

Machine Learning Optimized Design of Experiments at the frontiers of computation

methods and new perspectives

Seminar at CP3, UCLouvain, Belgium

Prof. 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

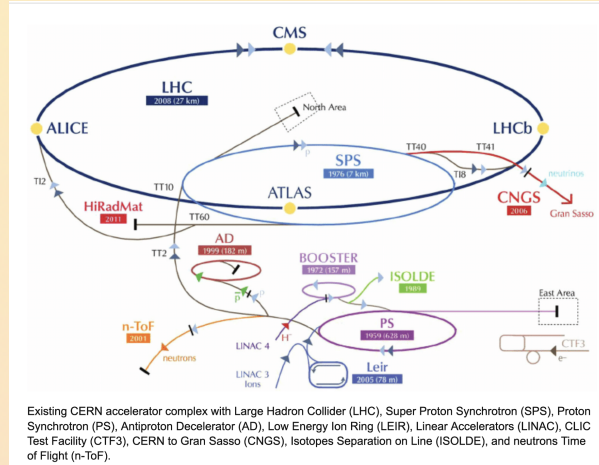
https://www.hep.uniovi.es/vischia/persistent/2025-07-01_DifferentiableProgrammingAtFrontiersComputationSeminarAtCP3UCLouvain_vischia.html

to get the version with working animations

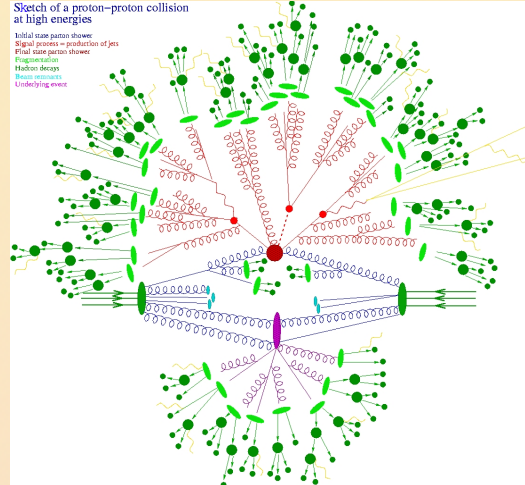
Complex experimental apparata

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

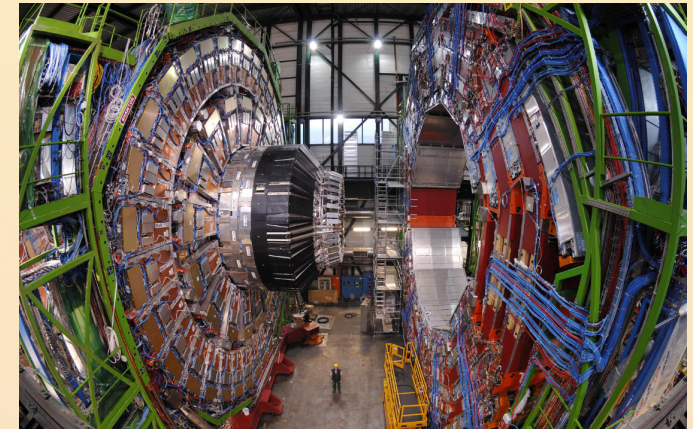
Complex accelerators



Complex phenomena



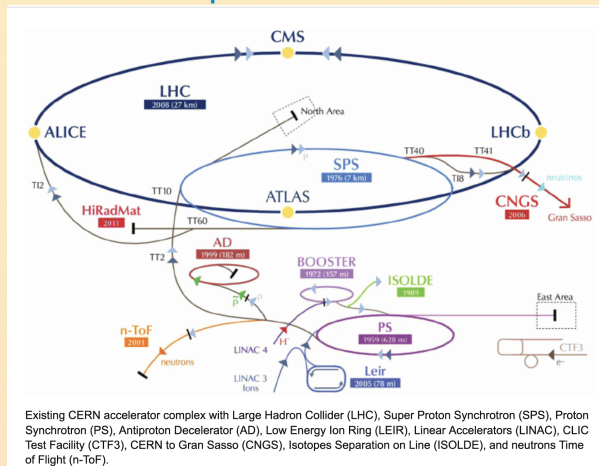
Complex experiments and reconstruction



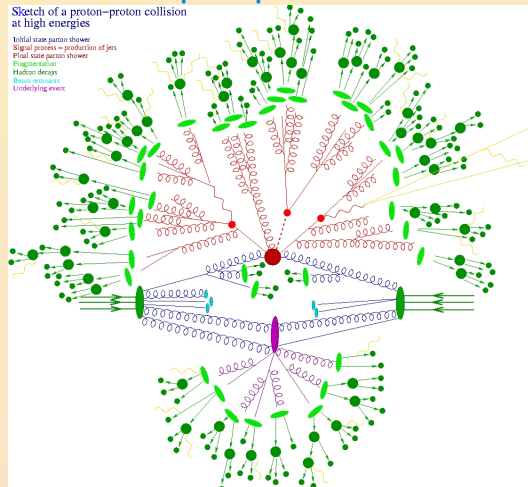
Complex experimental apparata

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

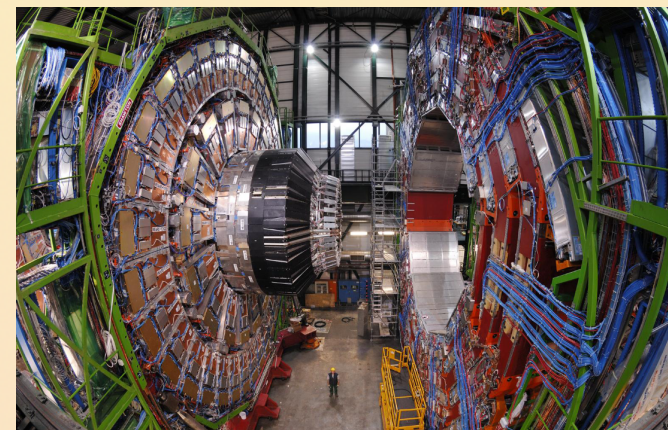
Complex accelerators



Complex phenomena



Complex experiments and reconstruction



$$P(\mathbf{x}|\alpha) = \frac{1}{A_\alpha \sigma_\alpha} \int d\Phi(y) \frac{dx_1 dx_2}{x_1 x_2 s} f(x_1) f(x_2) |\mathcal{M}_\alpha(y, x_1, x_2)|^2 W(\mathbf{x}|y) \epsilon_\alpha(y)$$

Normalization factor

We collide protons and that is a mess

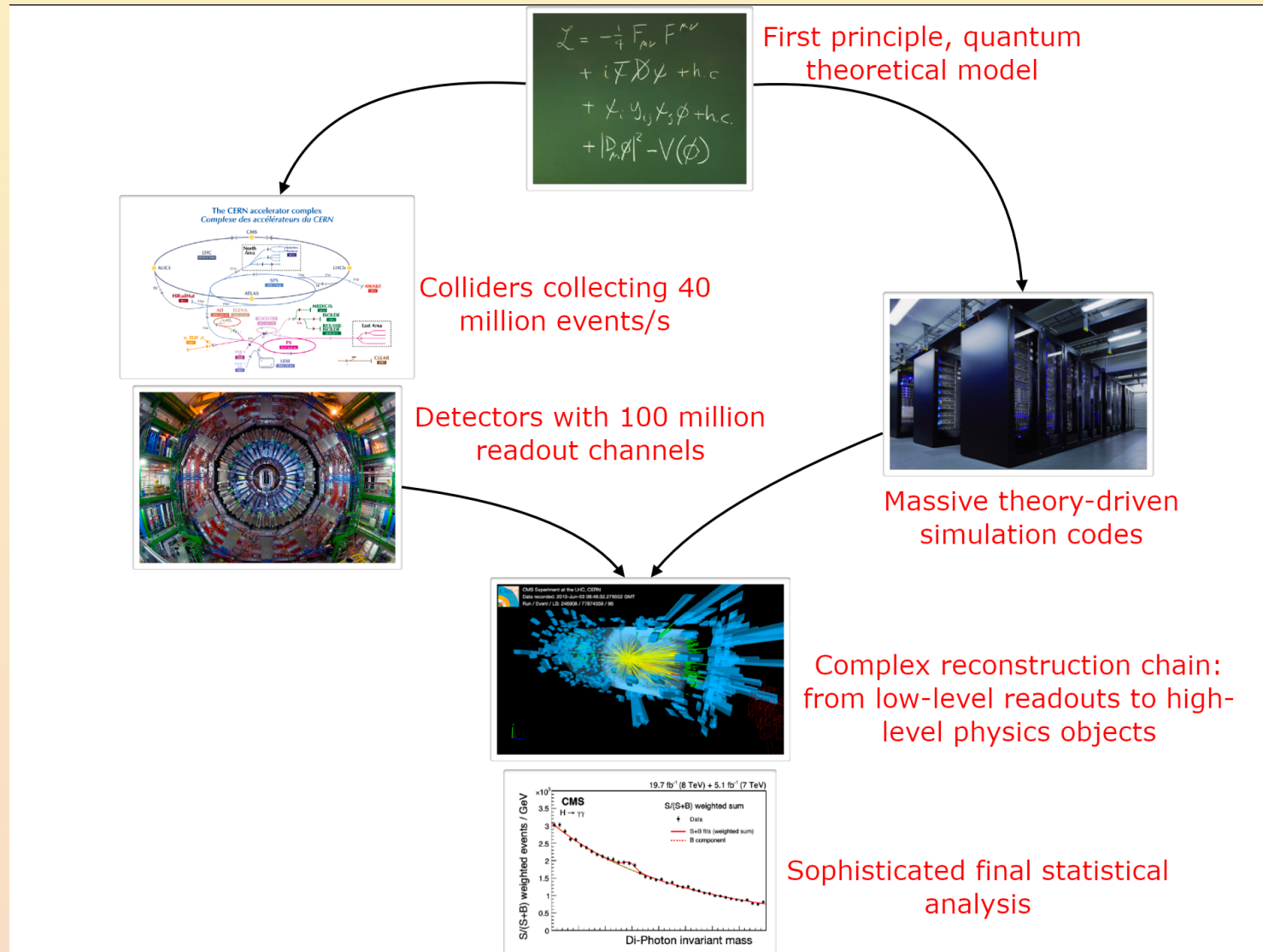
Linear operator in Hamiltonian formalism, describes the physics process

Detector response, experimental efficiencies

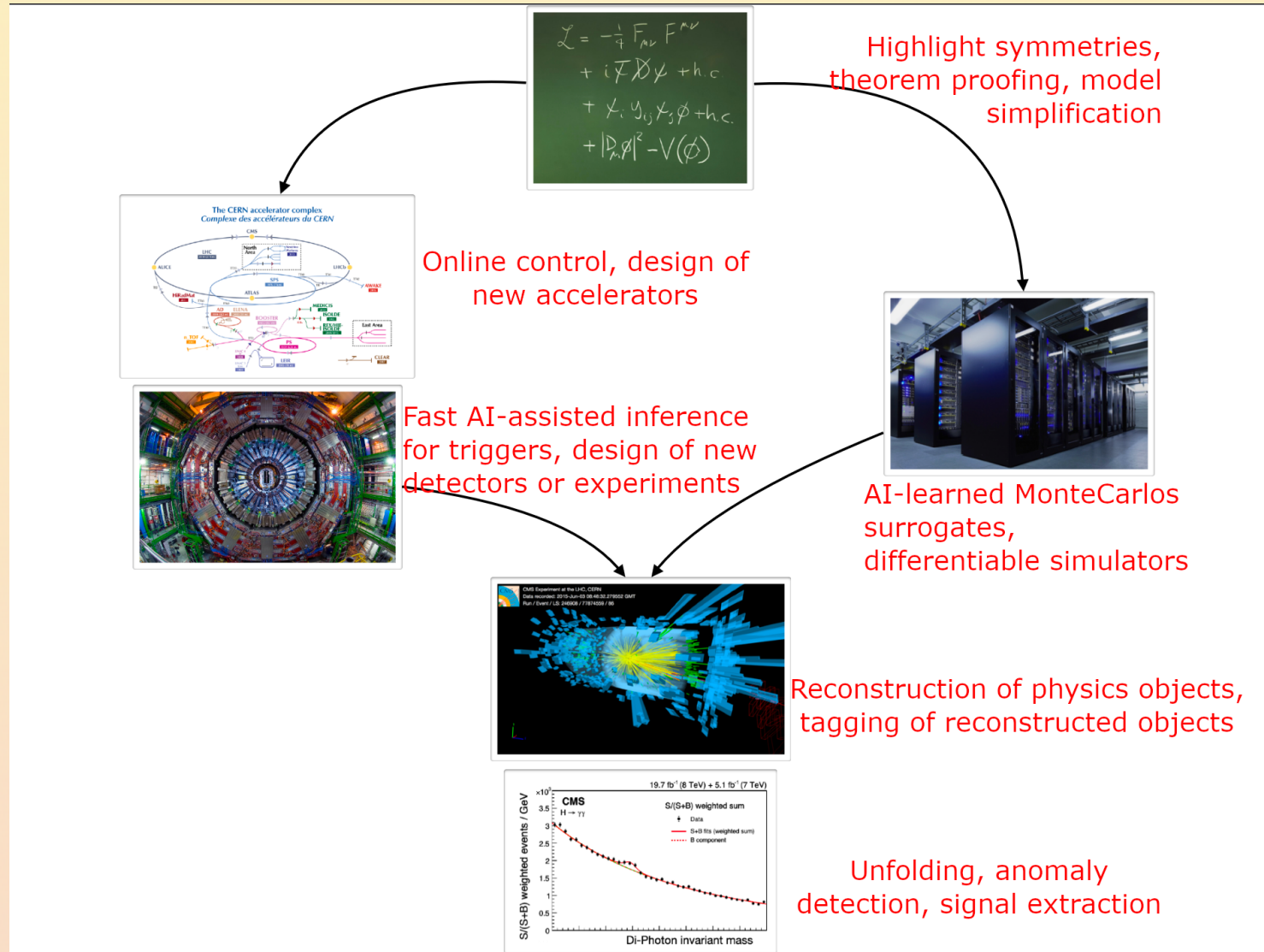
- Stochastic processes $\rightarrow$ intractable likelihood (matrix element, parton shower, detector simulation... result in latent variables)

- Costly MonteCarlo simulators to generate $x \sim p(x|\theta)$: **for each event, several thousands randomized choices**

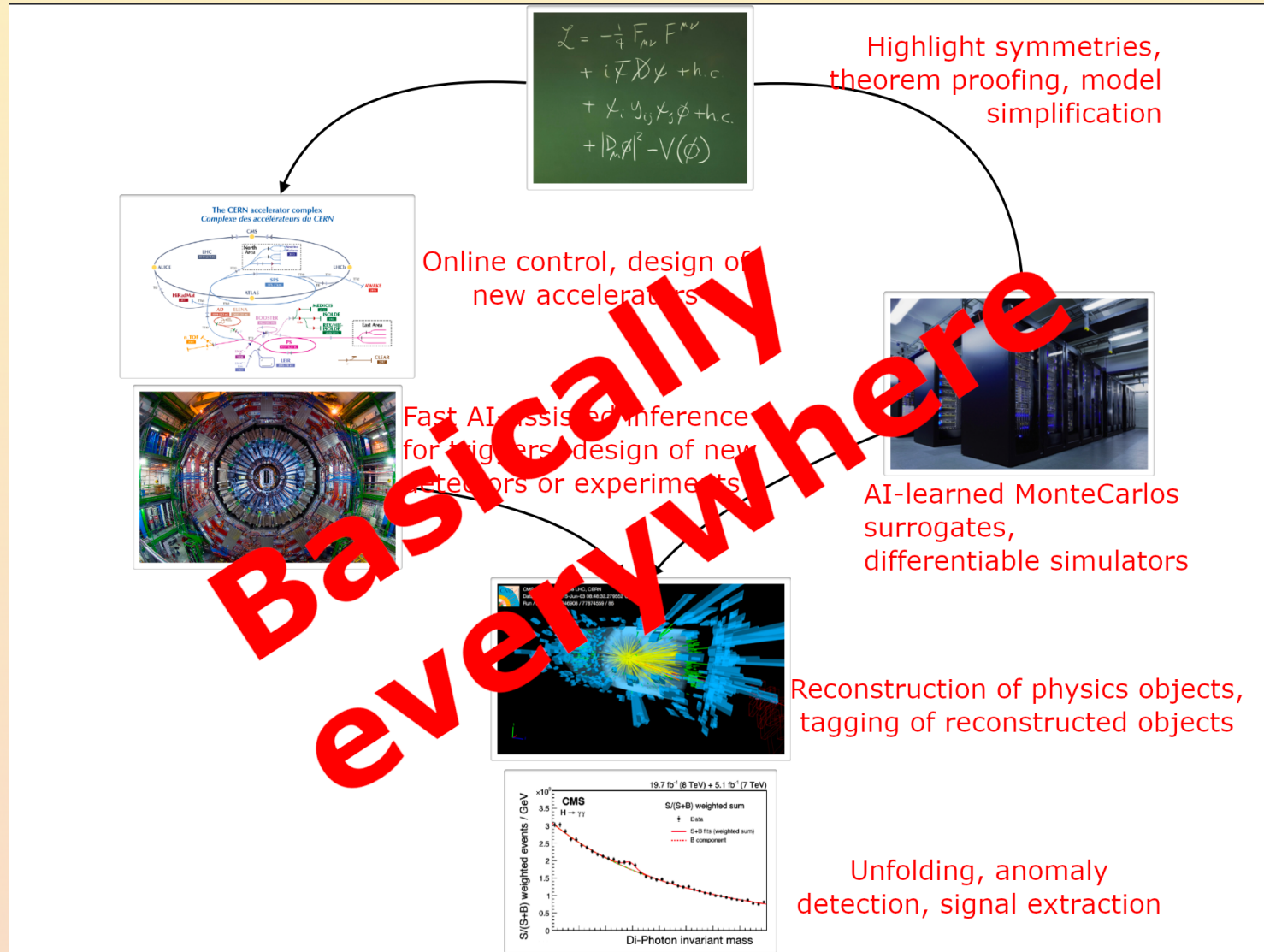
What do we do in HEP with accelerators



Where we can plug AI



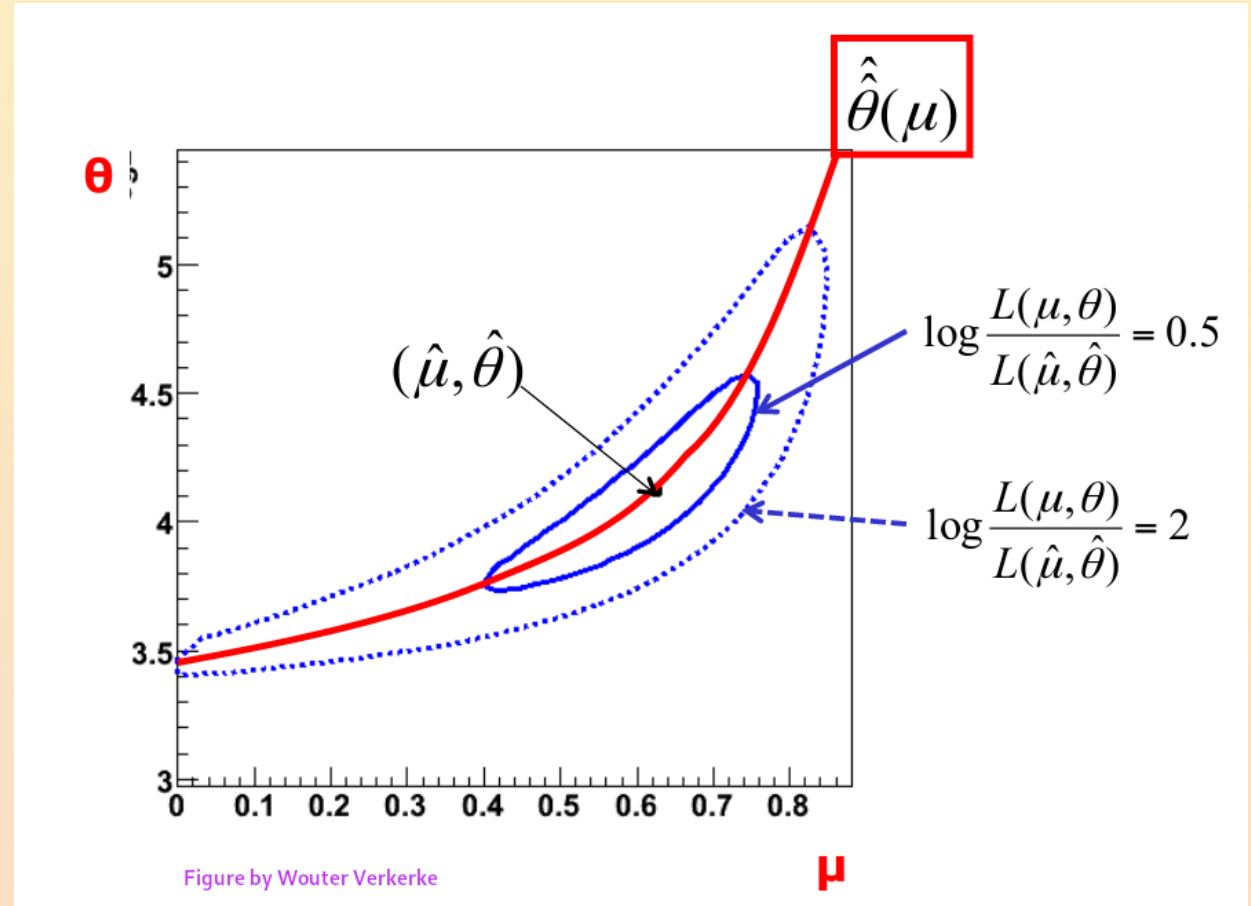
Where we can plug AI



The Likelihood Ratio:

$$\lambda(\mu) := \frac{\mathcal{L}(\mu, \hat{\theta})}{\mathcal{L}(\hat{\mu}, \hat{\theta})}$$

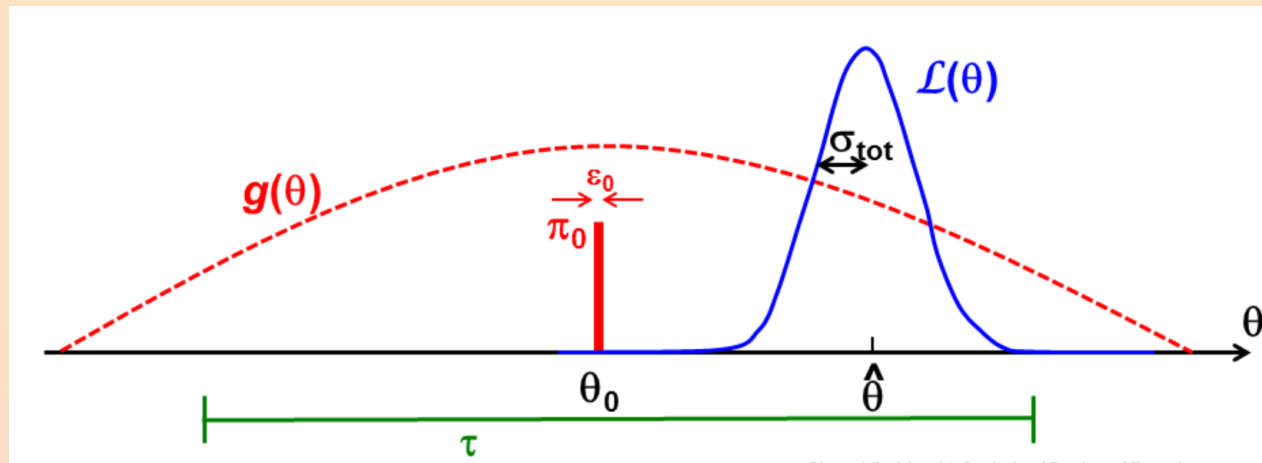
- Typical likelihood fit nowadays: $\mathcal{O}(1000)$ nuisance parameters
 - "Systematic uncertainties" (e.g. uncertainty in calibrations...)
- **Profiling**: eliminate dependence on θ by taking conditional MLEs
 - Bayesians marginalize [Demortier, 2002](#)



- $\lambda(\mu)$ distribution by toy data, or use Wilks theorem: $\lambda(\mu) \sim \exp\left[-\frac{1}{2}\chi^2\right] \left(1 + \mathcal{O}\left(\frac{1}{\sqrt{N}}\right)\right)$ under some regularity conditions

