technologies
Zaragoza, Spain

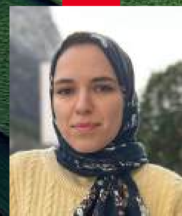
4 Academic partners
1 RTO (CEA)
1 SME (BB)

MINES Saint-Étienne



E. Ismailova,
Ecole des mines de
Saint-Etienne,
Gardanne, France

cea



L. Kadura
CEA-LITEN,
Grenoble, France

TECHNISCHE UNIVERSITÄT DRESDEN

H. Kleemann,
IAPP, TUD
Dresden, Germany

Fabricate

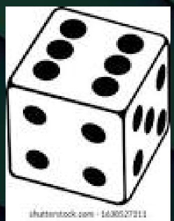
Bayesian inference

- Statistical Method for determining the probability of an event based on prior knowledge
- Based on the use of Bayes' law and conditional probabilities
- In algorithms:
  - Very hard to solve exactly
  - Typically use Monte-Carlo techniques or sampling (Gibbs, Metropolis-Hastings)
- Applications
  - Robotics
  - Spam detection
  - Bioinformatics/healthcare applications
  - Ecology
  - Finance

BAYFLEX: Continuos Health Monitoring

Introduction to probability

- Define a sample space: $\Omega = (w_1, w_2, \ldots, w_n)$
- Define measurable events S that will be defined by probabilities
- Define a probability distribution $P(S, \Omega)$ that maps events S and to real values s.t. $P(\Omega) = 1$, for events $\alpha, \beta \in S$ and $\alpha \cup \beta = \emptyset$ then $P(\alpha \cup \beta)$=P(α) $+ P(\beta)$

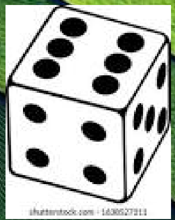


$$\Omega = (1,2,3,4,5,6)$$

$P(S, \Omega)$=(1/6,1/6, 1/6, 1/6, 1/6, 1/6)

P(4,5) = P(4) + P(5) = 1/3 **Joint probability distribution**

Conditional Probability



P(5,6) = P(5) + P(6) = 1/3 **Joint probability**

What is the probability that we rolled a 6 (event F) given that we know we rolled a number over 4 (event E)?

P(F|E)=? **Conditional probability**

$$P(F|E) = \frac{P(F \cap E)}{P(E)} = \frac{1/6}{1/3} = 1/2$$

Note that we must have : $\sum_i P(F_i|E) = 1$

Bayes'rule

Given a conditional probability $$P(F|E) = \frac{P(F \cap E)}{P(E)}$$

We notice that when we write: $$P(E|F) = \frac{P(E \cap F)}{P(F)}$$

It must be that: $$P(E \cap F) = P(F \cap E)$$

Which leads to:

$$P(F|E) = \frac{P(E|F)P(F)}{P(E)}$$

Bayes'rule

Other ways to write Bayes' rule

Assume there is a set of hypotheses H_1, H_2,..., H_n forming a sample space Ω. An event E occurs that tells us hypothesis i is correct.

$$P(H_i|E) = \frac{P(E|H_i)P(H_i)}{P(E)}$$

Recalling that : $\sum_i P(H_i|E) = 1$ **we have** $\sum_i P(H_i|E) = \frac{\sum_i P(E|H_i)P(H_i)}{P(E)} = 1$

Or $P(E) = \sum_i P(E|H_i)P(H_i)$

$$P(H_i|E) = \frac{P(E|H_i)P(H_i)}{\sum_i P(E|H_i)P(H_i)}$$

$$P(H_i|E) = \frac{P(E|H_i)P(H_i)}{P(E|H_i)P(H_i) + (P(\overline{H_i})(P(E|\overline{H_i}))}$$

Muller C-element for Bayes' law

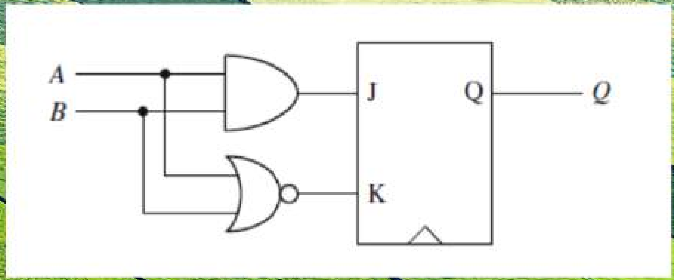
$$P(Z) = \frac{P(X)P(Y)}{P(X)P(Y) + \big(1 - P(X)\big)\big(1 - P(Y)\big)}$$

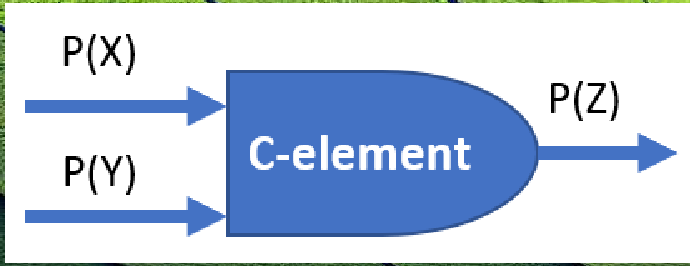
BAYES' RULE $$P(H_i|E) = \frac{P(E|H_i)P(H_i)}{P(E|H_i)P(H_i) + (P(\overline{H_i})(P(E|\overline{H_i}))}$$

