Differentiable programming for the frontiers of computation

methods and new perspectives

2nd COMCHA Workshop, A Coruña, Spain

Dr. Pietro Vischia
pietro.vischia@cern.ch
vischia.github.io

If you are reading this as a web page: have fun! If you are reading this as a PDF: please visit

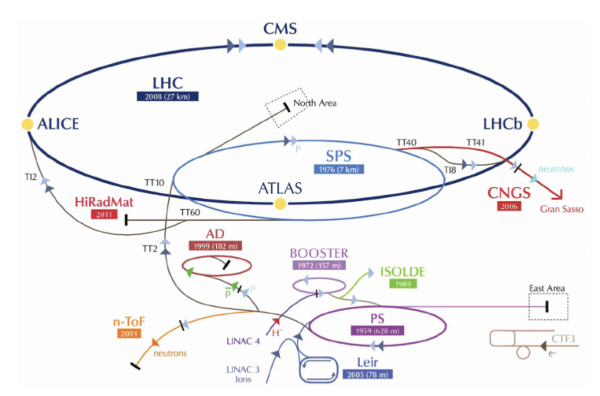
https://www.hep.uniovi.es/vischia/persistent/2024-10-04_DifferentiableProgrammingForTheFrontiersOfComputationAt2ndCOMCHAWorkshop_vischia.html

to get the version with working animations

Complex experimental apparata

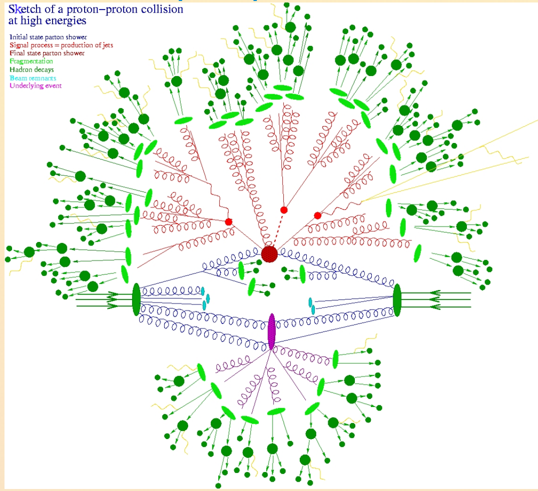
- 2020 European Strategy (EUSUPP): "New large, long-term projects, pushing technological skills to the limit"

Complex accelerators

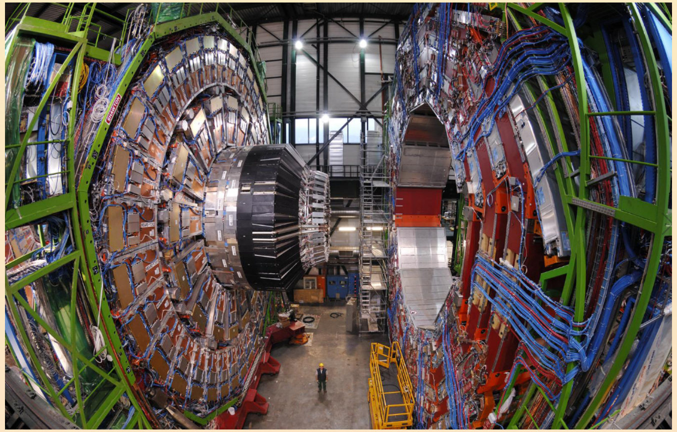


Existing CERN accelerator complex with Large Hadron Collider (LHC), Super Proton Synchrotron (SPS), Proton Synchrotron (PS), Antiproton Decelerator (AD), Low Energy Ion Ring (LEIR), Linear Accelerators (LINAC), CLIC Test Facility (CTF3), CERN to Gran Sasso (CNGS), Isotopes Separation on Line (ISOLDE), and neutrons Time of Flight (n-ToF).

Complex phenomena



Complex experiments and reconstruction

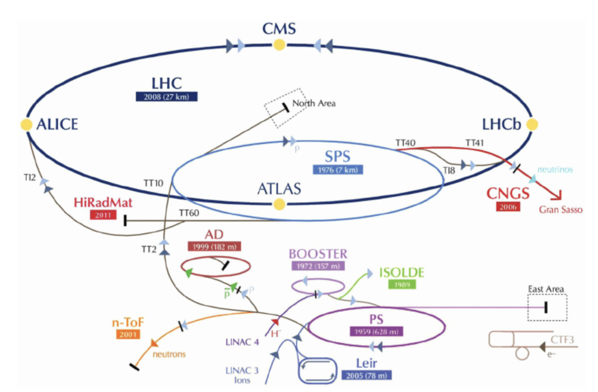


—
Image from 10.1103/PhysRevSTAB.16.054801 and from the CMS Collaboration

Complex experimental apparata

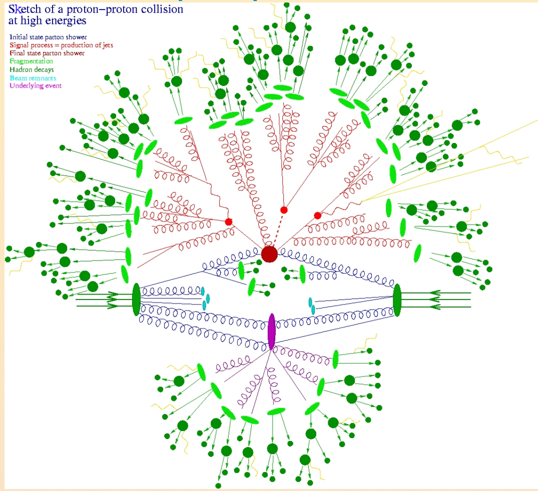
- 2020 European Strategy (EUSUPP): "New large, long-term projects, pushing technological skills to the limit"

Complex accelerators

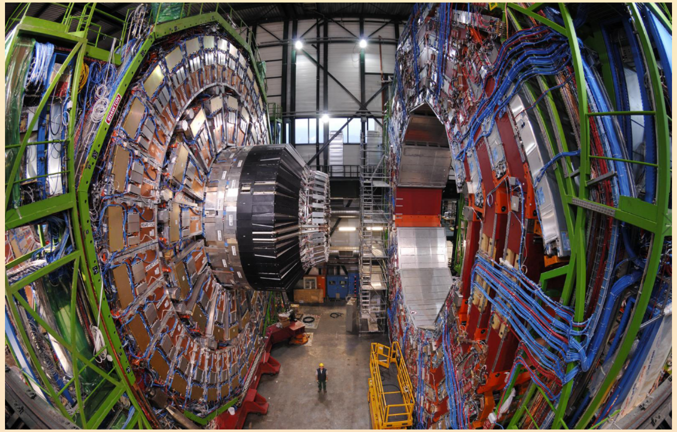


Existing CERN accelerator complex with Large Hadron Collider (LHC), Super Proton Synchrotron (SPS), Proton Synchrotron (PS), Antiproton Decelerator (AD), Low Energy Ion Ring (LEIR), Linear Accelerators (LINAC), CLIC Test Facility (CTF3), CERN to Gran Sasso (CNGS), Isotopes Separation on Line (ISOLDE), and neutrons Time of Flight (n-ToF).

Complex phenomena



Complex experiments and reconstruction

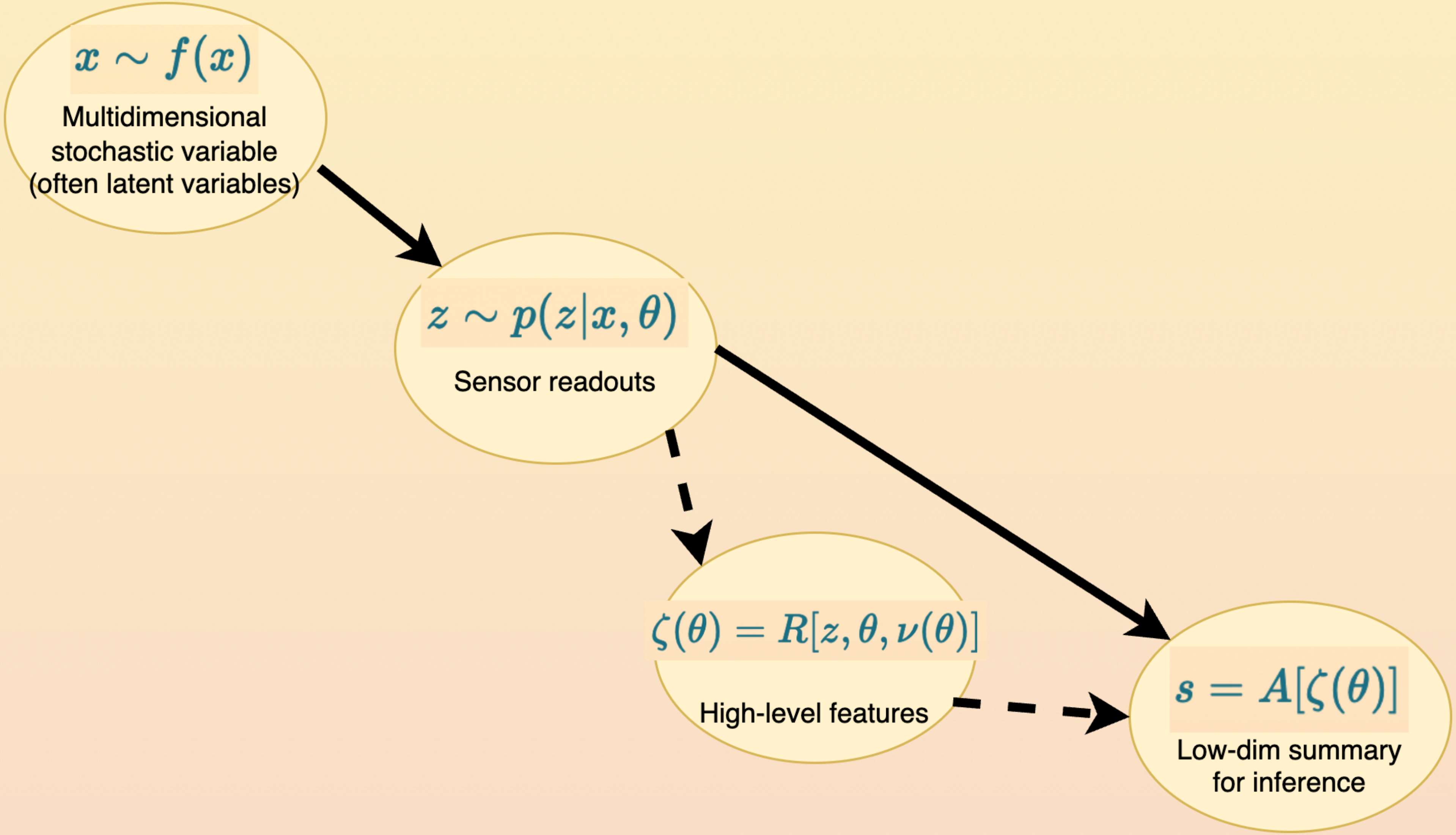


$$P(\mathbf{x}|\alpha) = \underbrace{\boxed{\frac{1}{A_\alpha \sigma_\alpha}}}_{\text{Normalization factor}} \int \underbrace{\boxed{d\Phi(y) \frac{dx_1 dx_2}{x_1 x_2 s} f(x_1) f(x_2)}}_{\text{We collide protons and that is a mess}} \underbrace{\boxed{|\mathcal{M}_\alpha(y, x_1, x_2)|^2}}_{\text{Linear operator in Hamiltonian formalism, describes the physics process}} \underbrace{\boxed{W(\mathbf{x}|y)\, \epsilon_\alpha(y)}}_{\text{Detector response, experimental efficiencies}}$$

- Stochastic processes $\longrightarrow$ intractable likelihood (matrix element, parton shower, detector simulation... result in latent variables)
- Costly MonteCarlo simulators to generate $x \sim p(x|\theta)$ (for each event, several thousand randomized choices)

Image from 10.1103/PhysRevSTAB.16.054801 and from the CMS Collaboration

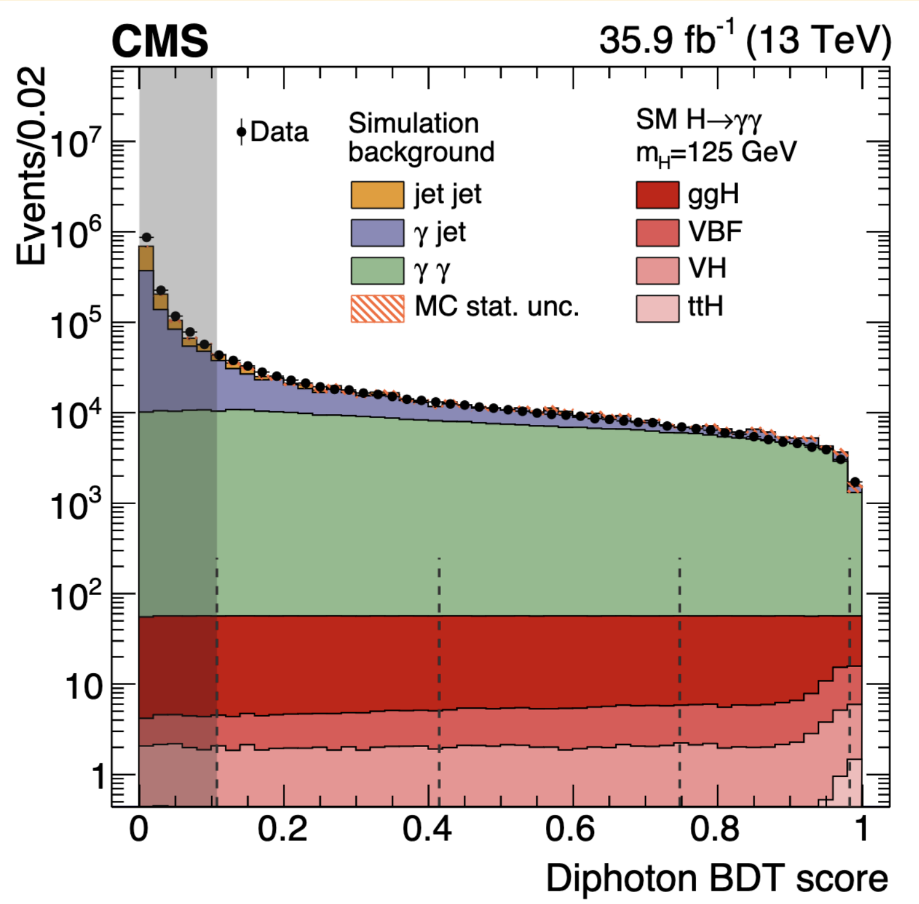
Typical analysis pipeline

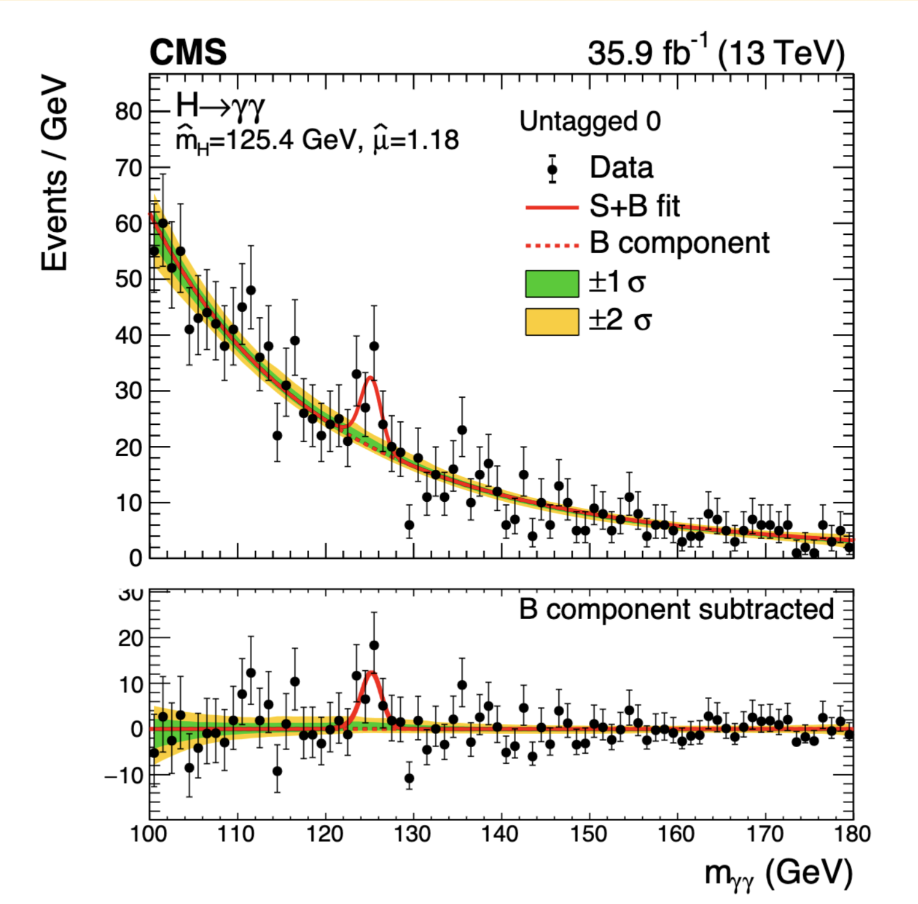


—
Formulas from our white paper

Can we go beyond histograms?

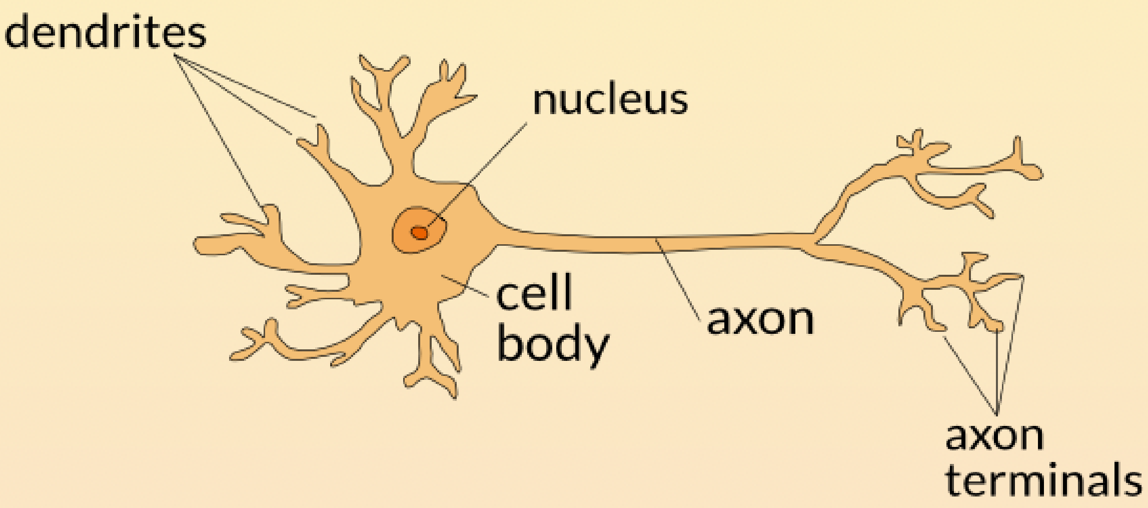
- Histograms are likelihood-free (count events, assume Poisson per bin, global likelihood as product)
- Can we optimize inference procedures through intractable problems?