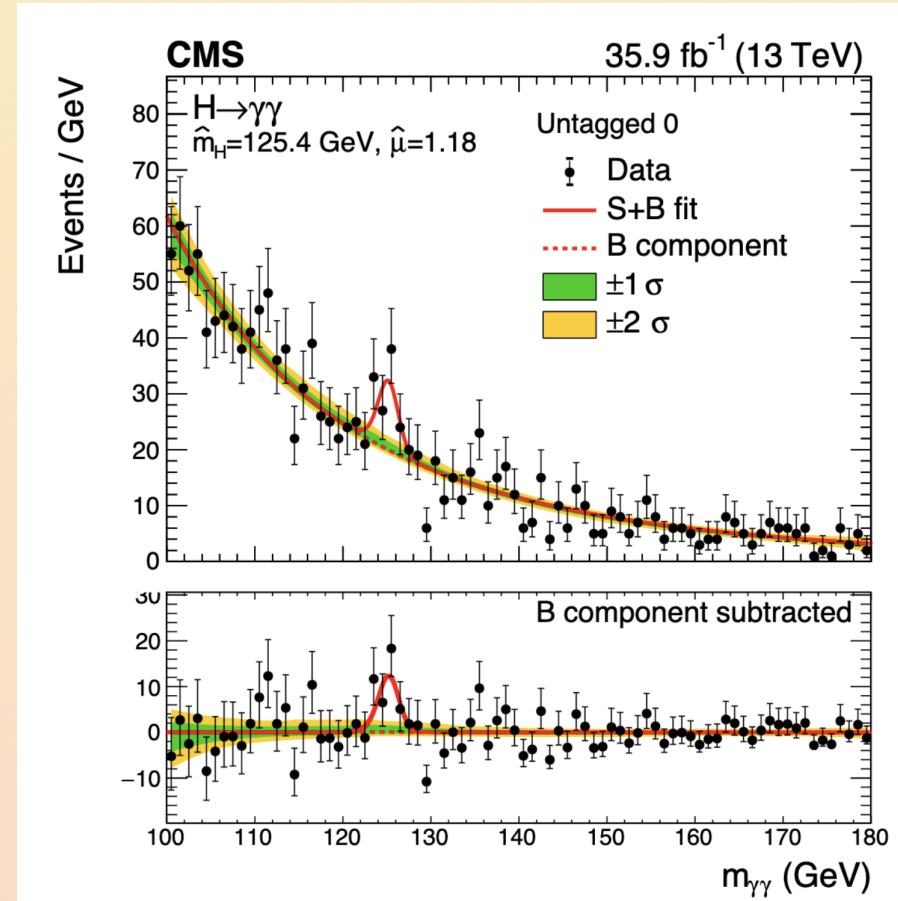
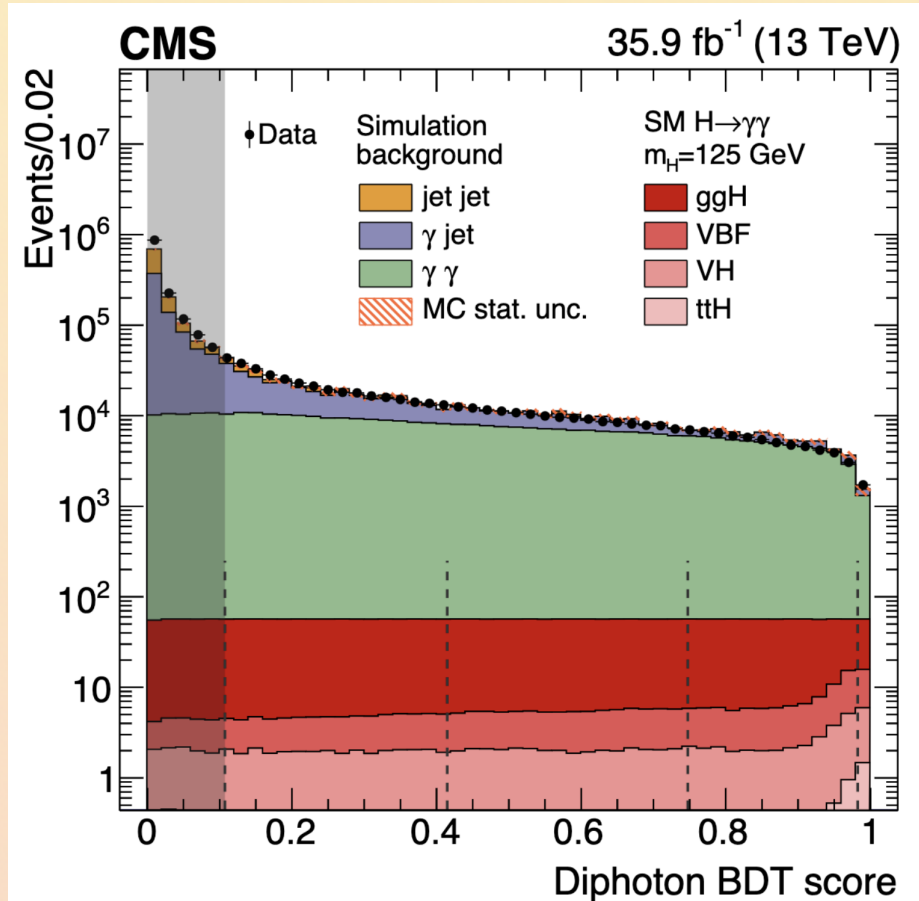
Our point null is really point null the Jeffreys-Lindley paradox

- Data X (N data sampled from $f(x|\theta)$)
 - $H_0: \theta = \theta_0$. Prior: π_0 (non-zero for point mass, Dirac's δ , counting measure)
 - $H_1: \theta \neq \theta_0$. Prior: $\pi_1 = 1 - \pi_0$ (usual Lebesgue measure)
- Conditional on H_1 being true: -Prior probability density $g(\theta)$ -If $f(x|\theta) \sim \text{Gaus}(\theta, \sigma^2)$, then the sample mean $\bar{X} \sim \text{Gaus}(\theta, \sigma_{\text{tot}} = \sigma/\sqrt{N})$
- Likelihood ratio of H_0 to best fit for $H_1: \lambda = \frac{\mathcal{L}(\theta_0)}{\mathcal{L}(\hat{\theta})} = \exp(-Z^2/2) \propto \frac{\sigma_{\text{tot}}}{\tau} B_{01}; Z := \frac{\hat{\theta} - \theta_0}{\sigma_{\text{tot}}}$
 - λ disfavors the null hypothesis for large significances (small p-values), independent of sample size
 - B_{01} includes σ_{tot}/τ (Ockham Factor, penalizing H_1 for imprecise determination of θ), sample dependent!
- **For arbitrarily large Z (small p-values), λ disfavors H_0 , while there is always a N for which B_{01} favours H_0 over H_1**

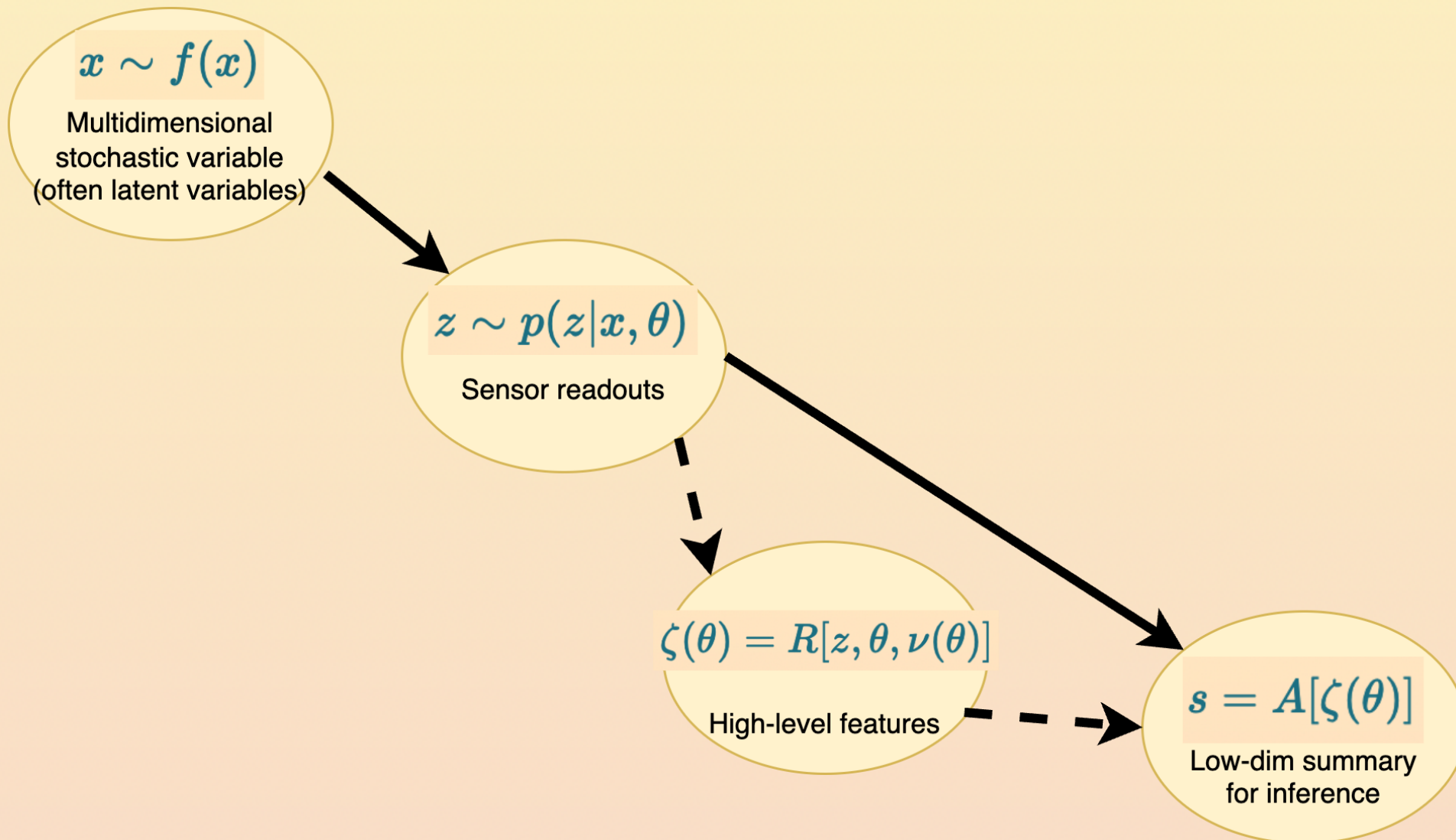


Can we go beyond histograms?

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

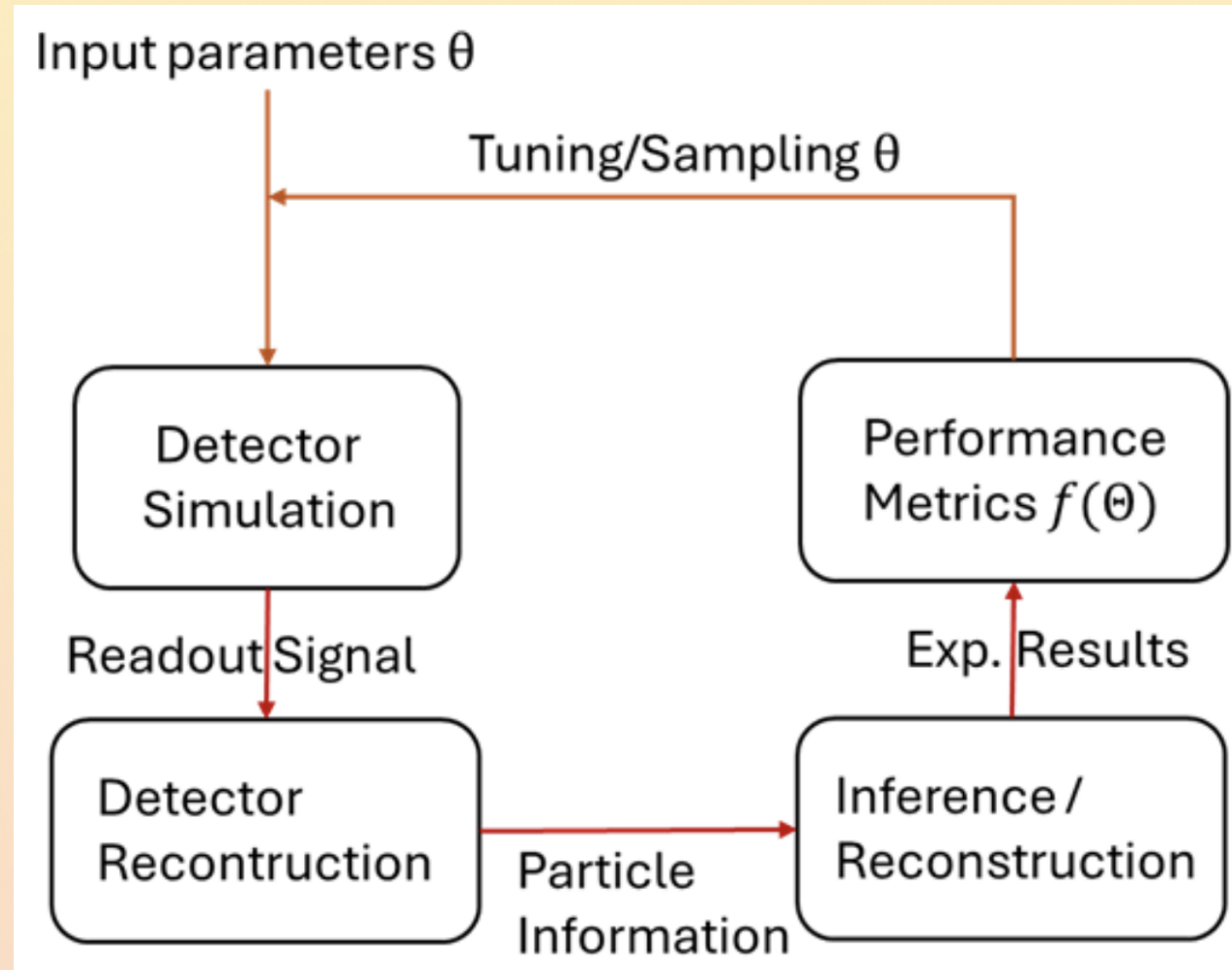


Typical analysis pipeline from the point of view of experiment design



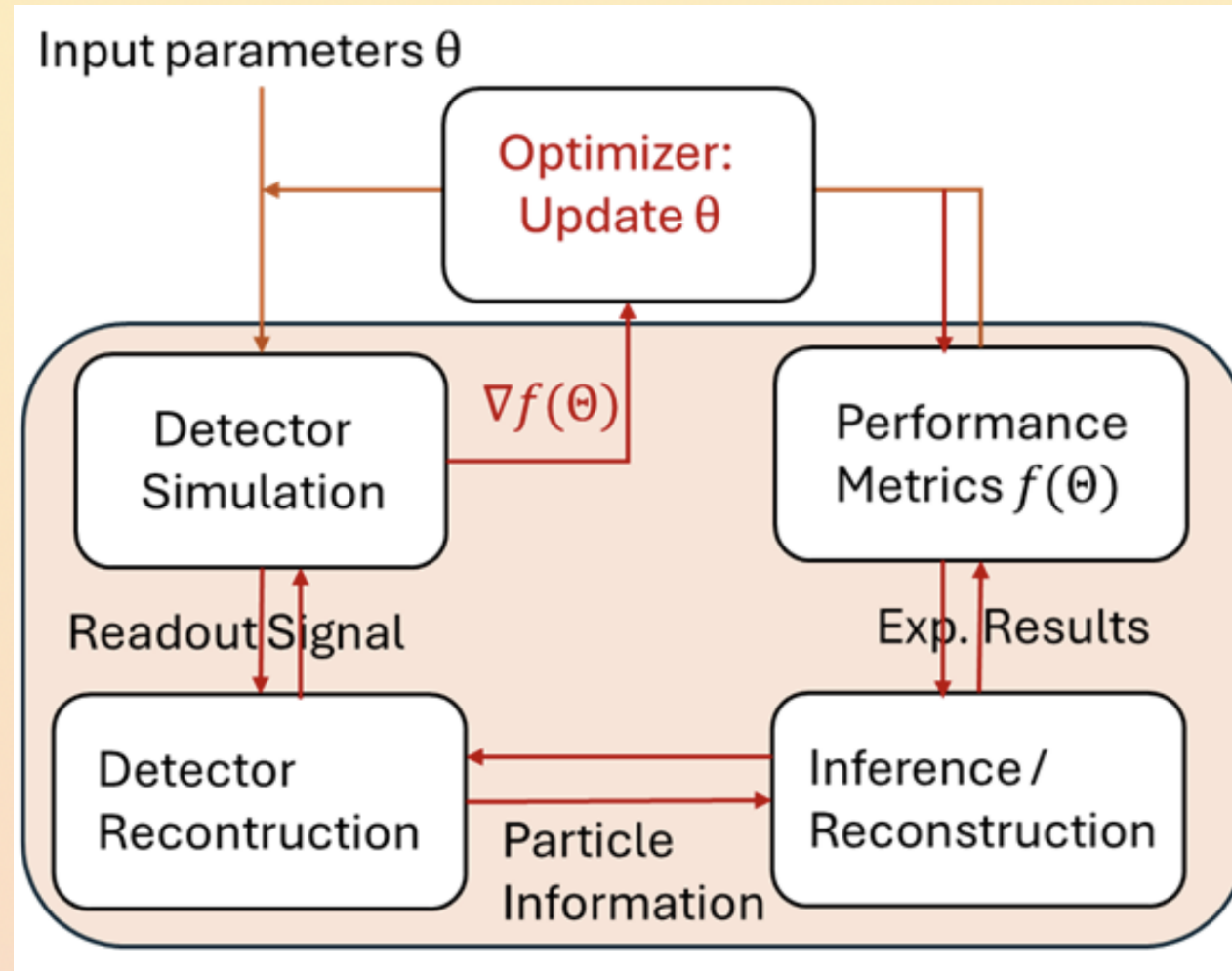
Traditional experiment/detector design

- Manual or sampled design process is inefficient and sometimes even unfeasible



AI-assisted experiment/detector design

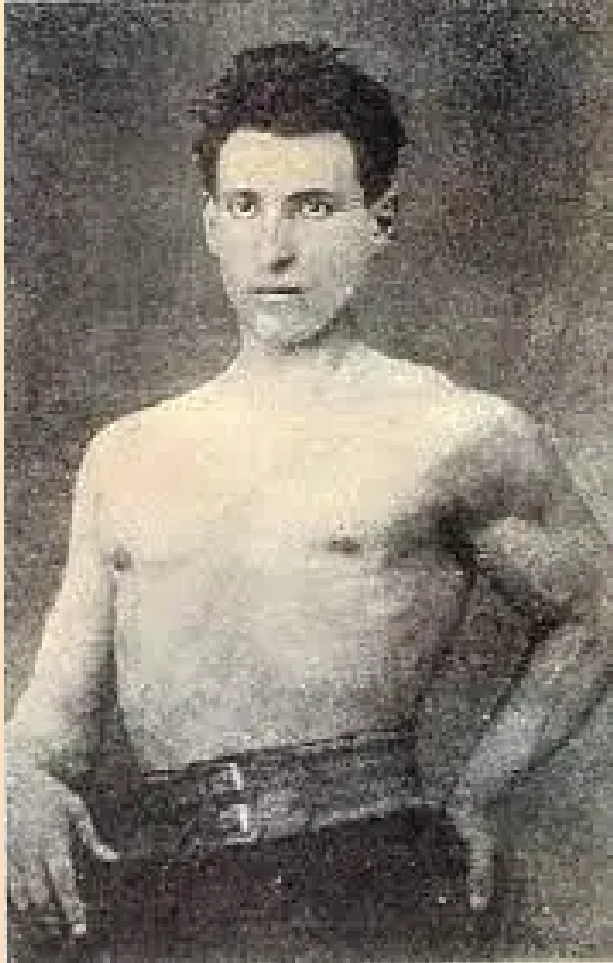
- Optimize a cost function's multidimensional landscape via gradient or reinforcement learning methods



Santiago Ramón y Cajal

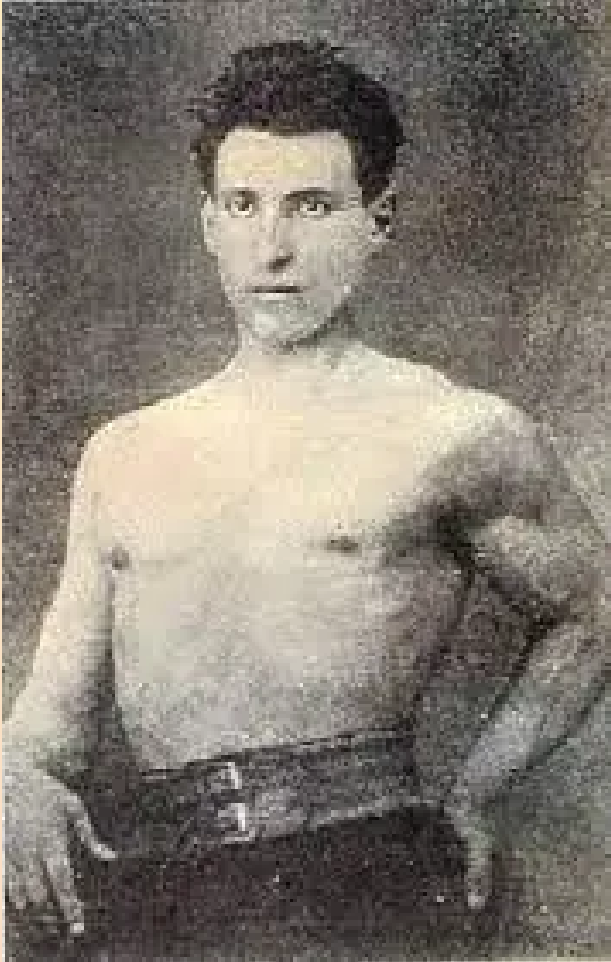
First become a camorrista and a bodybuilder...

*"El prurito de lucir el esfuerzo de mi brazo me arrastró más de una vez, contra mi temperamento nativamente bonachón, a parecer camorrista y hasta agresivo."*¹

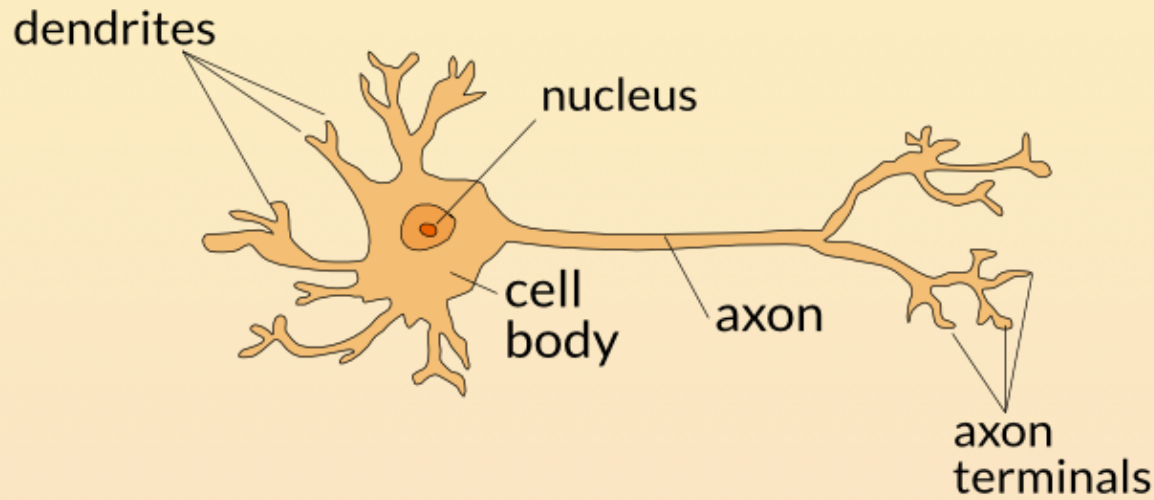


Santiago Ramón y Cajal ... then get the Nobel Prize (1906, in Physiology or Medicine)

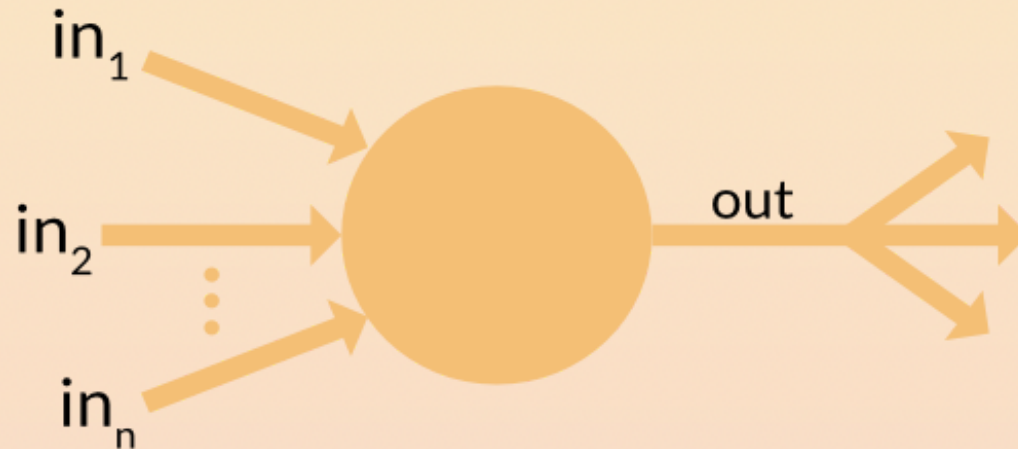
*"El prurito de lucir el esfuerzo de mi brazo me arrastró más de una vez, contra mi temperamento nativamente bonachón, a parecer camorrista y hasta agresivo."*¹



From Neurons to Perceptrons



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

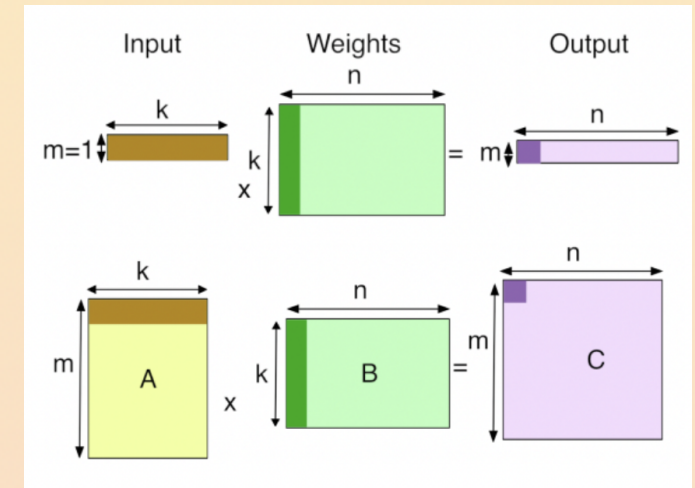


$$y = f\left(b_i + \sum w_i x_i\right)$$

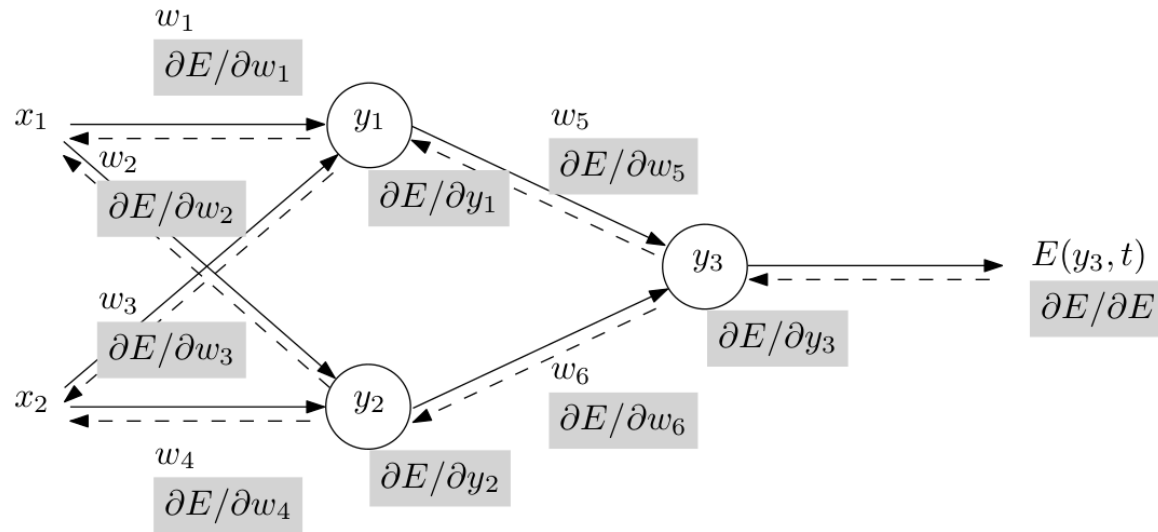
Empirical Risk Minimization Vapnik, '90s