Define: $$\boldsymbol{P(E^*) = \frac{P(E|H_i)}{P(E|H_i) + P(E|\overline{H_i})}}$$

$$P(H|E1) = \frac{P^*(E)P(H)}{P^*(E)P(H) + (1 - P^*(E))(1 - P(H))}$$

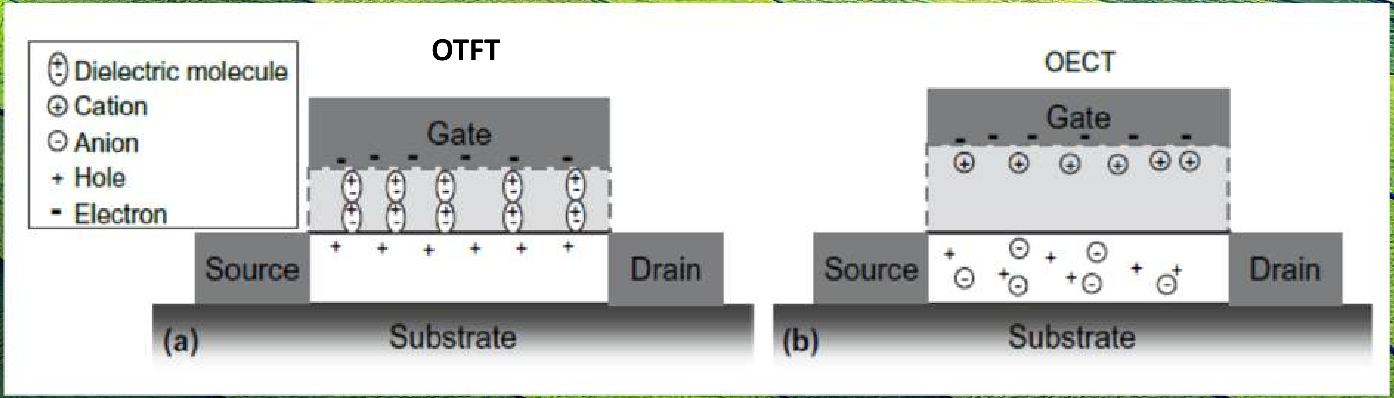
J.S. Friedman IEEE Trans CAS I **63** 895 (2016)

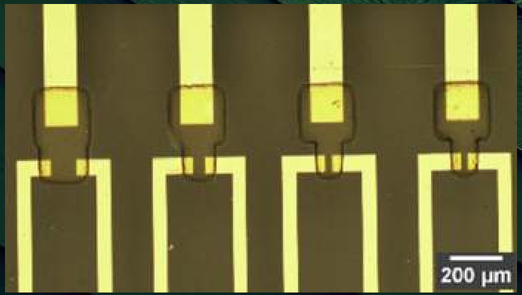




A	B	C
0	0	0
0	1	C_{prev}
1	0	C_{prev}
1	1	1

Organic Electrochemical Transistors (OECTs)

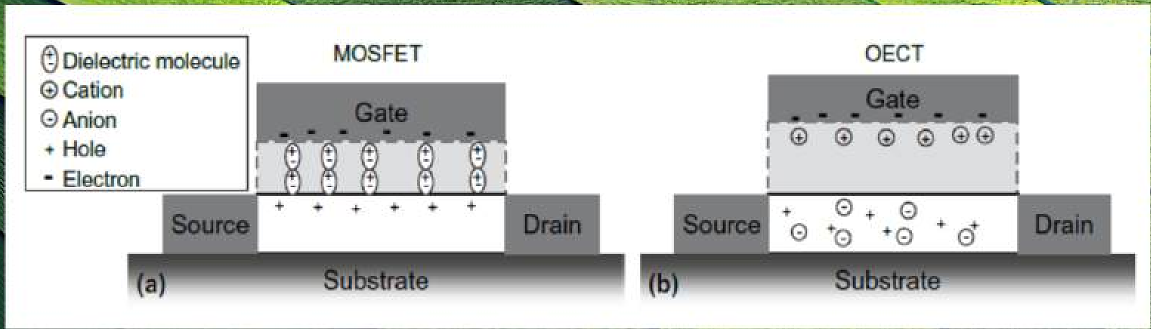




First OECT arrays for testing (TUD)

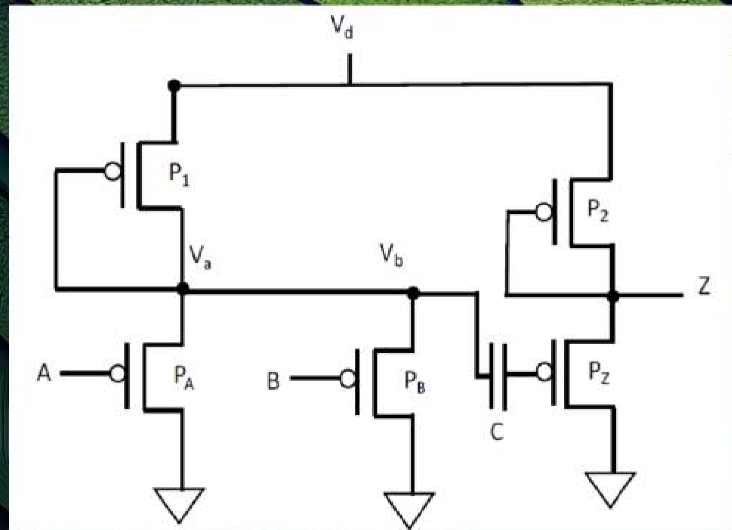
OTFT	OECT
Gate action: Dielectric	Gate action: electrolyte
Charge accumulated oxide interface	Charge accumulated throughout channel
Immobile charges, double later	Mobile ionic charges

