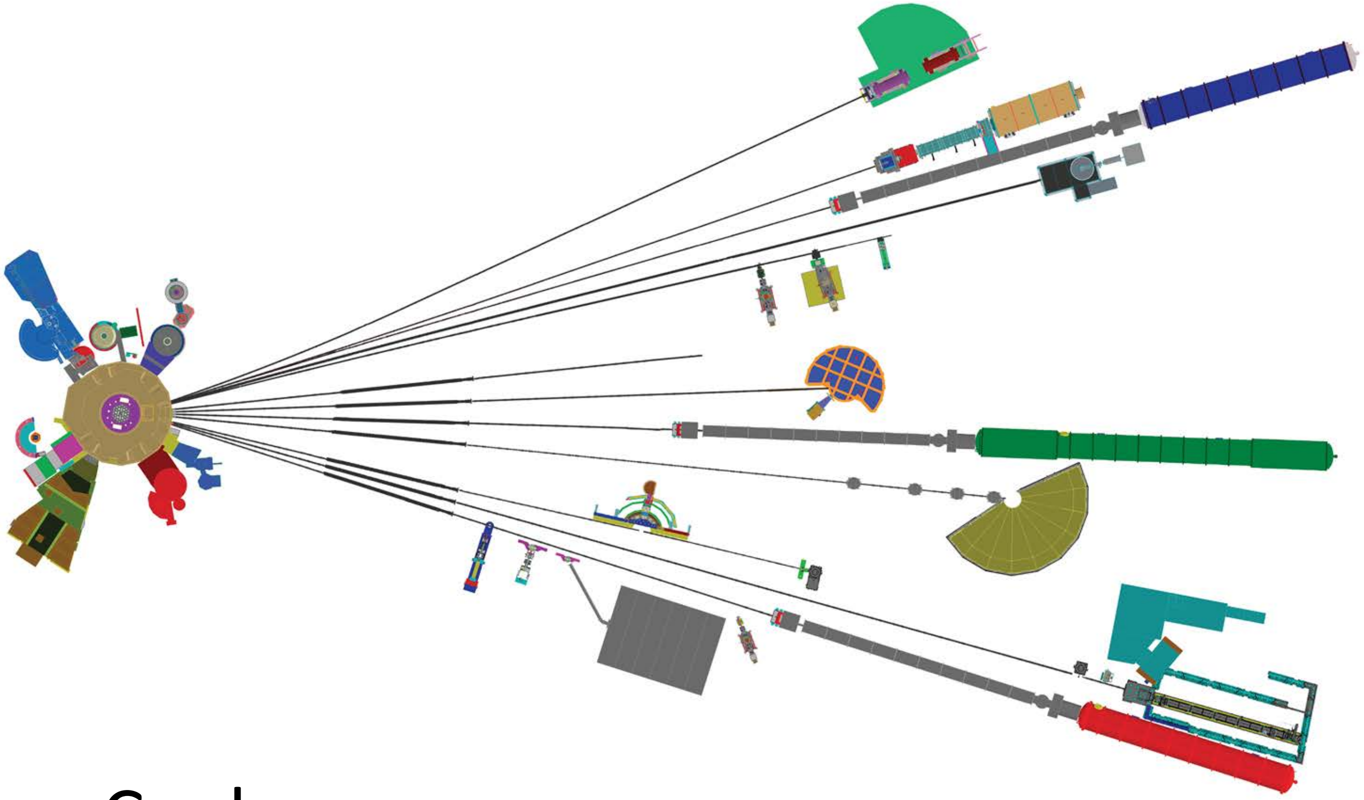
Neutronics simulations



Jeremy Cook
Physicist, Research Facility Operations Group
NIST Center for Neutron Research

Neutronics simulations

Outline

- Why do simulations?
- Available codes for neutron instrumentation
- Acceptance diagrams
- Monte Carlo simulations (emphasis on neutron guides and shielding)
    - Concepts and example for neutron guide
    - Example of neutron guide profile optimization
    - Example of supermirror *m* optimization
    - Simple example for estimating neutron guide shielding
    - Use of neutron guide simulation results for MCNP shielding calculations

NIST Center for Neutron Research

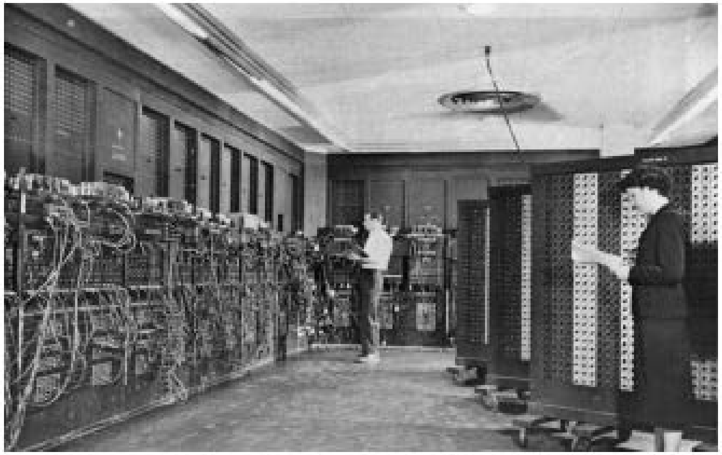
Neutronics simulations

Why do simulations?

- Modern computing technology and simulation codes offer a very cheap and powerful design and optimization tool
- Can generate and store a wide variety of statistical quantities that would be **<u>very difficult or impossible to access in an experiment</u>**
- Use for problems that are difficult to solve analytically or require excessive approximations

Neutronics simulations

Why simulations?



ENIAC– the first electronic computer, University of Pennsylvania. Solved ballistic trajectory problems for Army Ballistics Research Lab. Used electron tubes instead of mechanical counters. Minutes instead of days. Declassified in 1946.

First electronic computer (ENIAC, 1945) 30 tons, 20 ft x 40 ft room, 18,000 vacuum tubes, 100 kHz, 20 word memory

Up to 100k simple addition operations/s, 357 multiplication operations/s or 38 division or square root operations/s

(First computer code: John Von Neumann)

Summit (ORNL) 200 petaflops (200,000 trillion floating point operations per second), Frontier (1.5 exaflop – 1,500,000 trillion floating point operations per second, anticipated 2021)

NIST Center for Neutron Research

Neutronics simulations
Recent developments for neutron scattering

- New and upgraded neutron facilities pushed development of publicly-available, crowd-sourced simulation codes
  - Continual code maintenance/ development and debugging
  - Comprehensive documentation and online tutorials
  - Tested by many users!

- Some private codes developed over many years but not publicly-available (e.g. mine!)

NIST Center for Neutron Research

Neutronics simulations

What is available?

- Some publicly-available, multi-platform neutronics codes for neutron scattering instrumentation
  - **NISP** (Phil Seeger, L. Daeman (LANL) uses MCNP-style geometry input) (http://www.paseeger.com/) – limited support
  - * **McStas** (http://www.mcstas.org/)
    - A general tool for simulating neutron scattering instruments and experiments. Actively supported by DTU Physics (formerly RISØ DTU and RISØ Natnl. Lab), European Spallation Source (ESS), University of Copenhagen, Paul Scherrer Institute (PSI) and Institut Laue-Langevin (ILL)
  - * **Vitess** (https://www.helmholtz-berlin.de/forschung/oe/em/transport-phenomena/neutronmethods/vitess/index_en.html)
    - Virtual Instrumentation Tool for neutron scattering at pulsed and continuous sources (currently part of the German in-kind contribution to the **ESS project** (WP K7))
  - * **IDEAS** (Instrument Design and Experiment Assessment Suite)
  - If you want to learn **DO THE TUTORIALS**

(* see also Neutron News 11/4 (2000) 25-28)

Neutronics simulations

Why simulations?

- Other well-established (and tested) Monte Carlo particle transport codes e.g. **GEANT4** (**GE**ometry**AN**d**T**racking), **MCNP** (Monte Carlo N-Particle)
  - **GEANT4** (CERN) – Developed primarily for high-energy physics
  - **MCNP** (Los Alamos) – Developed originally for nuclear fission criticality and reactor physics
    - MCNP6– unified features of MCNPX and MCNP5 including high energy capabilities and particles of MCNPX)
    - Good for nuclear reactor simulations/design (criticality problems), shielding design, etc.
    - **Neutron coherent scattering not handled by MCNP (cannot be used directly for guide simulations)**

Neutronics simulations

*Acceptance diagrams (for neutron optics design)

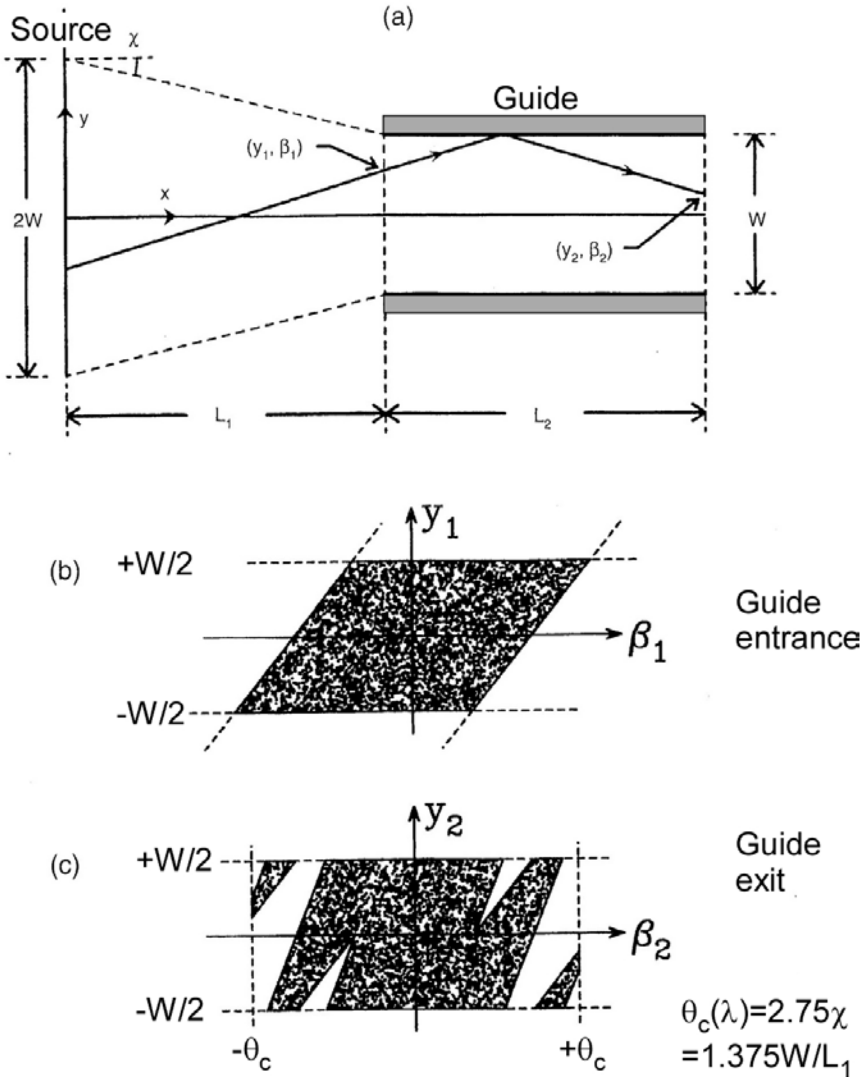
- Danger of “blind” Monte Carlo simulation is possibility of not recognizing erroneous results (e.g. due to erroneous input)
- Acceptance diagrams valuable for understanding “allowed” regions of parameter space (usually **space-angle**) that are potentially transmitted by a guide
- **Restriction**: Acceptance diagram is for a **unique neutron energy**/ wavelength (**also 2-D**)
- Horizontal and vertical 2-D transmissions can be decoupled for rectangular cross-section guides (not the case for e.g. circular cross-sections)
- Examples of acceptance diagrams for *curved guides* in “theory” presentation

NIST Center for Neutron Research

Neutronics simulations

Acceptance diagrams

- An example from literature (J.R.D. Copley,J.Neutron.Res 1/2 (1993) 21-36)

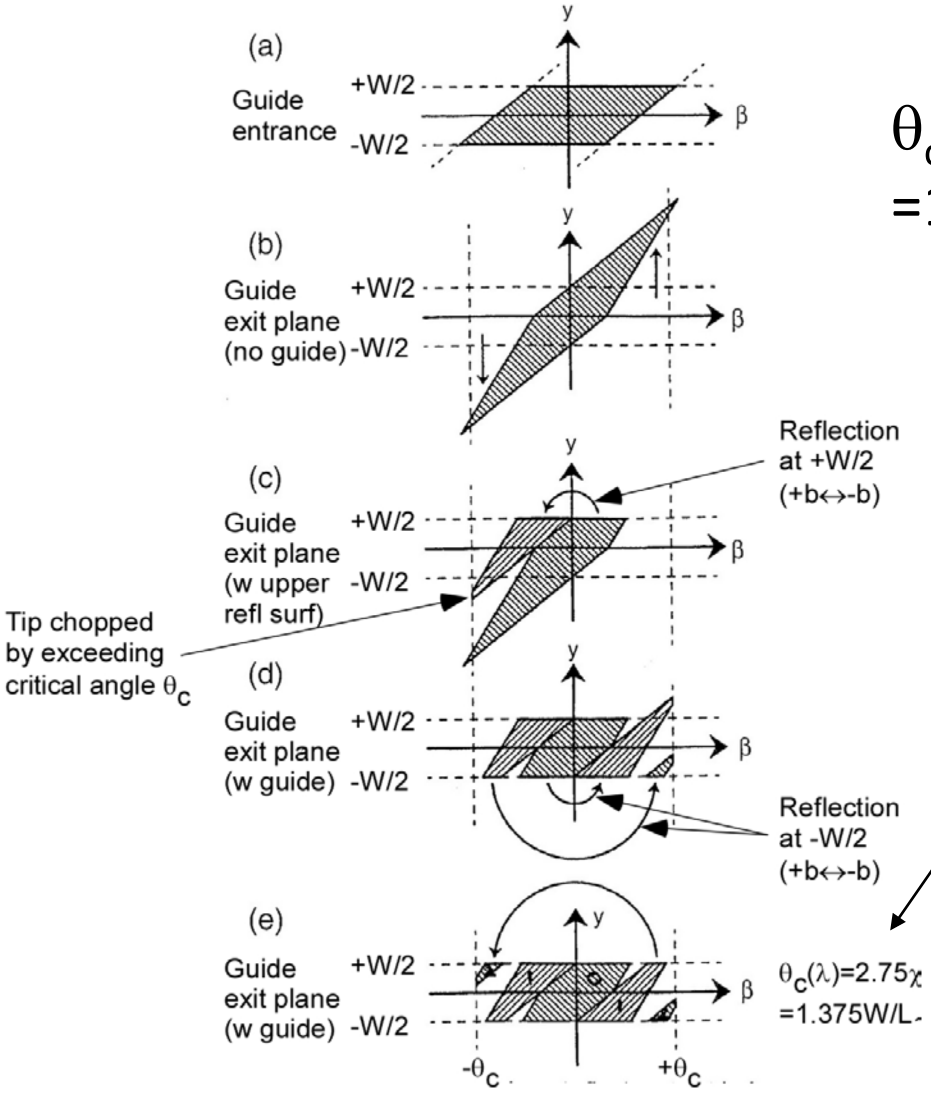


$\theta_c(\lambda)=2.75\chi$
$=1.375W/L$

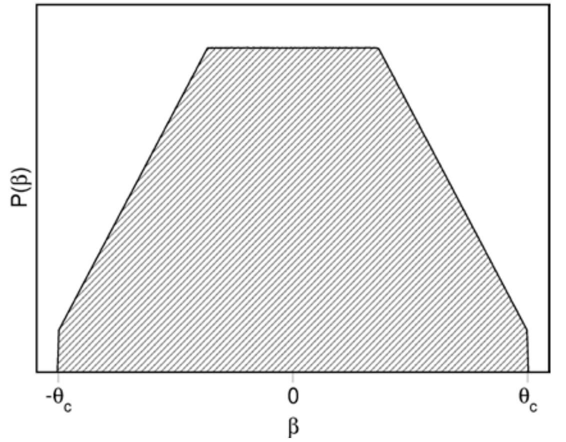
Neutronics simulations

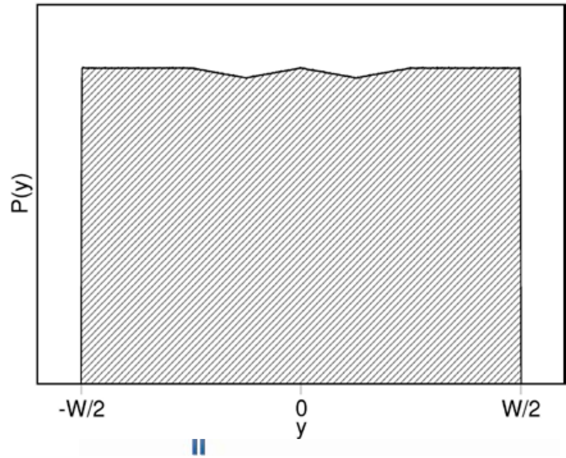
Acceptance diagrams

- An example from literature (J.R.D. Copley,J.Neutron.Res 1/2 (1993) 21-36)



