CORTEX

Core monitoring techniques and experimental validation and demonstration

Noise induced by FA vibrations in time-domain diffusion codes: PARCS and FEMFFUSION

WP1/WP2 Validation Workshop
Toni Vidal-Ferràndiz
anvifer2@upv.es

This project has received funding from the Euratom research and training programme 2014-2018 under grant agreement No 754316.

UPV Efforts

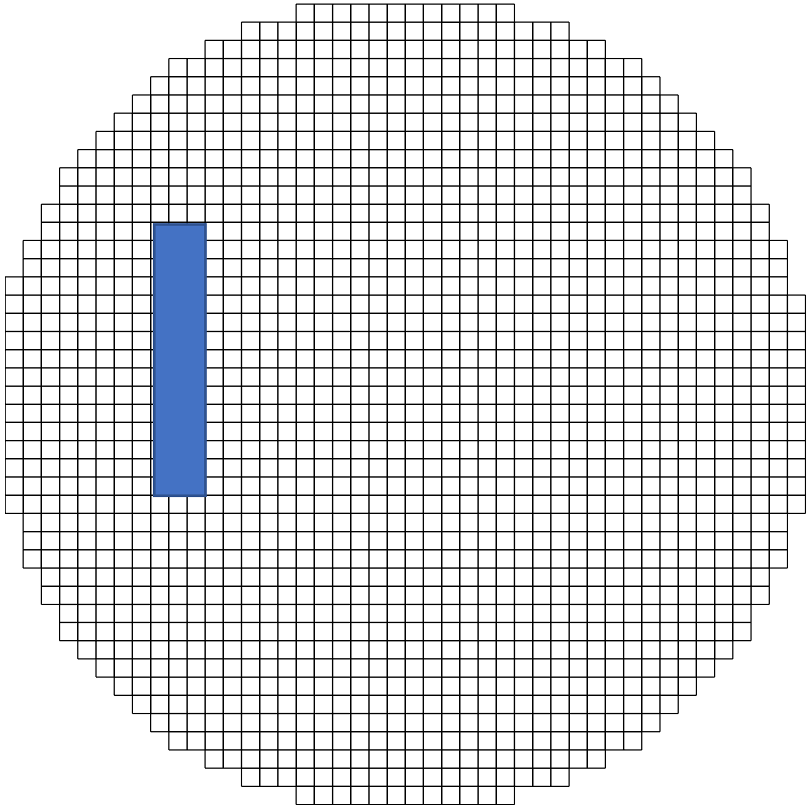
- Time-dependent computations.
- Diffusion approximation.
- Two group approximation
- Direct representation of the FA oscillation.
- Fixed grid with volume-homogenized materials.
  - Local refined meshes gives (essentially) the same results.
- Postprocess through FFT and CPSD.

CROCUS model

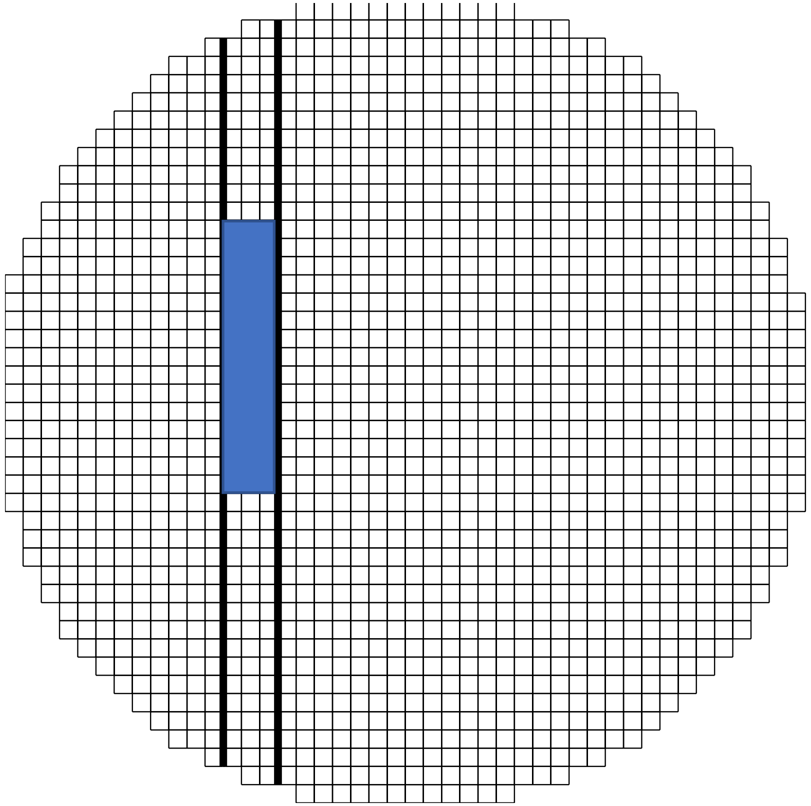
- A 2D model, based on EPFL PARCS model has been produced.
- Two time-dependent codes has been used.
  - **PARCS**: Using a finite difference method.
  - **FEMFFUSION**: Using a finite element method.
- Two meshes have been used:
  - A uniform mesh: 44x44 cells.
  - A local refined around the vibrating assembly. 96x44 cells.
- The computation have been done with high accuracy for the spatial and time discretization.
  - 5^{th} degree polynomials in the FEM.
  - 8x8 inner grid per cell in PARCS.