Dynamic Muller C-element

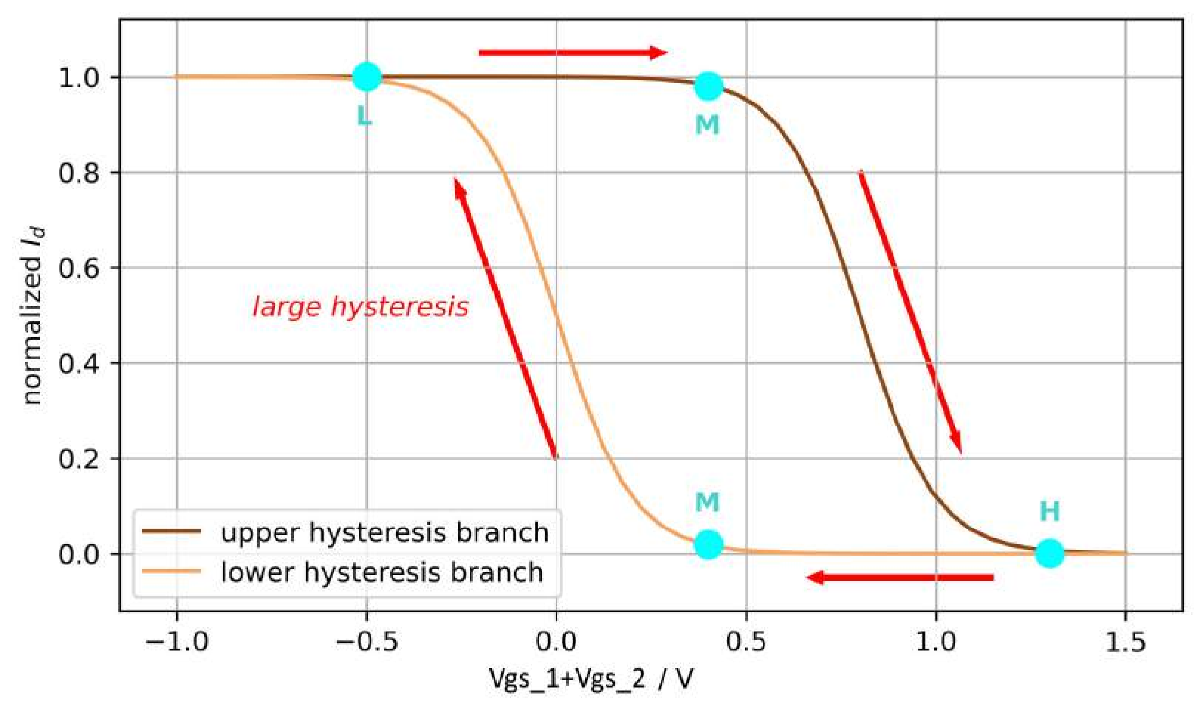


A	B	C
0	0	0
0	1	C_{prev}
1	0	C_{prev}
1	1	1

Large capacitance of the electrolyte lends itself to realization of a compact Muller c-element



