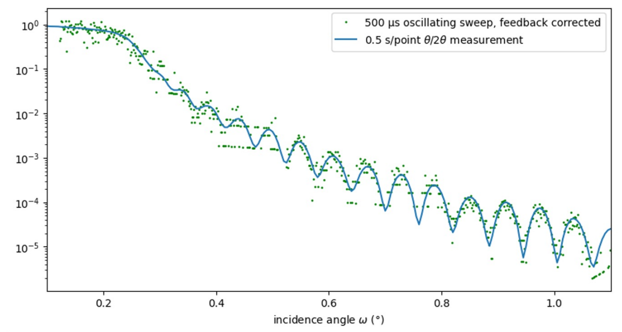
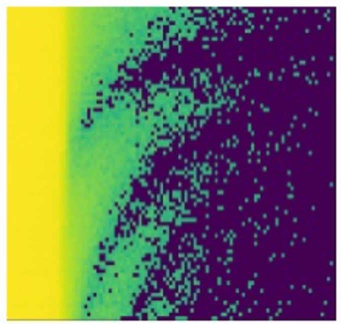
Quick XRR down to 300 µs and fast, accurate AI analysis

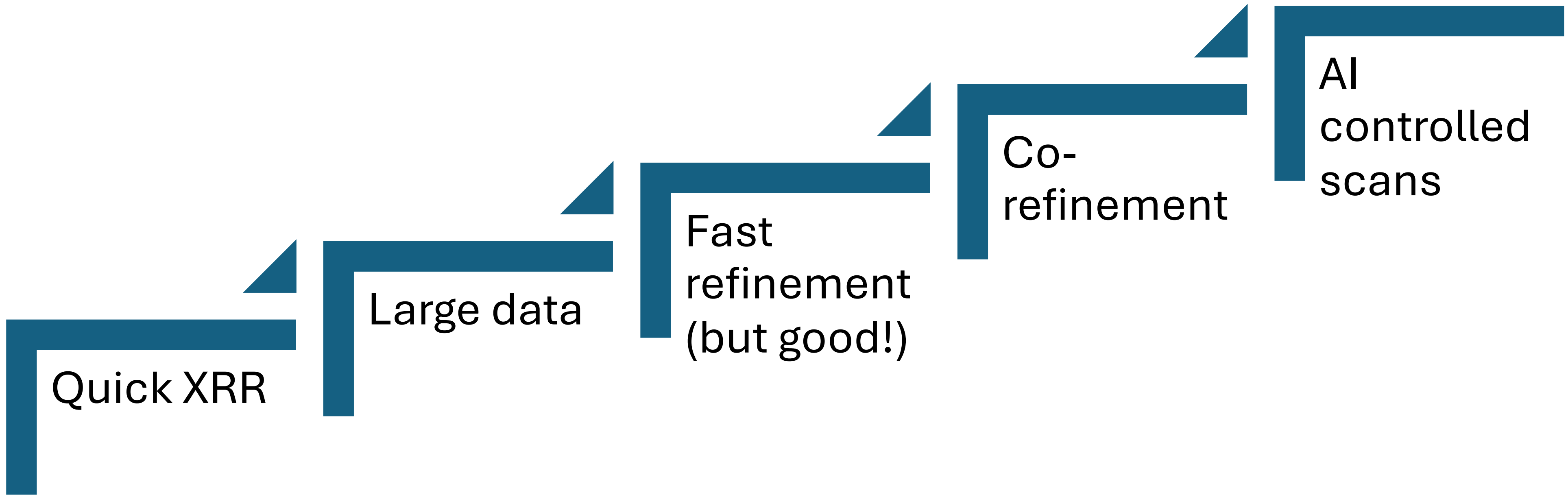
Stefan Kowarik, David Marecek-Schumi, Erwin Pfeiler, Michael Haberl, Frieda Sorgenfrei, Saeid Alirezazadeh, Maximilian Eder, Florian Bertram*

University of Graz, Austria

*DESY, Hamburg, Germany

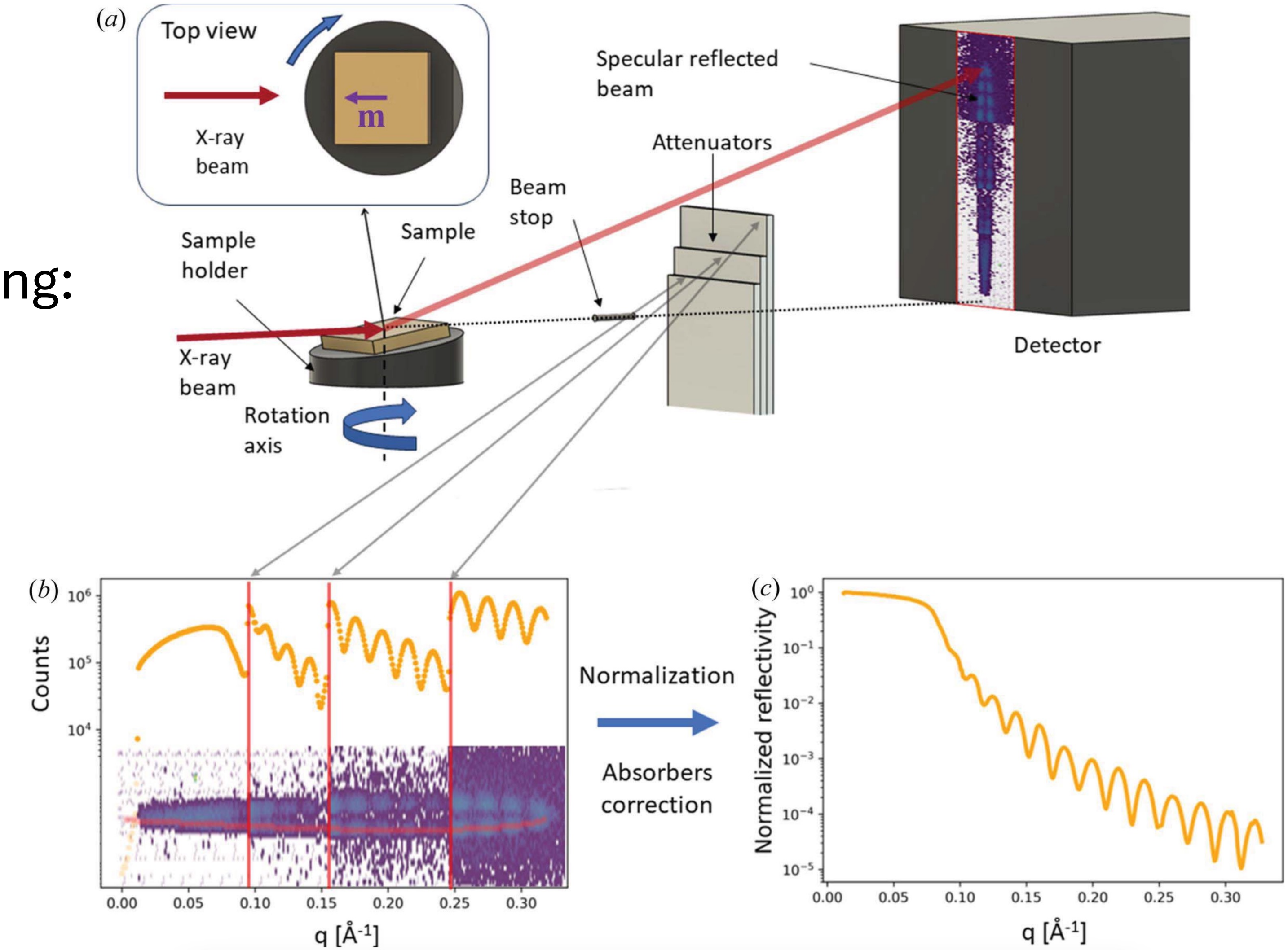





Quick XRR
Large data
Fast refinement (but good!)
Co-refinement
AI controlled scans

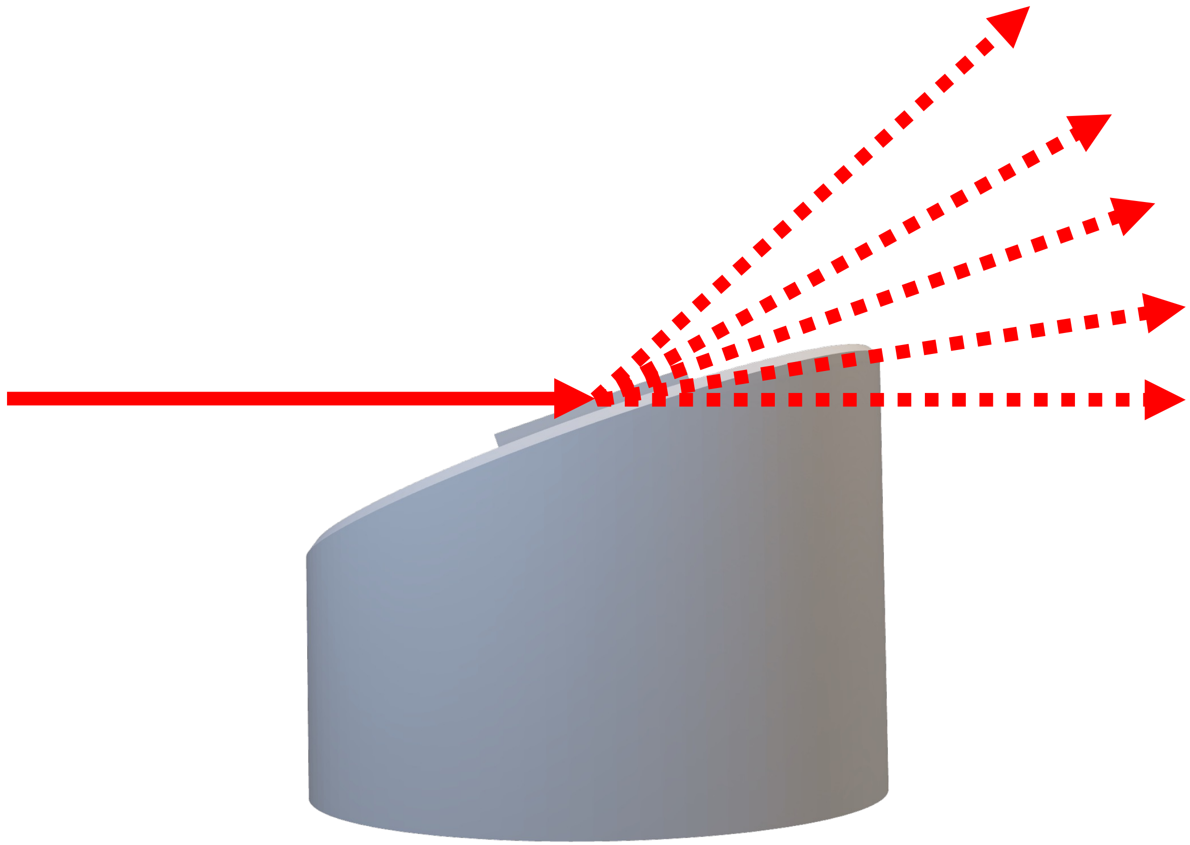
Quick XRR

- fast sample scanning:
  - rotating wedge
  - galvo scanner
- DESY P08 high resolution
- down to 1.4 ms / 270 µs per curve