CROCUS model

- Meshes

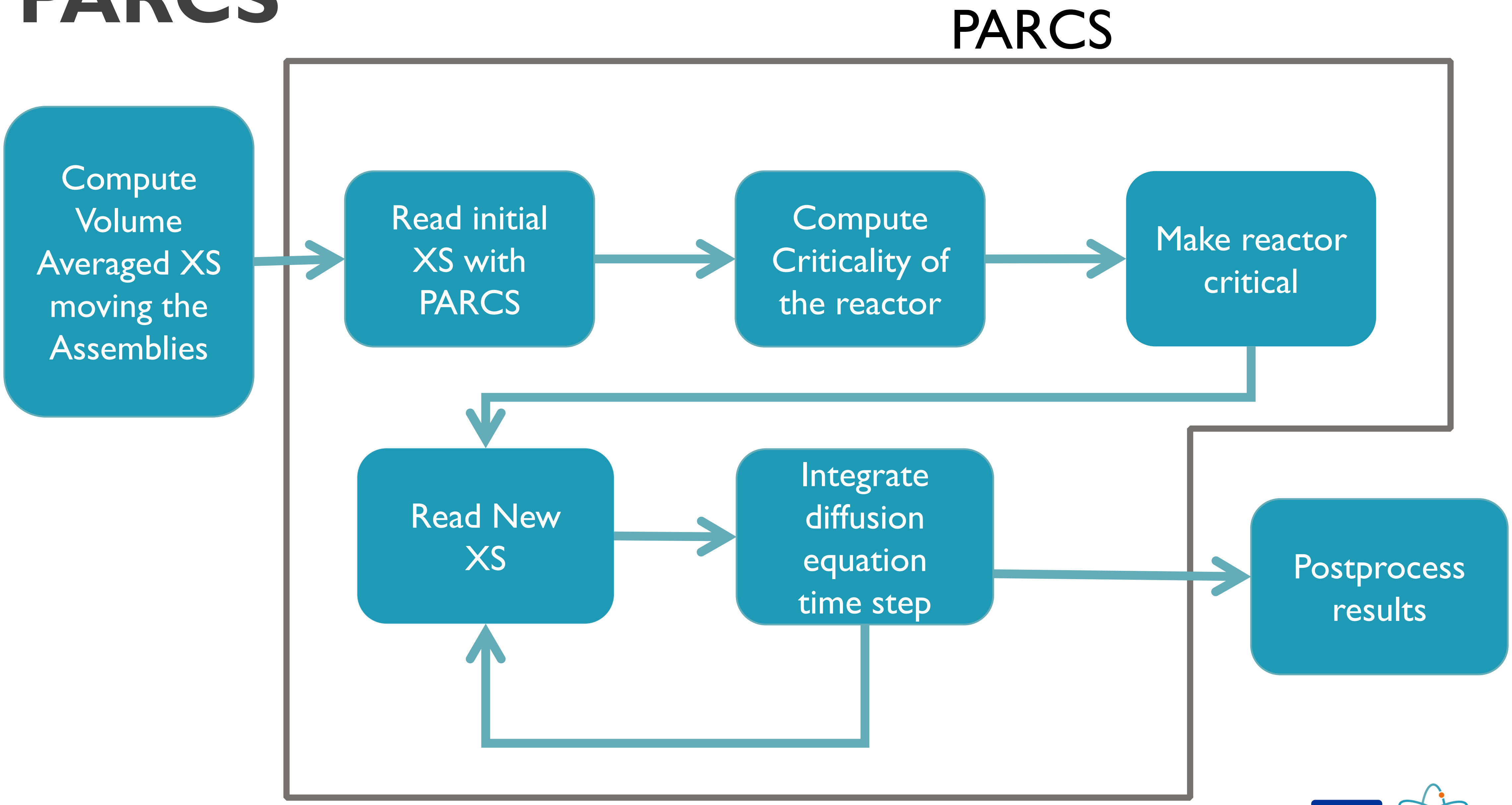


(a) Uniform Uniform

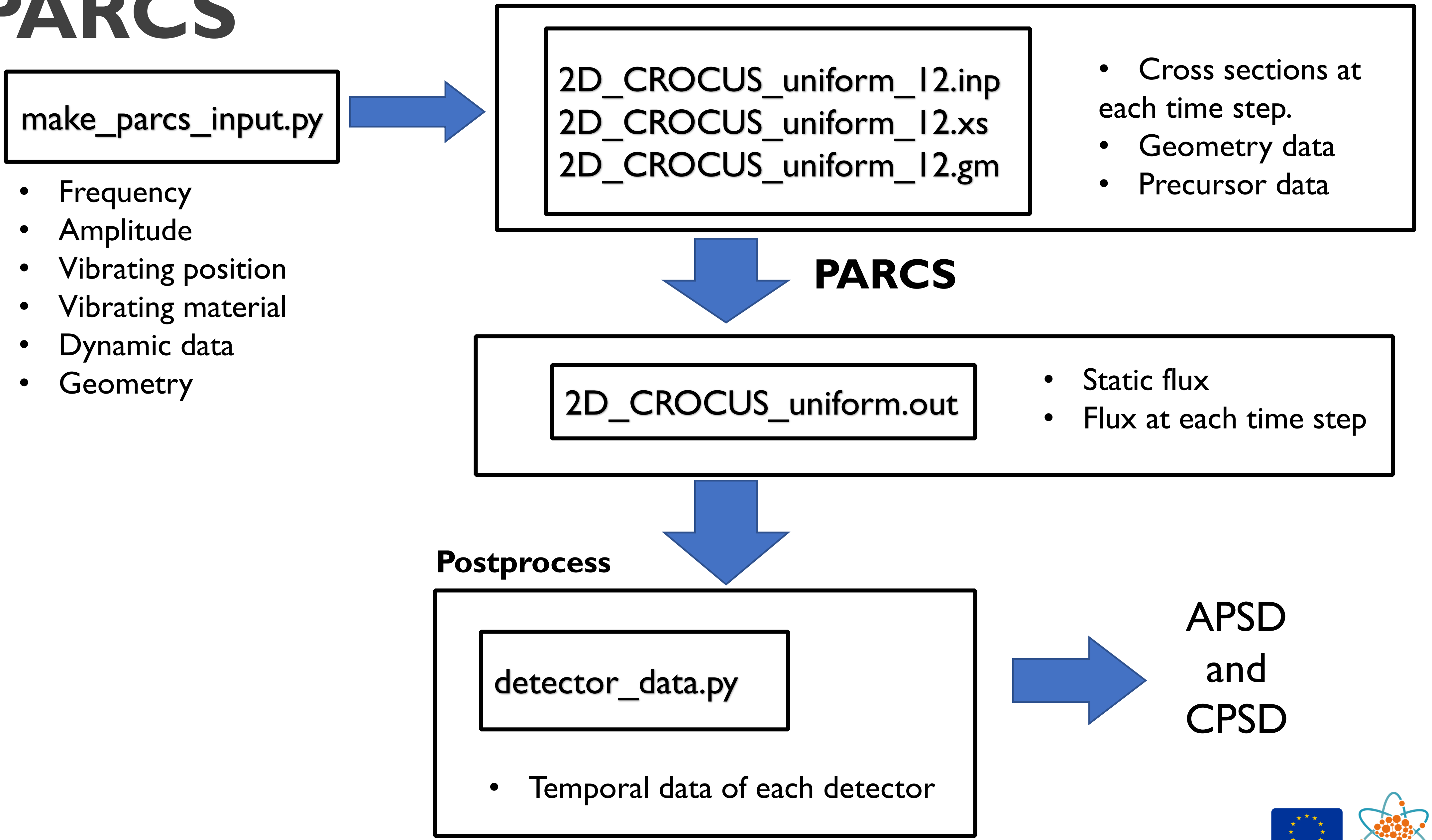


(b) Local Mesh

PARCS


PARCS
Compute Volume Averaged XS moving the Assemblies
Read initial XS with PARCS
Compute Criticality of the reactor
Make reactor critical
Read New XS
Integrate diffusion equation time step
Postprocess results