Müller C element with OECT

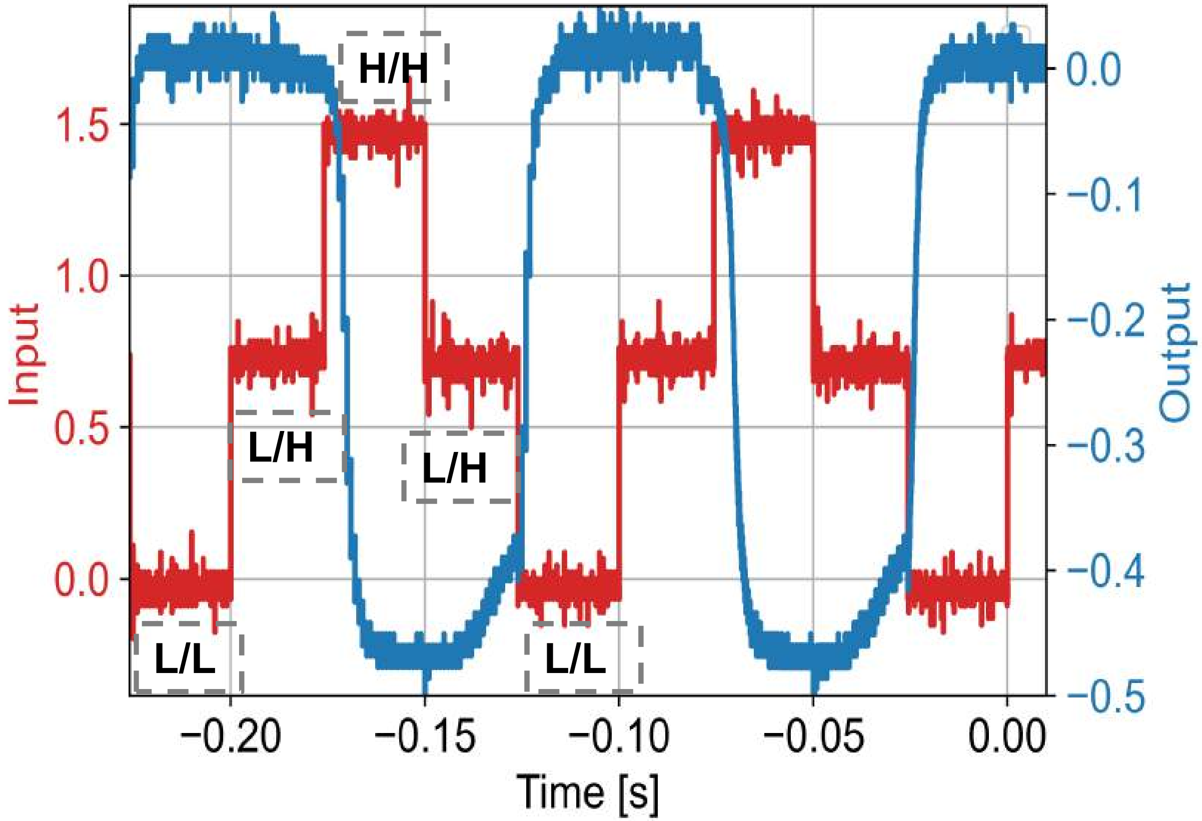


- OECT has hysteresis which can play role as a memory

From Yeohoon Yoon, TUD

P-type OECT

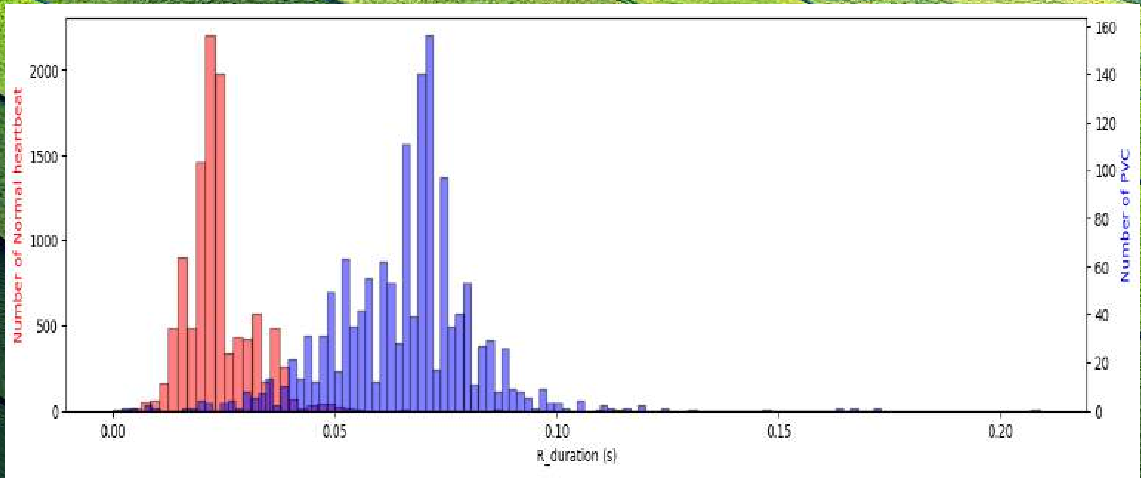
A	B	C
0	0	0
0	1	C_{prev}
1	0	C_{prev}
1	1	1



t(s)