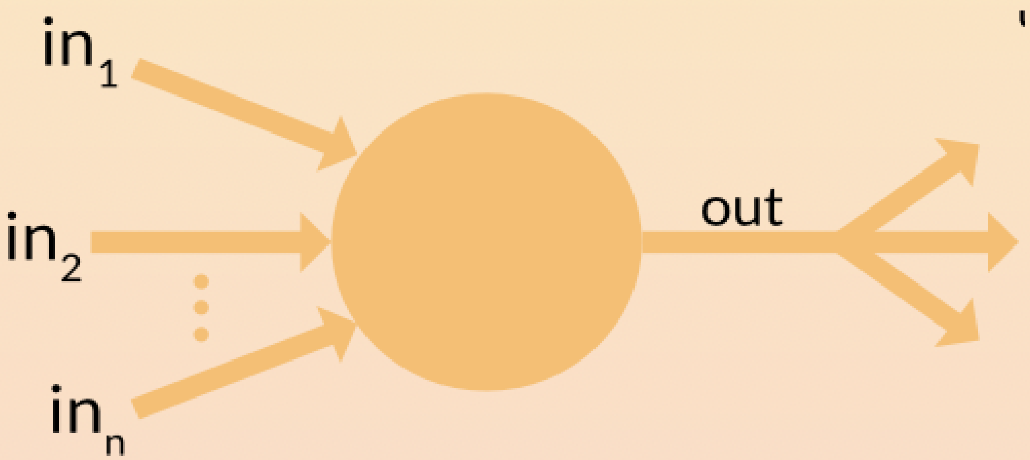


Figures from JHEP 11 (2018) 185

From Neurons to Perceptrons



$$\begin{aligned} I(t) = & \quad C\frac{dV}{dt} \\ + & \quad G_{Na}m^3h(V - V_{Na}) \\ + & \quad G_K n^4 (V - V_K) \\ + & \quad G_L(V - V_L) \end{aligned}$$

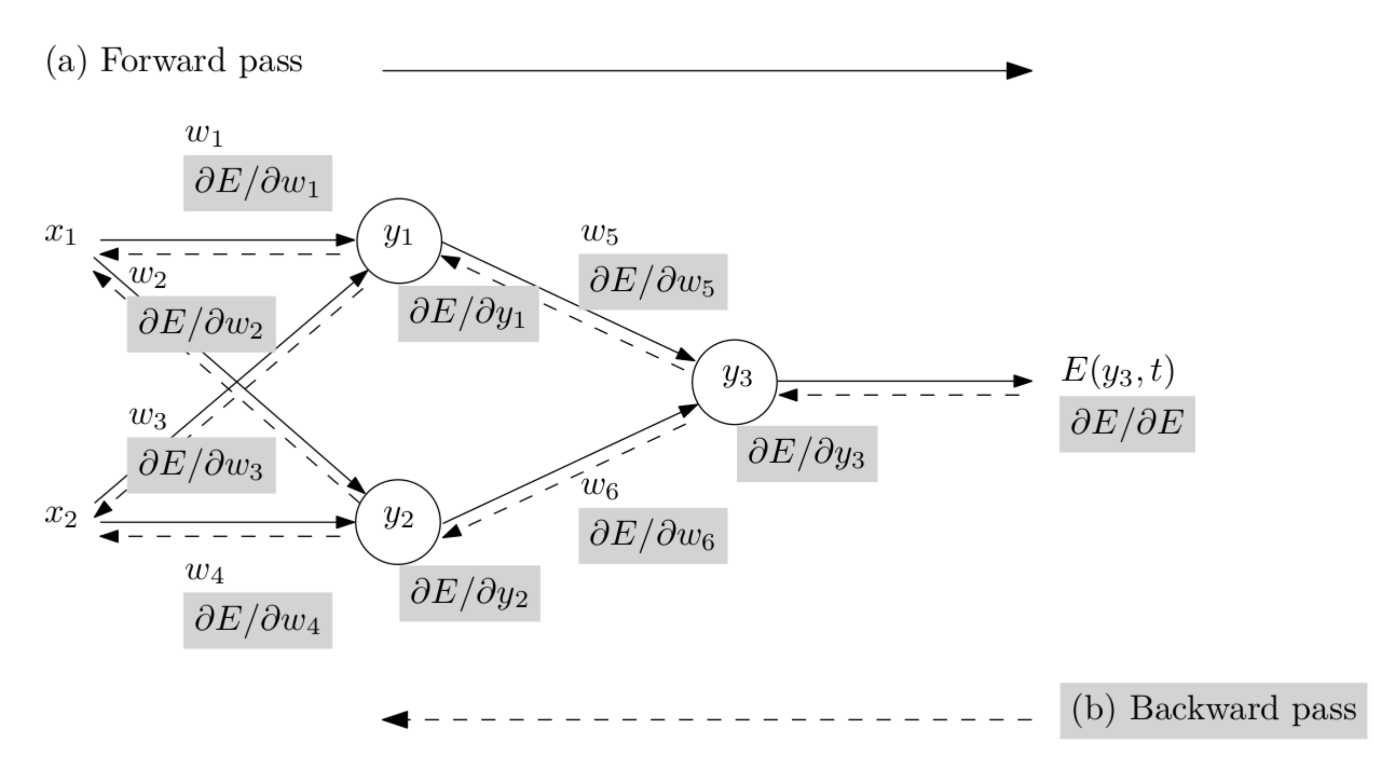


$$y = f\left(b_i + \sum w_i x_i\right)$$

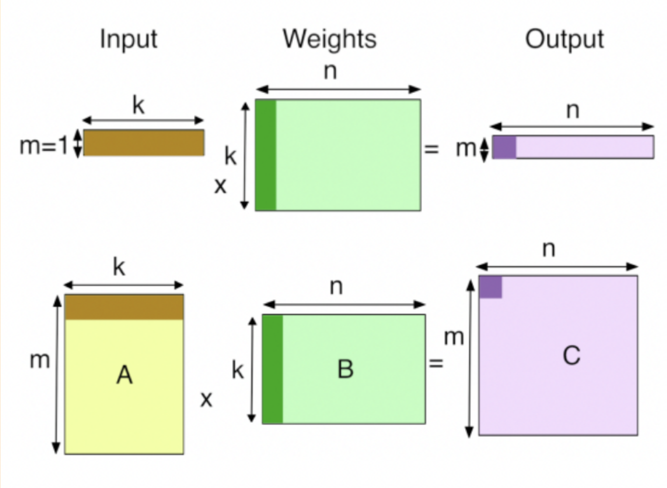
Image from https://appliedgo.net/perceptron/

Empirical Risk Minimization

$$\mathbf{J}(\mathbf{W}) = \frac{1}{n}\sum_{i=1}^{n}\mathcal{L}(f(x^{(i)};\mathbf{W}), y^{*(i)}), \qquad \mathbf{W}^0 = argmin_{\mathcal{W}}\mathbf{J}(\mathbf{W}), \qquad \mathbf{W} \leftarrow \mathbf{W} + \eta\frac{\partial \mathbf{J}(\mathbf{W})}{\partial \mathbf{W}}$$

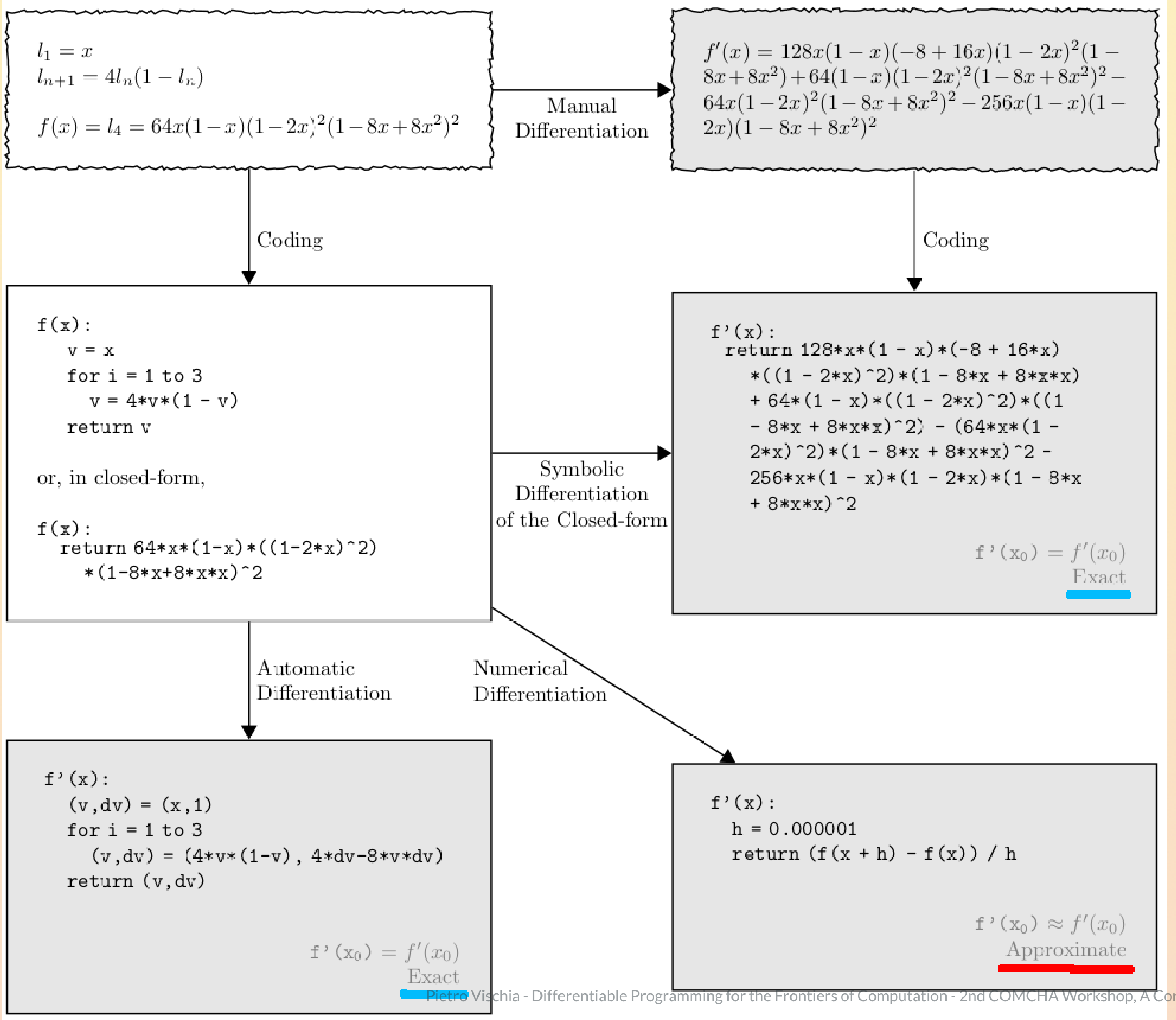


- Efficient matrix multiplication in dedicated hardware (GPUs, FPGAs)



Derive

- Manual
  - Error prone, unfeasible
- Symbolic
  - Expression swell, despite improvements
- Numerical
  - Truncation and rounding
- Automatic differentiation
  - Algorithmic differentiation
  - AD
  - Autodiff
  - Algodiff
  - Autograd



—
Image from Güneş Baydin et al, JMLR 18 (2018) 1--43

Autodiff powers most of modern ML

- By design, simple in software

```
import torch, math
x0 = torch.tensor(1., requires_grad=True)
x1 = torch.tensor(2., requires_grad=True)
p = 2*x0 + x0*torch.sin(x1) + x1**3
print(p)
p.backward()
print(x0.grad, x1.grad)
```

yielding

```
Primal: tensor(10.9093, grad_fn=<AddBackward0>)
Adjoint: tensor(2.9093) tensor(11.5839)
```

- Computational cost of calculating $\mathbf{J}_f(\mathbf{x})$ in $\mathbb{R}^n \times \mathbb{R}^m$ for $f : \mathbb{R}^n \to \mathbb{R}^m$
  - $\mathcal{O}(n \text{ time}(f))$
  - $\mathcal{O}(m \text{ time}(f))$

$$y(\mathbf{x}) = 2x_0 + x_0\, sin(x_1) + x_1^3$$

Fwd Primal Trace Atomic operation	Value in $(1,2)$	***Fwd Tangent Trace (set $\dot{x}_0 = 1$ to compute $\frac{\partial y}{\partial x_0}$)*** Atomic operation	Value in $(1,2)$
$v_0 = x_0$ $v_1 = x_1$	1 2	$\dot{v}_0 = \dot{x}_0$ $\dot{v}_1 = \dot{x}_1$	1 0
$v_2 = 2v_0$ $v_3 = sin(v_1)$ $v_4 = v_0 v_3$ $v_5 = v_1^3$ $v_6 = v_2 + v_4 + v_5$	2 0.9093 0.9093 8 10.9093	$\dot{v}_2 = 2\dot{v}_0$ $\dot{v}_3 = \dot{v}_1 cos(v_1)$ $\dot{v}_4 = \dot{v}_0 v_3 + v_0 \dot{v}_3$ $\dot{v}_5 = 3\dot{v}_1 v_1^2$ $\dot{v}_6 = \dot{v}_2 + \dot{v}_4 + \dot{v}_5$	2×1 0×-0.41 $1 \times 0.9093 + 1 \times 0$ $3 \times 0 \times 4$ $2 + 0.9093 + 0$
$y = v_6$	10.9093	$\dot{y} = \dot{v}_6$	2.9093

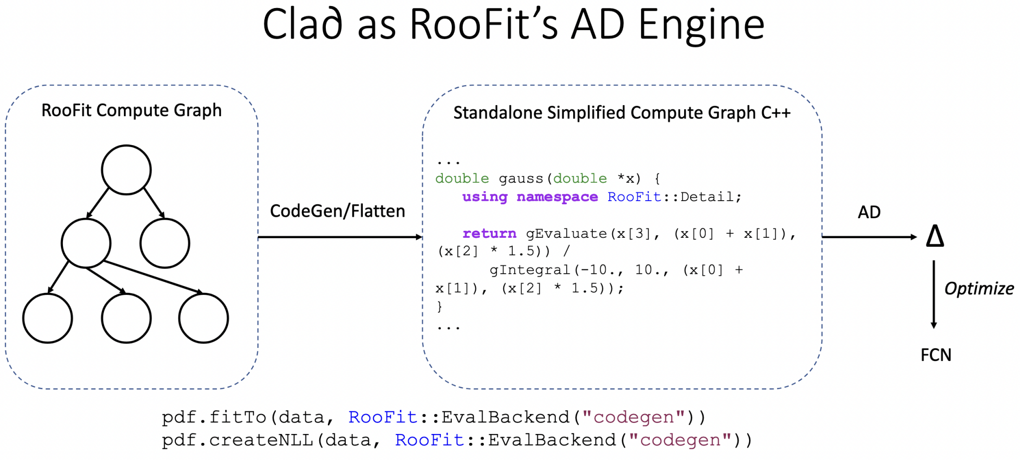
Fwd Primal Trace Atomic operation	Value in $(1,2)$	***Rev Adjoint Trace (set $\bar{y} = 1$ to compute $\frac{\partial v}{\partial y}$)*** Atomic operation	Value in $(1,2)$
$v_0 = x_0$ $v_1 = x_1$	1 2	$\bar{x}_0 = \bar{v}_0$ $\bar{x}_1 = \bar{v}_1$	2.9093 11.5839
$v_2 = 2v_0$ $v_3 = sin(v_1)$ $v_4 = v_0 v_3$ $v_5 = v_1^3$ $v_6 = v_2 + v_4 + v_5$	2 0.9093 0.9093 8 10.9093	$\bar{v}_0 = \bar{v}_0 + \bar{v}_2 \partial v_2/\partial v_0$ $\bar{v}_0 = \bar{v}_4 \partial v_4/\partial v_0$ $\bar{v}_1 = \bar{v}_1 + \bar{v}_3 \partial v_3/\partial v_1$ $\bar{v}_1 = \bar{v}_5 \partial v_5/\partial v_1$ $\bar{v}_2 = \bar{v}_6 \partial v_6/\partial v_2$ $\bar{v}_3 = \bar{v}_4 \partial v_4/\partial v_3$ $\bar{v}_4 = \bar{v}_6 \partial v_6/\partial v_4$ $\bar{v}_5 = \bar{v}_6 \partial v_6/\partial v_5$	$\bar{v}_0 + \bar{v}_2 \times 2 = 2.9093$ $\bar{v}_4 \times v_3 = 0.9093$ $\bar{v}_1 + \bar{v}_3 \times cos(v_1) = 11.5839$ $\bar{v}_5 \times 3v_1^2 = 12$ $\bar{v}_6 \times 1 = 1$ $\bar{v}_4 \times v_0 = 1$ $\bar{v}_6 \times 1 = 1$ $\bar{v}_6 \times 1 = 1$
$y = v_6$	10.9093	$\bar{v}_6 = \bar{y}$	1

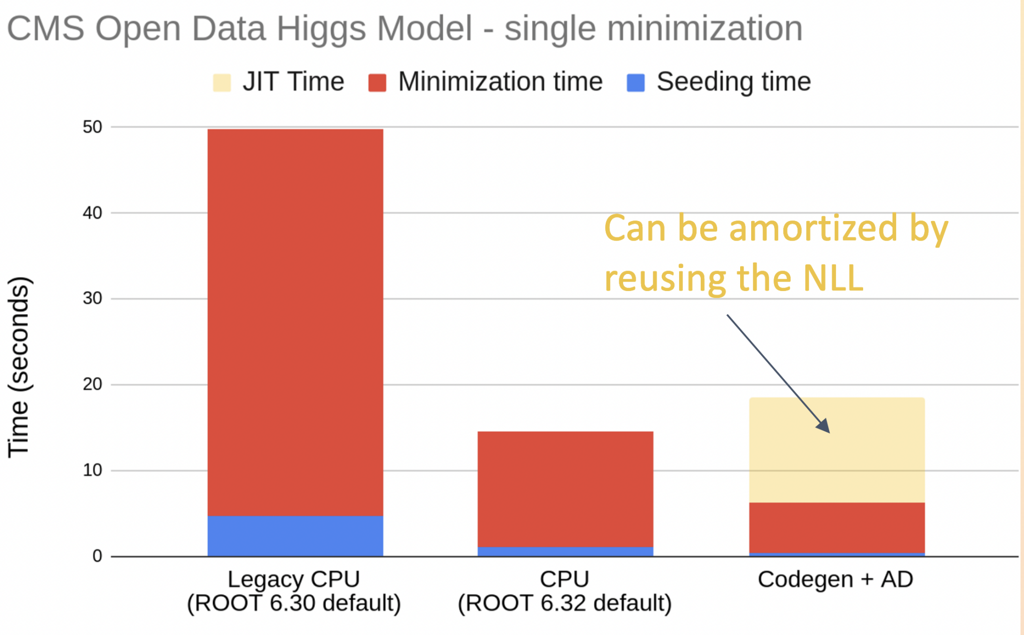
Image by P. Vischia, book in preparation

RooFit

- Clang/LLVM plugin run at compilation time
- Produces C++ code (readable, explainable)

- Huge gains in ATLAS and CMS open data benchmarks
- Faster gradient
- Numerically stable gradient
- Readable and shareable





Differentiable Programming (2018)

Execute differentiable functions (programs)
via automatic differentiation

Yann LeCun
January 5, 2018 ·

OK, Deep Learning has outlived its usefulness as a buzz-phrase.
Deep Learning est mort. Vive Differentiable Programming!

Yeah, Differentiable Programming is little more than a rebranding of the modern collection Deep Learning techniques, the same way Deep Learning was a rebranding of the modern incarnations of neural nets with more than two layers.

But the important point is that people are now building a new kind of software by assembling networks of parameterized functional blocks and by training them from examples using some form of gradient-based optimization.

An increasingly large number of people are defining the networks procedurally in a data-dependent way (with loops and conditionals), allowing them to change dynamically as a function of the input data fed to them. It's really very much like a regular progam, except it's parameterized, automatically differentiated, and trainable/optimizable. Dynamic networks have become increasingly popular (particularly for NLP), thanks to deep learning frameworks that can handle them such as PyTorch and Chainer (note: our old deep learning framework Lush could handle a particular kind of dynamic nets called Graph Transformer Networks, back in 1994. It was needed for text recognition).

People are now actively working on compilers for imperative differentiable programming languages. This is a very exciting avenue for the development of learning-based AI.

Important note: this won't be sufficient to take us to "true" AI. Other concepts will be needed for that, such as what I used to call predictive learning and now decided to call Imputative Learning. More on this later....

1.8K   186 Comments 464 Shares

—
Screenshot of Yann LeCun's facebook post

The usual suspect, in 1990

Making the World Differentiable: On Using Self-Supervised Fully Recurrent Neural Networks for Dynamic Reinforcement Learning and Planning in Non-Stationary Environments

Jürgen Schmidhuber*
Institut für Informatik
Technische Universität München
Arcisstr. 21, 8000 München 2, Germany
schmidhu@tumult.informatik.tu-muenchen.de

—
Photo from Juergen

The usual suspect, in 1990

Concluding Remarks

Program Inputs Differentiable with Respect to Programs

Let us view a network with a fixed topology as a computer. Its *program* is the weight matrix. One of the most interesting aspects of many connectionist algorithms is that program outputs are differentiable with respect to programs. A simple program generator (the gradient descent procedure) produces increasingly successful programs if the desired outputs are known.

In typical reinforcement learning situations, the environment is not *a priori* represented in a differentiable form. So the main reason for building connectionist world models in the style above is to 'make the world differentiable'. Thus even *program inputs* can become differentiable with respect to programs. World models thereby close the gap between outputs and inputs. A differentiable world model allows the program generator to perform an informed search for better goal directed programs.