PARCS

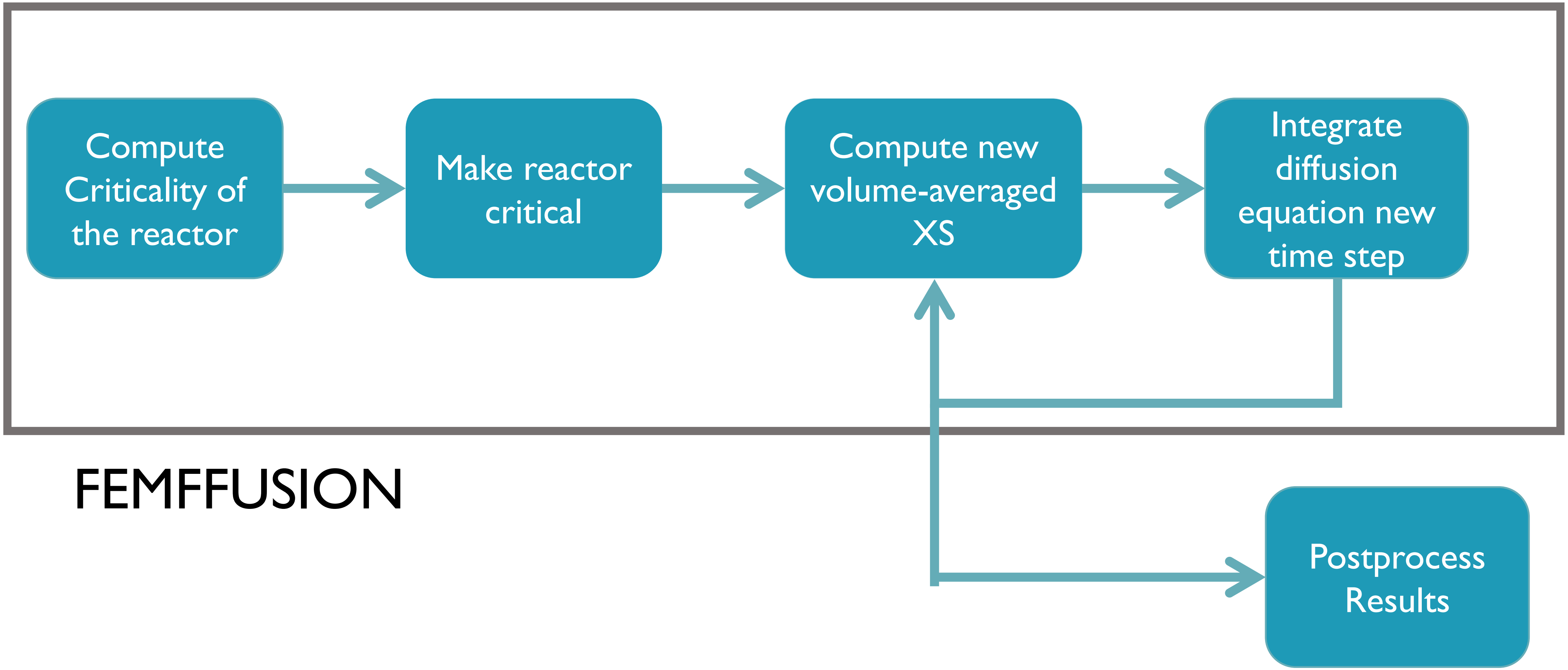

make_parcs_input.py
• Frequency
• Amplitude
• Vibrating position
• Vibrating material
• Dynamic data
• Geometry
2D_CROCUS_uniform_12.inp
2D_CROCUS_uniform_12.xs
2D_CROCUS_uniform_12.gm
• Cross sections at each time step.
• Geometry data
• Precursor data
PARCS
2D_CROCUS_uniform.out
• Static flux
• Flux at each time step
Postprocess
detector_data.py
• Temporal data of each detector
APSD
and
CPSD

FEMFFUSION

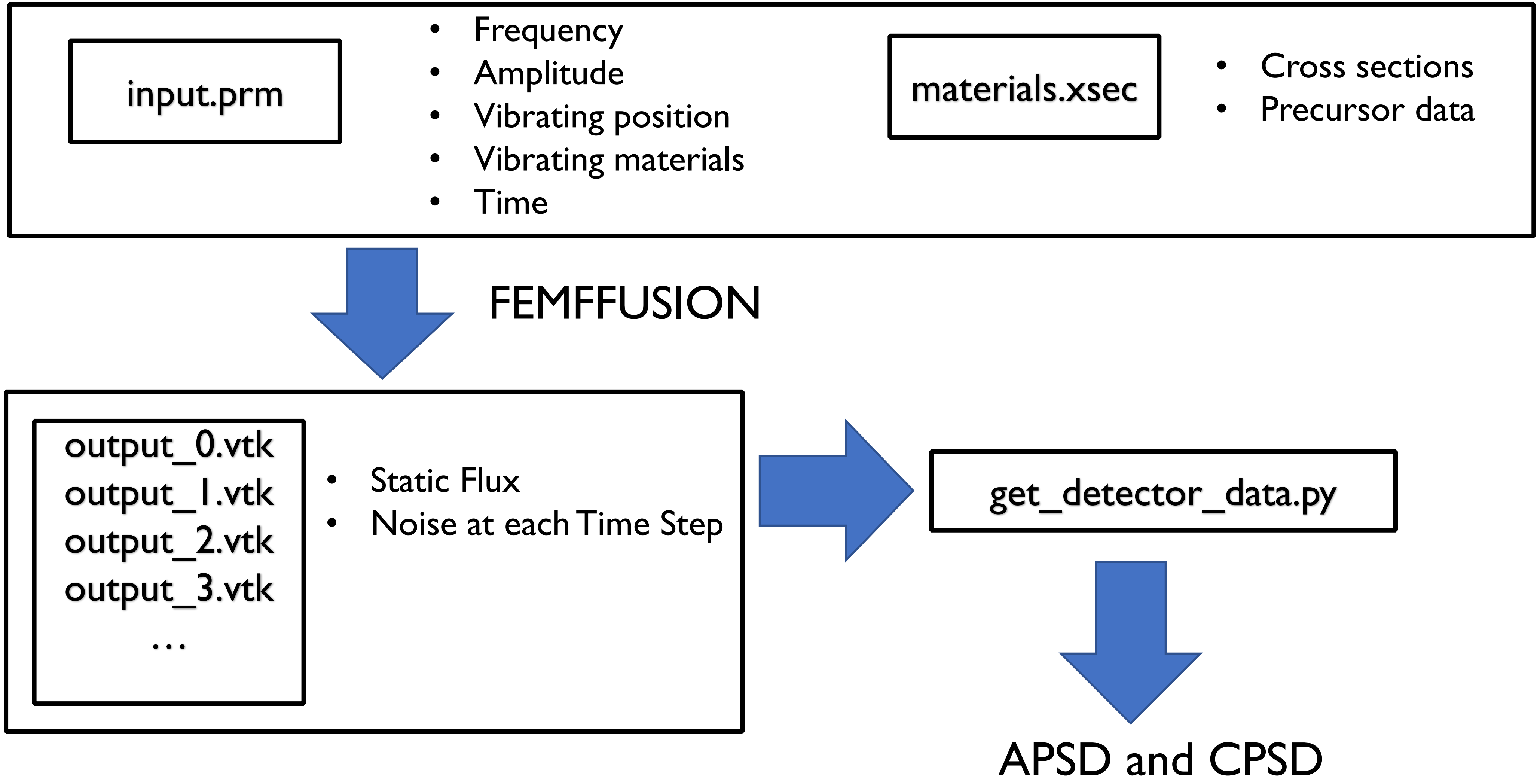
- Open repository at:

  https://bitbucket.org/Zonni/femffusion_vibration

FEMFFUSION


Compute Criticality of the reactor
Make reactor critical
Compute new volume-averaged XS
Integrate diffusion equation new time step
FEMFFUSION
Postprocess Results

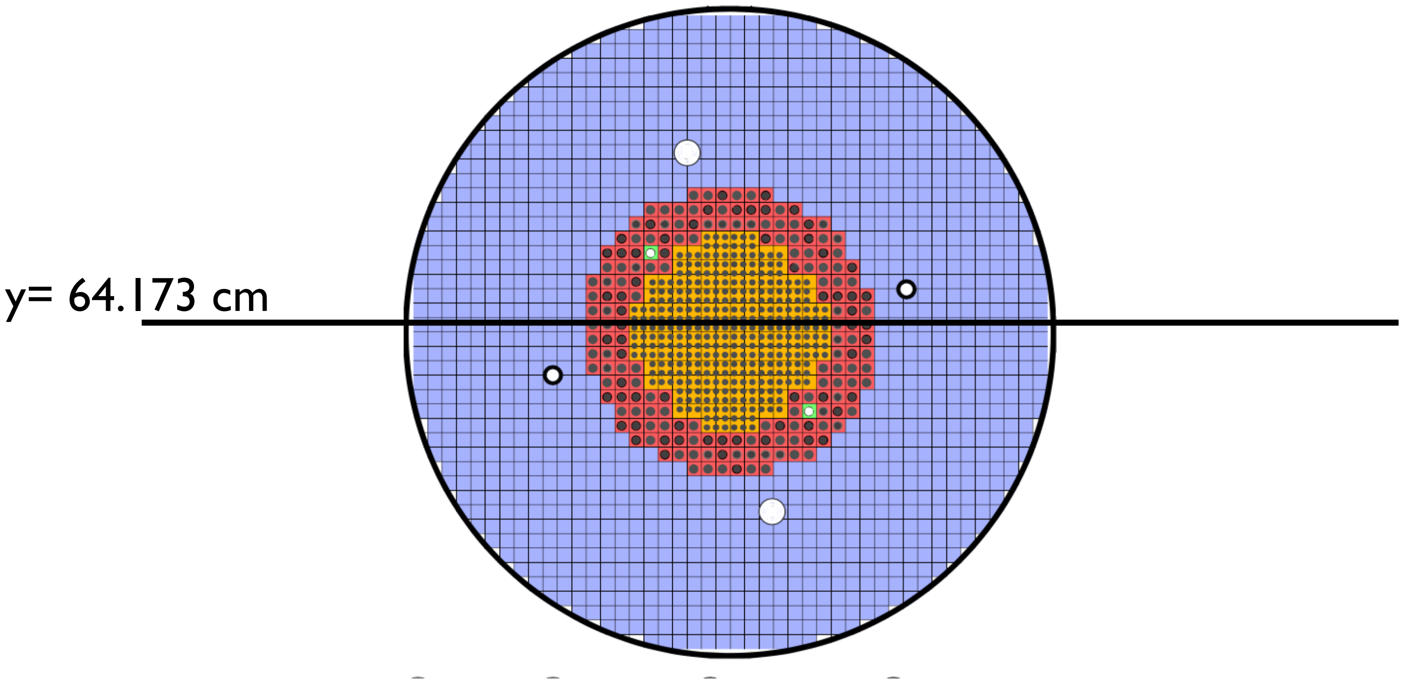
FEMFFUSION


input.prm
Frequency
Amplitude
Vibrating position
Vibrating materials
Time
materials.xsec
Cross sections
Precursor data
FEMFFUSION
output_0.vtk
output_1.vtk
output_2.vtk
output_3.vtk
…
Static Flux
Noise at each Time Step
get_detector_data.py
APSD and CPSD