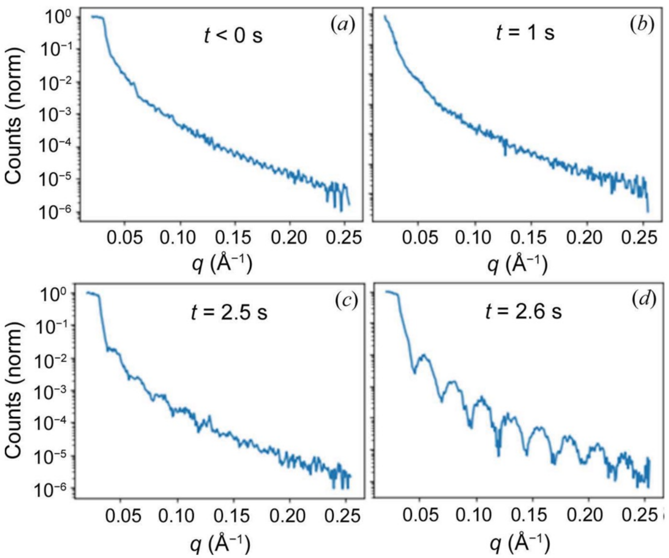


JAC Appl Cryst JOURNAL OF APPLIED CRYSTALLOGRAPHY

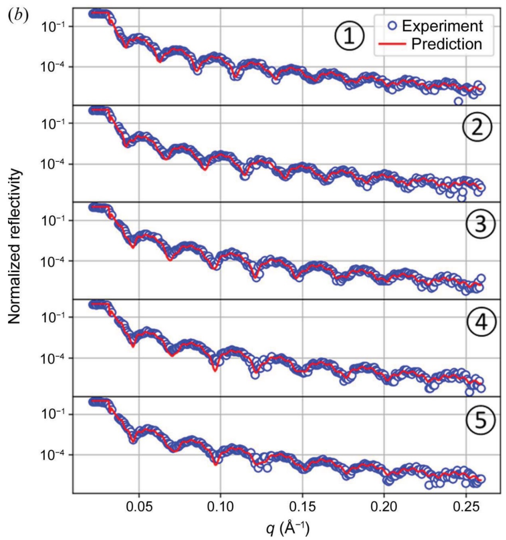
Schumi-Mareček *et al.*
Volume 57 | **Part 2** | April 2024 | Pages 314–323 | 10.11107/S1600576724001171

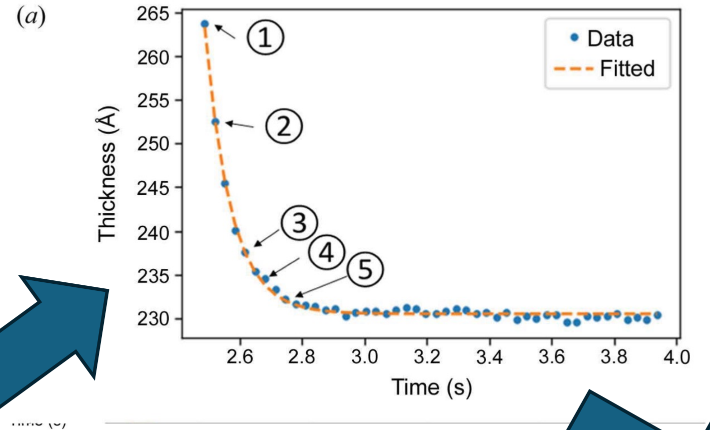


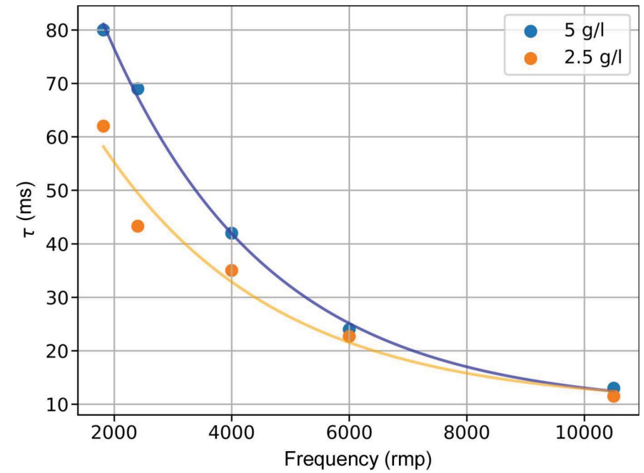
Resolving mass transport in spin coating with XRR