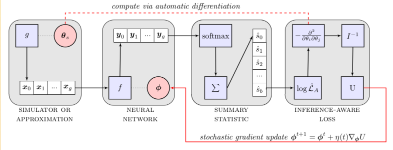
The degree of informedness of this search for suitable programs is a principle difference between the approach presented in this paper and the reinforcement learning algorithms for recurrent nets in

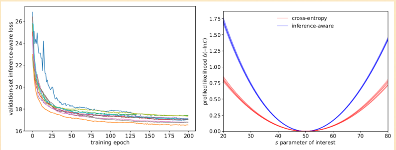
—
Photo from Juergen

Lot of efforts, plus several works in this session!

In the following, I will describe some recent developments and then focus on work with my students and collaborators

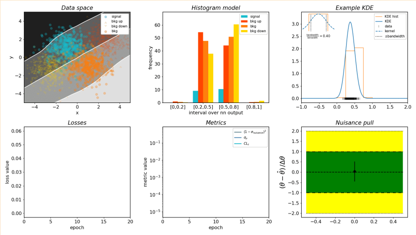
INFERNO (10.1016/j.cpc.2019.06.007)





(a) inference-aware training loss (b) profile-likelihood comparison

neos (10.1088/1742-6596/2438/1/012105)



Local generative surrogates (2002.04632)

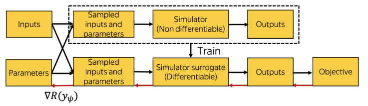
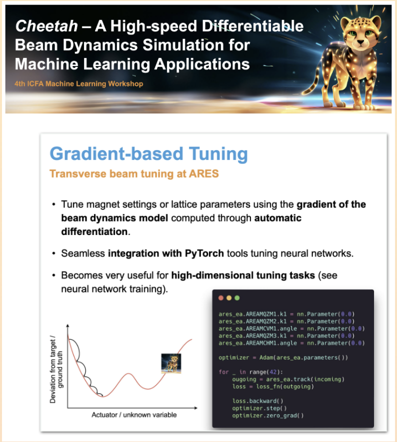
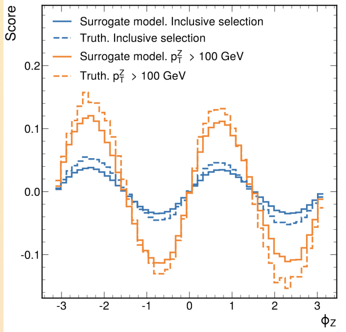


Figure 1: Simulation and surrogate training. *Black:* forward propagation. *Red:* error backpropagation.

Cheetah (2401.05815)



CP-optimal observables (2405.13524, Cruz et al. (P.V.))



SWGO optimization (2310.01857)

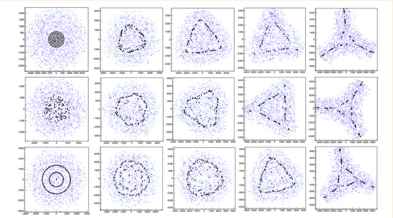
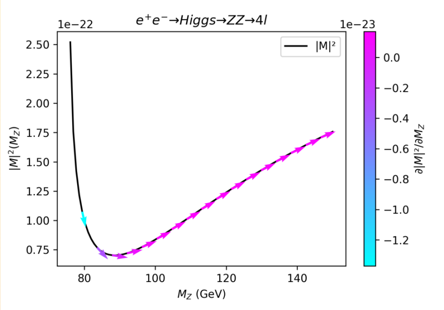


Figure 14: Convergence of three initial layouts (top to bottom: packed ball, wide random ball, two annuli) during a 500-epochs training. From left to right, the configurations of 126 units (129 in the bottom one) are shown at epoch 1,50,150,300, 500. See the text for more detail.

MadJax (10.1088/1742-6596/2438/1/012137)



Differentiable showers in GEANT4 (2405.07944)

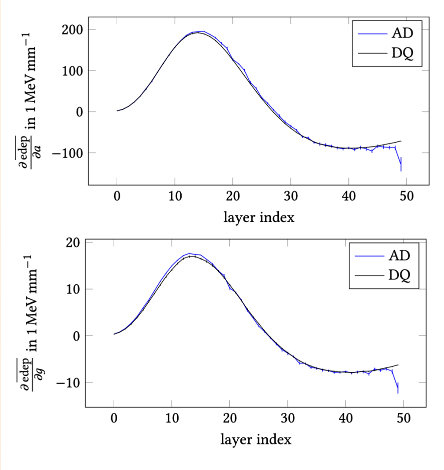


Figure 10: Algorithmic derivative of the edep with respect to the absorber thickness a (top) and gap thickness g (bottom).

Short-term solution: differentiable surrogate models

- Subset of relatively simple class of functions (but they must be able to reproduce $F(\cdot)$ well)
- Learn by training (hic sunt leones), (but $N(\text{eval } F) \geq \mathcal{O}(dim(\theta))$)
- Automatically get AD out of the box even if original $F(\cdot)$ is not differentiable
- Evaluation of surrogate (for optimization) much faster than evaluation of $F(\cdot)$

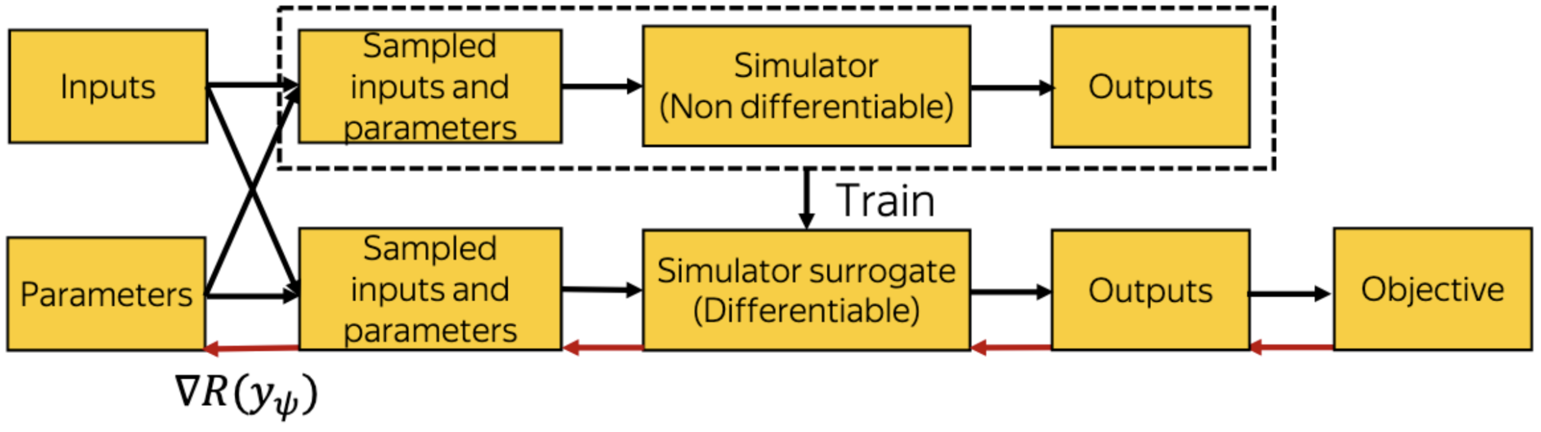


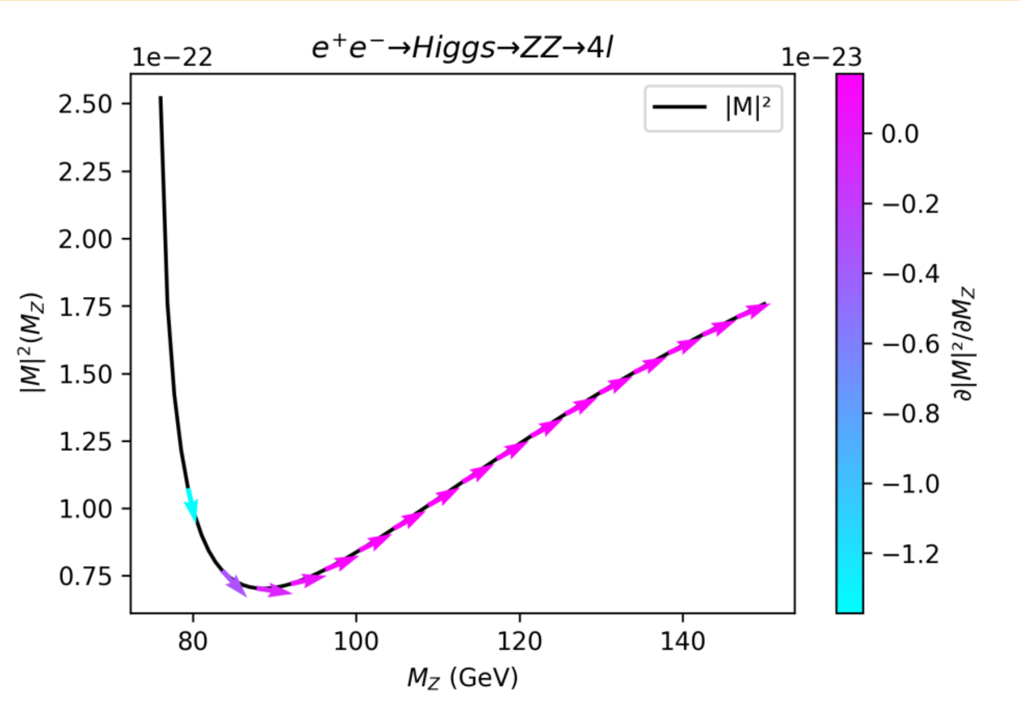
Figure 1: Simulation and surrogate training. *Black:* forward propagation. *Red:* error backpropagation.

—
Image from Shirobokov,..., Kagan, ... et al.

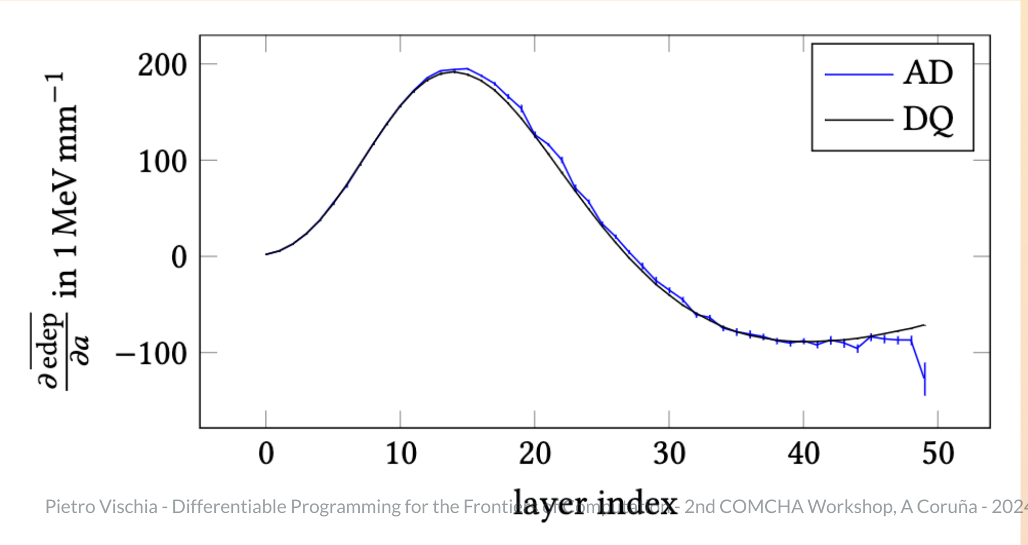
Long-term solution: make everything differentiable

- Multi-channel integration speed up in MadGraph(MadNIS)

MadJax (2203.00057)
(differentiable matrix element computation)

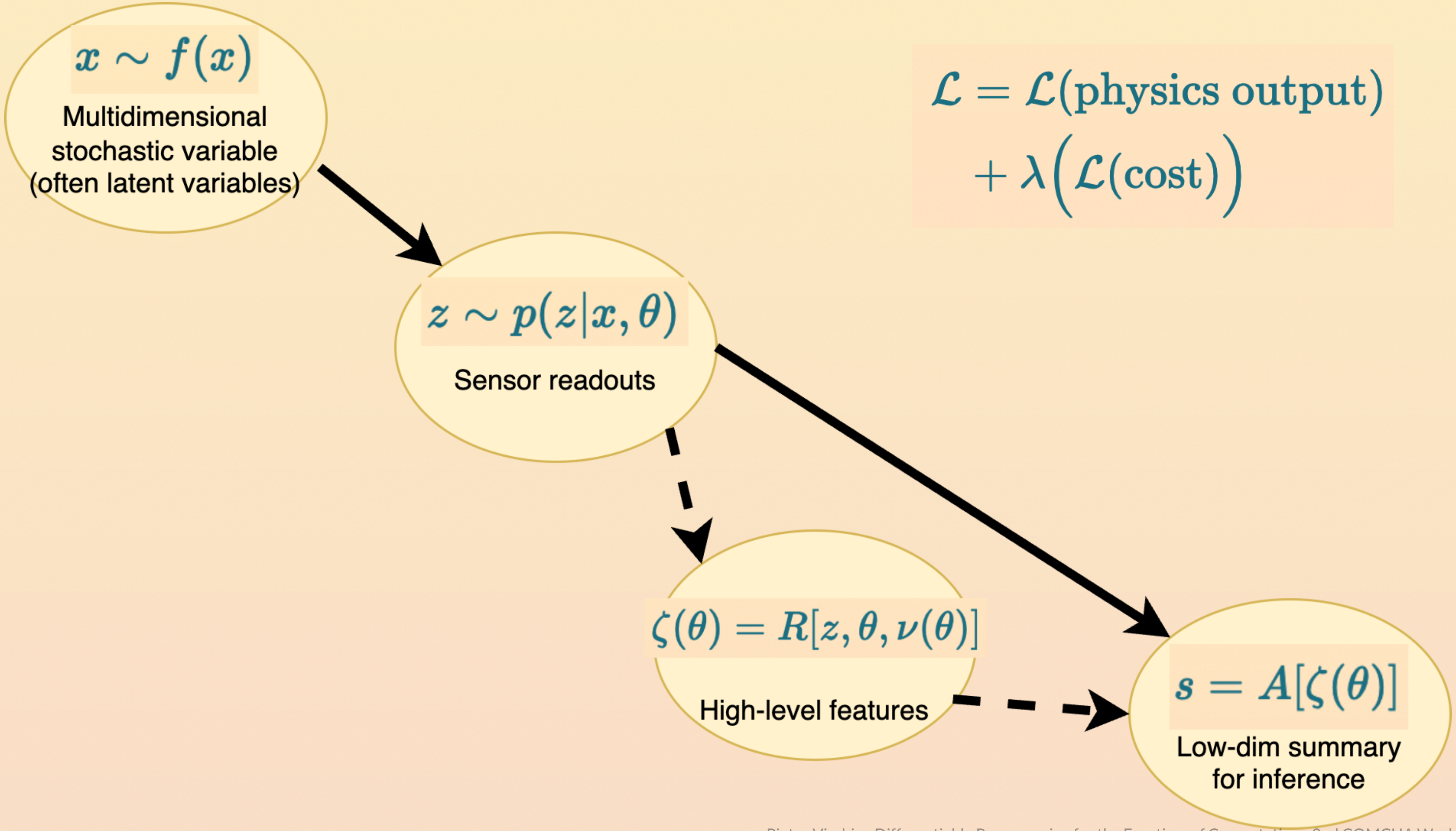


- Detector simulation: GATE/GEANT4 numerically differentiable (in small ranges) (2202.05551)
- Differentiable electromagnetic showers for GEANT4 (2405.07944)
  - **Derivgrind**: insert AD logic into the program (a sort of debugger): cannot support tricky cases
  - **CoDiPack**: operator overloading (e.g. replace `double` type): can run out of memory when storing the real-arithmetic evaluation graph (tape)
  - **Clad**: compiler-based source transformation tools: could use smaller tapes, more advanced optimization



Measurement-aware detector opt.!

- Joint optimization of design parameters w.r.t. inference made with data
- MODE White Paper, 10.1016/j.revip.2023.100085 (2203.13818), 117-page document, physicists + computer scientists



$$\mathcal{L} = \mathcal{L}(\text{physics output}) + \lambda\Big(\mathcal{L}(\text{cost})\Big)$$

Formulas from our white paper

Feasibility within constraints

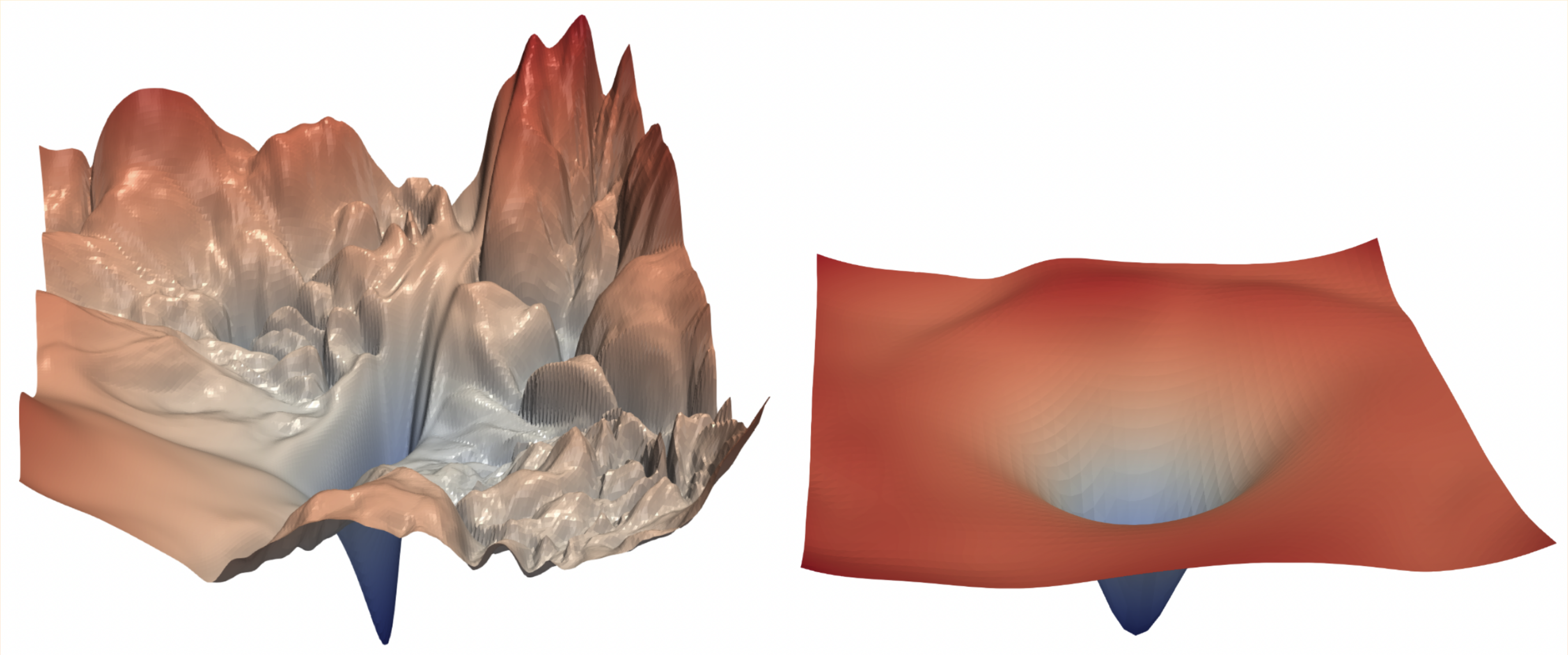
$$\hat{\theta} = arg\ min_\theta \int \underbrace{L[\overbrace{A(\zeta)}^{\text{Depends on z and nuisances}}, \overbrace{c(\theta)}^{\text{Cost of the layout with parameters theta}}]}_{\text{Weight desirable goals while obeying cost constraints}} \overbrace{p(z|x,\theta)}^{\text{Closed form}} f(x)dxdz\ ,$$

- Costs can be monetary but also any case-specific technical constraint
    - Local, specific to the technology used (e.g. active components material)
    - Global, describing overall detector conception (e.g. number, size, position of detector modules)
- Fixed costs can be added separately to the loss function