$\theta_c(\lambda)=2.75\chi$
$=1.375W/L$

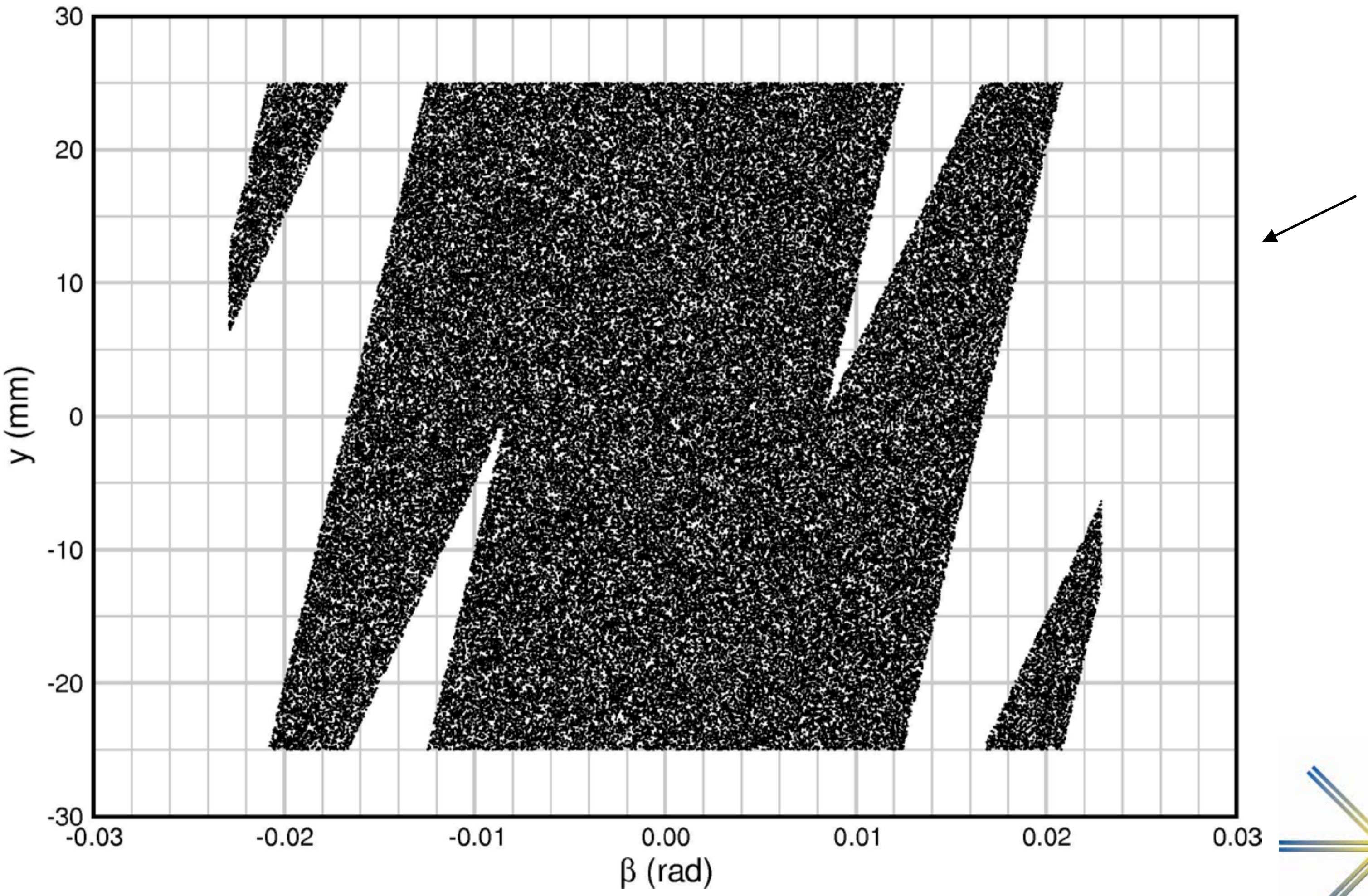




Neutronics simulations

Monte Carlo simulation of same geometry

- An example from literature (J.R.D. Copley,J.Neutron.Res 1/2 (1993) 21-36)



Monte Carlo simulation at guide exit (*beamline2*)

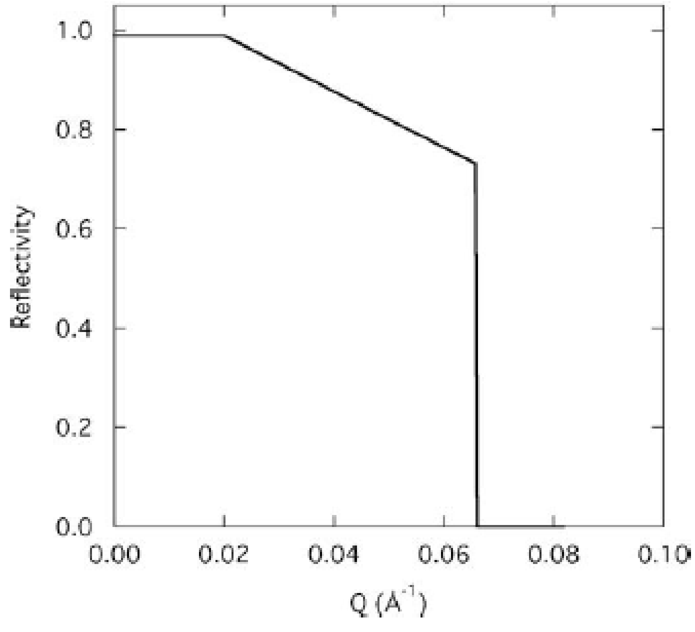
$\theta_c(\lambda)=2.75\chi$
$=1.375W/L$

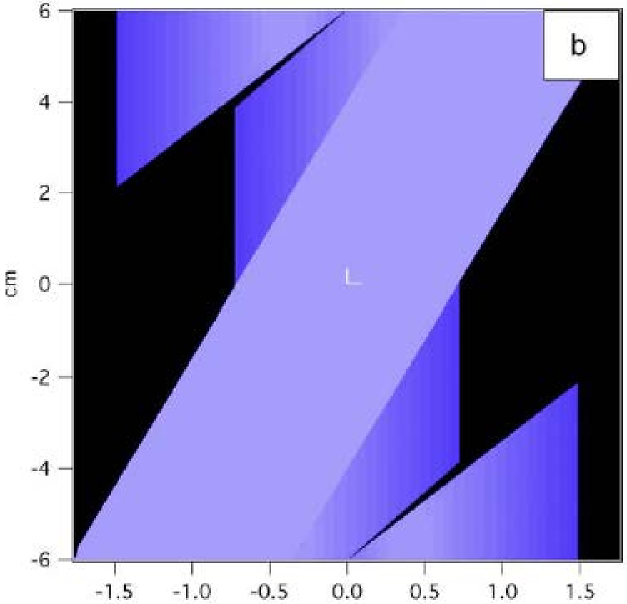
Neutronics simulations

Acceptance diagrams

- Acceptance diagrams often assume **uniform, perfect reflectivity** (R=1) for $\theta \leq \theta_c$ and R=0 for $\theta > \theta_c$
- More sophisticated treatments incorporate **more realistic reflectivity**

(e.g. Bentley and Anderson Nuclear Instruments and Methods in Physics Research A 602 (2009) 564–573)





Neutronics simulations

Monte Carlo simulations

- Acceptance diagrams reveal ***allowed*** spatial-angular regions and give good insight
- BUT… realistic reflectivities can render some of the *allowed* regions almost empty!
- Latter phases of optical design usually performed with **Monte Carlo simulations** using realistic reflectivity models (both *x,y* dimensions and multi-wavelength are combined in one simulation)

Neutronics simulations

Monte Carlo Method

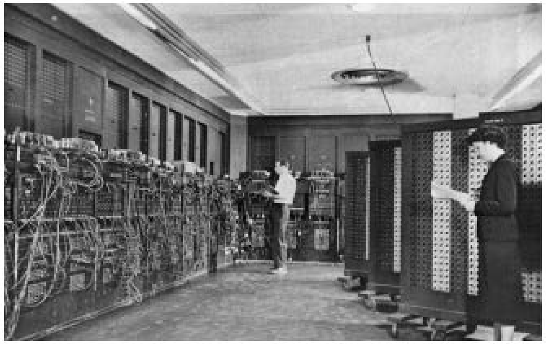
- Monte Carlo Method originated Ulam, Von Neumann, Richtmeyer, Metropolis, Fermi (mid-late 1940‘s)

> “*After spending a lot of time trying to estimate success by combinatorial calculations, I wondered whether a more practical method…might be to lay it out say one hundred times and simply observe and count the number of successful plays*” – S. Ulam

- **Uses random number generation and probabilistic models describing the system to estimate the outcome**

- Name derives from the famous Casino at Monte Carlo (suggested by Metropolis)

Stanislaw Ulam



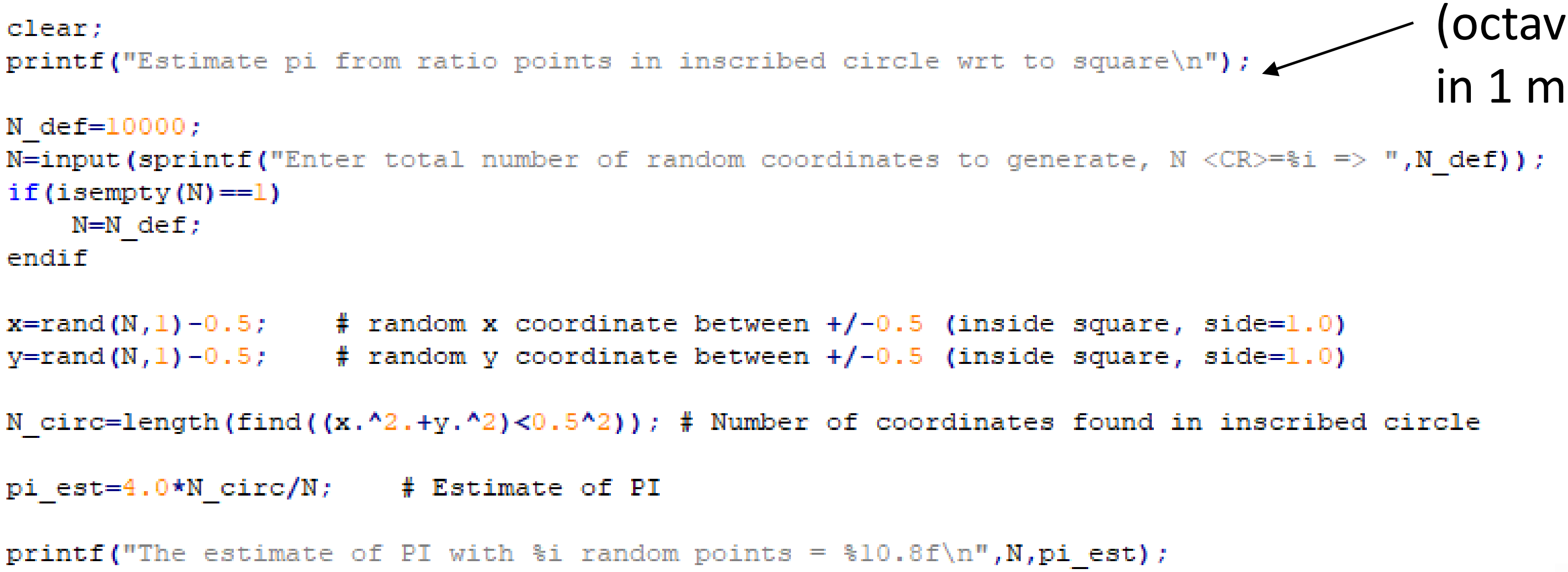
ENIAC– the first electronic computer, University of Pennsylvania. Solved ballistic trajectory problems for Army Ballistics Research Lab. Used electron tubes instead of mechanical counters. Minutes instead of days. Declassified in 1946.

NIST Center for Neutron Research

Neutronics simulations
Monte Carlo Method

- “Trivial” MC example: **Estimate value of π (rejection sampling)**
  1. Generate N random points inside a square
  2. Count number of points that fall inside inscribed circle, N_c (i.e., reject points outside circle)
  3. Estimate of π is $4 \times N/N_c$

Can write code (octave/matlab) in 1 minute



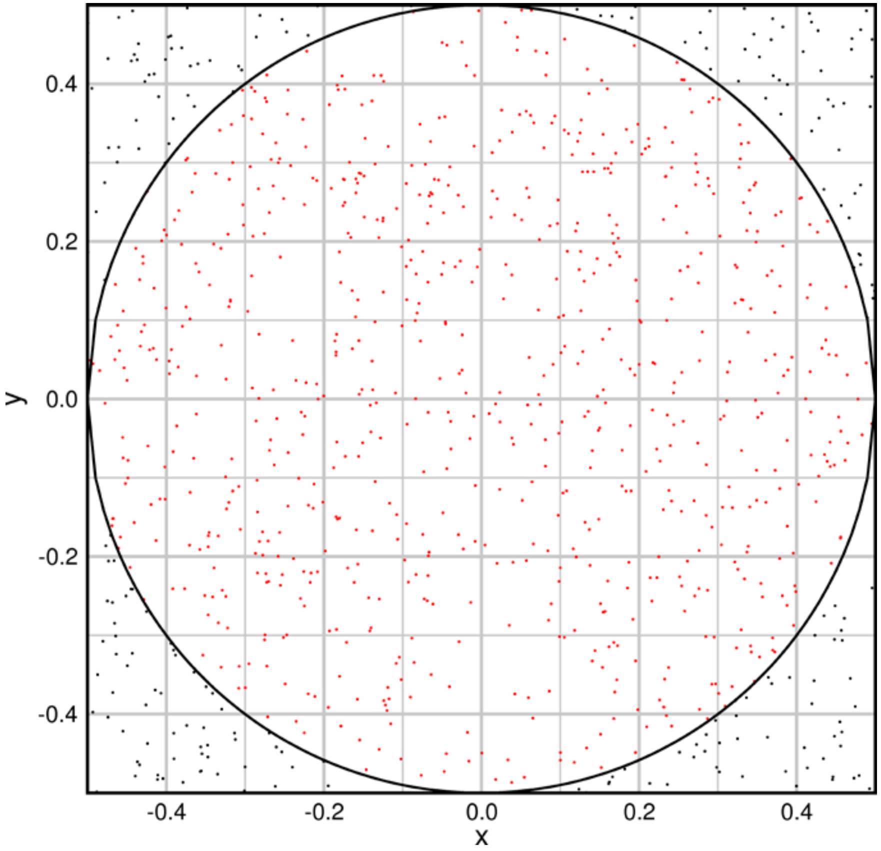
```
clear;
printf("Estimate pi from ratio points in inscribed circle wrt to square\n");

N_def=10000;
N=input(sprintf("Enter total number of random coordinates to generate, N <CR>=%i => ",N_def));
if(isempty(N)==1)
    N=N_def;
endif

x=rand(N,1)-0.5;    # random x coordinate between +/-0.5 (inside square, side=1.0)
y=rand(N,1)-0.5;    # random y coordinate between +/-0.5 (inside square, side=1.0)

N_circ=length(find((x.^2.+y.^2)<0.5^2)); # Number of coordinates found in inscribed circle

pi_est=4.0*N_circ/N;    # Estimate of PI

printf("The estimate of PI with %i random points = %10.8f\n",N,pi_est);
```