Numerical Examples

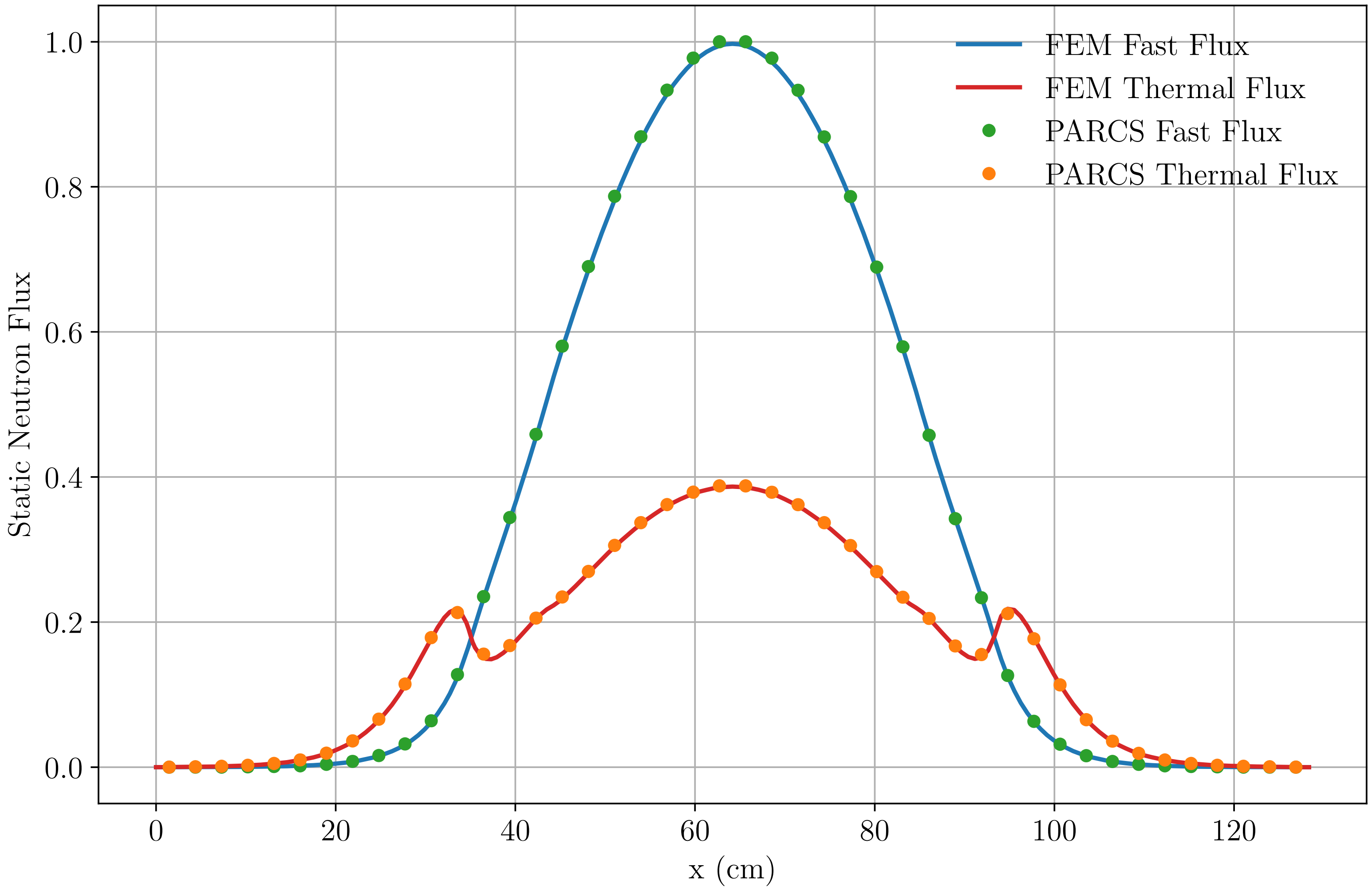
- Experiment 12 of 1st campaign, 2 mm amplitude, 0.1Hz.
- Most of the results at centre line y= 64.173 cm
- The updated data can be found at Chalmers ftp:

  upv/WP2_Validation_Workshop_UPV



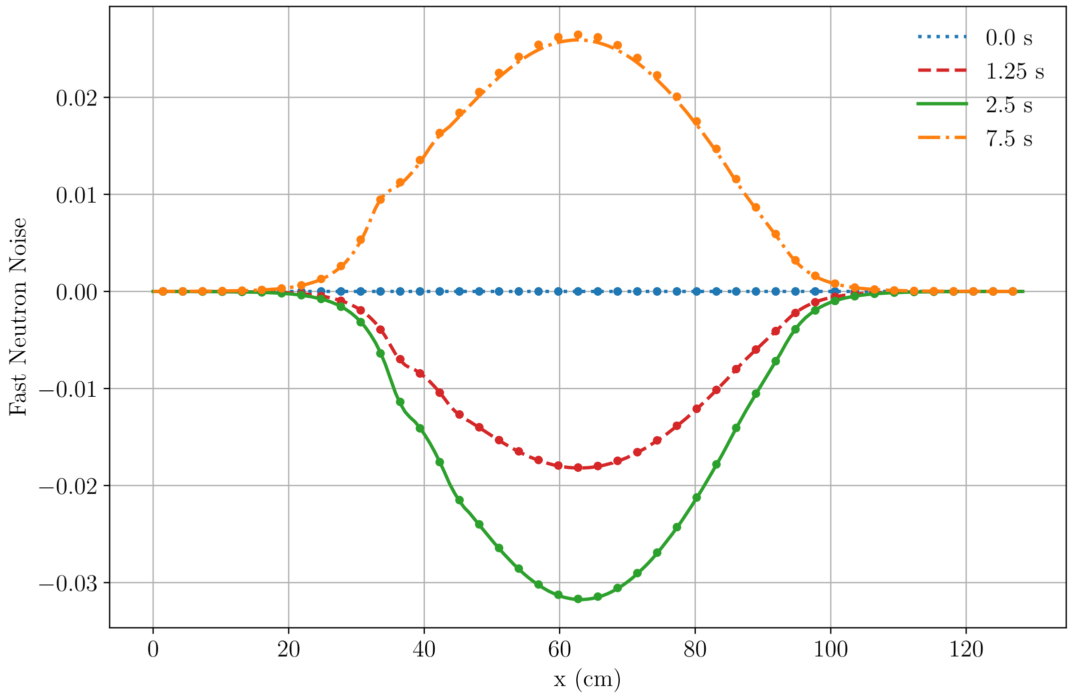
Static Flux Results

- Results at the center line y= 64.173 cm

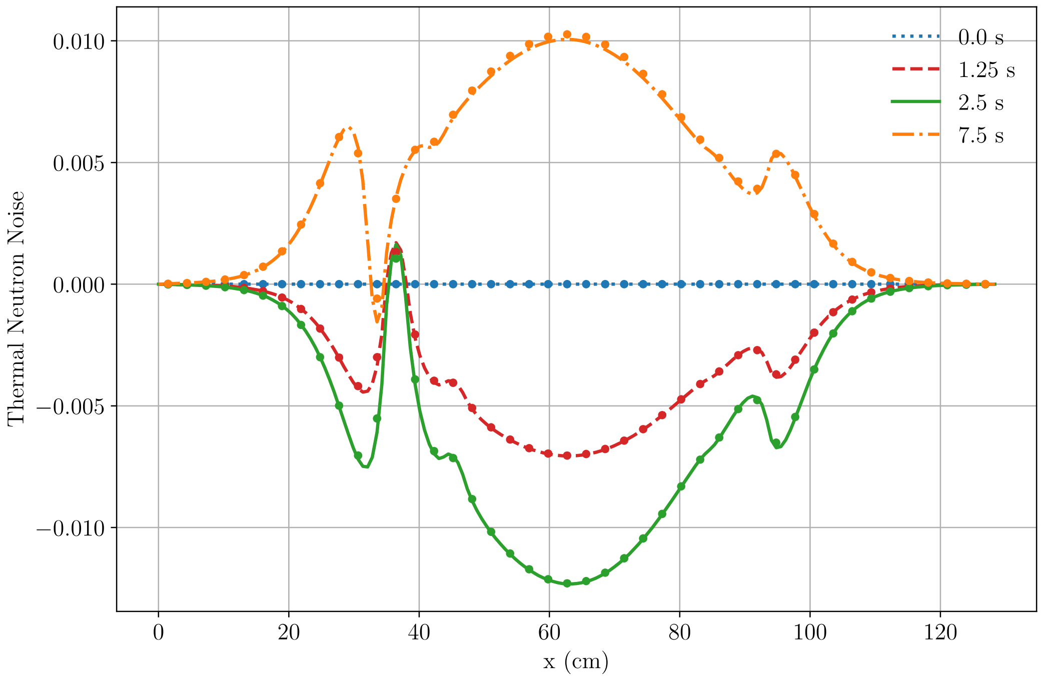


Static Flux Results

- Results at the center line y= 64.173 cm
- Noise: $\delta\phi(x,t) = \phi(x,t) - \phi(x,0)$
- **Uniform refinement.**
- Done with noise_time_results.py