—
From our white paper and internal TomOpt presentation

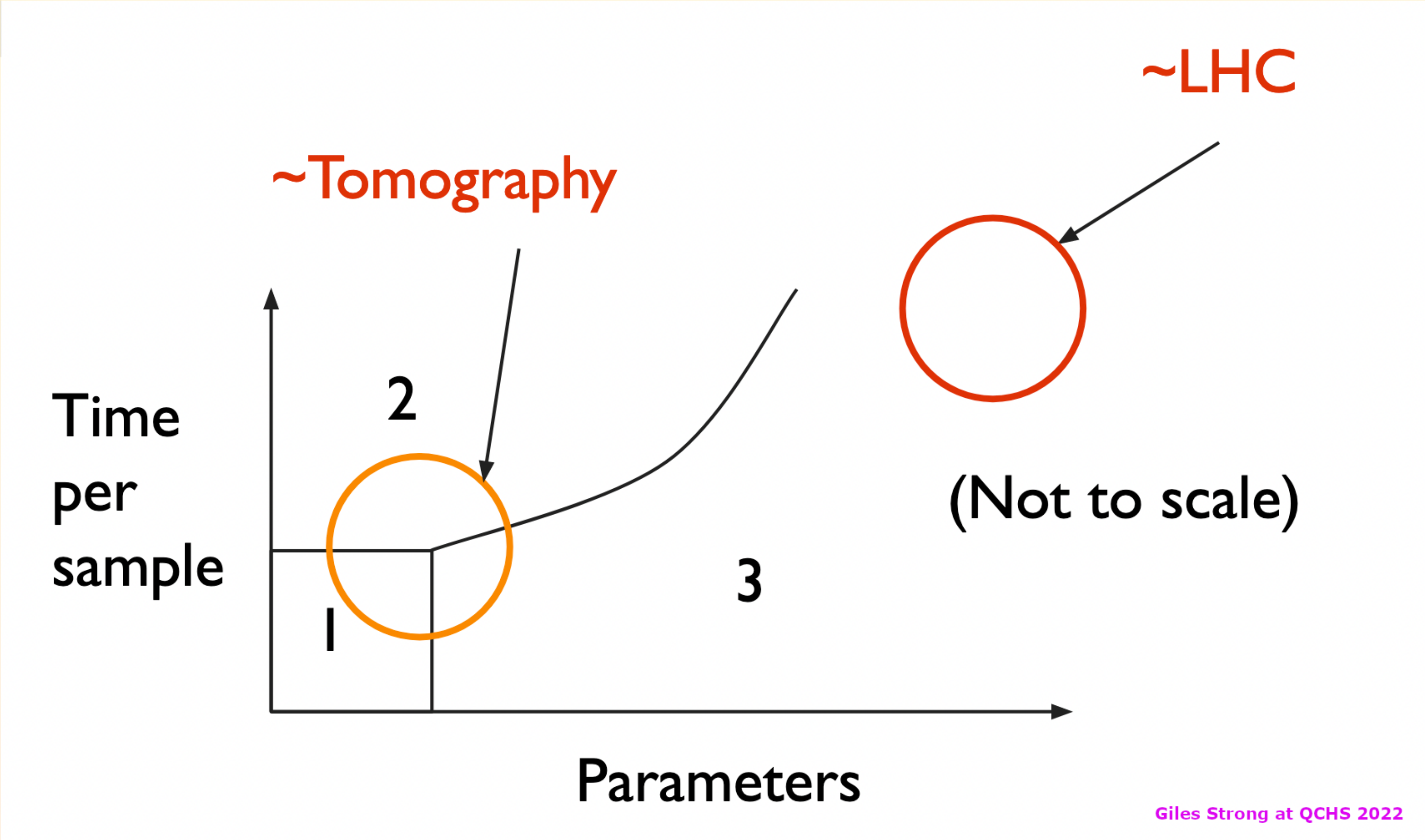
Assist the physicist with a landscape of solutions

- Results are as good as your parameterization: cannot parameterize everything!
- "The optimal solution" is unrealistic: provide feasible solutions near optimality
- The physicist will fine tune



—
Image from Li et al (taken as pictorial representation out of its original context)

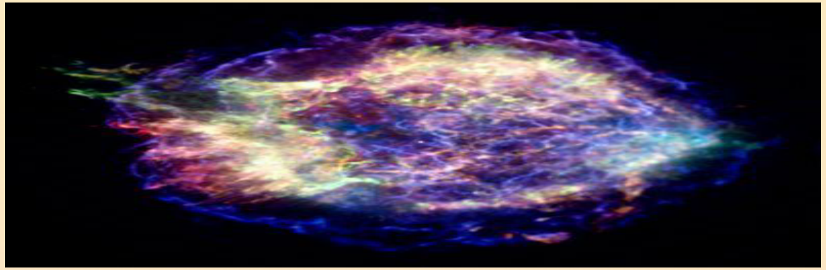
Method of choice depends on scale

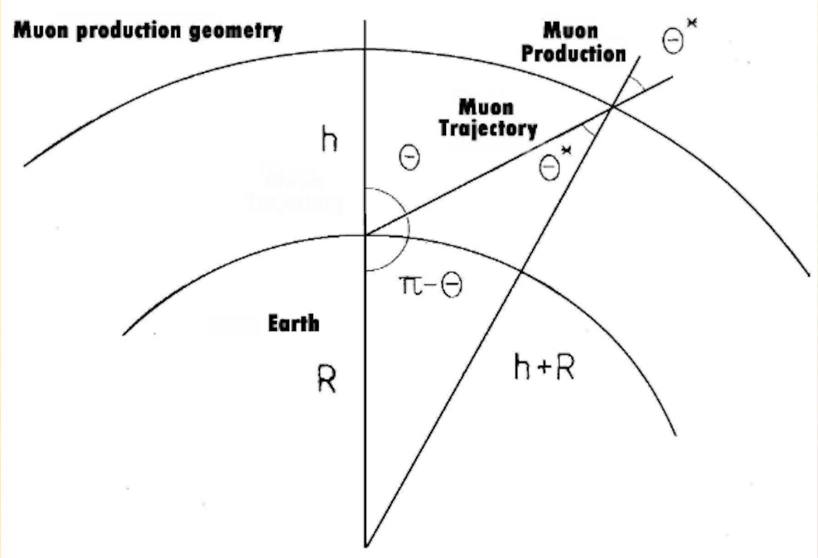


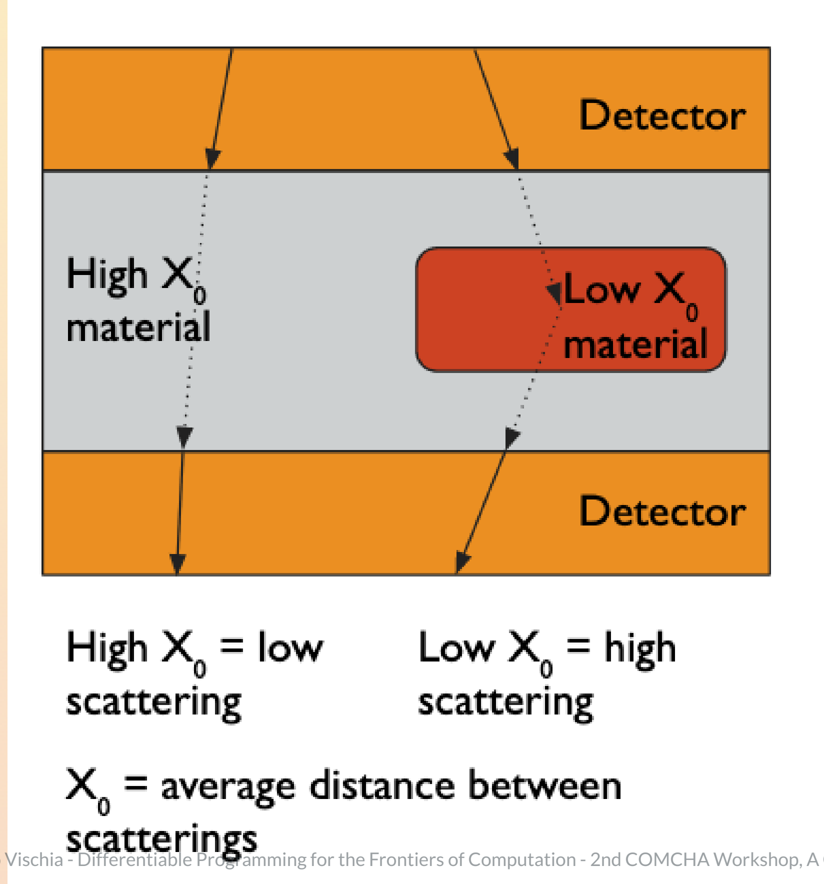
1. Grid/random search
2. Bayesian opt, simulated annealing, genetic algos, ...
3. Gradient-based optimization (Newton, BFGS, gradient descent, ...)

Muon tomography

- Muons from high-energy primary cosmic rays that impact with the atmosphere
- Infer properties (e.g. 3D map of elemental composition X_0) of unknown volume: stochasticity!
- Tomography by absorption (measure missing flux, e.g. pyramids, volcanes) or scattering (measure deflection, e.g. containers, furnaces, statues)





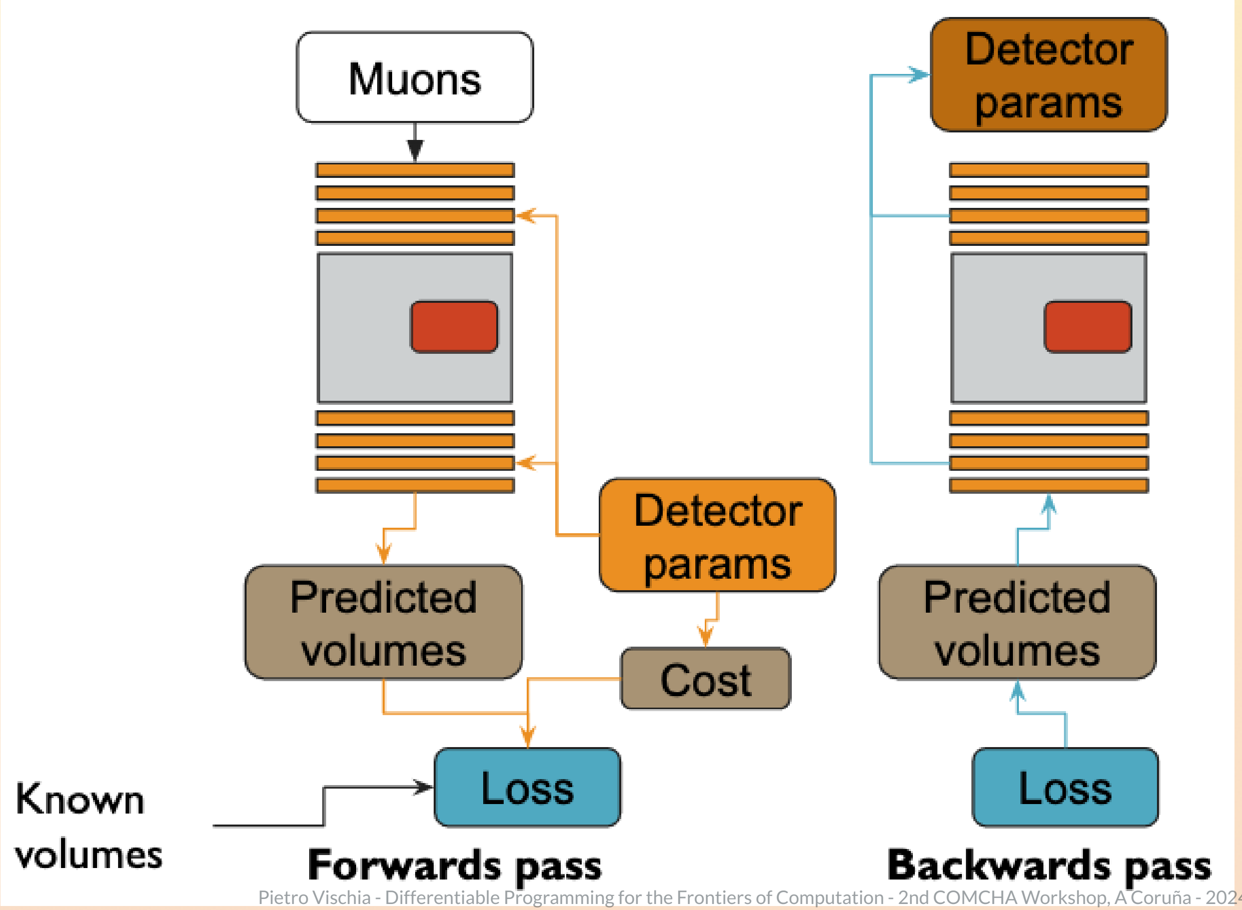


Images from uncw.edu, Guan et al. (2015), and the TomOpt project

TomOpt

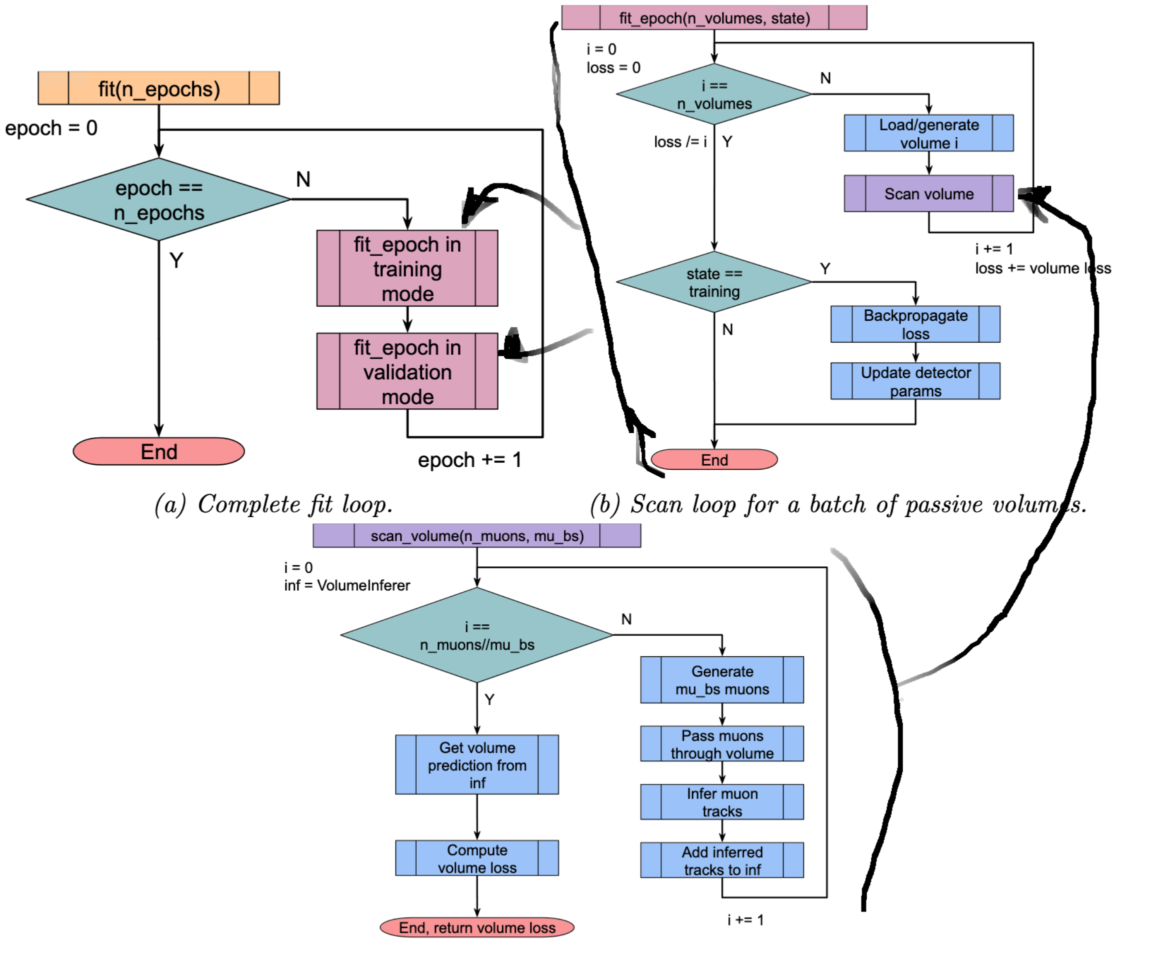
- Differential optimization of muon-tomography detectors
  - 10.1088/2632-2153/ad52e7 (Mach. Learn.: Sci. Technol. 5 035002)
  - Modular design in python, autodiff via PyTorch

- Inference chain as differentiable pipeline
  - Can compute $p(\Delta output|\Delta detector\ parameters)$
- Task as loss function
  - Including target (e.g. prediction uncertainty), costs, constraints
- Backpropagate and optimize as usual
  - Gradient descent



—
Figure from the TomOpt project

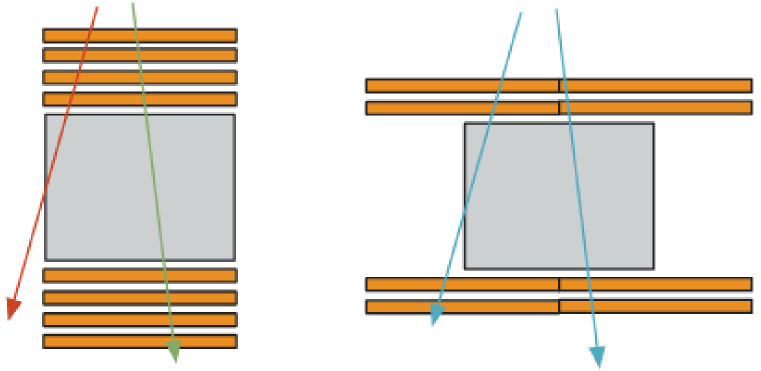
Optimize a parametric design under constraints



(a) *Complete fit loop.* (b) *Scan loop for a batch of passive volumes.*

(c) *Scan loop for muons over a single passive volume.*

Figure 6: Breakdown of the fitting procedure of detectors in TomOpt



Example 1: Muons measured precisely but less efficiently

Example 2: Muons measured less precisely but more efficiently

Figure from the TomOpt project and 10.1088/2632-2153/ad52e7

Make things differentiable

- Panels made differentiable only during training
- Associate to a muon hit a distribution based on resolution and efficiency

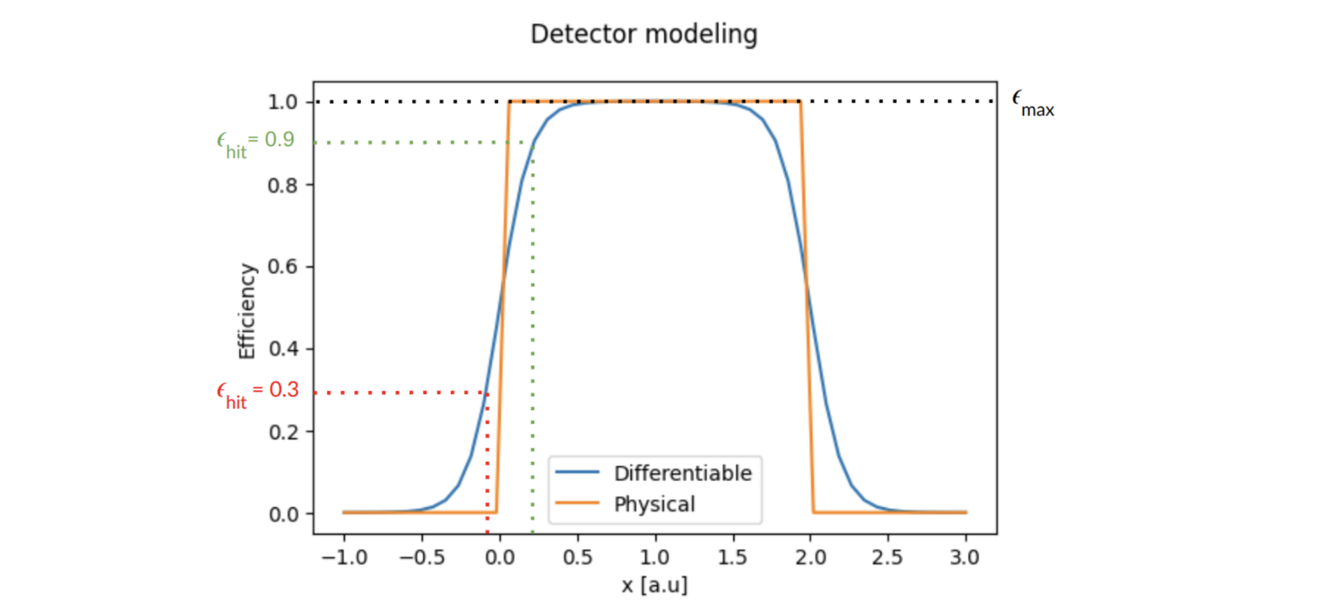
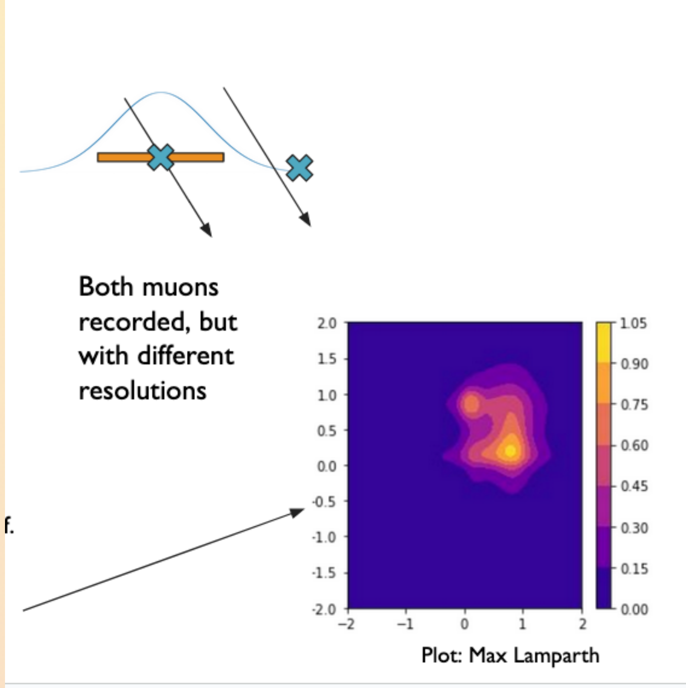


Figure 2: Example of detector panel modeling with sigmoid function used during optimisation (blue) and with rectangle function used during validation (orange).



