Reverse tracing in SIMRES & McStas
Towards faster ray-tracing simulations

J. Šaroun, J. Navrátil
Nuclear Physics Institute of the CAS, Řež

Nuclear Physics Institute
Czech Academy of Sciences

Variance reduction in MC ray-tracing

Ray-tracing employs the *rejection sampling* method, which suffers from very low event probability, *p* ... becomes worse with increasing dimension

Relative variance for binomial distribution: $$\frac{\sigma}{p} = \sqrt{\frac{1-p}{Np}}$$

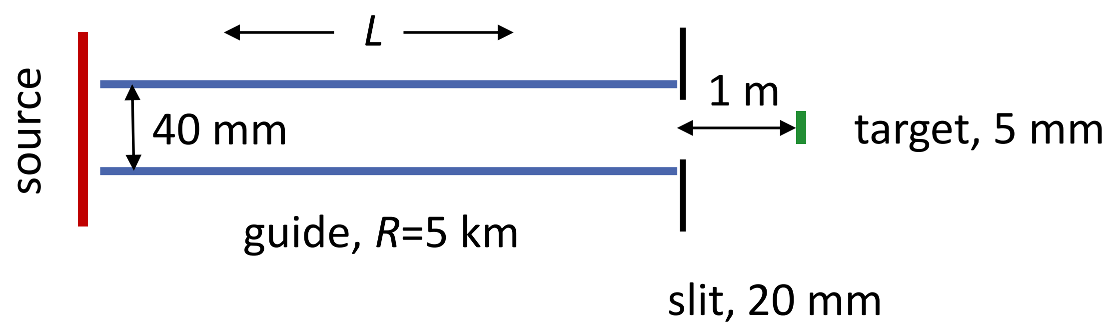
p can be increased by optimization of the sampling volume
(typically n-dimensional box, $n \geq 6$)

What is the optimum sampling volume ?

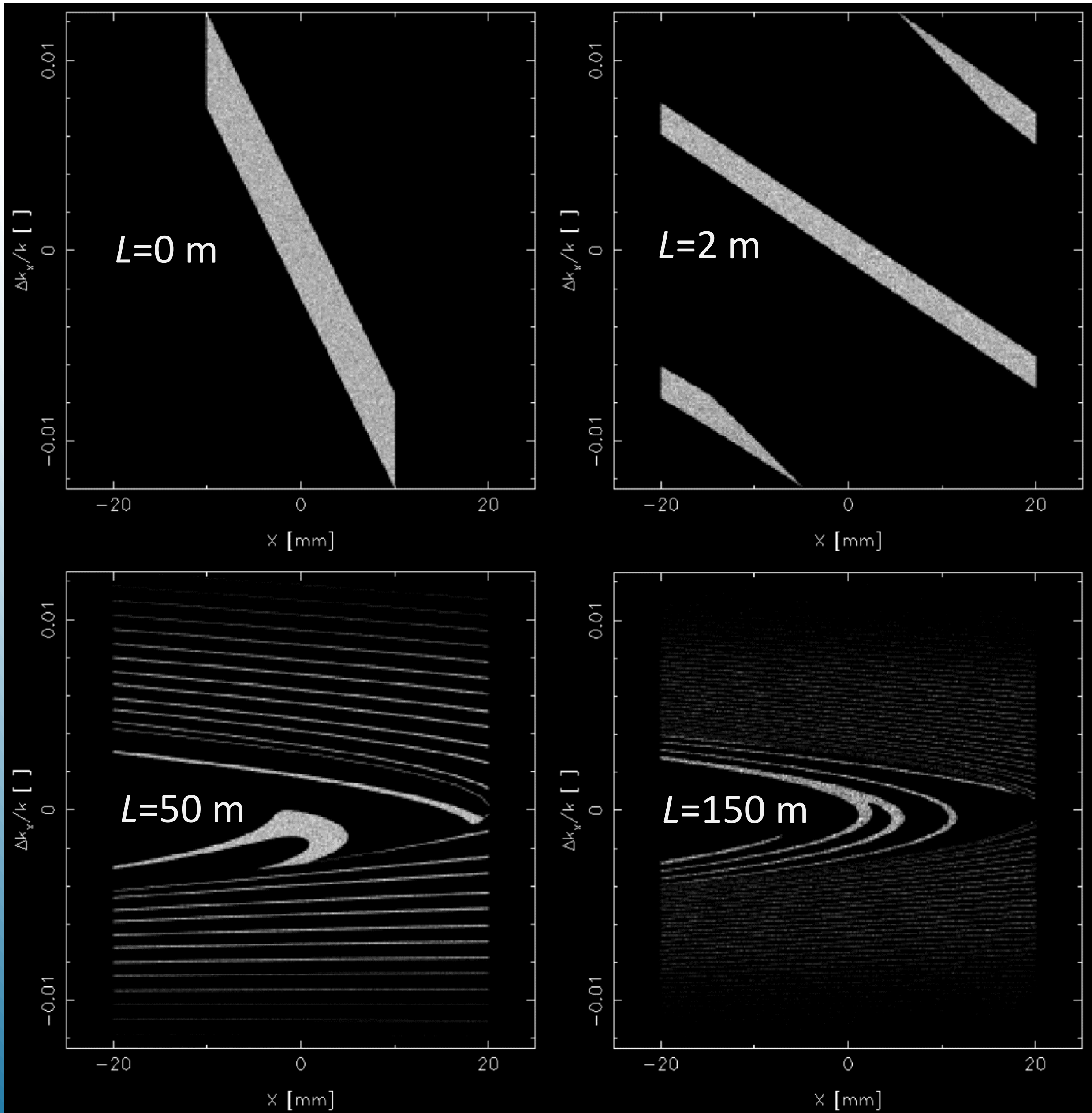
Consider simple 2-dimensional example:

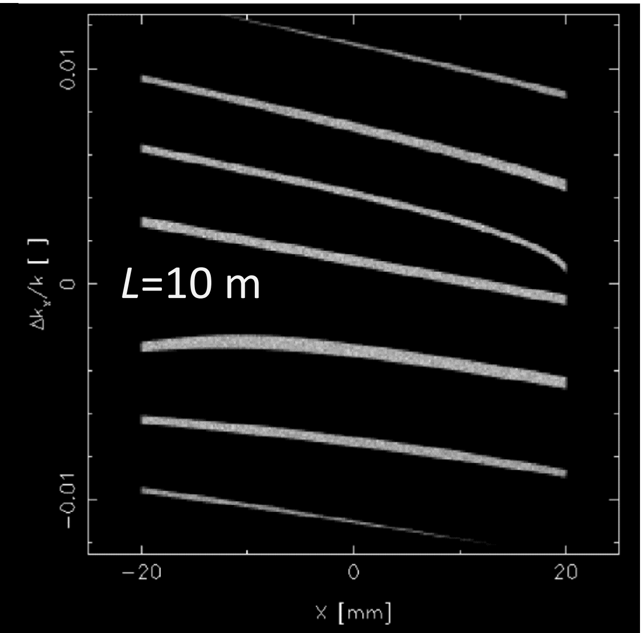
Usual situation: sample << source

How to choose *x*, v_x at the source to hit the sample?



Optimum sampling volume as a function of guide length





may be

- McStas: *x*, *vx*, *t* focusing
- SIMRES: estimation from covariance matrix
- acceptance diagrams ...

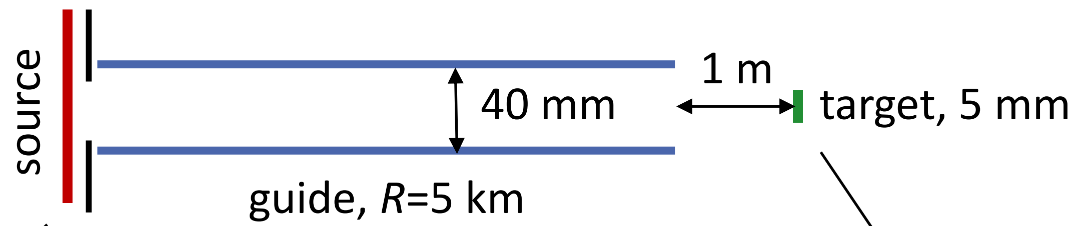
importance sampling?