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

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



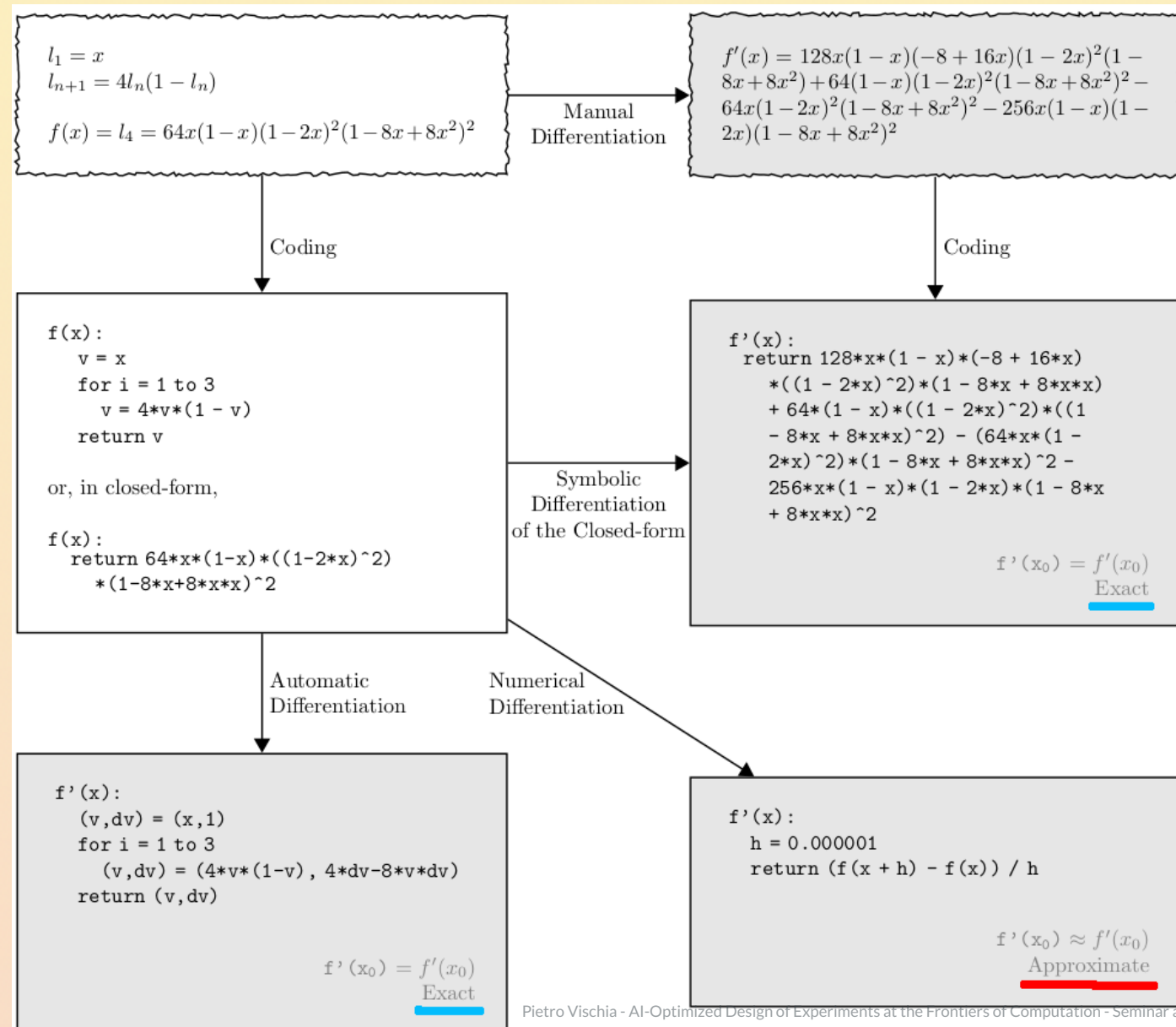
(a) Forward pass



(b) Backward pass

Derive

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



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 $f: \mathbb{R}^n \rightarrow \mathbb{R}^m$
 - $\mathcal{O}(n \text{ time}(f))$
 - $\mathcal{O}(m \text{ time}(f))$

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

<i>Fwd Primal Trace</i> Atomic operation	Value in (1, 2)	<i>Fwd Tangent Trace (set $\dot{x}_0 = 1$ to compute $\frac{\partial y}{\partial x_0}$)</i> 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 $1 \times 0.9093 + 1 \times 0$ 0 $3 \times 0 \times 4$ $2 + 0.9093 + 0$
$y = v_6$	10.9093	$\dot{y} = \dot{v}_6$	2.9093

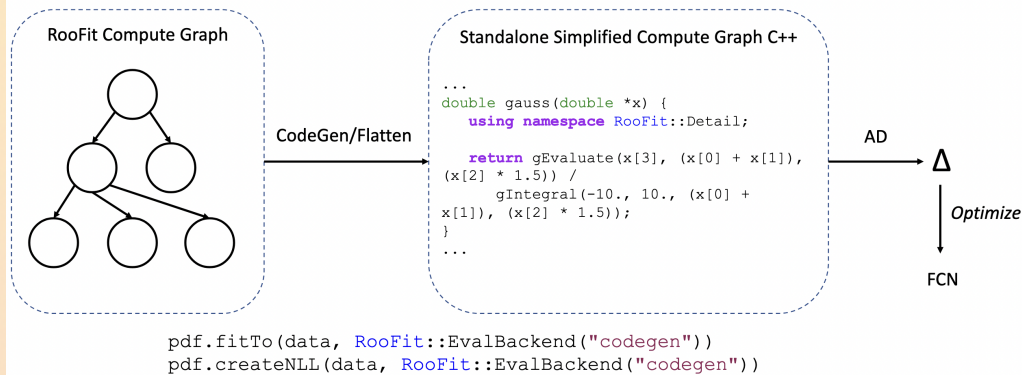
<i>Fwd Primal Trace</i> Atomic operation	Value in (1, 2)	<i>Rev Adjoint Trace (set $\bar{y} = 1$ to compute $\frac{\partial v}{\partial y}$)</i> Atomic operation	Value in (1, 2)
$v_0 = x_0$ $v_1 = x_1$	1 2	$\bar{x}_0 = \bar{v}_0$ $\bar{x}_1 = \bar{v}_1$	2.9093 11.5839
$v_2 = 2v_0$ $v_3 = \sin(v_1)$ $v_4 = v_0 v_3$ $v_5 = v_1^3$ $v_6 = v_2 + v_4 + v_5$	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

Replace numerical minimization (e.g. Minuit in RooFit)

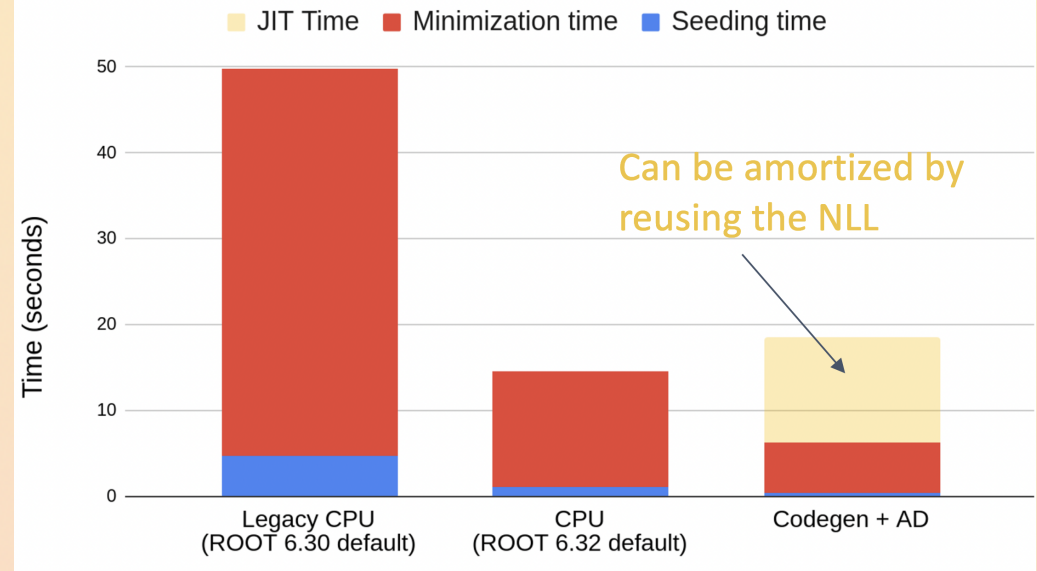
- Clang/LLVM plugin run at compilation time
- Produces C++ code (readable, explainable)

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

Clad as RooFit's AD Engine



CMS Open Data Higgs Model - single minimization



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 program, 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

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

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.

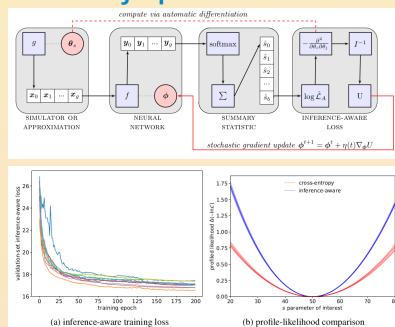
In typical reinforcement learning situations, the environment is not *a priori* represented in a differentiable form. So the main reason for building connectionist world models in the style above is to 'make the world differentiable'. Thus even *program inputs* can become differentiable with respect 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

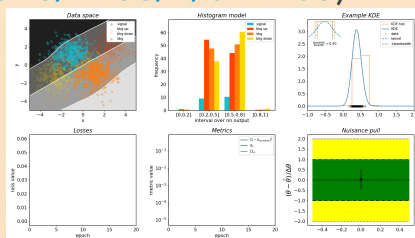
Lot of efforts in recent years!

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](https://arxiv.org/abs/10.1016/j.cpc.2019.06.007))



neos ([10.1088/1742-6596/2438/1/012105](https://arxiv.org/abs/10.1088/1742-6596/2438/1/012105))



Local generative surrogates ([2002.04632](https://arxiv.org/abs/2002.04632))

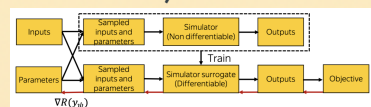


Figure 1: Simulation and surrogate training. *Black*: forward propagation. *Red*: error backpropagation.

Cheetah ([2401.05815](https://arxiv.org/abs/2401.05815))



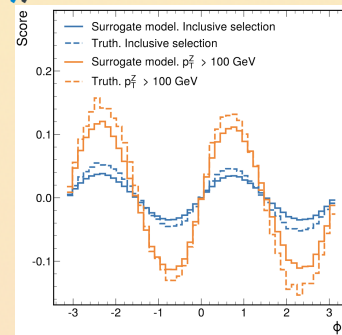
Gradient-based Tuning

Transverse beam tuning at ARES

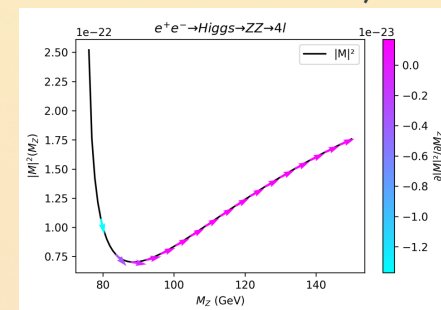
- Tune magnet settings or lattice parameters using the **gradient of the beam dynamics model** computed through automatic differentiation.
- Seamless integration with PyTorch tools tuning neural networks.
- Becomes very useful for **high-dimensional tuning tasks** (see neural network training).



CP-optimal observables ([2405.13524](https://arxiv.org/abs/2405.13524), Cruz et al. (P.V.))



MadJax ([10.1088/1742-6596/2438/1/012137](https://arxiv.org/abs/10.1088/1742-6596/2438/1/012137))



Differentiable showers in GEANT4 ([2405.07944](https://arxiv.org/abs/2405.07944))

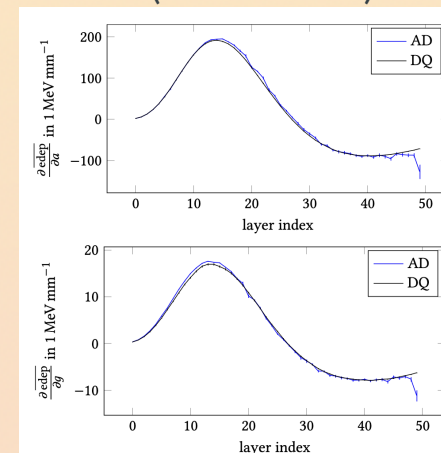


Figure 10: Algorithmic derivative of the edep with respect to the absorber thickness a (top) and gap thickness g (bottom).

SWGO optimization ([2310.01857](https://arxiv.org/abs/2310.01857))

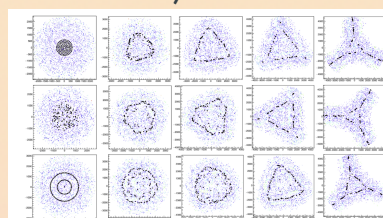
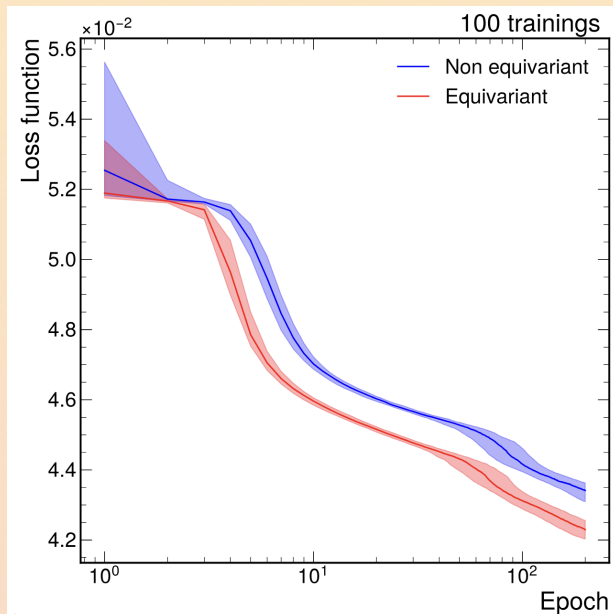


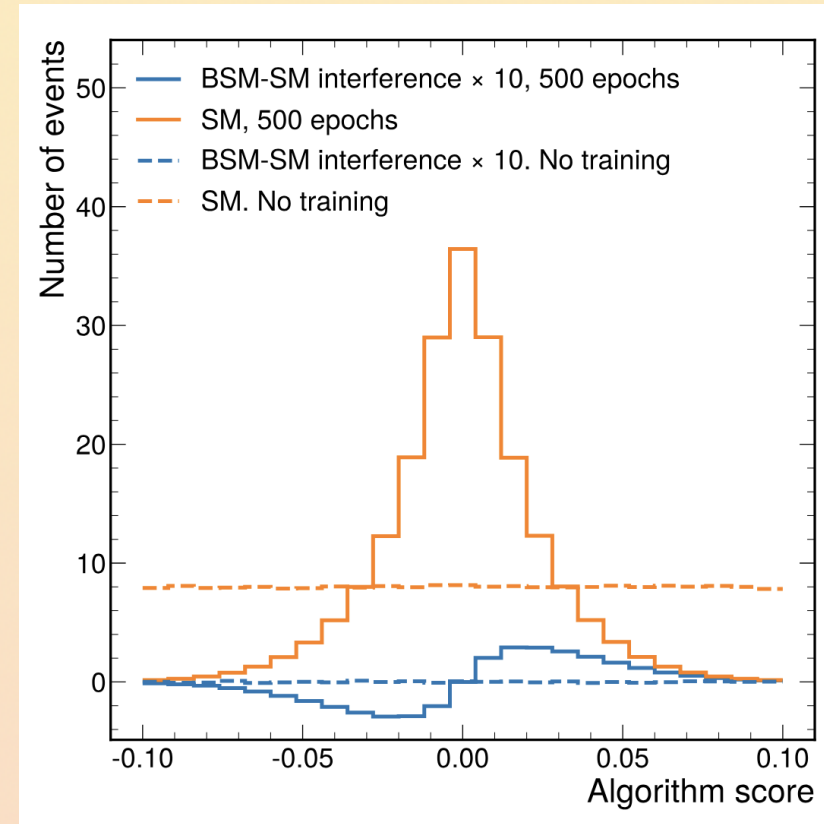
Figure 11: Convergence of three initial layouts (top to bottom: packed ball, wide random ball, two annuli) during a 500-epochs training. From left to right, the configurations of 196 units (129 in the bottom one) are shown at epoch 1, 50, 150, 300, 500. See the text for more detail.

Inductive bias for CP S. Sanchez, M. Kolosova, C. Ramón, P.V. *Phys. Rev. D* 110, 096023

- Enforce equivariance in the output of a neural network
 - After training score is CP-odd (even) for CP-odd (even) processes
 - Any SM-like mismodelling/background will be symmetric by construction!
 - Constructive/destructive interference pattern for positive/negative values
- Equivariance as inductive bias speeds up convergence
 - Between **40% and 300% less iterations** needed to achieve the same loss value!!!



- Equivariance respected at all stages of training
 - The observable is robust even before training convergence



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 $N(\text{eval } F) \geq \mathcal{O}(\text{dim}(\theta))$)
- 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 $F(\cdot)$

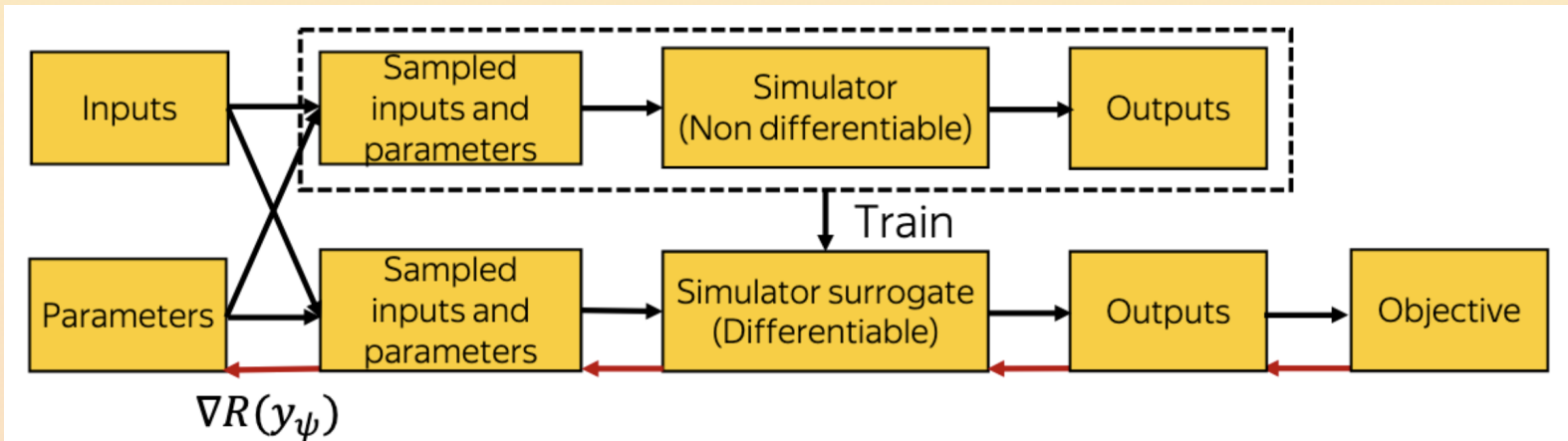
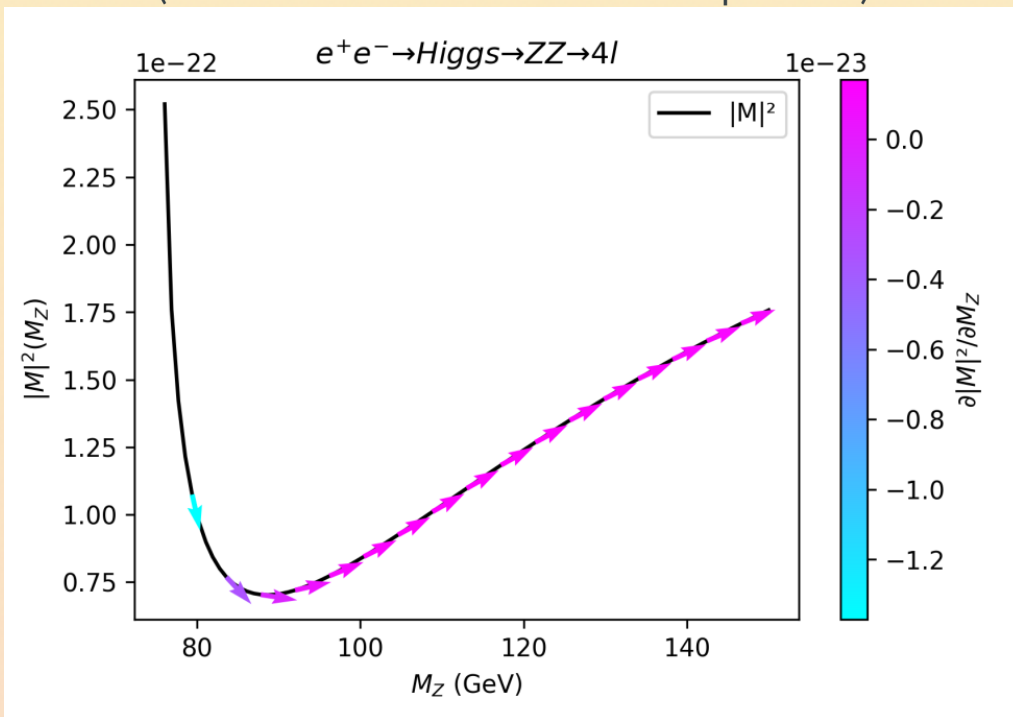


Figure 1: Simulation and surrogate training. *Black*: forward propagation. *Red*: error backpropagation.

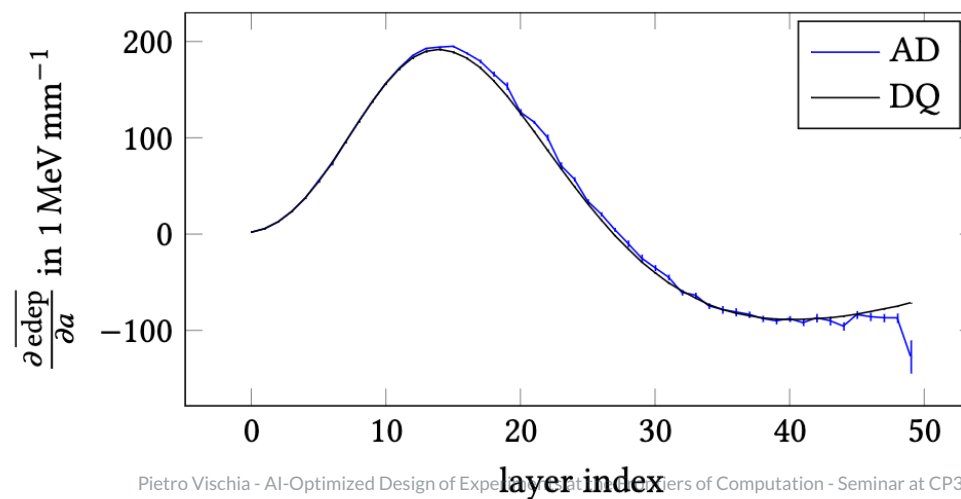
Long-term solution: make everything differentiable

- Multi-channel integration speed up in MadGraph([MadNIS](#))

MadJax ([2203.00057](#))
(differentiable matrix element computation)

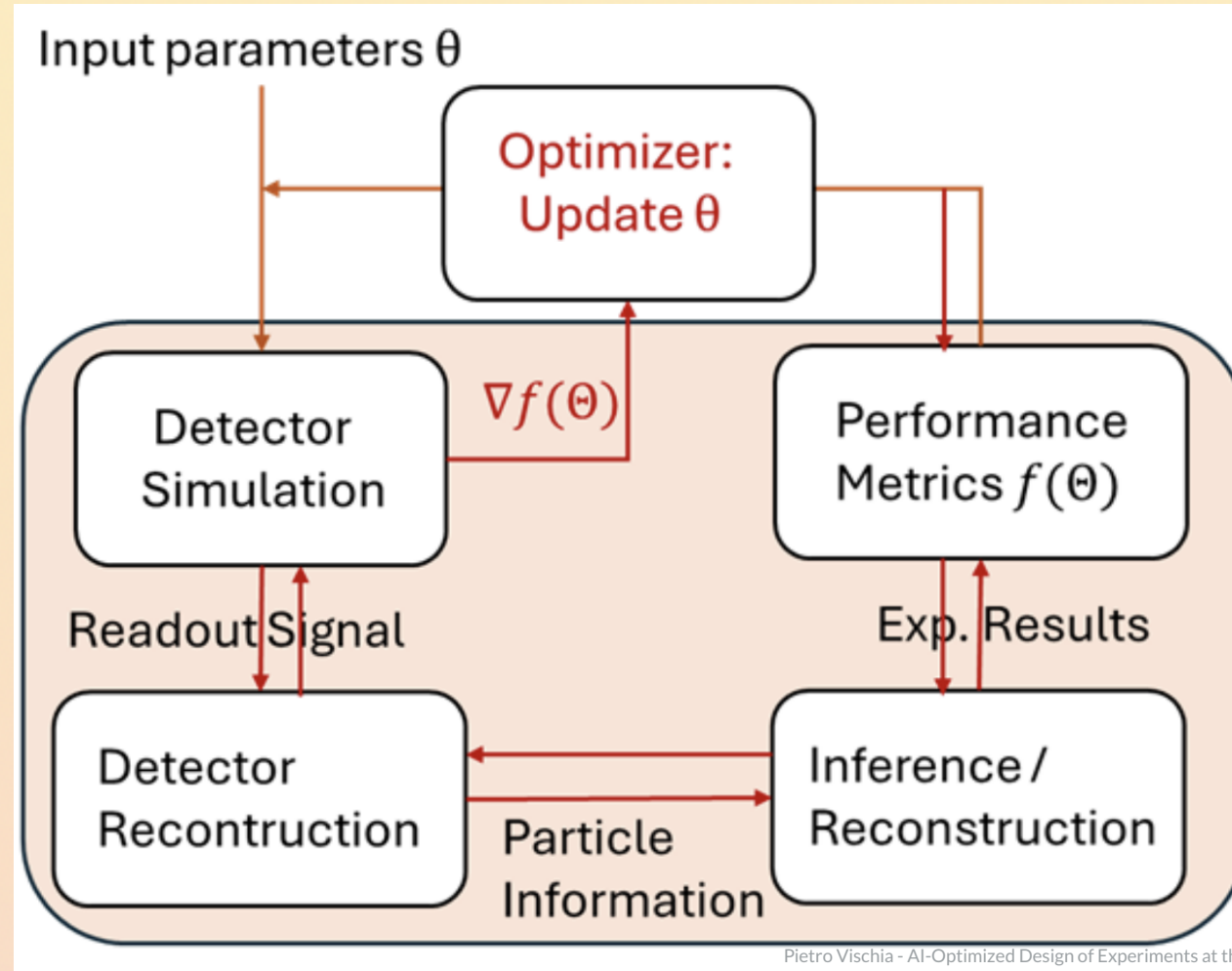


- Detector simulation: GATE/GEANT4 numerically differentiable (in small ranges) ([2202.05551](#))
- Differentiable electromagnetic showers for GEANT4 ([2405.07944](#))
 - Derivgrind**: insert AD logic into the program (a sort of debugger): cannot support tricky cases
 - CoDiPack**: operator overloading (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](https://arxiv.org/abs/2203.13818) (2203.13818), 117-page document, physicists + computer scientists