Heavy-Metal Benchmark

Figure from 10.1088/2632-2153/ad52e7, Boris Bukovsky, viaPixabay

Heavy-Metal Benchmark

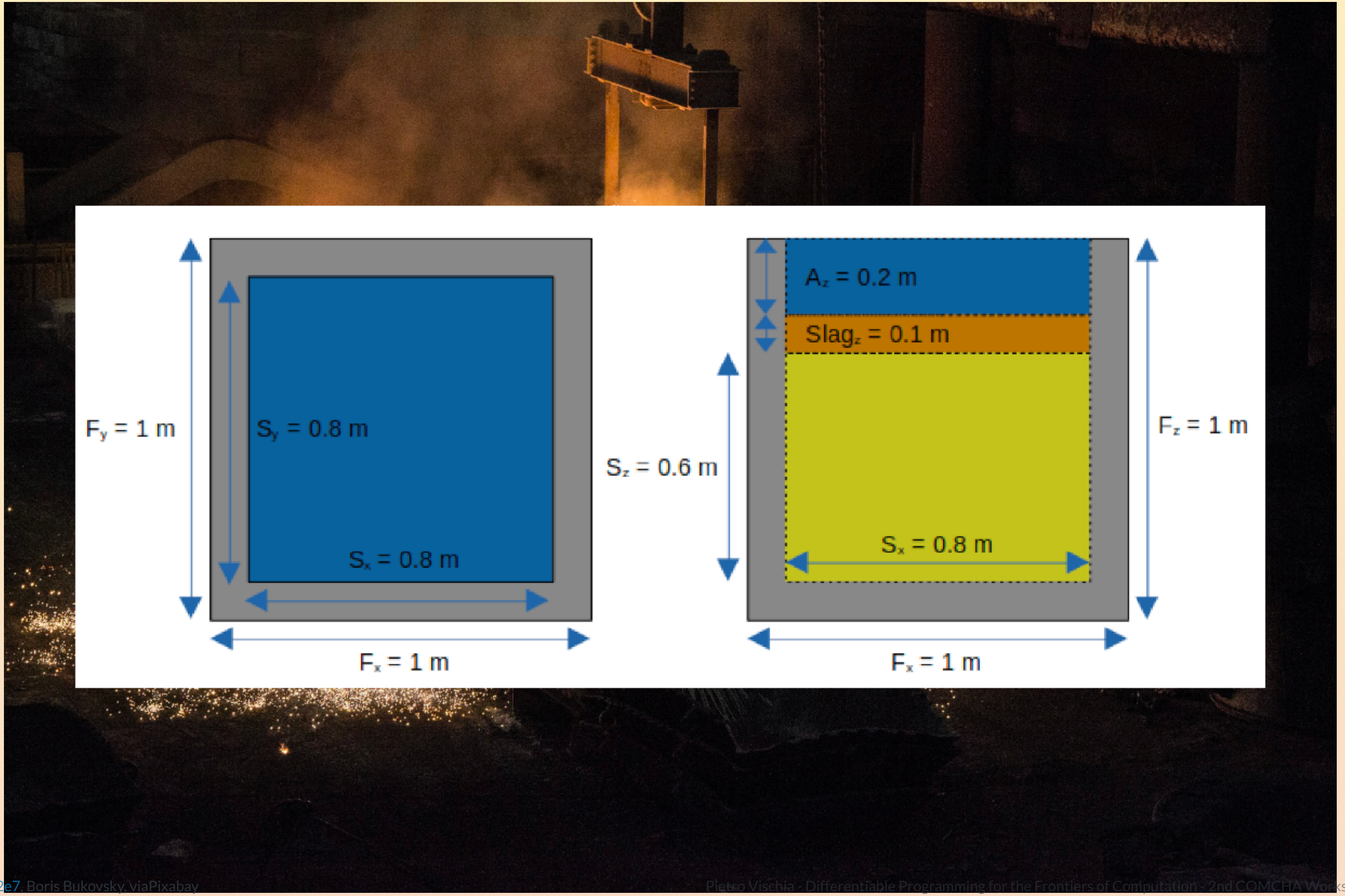
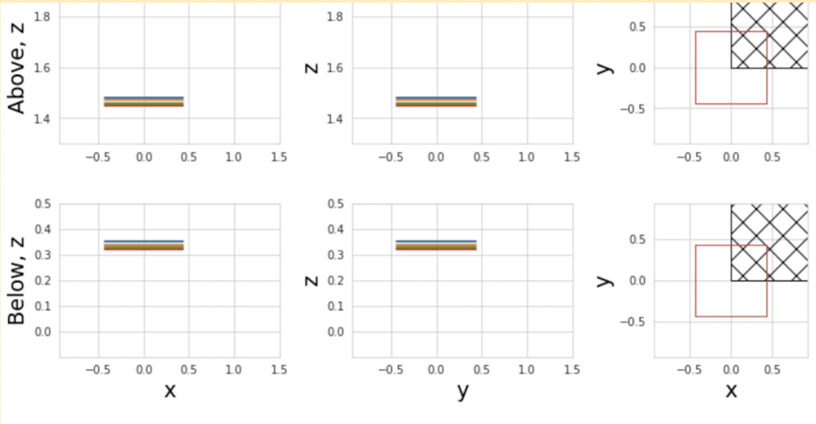
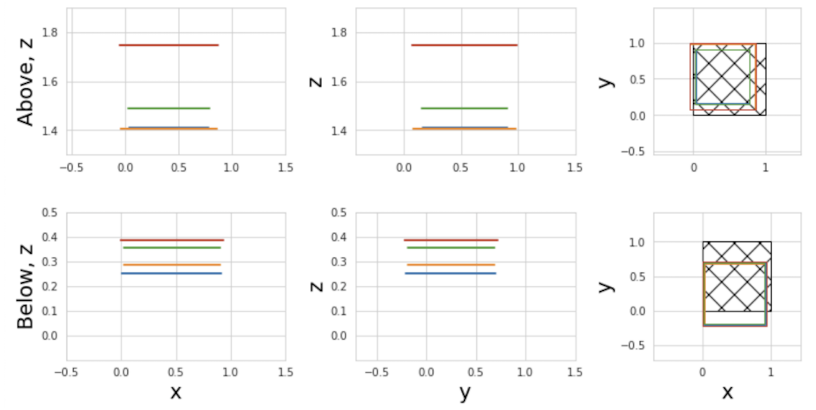


Figure from 10.1088/2632-2153/ad52e7, Boris Bukovsky via Pixabay

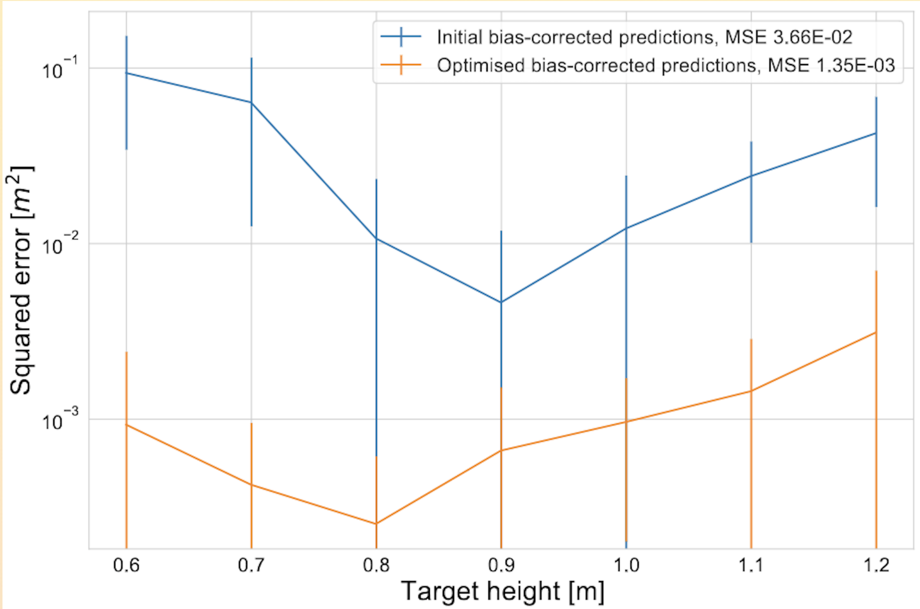
Optimized design is more performant

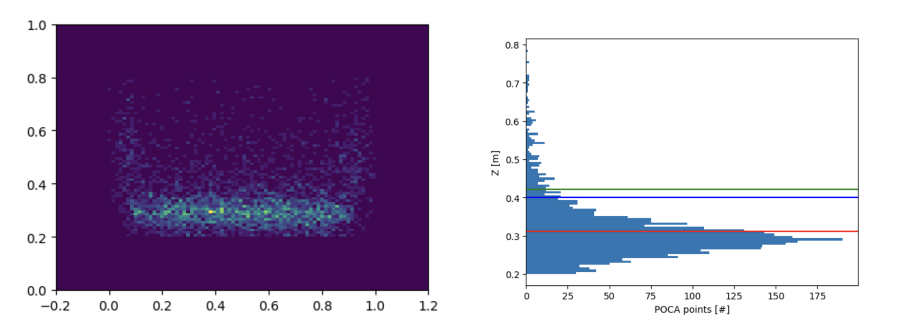


(a) Initial detector configuration.



(b) Detector configuration after stage one optimisation process.

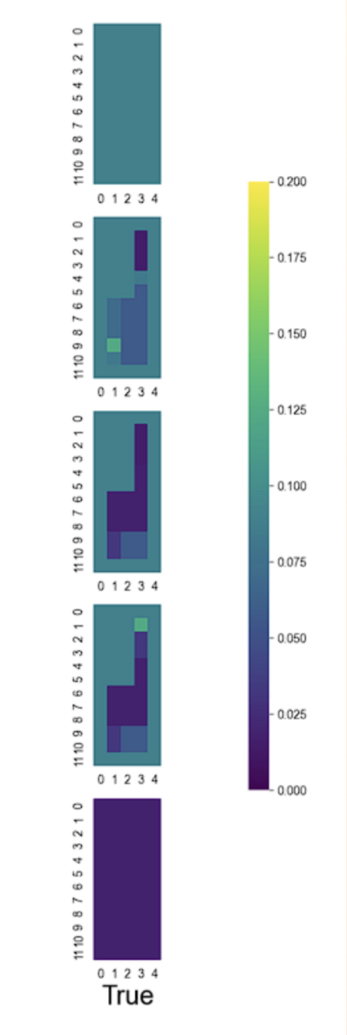


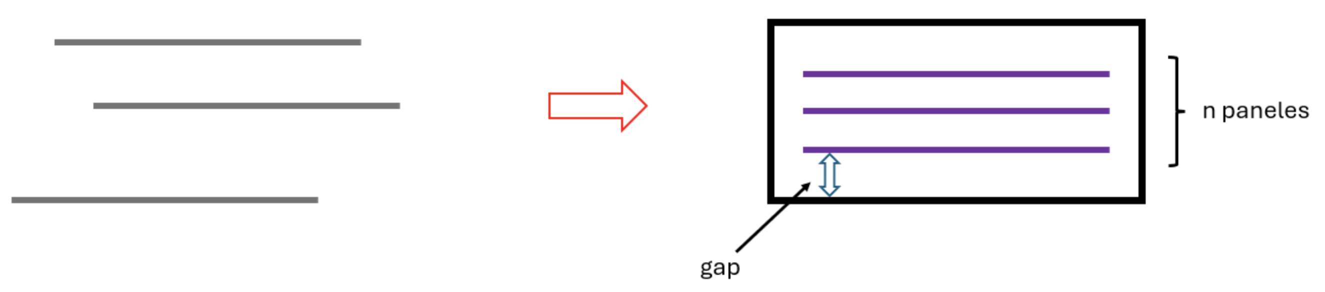


(Blue: true steel level. Red: prediction. Green: bias-corrected prediction)

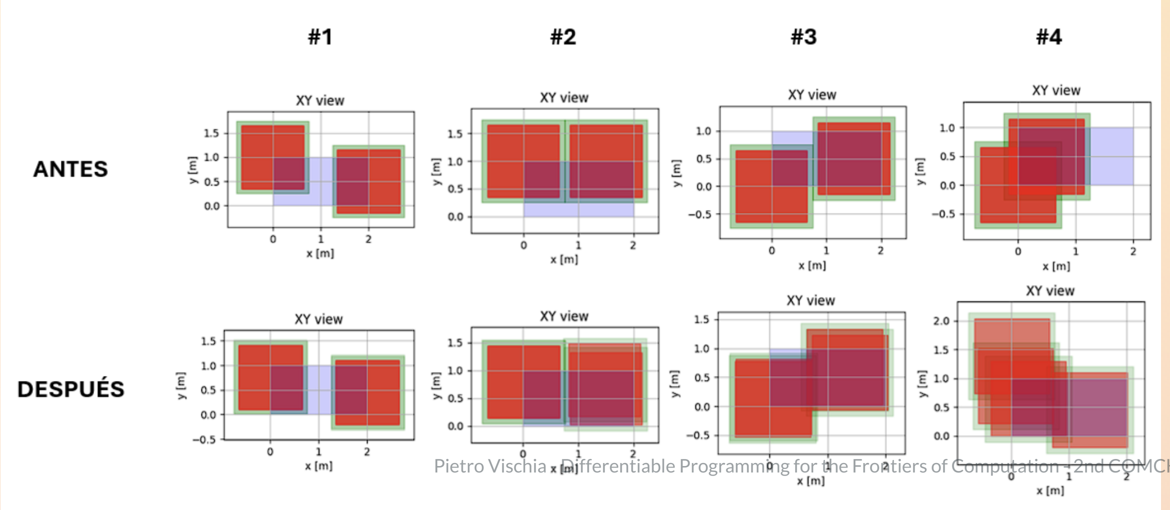
Extension: hodoscopes

- Setup from European plan for border control: hodoscope structure already decided, only placement can be changed





- Scan a lorry: significant code restructuring to account for hodoscope structure
- Accepted by the MARESEC conference for paper publication
  - Zaher, Lagrange, Àlvarez Lueje et al. (P.V.), 2024 (BSc thesis of Samuel Àlvarez Lueje)



Neutron Tomography

- See the excellent talk by María Pereira Martínez, in this session!!!
  - A summary is in backup

—
Image from timesofindia.com

Thrive in asymmetries or lack thereof

2006: genetic algorithms

2024: SWGO tanks placement optimization

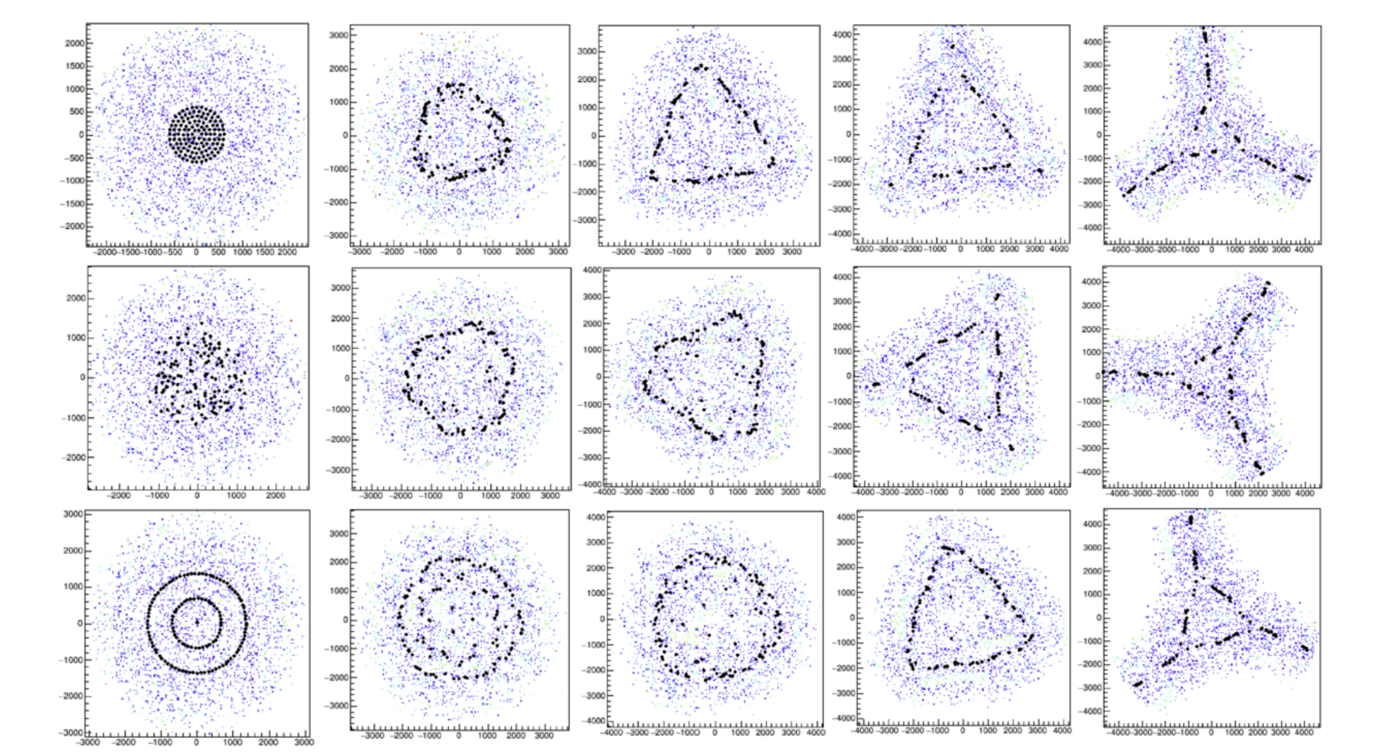
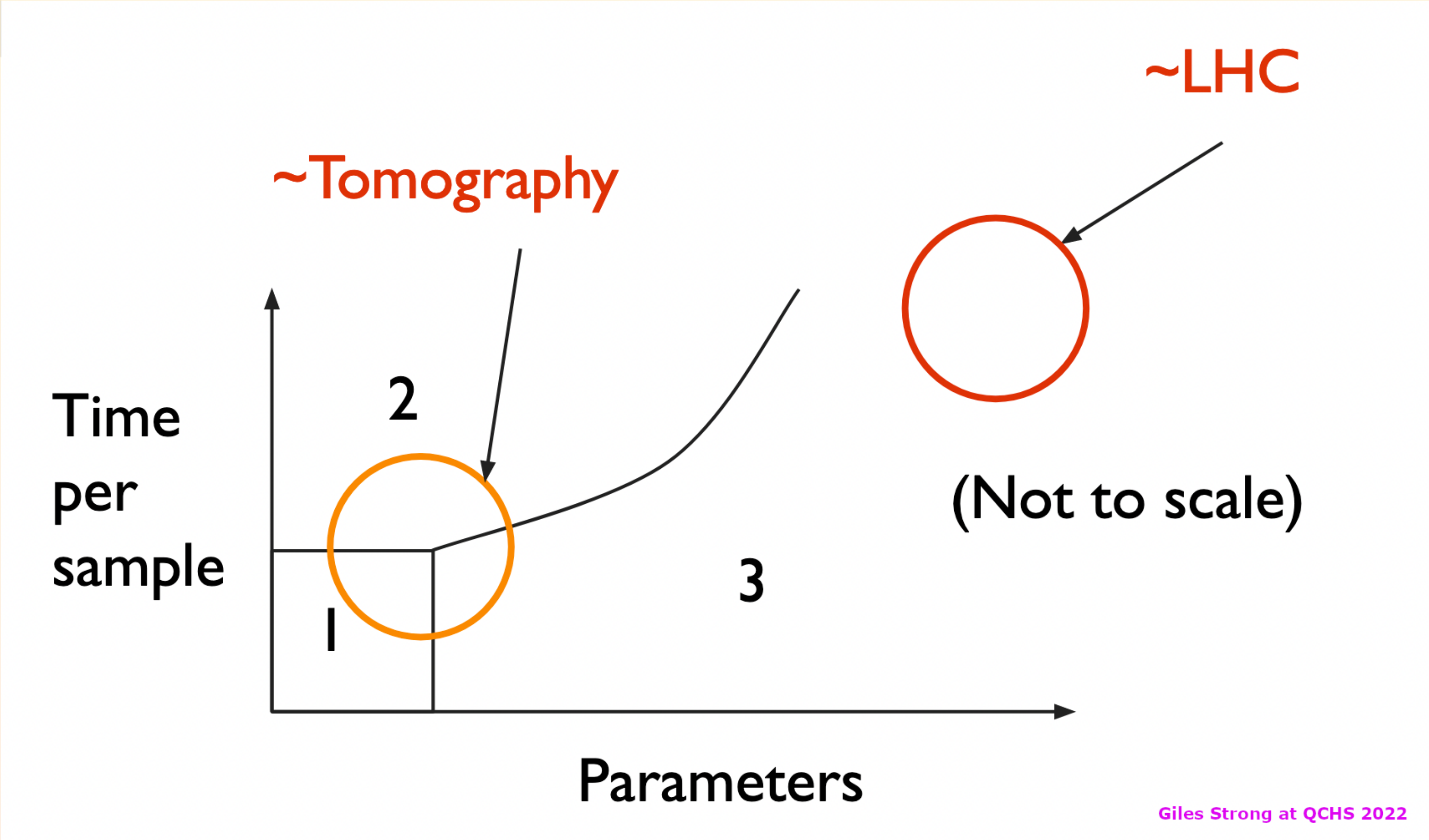


Figure 14: Convergence of three initial layouts (top to bottom: packed ball, wide random ball, two annuli) during a 500-epochs training. From left to right, the configurations of 126 units (129 in the bottom one) are shown at epoch 1,50,150,300, 500. See the text for more detail.