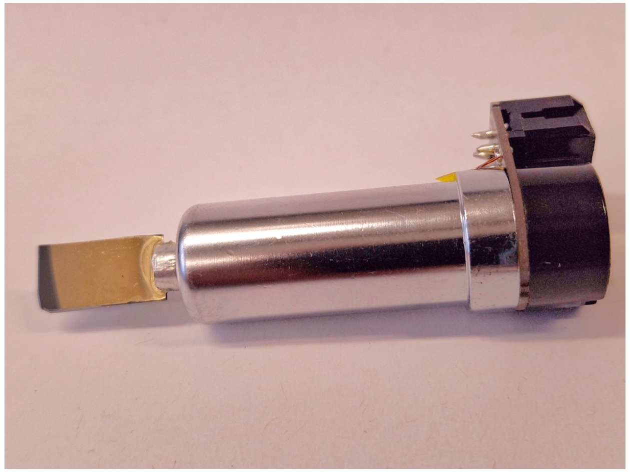
neural network



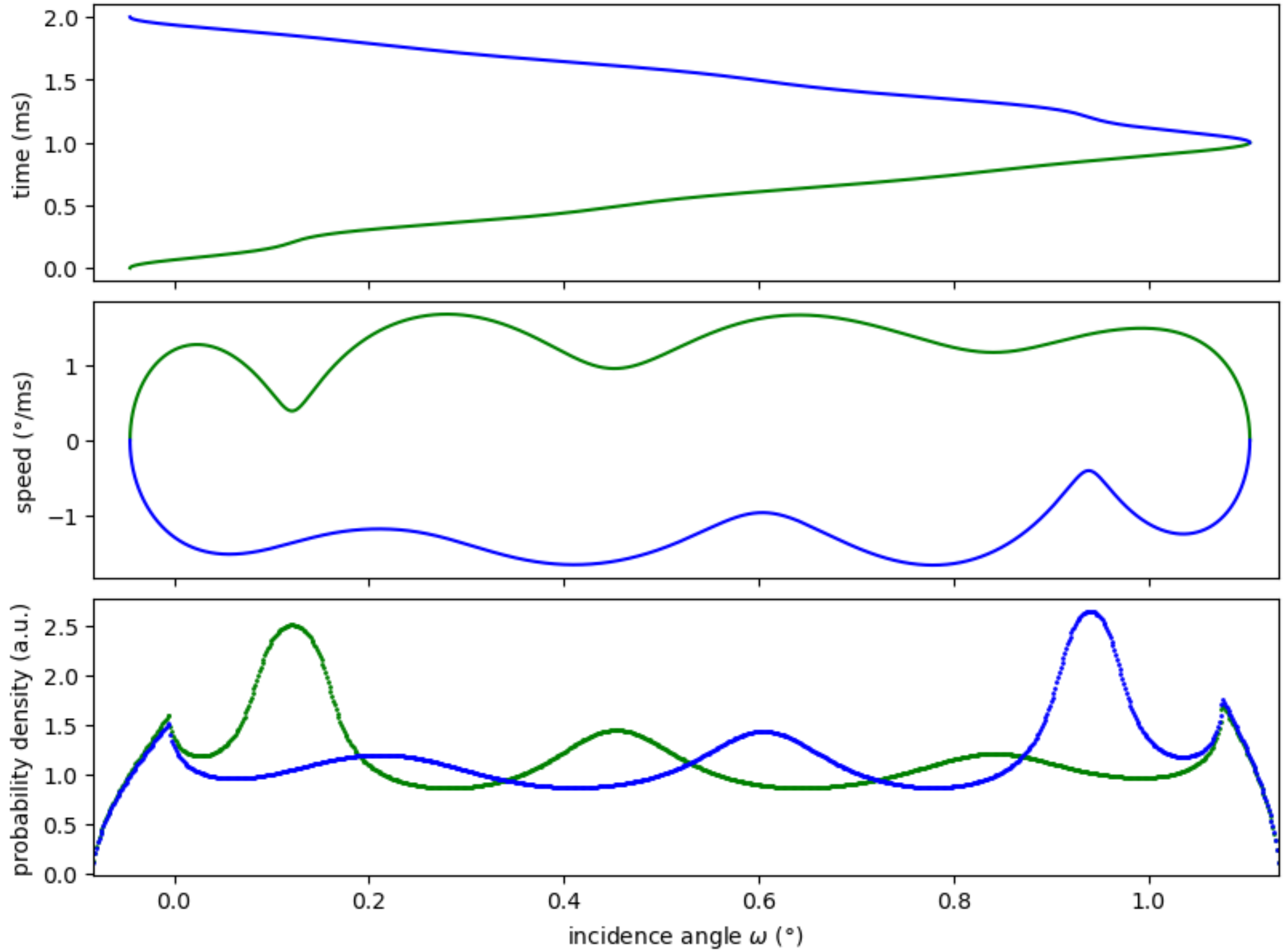




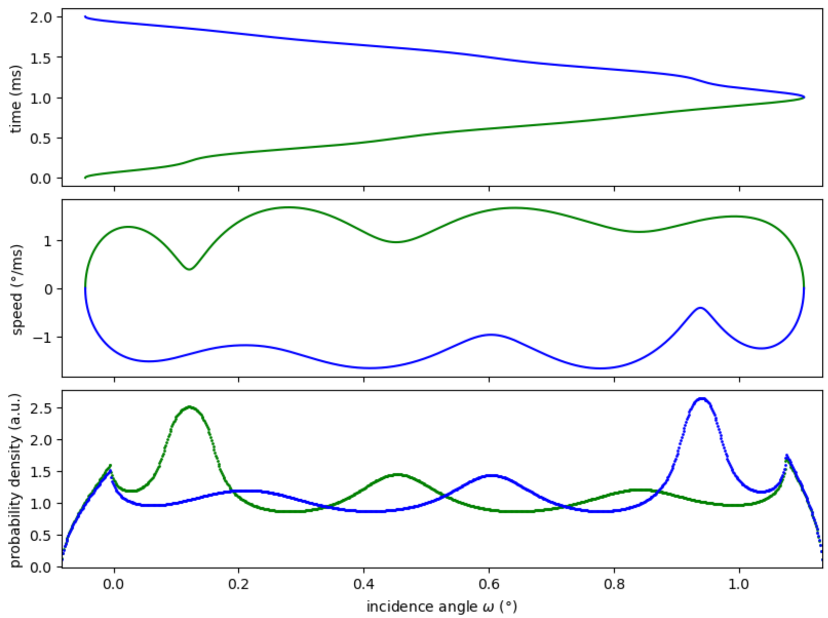
Quick XRR – into the µs range

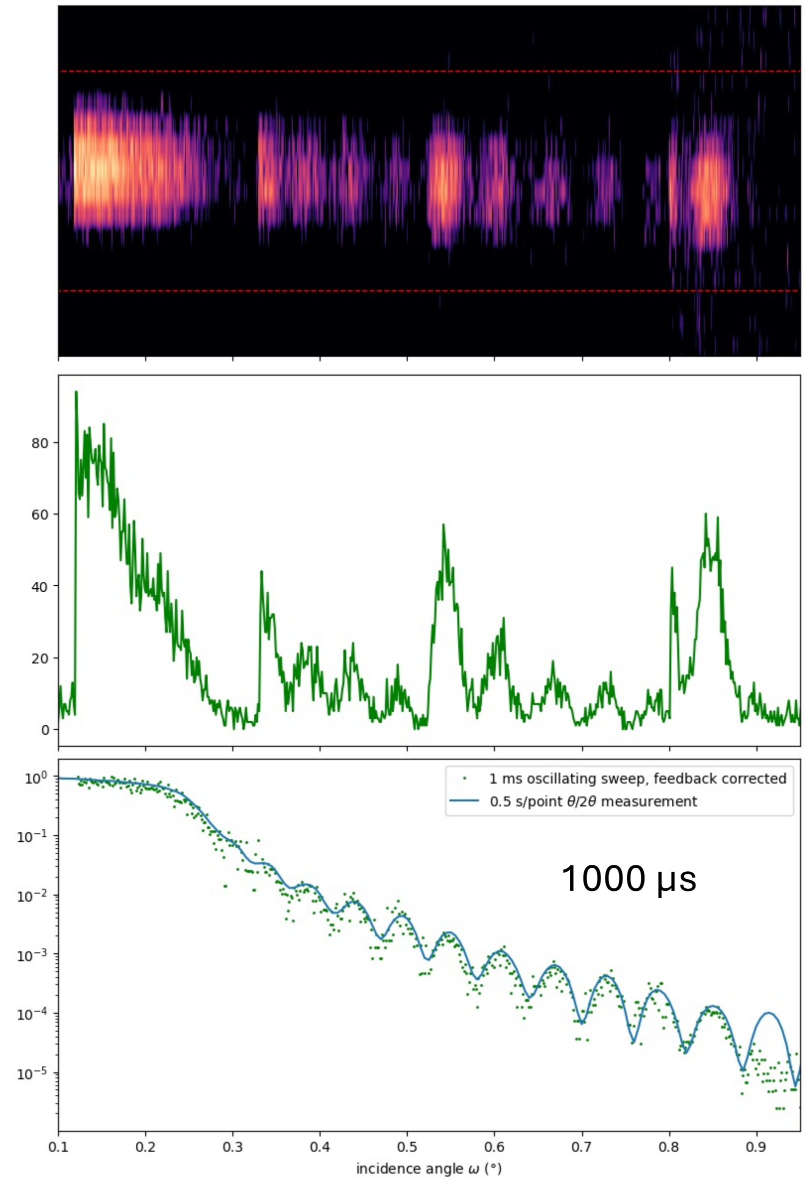


Galvo Mirror

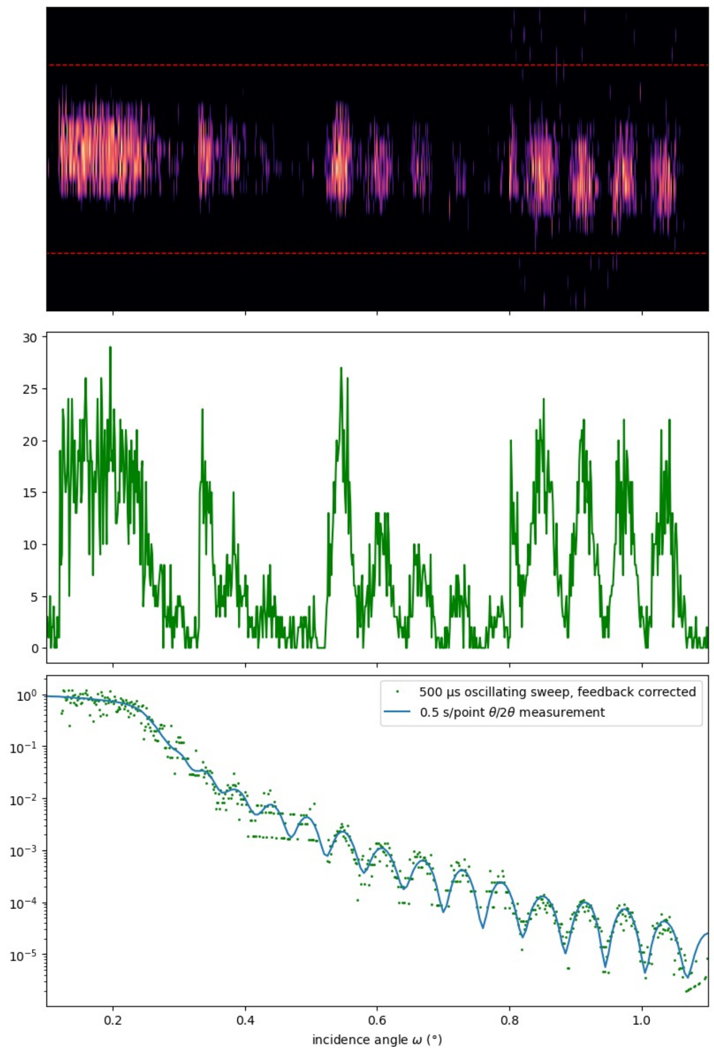


Quick XRR 1 ms:



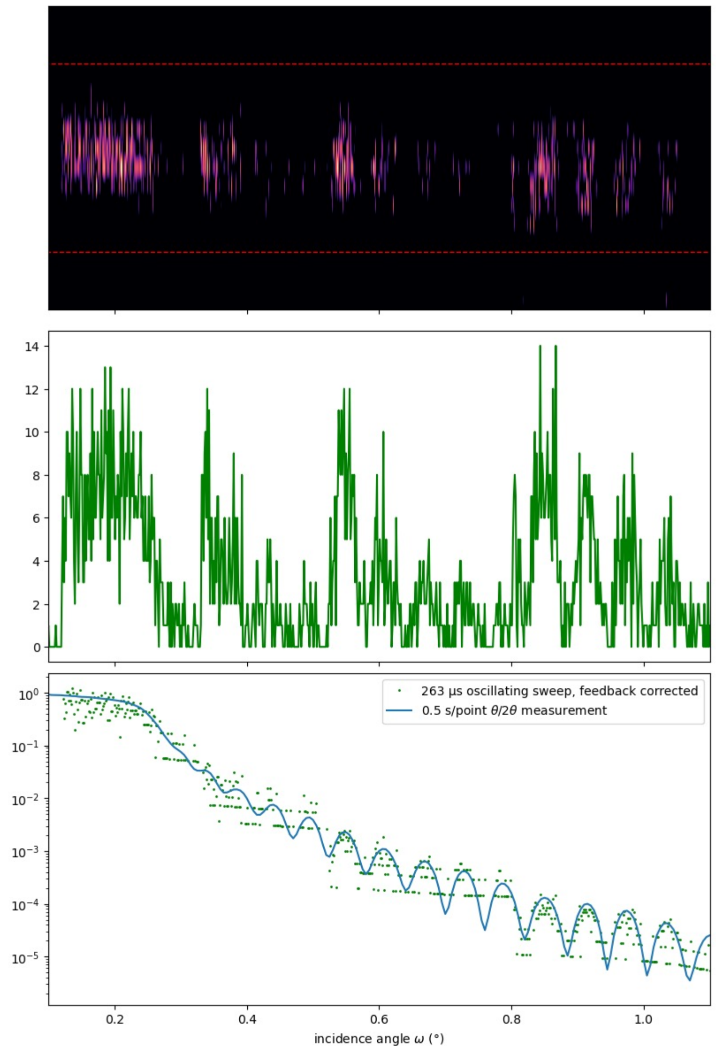


500 µs



500 µs oscillating sweep, feedback corrected
0.5 s/point θ/2θ measurement
incidence angle ω (°)