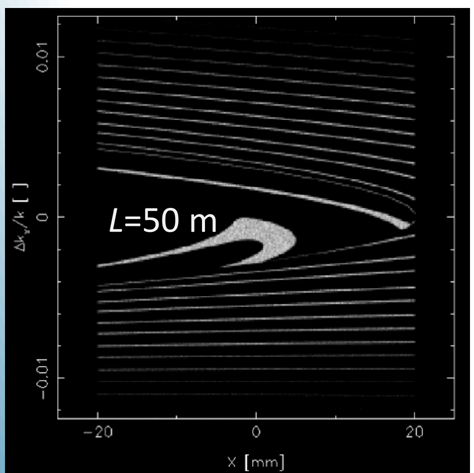
can't be handled

Reversibility of ray-tracing

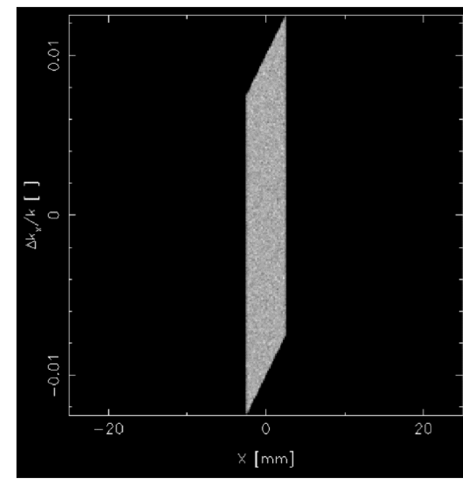
Reversibility is guaranteed by the invariance of underlying physical laws with respect to time inversion:

$$\mathbf{p}' = -\mathbf{p}\ , \qquad t' = -t$$





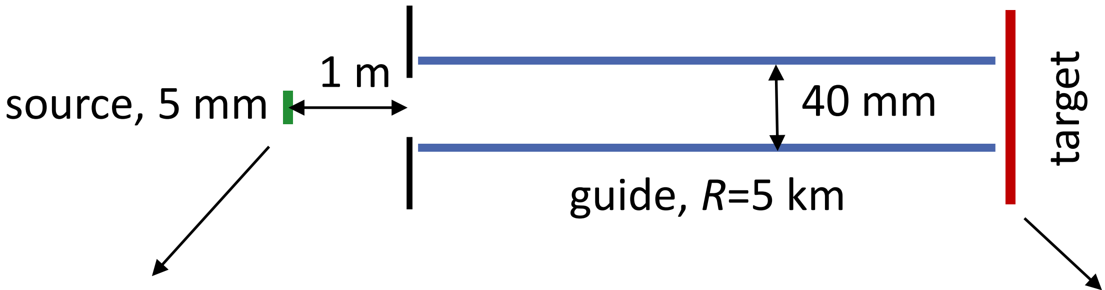
Isomorphism between the two spaces
= there is an inverse map ...

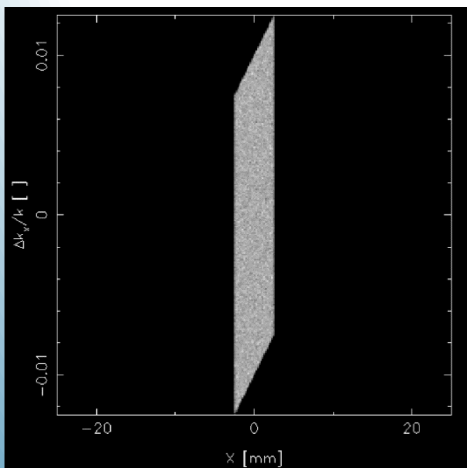


Reversibility of ray-tracing

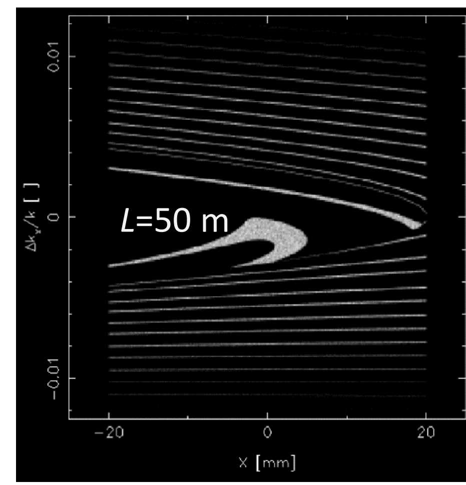
Reversibility is guaranteed by the invariance of underlying physical laws with respect to time inversion:

$$\mathbf{p}' = -\mathbf{p}\ , \qquad t' = -t$$





Isomorphism between the two spaces
= there is an inverse map ...



Solutions

- allow tracing in any direction (SIMRES)
- or build an inverted instrument model (McStas)

Variance reduction by reverse ray-tracing

Prerequisites for true reverse tracing

- SW components must handle tracing in any direction
- × Neutron generator must be decoupled from the moderator.
- × Simulation kernel must handle the **p**, *t* reversal.