Figures from doi:10.2514/6.2006-7242 and 2310.01857

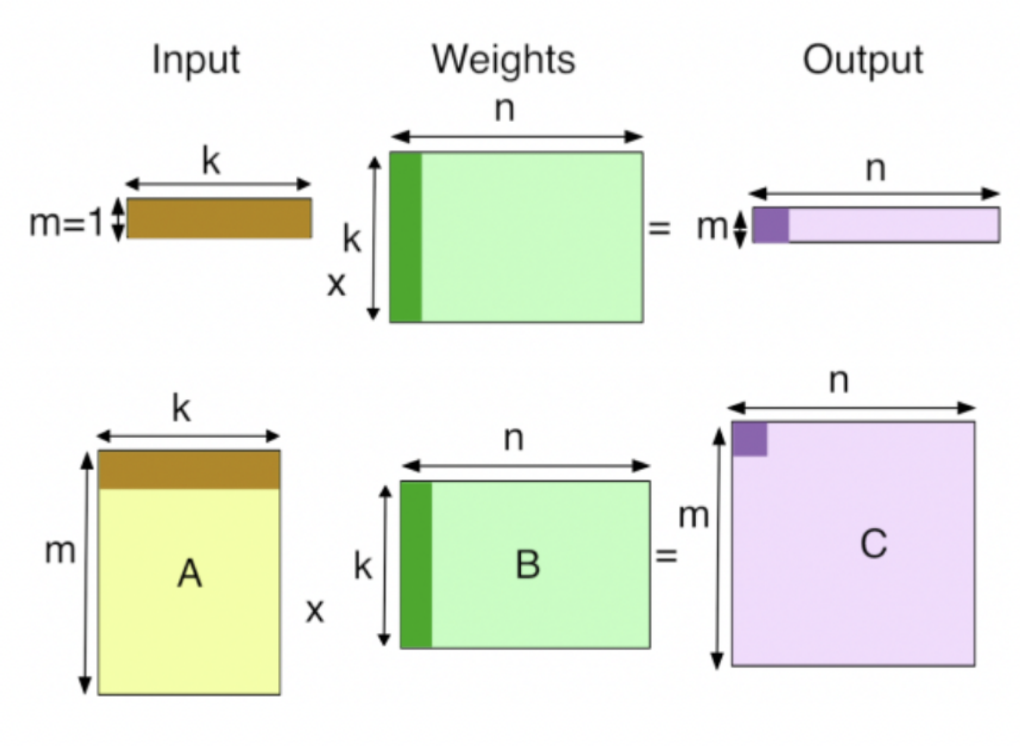
Experimental design: present and future

- Gradient descent applied to experiment design works!!!
  - Discreteness and stochasticity mostly solvable or avoidable
- Can we make it scale?

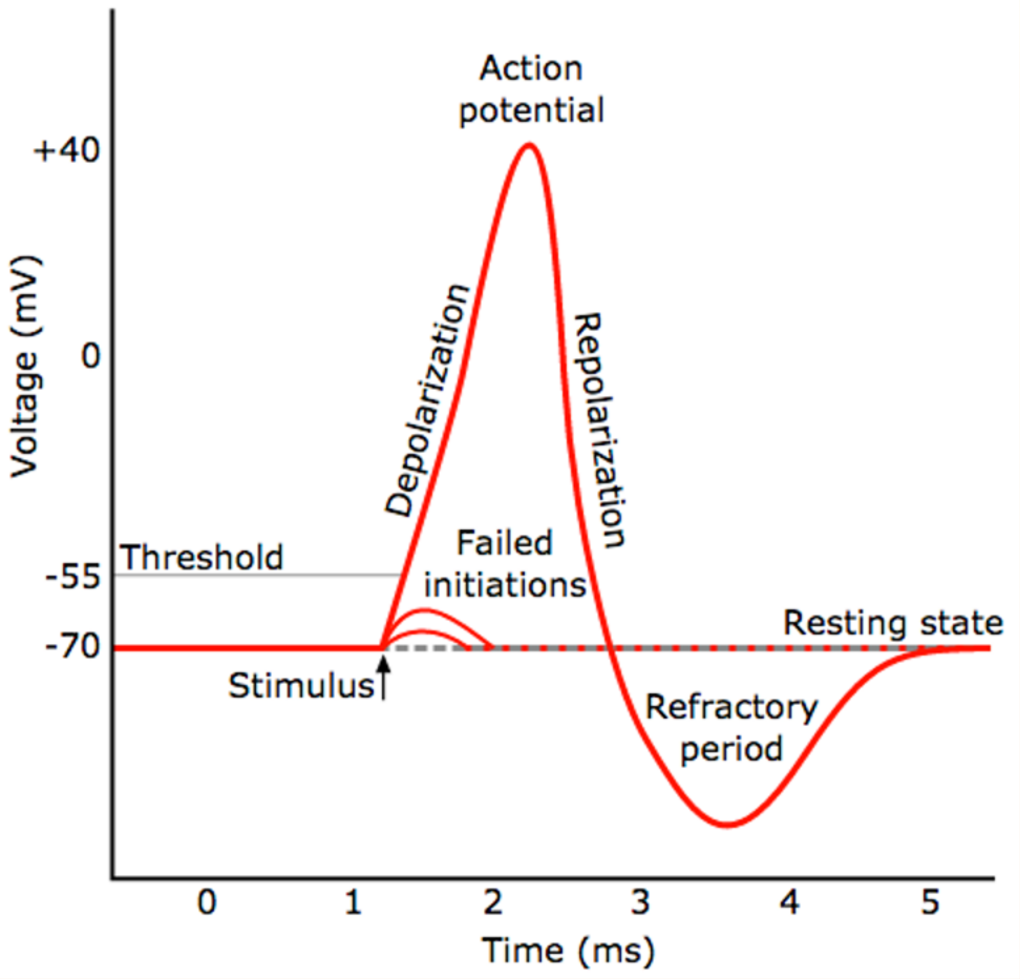


Paradigm shift

From perceptrons and matrices...



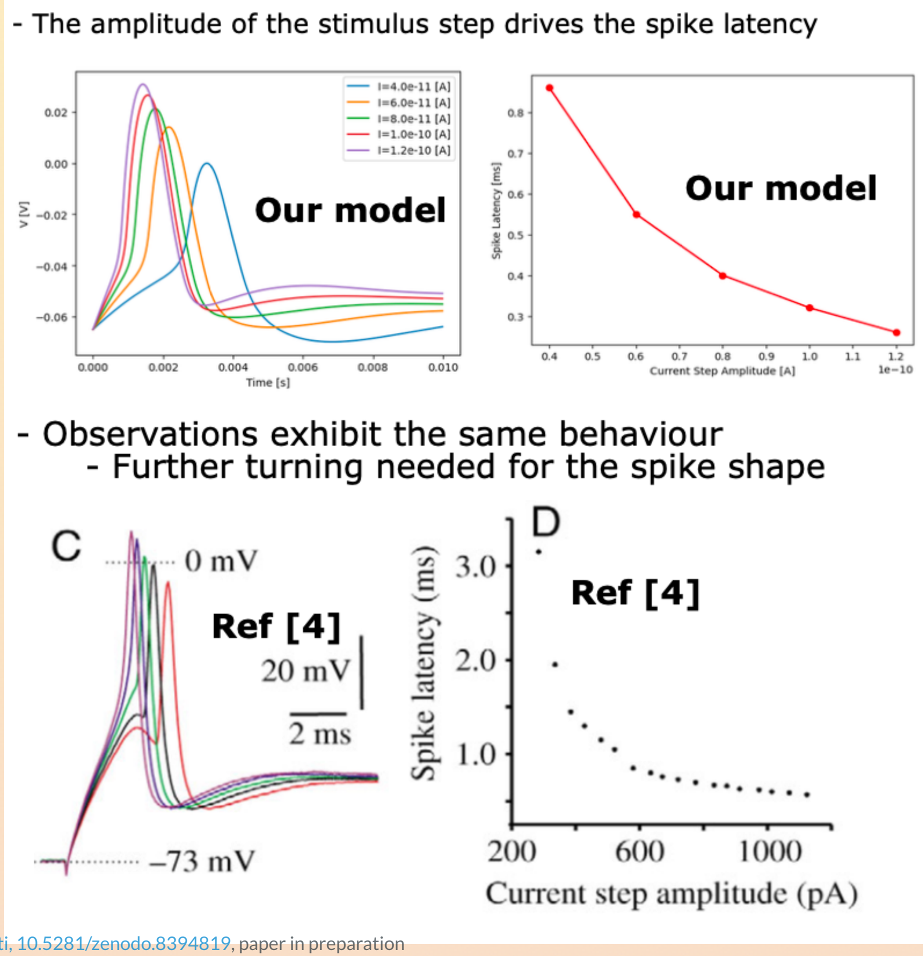
to spiking neurons modulating spatiotemporal patterns

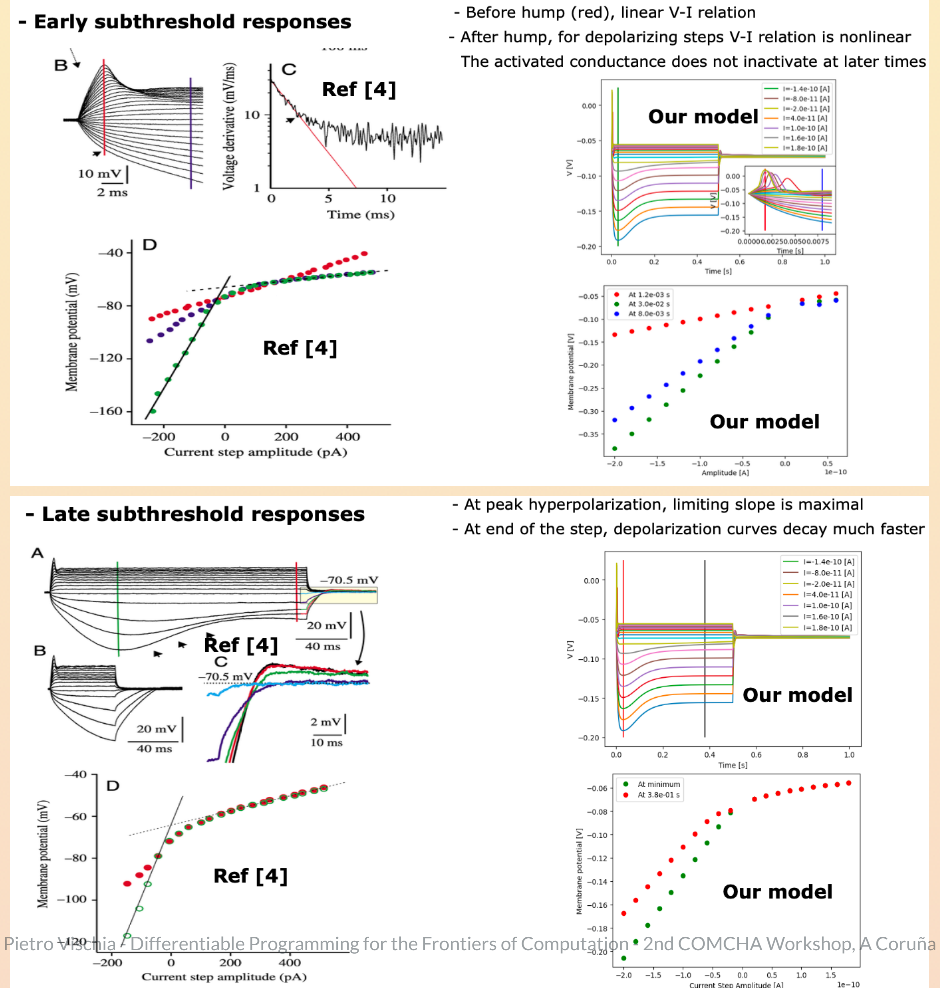


—
Figures by Gilles Louppe and bioserver

Realistic models for biological neurons

- Spherical neuron with four channels (different thresholds and time constants) for *Gymnotus Omarorum*
    - Vischia, Caputi 2023: computational model compared with data from "[4]" (J Exp Biol (2006) 209 (6): 1122–1134.)

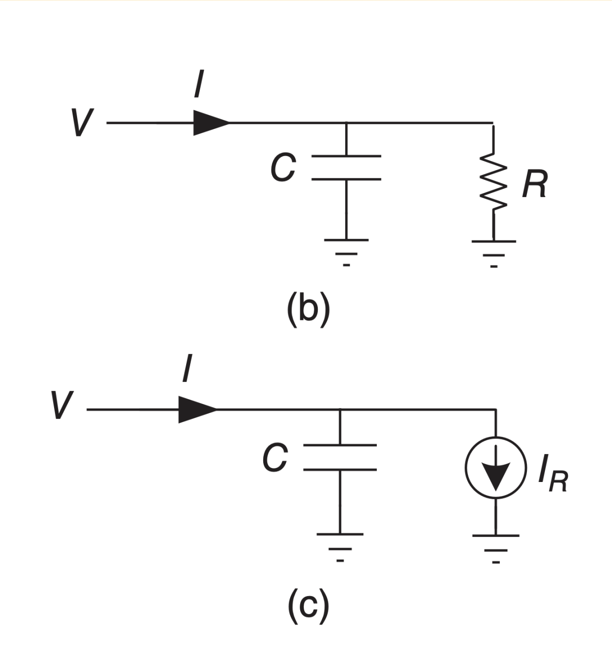




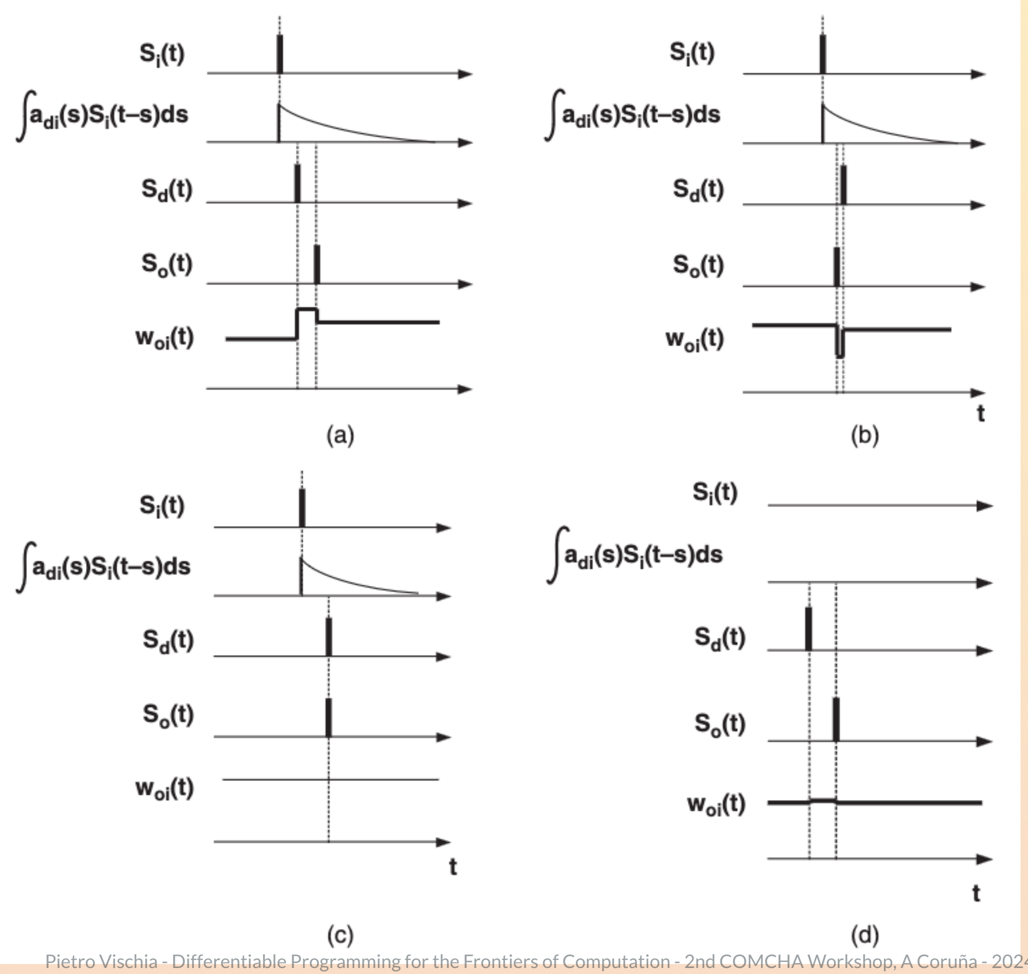
Spiking networks

- Neuronal model vastly simplified: the (Leaky) Integrate-and-fire Model

$$C\frac{dV(t)}{dt} = I - I_L$$



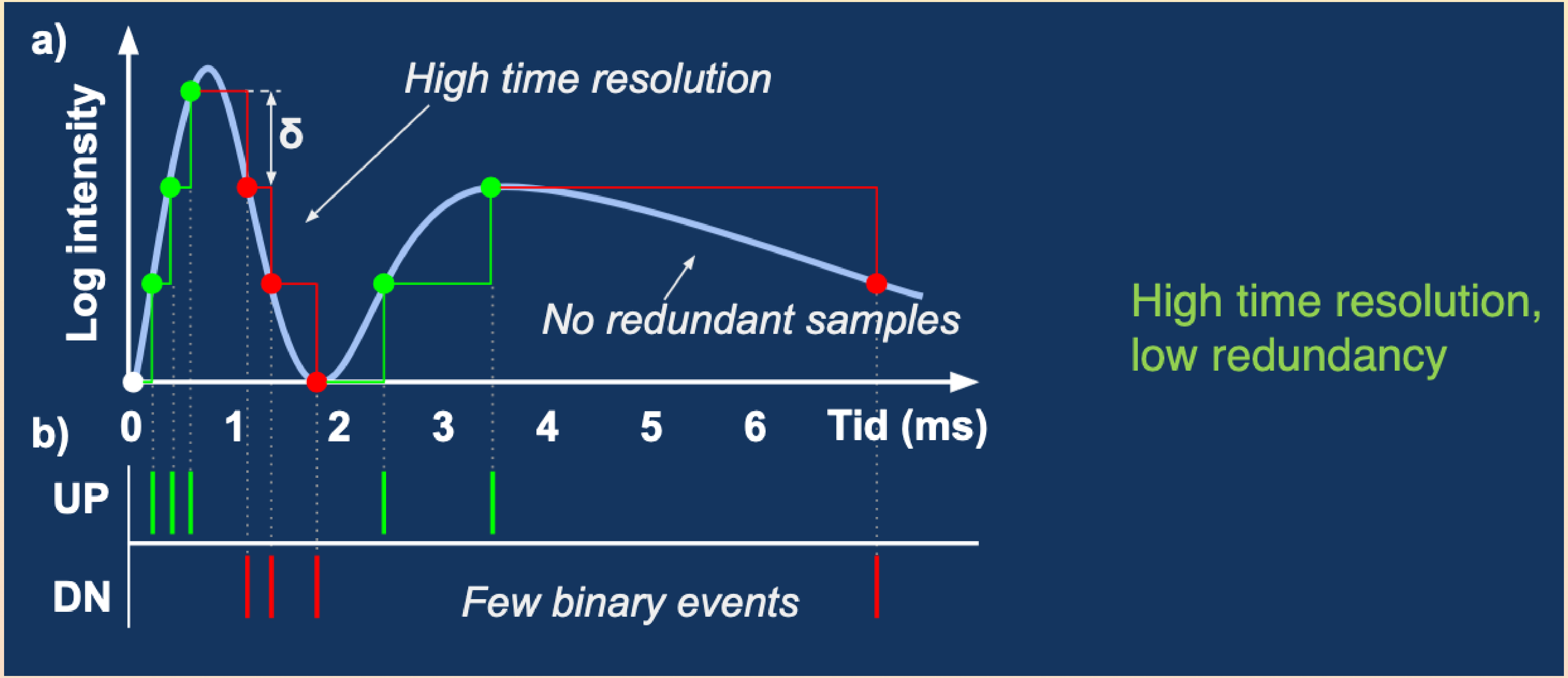
Training strategies (e.g. Remote Supervised Method)