Feasibility within constraints

Depends on z and nuisances

Cost of the layout with parameters theta

Closed form

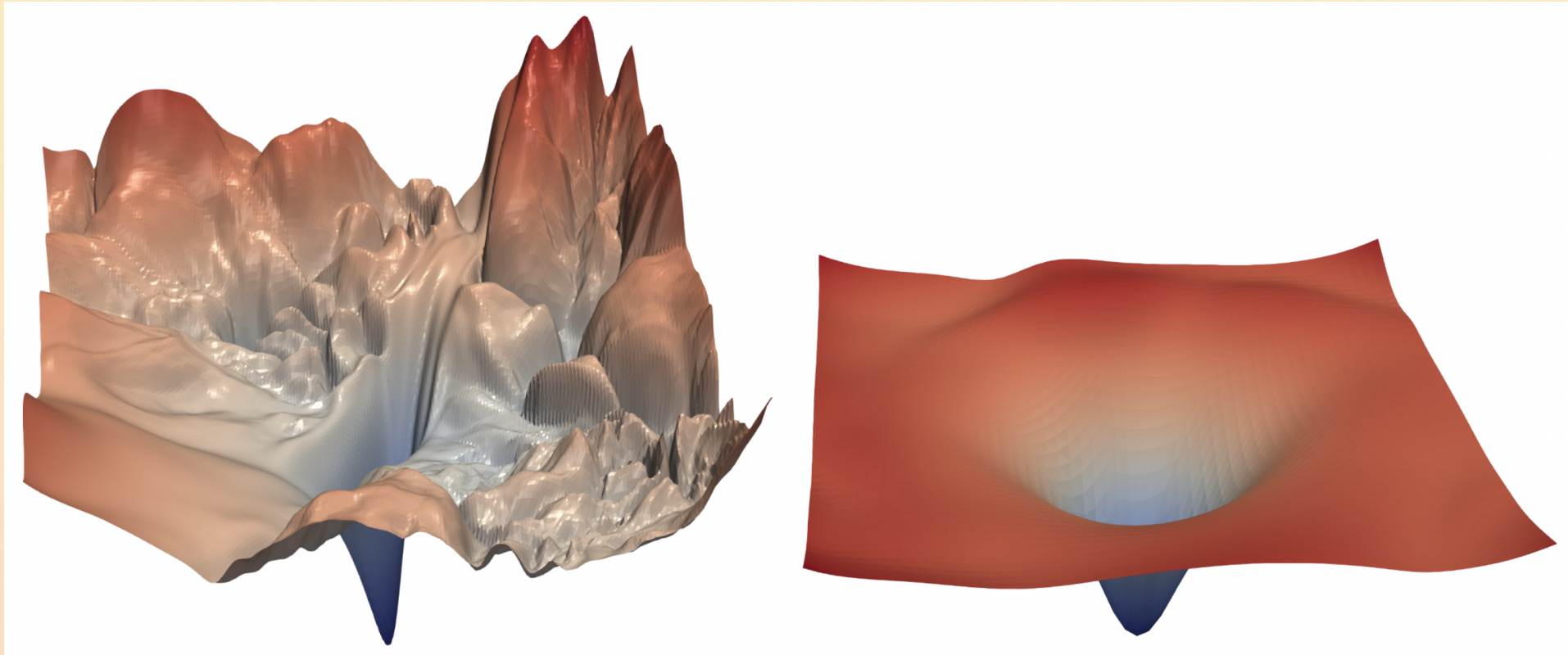
$$\hat{\theta} = \arg \min_{\theta} \int L[A(\zeta), c(\theta)] p(z|x, \theta) f(x) dx dz ,$$

Weight desirable goals while obeying cost constraints

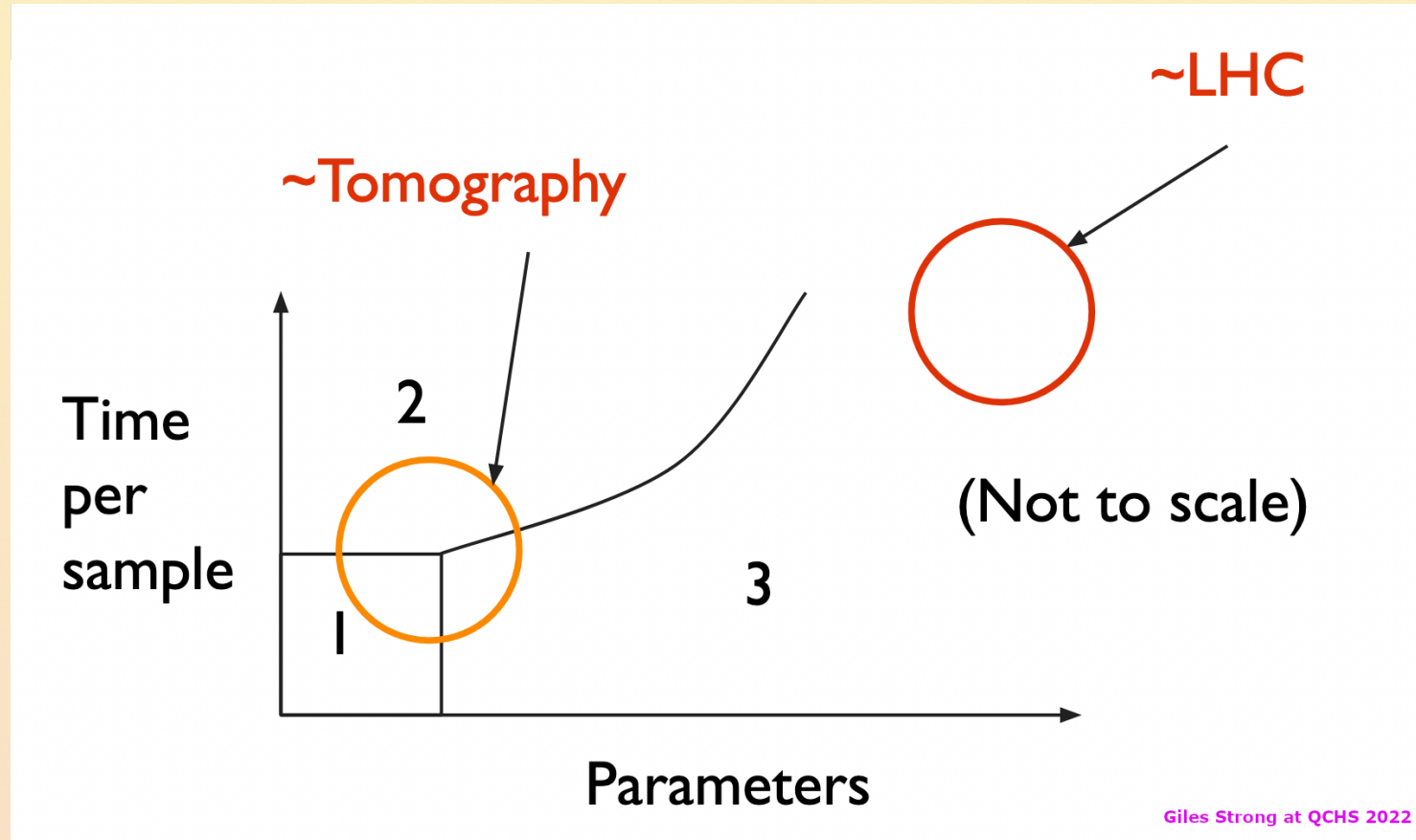
- Costs can be monetary but also any case-specific technical constraint
 - Local, specific to the technology used (e.g. active components material)
 - Global, describing overall detector conception (e.g. number, size, position of detector modules)
- Fixed costs can be added separately to the loss function

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



Method of choice depends on scale

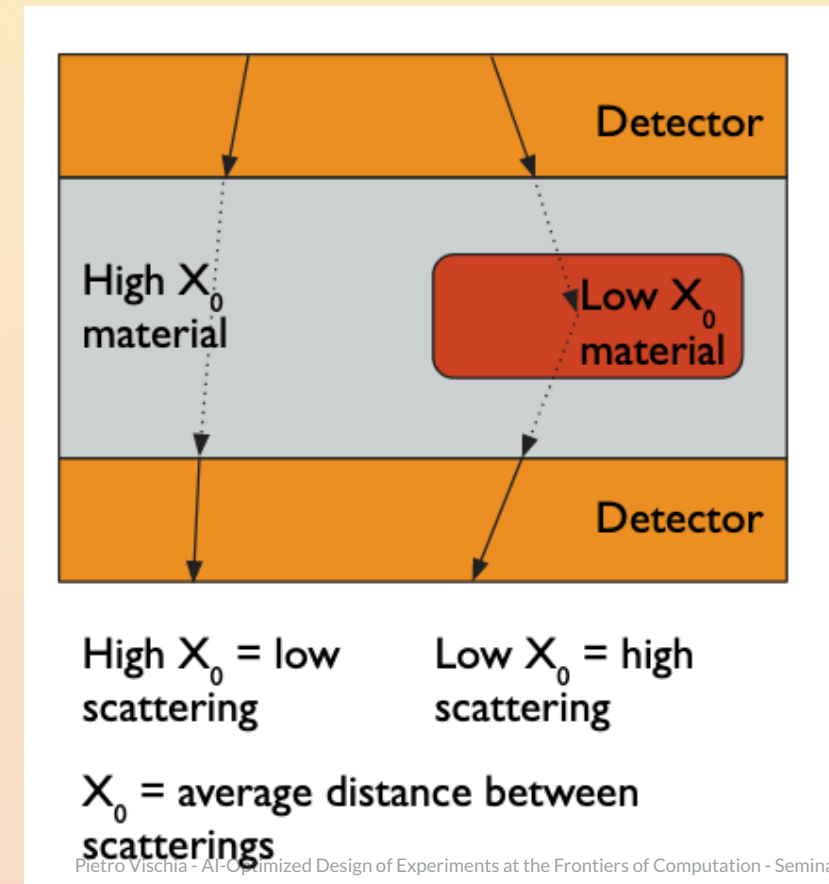
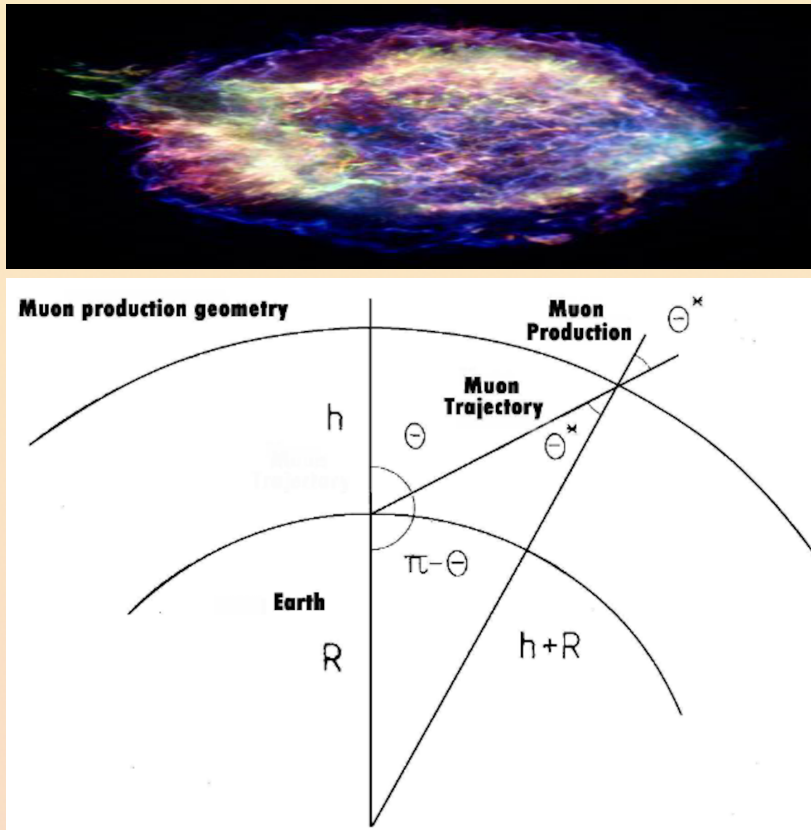


Giles Strong at QCHS 2022

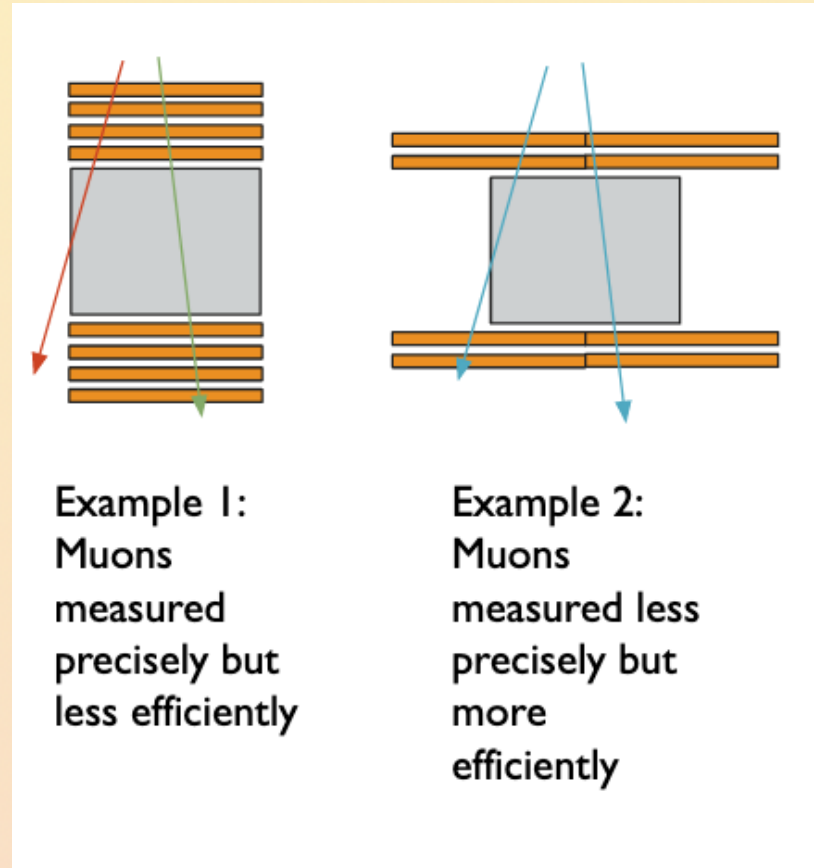
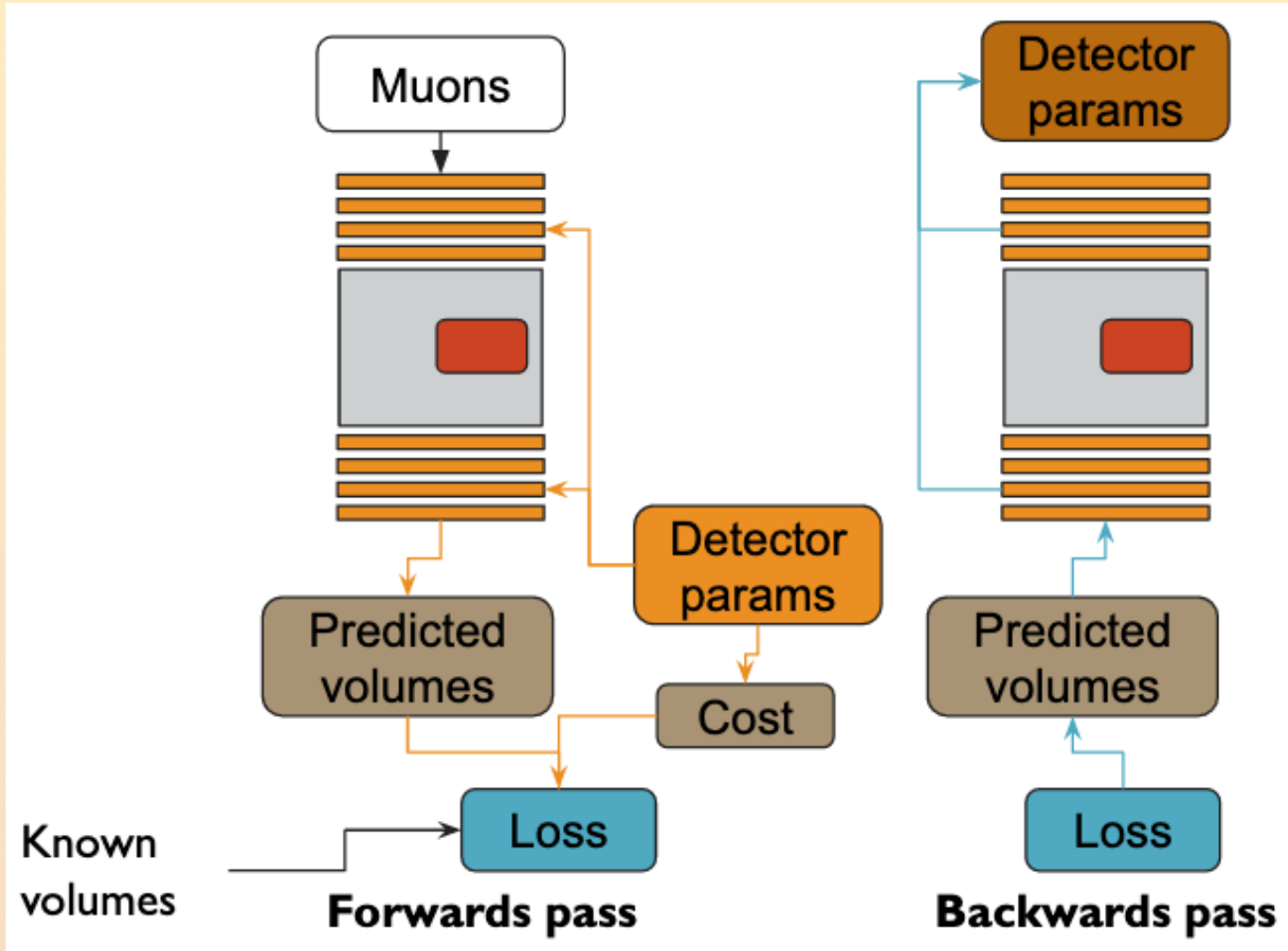
1. Grid/random search
2. Bayesian opt, simulated annealing, genetic algos, ...
3. [Gradient-based optimization](#) (Newton, BFGS, [gradient descent](#), ...)

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, volcanoes) or **scattering** (measure deflection, e.g. containers, furnaces, statues)



Optimize a parametric design under constraints



TomOpt G. Strong et al. (P.V.) Mach. Learn.: Sci. Technol. 5 035002

- Differential optimization of muon-tomography detectors
- 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

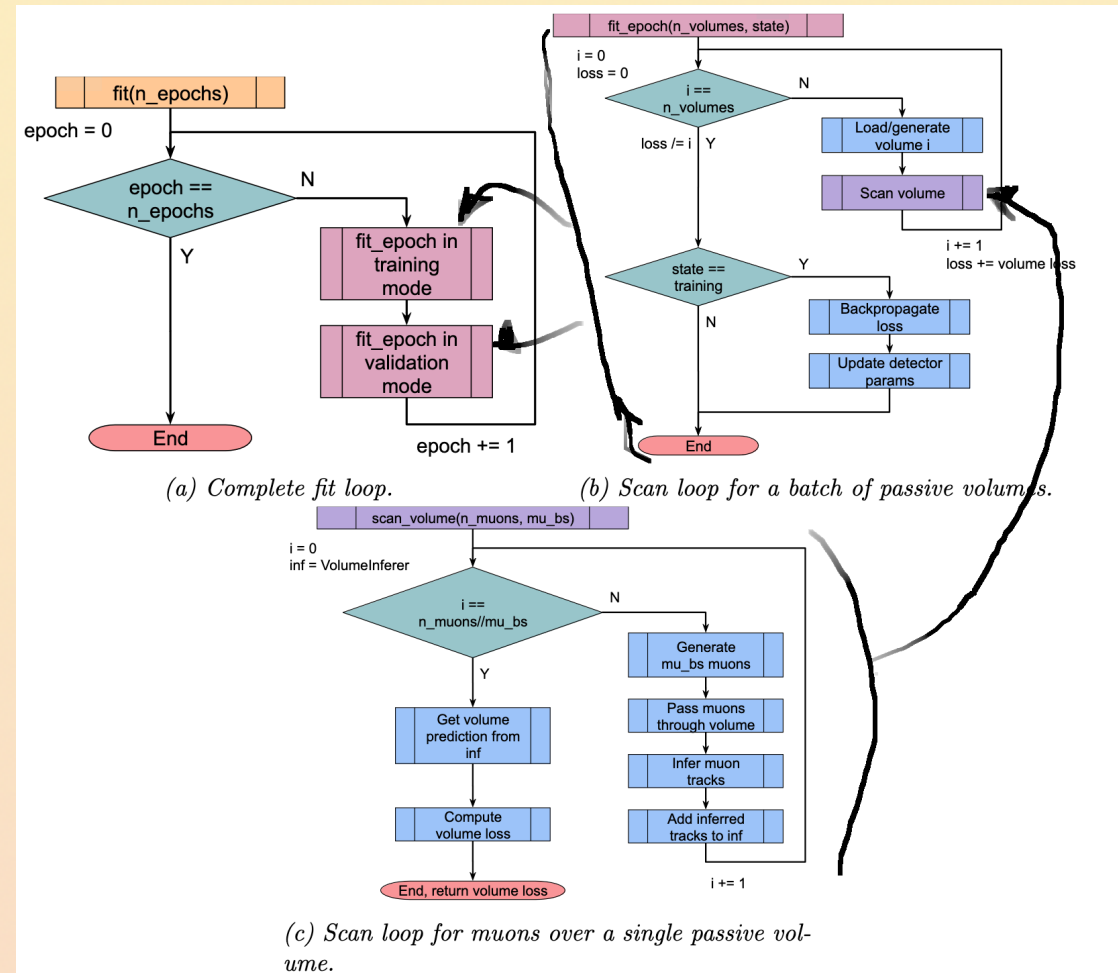
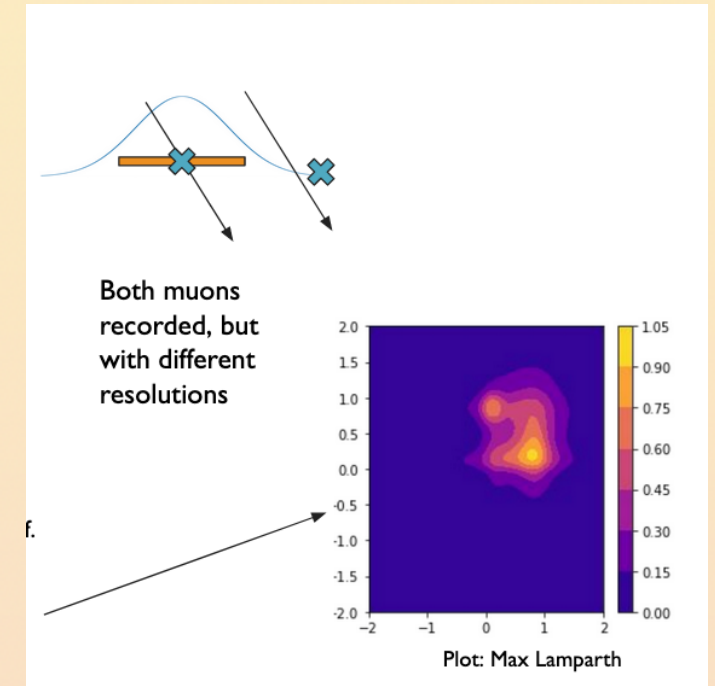
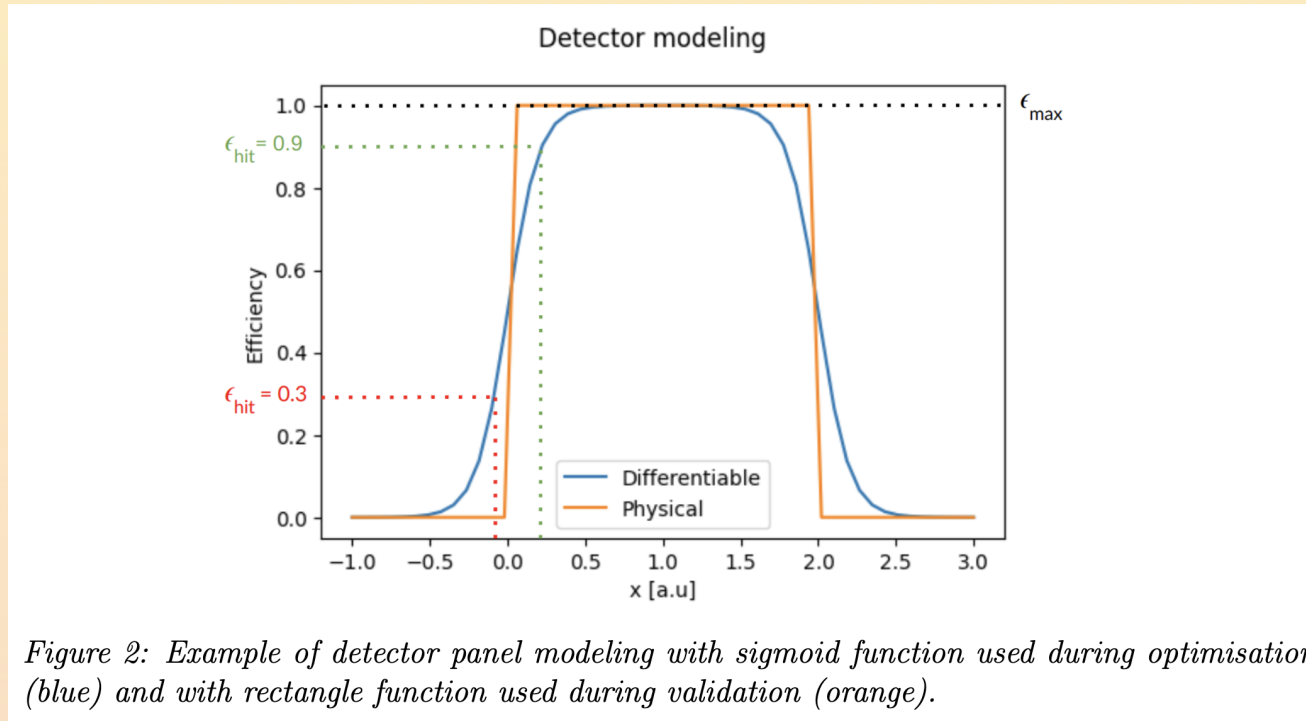


