Ultracold neutrons (UCN) simulations using Geant4

Marcin Kuźniak
(for the UCN group)

NIM A **552** (2005) 513

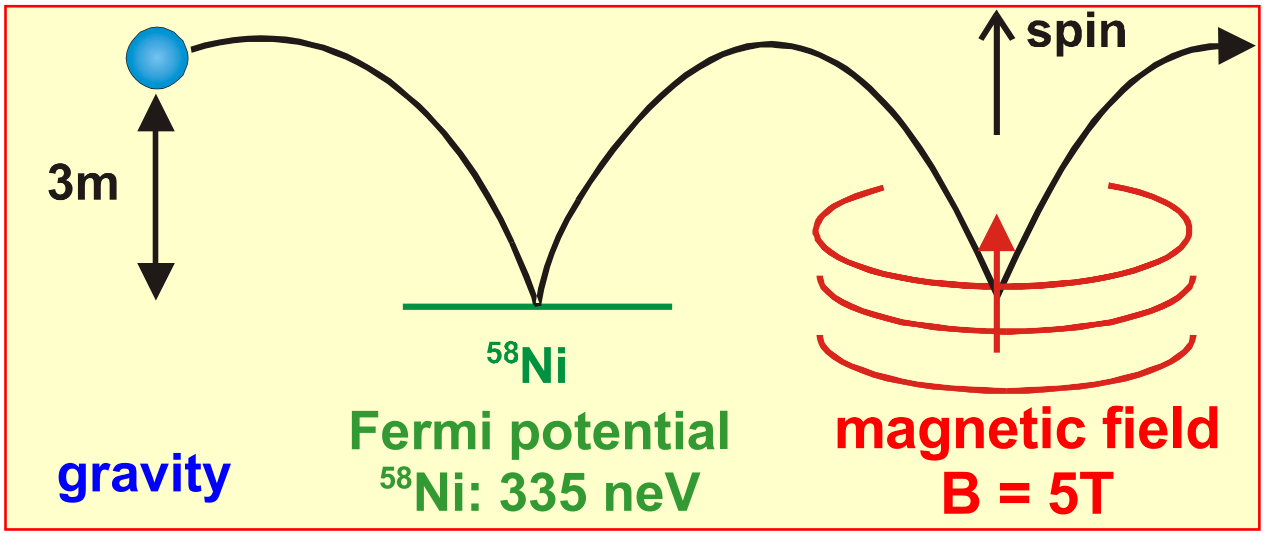
International Workshop on Applications of Advanced Monte Carlo Simulations in Neutron Scattering

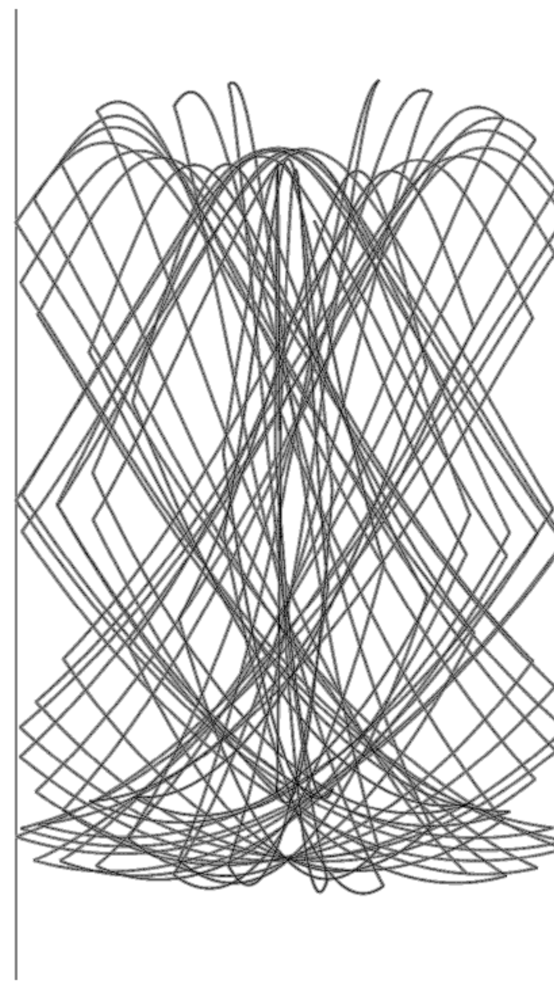
Outline

- Motivation
  - Required functionality
  - Geant4 features and limitations
- Implementation of UCN physics
  - Gravity, materials, magnetic fields, spin
- Benchmarks against other codes
- Example applications
- Simulation of the neutron electric dipole moment experiment

UCN properties

- Affected by:





$V_g = mgh$

$V_F \sim 300$ neV

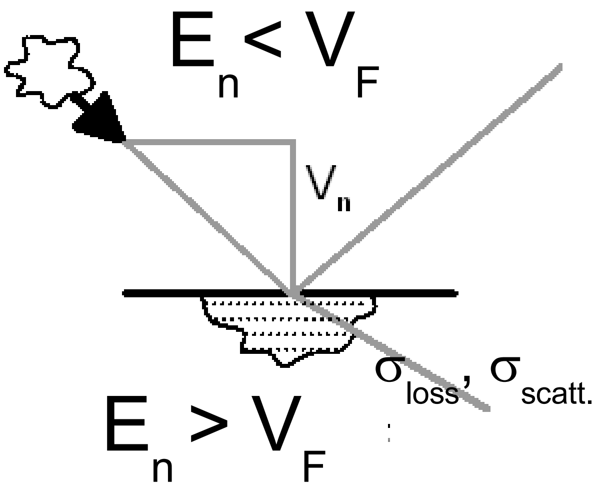
$\mu = -60$ neV/T

Geant4

- Geant4 toolkit:
  - Provides universal, widely used tracking engine
  - Complicated geometries are easy to implement
  - But does not contain the UCN physics
- Most of the 'conventional' MC codes solve problems analytically (problems with complicated geometries)
- Extended version => Geant4UCN