263 µs



263 µs oscillating sweep, feedback corrected
0.5 s/point θ/2θ measurement
incidence angle ω (°)

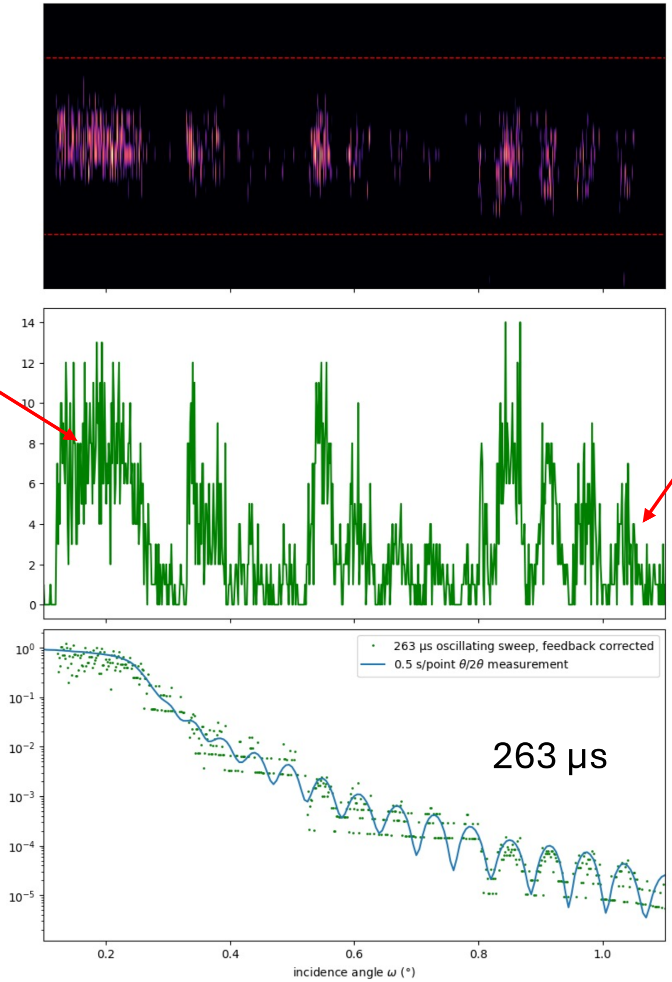
The limits of synchrotron and detector technology

Limited by Eiger 2 photon counting detector:

- XRR = 1000 pixels long
- beam on pixel for 263 ns
- Eiger limit: 1 photon per 100 ns

⇒ max 2-3 photons per pixel

⇒ we want better (integrating) detectors



Limited by Petra III synchrotron:

- 10^{12} photons per second
- XRR: / 10^5
- 263 ns: / $4*10^6$

⇒ 2-3 photons left

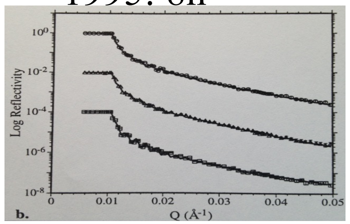
⇒ we want better synchrotrons

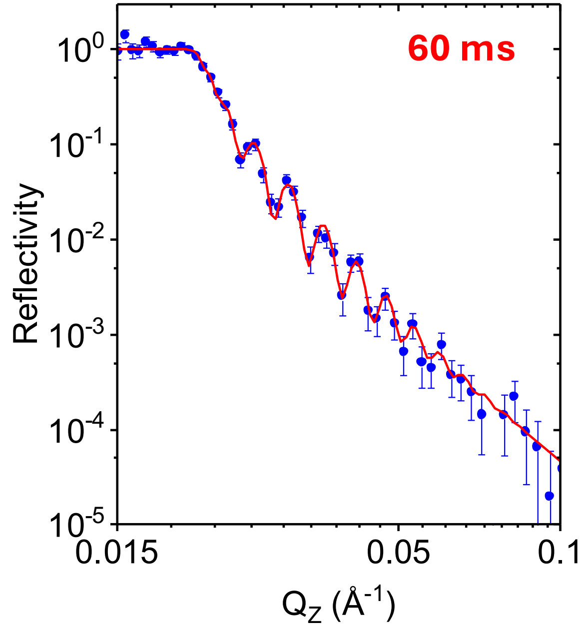
Large datasets

- 1 ms reflectivity for 1 Minute => 60000 XRR curves

  (see https://github.com/DavidMarecek2/Millisecond-XRR for many curves)
- The natural way for any techniques is to produce more and more data!
- **Thomas Saerbeck** @ D17 ILL:
  60 ms NR curves at ILL

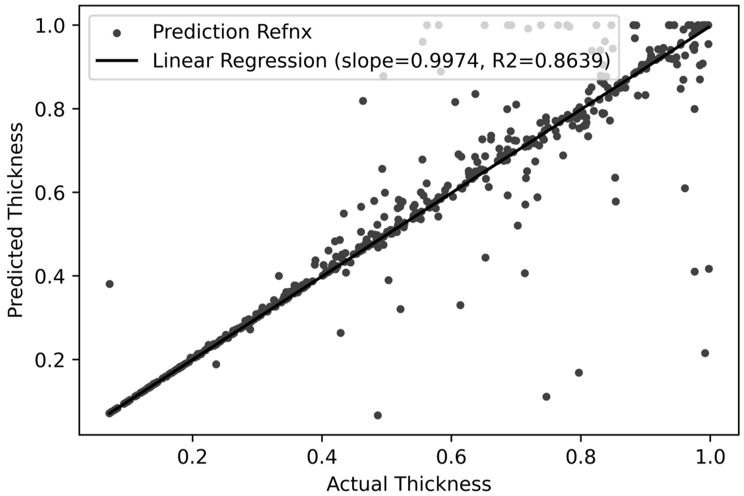
1995: 8h





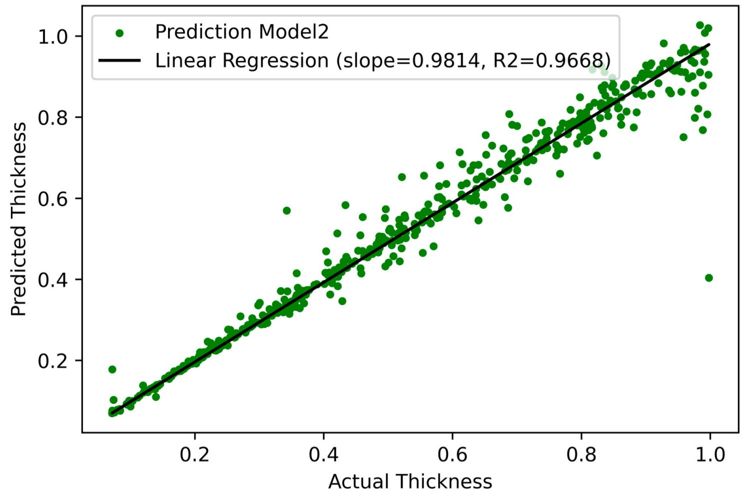
- Synchrotrons among the largest producers of scientific data!

Fast analysis of quick XRR – the new standard?



<u>Genetic algorithm fit</u> of film thickness (synthetic data + refnx)

- **fitting time: > seconds**



<u>Neural network prediction</u> of thickness

prediction time: milliseconds

⇒ **Higher accuracy for NN + least mean square fit**

⇒ **NN + LMS new state of the art?!**

A. Greco ..., F. Schreiber, <u>S. Kowarik</u>, J. Appl. Cryst. (2019). 52, 1342
https://github.com/schreiber-lab/mlreflect

Any way to make interoperable modules / python libraries for XRR fitting?

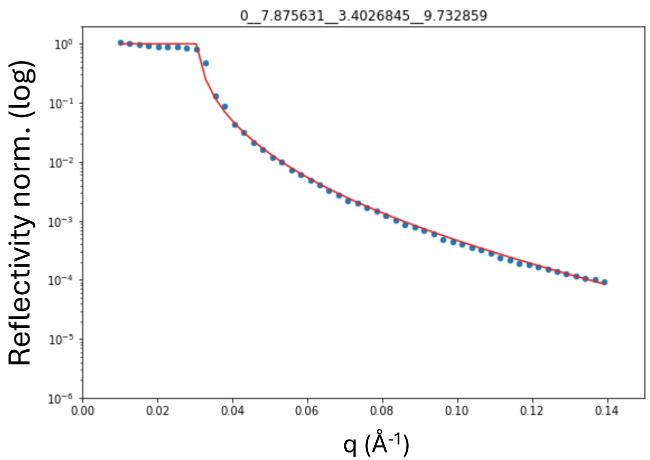
- refnx - Andrew Nelson
- genx - Artur Glavic
- refl1d - Brian Maranville
- RasCal - Arwel Hughes
- mlreflect – V. Starostin
- anaklasis - Alexandros Koutsioumpas
- BornAgain - Joachim Wuttke
- easyReflectivity - Andrew McCluskey

⇒ user facing standardisation
⇒ under the hood standardisation
⇒ ORSO file formats, model language & python