Neutronics simulations

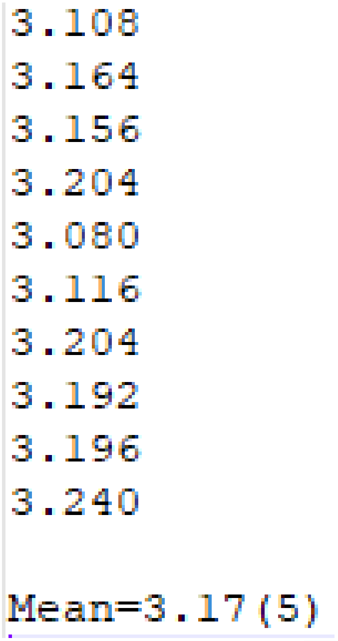
Monte Carlo Method

- “Trivial” MC example: **<u>Estimate</u> value of π (rejection sampling)**

N=1000



10 trials



```
3.108
3.164
3.156
3.204
3.080
3.116
3.204
3.192
3.196
3.240

Mean=3.17(5)
```

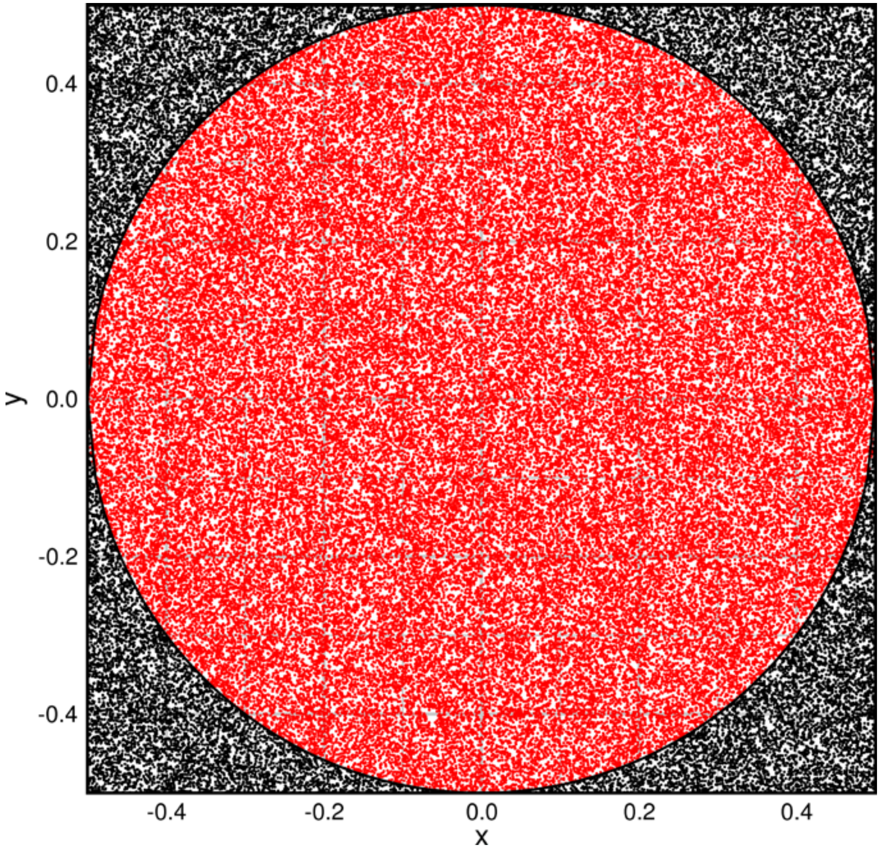
NIST Center for Neutron Research

Neutronics simulations

Monte Carlo Method

- “Trivial” MC example: **Estimate value of π (rejection sampling)**

N=100000



10 trials

```
3.15296
3.14172
3.14224
3.13804
3.14560
3.14692
3.14232
3.14868
3.13932
3.14040

Mean=3.144(5)
```

Neutronics simulations

GOLDEN RULE OF SIMULATIONS!

GARBAGE IN =
GARBAGE OUT!

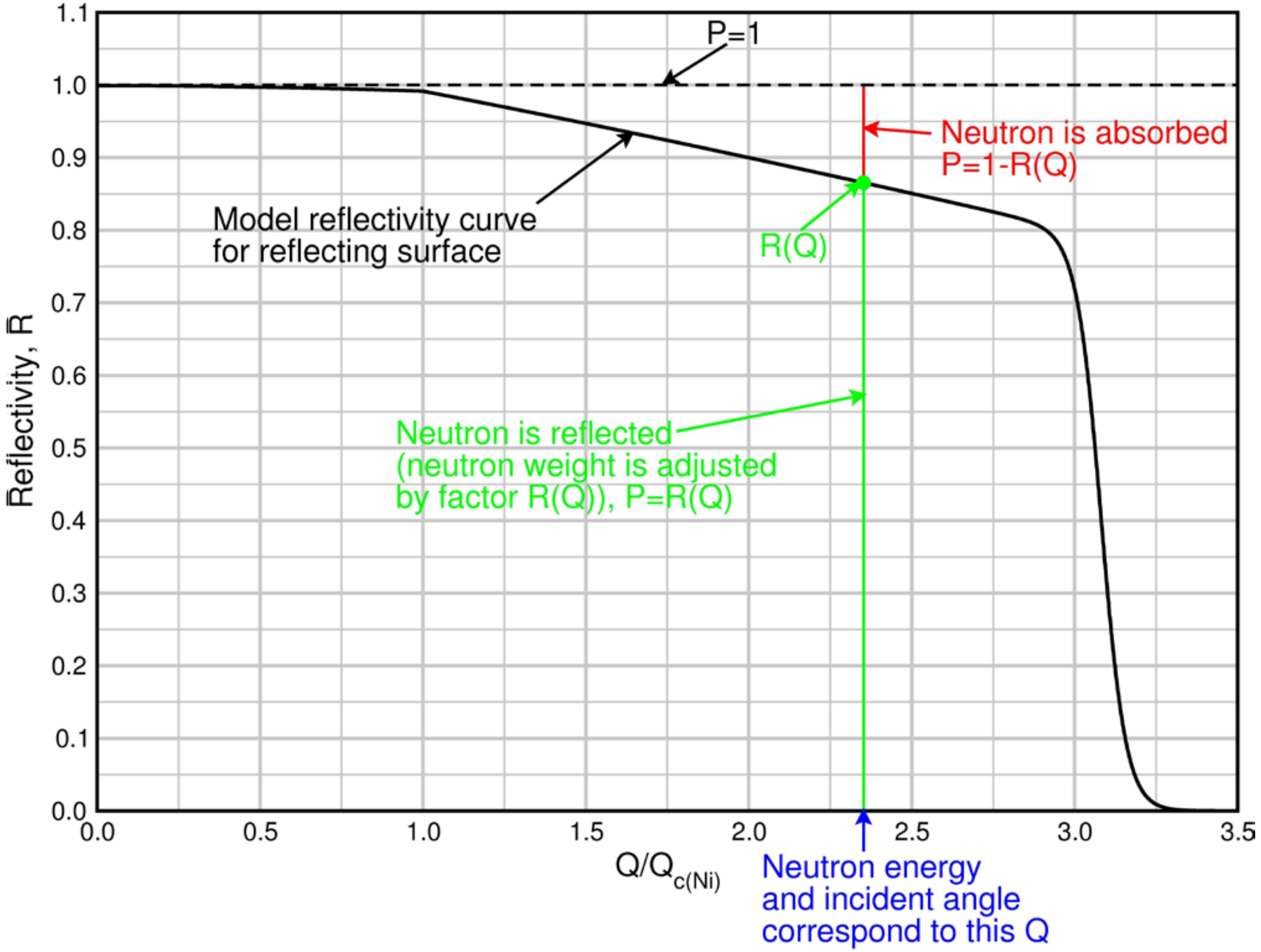
Neutronics simulations

What is needed?

- Most important:
  - Accurate source and geometry specification
    - Usually advisable to specify as accurately as known, want to avoid cumulative round-offs e.g. in geometry specification
  - Define what quantity(ies) is(are) to be tallied and stored (e.g. number of neutrons crossing a defined area per unit time, etc.)
  - If absolute numbers are required: Careful attention to **normalization factors**
    - Example: MCNP default output is normalized “per source particle”. If want e.g. cm^{-2}s^{-1} must provide a normalization factor=number real particles/s divided by tally area (NCNR reactor core number of fission neutrons/s at 20MW= 1.525×10^{18})

Neutronics simulations
Monte Carlo selection example

- An example with **no random number generation**: Determine the probability of reflection in a neutron guide based on the incident neutron energy, incident angle and a model reflectivity curve (initial trajectory may have been created with random number(s))



NIST Center for Neutron Research

Neutronics simulations
Monte Carlo selection example

- **Rejection sampling** (if direct sampling not possible/ feasible): - **for previous reflectivity example**
  - If RAN([0,1]) $\leq$ R(Q) keep neutron with “weight” $w_n=1$
  - If RAN([0,1]) $>$ R(Q) start new neutron trajectory from source immediately (i.e. $w_n \rightarrow 0$)

- Usually more efficient to perform adjust neutron weight:
  - Keep “successfully reflected” neutron with adjusted weight (probability) **$w_n(\text{after})=w_n(\text{before})\times R(Q)$**

  - Can also tally “failed” (unreflected) neutron e.g. for generating a gamma source from lost neutrons for shielding calculations **$w_n(\text{after})=w_n(\text{before})\times(1-R(Q))$**

  - For transmission calculations can impose a “**weight cutoff**” if w_n gets so small that its transmission probability is negligible (stop wasting computer time by continuing to track an almost non-existent neutron) or impose a “Russian roulette” (statistical kill) limit

Neutronics simulations

Coordinate system

- Convenient to arrange coordinate system with e.g. *z*-axis parallel to current guide element axis with *x*-axis (side to side) and *y*-axis (up and down)

- Use coordinate transformations at “kinks” (e.g. between elements of a polygonal curved guide) to establish local *z*-axis along new element axis (trajectory vector = ($k_x\boldsymbol{i}$, $k_y\boldsymbol{j}$, $k_z\boldsymbol{k}$) is **defined within local element coordinate system**)

e.g. element rotation by φ around a vertical axis $\begin{pmatrix} k_x' \\ k_y' \\ k_z' \end{pmatrix} = \begin{pmatrix} \cos\varphi & 0 & \sin\varphi \\ 0 & 1 & 0 \\ -\sin\varphi & 0 & \cos\varphi \end{pmatrix} \begin{pmatrix} k_x \\ k_y \\ k_z \end{pmatrix}$

- For neutron (specular) reflection in general $\mathbf{k}_\mathbf{R} = \mathbf{k}_\mathbf{I} - 2(\mathbf{k}_\mathbf{I}.\hat{\mathbf{N}})\hat{\mathbf{N}}$

  $\mathbf{k}_\mathbf{R}$=reflected *k*-vector, $\mathbf{k}_\mathbf{I}$=incident *k*-vector, $\hat{\mathbf{N}}$ = unit normal to surface

Note: For **parallel-sided** guide element, a reflection involves **only a change of sign of component of k-vector normal to the reflection surface**

Neutronics simulations

Monte Carlo selection example

- Neutron guide simulations
  - Usually want to calculate the intensity of neutrons exiting a guide
  - Calculate “transmissions” (number of neutrons out/number of neutrons in)
  - Obtain absolute intensities when given the **source brightness function, materials cross-sections etc.**
  - May also store “lost” neutron information

- Source brightness = number of neutrons emitted per unit source area per unit time per unit wavelength (or energy) per unit solid angle (in the direction of the guide entrance)

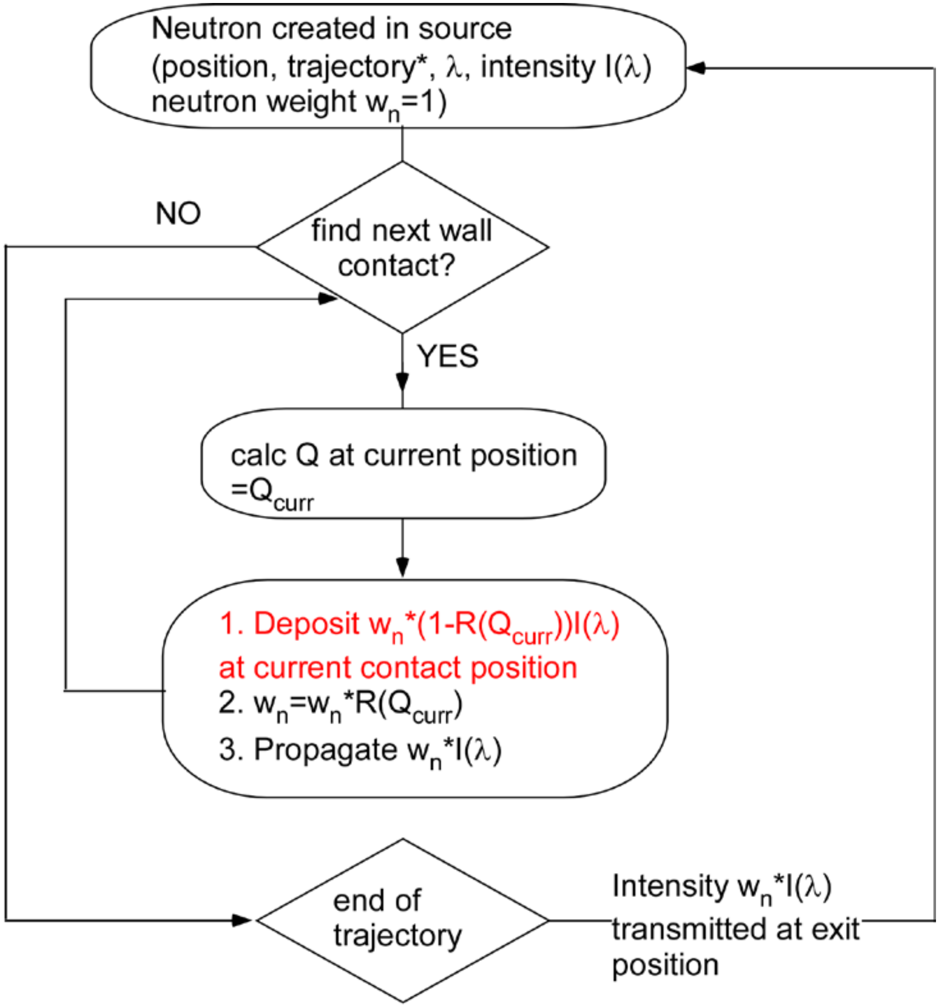
$$\frac{d^4 n_s(\lambda)}{dA_s \, d\lambda \, d\Omega \, dt}$$

- Source brightness may be obtained from careful measurements (we have attempted this a couple of times at NCNR for our cold sources**) or from e.g. MCNP simulations of the source

---

** J.C. Cook, J.G.Barker, J.M.Rowe, R.E.Williams, C.Gagnon, R.M.Lindstrom, R.M. Ibberson, D.A.Neumann, Nuclear Instruments and Methods in Physics Research A792 (2015) 15–27

Neutronics simulations
Monte Carlo ray tracing in a guide



Neutron created in source (position, trajectory*, λ, intensity I(λ) neutron weight w_n=1)

find next wall contact?