Images from Zheng, Mazumder (Wiley, 2019)

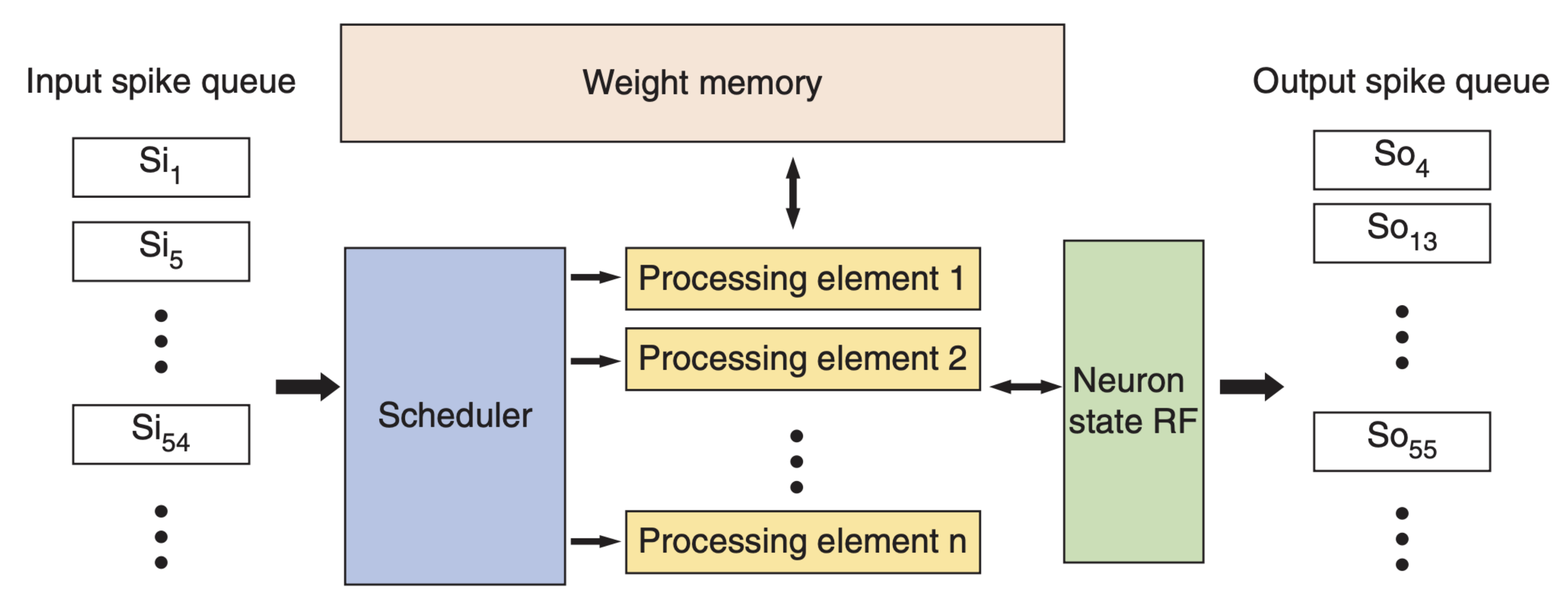
Event-driven computation

- Event-driven computations
  - "when a spike occurs, compute something": realtime operations by reducing bus width ($N_{axons} \rightarrow log_2 N_{axons}$) in CMOS or memristors
  - Work in progress on various applications
  - Q-Pix (see talk by Shion Kubota) uses same natural representation: maybe synergies?



Energy-efficient architectures

Sparser inputs $\longrightarrow$ less time and energy



	Brain	Spikey	SpiNNaker	R2600X	Intel mobile	RTX2070
Housekeeping	4.75E-11	1.37E-06	1.66E-04	4.49E-04	1.23E-04	**9.76E-07**
Resting potential	5.77E-11	**3.83E-08**	8.99E-05	4.77E-05	4.25E-05	3.63E-06
Action potential	1.96E-11	**4.39E-10**	1.04E-08	3.04E-08	4.46E-09	4.71E-09
Transmission	8.17E-15	**1.08E-11**	9.59E-09	5.82E-08	2.14E-08	3.40E-09
Single neuron	2.49E-10	**1.49E-06**	3.33E-04	9.62E-04	3.37E-04	3.18E-05
Full brain	2.15E+01	**1.29E+05**	2.87E+07	8.29E+07	2.90E+07	2.74E+06

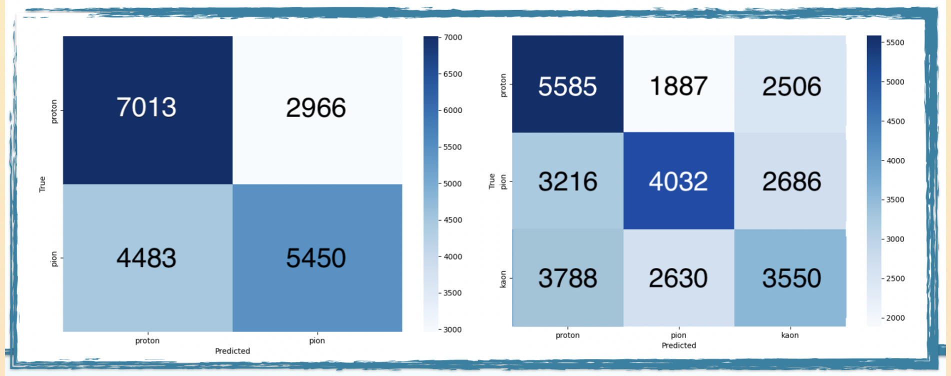
Values for the simulation of 1s of model time are reported in Joule. The single neuron and full brain estimates assume a fan-out of 2,000 synapses and a spike rate of 4Hz. R2600X: AMD Ryzen 2600X. Intel mobile: Intel Core i7-4710MQ. RTX2070: NVIDIA RTX 2070. Both CPUs are measured using a PeakTech power meter. The lowest values from simulators/emulators are highlighted in bold.

Image from Zheng, Mazumder (Wiley, 2019)

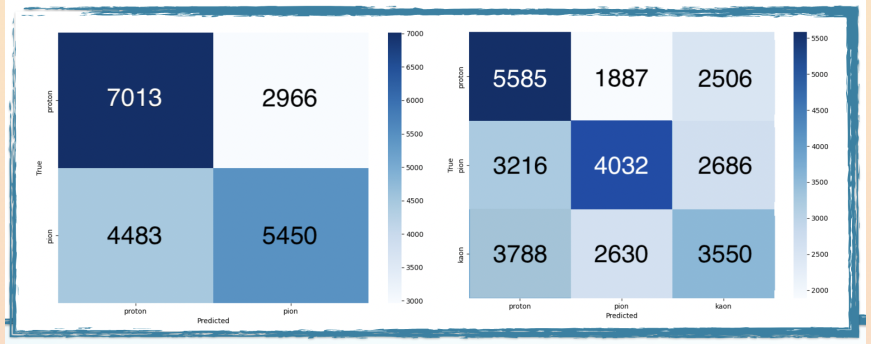
Calorimetry with AD and neuromorphic computing

- Starting work in Oviedo (also within DRD6) to expand first MODE studies

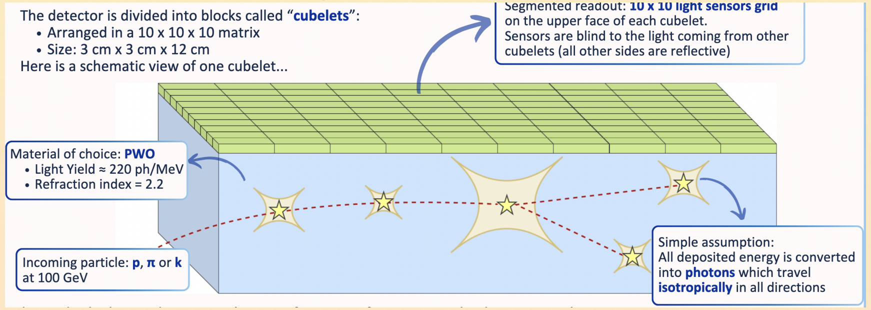
- Particle ID at high granularity
  - Towards integrated tracking-calorimeter ID
- Detailed shower profile



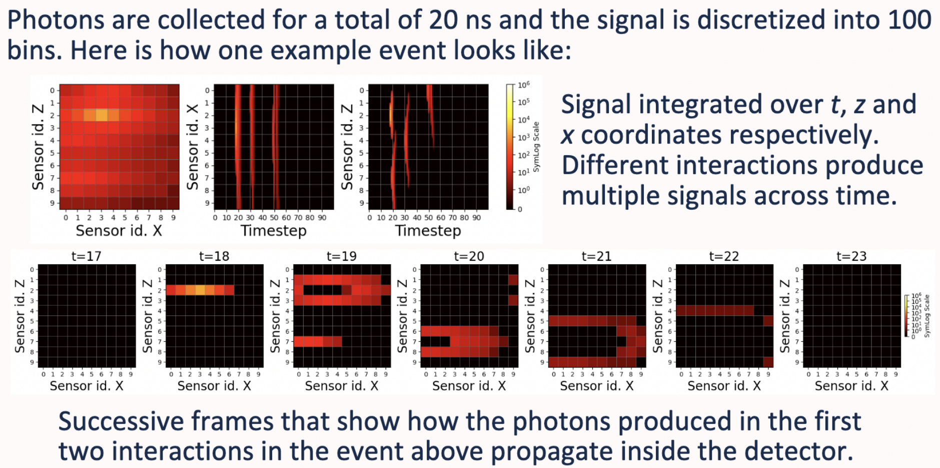
- Classification based on shower properties
  - Planned: 3D profile combining BDTs, CNNs, RNNs



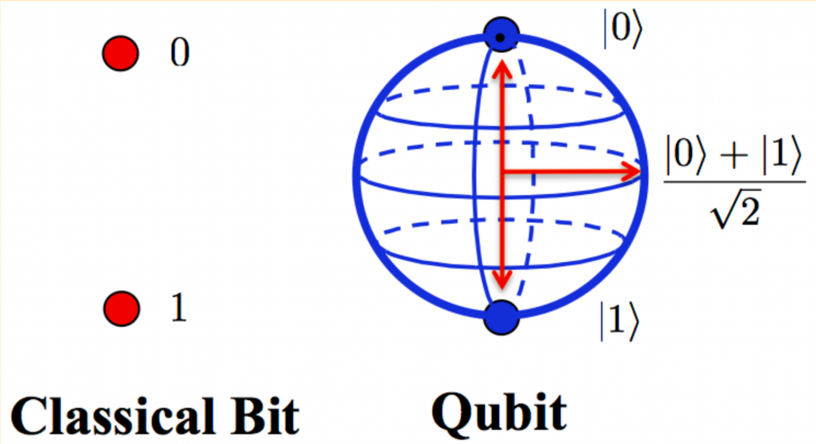
- Neuromorphic readout via network of nanowires
  - Fast, energy-efficient local computation
  - Generate informative high-level primitives



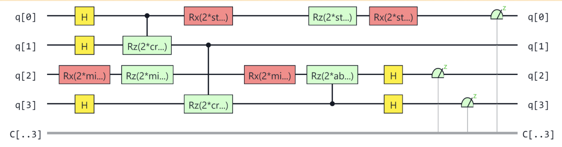
- Time evolution accessible via spiking networks

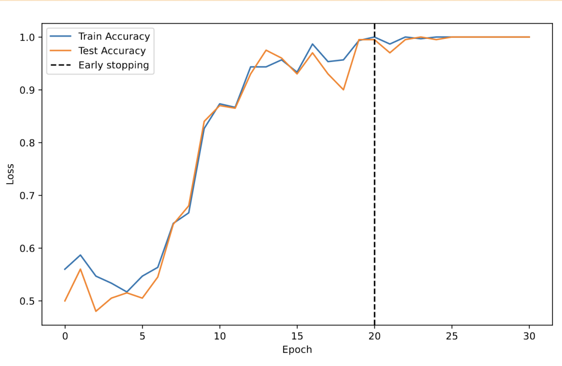


Differentiable quantum optimization

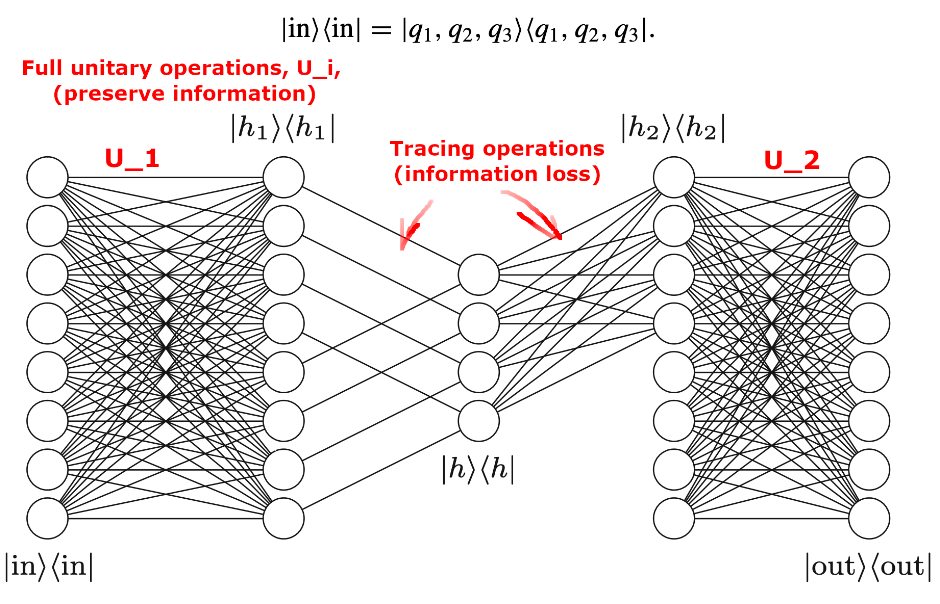


- Quantum representation can have advantages
  - Lower dimensional, and achieving high efficiency with small datasets
  - BSc thesis of Manuel Uría García (paper in preparation M.U.G. and José M. Uría (UriaXait SL)





- Natural representation of neural networks by qubit operations



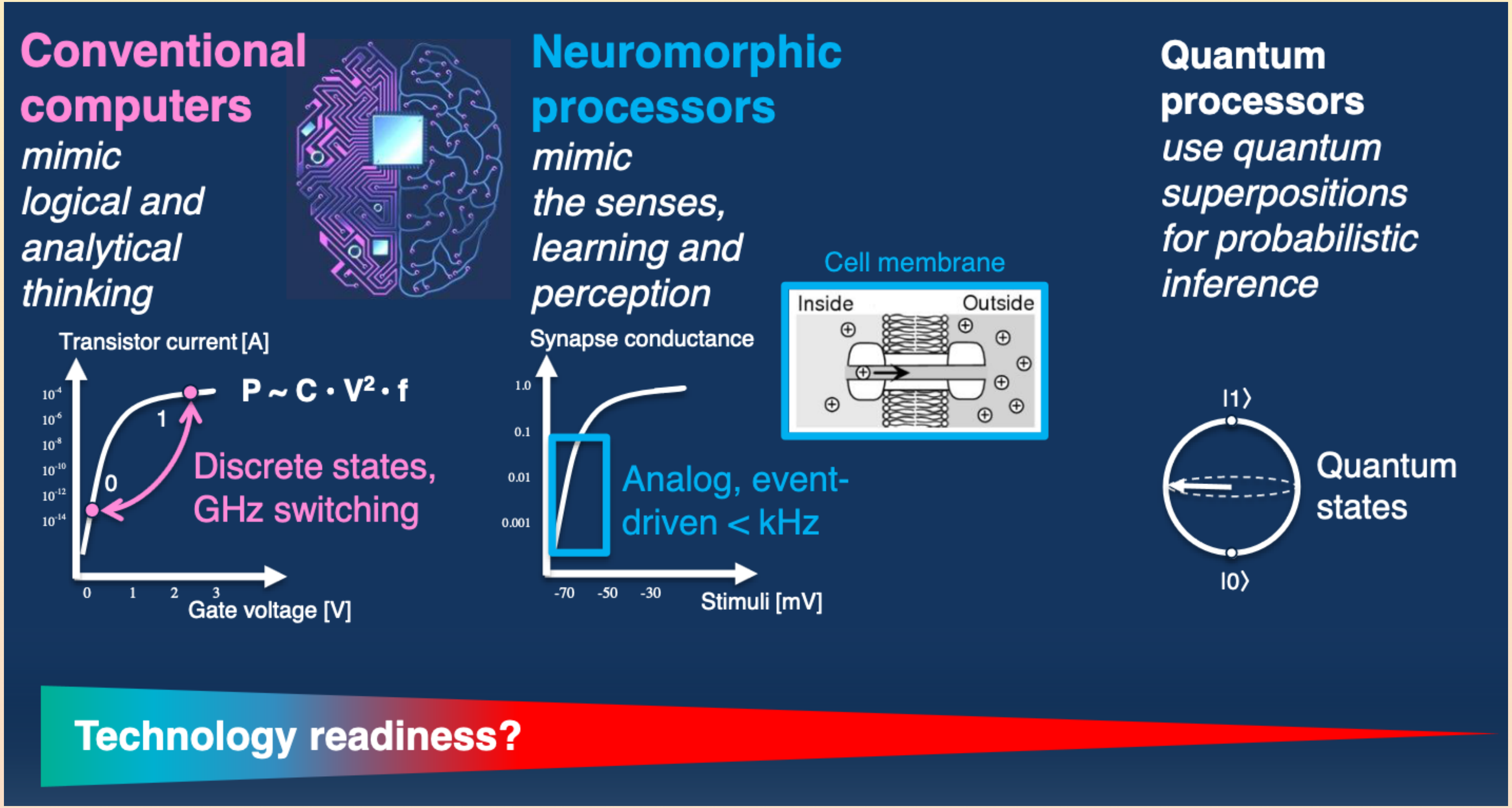
- Gradient descent exploits intrinsic analytic differentiability of quantum circuits

$$\begin{aligned}
\partial_\mu \langle\psi(x,\theta)|\sigma_z|\psi(x,\theta)\rangle &= \langle 0|\dots\partial_\mu \mathrm{e}^{-i\mu\sigma}\dots\sigma_z\dots\mathrm{e}^{i\mu\sigma}\dots|0\rangle \\
&\quad+\langle 0|\dots\mathrm{e}^{-i\mu\sigma}\dots\sigma_z\dots\partial_\mu\mathrm{e}^{i\mu\sigma}\dots|0\rangle \\
&= \langle 0|\dots(-i\sigma)\mathrm{e}^{-i\mu\sigma}\dots\sigma_z\dots\mathrm{e}^{i\mu\sigma}\dots|0\rangle \\
&\quad+\langle 0|\dots\mathrm{e}^{-i\mu\sigma}\dots\sigma_z\dots(i\sigma)\mathrm{e}^{i\mu\sigma}\dots|0\rangle \\
&= \langle 0|\dots(1-i\sigma)\mathrm{e}^{-i\mu\sigma}\dots\sigma_z\dots(1+i\sigma)\mathrm{e}^{i\mu\sigma}\dots|0\rangle \\
&\quad+\langle 0|\dots(1+i\sigma)\mathrm{e}^{-i\mu\sigma}\dots\sigma_z\dots(1-i\sigma)\mathrm{e}^{i\mu\sigma}\dots|0\rangle
\end{aligned}$$

Image by prateekvjoshi and M. Schuld, F. Petruccione (2018)

Need for new paradigma

- If you are interested in Neuromorphic computing or Quantum computing, drop me a line!