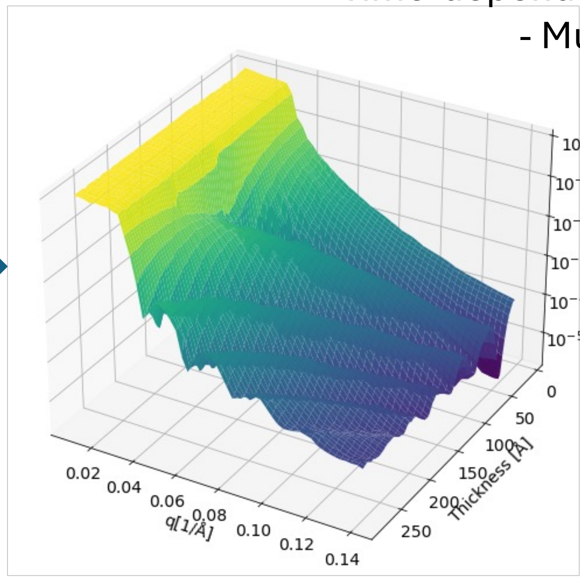
Co-Refinement of large(ish) data

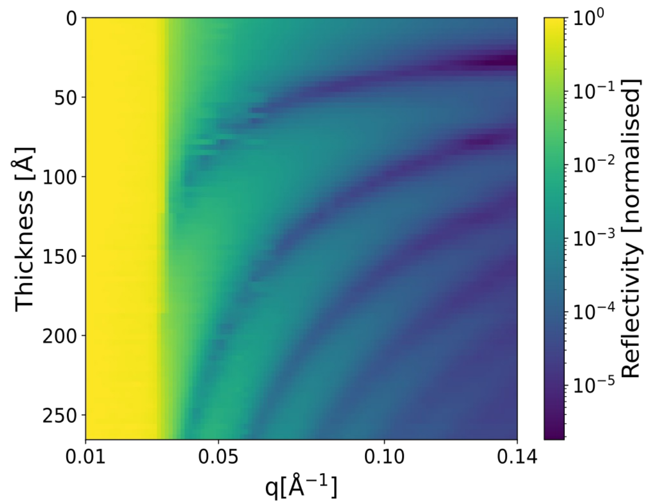
In situ XRR data of thin film growth

Dense neural network fit
- single curve -



Time-dependent X-ray reflectivity data
- Multiple curves -



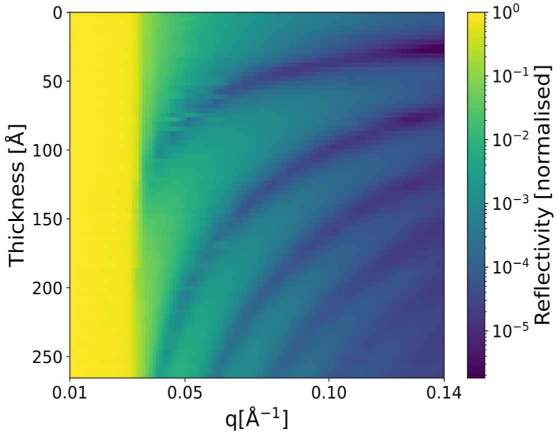


Individual curve fits:

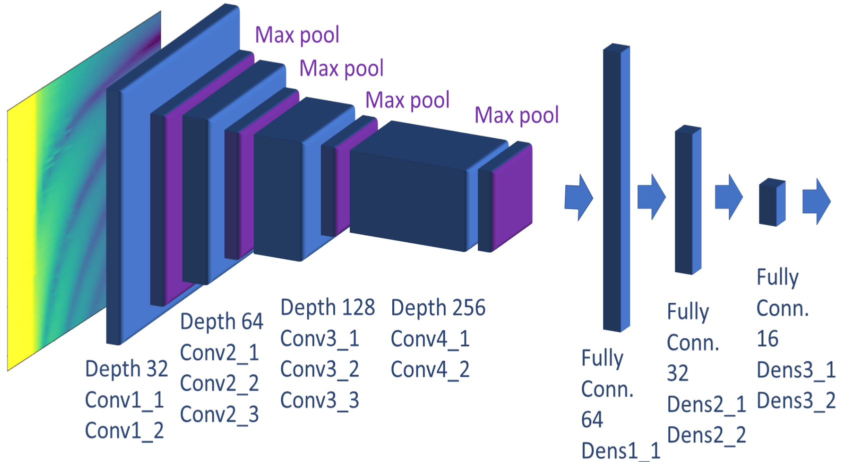
=> no co-refinement | => no physics informed model | => better fit $R(q,t)$!

(A. Greco ..., S. Kowarik, J. Appl. Cryst. (2019). 52, 1342–1347)

A growth model fit can be implemented with a CNN and refnx



240 parameter → 10 growth model parameters



Growth model parameters (growth rate, layer coverages) prediction

Crystalline lattice planes:

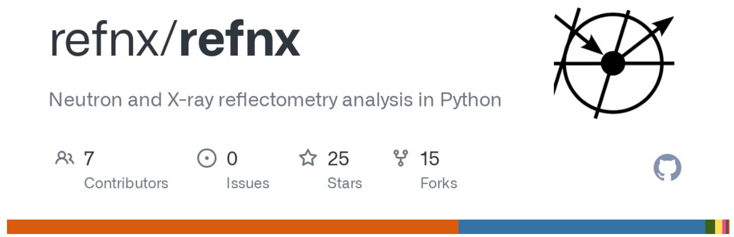
$$\frac{d\theta_n}{dt} = \begin{cases} R_1(1-\theta_1) + R_{n>1}(\theta_1-\xi_1), & n=1, \\ R_{n>1}(\xi_{n-1}-\xi_n), & n>1 \end{cases}$$

Feeding zones for island growth:

$$\xi_n = \begin{cases} 0, & \theta_n<\theta_{n,\mathrm{cr}}, \\ 1-\exp(-\{[-\ln(1-\theta_n)]^{1/2} - [-\ln(1-\theta_{n,\mathrm{cr}})]^{1/2}\}^2) \end{cases}$$

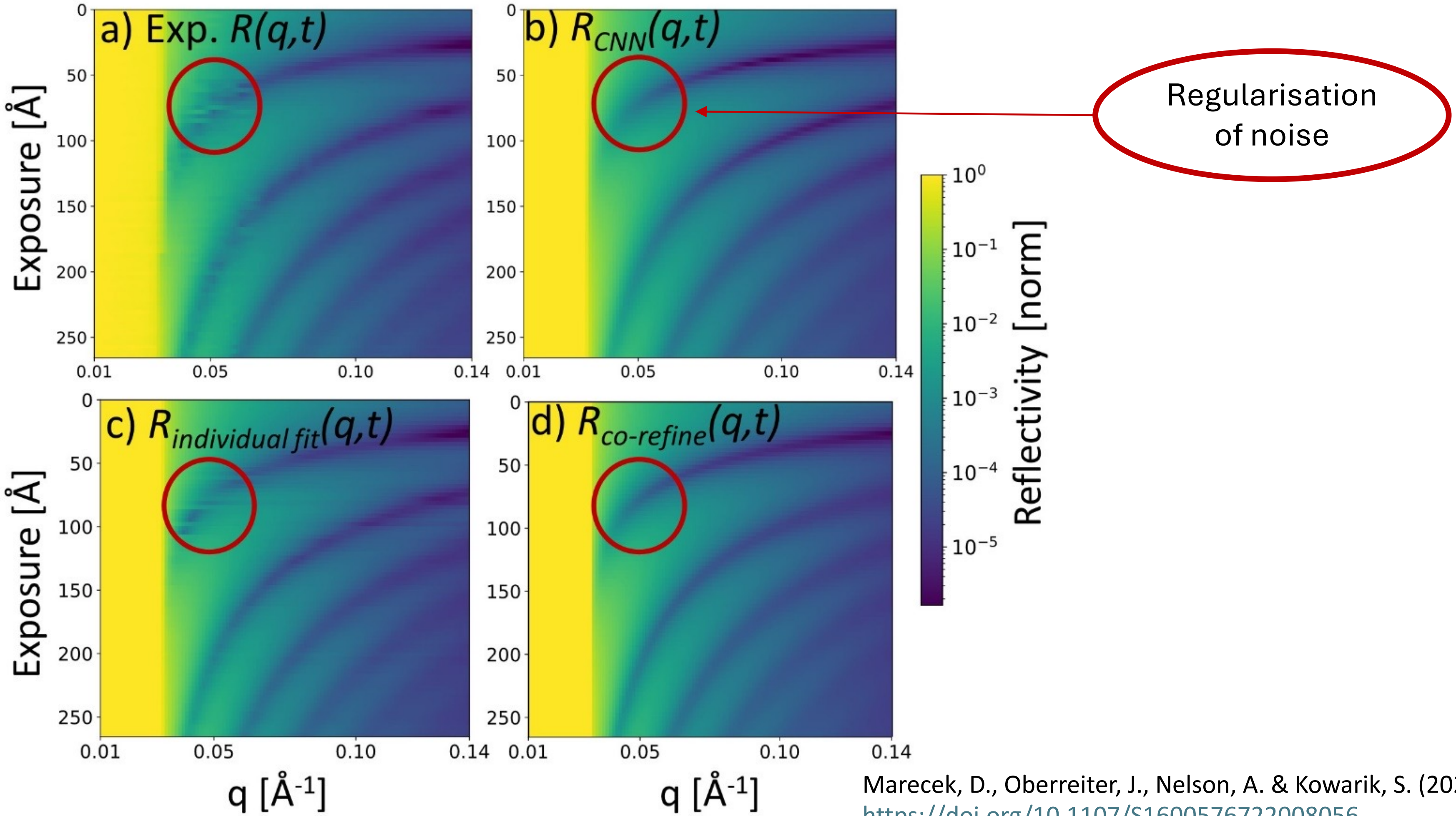
Critical island size for nucleation:

$$\theta_{n,\mathrm{cr}} = \begin{cases} a\dfrac{0.5\tanh[-0.5(t+b)]+0.5}{0.5\tanh(-0.5b)+0.5}, & \theta_{n,\mathrm{cr}}\ge c \\ (c-f)\exp[-d(t-t_c)]+f, & \theta_{n,\mathrm{cr}}<c. \end{cases}$$