NO

YES

calc Q at current position $=Q_{curr}$

1. Deposit $w_n*(1-R(Q_{curr}))I(\lambda)$ at current contact position
2. $w_n=w_n*R(Q_{curr})$
3. Propagate $w_n*I(\lambda)$

end of trajectory

Intensity $w_n*I(\lambda)$ transmitted at exit position

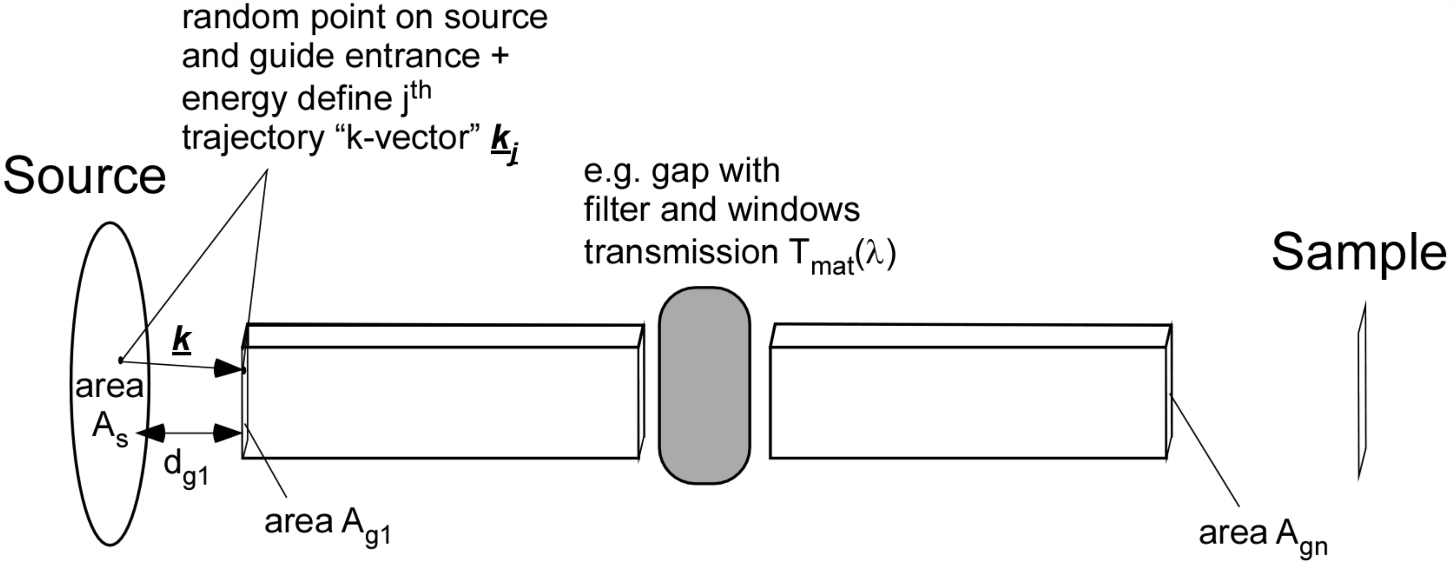
Loss tallying (may be switched off if only interested in transmission)

* May be biased to enter guide for efficiency

NIST Center for Neutron Research

Neutronics simulations

Guide simulation geometry (Simple example: uniform source brightness, discrete energy/wavelength)


random point on source and guide entrance + energy define j^{th} trajectory "k-vector" $\underline{\boldsymbol{k}_j}$

e.g. gap with filter and windows transmission $T_{mat}(\lambda)$

Neutrons/s entering guide per unit wavelength (around λ)

$$\frac{dN_{g1}(\lambda)}{d\lambda dt} = \underbrace{\frac{d^4N_s(\lambda)}{dA_s d\lambda d\Omega dt}}_{B(\lambda)} A_s \Delta\Omega_{ent} \approx B(\lambda) A_s \frac{A_{g1}}{d_{g1}^2}$$

wgt contains cumulative weight of neutron from ray-tracing at exit, $T_{mat,j}(\lambda)$ etc.

Neutron flux exiting guide per unit wavelength

$$\frac{d\varphi_{gn}(\lambda)}{d\lambda} = B(\lambda) \frac{A_s A_{g1}}{d_{g1}^2 A_{gn}} \frac{1}{N_{start}(\lambda)} \sum_{j=1}^{N_{start}(\lambda)} wgt_{gn,j} \cos(\chi_j)$$

where $\cos(\chi_j) \equiv \hat{\mathbf{A}}_{\mathbf{g1}}.\hat{\mathbf{k}}_{\mathbf{j}}$

Neutronics simulations

Guide simulation code philosophy

- **McStas**
  - Very versatile – easy to build in new features and instrument components
  - Uses “easy to understand” metalanguage with GUI for problem definition (source, geometry, required input data, etc.)
  - Converts metalanguage into C code
  - Can run generated C code in parallel MPI (multi-processor) and hyperthreading (OMP)
  - **Continuous neutron energy**/wavelength (all energy-dependent quantities need to be looked-up or calculated for every neutron trajectory in the simulation – time-consuming)
  - Integral fluxes etc. obtained from summations over histogram energy/ wavelength bins
  - Lost neutron tallying

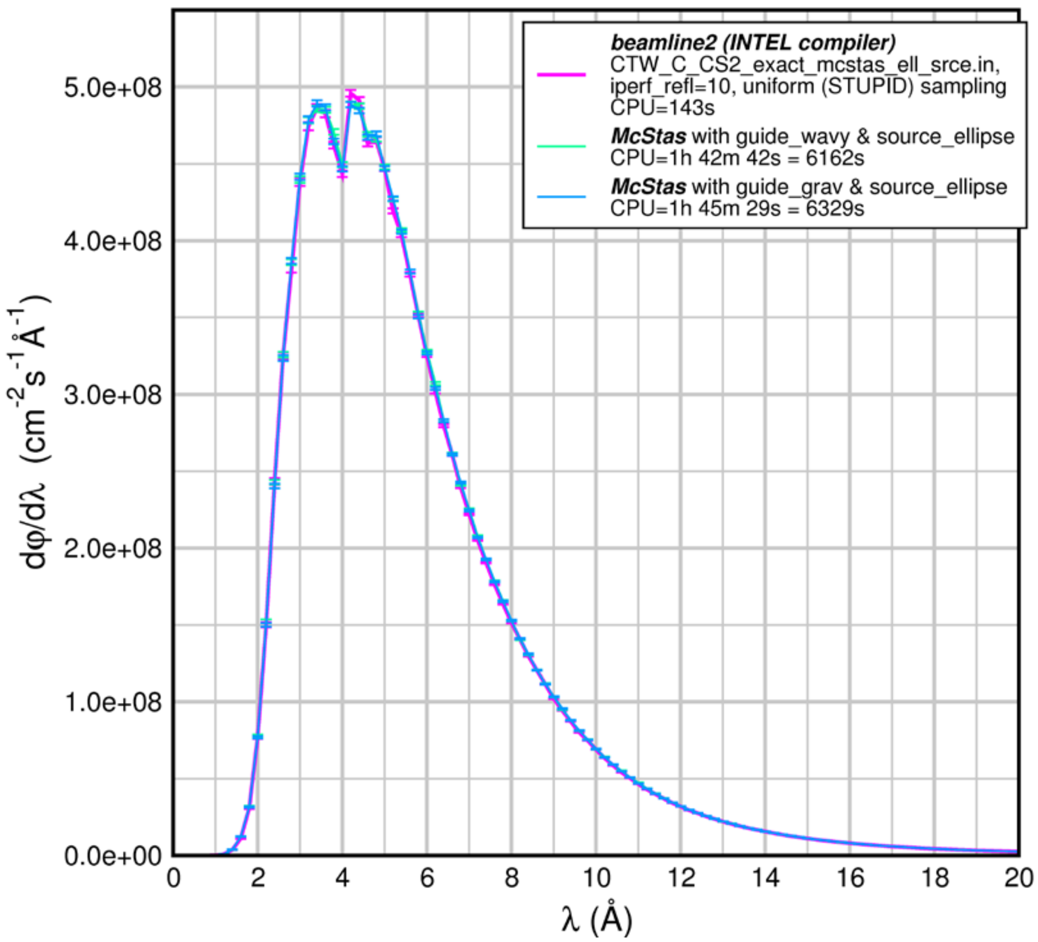
Neutronics simulations

Guide simulation code philosophy

- **My code (*beamline2* – Fortran 90-2003)**
  - More rigid format input (less versatile for additions of new components, options) (considering more versatile input)
  - **Discrete neutron energy**/ wavelength (by design) – **all energy-dependent quantities can be calculated outside of main calculation loops** (very efficient)
  - Integral fluxes etc. obtained by integration (only need to be careful of step size)
  - Discrete energy approach produces statistically-identical results to McStas for guides much faster
  - Lost neutron tallying (turned off when not needed)

Neutronics simulations
Guide simulation code philosophy

- Produce statistically similar results (discrete energy > factor 40 faster/CPU than McStas for same # histories!) – example NGC



Single processor benchmark beamline2 vs McStas (guide similar to NG-C) on Dell Optiplex 990 i7-2600 (64 bit) 3.4GHz 8G

problem set for close equivalence for both programs (ALL SIMS 98000000 total histories each)

Neutronics simulations
Guide simulation: Guide profile optimization

- Example: Re-optimization of focusing guide for HFBS
  - Start with an approximation (original HFBS tapered guide) divided into a number of elements
  - Optimize profile with “blanket” supermirror coating of **high *m*** (e.g. all *m*=4)
  - Choose optimization criterion (e.g. neutrons/s on a defined area), wavelength/energy range, range of guide elements to adjust …
  - Iteratively adjust entrance/exit dimensions of defined elements with constraint e.g. $w_{ex,i}=w_{ent,i+1}$ or $h_{ex,i}=h_{ent,i+1}$
  - Alternate entrance-to-exit, exit-to-entrance each time finding (local) optimum of criterion for each element (stop/ adjust dimension step length if going away from optimum)
  - Repeat trying to converge on *global* optimum

NIST Center for Neutron Research

Neutronics simulations

Guide simulation: Guide profile optimization

- (For *beamline2*) perl script engine:
  1. Runs simulation code
  2. Analyses and plots results
  3. Updates input to “best yet” and adjusts step size according to results
  4. Repeats until convergence criterion

- McStas has similar utility *guide_bot* (MATLAB script, Mads Bertelsen)

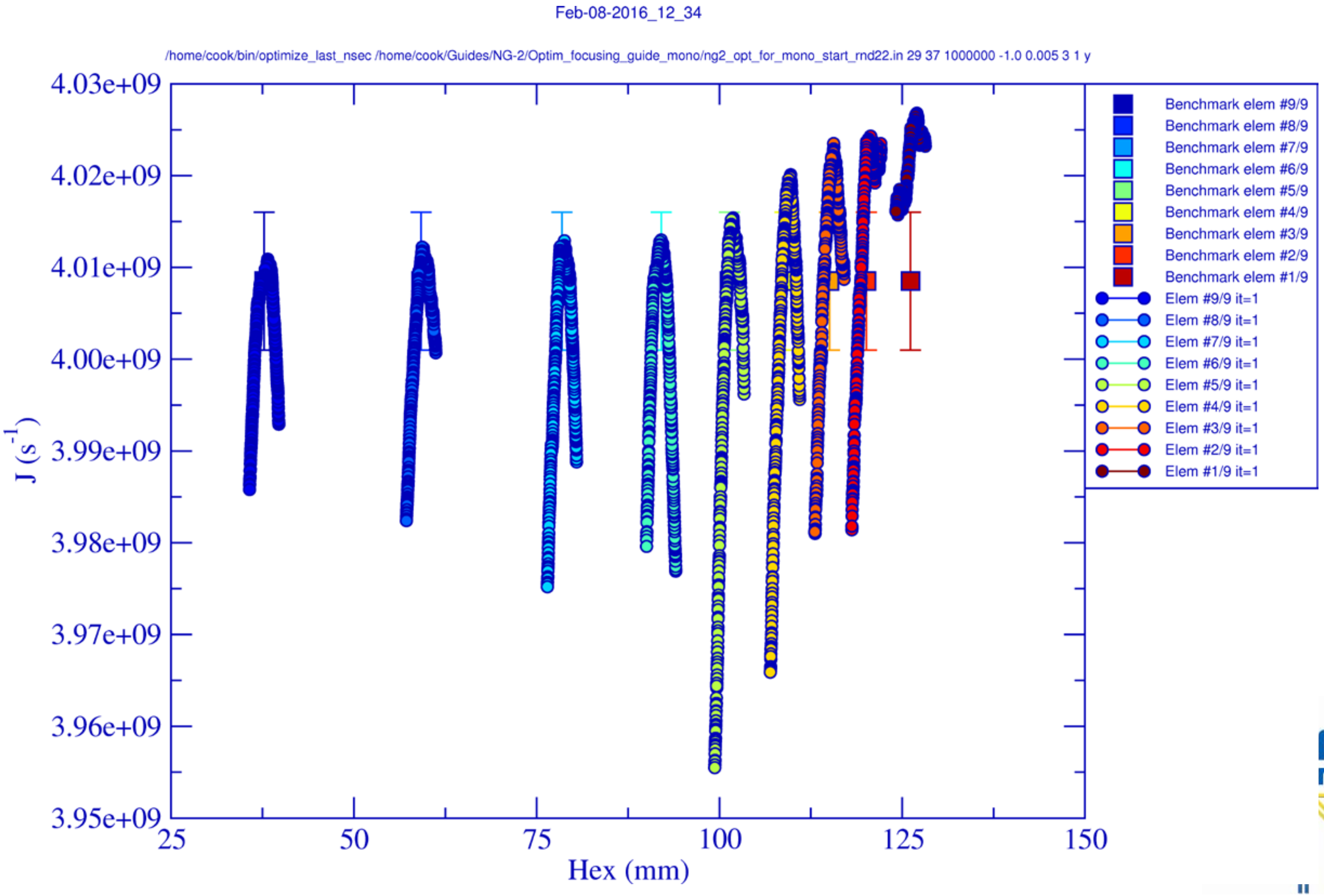
- *guide_bot* modified by Leland Harriger (NCNR) to optimize bi-elliptical replacement for NG-5 whilst accounting for monochromator performance

NIST Center for Neutron Research

Neutronics simulations

Guide simulation: Guide profile optimization

- Example: re-optimization of focusing guide for HFBS (beamline2)



Feb-08-2016_12_34

/home/cook/bin/optimize_last_nsec /home/cook/Guides/NG-2/Optim_focusing_guide_mono/ng2_opt_for_mono_start_rnd22.in 29 37 1000000 -1.0 0.005 3 1 y