Changes

- Gravity
- User defined material properties
  - ... used for boundary processes:
    - diffuse and specular reflection, transmission, spin flip
  - ... and interactions with materials:
    - absorption, losses, scattering
- Shutters, which can change the state in time
- Spin propagation using the Bloch-equation (adiabatic assumption)

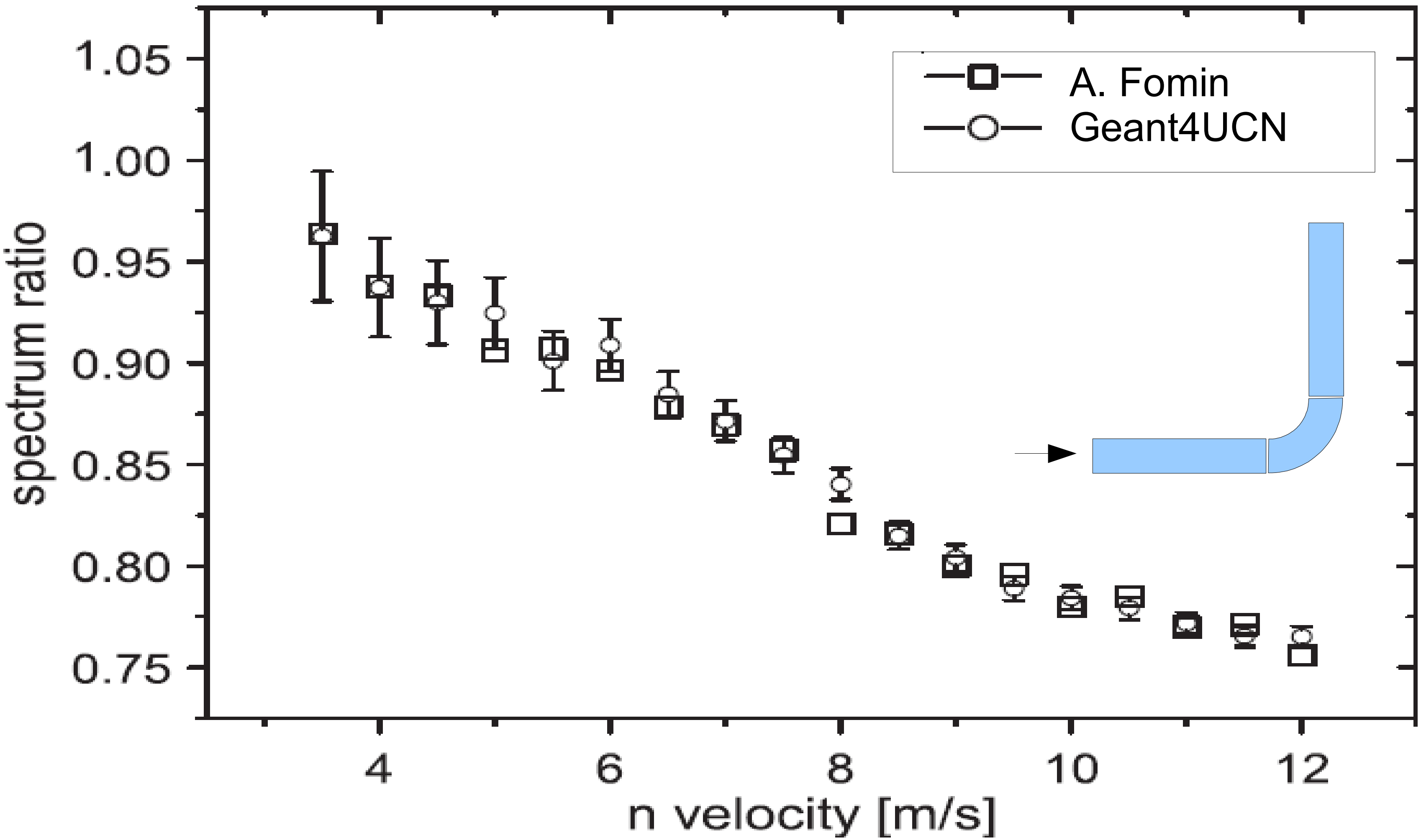


Benchmarks

- Tests against conventional MC code (by A. Fomin):
  - Velocity spectrum transformation through a simple guide system
  - Wall collision frequency during storage in a Be-coated bottle
- Test against VITESS (G. Zsigmond)
  - Propagation through a neutron guide
- Comparison with analytical calculations
  - Spin rotation