Growth model parameters fitting

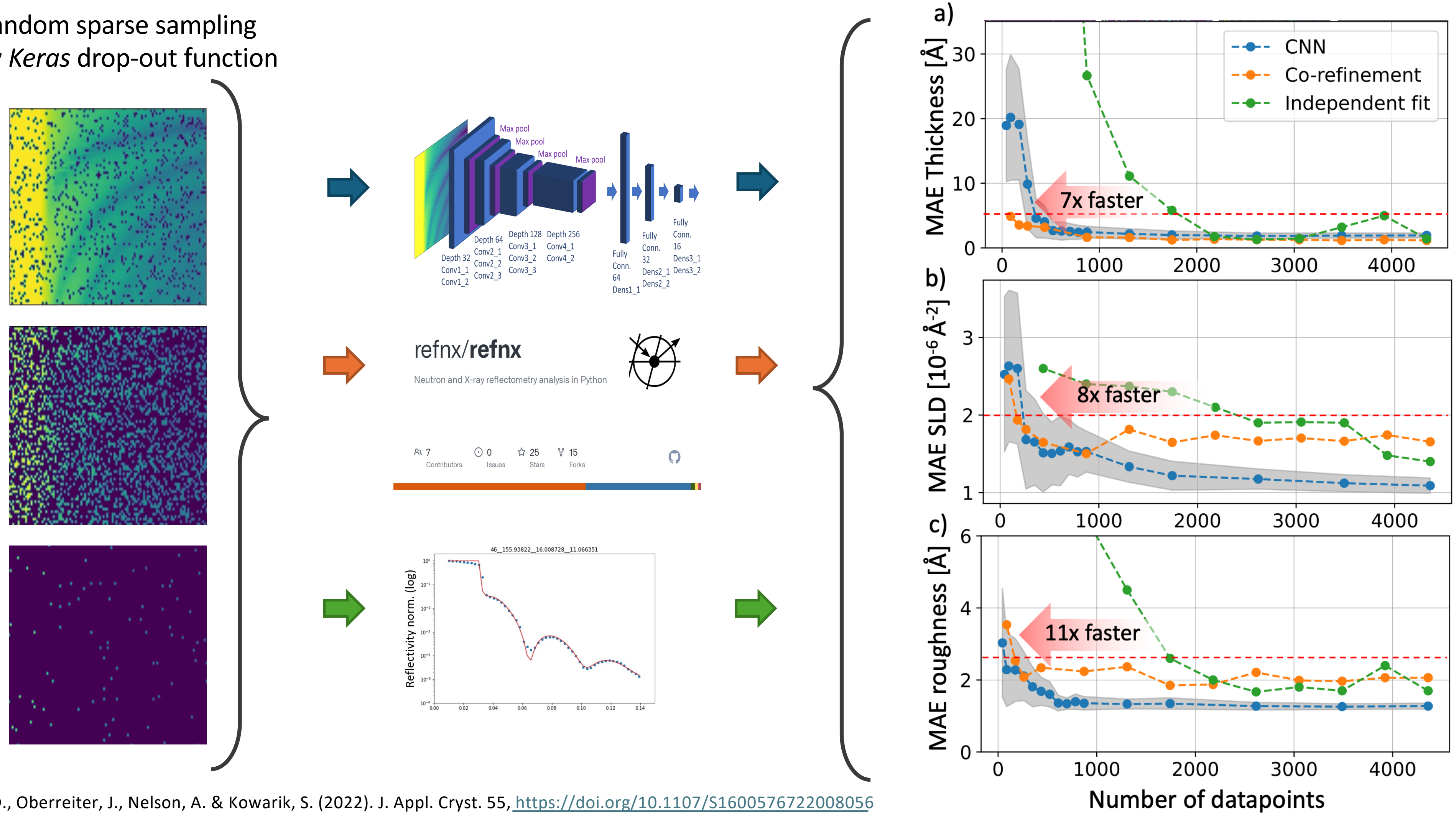
Fitting / prediction with physics model works!



Marecek, D., Oberreiter, J., Nelson, A. & Kowarik, S. (2022). J. Appl. Cryst. 55, https://doi.org/10.1107/S1600576722008056

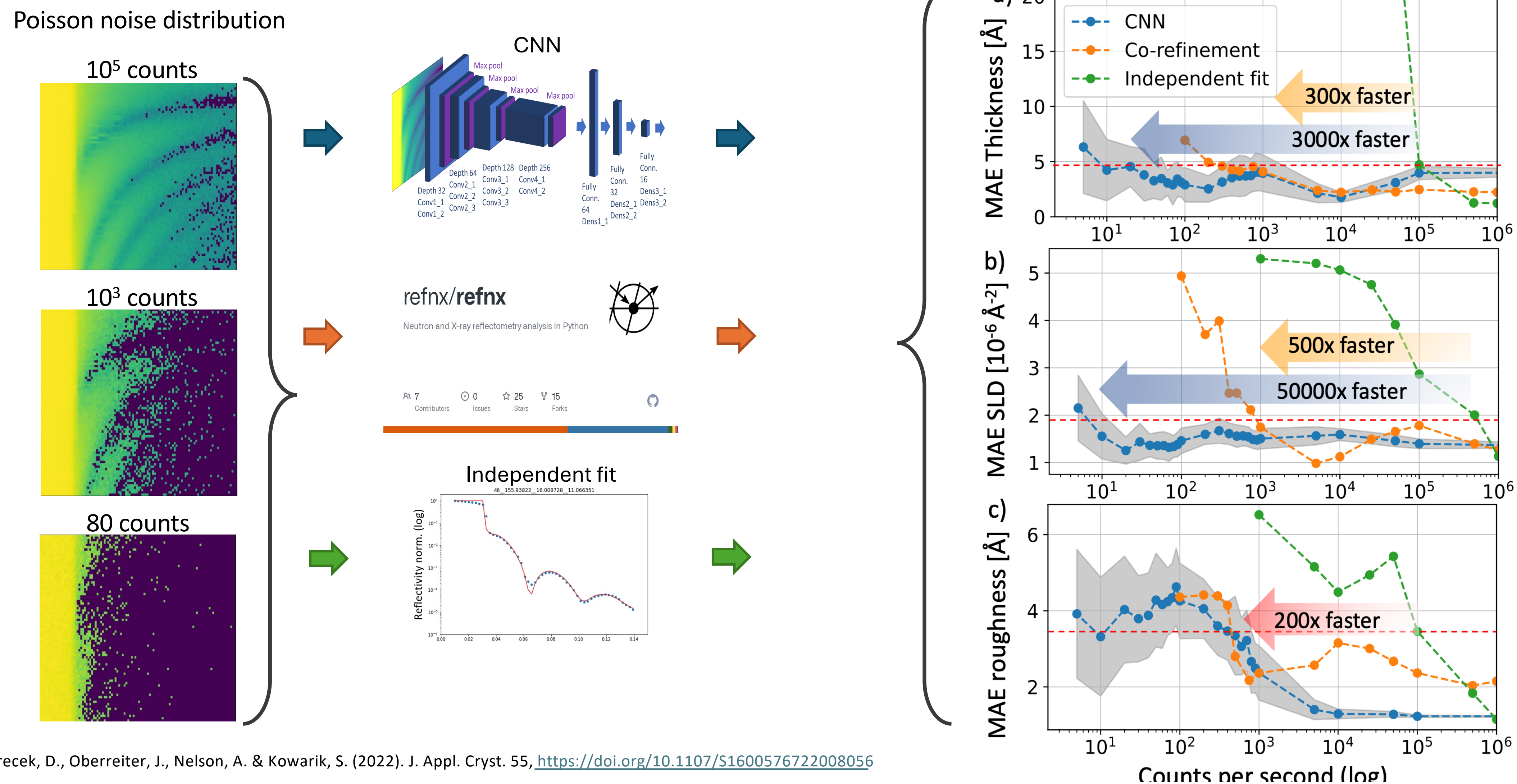
We can measure less: sparse sampling

Random sparse sampling
by *Keras* drop-out function



Marecek, D., Oberreiter, J., Nelson, A. & Kowarik, S. (2022). J. Appl. Cryst. 55, https://doi.org/10.1107/S1600576722008056

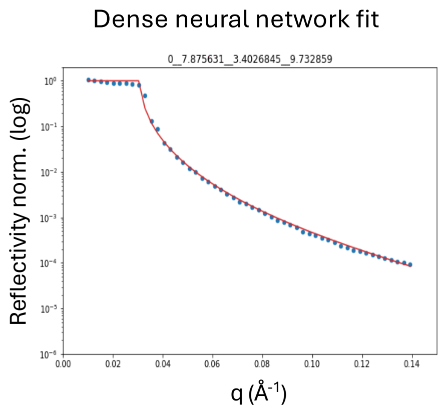
We can measure at lower X-ray flux



Marecek, D., Oberreiter, J., Nelson, A. & Kowarik, S. (2022). J. Appl. Cryst. 55, https://doi.org/10.1107/S1600576722008056

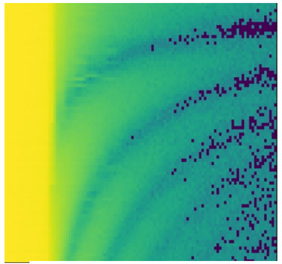
Can we regularise less strictly for higher accuracy?

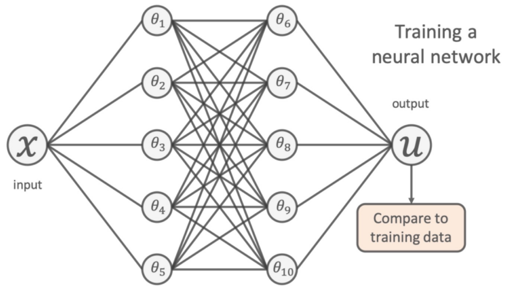
Physics informed neural networks



No regularisation

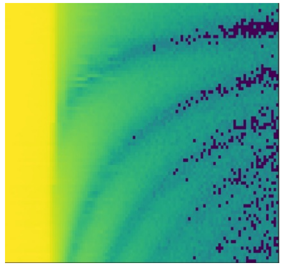
$$d(t_1), d(t_2), d(t_3), d(t_4), \ldots$$