Not implemented in McStas ⇒ the workaround solution requires

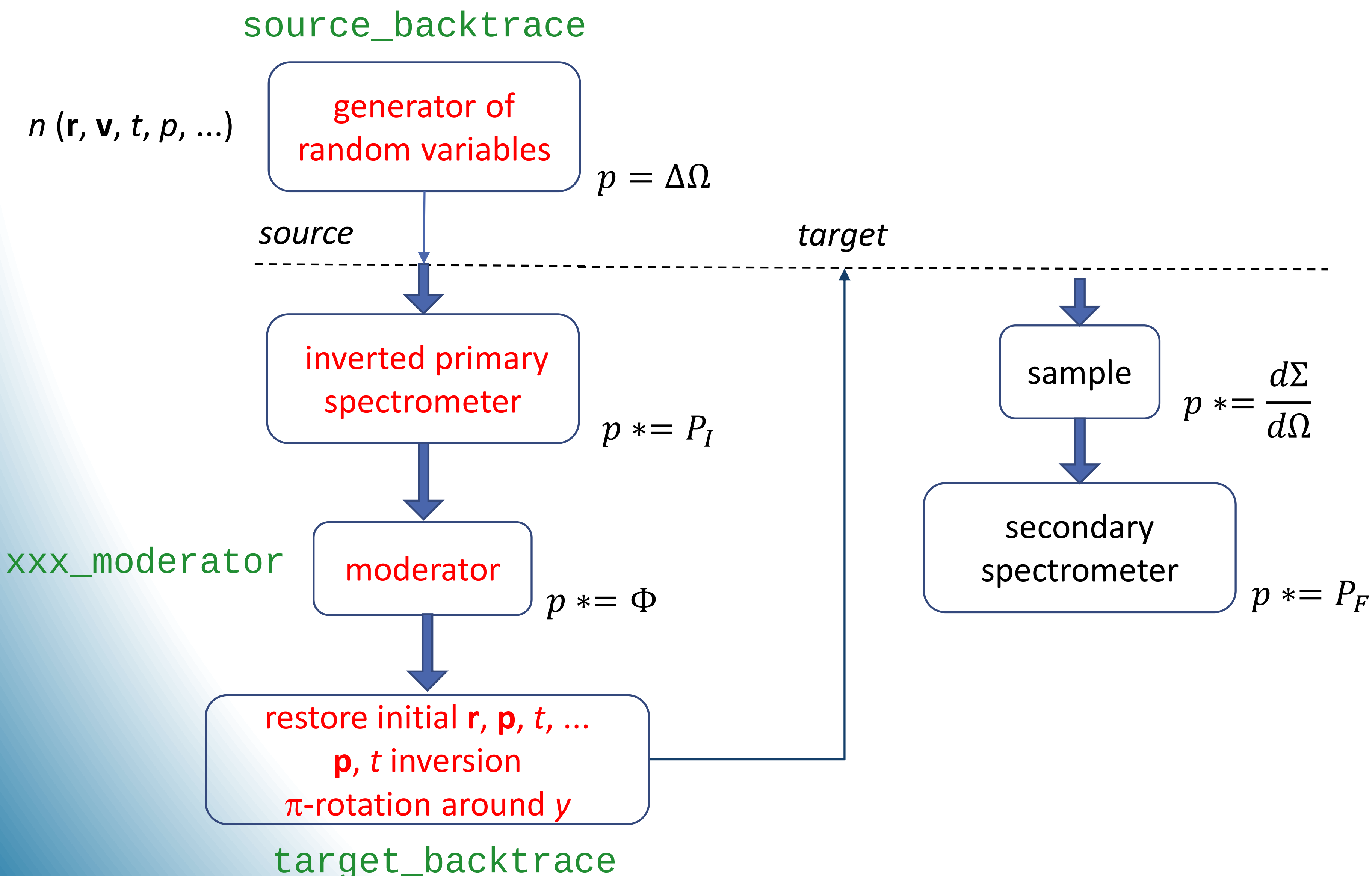
- instrument inversion
- two special components for tracing reversal
- modification of moderators

Work done in early 2016: Simple example for McStas

- Includes required new components and instrument files for testing
- Can be used as a template for real work
- Presented at the last WP8 meeting