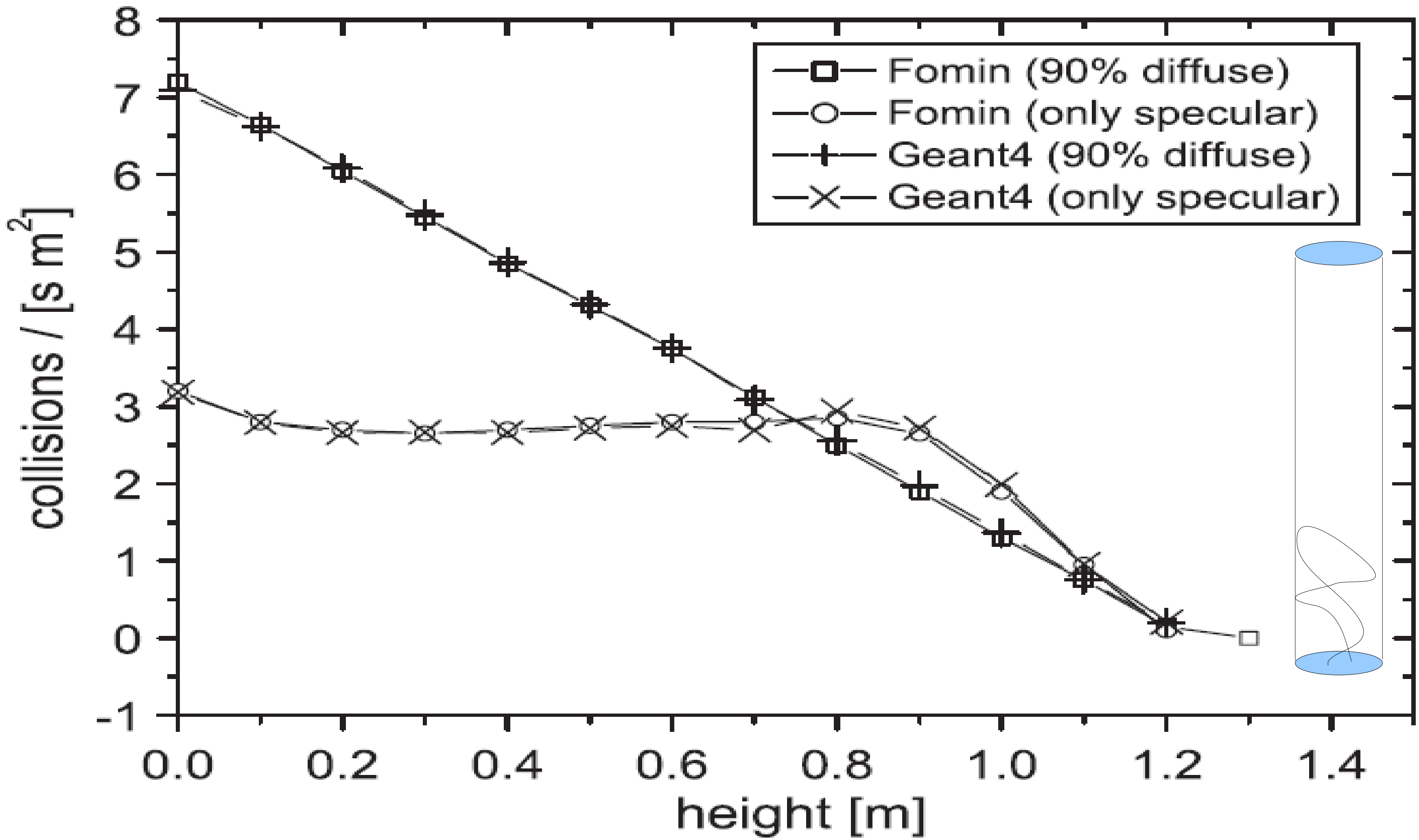
ν transformation

2m horizontal section + 90º bend (1 m radius) + 2 m vertical section

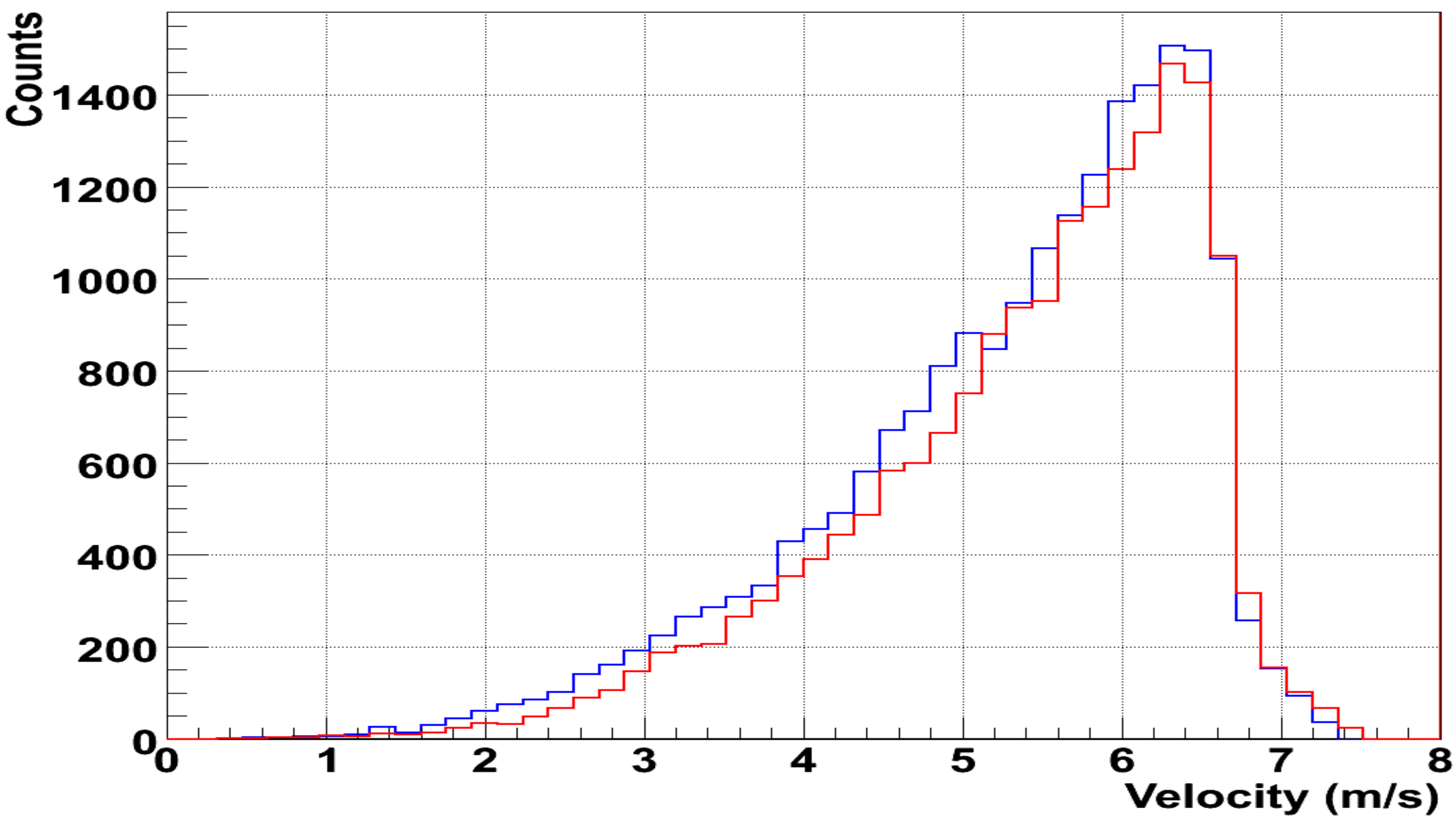


Collision frequency

Be-coated bottle (R=250mm, h=2700mm), mono-energetic UCNs (131neV)