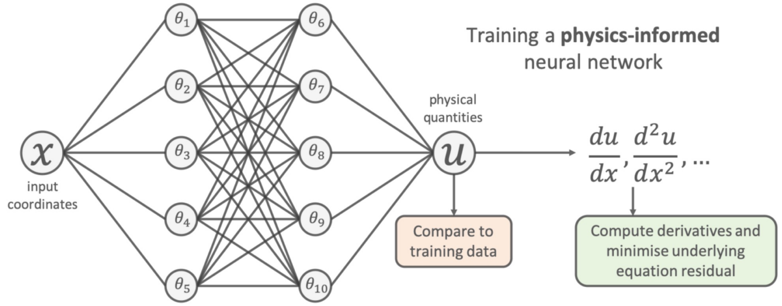




Strict regularisation

$$d(t) = \color{red}{growth\,rate} * t$$

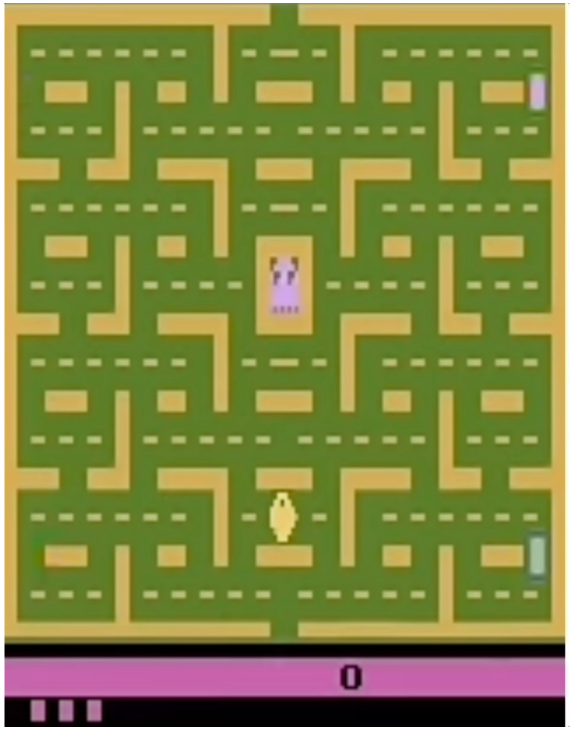


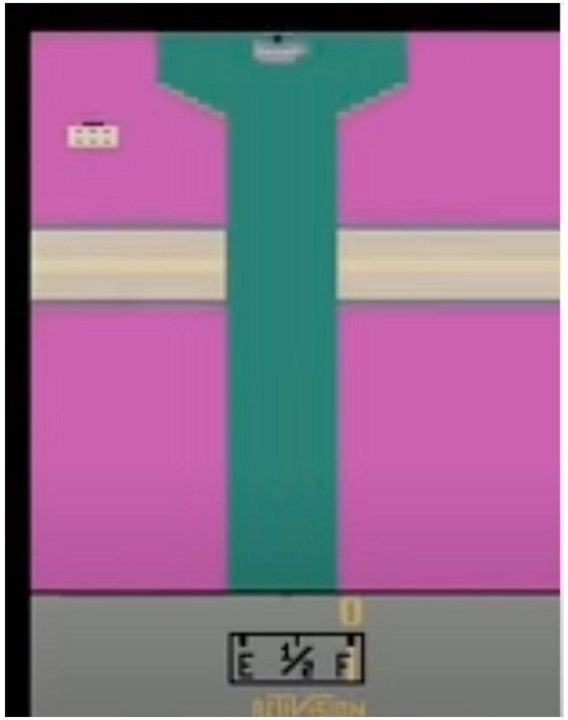


Approximate regularisation
add term to MSE loss function:

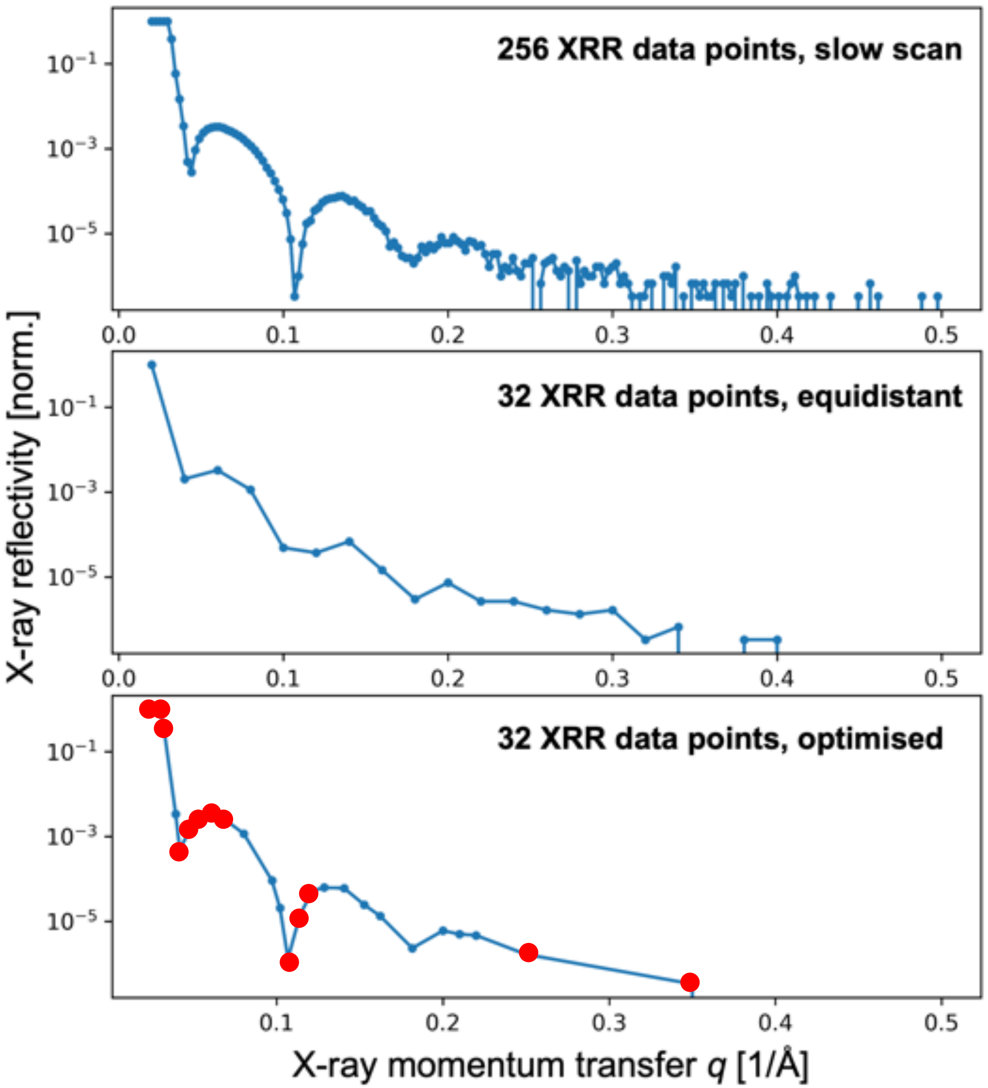
$$\frac{\partial \color{red}{d(t)}}{\partial t} - growth\,rate = 0$$

2014 Playing Atari with Deep Reinforcement Learning – Google DeepMind





Playing Atari with Deep Reinforcement Learning
Volodymyr Mnih, et al. https://arxiv.org/abs/1312.5602.



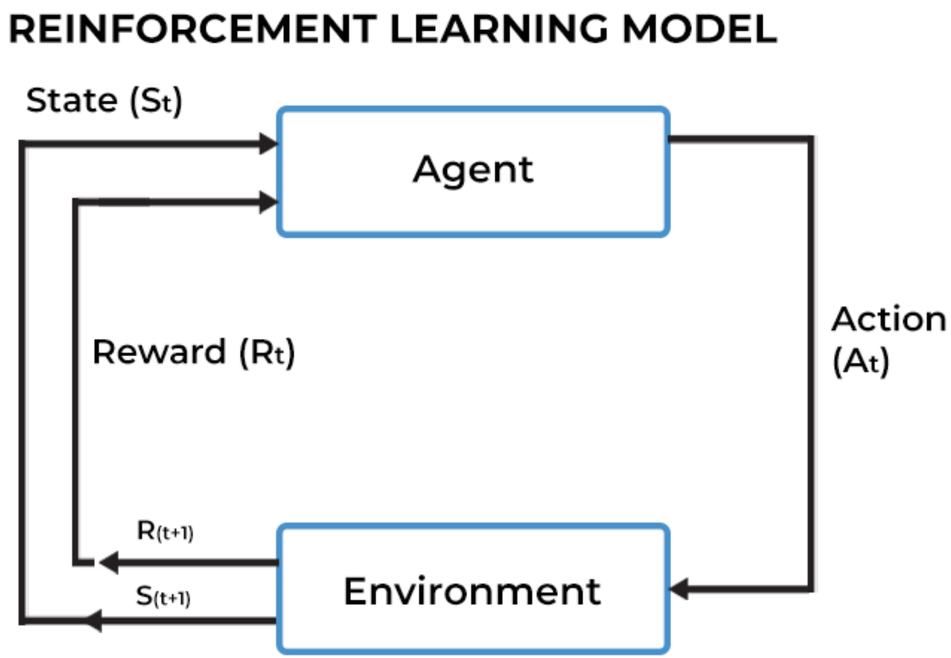
Why Reinforcement Learning?