Figure 6: Breakdown of the fitting procedure of detectors in TOMOPT

Make things differentiable

G. Strong et al. (P.V.) Mach. Learn.: Sci. Technol. 5 035002

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



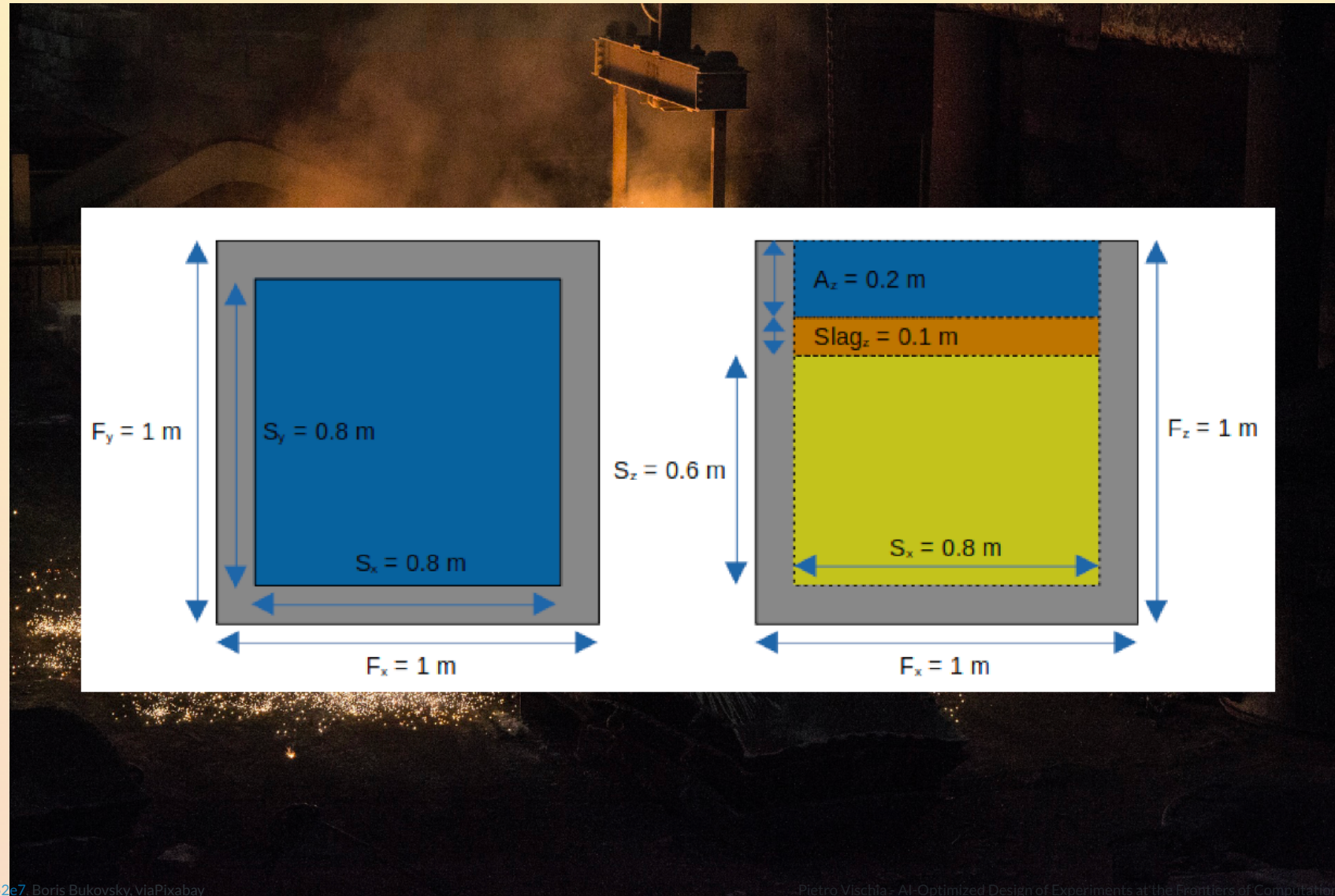
Heavy-Metal Benchmark

G. Strong et al. (P.V.) Mach. Learn.: Sci. Technol. 5 035002



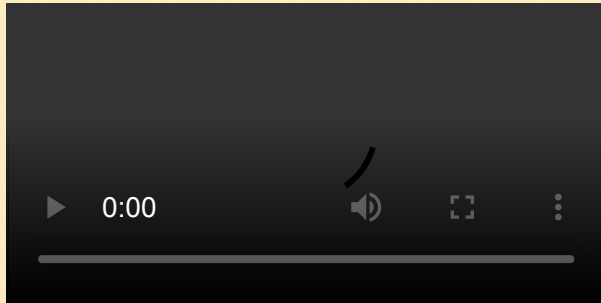
Heavy-Metal Benchmark

G. Strong et al. (P.V.) Mach. Learn.: Sci. Technol. 5 035002

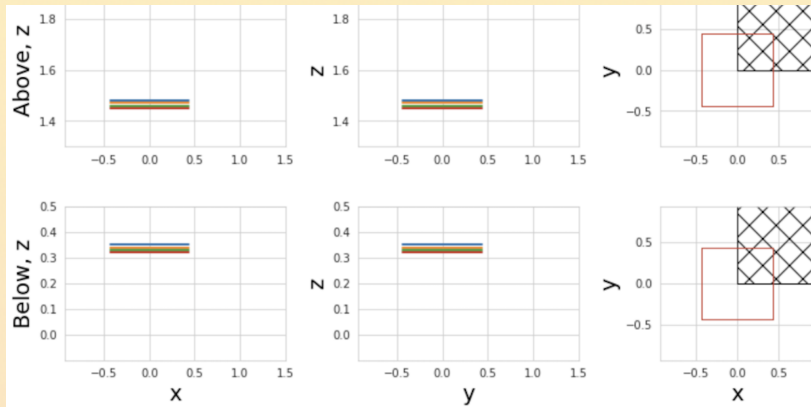


Optimized design

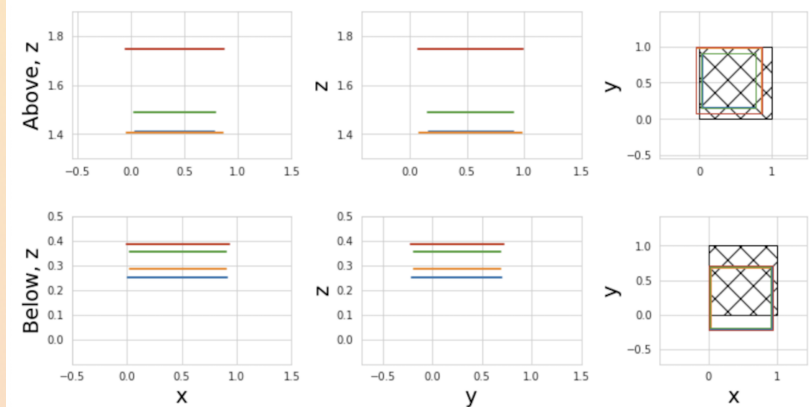
G. Strong et al. (P.V.) [Mach. Learn.: Sci. Technol. 5 035002](#)



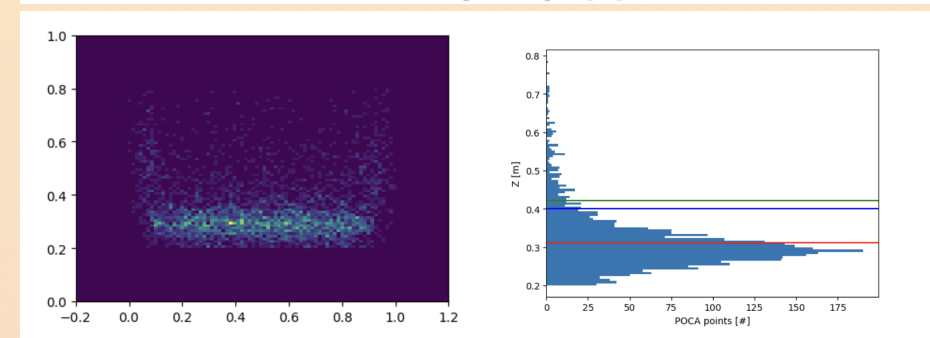
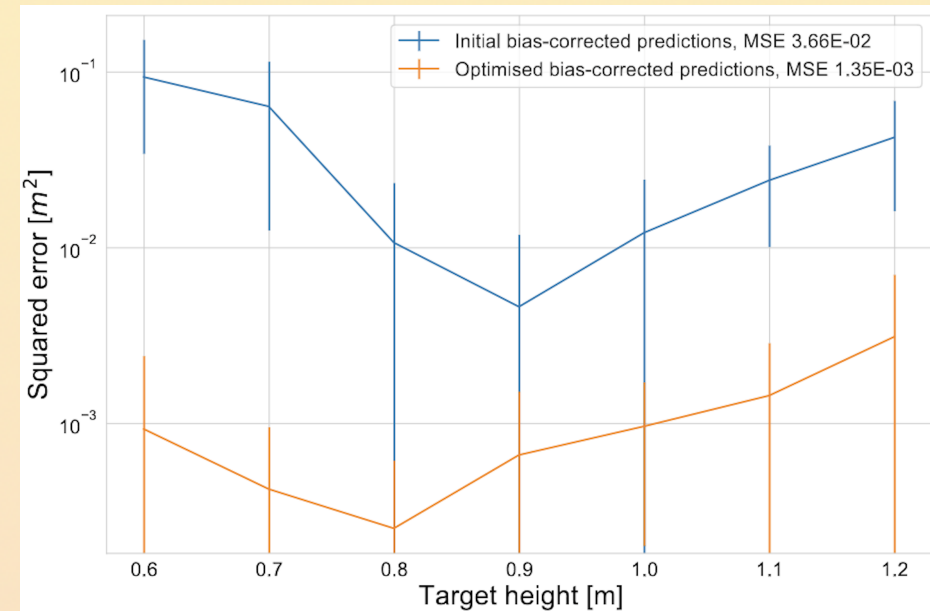
Optimized design is more performant Mach. Learn.: Sci. Technol. 5 035002



(a) Initial detector configuration.



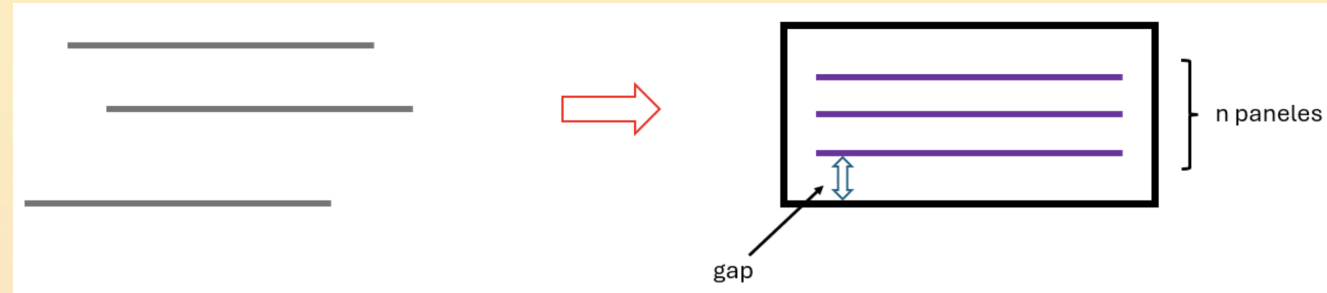
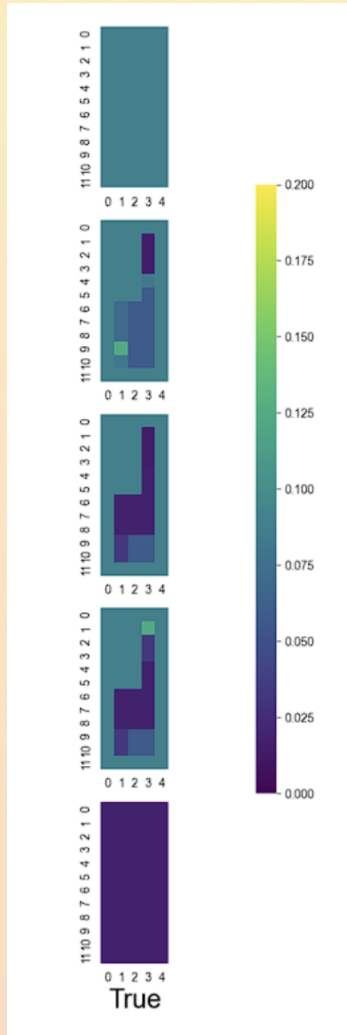
(b) Detector configuration after stage one optimisation process.



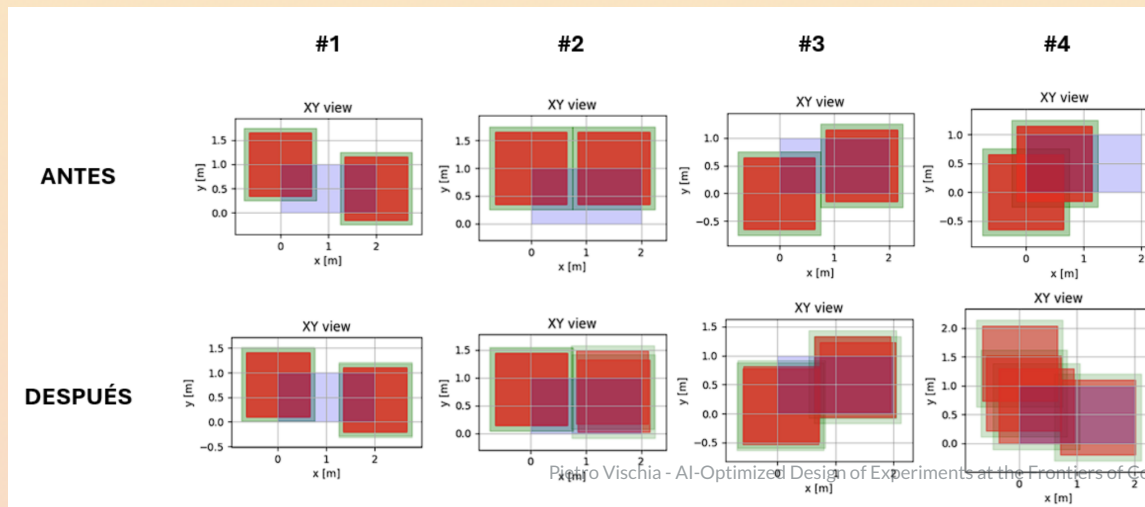
(Blue: true steel level. Red: prediction. Green: bias-corrected prediction)

Extension: hodoscopes Z. Zaher et al. (P.V.) Particles 2025 8(2) 53

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

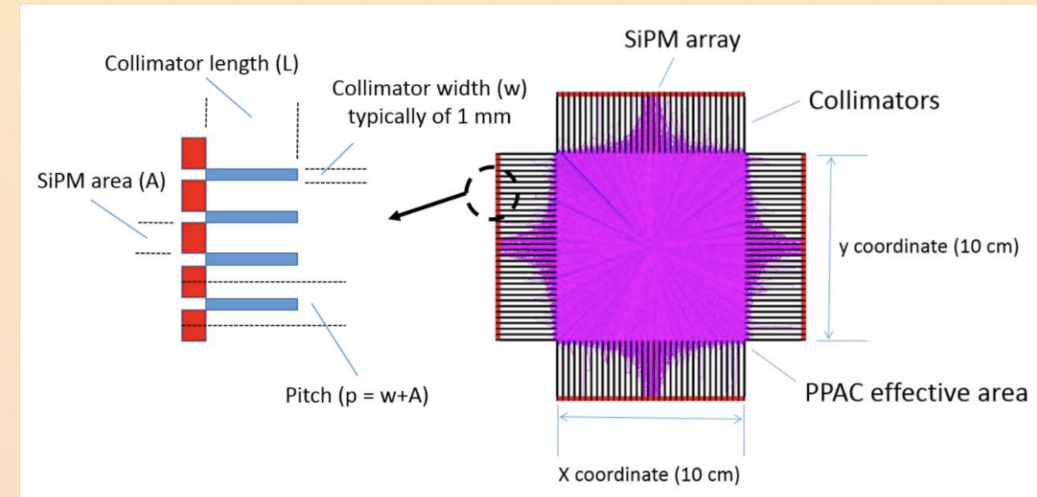
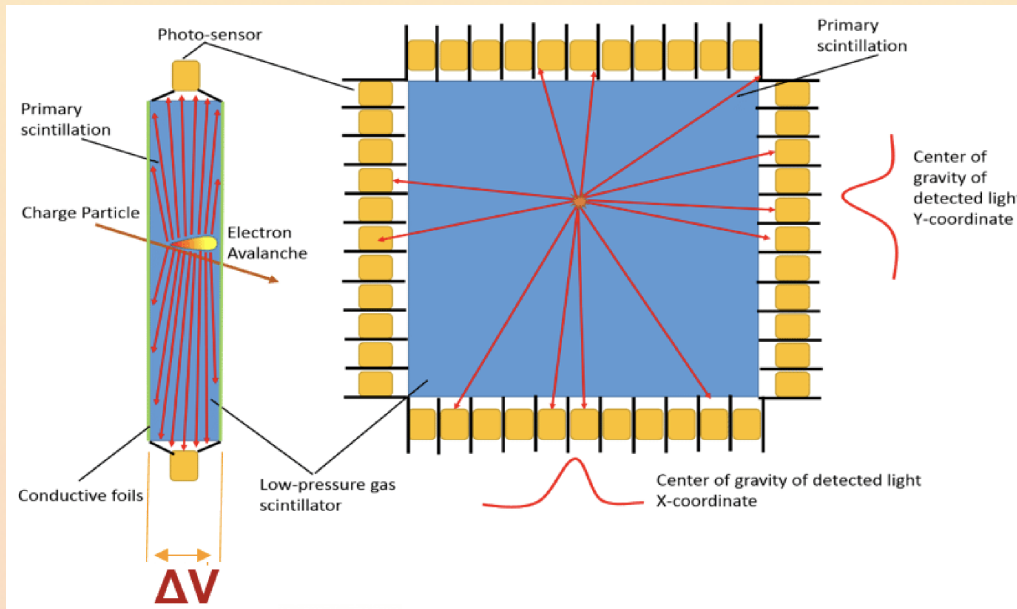


- Scan a lorry: significant code restructuring to account for hodoscope structure
- Accepted by the MARESEC conference for paper publication
 - [Zaher, Lagrange, Álvarez Lueje et al. \(P.V.\), 2024](#) (BSc thesis of Samuel Álvarez Lueje)



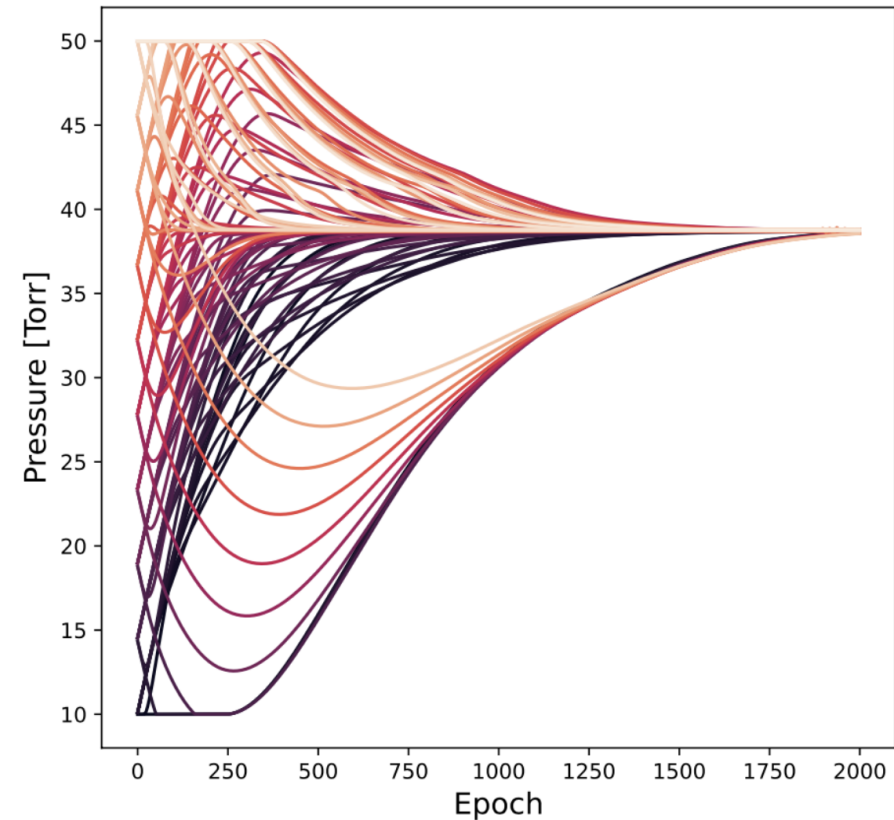
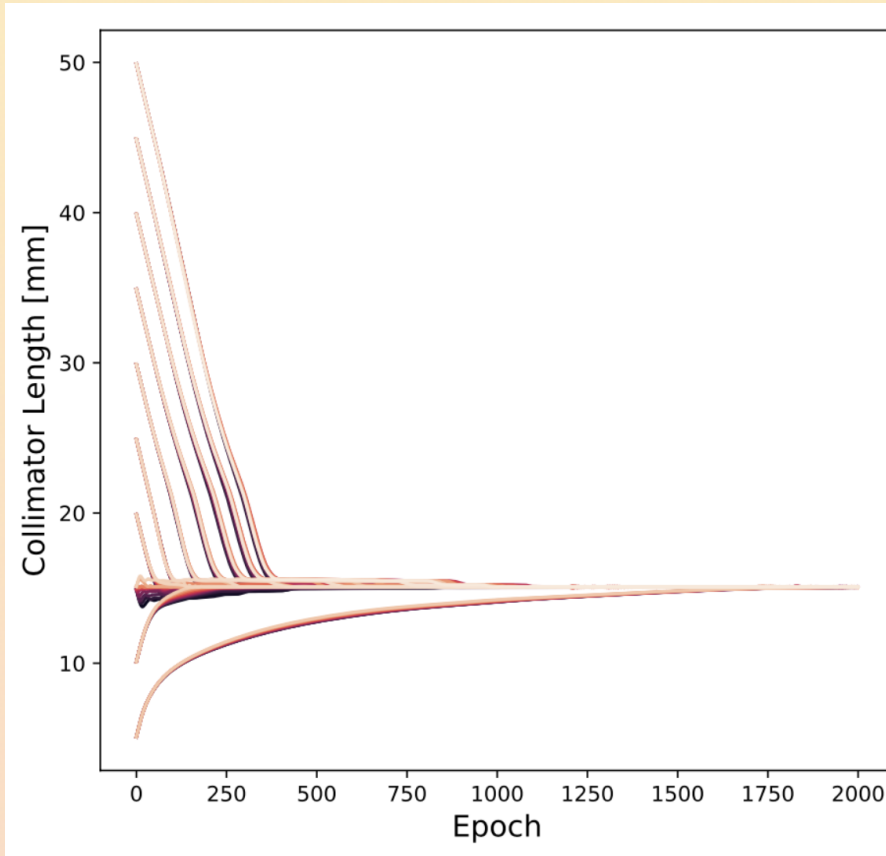
Neutron Tomography M. Pereira Martínez, X. Cid Vidal, P.V. Particles 2025 8(1) 26

- GEANT4 model of a $10 \times 10\text{cm}^2$ 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)



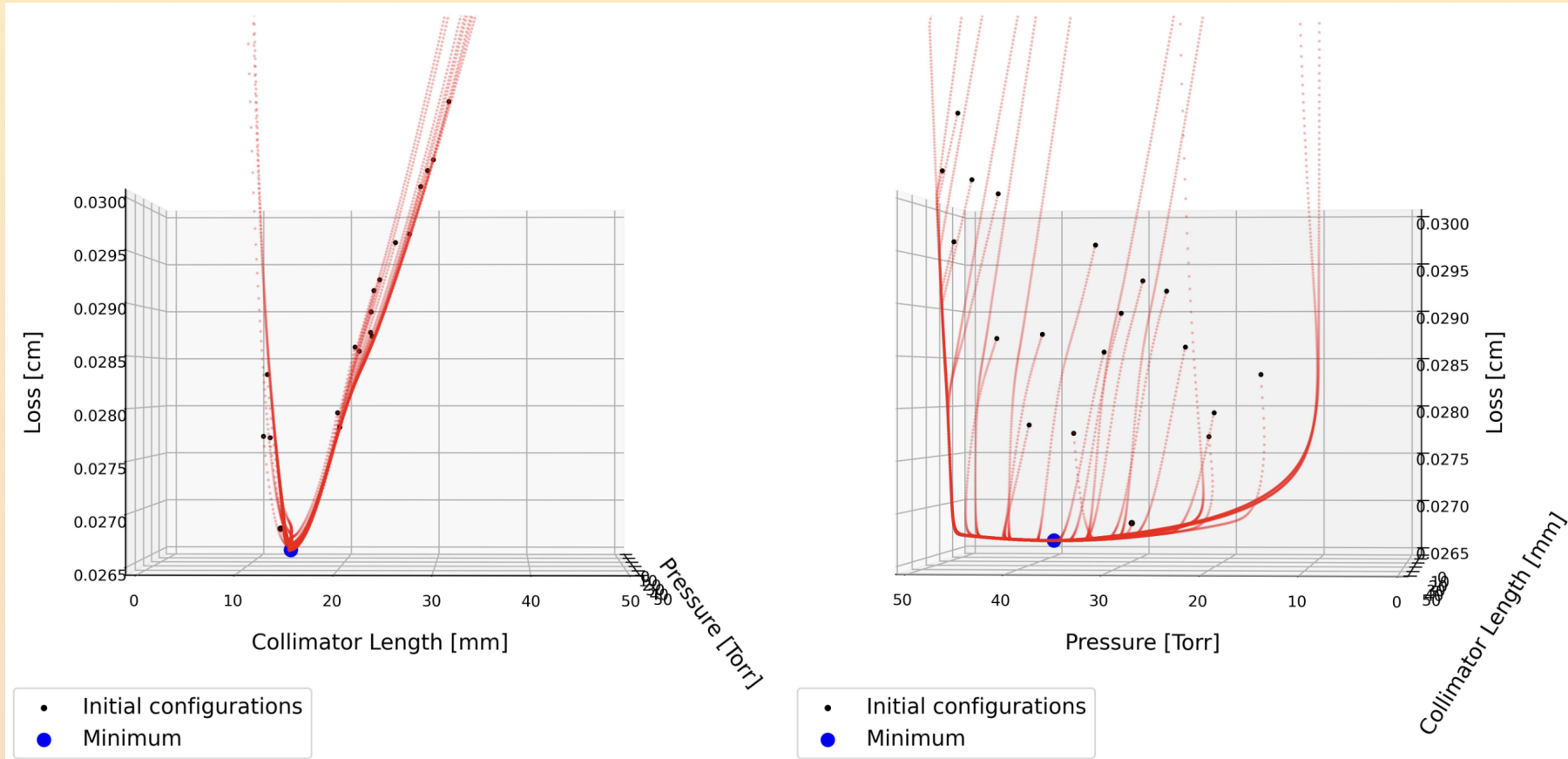
Neutron Tomography M. Pereira Martínez, X. Cid Vidal, P.V. *Particles* 2025 8(1) 26