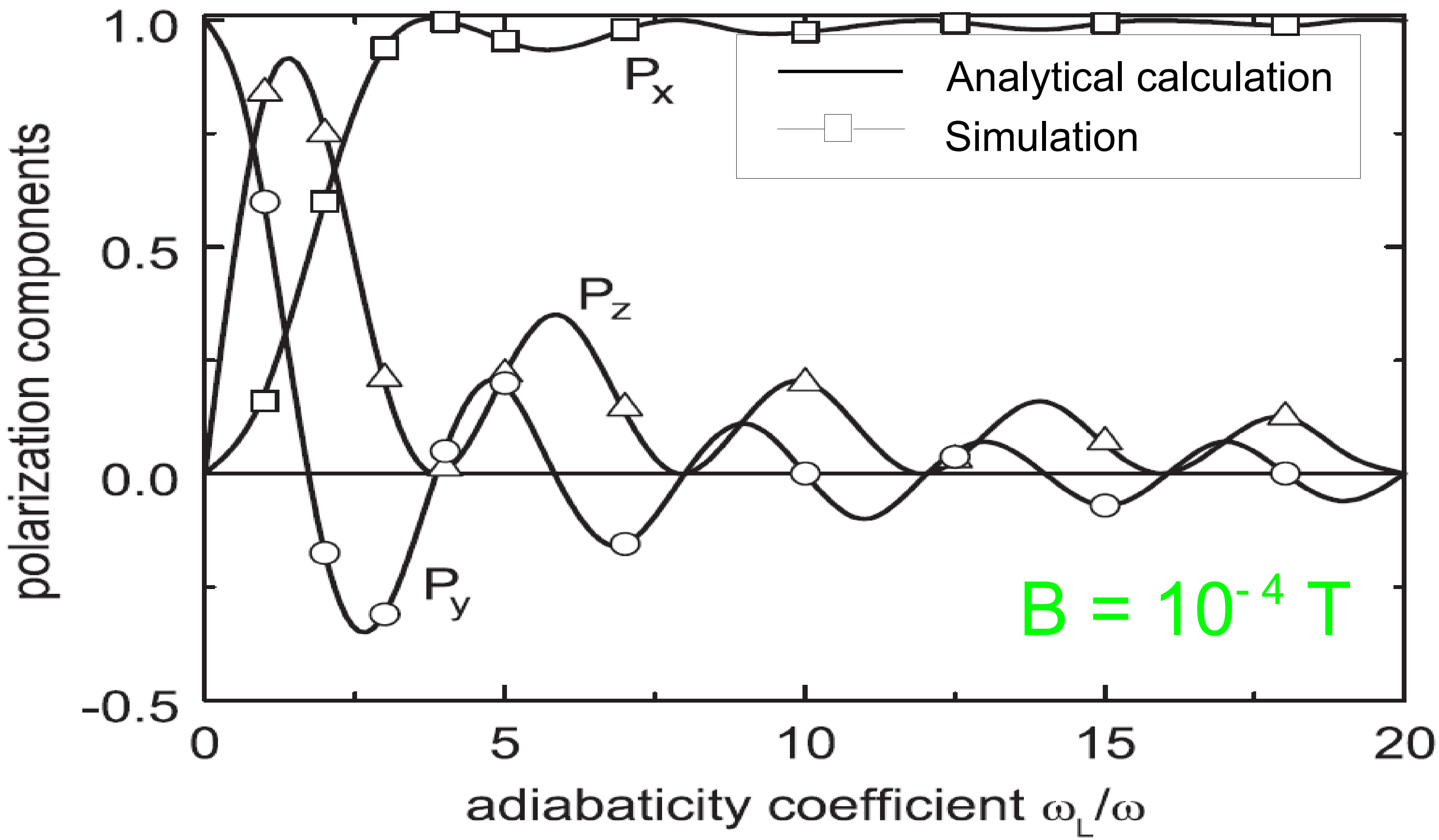
Energy spectra after 5 m long guide



$\pi/2$ rotation

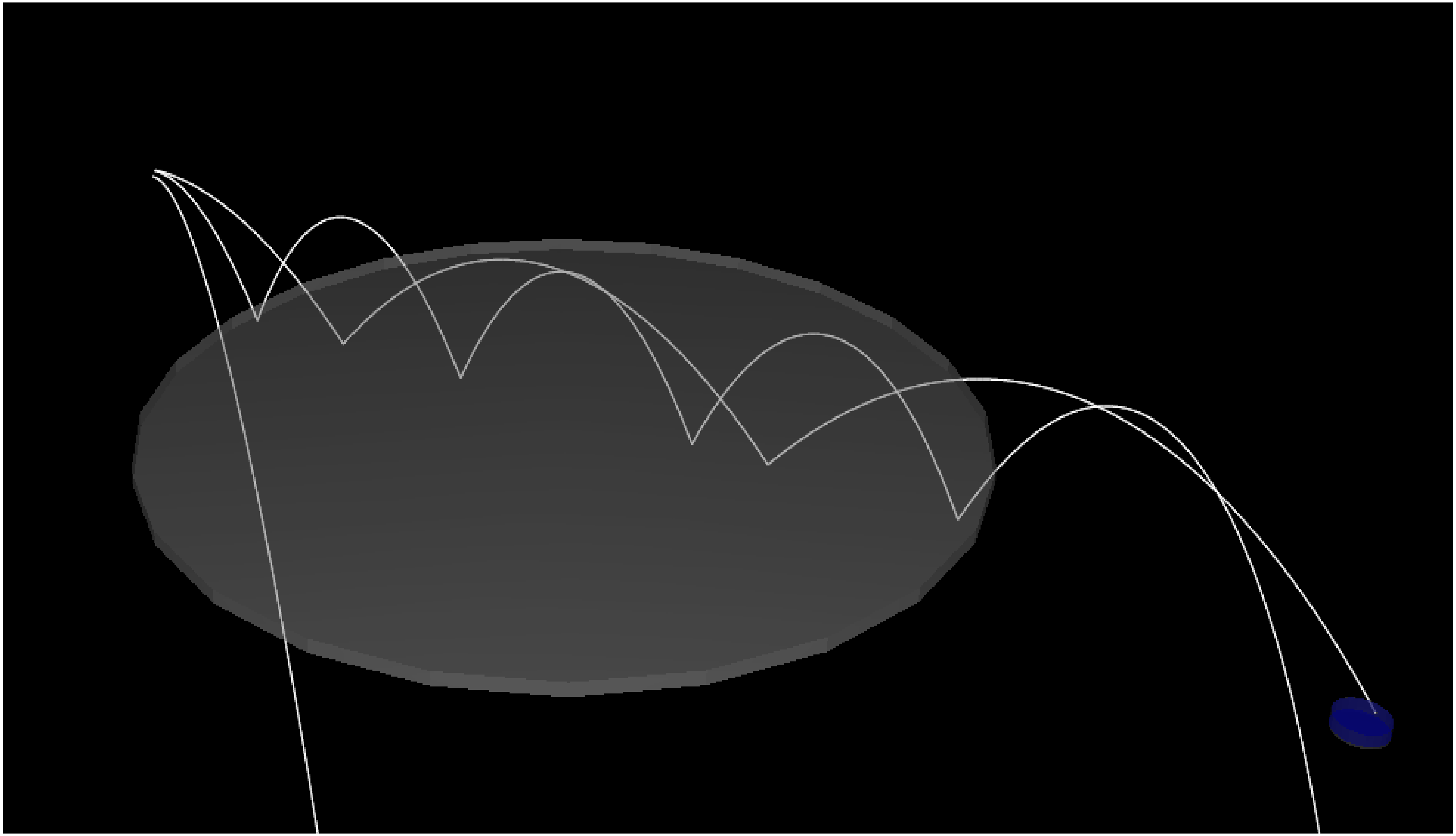
P.A. Seeger, L.L. Daemen, *Numerical solution of Bloch's Equation...*, NIM A **457** (2001), 338

(adaptive step size RK4 algorithm)