- Measure fast & get best data possible results.
- Hard to formulate as classical optimization problem

⇒Treat it as unknown environment and let an agent learn to measure at points which lead to good predictions

⇒see also Jos Cooper: HOGBEN code / Fisher information

Reinforcement Learning for XRR Measurements

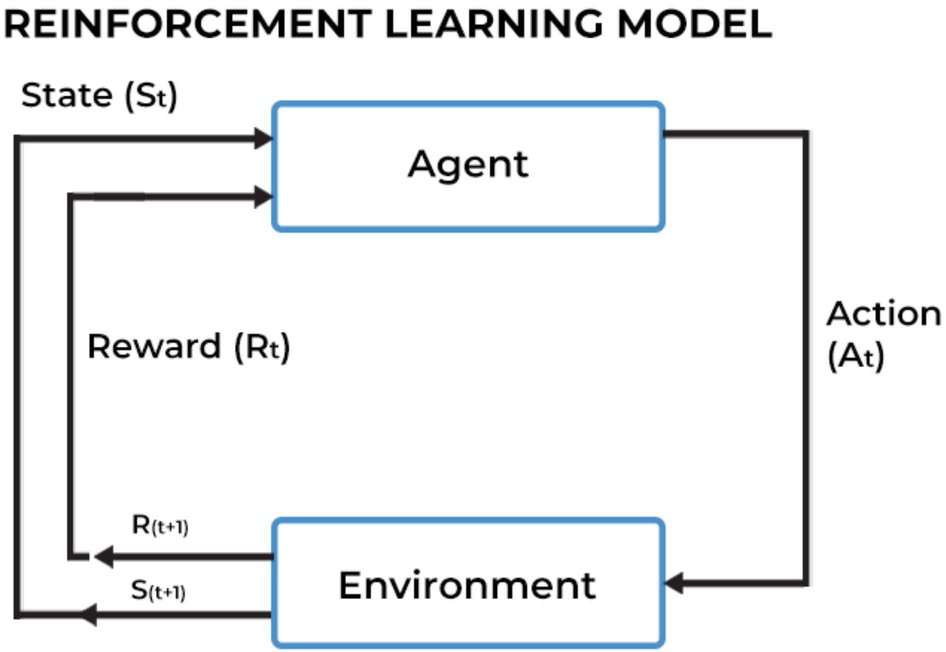


First, we discretize the whole problem:

- Action: diffractometer measures for one second at q= …
- State: An Array with
  - values for each q position
  - the measured counts per second
  - the overall measurement time for q

q	0.01	0.02	0.03	…	0.24	0.27	0.30
CPS	2300	0	450	…	6	0	4
time	1	0	2	…	2	0	1

Reinforcement Learning for XRR Measurements

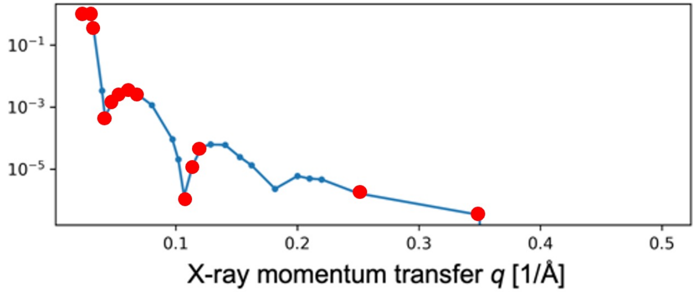


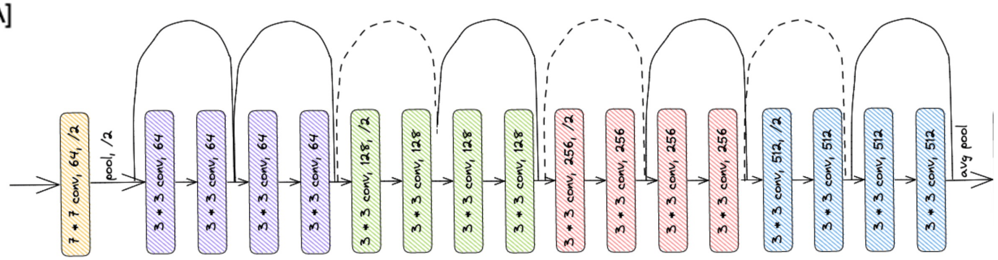
The Reward Function

- Reward measurements which give good predictions
- Lead to stable learning
- Encourage exploration
- Later: punish long moving times

$$R_t = \lambda \cdot \frac{1}{RMSE} \cdot \frac{\#\, pts.meas.}{256} + \begin{cases} 1 & \text{for new } q \text{ point} \\ (-1) & \text{for same } q \text{ point} \end{cases}$$

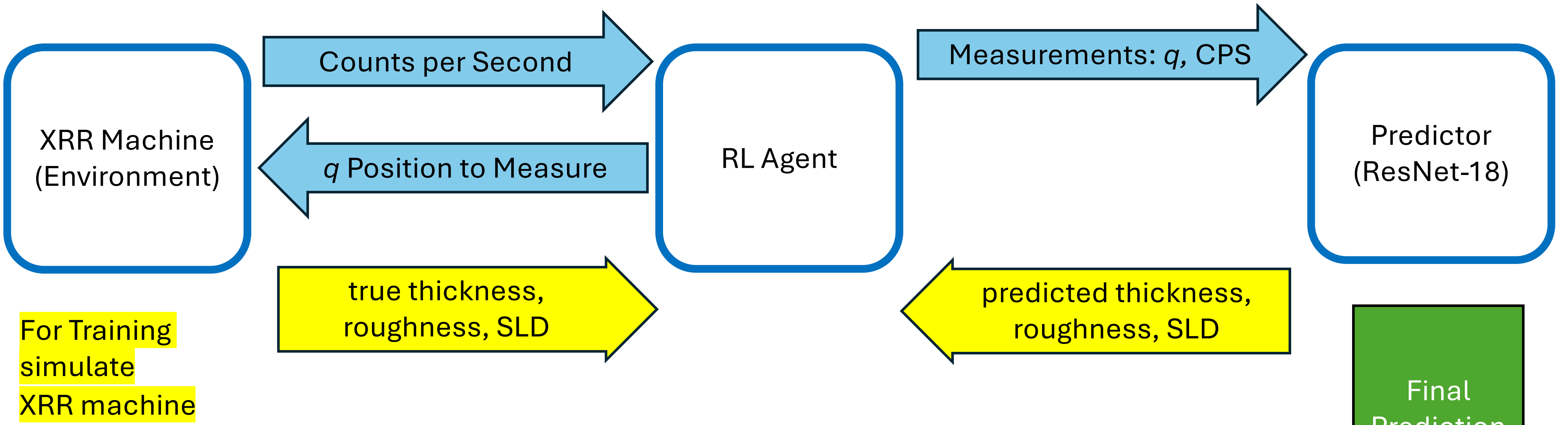
Predictions with partial reflectometry data: ResNet 18 for RMSE





Output:
thickness,
roughness,
density

The Predictor-Agent Interaction & Agent Training



Agent:

- Actor Network: Input State vector, returns probability for each discrete action, then sample from this distribution
- Critic Network: Evaluates the actions taken by actor

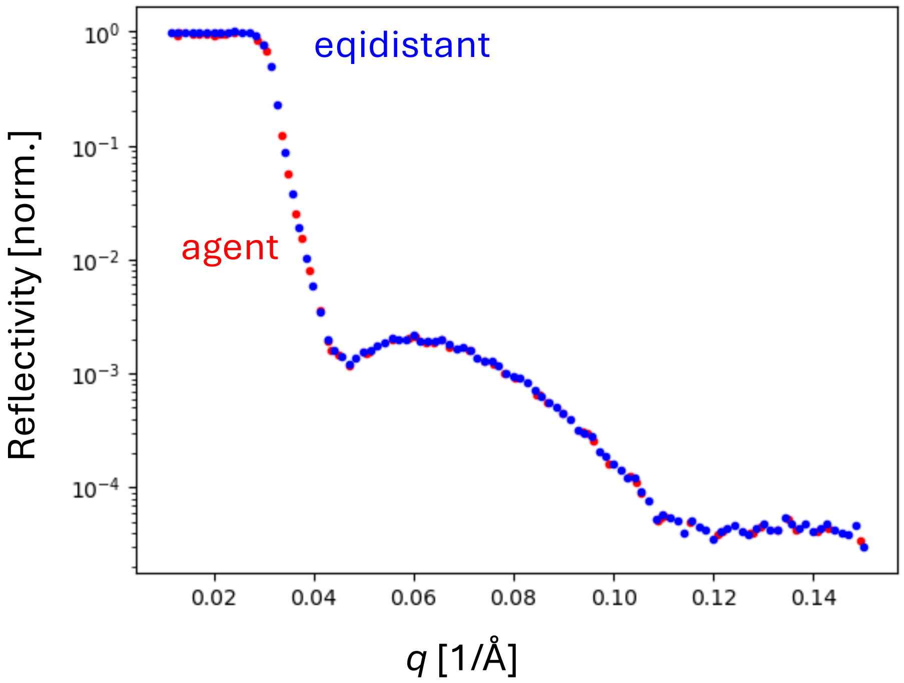
Actor / Critic optimized with gradient based method + clipping
Proximal Policy Optimization (PPO) arXiv:1707.06347 (OpenAI)

Result:
Agent learns to choose *q* points better than an equidistant scan

- Predictions with < 2% error
- Parameter ranges:

Thickness d	50 – 400 Å
SLD	5 – 12 10^{-6} 1/Å^2
Roughness	3 – 12 Å

How does it work for a real PMMA sample?



- Adaptive scanning implemented via BLISS (python)
- Prediction of thickness
  - 256 pts equidistant: d = 80 Å
  - 64 pts equidistant: d = 66 Å
  - 64 pts agent: d=79 Å

Quick XRR

263 / 500 µs

Large data

@ Synchrotrons
but also for neutrons!

Fast refinement (but good!)

ms instead of sec.
better than diff. evolution

Co-refinement

sparse sampling,
low S/N ratio ok

AI controlled scans

example:
4x faster XRR

How can we analyse large XRR datasets

Outline

- quick XRR: into the µs range
  - galvo speed record
  - spin coating example
- AI XRR analysis
  - new state of the art
  - live analysis (ML reflect)
- “big data” in XRR: analysis challenges and co-refinement
  - closed loop
  - adaptive reflectometry – self driving diffractometer

- **“Quick XRR down to 300 µs and fast, accurate AI analysis”**
- Abstract: We show that we can accurately measure XRR curves in (sub)-millisecond timeframes using fast scanning galvo mirrors / rotating samples, and going to the limits of detectors and photon flux at synchrotrons. Leveraging the speed of quick XRR, this method permits the real-time monitoring of rapid thin film deposition processes such as spin coating. The resultant high volume of XRR data, often reaching tens of thousands of curves, benefits from the integration of a rapid machine learning algorithms. These AI / machine learning algorithms not only expedite the analysis but also surpass traditional differential evolution methods with fewer outliers. Additionally, AI-controlled adaptive in XRR measurements can contribute to a fourfold reduction in the number of measurement points required for comparable results. These methods together pave the way for XRR to study ever faster processes in fields such as electrochemistry or thin film growth.

“big data” in XRR: analysing and co-refinement

- databases, reproducibility
  - One goto database for the community, standard data formats?