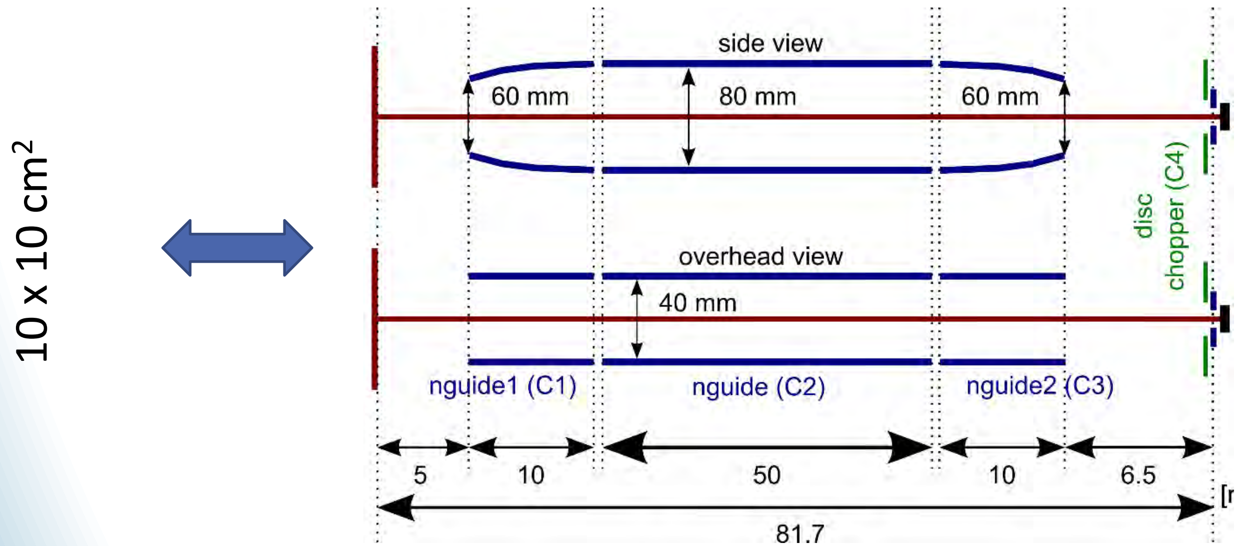
Reverse tracing in McStas

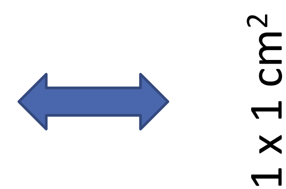
`new McStas components`



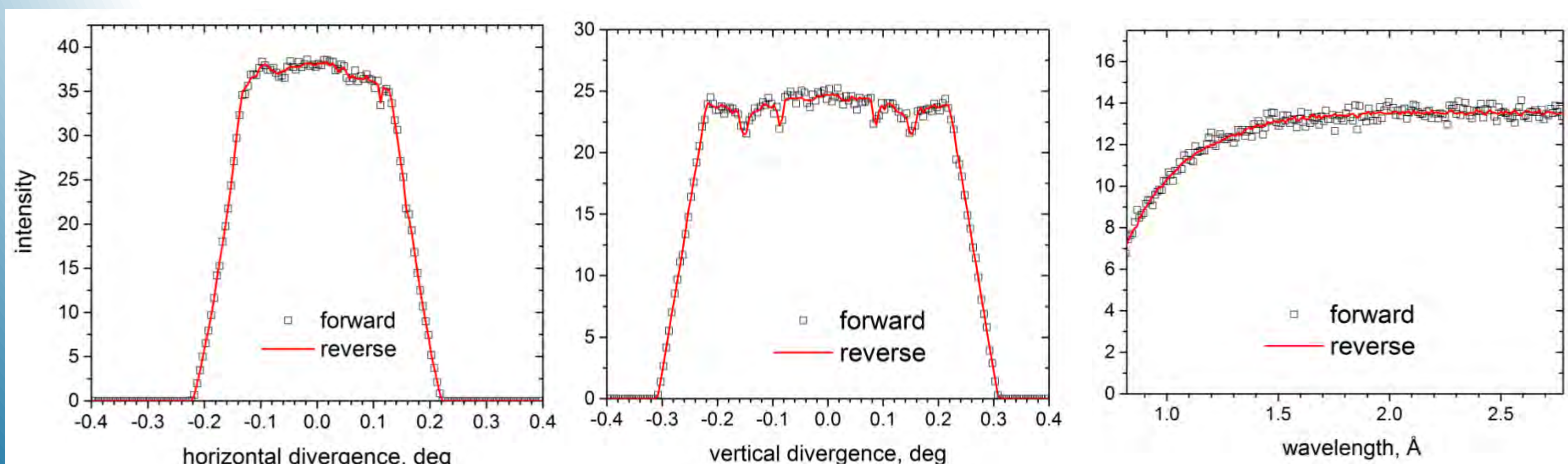
Simple instrument example

- includes commented source code
- short manual to be finished



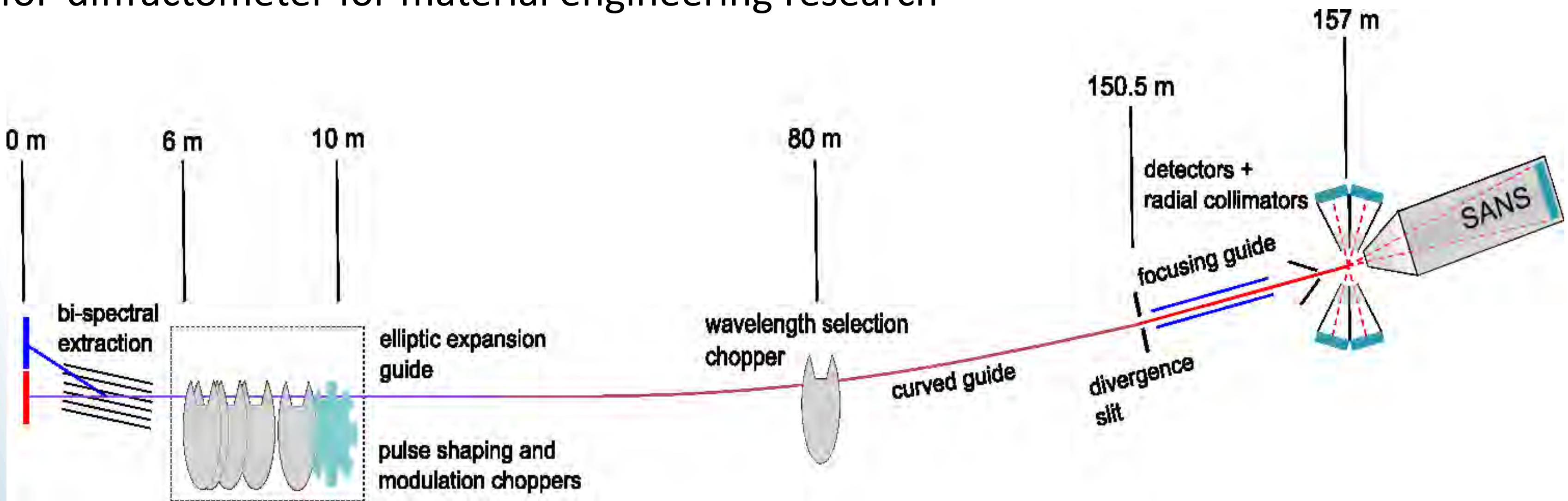


CPU time gain factor: 127



Application to BEER@ESS

ToF diffractometer for material engineering research



McStas model:

- includes 35 components
- primary spectrometer written in two versions: direct and inverted (both are defined by the same set of input parameters)
- 2 adapted components (ESS moderator and diffraction monitor)
- 1 new component (semi-transparent multichannel guide)

Application to BEER@ESS

Added code: **`ESS_Moderator_backtrace.comp`**

plus `ess_source-lib-NPI` (extension to the McStas library)

Provides ESS moderator brightness functions without actually generating neutrons (just adding weights). *Only version 2014 (pancake) is done, implementation of the butterfly model is planned after the McStas update release.*

`PSDcyl_dhkl_monitor.comp`

Adapted PSDcyl_monitor.comp, generates diffractograms from detected neutron positions in 3D (*x*, *y*, *t*).

Runtime parameters for high-level instrument control:

```
DEFINE INSTRUMENT BEER(lam0 = 1.8, dlam = 1, frac = 1, int operating_mode = 6,
Guide_curvature = 0.06,int verbose = 0, S1_width = 0.040, S1_height = 0.0684,
S2_width = 0.040,S2_height = 0.0502, S3_width = 0.005, S3_height = 0.010, S3_dist
= 0.050, int GF3a_on = 1)
```

```
TRACE
%include "input_parameters.instr"
// choose tracing direction:
//%include "BEERforward.instr"
%include "BEERreverse.instr"
%include "secondary_part.instr"
```