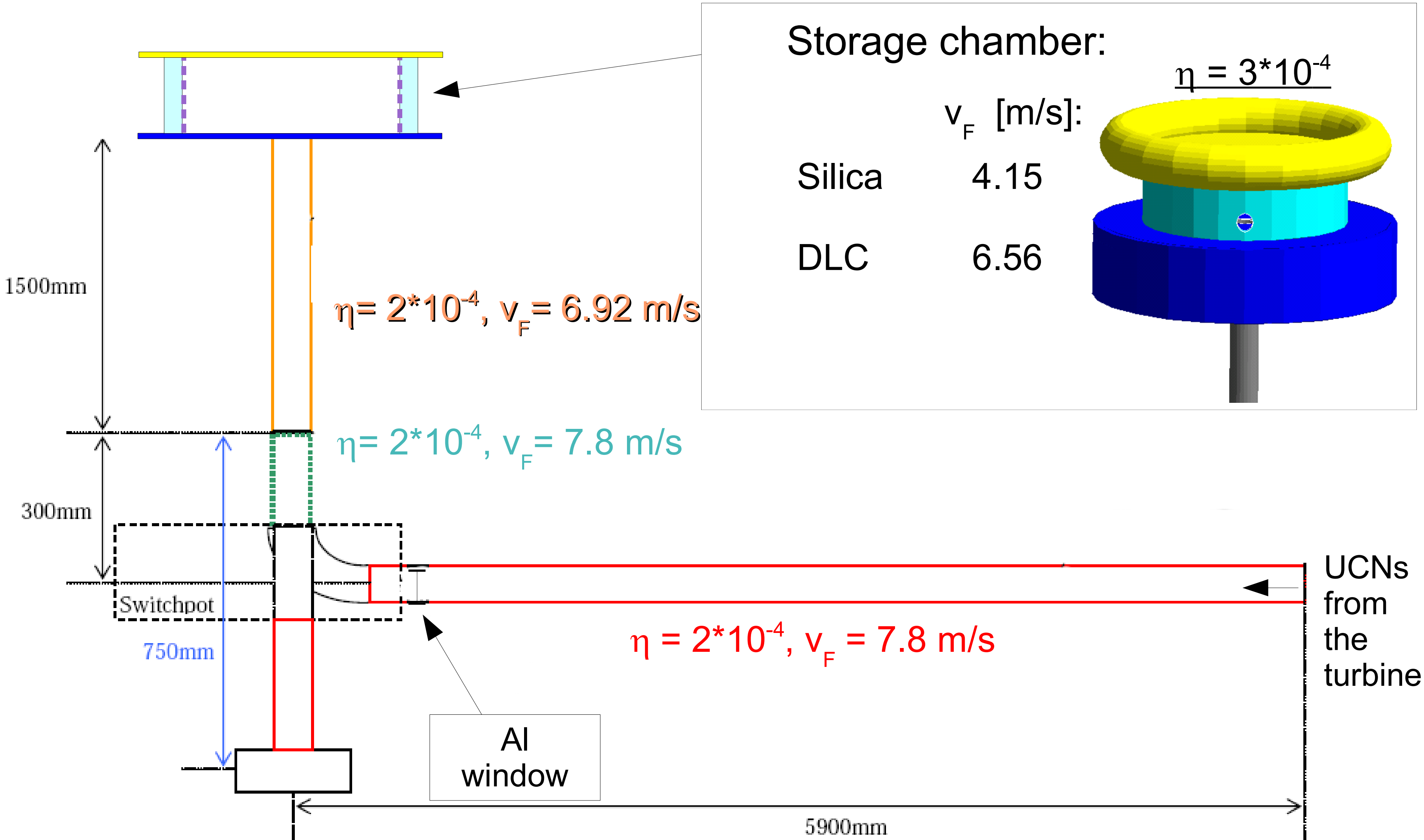


Examples

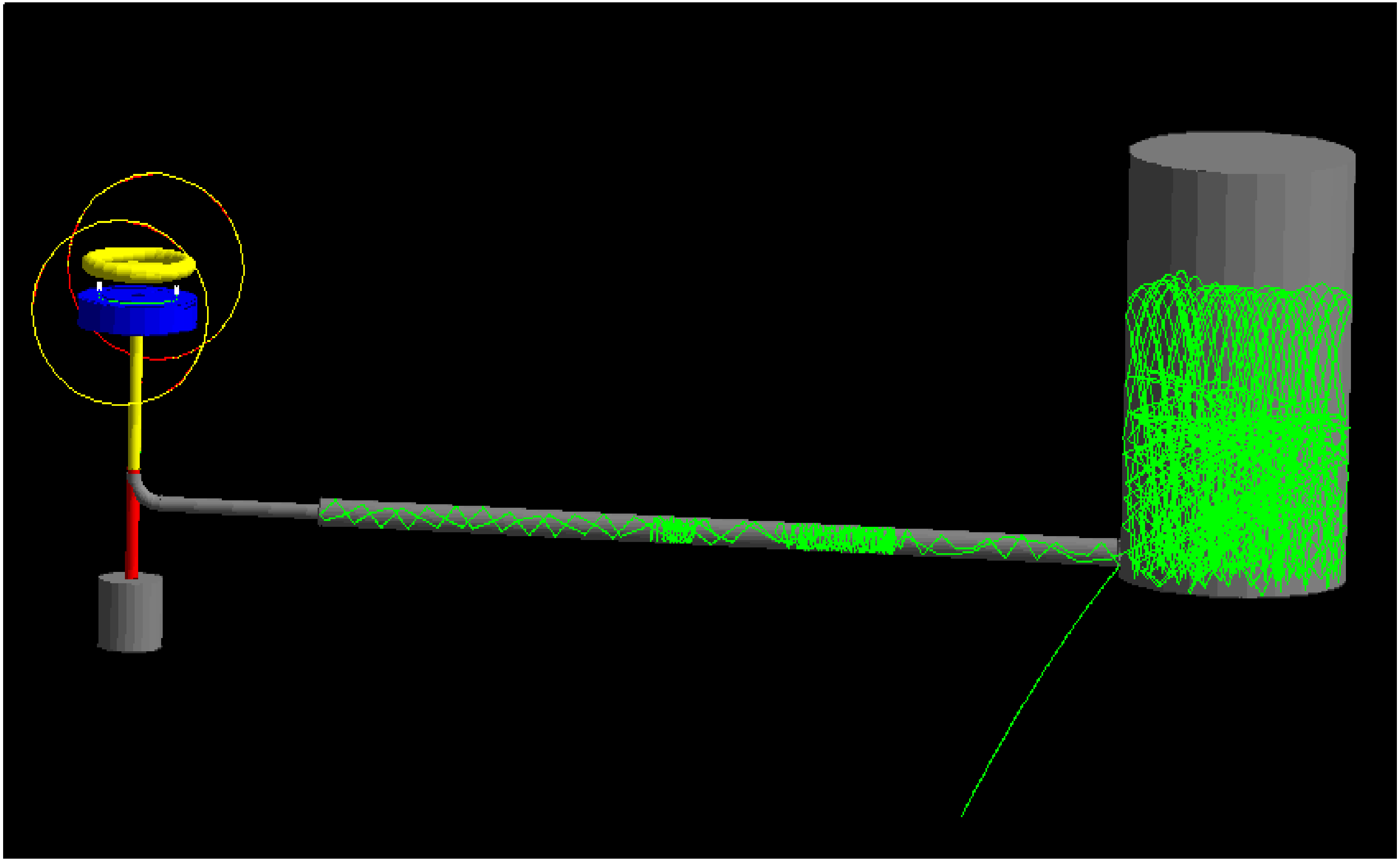
- Sample programs available with Geant4UCN library