- Results for L give the same result as traditional studies from [10.1088/1748-0221/13/10/P10006](https://doi.org/10.1088/1748-0221/13/10/P10006)
- Remarkably stable **regardless of initial configuration**
- MSc thesis of María Pereira Martínez, doctorate ongoing



Neutron Tomography M. Pereira Martínez, X. Cid Vidal, P.V. Particles 2025 8(1) 26

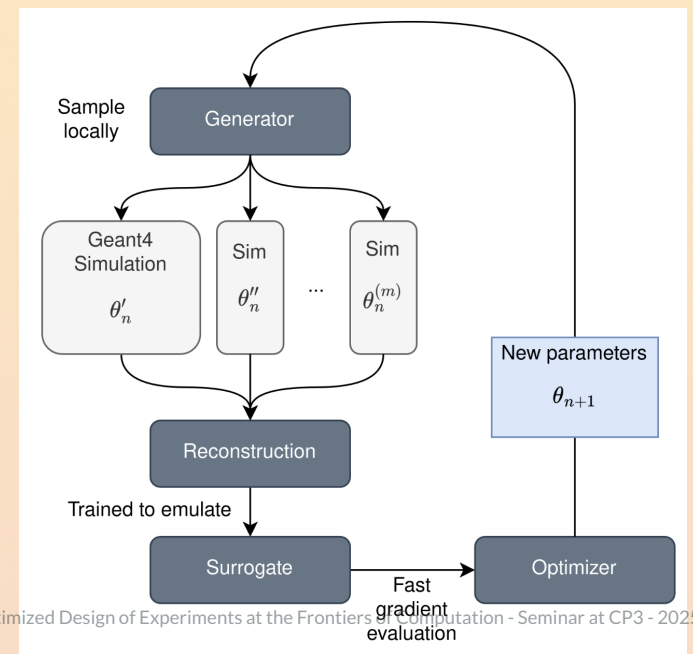
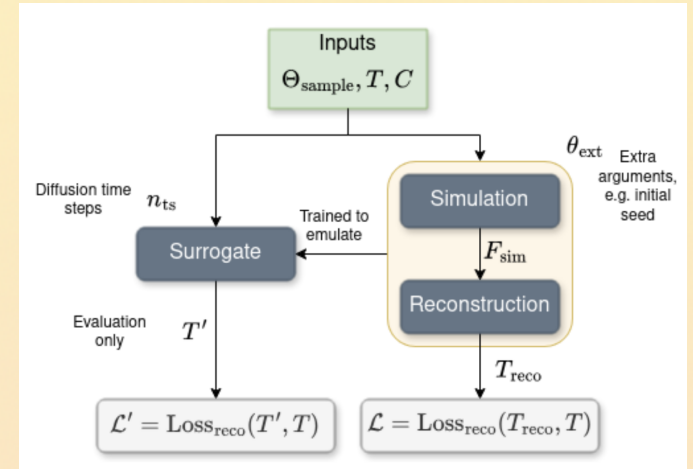
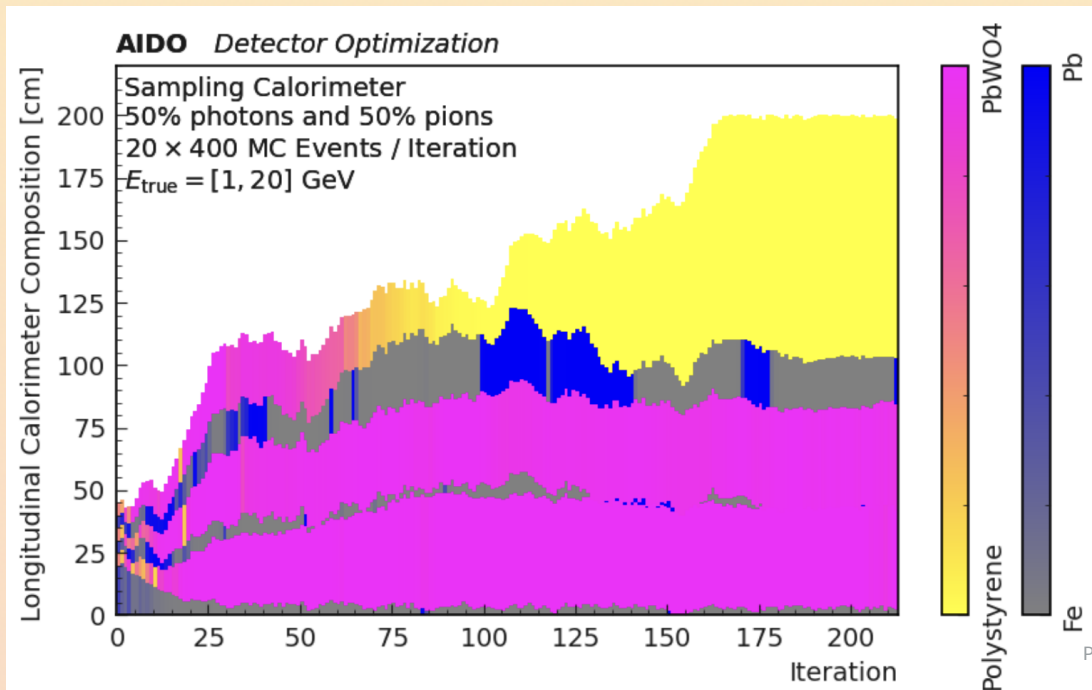
- Results for L give the same result as traditional studies from [10.1088/1748-0221/13/10/P10006](https://doi.org/10.1088/1748-0221/13/10/P10006)
- Remarkably stable **regardless of initial configuration**
- MSc thesis of María Pereira Martínez, doctorate ongoing



E2E detector optimization

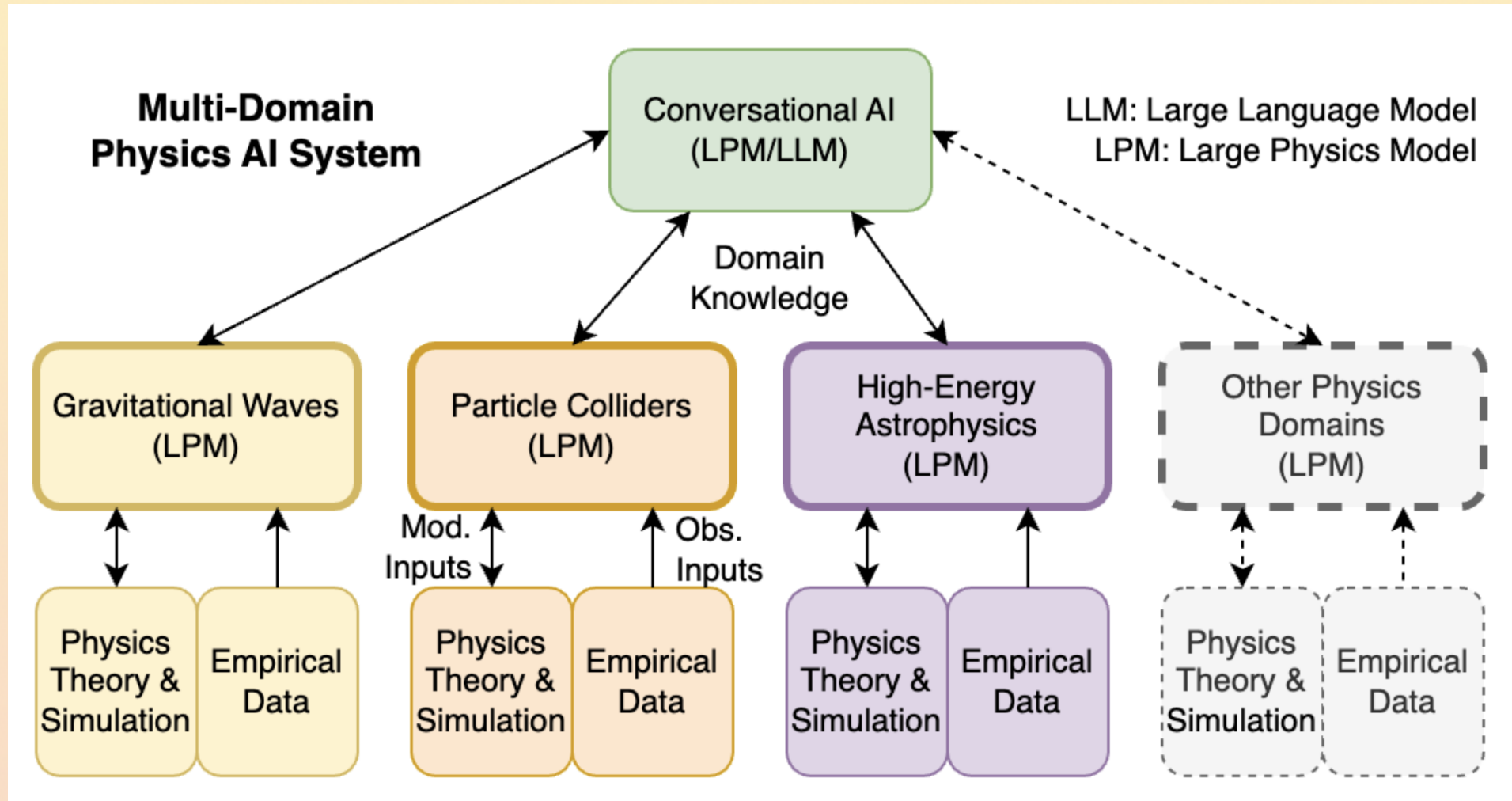
K. Schmidt et al. (P.V.) *Particles* 2025 8(2) 47

- AIDO: Diffusion model replaces the full simulation and reconstruction chain
 - This work: focus on sampling calorimeter
 - General approach extendable to full detector
- Learn directly output of reconstruction step
 - Surrogate via parallelized GEANT4 simulations
- Constrained optimization: 200cm length, 200k EUR budget
 - Better materials (lead for absorber, $PbWO_4$ for scintillator) cost more



Component of Large Physics Models? K. G. Barman et al. (P.V.) 2501.05382

- Tailored model for detector optimization may inform future conversational AI physics model



Thrive in asymmetries or lack thereof

2006: genetic algorithms



2024: SWGO tanks placement optimization

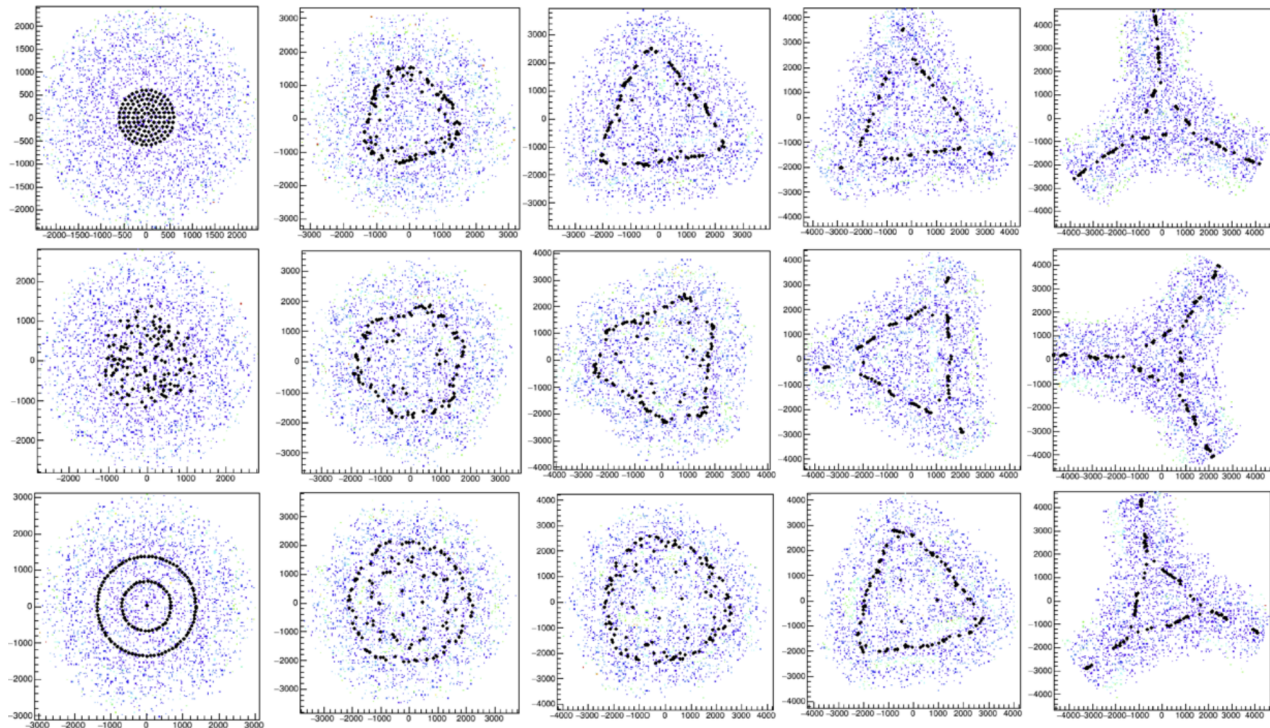
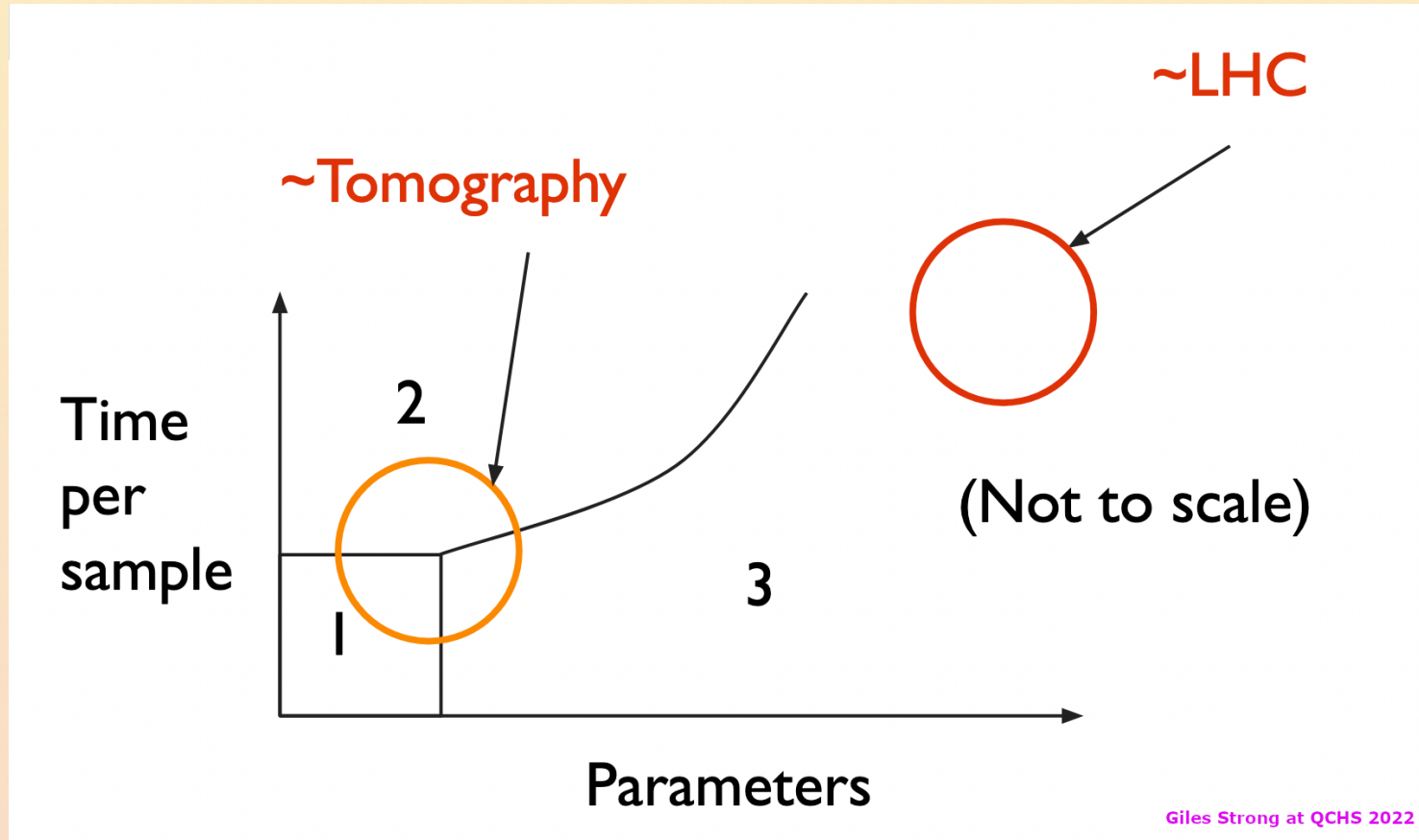


Figure 14: Convergence of three initial layouts (top to bottom: packed ball, wide random ball, two annuli) during a 500-epochs training. From left to right, the configurations of 126 units (129 in the bottom one) are shown at epoch 1,50,150,300, 500. See the text for more detail.

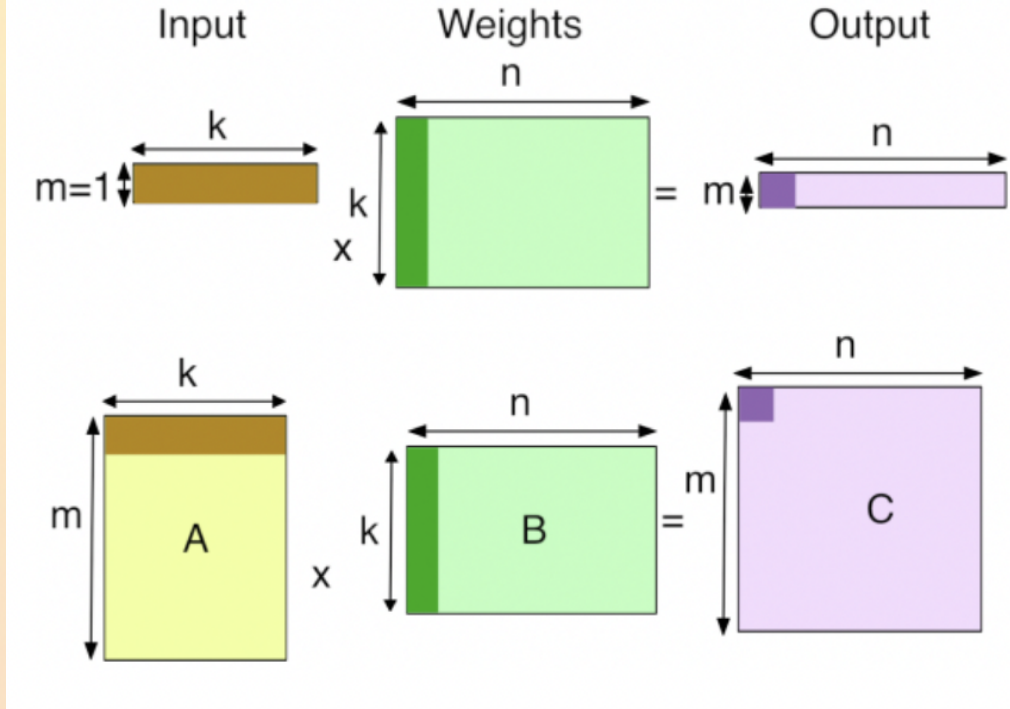
Experimental design: present and future

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

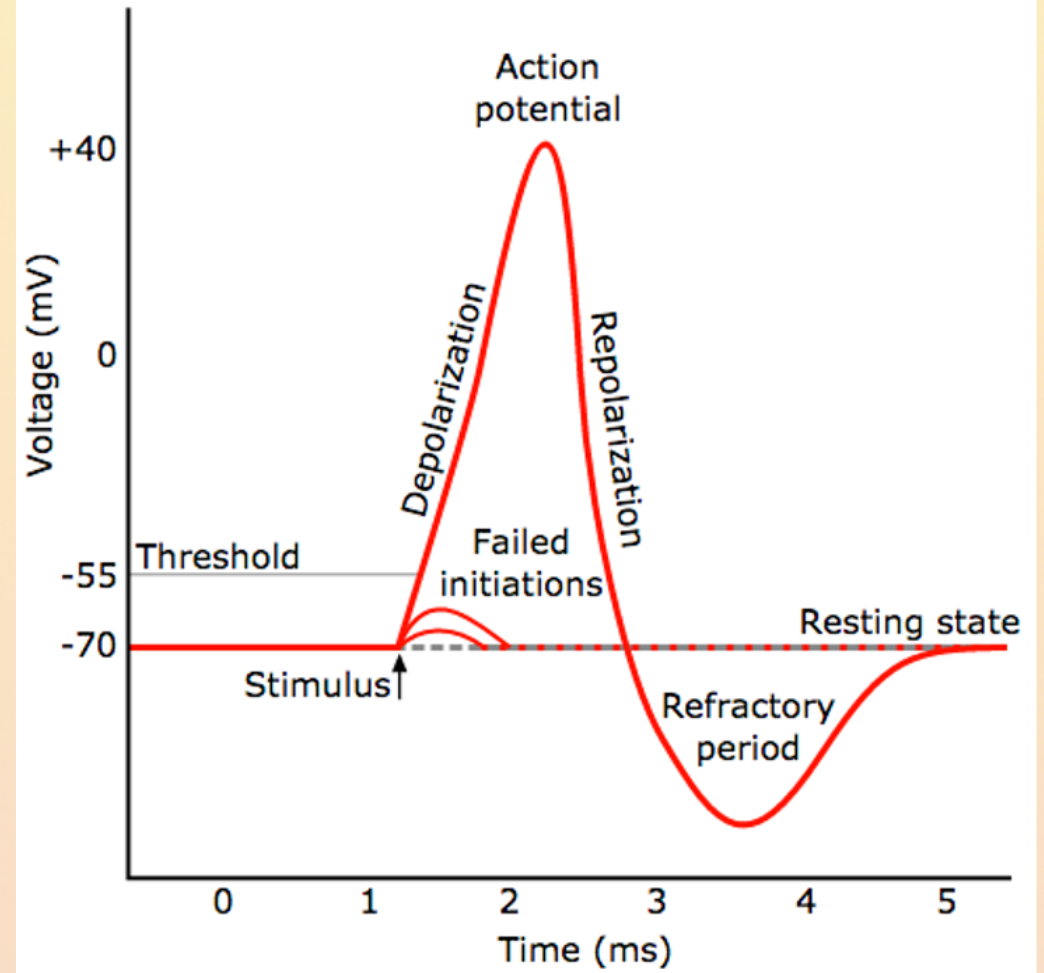


Paradigm shift

From perceptrons and matrices...



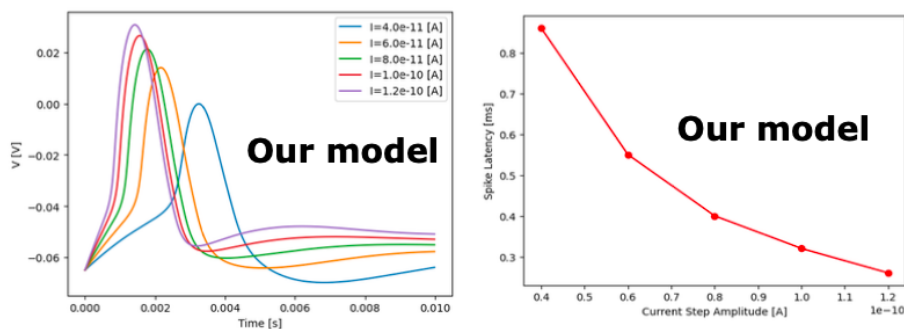
to spiking neurons modulating spatiotemporal patterns



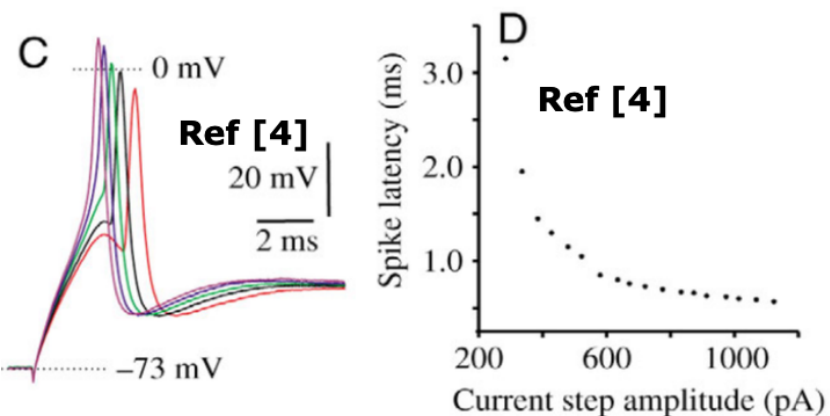
Realistic models for biological neurons

- 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.)