From Yeohoon Yoon, TUD

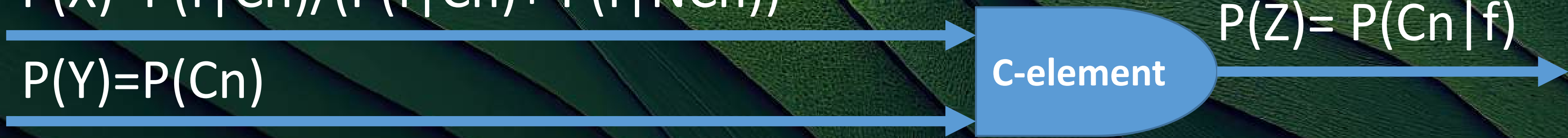
PVC Classification Task



Classification of normal versus PVC heartbeat

$P(X)=P(f|Cn)/(P(f|Cn)+ P(f|NCn))$

$P(Y)=P(Cn)$

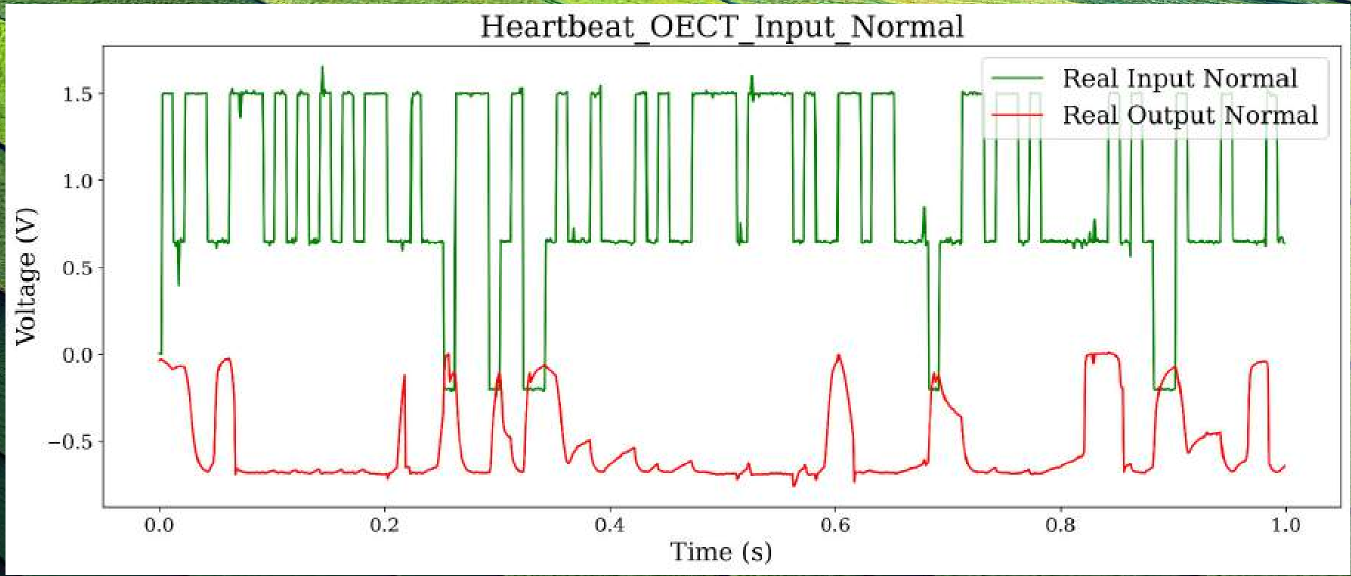


$P(Z)= P(Cn|f)$

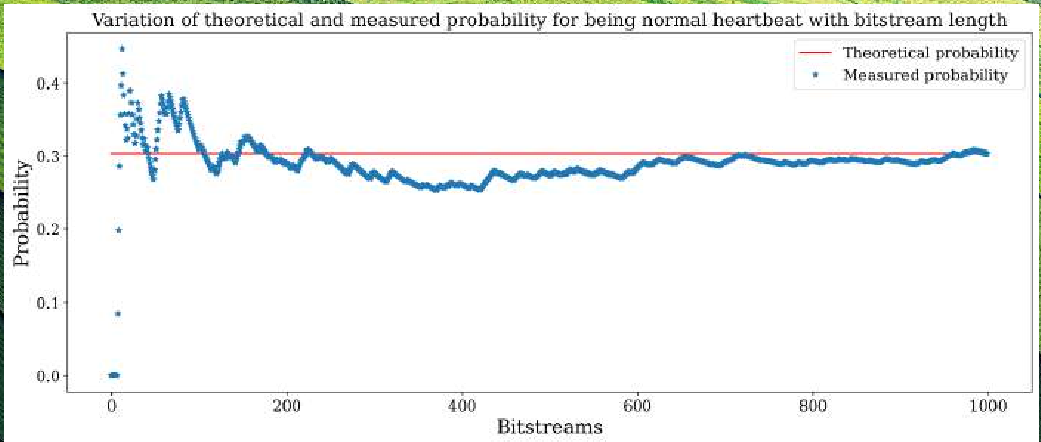
Since the difference between the two distributions is significant, for a large number of heartbeats, P(X) tends to 0 or 1