Other parameters are provided in a file

TRACE body is just an envelope which allows easy choice of tracing direction

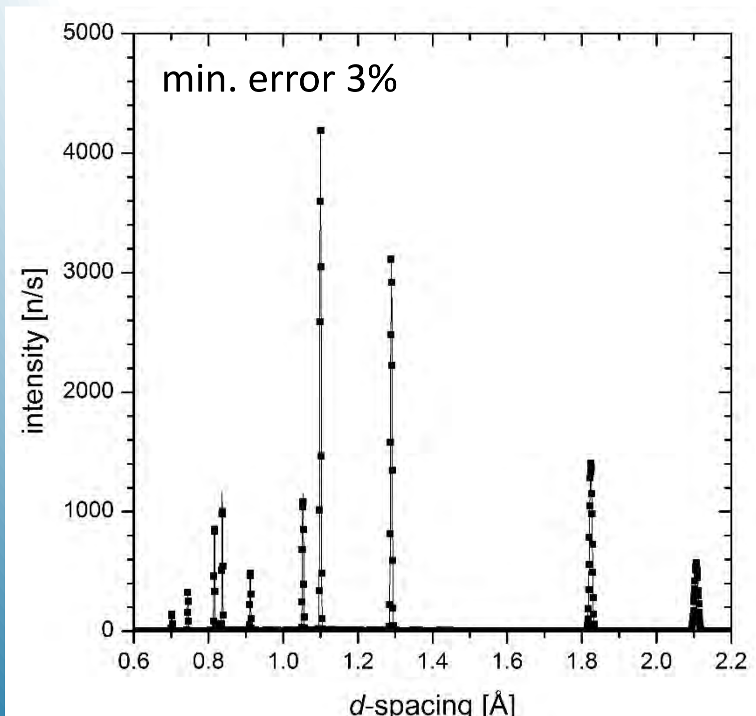
Test with γ-Fe sample (Fe_Gamma.laz)

Conditions:

cylindrical sample:	7 x 20 mm^2
input slit:	5 x 10 mm^2, 50 mm before the sample
output radial collimator:	4 mm resolution (*fwhm*)
resolution mode:	medium, $\Delta d/d$=0.3%
detector coverage:	2θ = 75° – 105°, ϕ = ± 15°

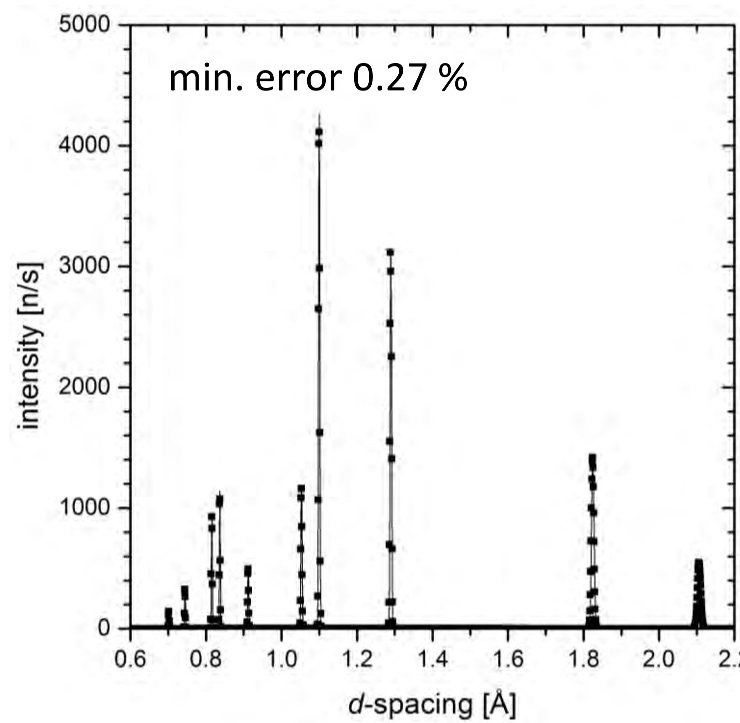
forward

trials: 5 x 10^9 , computing time: 12 h



reverse

trials: 5 x 10^8 , computing time: 4.2 h



Test with γ-Fe sample (Fe_Gamma.laz)

Simulation statistics (counts at the sample)

	forward	reverse	gain factor
Trials / 10^6	5000	500	
Counts / 10^6	0.18	24	
Efficiency [%]	0.0037	4.7	1300
Accuracy [%]	0.34	0.031	
Time [h]	11.95	4.15	
Time to 0.3 % accuracy [h] (*)	9.2	0.045	**200**