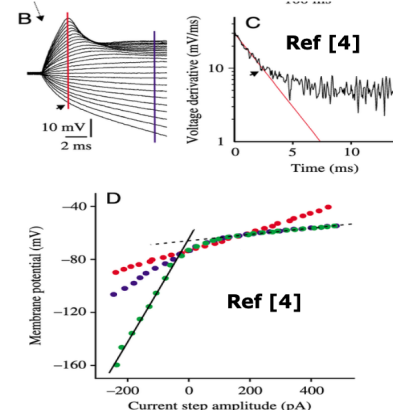
- The amplitude of the stimulus step drives the spike latency



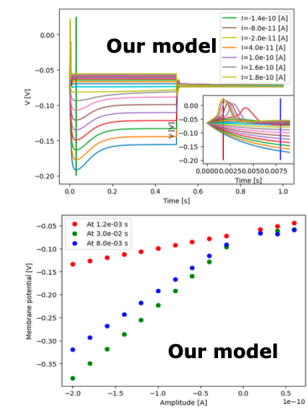
- Observations exhibit the same behaviour
- Further turning needed for the spike shape



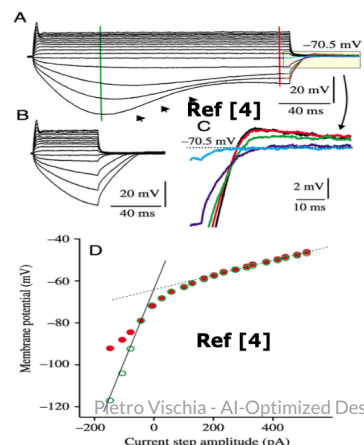
- Early subthreshold responses



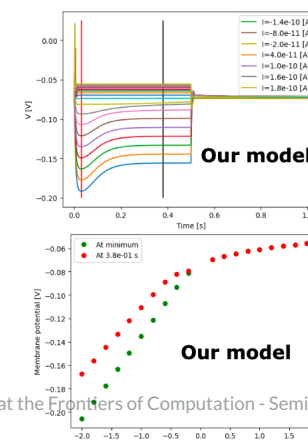
- Before hump (red), linear V-I relation
- After hump, for depolarizing steps V-I relation is nonlinear
The activated conductance does not inactivate at later times



- Late subthreshold responses



- At peak hyperpolarization, limiting slope is maximal
- At end of the step, depolarization curves decay much faster



Event-driven computation

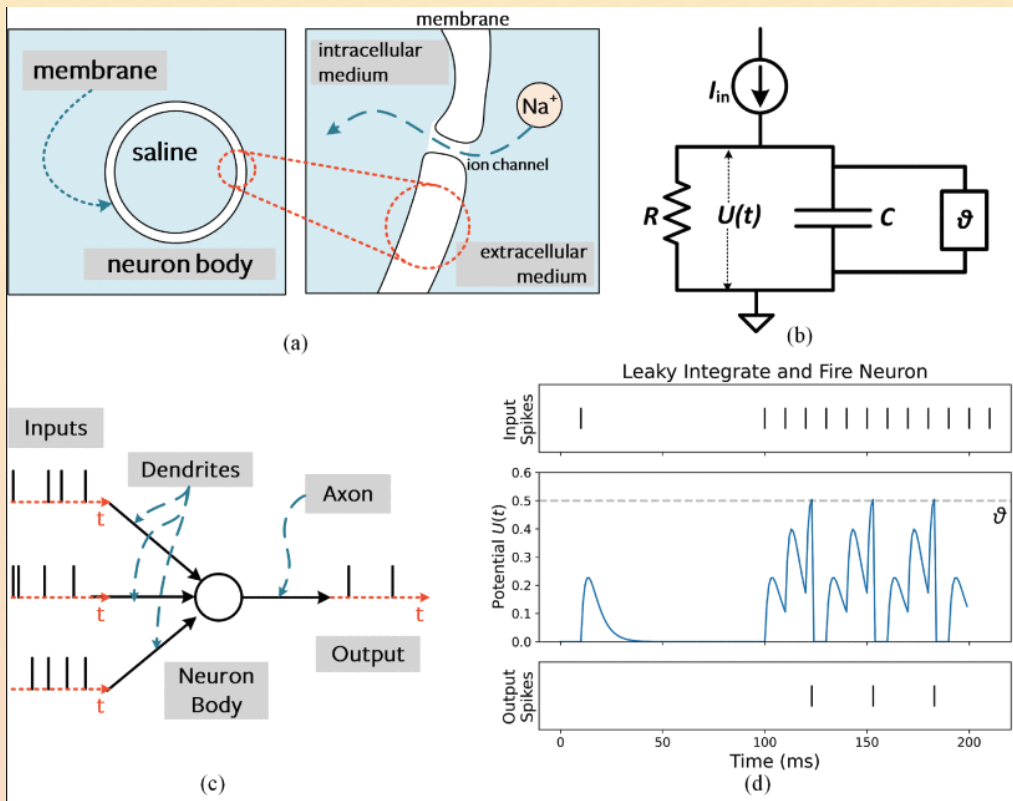
- Event-driven computations
 - "when a spike occurs, compute something": realtime operations by reducing bus width ($N_{axons} \rightarrow \log_2 N_{axons}$) in CMOS or memristors
 - Work in progress on various applications
 - Q-Pix (see talk by [Shion Kubota](#)) uses same natural representation: maybe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



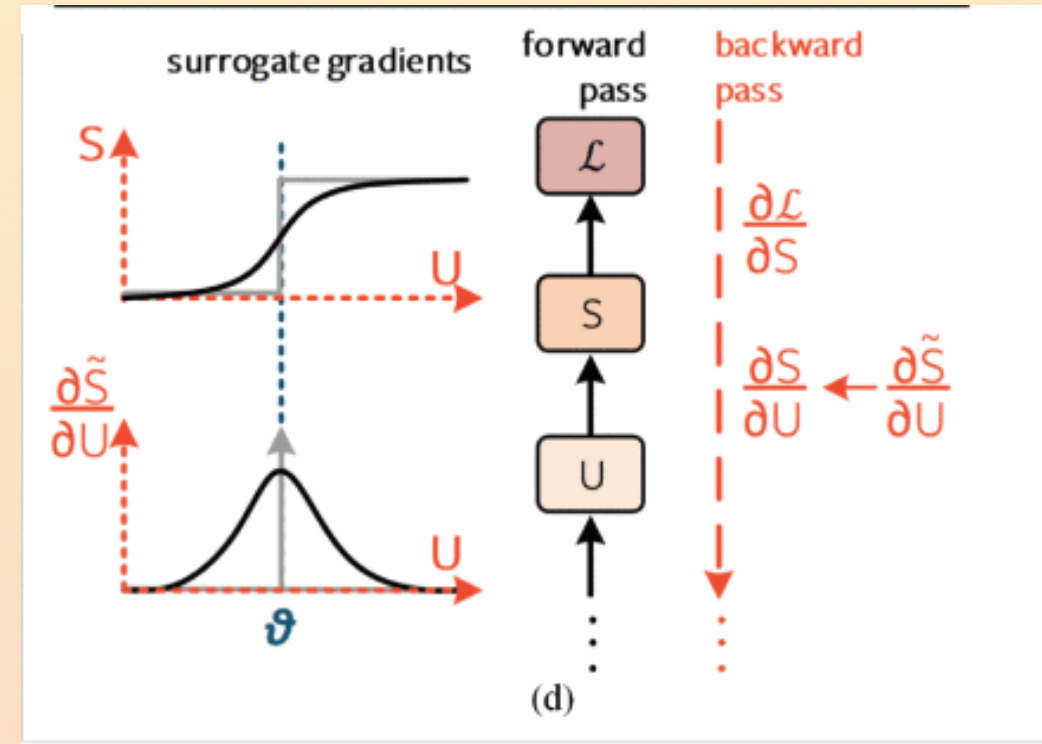
Artificial Spiking Networks

- Neuronal model vastly simplified: (Leaky) Integrate-and-fire Model. Other training schemes are available (not illustrated here)

$$U[t] = \underbrace{\beta U[t-1]}_{\text{decay}} + \underbrace{W X[t]}_{\text{input}} - \underbrace{S_{\text{out}}[t-1]\theta}_{\text{reset}}$$



- Surrogate gradient to mimick the usual gradient descent
 - Use automatic differentiation for training!

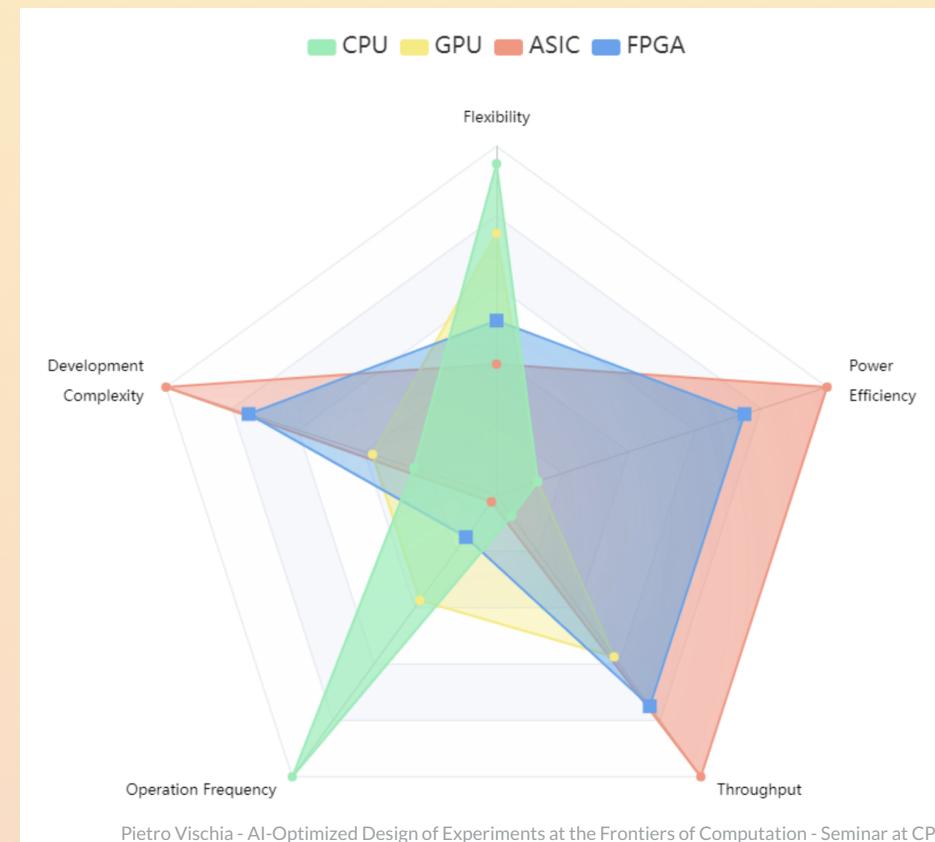
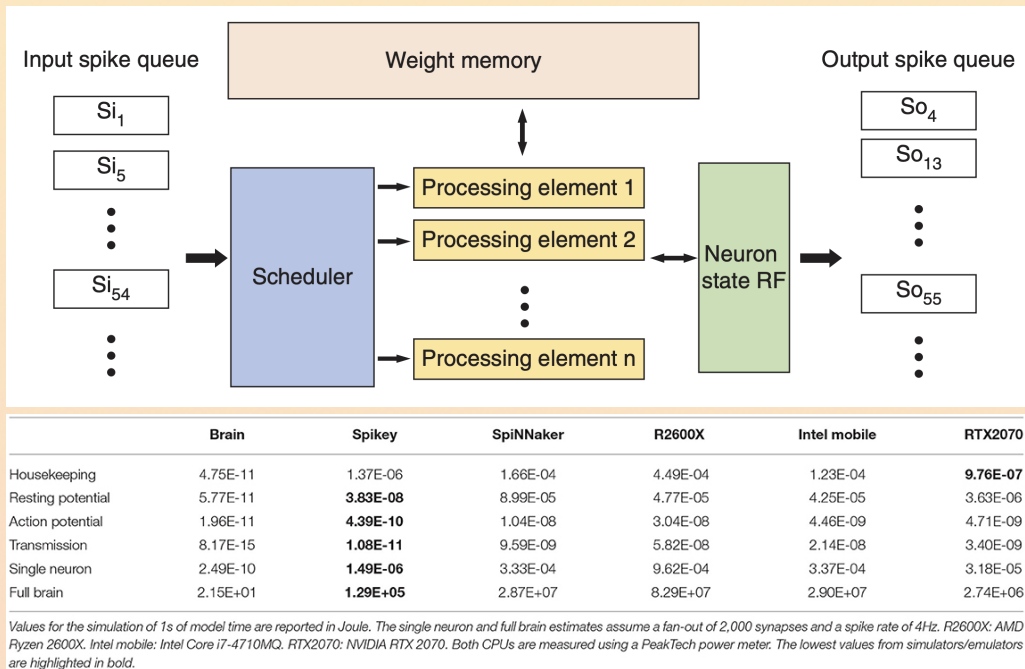


Energy-efficient architectures

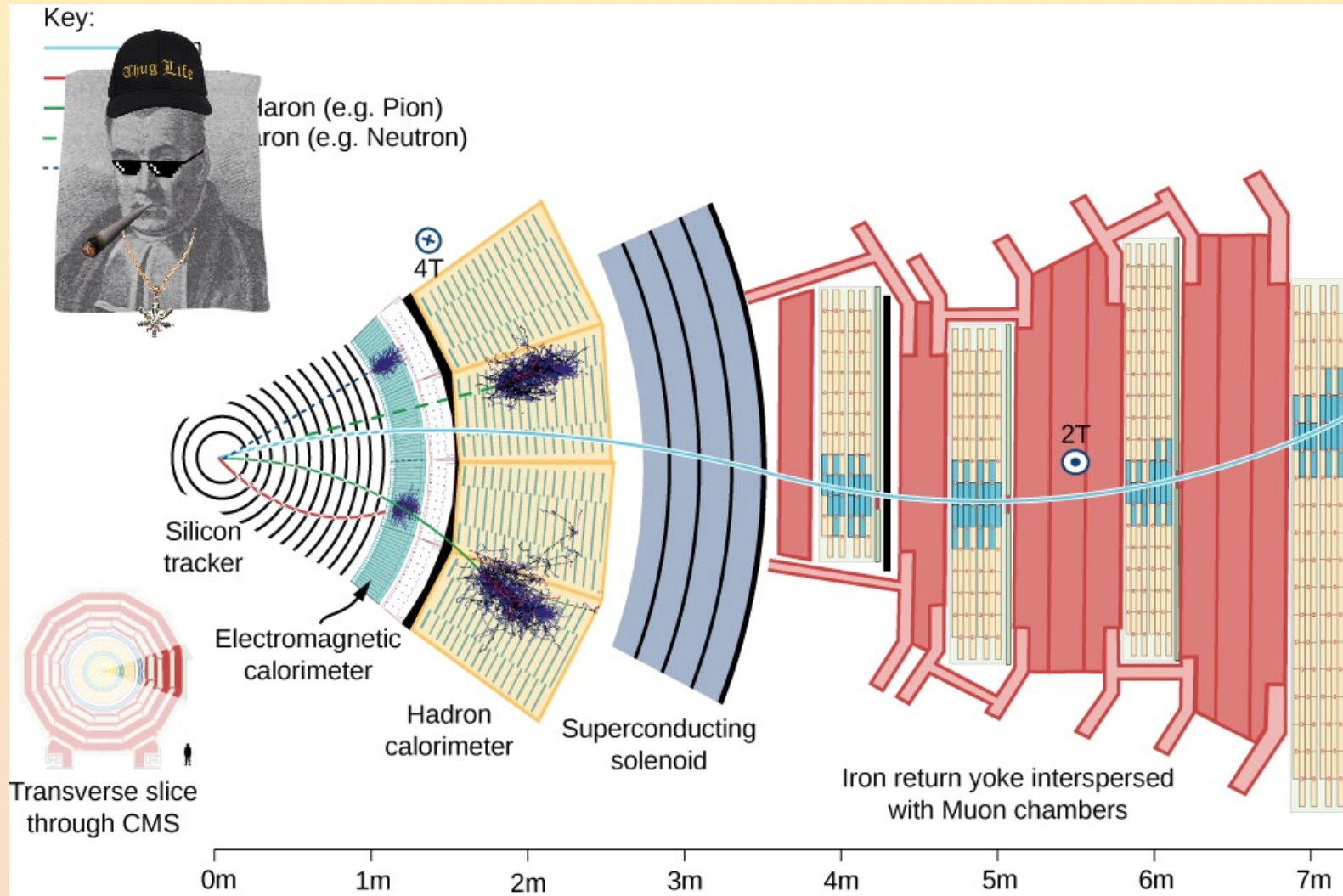
Sparser inputs → less time and energy

- Event-driven computation
- Efficient, threshold-based encoding
 - GPU (RTX3090): 40 GW simulation on discrete states
 - Human brain equivalent: **20 W on dynamic states**
- Energy consumption drastically limited in sparse settings

- Neuromorphic hardware has maximal energy efficiency
 - **Memristors are optimal**, otherwise CMOS/ASIC
 - FPGA implementations also available (e.g. [2502.20415](#), [2411.01628](#), [2307.03910](#), [2305.11252](#))

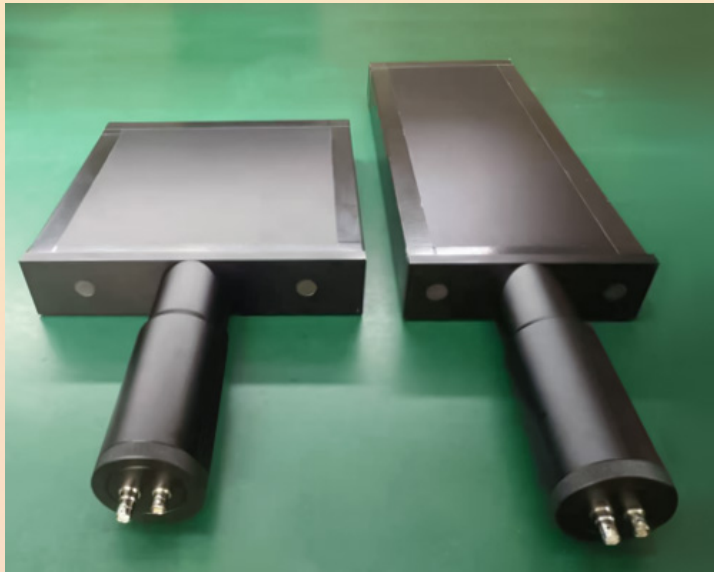


What is a calorimeter?



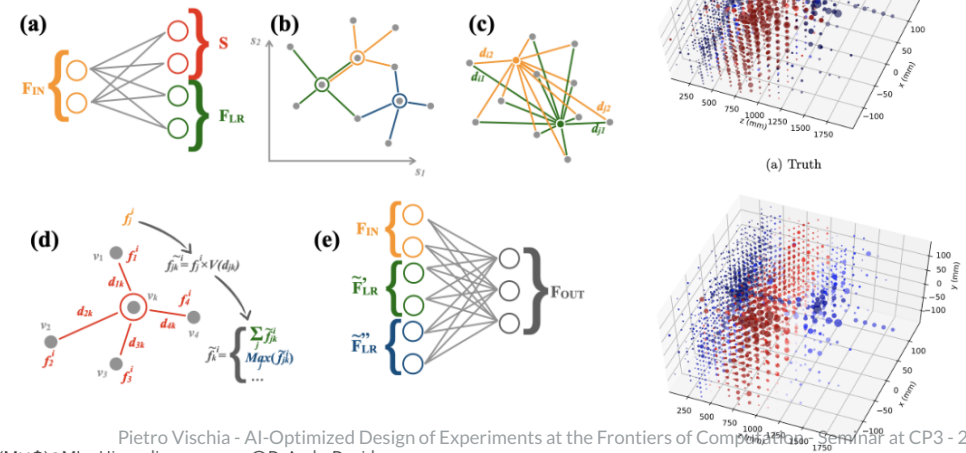
What is a calorimeter?

- Scintillating material (e.g. polystyrene, $PbWO_4$) producing light
- Readout (e.g. silicon photomultipliers)
- Absorber material (Iron, Lead)
- **Size of the scintillating elements drives resolution**



- **Increasing the granularity is costly**
 - E.g. **CMS spent 67M CHF to upgrade** in 2026-2030 the endcap calorimeter
 - High Granularity Calorimeter (HGCAL): **6 million** Si channels, 250000 scintillator tiles readout by SiPMs,
 - First imaging calorimeter at a high-PU hadron collider
 - Precision particle flow reconstruction (sensitivity to VBF and VBS improves, more precise jet substructure studies)
 - Timing extends reach for long-lived particles

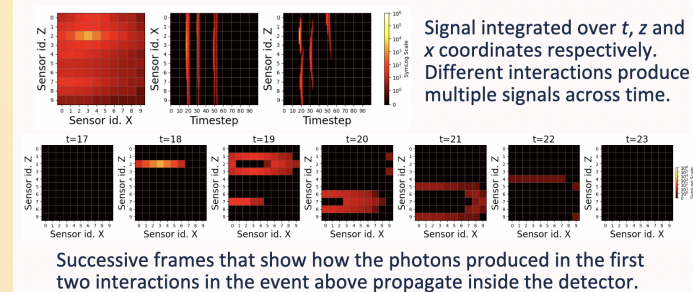
Learning representations of irregular particle-detector geometry with distance-weighted graph networks



Neuromorphic Readout for HCals E. Lupi et al. (P.V.) *Particles* 2025 8(2) 52

- Hadron Calorimeter readout via light-sensitive nanowires
 - Fast, energy-efficient local computation
 - Generate informative high-level primitives
- GEANT4 simulation of 100-GeV $p/K/\pi$
- Simulated calorimeter: 1000 cubelets, 1M nanowire cells
 - $300 \times 300 \times 1200 \text{mm}^3$: lateral/longitudinal containment of 100% (87%)
 - Record every 0.2ns for 20ns (90% of the shower is deposited)
- Time evolution accessible via spiking networks
 - **Bypass the need for high-granularity calorimeters**

Photons are collected for a total of 20 ns and the signal is discretized into 100 bins. Here is how one example event looks like:



The detector is divided into blocks called “**cubelets**”:

- Arranged in a $10 \times 10 \times 10$ matrix
- Size: 3 cm x 3 cm x 12 cm

Here is a schematic view of one cubelet...

Material of choice: **PWO**

- Light Yield $\approx 220 \text{ ph/MeV}$
- Refraction index = 2.2

Incoming particle: **p, π or k**
at 100 GeV

Segmented readout: **10×10 light sensors grid** on the upper face of each cubelet. Sensors are blind to the light coming from other cubelets (all other sides are reflective)

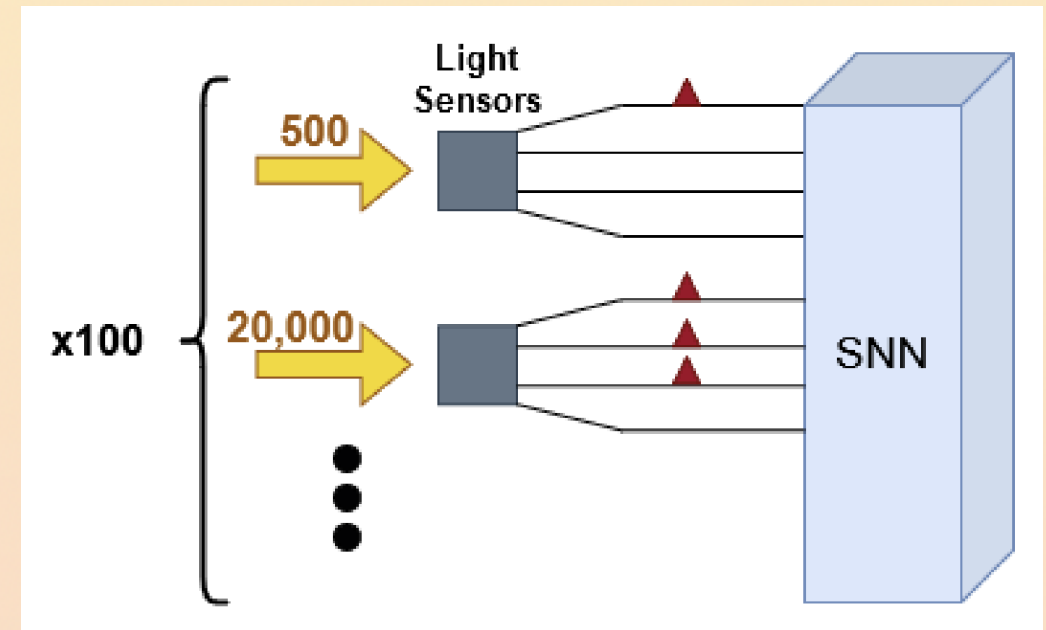
Simple assumption:
All deposited energy is converted into **photons** which travel **isotropically** in all directions

Encoding sensor data E. Lupi et al. (P.V.) *Particles* 2025 8(2) 52

- Each sensor feeds four channels to a spiking network → have to convert incoming data into a spike train
- For each time step t , each channel can either be activated (carry a spike) or inactivated
 - Define threshold on incoming light that activates the channel
- Each channel has a different activation threshold → spikes carry both timing and intensity information
- For the i -th channel and t -th timestep is the following:

$$S^i[t] = \begin{cases} 1 & \text{if } N_{ph}[t] \geq 10^{i+2} \\ 0 & \text{otherwise} \end{cases}$$

$i = 1, 2, 3$



Decoding scheme E. Lupi et al. (P.V.) *Particles* 2025 8(2) 52

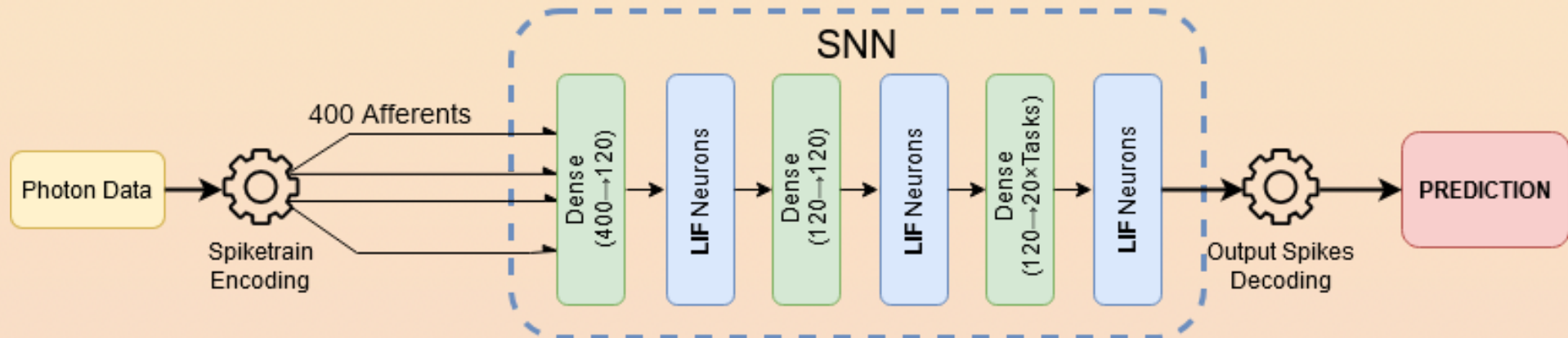
- Decode spike train via fully connected SNN out of LIF neurons
 - LIF neurons with learnable U_{thr} and β , and **arctan** function as surrogate gradient.
- Last layer of SNNs natural choice as a regressor (either output spikes or membrane potential)

Membrane potential: higher precision, higher latency and computational complexity