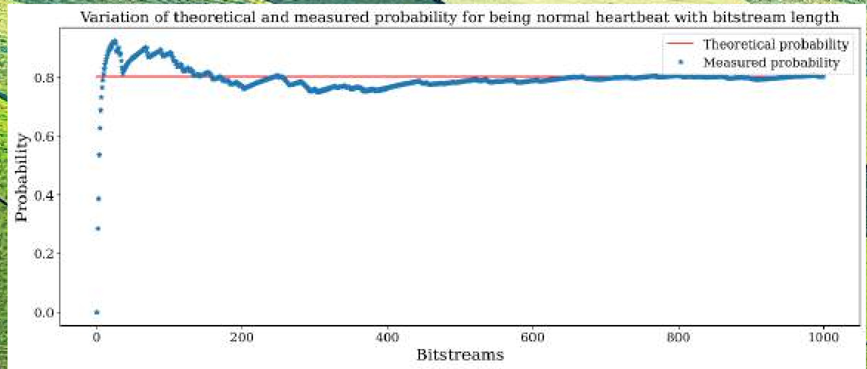
Example measurements

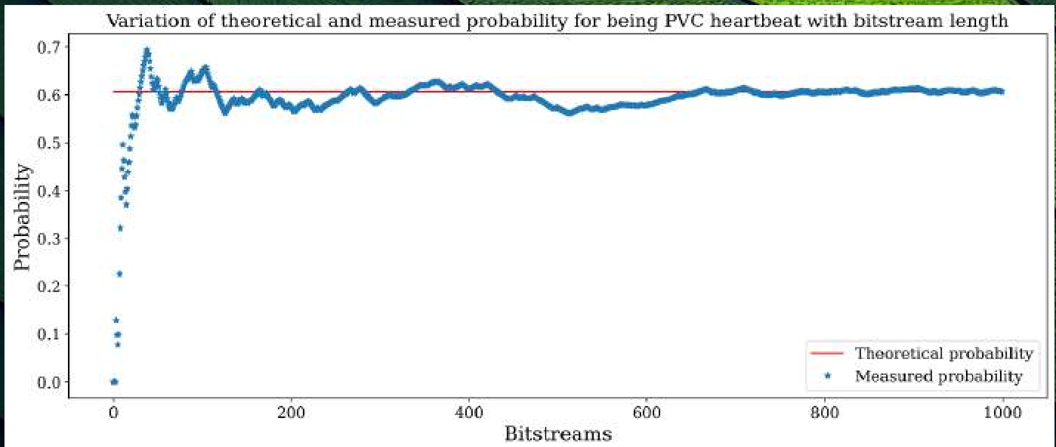
#312

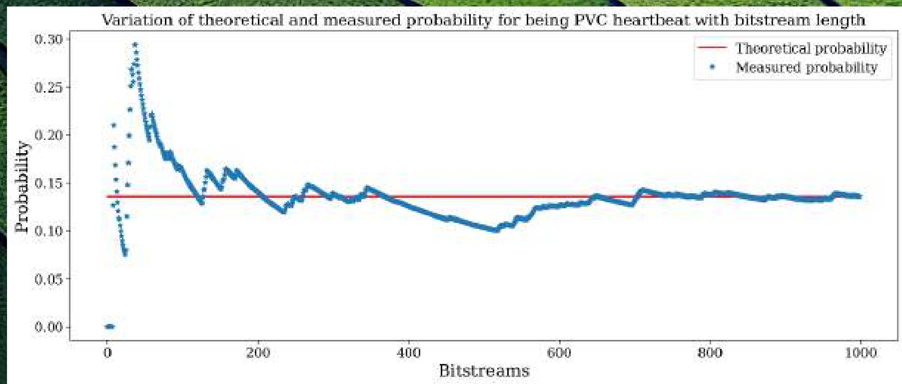


Variation of probabilities





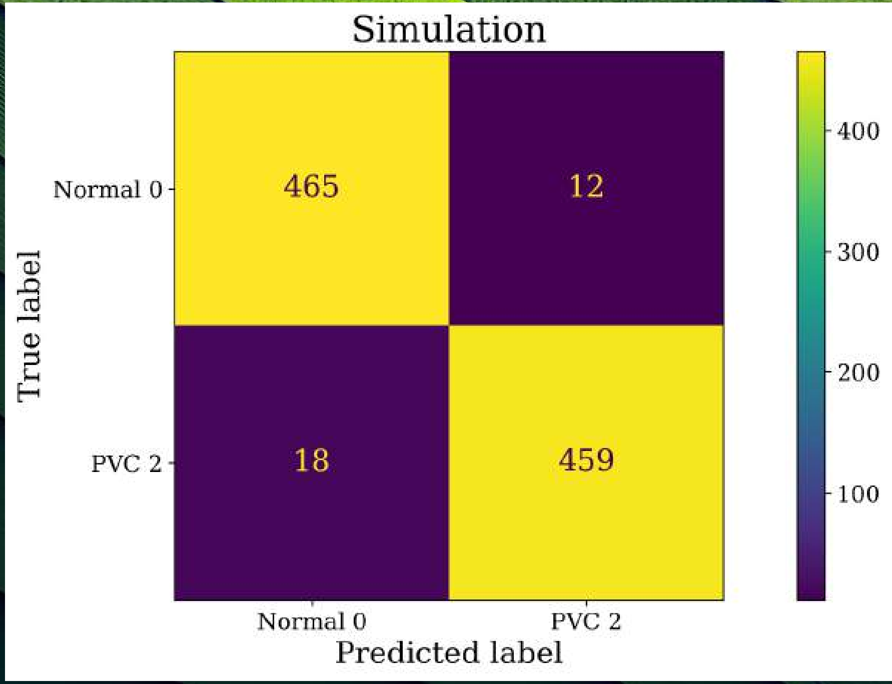




Classification Results

Vin_high_state (1)	Vin_low_state (0)
1.5V	-0.2V

100 bits per heartbeat_Per bit lasts 0.01s_954 heartbeats



Conclusions

Organic technologies are a natural paradigm for stochastic computing

Demonstration of Muller c-element implementing Bayes' rule in OECTs

Acknowledgements

BAYOEN

Sami Nakouzi ***Zonglong Li*** ***Yerin Kim***

Ie Mei Bhattacharyya

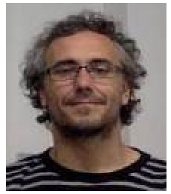
Yvan Bonnassieux

Djamal Zilal (M2 2023)
Patryk Golec (Visiting PhD 2023)

LPICM, CNRS-Ecole Polytechnique, Palaiseau, France

BAYFLEX Consortium

B. Iniguez,
Universitat Rovira i Virgili
Tarragona, Spain



L. Montesano
Bit & Brain
Technologies
Zaragoza, Spain

H. Kleemann,
IAPP, TUD
Dresden, Germany

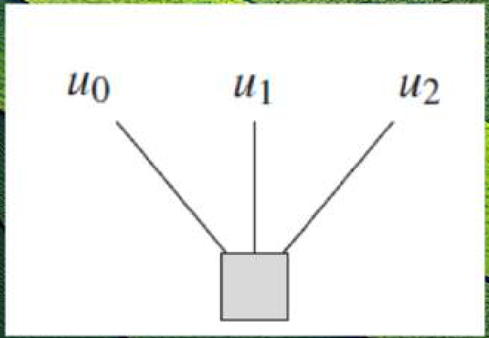
E. Ismailova,
Ecole des mines de
Saint-Etienne,
Gardanne, France