One iteration for 9 elements entrance/exit heights

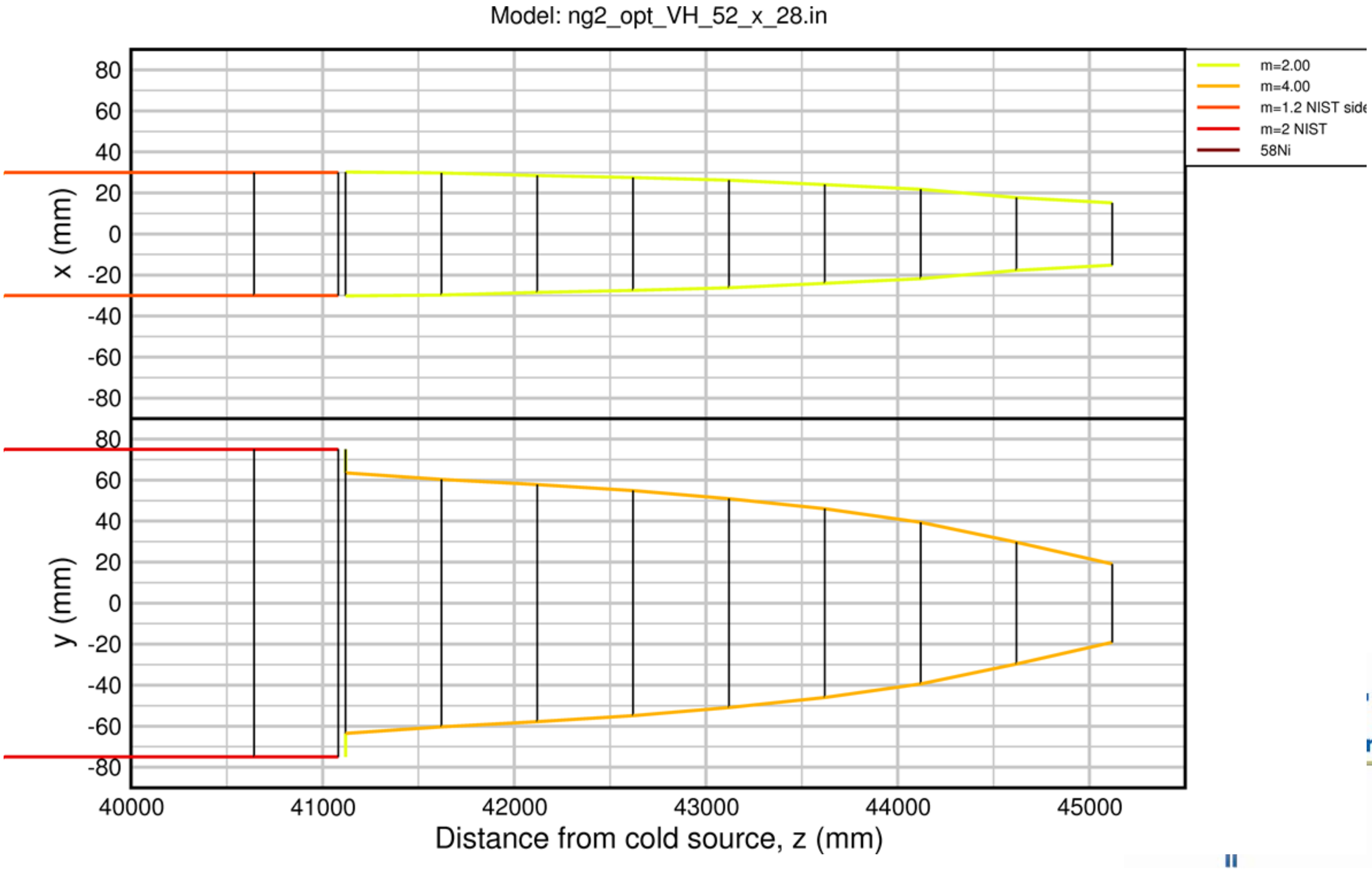
NIST Center for Neutron Research

Neutronics simulations

Guide simulation: Guide profile optimization

Example: re-optimization of focusing guide for HFBS

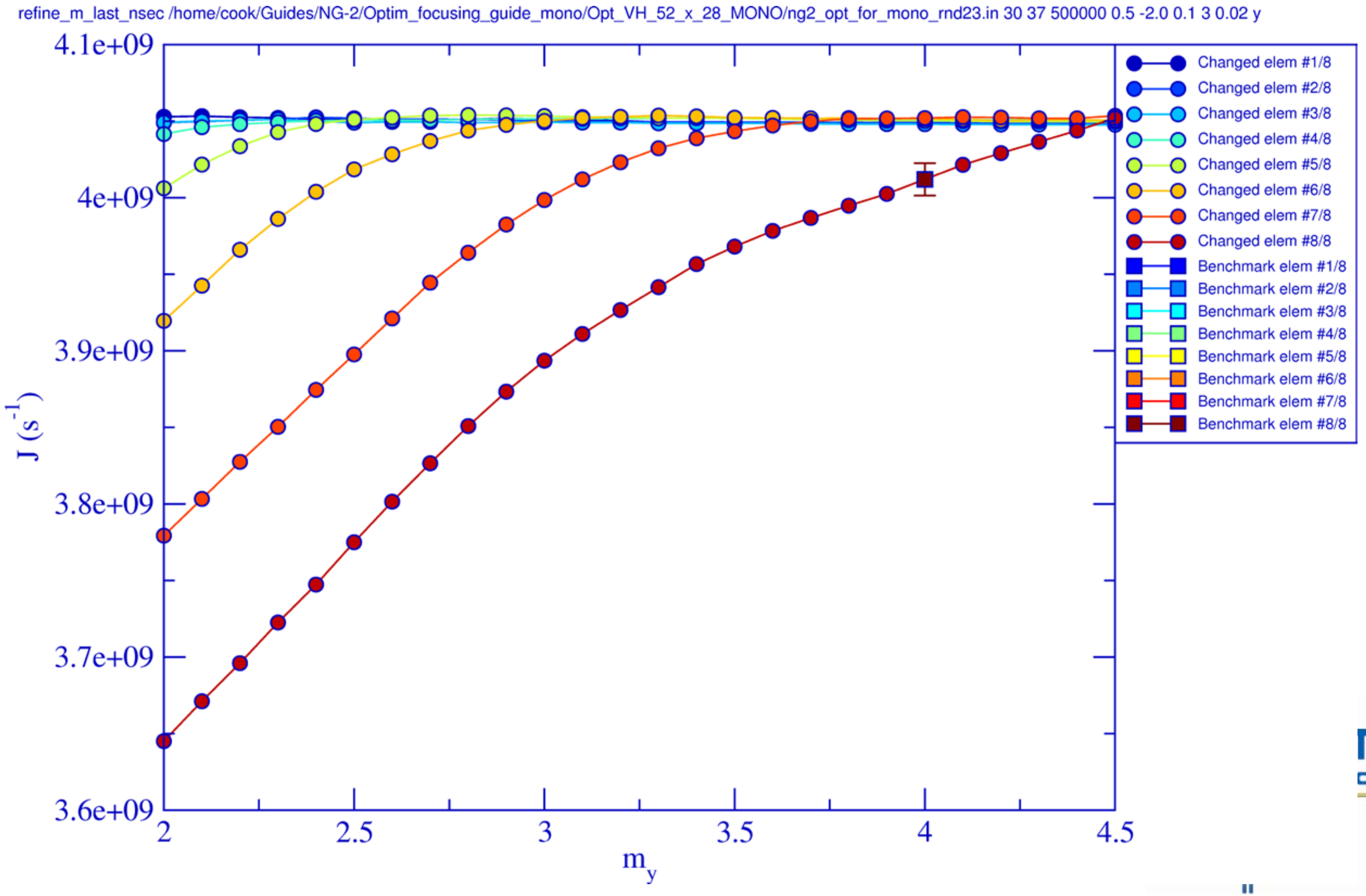
Optimized profile config



Neutronics simulations

Guide simulation: Guide mirror optimization

- Optimization of *m* (top/bottom surfaces)

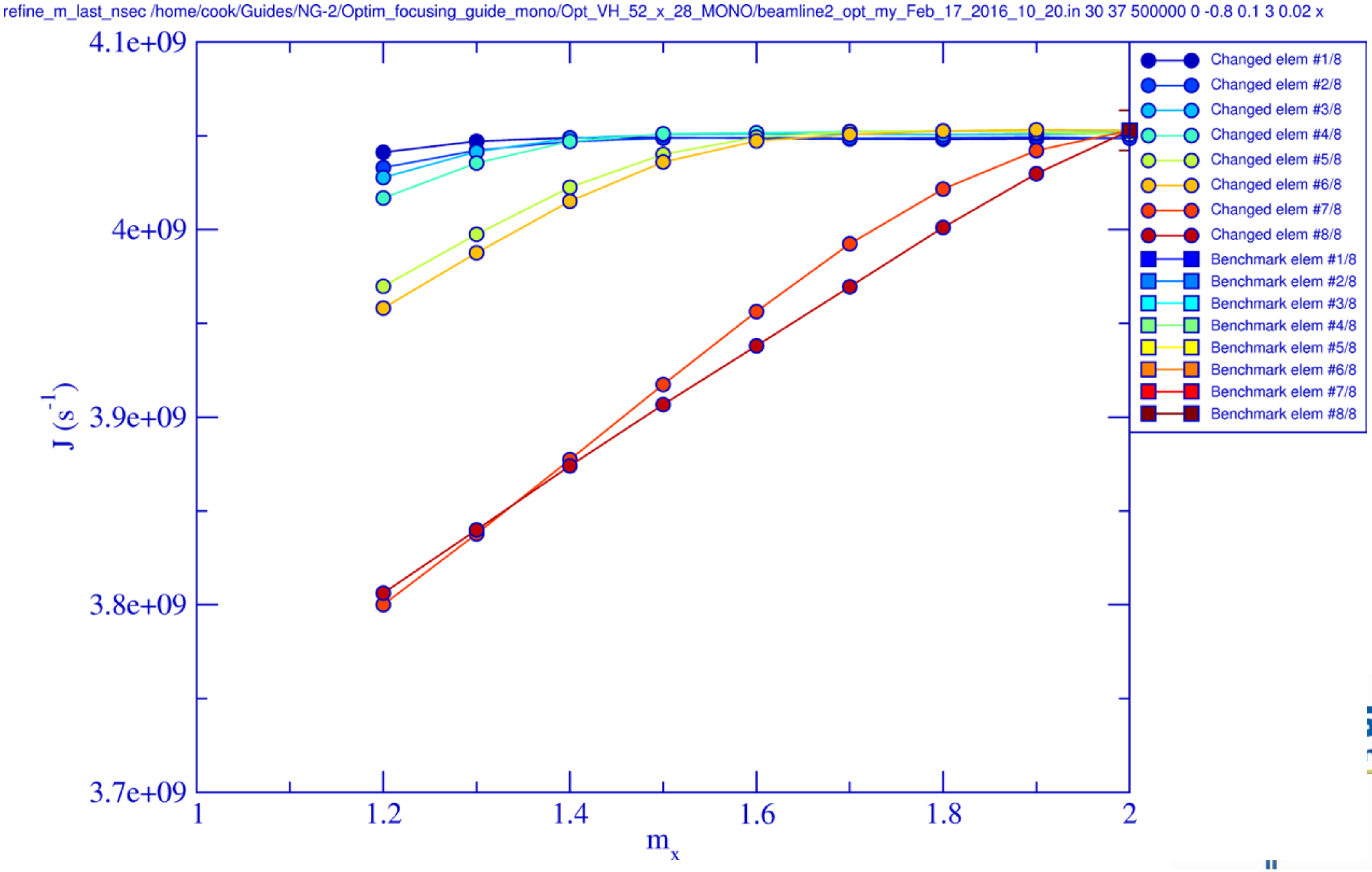


```
refine_m_last_nsec /home/cook/Guides/NG-2/Optim_focusing_guide_mono/Opt_VH_52_x_28_MONO/ng2_opt_for_mono_rnd23.in 30 37 500000 0.5 -2.0 0.1 3 0.02 y
```

Neutronics simulations

Guide simulation: Guide mirror optimization

- Optimization of *m* (sides)

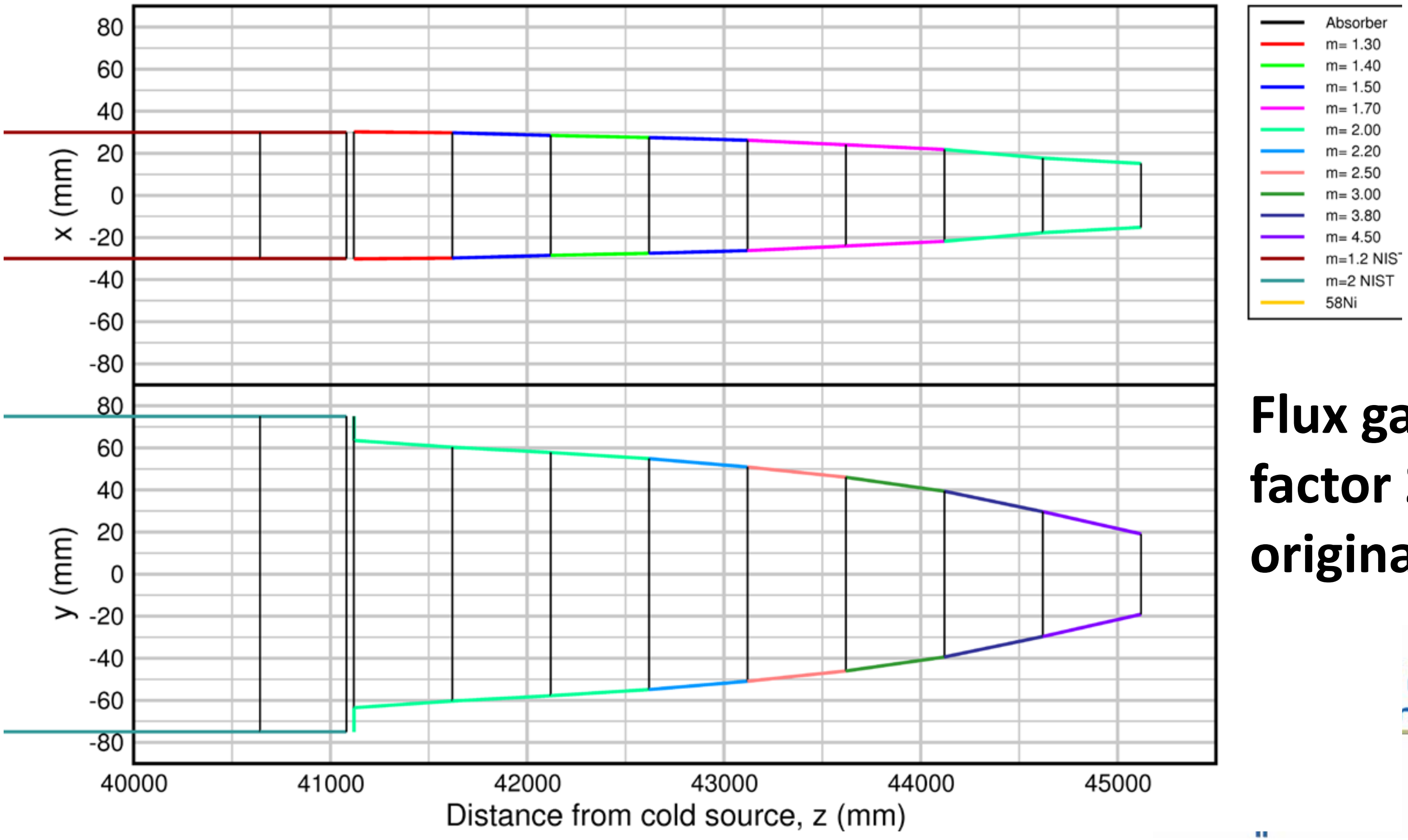


Neutronics simulations

Guide simulation: Guide mirror optimization

Example: re-optimization of focusing guide for HFBS

Final optimized profile config with optimized *m* coatings


Model:ng2_opt_VH_52_x_28_opt_mxy_to_gex.in

Flux gain ~ factor 2 wrt original

Neutronics simulations

Guide simulation: Guide profile optimization

- Why optimize (minimize) *m*?
  - Reduces supermirror cost (in principle) – (number of supermirror layers increases ~ m^4, cost ∝ thickness~ m^3??

  - Can eliminate unwanted transmitted neutrons that never reach target

  - BUT… recently manufacturers prefer modest number (coarser) *m* “steps” per order (control number of sputtering machine setups!)