The MODE Collaboration

https://mode-collaboration.github.io/

- Joint effort
  - Particle physicists, Nuclear physicists, Astrophysicists, Computer scientists, Mathematicians
- If you are interested, join us!!!
  - Estremely loose statute, we mostly strive to talk regularly and collaborate in projects

At INFN and Università of Padova Dr. **Tommaso Dorigo**, Dr. **Pablo De Castro Manzano**, Dr. **Federica Fanzago**, Dr. **Lukas Layer**, Dr. **Giles Strong**, Dr. **Mia Tosi**, and Dr. **Hevjin Yarar**
At Université catholique de Louvain Dr. **Andrea Giammanco**, Prof. **Christophe Delaere**, and Mr. **Maxime Lagrange**
At Universidad de Oviedo and ICTEA Dr. **Pietro Vischia**
At Université Clermont Auvergne, Prof. **Julien Donini**, and Mr. **Federico Nardi** (joint with Universitá di Padova)
At the Higher School of Economics of Moscow, Prof. **Andrey Ustyuzhanin**, Dr. **Alexey Boldyrev**, Dr. **Denis Derkach**, and Dr. **Fedor Ratnikov**
At the Instituto de Física de Cantabria, Dr. **Pablo Martínez Ruíz del Árbol**
At CERN, Dr. **Sofia Vallecorsa**
At Karlsruher Institut für Technologie, Dr. **Jan Kieseler**
At University of Oxford Dr. **Atilim Gunes Baydin**
At New York University Prof. **Kyle Cranmer**
At Université de Liège Prof. **Gilles Louppe**
At GSI/FAIR Dr. **Anastasios Belias**
At HEPHY Vienna (OeAW) Dr. **Claudius Krause**
At Uppsala Universitet Prof. **Christian Glaser**
At TU-München, Prof. **Lukas Heinrich** and Mr. **Max Lamparth**
At Durham University Dr. **Patrick Stowell**
At Lebanese University Prof. **Haitham Zaraket**
At University of Kaiserslautern-Landau Mr. **Max Aehle**, Prof. **Nicolas Gauger**, Dr. **Lisa Kusch**
At University of Applied Sciences Worms Prof. **Ralf Keidel**
At Princeton University Prof. **Peter Elmer**
At University of Washington Prof. **Gordon Watts**
At SLAC Dr. **Ryan Roussel**
At Lulea University of Technology Prof. **Fredrik Sandin** and Prof. **Marcus Liwicki**
At IGFAE and Universidad de Santiago de Compostela Prof. **Xabier Cid Vidal**

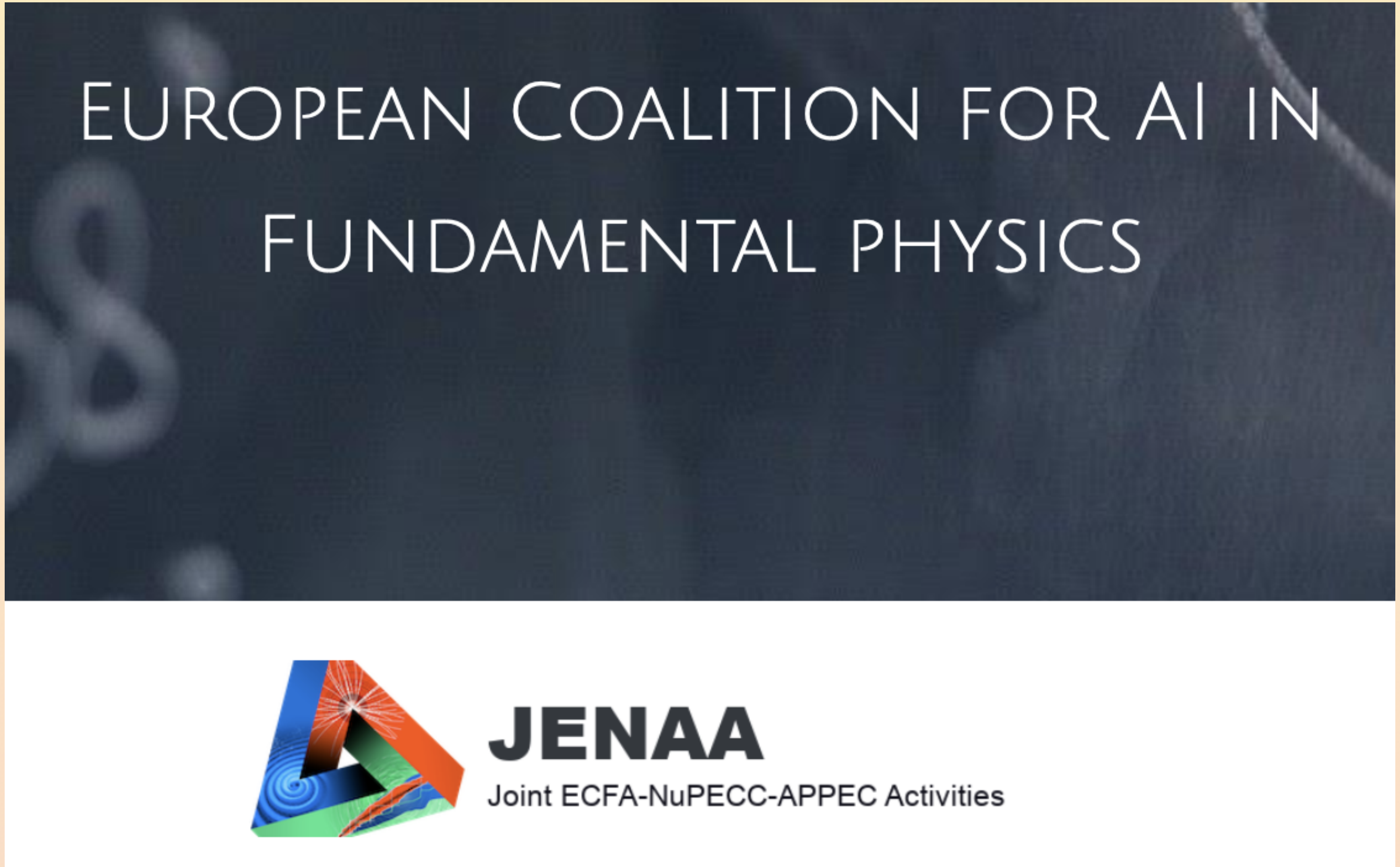
The Scientific Coordinator of the MODE Collaboration is Dr. **Tommaso Dorigo**, INFN-Sezione di Padova
The Steering Board of the MODE Collaboration includes:

- Prof. **Julien Donini**, UCA
- Dr. **Tommaso Dorigo**, INFN-PD
- Dr. **Andrea Giammanco**, UCLouvain
- Dr. **Fedor Ratnikov**, HSE
- Dr. **Pietro Vischia**, UniOvi

- Last week we had our Fourth Workshop!!!

European AI structures

- European initiative for advancing the use of AI in Fundamental Physics: https://eucaif.org
  - The First EuCaif conference took place in Amsterdam beginning of May
  - Work Package 2: Experiment Design

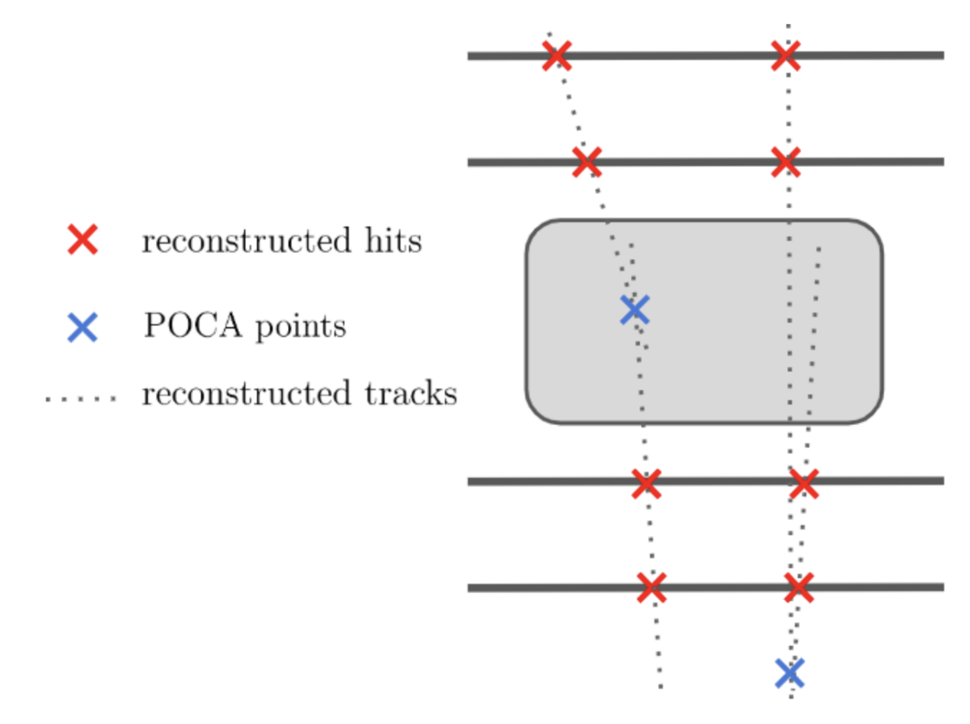


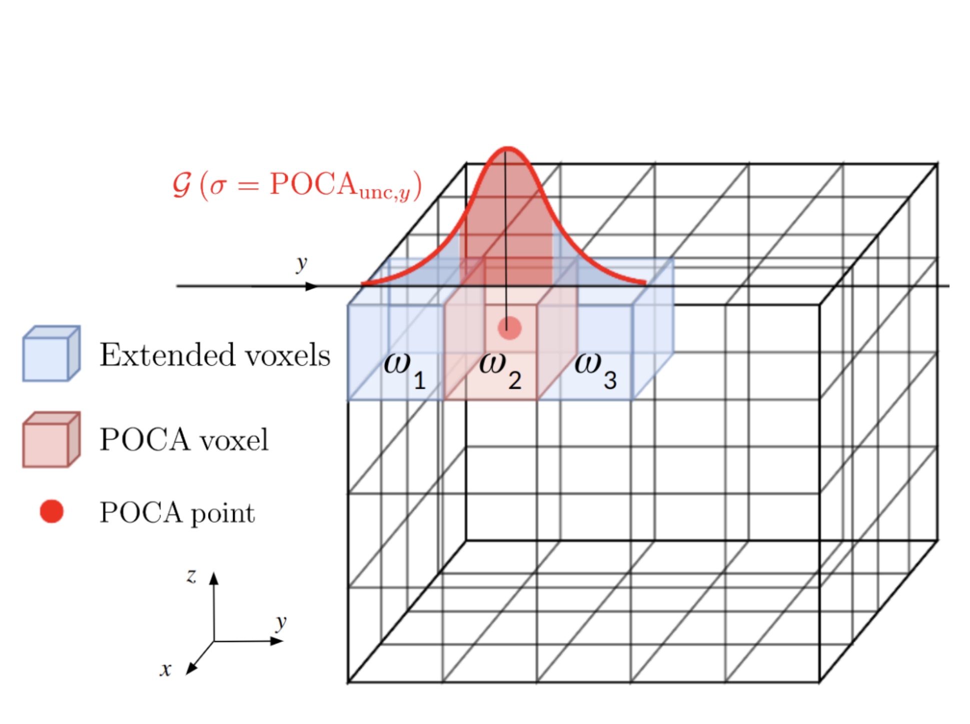
Thank you!

Backup

From POCA to extended POCA

- POCA (POint of Closest Approach) assumes one scattering in one point $\longrightarrow$ bias
  - invert model to compute X_0, then average X_0 per voxel
- Extend to nearby bins within uncertainty
  - Extrapolated from track uncertainty from analytical fit
- Assume one scattering $\longrightarrow$ bias!

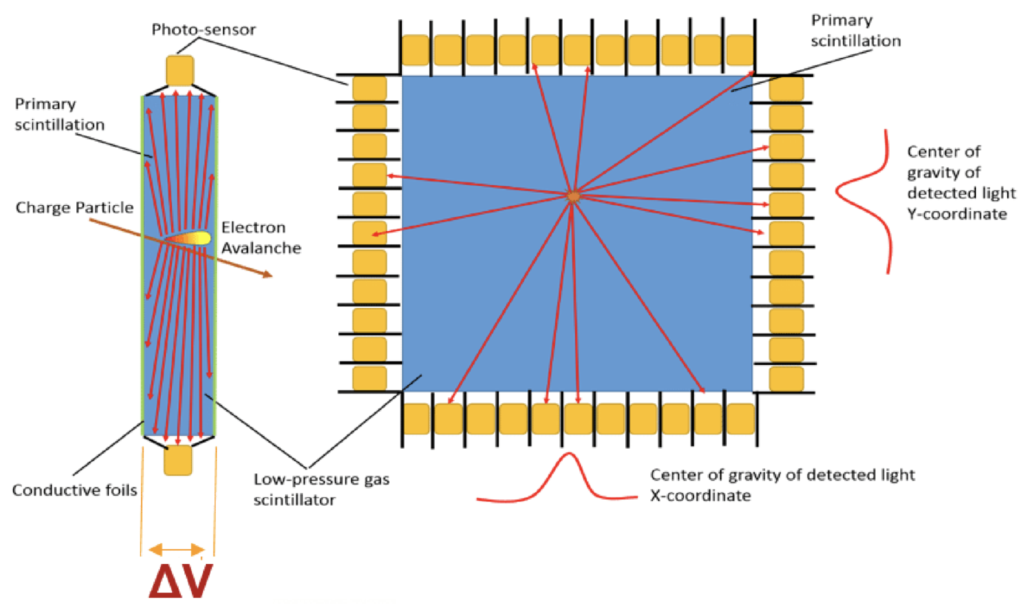


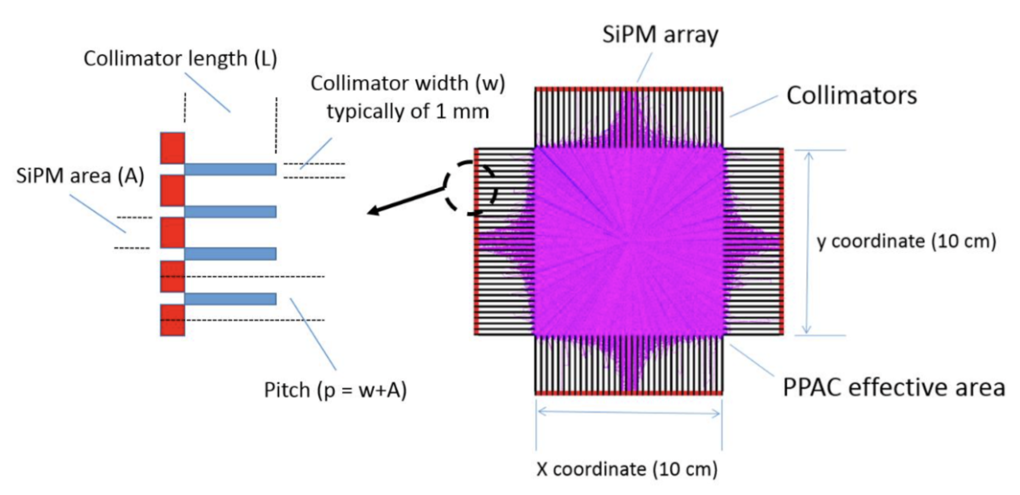


—
Figure from 10.1088/2632-2153/ad52e7

Neutron Tomography

- GEANT4 model of a $10 \times 10cm^2$ O-PPAC (Parallel-Plate Avalanche Counter with Optical Readout) from *Neutron Insights*
  - Parallel electrodes with $3mm$ gap filled with low-pressure scintillating gas mixture (CF4) with high electroluminescent light yield
  - Readout via array of small silicon photomultipliers SiPMs
- Parametric neural network surrogate of the GEANT4 simulation
  - p: higher pressure $\rightarrow$ higher electroluminescence yield (up to a threshold), but larger voltage (energy expenditure)
  - L: collimator length: tradeoff between accurate light localization (high L) and higher photon statistics (low L)

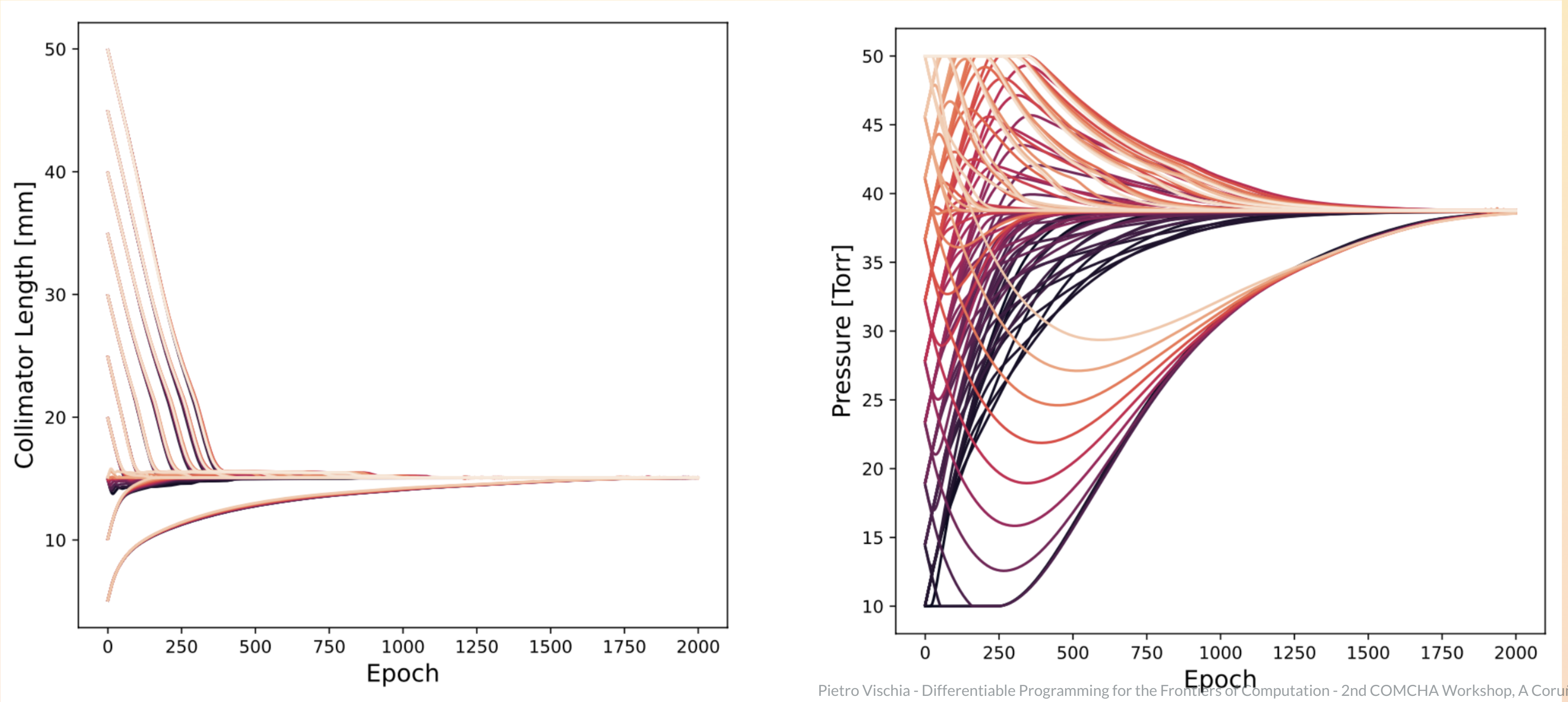




—
Images from 10.1088/1748-0221/13/10/P10006

Neutron Tomography: optimization

- Results for L give the same result as traditional studies from 10.1088/1748-0221/13/10/P10006
- Remarkably stable regardless of initial configuration
- MSc thesis of María Pereira Martínez
    - Paper in preparation (w/ M.P.M., Xabier Cid Vidal)



Neutron Tomography: optimization

- Results for L give the same result as traditional studies from 10.1088/1748-0221/13/10/P10006
- Remarkably stable regardless of initial configuration
- MSc thesis of María Pereira Martínez
  - Paper in preparation (w/ M.P.M., Xabier Cid Vidal)