L. Kadura
CEA-LITEN,
Grenoble, France

laurie.calvet@cnrs.fr

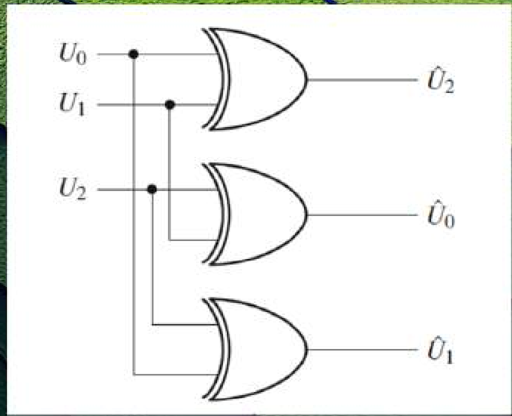
Belief propagation

Parity check



Take two bits u_0 and u_1: To each one we add a parity bit such that $u_0 \oplus u_1 = u_2$ The resulting truth table, shown to the right, is an XOR. If we are only able to retrieve two of the bits, we can correct the 3rd bit.

u_0	u_1	u_2
0	0	0
0	1	1
1	0	1
1	1	0



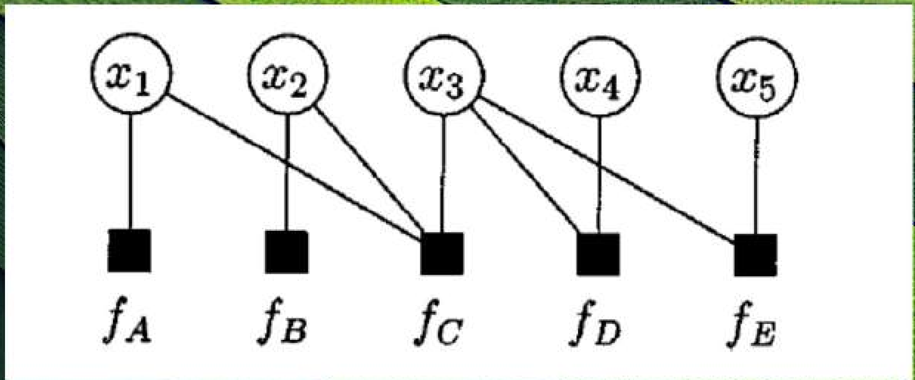
Now, consider the bits as probabilities $p_0, p_{1,} p_2$ corresponding to the probability that u_0, u_1, u_2 are 1. An estimate of the extrinsic probability that one bit can be obtained based on the other two since:

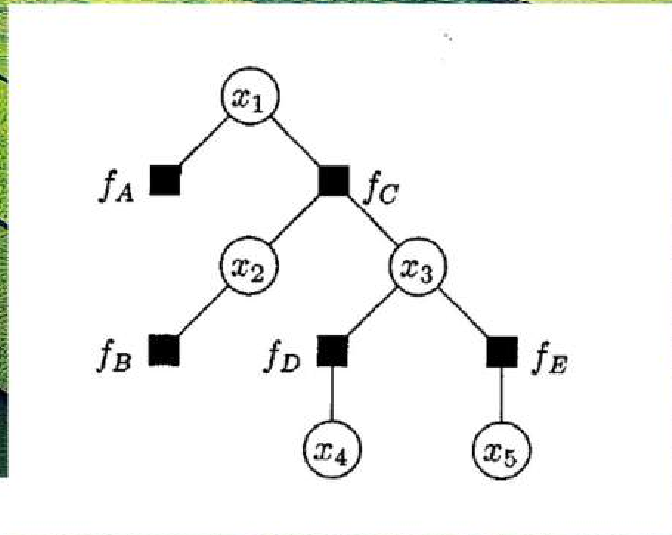
$$p_{1\,|\,0,2} = p_0\,(1 - p_2) + (1 - p_0)\,p_2$$

➔ This is known as belief propagation

V. Gaudet and A. Rapley, “*Electron. Lett.*, vol. 39, no. 3, pp. 299–301, 2003

Factor graphs for the sum-product algorithm





$$g(x_1, x_2, x_3, x_4, x_5)$$
$$= f_A(x_1) f_B(x_2) f_C(x_1, x_2, x_3) f_D(x_3, x_4) f_E(x_3, x_5)$$