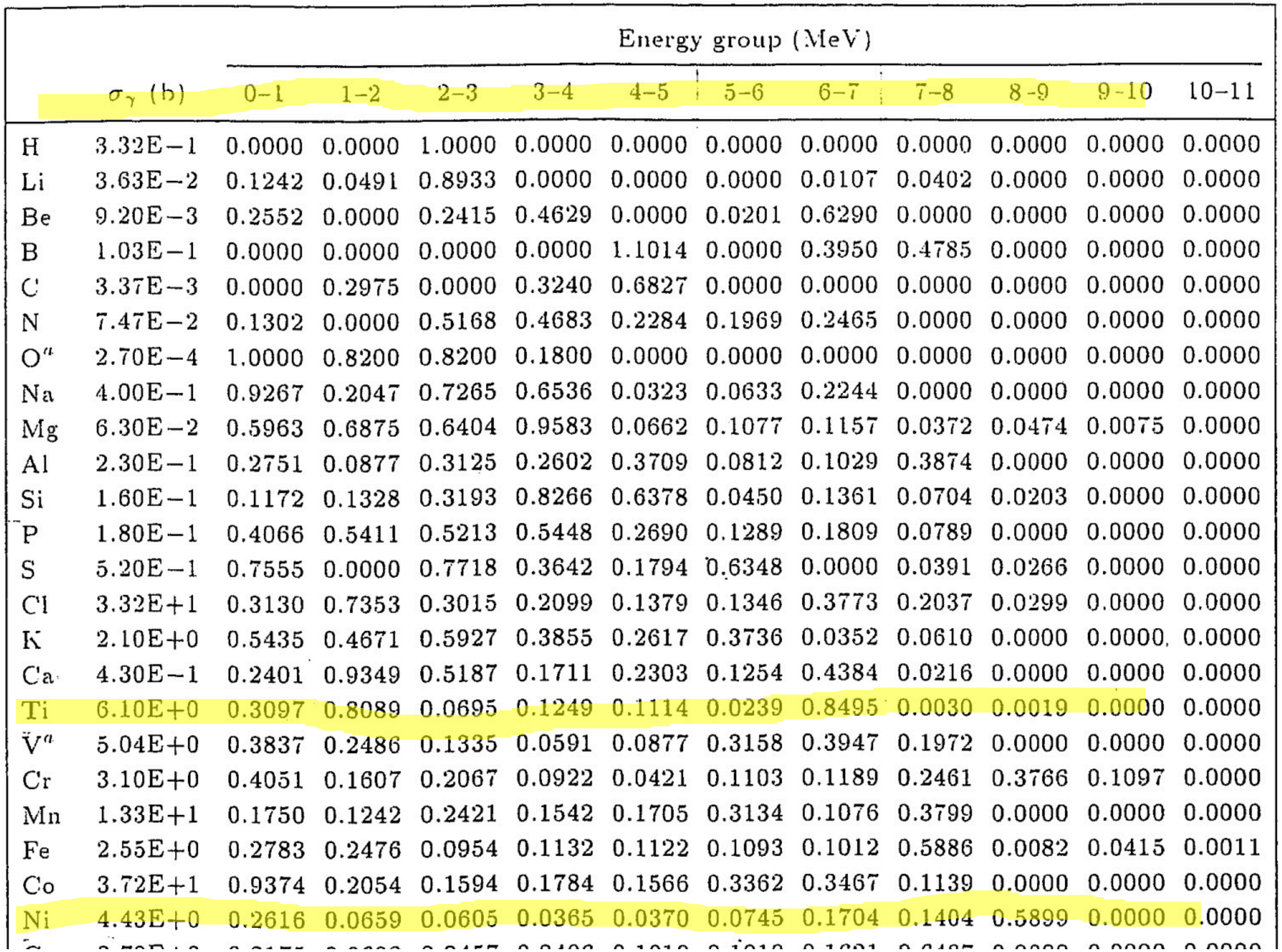
  - Reducing *m* reduces supermirror thickness ($\propto m^3$) and consequently **gamma production** – (energetic gammas from supermirror materials often drive guide shield thickness/ weight)

NIST Center for Neutron Research

Neutronics simulations

Guide simulation: Guide profile optimization

- Why optimize *m*?



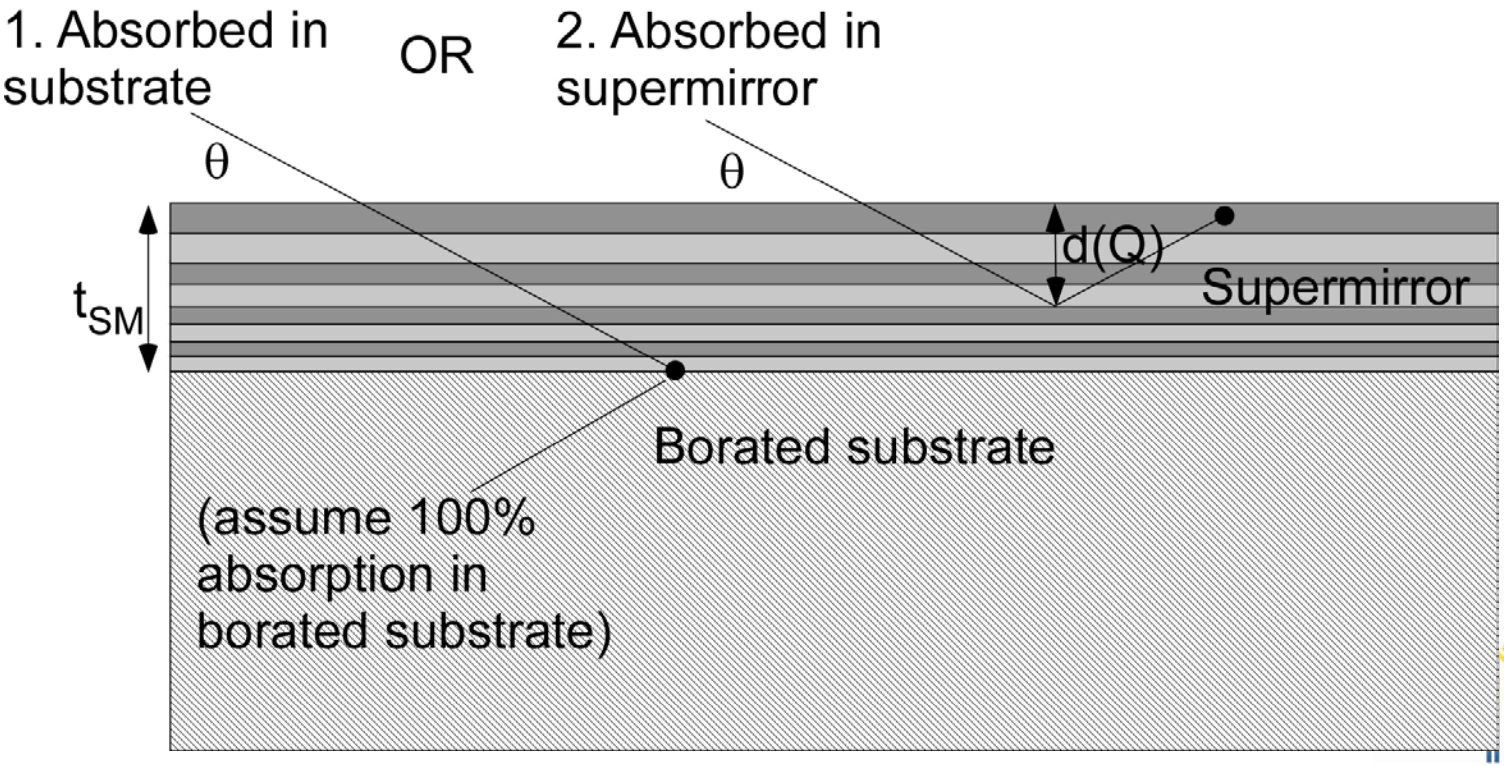
	σ_γ (b)	Energy group (MeV) 0–1	1–2	2–3	3–4	4–5	5–6	6–7	7–8	8–9	9–10	10–11
H	3.32E−1	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Li	3.63E−2	0.1242	0.0491	0.8933	0.0000	0.0000	0.0000	0.0107	0.0402	0.0000	0.0000	0.0000
Be	9.20E−3	0.2552	0.0000	0.2415	0.4629	0.0000	0.0201	0.6290	0.0000	0.0000	0.0000	0.0000
B	1.03E−1	0.0000	0.0000	0.0000	0.0000	1.1014	0.0000	0.3950	0.4785	0.0000	0.0000	0.0000
C	3.37E−3	0.0000	0.2975	0.0000	0.3240	0.6827	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
N	7.47E−2	0.1302	0.0000	0.5168	0.4683	0.2284	0.1969	0.2465	0.0000	0.0000	0.0000	0.0000
O[a]	2.70E−4	1.0000	0.8200	0.8200	0.1800	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Na	4.00E−1	0.9267	0.2047	0.7265	0.6536	0.0323	0.0633	0.2244	0.0000	0.0000	0.0000	0.0000
Mg	6.30E−2	0.5963	0.6875	0.6404	0.9583	0.0662	0.1077	0.1157	0.0372	0.0474	0.0075	0.0000
Al	2.30E−1	0.2751	0.0877	0.3125	0.2602	0.3709	0.0812	0.1029	0.3874	0.0000	0.0000	0.0000
Si	1.60E−1	0.1172	0.1328	0.3193	0.8266	0.6378	0.0450	0.1361	0.0704	0.0203	0.0000	0.0000
P	1.80E−1	0.4066	0.5411	0.5213	0.5448	0.2690	0.1289	0.1809	0.0789	0.0000	0.0000	0.0000
S	5.20E−1	0.7555	0.0000	0.7718	0.3642	0.1794	0.6348	0.0000	0.0391	0.0266	0.0000	0.0000
Cl	3.32E+1	0.3130	0.7353	0.3015	0.2099	0.1379	0.1346	0.3773	0.2037	0.0299	0.0000	0.0000
K	2.10E+0	0.5435	0.4671	0.5927	0.3855	0.2617	0.3736	0.0352	0.0610	0.0000	0.0000	0.0000
Ca	4.30E−1	0.2401	0.9349	0.5187	0.1711	0.2303	0.1254	0.4384	0.0216	0.0000	0.0000	0.0000
Ti	6.10E+0	0.3097	0.8089	0.0695	0.1249	0.1114	0.0239	0.8495	0.0030	0.0019	0.0000	0.0000
V[a]	5.04E+0	0.3837	0.2486	0.1335	0.0591	0.0877	0.3158	0.3947	0.1972	0.0000	0.0000	0.0000
Cr	3.10E+0	0.4051	0.1607	0.2067	0.0922	0.0421	0.1103	0.1189	0.2461	0.3766	0.1097	0.0000
Mn	1.33E+1	0.1750	0.1242	0.2421	0.1542	0.1705	0.3134	0.1076	0.3799	0.0000	0.0000	0.0000
Fe	2.55E+0	0.2783	0.2476	0.0954	0.1132	0.1122	0.1093	0.1012	0.5886	0.0082	0.0415	0.0011
Co	3.72E+1	0.9374	0.2054	0.1594	0.1784	0.1566	0.3362	0.3467	0.1139	0.0000	0.0000	0.0000
Ni	4.43E+0	0.2616	0.0659	0.0605	0.0365	0.0370	0.0745	0.1704	0.1404	0.5899	0.0000	0.0000

Prompt gammas per neutron capture

Neutronics simulations
Rough dimensioning of guide shielding

- Monte Carlo simulation can store lost neutron information which can be used to estimate gamma source for shielding calculations
  - **Example: Ni-Ti supermirror on borated glass substrate**

Trajectory “fails” reflectivity R(Q), prob 1-R(Q)



Neutronics simulations

Rough dimensioning of guide shielding

- Simplifications for supermirror (use approximate models)

$$t_{SM}(m)[\mathrm{cm}] \approx 1.33\times10^{-5} m^{3}$$

$$f_{Ni} \approx f_0 + (1 - f_0) m^{-2.83}; \quad with\ f_0 = 0.511$$

$$t_{Ni} = f_{Ni} t_{SM}(m)$$

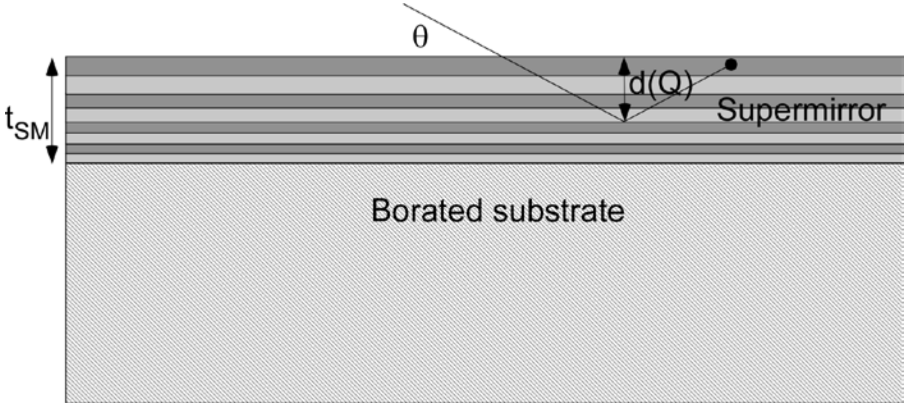
$$t_{Ti} = t_{SM}(m) - t_{Ni}$$

etc.

Neutronics simulations

Rough dimensioning of guide shielding

- Relative probability of processes (**quite difficult to do rigorously!)**
  1. **Absorption in borated substrate** (assume 100% absorption, if reached) – production of $I_{B,gp}$ gamma (0.93 478keV gammas in corresponding energy group)
  2. **Absorption in supermirror** (worse for shielding) – production of Ni or Ti capture gamma
     - i. Using $\Sigma_{SM}(\lambda)t_{SM} \approx \Sigma_{Ni}(\lambda)t_{Ni} + \Sigma_{Ti}(\lambda)t_{Ti}$
     - ii. gp. gamma yield $I_{Ni,gp}$ for Ni **pro-rated** $P_{Ni} = \dfrac{\Sigma_{Ni}(\lambda)t_{Ni}}{\Sigma_{SM}(\lambda)t_{SM}}$ (and similar for Ti)
     - iii. **PROBLEM**: Where is neutron absorbed in SM? **Very conservative (upper limit):** choose d(Q)=t_{SM} and assume P(abs SM)=**P(<u>not</u> transmitted through maximum path 2t_{SM}/sinθ)**



Neutronics simulations

Rough dimensioning of guide shielding

Then

$$P_1\left(\text{abs substrate}\right) \square \frac{\exp\left(-\Sigma_{SM}\left(\lambda\right)\frac{t_{SM}}{\sin\theta}\right)}{\left[\exp\left(-\Sigma_{SM}\left(\lambda\right)\frac{t_{SM}}{\sin\theta}\right)+\left(1-\exp\left(-2\Sigma_{SM}\left(\lambda\right)\frac{t_{SM}}{\sin\theta}\right)\right)\right]} \longrightarrow I_{B,gp}$$

=0.93 into 478keV gamma energy gp

$$P_2\left(\text{abs SM}\right) \square \frac{1-\exp\left(-2\Sigma_{SM}\left(\lambda\right)\frac{t_{SM}}{\sin\theta}\right)}{\left[\exp\left(-\Sigma_{SM}\left(\lambda\right)\frac{t_{SM}}{\sin\theta}\right)+\left(1-\exp\left(-2\Sigma_{SM}\left(\lambda\right)\frac{t_{SM}}{\sin\theta}\right)\right)\right]}$$

$\nearrow$ $P_{Ni}I_{Ni,gp}$

$\searrow$ $P_{Ti}I_{Ti,gp}$

gamma spectrum from lost neutron weight w_n(1-R(Q)) is

$$w_n\left(1-R\left(Q\right)\right)\left[P_1 I_{B,gp}+P_2\left(P_{Ni}I_{Ni,gp}+P_{Ti}I_{Ti,gp}\right)\right]$$

A more sophisticated treatment (published 2019):