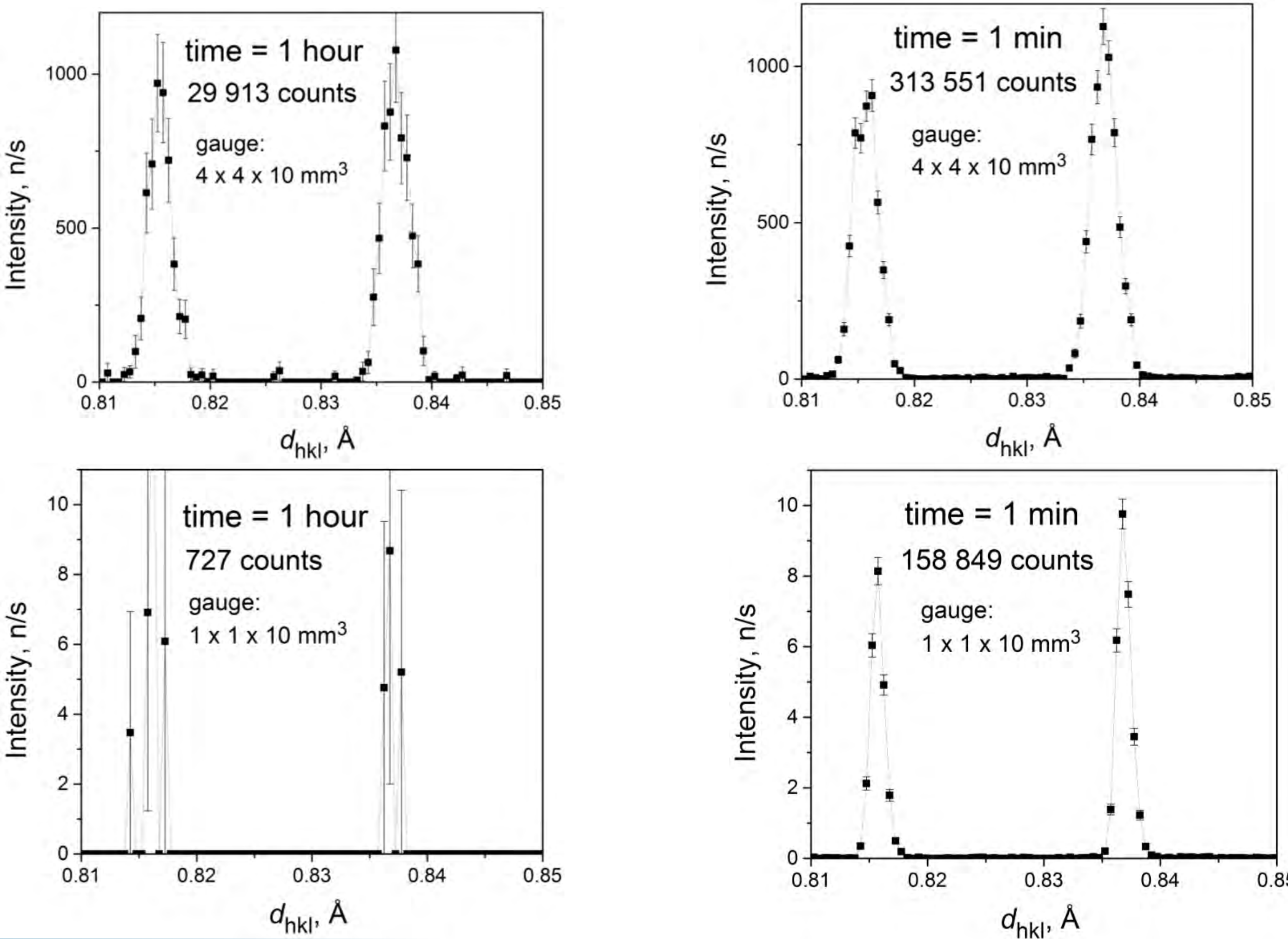
(*) corresponds to about 20 000 counts in the strongest peak, using SPLIT 20

NOTE: The gain would be much higher for high resolution setup with small gauge volume of ~ 10 mm^3
(typical for strain mapping experiments)

Test with γ-Fe sample (Fe_Gamma.laz)

forward tracing

reverse tracing



medium resolution

high resolution

Drawbacks of the present implementation in McStas

- no “user friendly” solution
- requires construction of an inverted primary spectrometer
- all source components need to be modified (this is sometimes quite easy)
- monitors within the primary spec. do not show true flux
  BUT: they *could* show “useful” neutrons = good for instrument optimization

Better solutions on the level of McStas compiler to be discussed ...

Advantages:

✓ gain about 2-3 orders of magnitude in CPU time

✓ speed independent of sample size !

✓ enables virtual experiments with small samples (e. g. strain scanning)
+ optimization of primary spec. based on scattering data

Thank you for your attention