The 'Factor' is essentially computing the marginal probability of the data

Kschischang IEEE Trans on Information Theory **47** 2001

From Factor graphs to stochastic computing

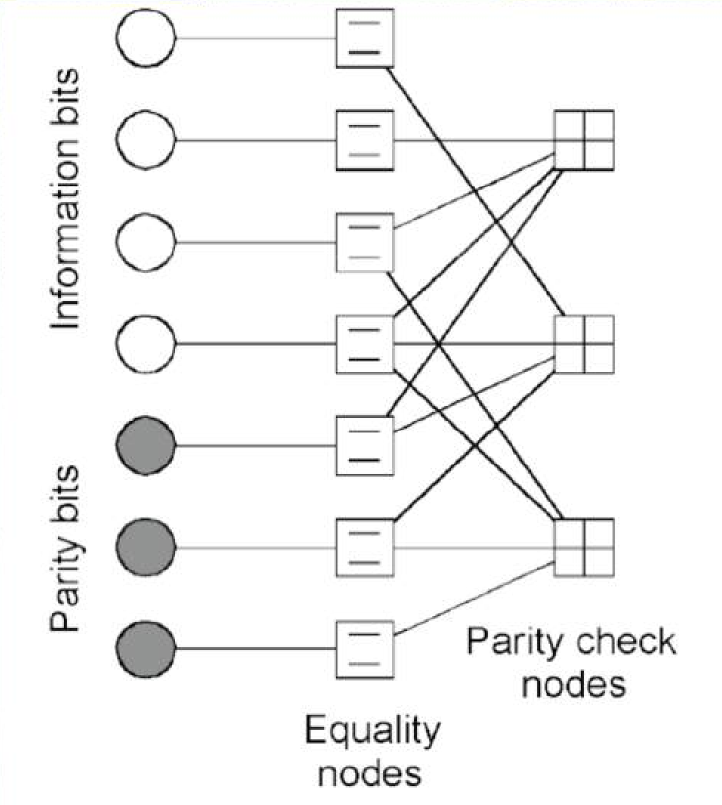


Figure 4: Hamming Code Factor Graph Constructed Using Equality and Parity-Check Nodes

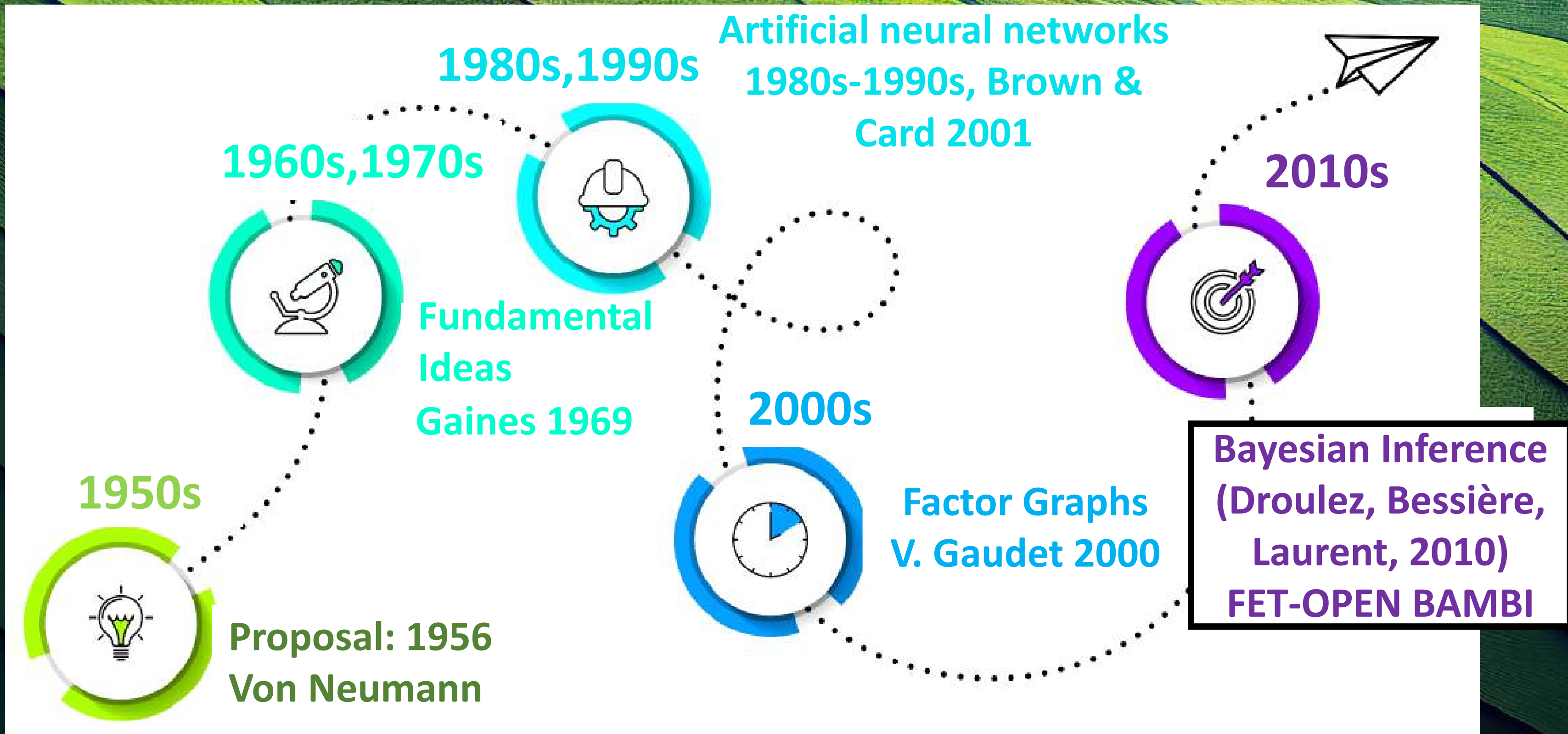
- Implementation of factor graphs can be realized in stochastic computing
- Two types of gates are needed: Parity check gates and Equality nodes

➔ Equality node is a c-element

$$P(C) = P(A)P(B)/[(P(A)P(B) - (1 - P(A))(1 - P(B))$$

V. Gaudet and A. Rapley, "*Electron. Lett.*, vol. 39, no. 3, pp. 299–301, 2003.terative decoding using stochastic computation,"

Timeline of Stochastic Computing



A. Coninx et al "Bayesian sensor fusion with fast and low power stochastic circuits," in *2016 IEEE ICRC*, 2016.