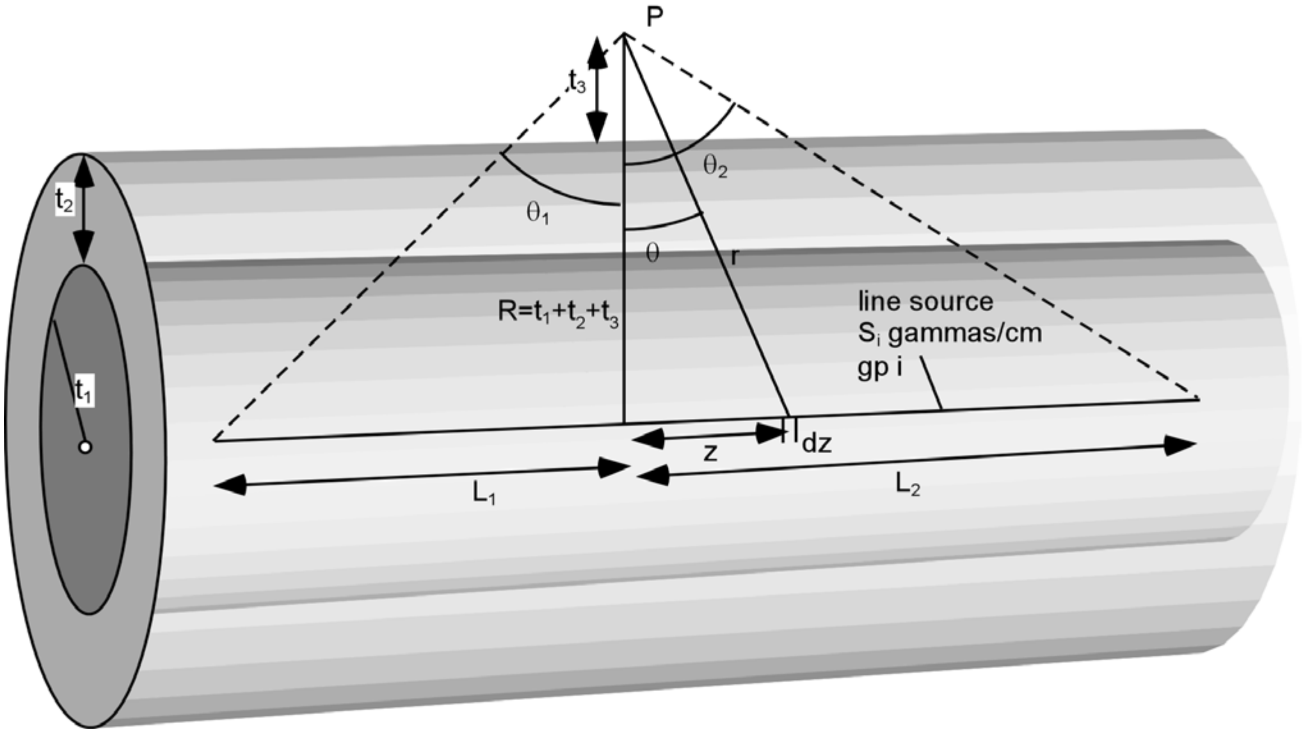
"Neutron absorption in supermirror coatings: Effects on shielding", R. Kolevatov, C. Schanzer, P. Böni, Nuclear Inst. and Methods in Physics Research, A 922 (2019) 98–107

NIST Center for Neutron Research

Neutronics simulations

Rough dimensioning of guide shielding

- Can used simplified source and shielding geometry
  1. e.g. Approximate guide gamma source as a line source
  2. e.g. Approximate shield as a concentric cylinder etc.
  3. Calculate *buildup factors* as a matrix wrt *r* and gamma energy group
  4. Numerical integration of dose rate at P

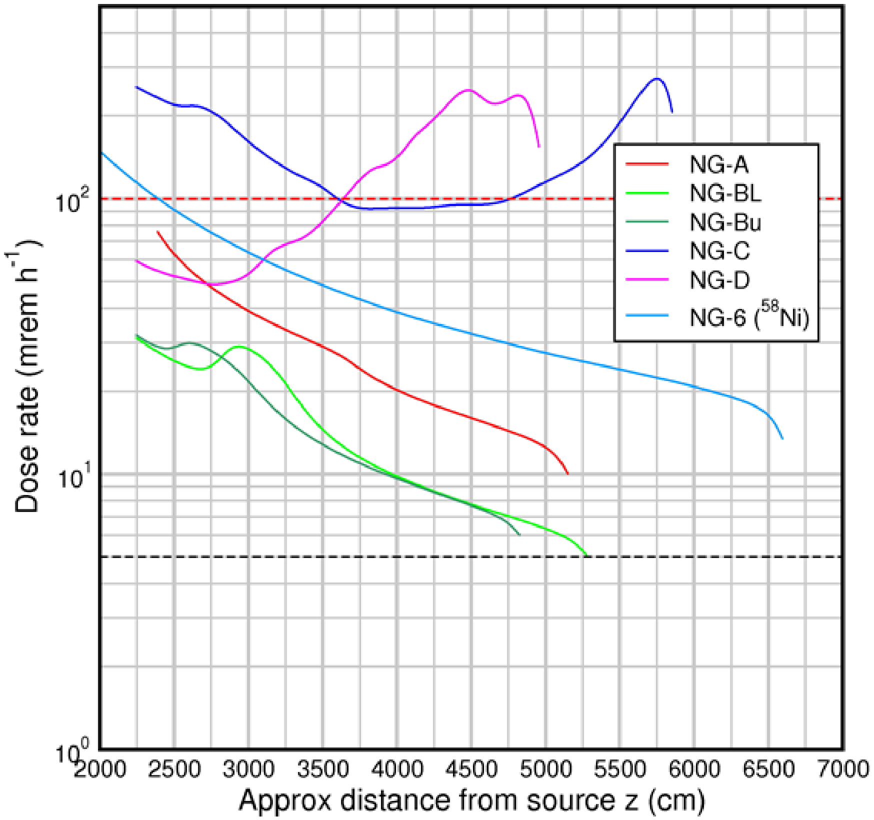


Neutronics simulations

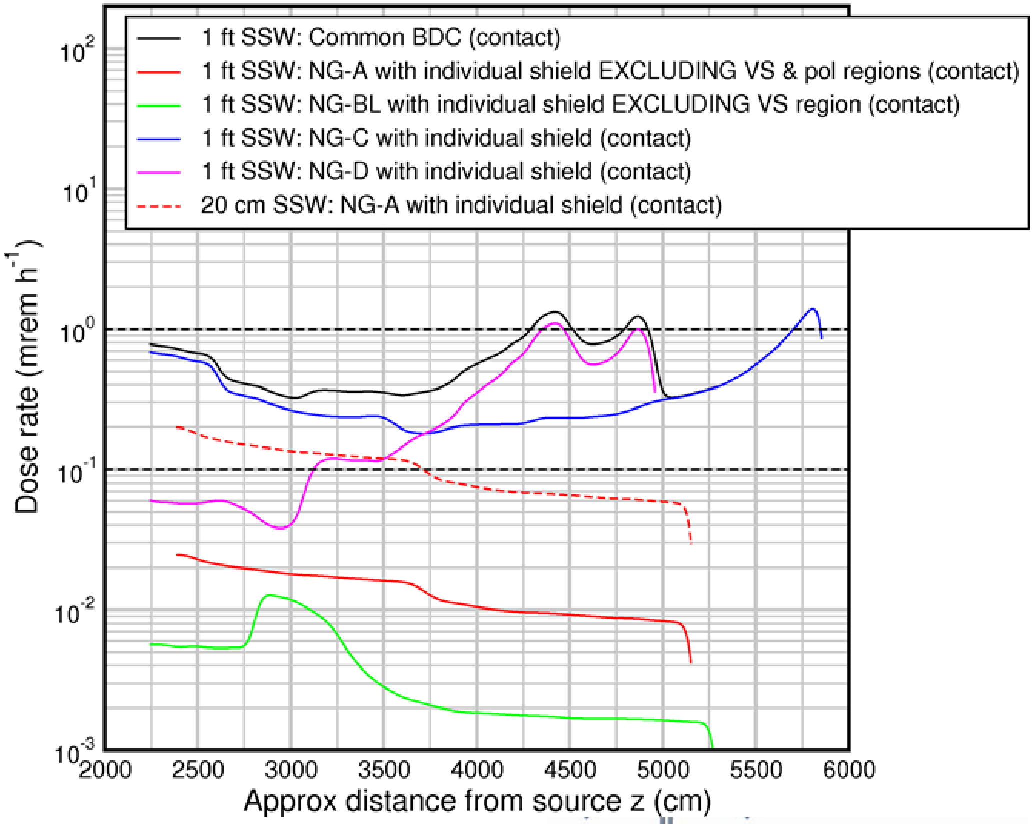
Rough dimensioning of guide shielding

- Some examples of simplified line source-cylindrical shield calculations for NGA-D

Est. dose rates at ~1m from UNSHIELDED guides due to γ line source
LH$_2$ cold source (region inside guide hall)



LH$_2$ cold source (region inside guide hall)

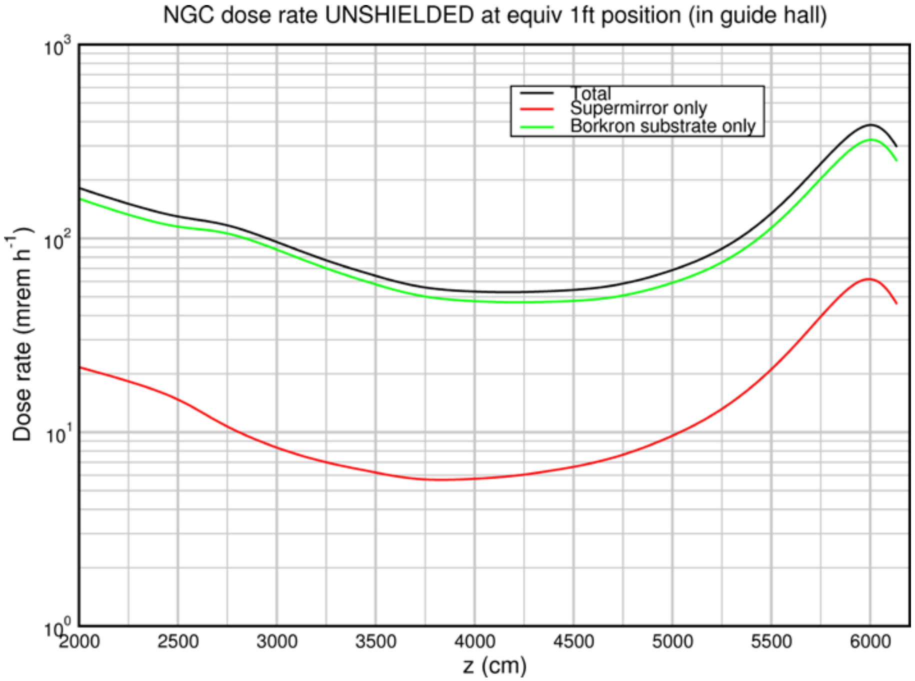


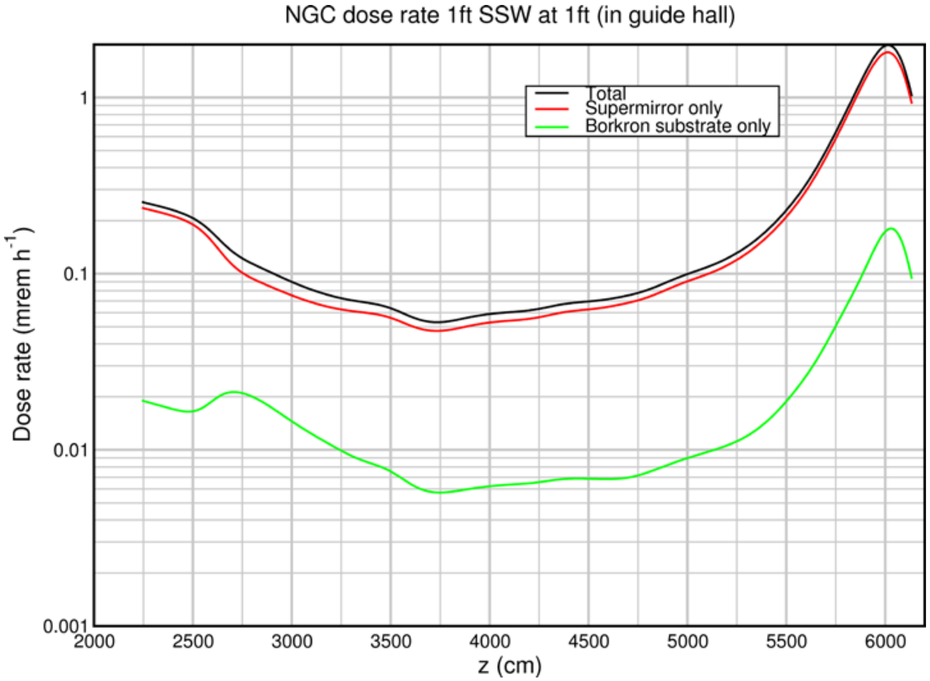
Neutronics simulations

Rough dimensioning of guide shielding

Supermirror contribution to dose rate simplified line source-cylindrical shield calculations for NGC

UNSHIELDED (**substrate dominates**) SHIELDED (**supermirror dominates**)





Neutronics simulations

Rough dimensioning of guide shielding

- Other in-guide sources (e.g. V or double-V polarizer) may require enhanced shielding

- e.g. **VSANS double-V**
  - ~ **0.3mm thick Si** at 0.75° × 2 ≈ **4.6cm Si traversed by beam**
  - Usually requires more than the standard 30cm SSW on neutron beams in the NCNR guide hall

Neutronics simulations

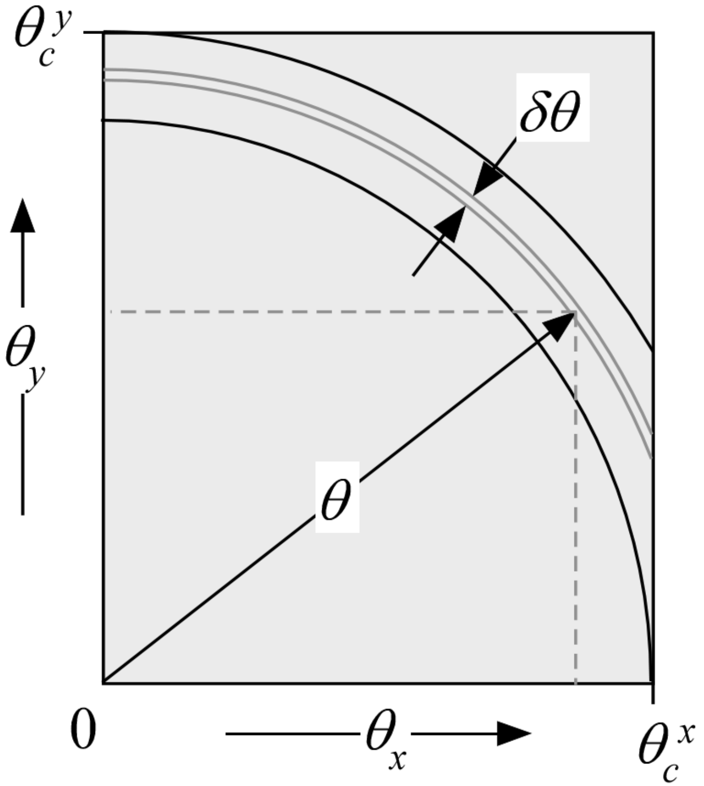
MCNP

- MCNP?
  - MCNP ***cannot*** do coherent scattering required for neutron transport in guides
  - ***Can*** approximate the neutron beam at the *exit* of a guide with required spectrum and energy-dependent divergence (remember $\theta_c \propto \lambda$)
  - Some limitations on MCNP user-defined source: e.g. cannot decouple horizontal and vertical divergence differences ⇒ approximate by mean polar angle