nEDM simulation

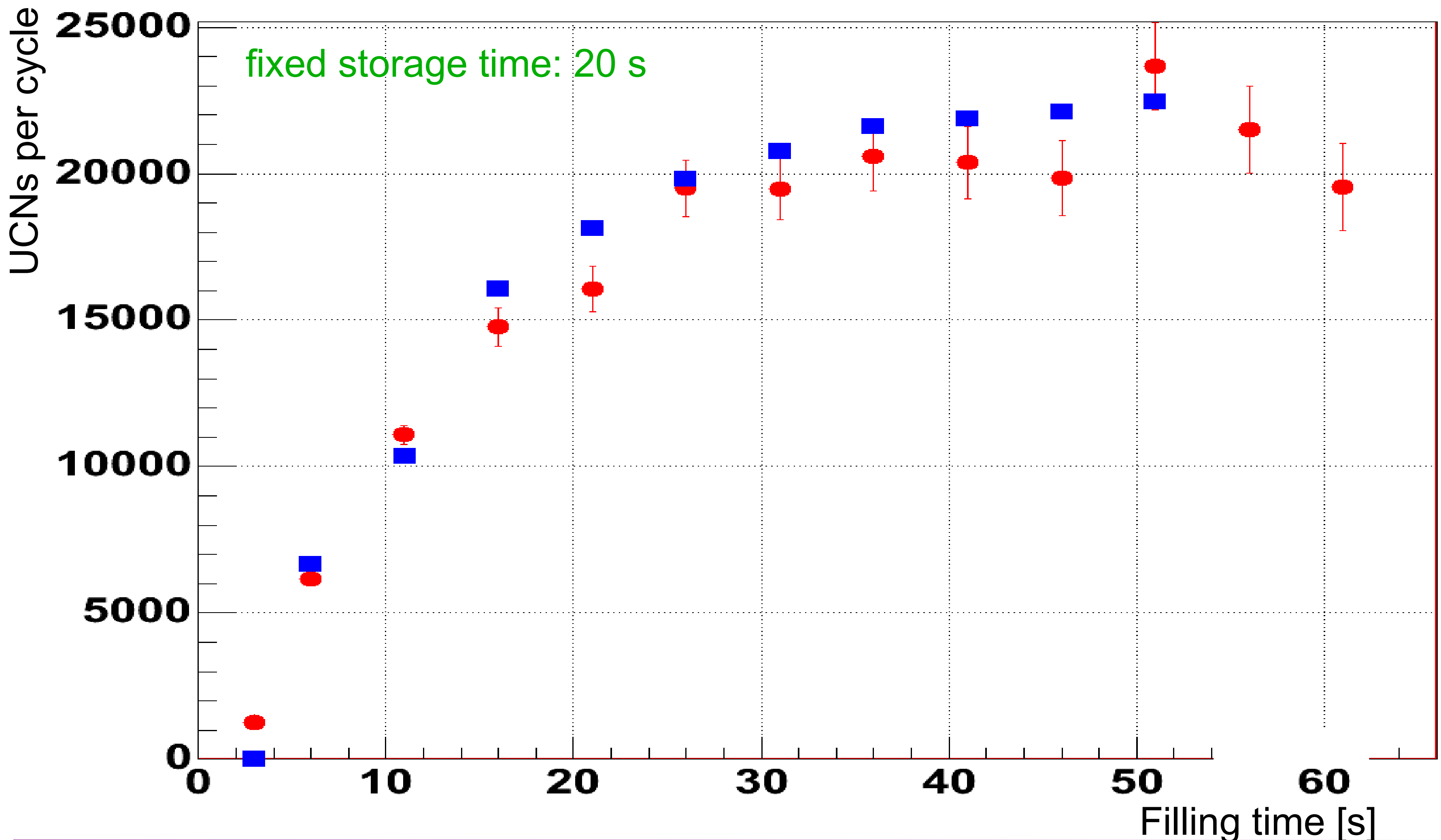


nEDM simulation



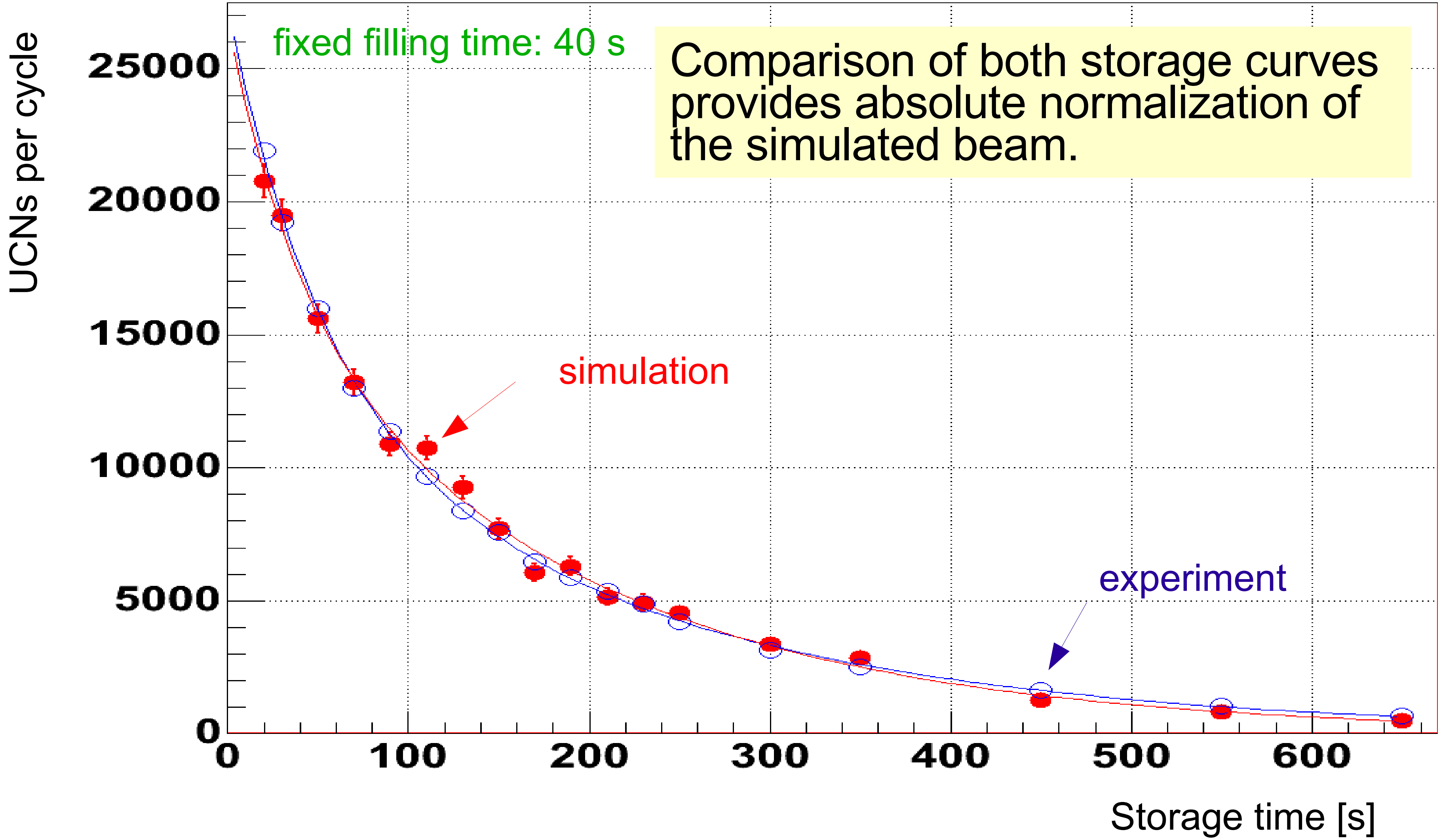
Filling

Experimental data from ILL



Storage

Experimental data from ILL



Status

- Presently executed on merlin cluster at PSI
- Frequently used in UCN project:
  - Understand existing experiments
  - Design of new experiments (nEDM, n lifetime, UCN source)
  - UCN and VCN moderation in solid D_2
- Growing number of users at PSI, ILL and in USA
- CVS access to the library
- http://ucn.web.psi.ch

Input

- Material constants compiled into the code:

```
// --- Fused Silica

G4double SI_POT[NUM] =              {90.0,          90.0};          // The nEDM Progress Report (March 2003), page 6
G4double SI_SPINFLIP[NUM] =         {0.0,           0.0};           // rel. per wall collision
G4double SI_ETA[NUM] =              {3.0e-4,        3.0e-4};        // loss coefficient W/V
G4double SI_DIFFUS[NUM] =           {0.10,          0.10};          // diffuse scattering probability
G4double SI_REFLECTIVITY[NUM] =     {1.0,           1.0};           // reflectivity, not used parameter
G4double SI_ABSCS[NUM] =            {0.08,          0.08};          // weighted average (Si, O)
G4double SI_LOSSCS[NUM] =           {0.0,           0.0};           // loss cross section at room temperature for Be