Output spikes: requires careful information encoding/decoding schemes (e.g. rate- or latency-based), but better performance

$$\hat{X} = \frac{1}{N_{pop}} \sum_{i=1}^{N_{pop}} U^{(i)} [T]$$

$$\hat{X} = \frac{1}{N_{pop}} \sum_{i=1}^{N_{pop}} \sum_{t=1}^T U^{(i)} [t]$$



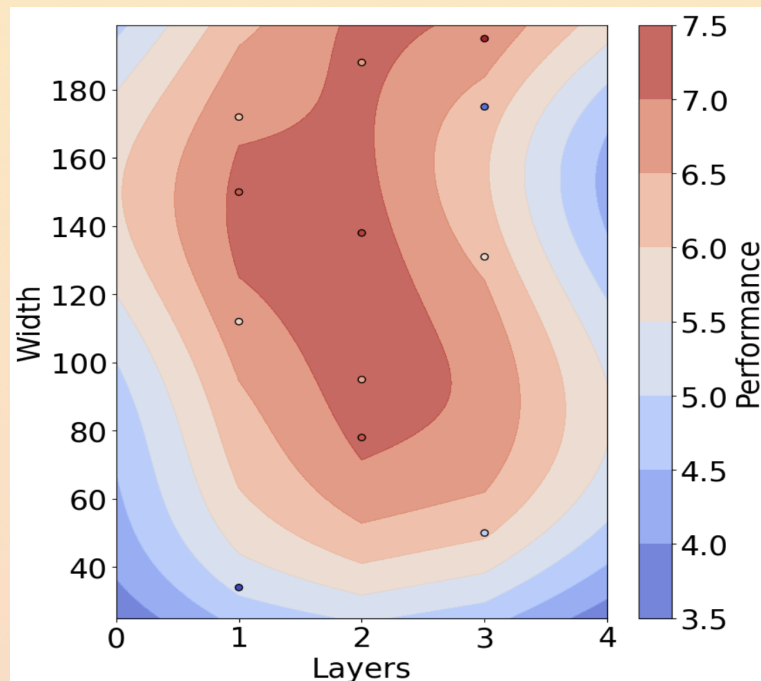
Network optimization

E. Lupi et al. (P.V.) *Particles* 2025 8(2) 52

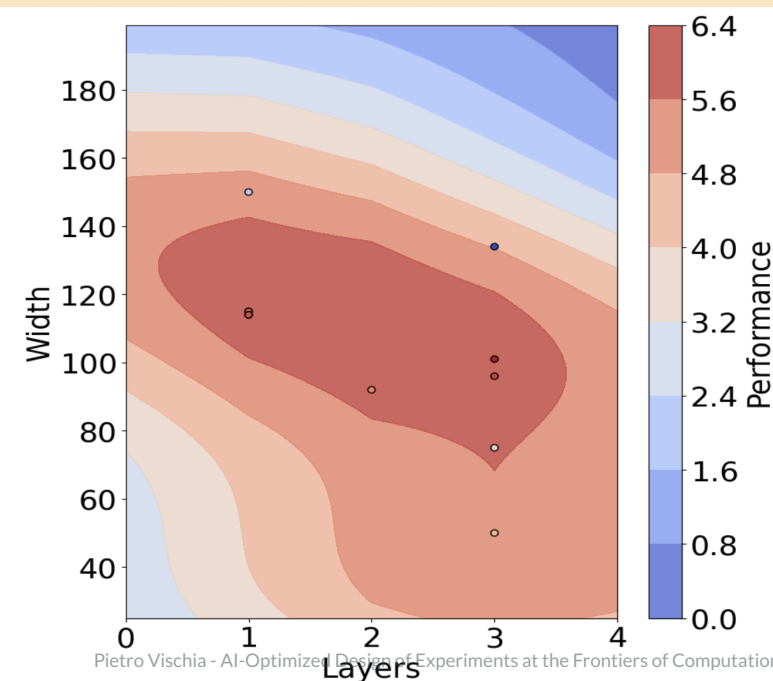
- Architecture optimized via Bayesian optimization, separately for each multi-target regression
 - Prior > gaussian surrogate model of the objective function
 - New observations generated maximizing Expected Improvement → used to update the prior in the function space
 - Sampling directed towards the maximum of the surrogate model (Conditioning on sampling towards maxima.
 - Access to full posterior of the performance P over the parameters

$$P = \frac{1}{N_{targets}} \sum_i^{N_{targets}} \frac{1}{\epsilon_i}$$

Regress log(E/MeV) and centroid



Regress log(E/MeV) and dispersion



Regression E. Lupi et al. (P.V.) Particles 2025 8(2) 52

- Single- and multi-target regression give **consistent results**
- Centroid of energy distribution determined to **less-than-one-cell accuracy**
- Large energy range results in nonoptimal error
- z dispersion more precise than x and y

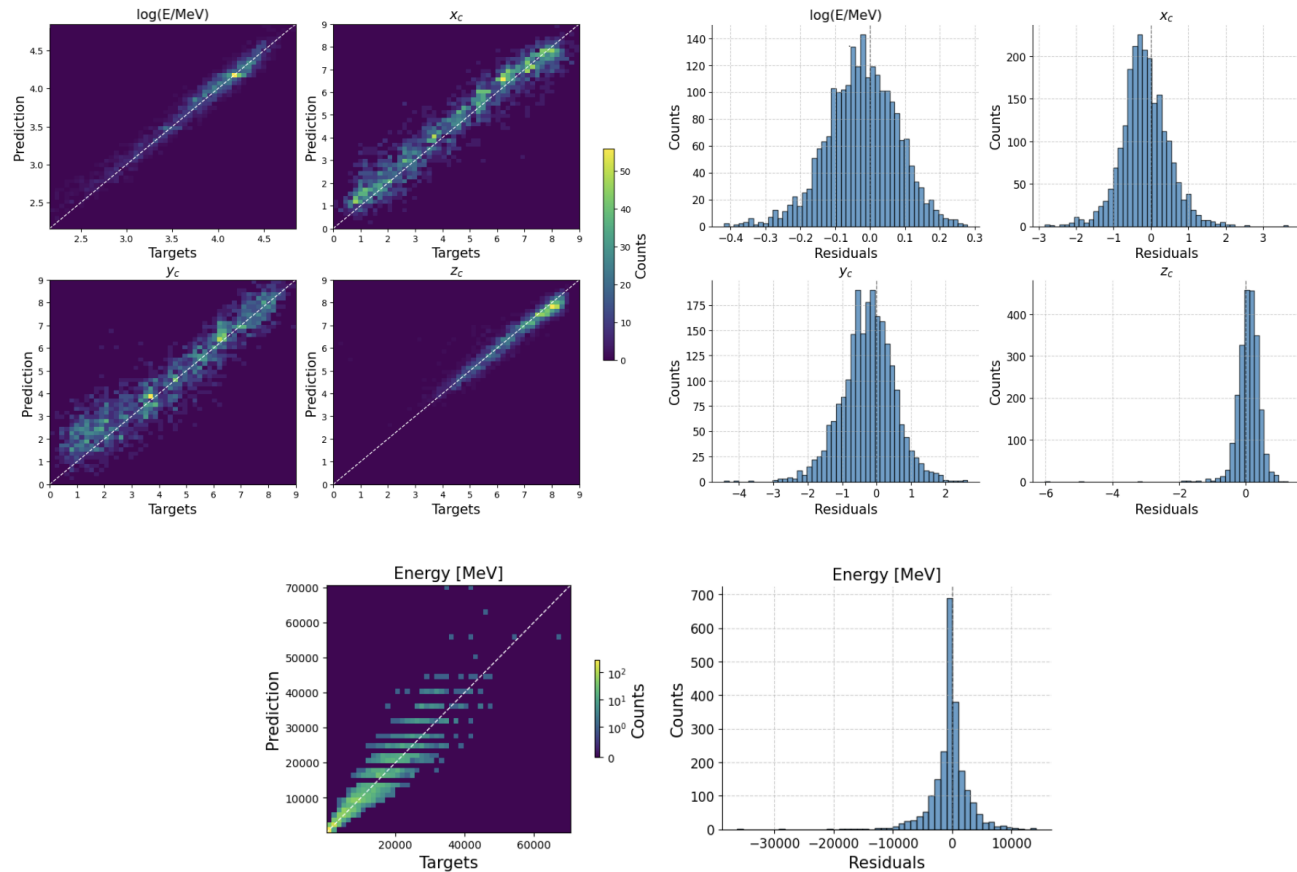
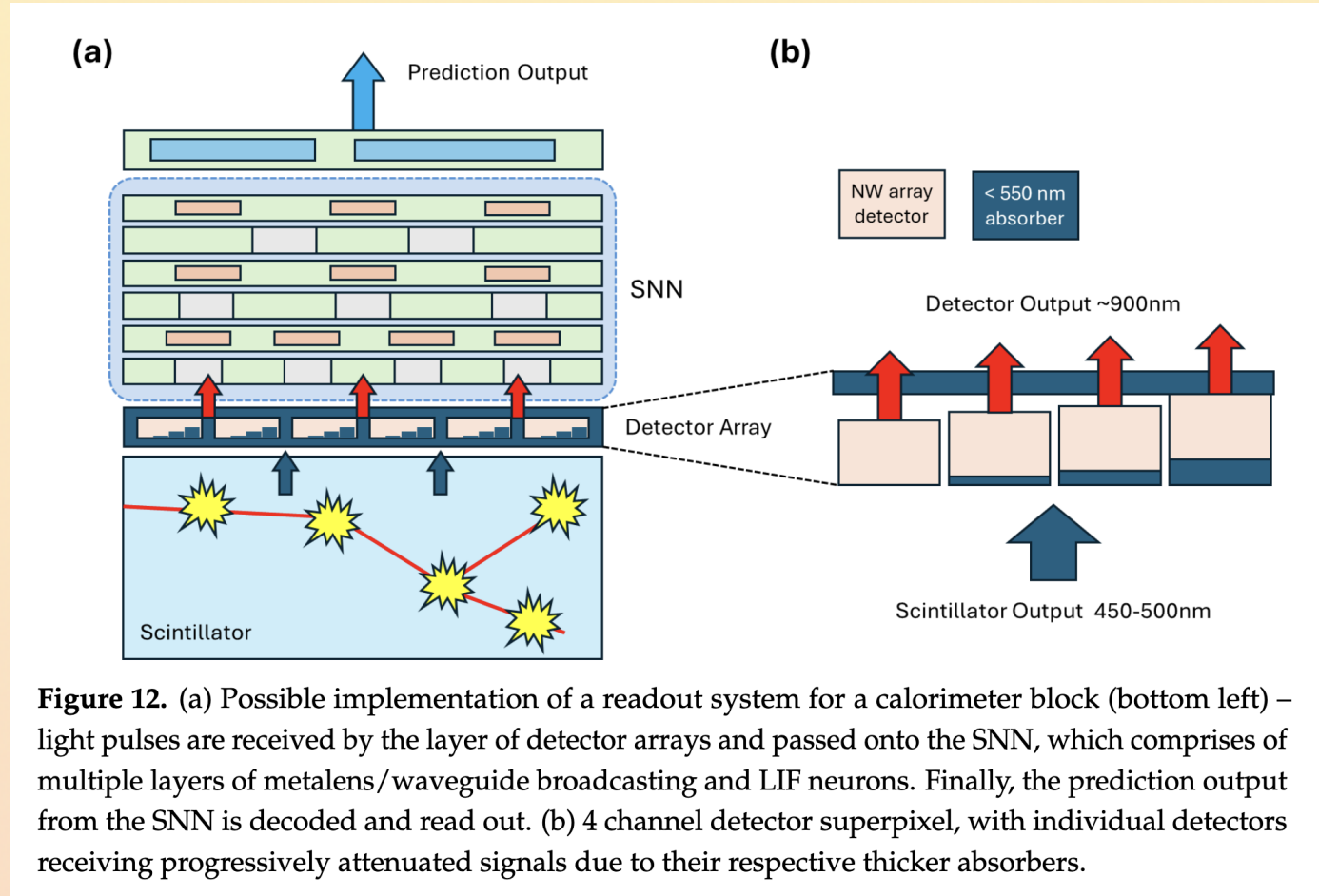


Figure 8. Output of the energy deposition and centroid network. The plots show the correlation between targets and predictions (left) and the residuals (right).

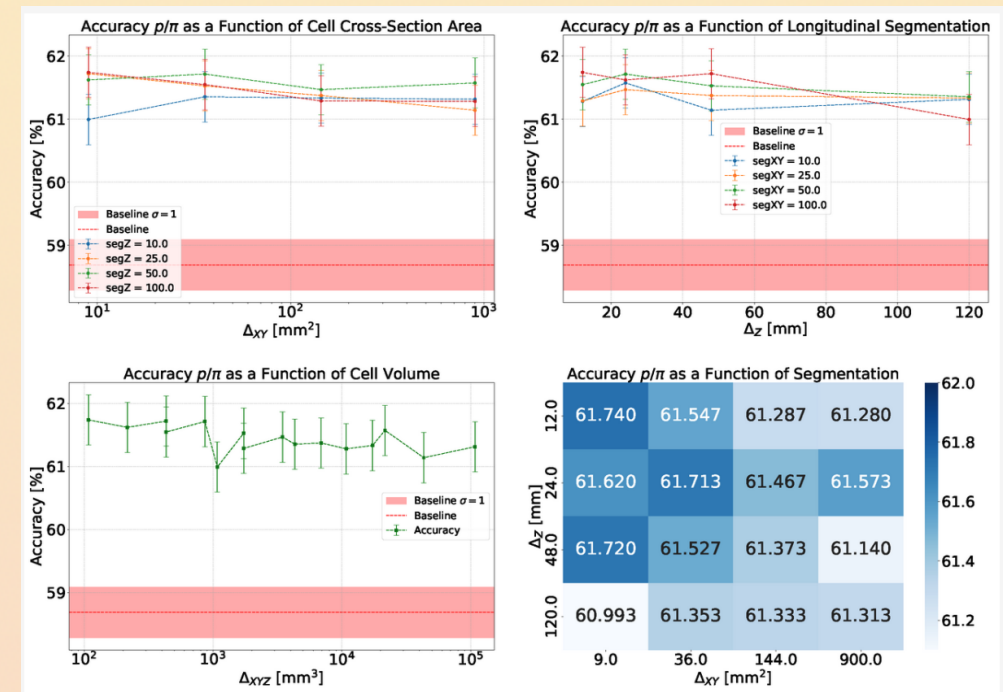
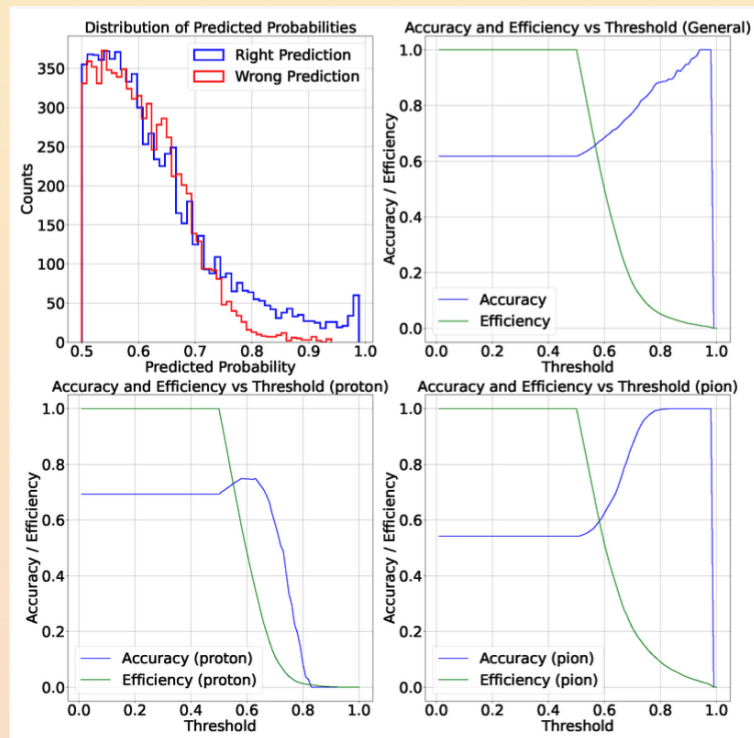
Nanophotonics implementation E. Lupi et al. (P.V.) *Particles* 2025 8(2) 52

- Most NC systems operate at 200 MHz (5ns cycle)
 - Required sub-ns (0.2ns) sampling implies challenges
- III-V semiconductor nanowire: detect and emit light, work as electronics
 - Spiking photonic neurons in the picoseconds
 - InP can detect light in the 450-550 nm bandwidth of $PbWO_4$
- Large energy range results in nonoptimal error
- z dispersion more precise than x and y

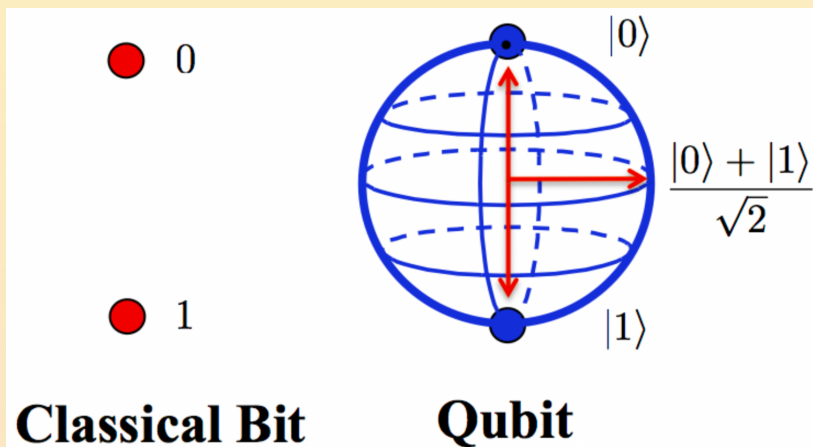


Why is segmentation useful? A. De Vita et al. (P.V.) Particles 2025, 8(2), 58

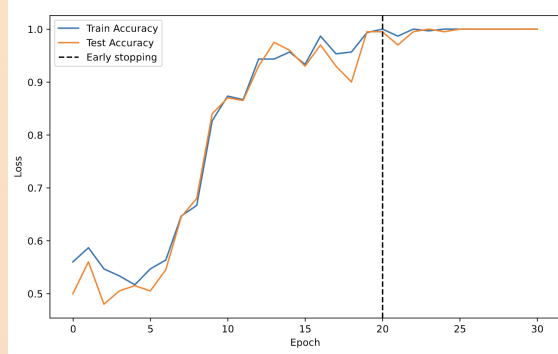
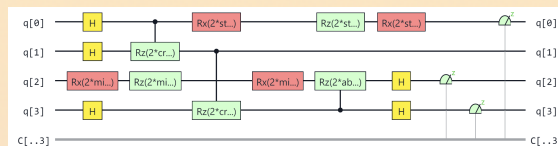
- High granularity opens the possibility of separating $p/K/\pi$
- Can define binary decisions with $\mathcal{O}(60\%)$ accuracy at 100% efficiency
- Neuromorphic implementation
 - can do this within the chips themselves!!!
 - can do that without increasing granularity of the calorimeter!!! (only the readout)



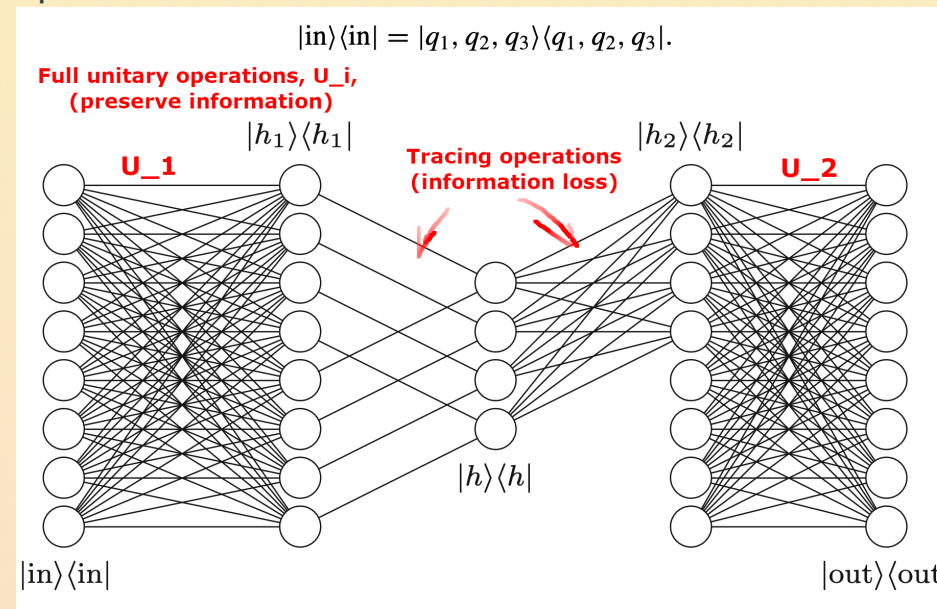
Differentiable quantum optimization



- 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

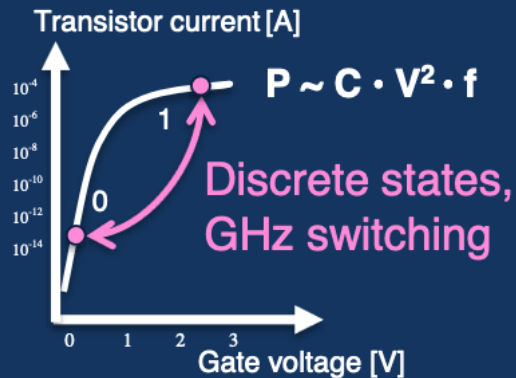
$$\begin{aligned}
 \partial_{\mu} \langle \psi(x, \theta) | \sigma_z | \psi(x, \theta) \rangle &= \langle 0 | \dots \partial_{\mu} e^{-i\mu\sigma} \dots \sigma_z \dots e^{i\mu\sigma} \dots | 0 \rangle \\
 &\quad + \langle 0 | \dots e^{-i\mu\sigma} \dots \sigma_z \dots \partial_{\mu} e^{i\mu\sigma} \dots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

Need for new paradigm

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

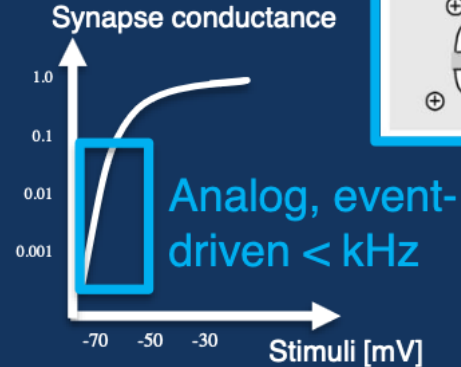
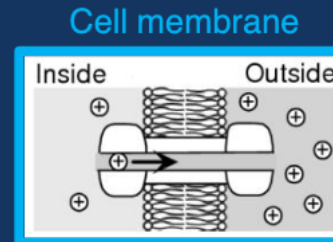
Conventional computers

*mimic
logical and
analytical
thinking*

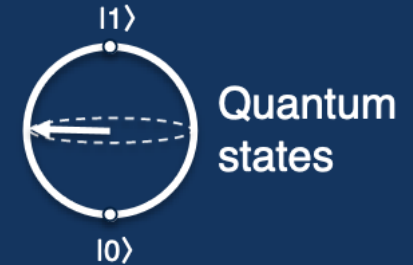


Neuromorphic processors

*mimic
the senses,
learning and
perception*



Quantum processors
*use quantum
superpositions
for probabilistic
inference*



Technology readiness?

The MODE Collaboration

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

- Machine-Learning Optimized Design of Experiments
- 52 members, 30 institutions, 13 countries (if you are interested, join us!!!)
- Co-founded in 2020, **coordinator since 11-2024**

At INFN and Università di Padova Dr. **Tommaso Dorigo**, Dr. **Pablo De Castro Manzano**, Dr. **Federica Fanzago**, Dr. **Giles Strong**, and Dr. **Mia Tosi**

At Université catholique de Louvain Dr. **Andrea Giammanco**, Ms. **Zahraa Daher**, and Mr. **Maxime Lagrange**

At Universidad Carlos III de Madrid and ICTEA Dr. **Pietro Vischia**

At University of Turin, Prof. **Julien Donini**, and Mr. **Federico Nardi** (joint with Università 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 la Universidad de Zaragoza, Dr. **Pablo Martínez Ruiz del Árbol**

At Karlsruher Institut für Technologie, Dr. **Jan Kieseler**

At University of Oxford, Dr. **David J. Morley**

At New York University, Prof. **David J. Morley**

At Université de Liège, Prof. **Christophe Driessens**

At GSI/FAIR, Dr. **Anastasios B. Katsouris**

At HEPHY Vienna (OeAW), Dr. **Christian Bock**

At Uppsala Universitet, Prof. **Christina E. Lundberg**

At TU-München, Prof. **Lukas Heinrich**

At Sheffield University, Dr. **Patrick Sullivan**

At Lebanese University, Prof. **Haitam El-Hachimi**

At University of Kaiserslautern-Landau, Dr. **Thomas G. Schaefer**

At Eindhoven University of Technology, Dr. **Thomas G. Schaefer**

At Princeton University, Prof. **Peter Elmer**

At University of Washington, Prof. **Gordon J. Gounis**

At SLAC, Dr. **Ryan Roussel**

At Luleå University of Technology, Prof. **Fredrik K. Johansson**

At IGFAE and Universidad de Santiago de Chile, Dr. **Andrés C. M. de la Torre**

At ISER, Ms. **Sarjeeta Gami**

At Carnegie Mellon University, Prof. **Ann Levent**

At IISER Kolkata, Dr. **Tousik Samoui**

At Sofia University, **Kalina Dimitrova**, Prof. **Peicho Petkov**

At ECAP - FAU Erlangen-Nürnberg, Dr. **Christian Bock**

At NASA, Dr. **Stephen M. Casey**

The Scientific Coordinator of the MODE Collaboration is Prof. **Pietro Vischia**, Universidad de Oviedo and ICTEA

The Steering Board of the MODE Collaboration includes:

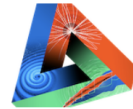
- Prof. **Tommaso Dorigo**, INFN-PD
- Prof. **Nicolas Gauger**, RPTU
- Dr. **Andrea Giammanco**, UCLouvain
- Prof. **Christian Glaser**, UU
- Prof. **Lisa Kusch**, TU/e
- Dr. **Fedor Ratnikov**, HSE

The Eu Coalition for AI

<https://eucaif.org>

- European initiative for advancing the use of AI in Fundamental Physics
 - Community for developing and promoting AI methods for Physics
 - Bring to AI community advances in AI from physics (see e.g. 2024 Nobel prize to Hopfield and Hinton)
 - Had already two conferences (2024, 2025)
 - **Work Package 2: Experiment Design** (leaders: T. Dorigo and P. Vischia)

EUROPEAN COALITION FOR AI IN
FUNDAMENTAL PHYSICS



JENAA
Joint ECFA-NuPECC-APPEC Activities

Thank you! Questions now or at the barbecue on Thursday!



Backup

From POCA to extended POCA

- POCA (POint of Closest Approach) assumes one scattering in one point → 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 → bias!