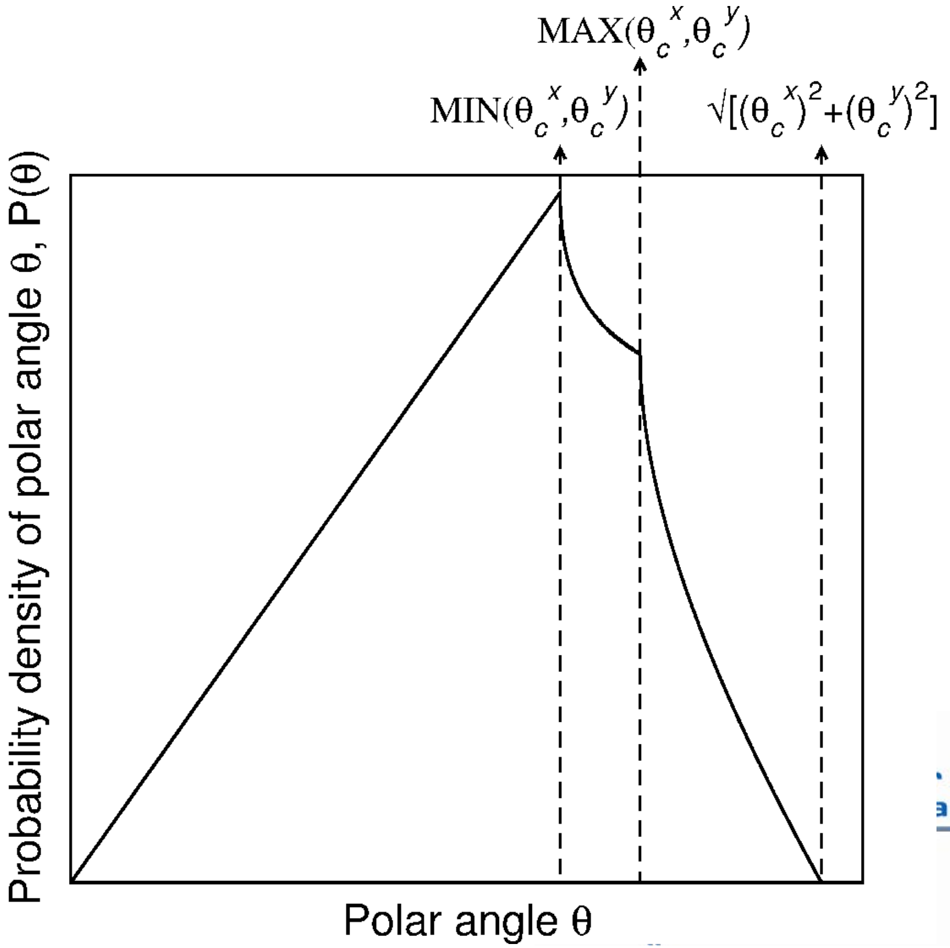
NIST Center for Neutron Research

Neutronics simulations
MCNP

- MCNP source approximation at exit of “perfect” neutron guide



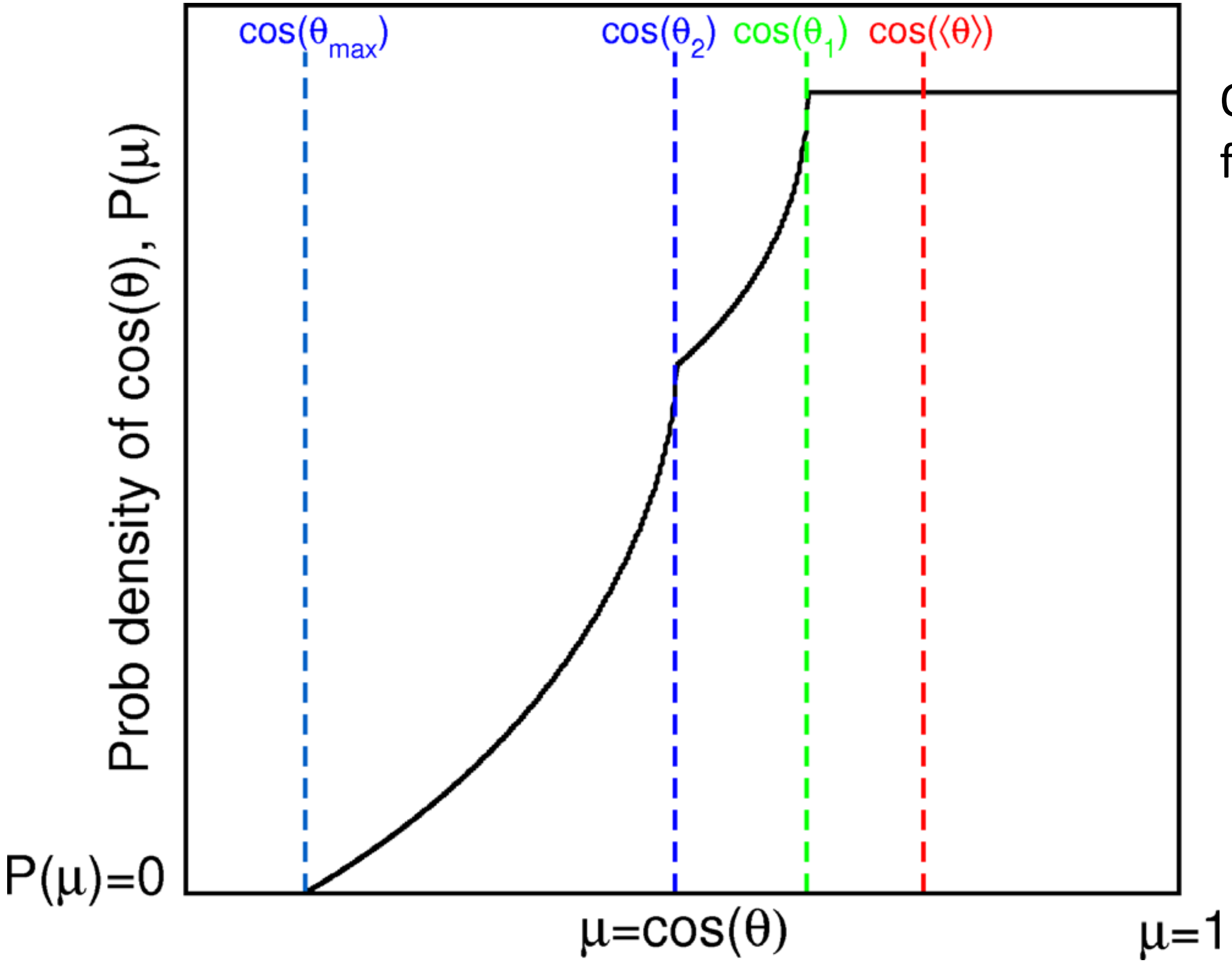
One such distribution for **each wavelength**



Neutronics simulations

MCNP

- MCNP source approximation at exit of “perfect” neutron guide



One such distribution for **each wavelength**

$$\theta_1 = \min\left(\theta_c^x, \theta_c^y\right)$$

$$\theta_2 = \max\left(\theta_c^x, \theta_c^y\right)$$

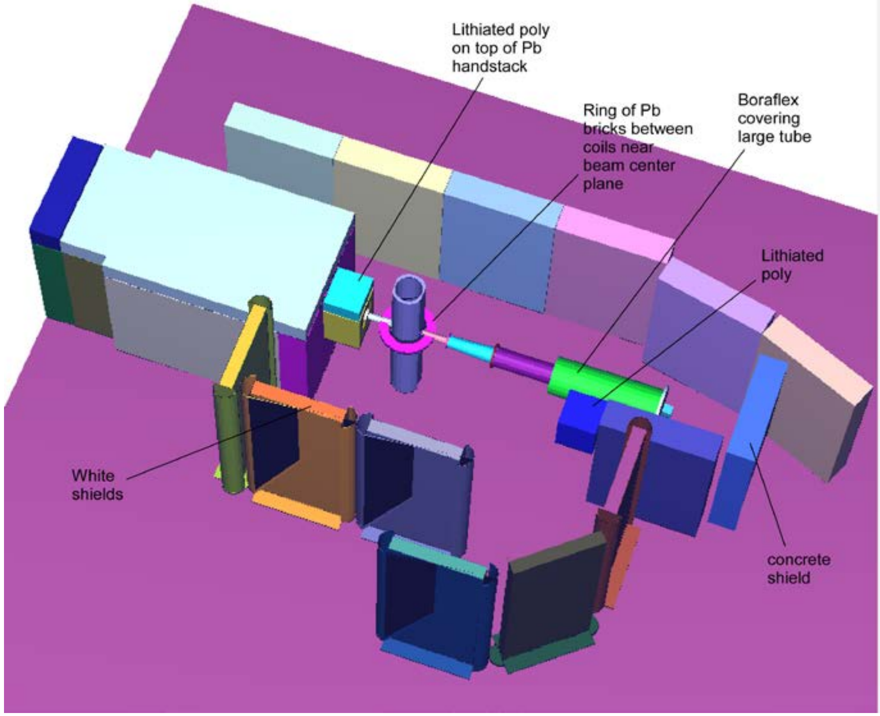
$$\theta_{max} = \sqrt{\left(\theta_c^x\right)^2 + \left(\theta_c^y\right)^2}$$

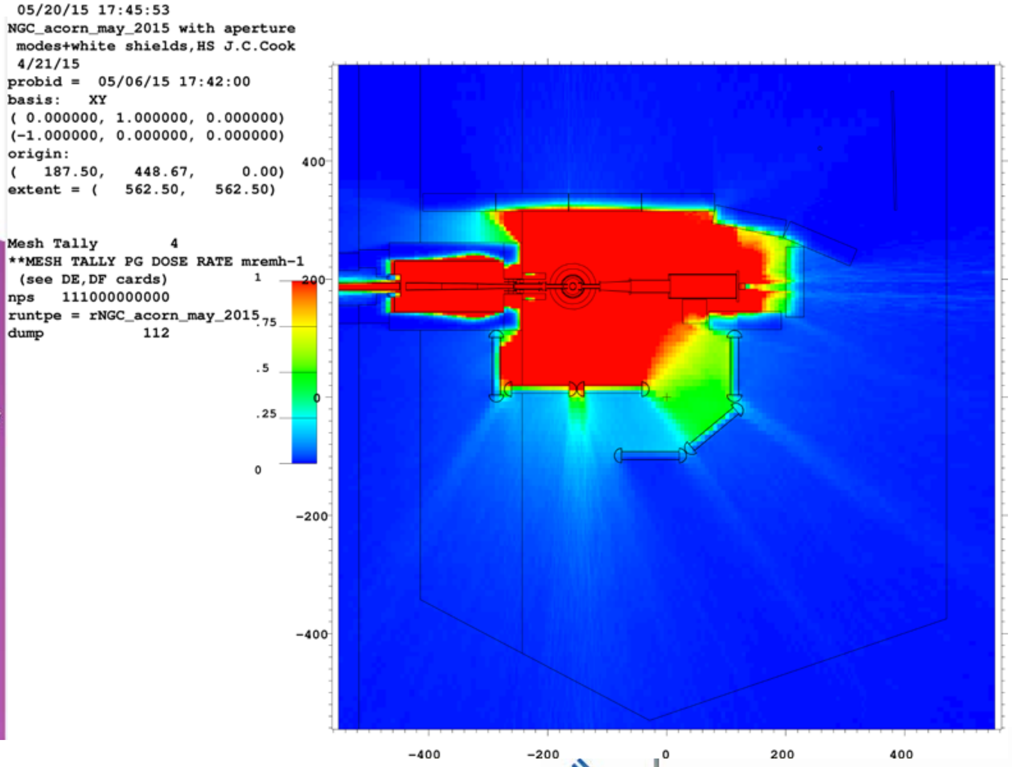
NIST Center for Neutron Research

Neutronics simulations

MCNP (with source derived from guide simulation)

- ACORN (NGC) shielding (MCNP source spectrum from guide simulation with energy-dependent divergence)





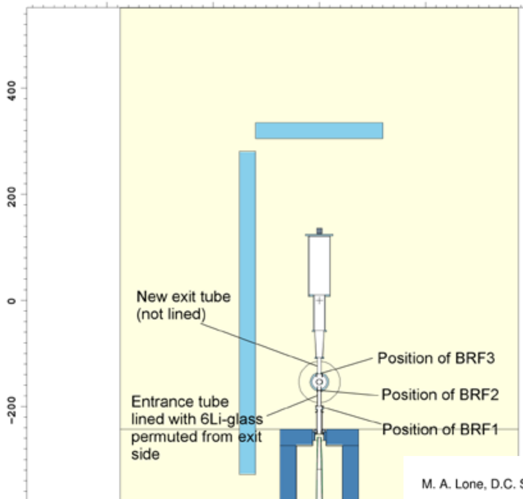
```
 05/20/15 17:45:53
NGC_acorn_may_2015 with aperture
 modes+white shields,HS J.C.Cook
 4/21/15
probid =  05/06/15 17:42:00
basis:   XY
( 0.000000, 1.000000, 0.000000)
(-1.000000, 0.000000, 0.000000)
origin:
(   187.50,   448.67,      0.00)
extent = (   562.50,   562.50)


Mesh Tally         4
**MESH TALLY PG DOSE RATE mremh-1
 (see DE,DF cards)
nps   111000000000
runtpe = rNGC_acorn_may_2015
dump             112
```

Neutronics simulations

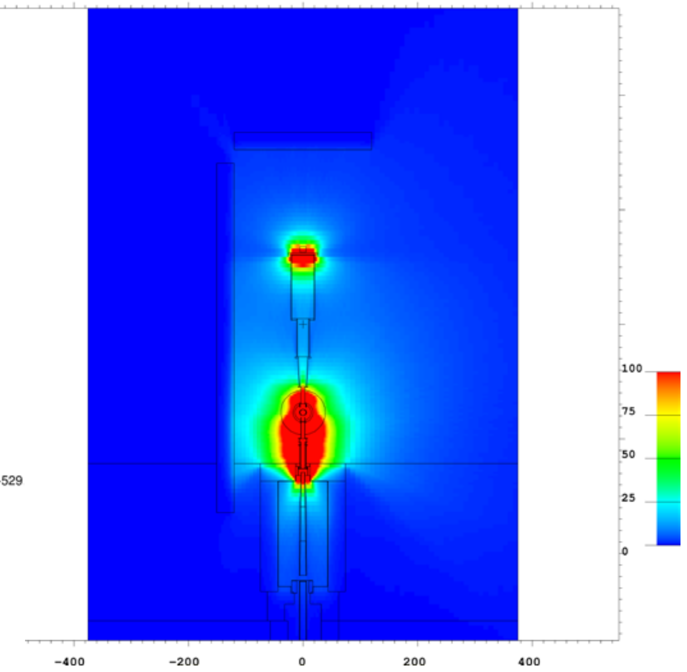
MCNP (with source derived from guide simulation)

- ACORN (NGC) shielding (Fast neutrons from ^{6}Li-containing materials using “cell sources” emitting fast n spectrum)

```
 03/20/14 14:04:21
NGC_acorn_2014a updated config w
 emiT coll J.C.Cook Feb 3, 2014

probid =  03/20/14 14:04:18
basis:    XY
( 1.000000,  0.000000,  0.000000)
( 0.000000,  1.000000,  0.000000)
origin:
(     0.00,    448.67,      0.00)
extent = (    551.34,    551.34)
```

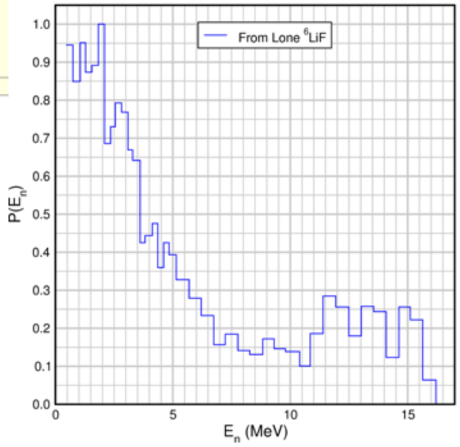


```
 03/24/14 09:47:08
NGC_acorn_2014a_6Li_fast fast n
 6Li(n,a) for NGC_acorn_2014a
 J.C.Cook 3/21/14
probid =  03/21/14 10:57:15
basis:    XY
( 1.000000,  0.000000,  0.000000)
( 0.000000,  1.000000,  0.000000)
origin:
(     0.00,    448.67,      0.00)
extent = (    551.34,    551.34)

Mesh Tally        14
 **MESH TALLY NEUTRON DOSE RATE
 mremh-1 (see DE,DF cards)
nps         150000000
runtpe = rNGC_acorn_2014a_6Li_fast
dump             16
```



Approximation to distribution of fig. 5 of ref.

M. A. Lone, D.C. Santry, and W. M. Inglis, Nucl.Instr.&.Meth 174 (1980) 521-529



Neutronics simulations

END

😴