G4double SI_SCATCS[NUM] =           {3.27,          3.27};          // weighted average (Si, O)
G4MaterialPropertiesTable *Tsilica = new G4MaterialPropertiesTable();
Tsilica->AddProperty("REFLECTIVITY", PP, SI_REFLECTIVITY,      NUM);
Tsilica->AddProperty("DIFFUSION",    PP, SI_DIFFUS,            NUM);
Tsilica->AddProperty("FERMIPOT",     PP, SI_POT,               NUM);
Tsilica->AddProperty("SPINFLIP",     PP, SI_SPINFLIP,          NUM);
Tsilica->AddProperty("LOSS",         PP, SI_ETA              , NUM);
Tsilica->AddProperty("LOSSCS",       PP, SI_LOSSCS           , NUM);
Tsilica->AddProperty("ABSCS",        PP, SI_ABSCS            , NUM);
Tsilica->AddProperty("SCATCS",       PP, SI_SCATCS           , NUM);
```

- Shutters and field ramping controlled from a macro file:

```
# set shutter state (open/close) number(1,2,3,..) time (s)
/shutter/use 1
/shutter/verbose 0
# ---control shutter number 2
/shutter/open 2 0
/shutter/close 2 40
/shutter/close 2 60
/shutter/close 2 230
```

```
# when does our field start in seconds
/fieldcube/starttime 3

# if there is a time dependent field
/fieldcube/timedependence 0
# if there is a timedependence, we need a file for this
/fieldcube/timefile time.dat
# visualization of the fieldpoints
/fieldcube/drawfield 0
```