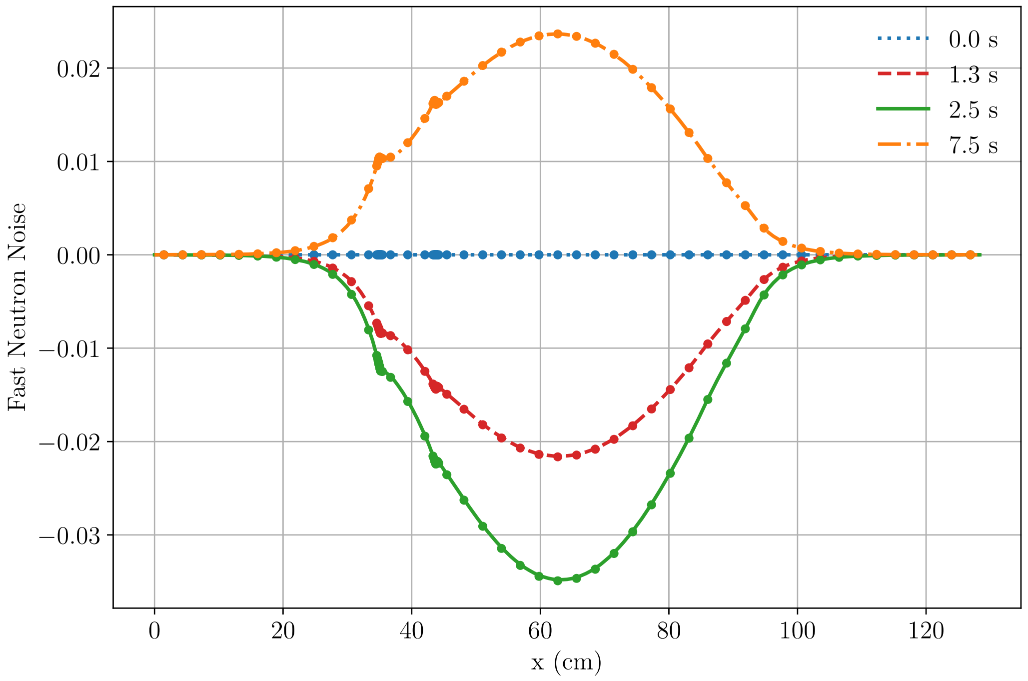
(a) Fast Flux Noise



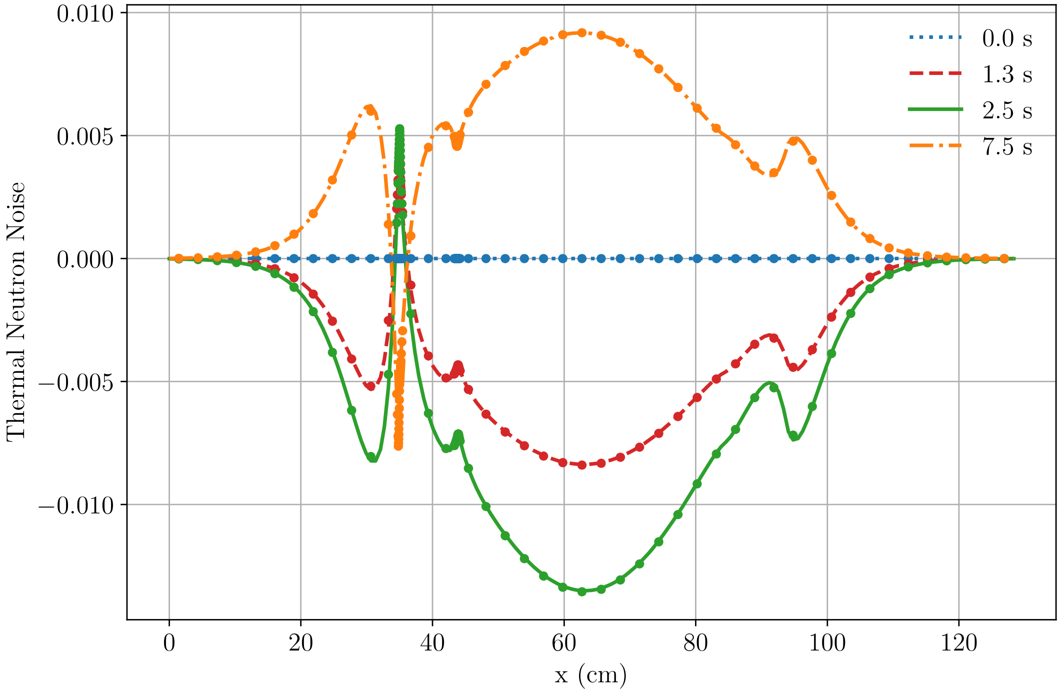
(b) Thermal Flux Noise

Time dependent noise

- Results at the center line y= 64.173 cm
- **Local refinement.**
- Done with noise_time_results.py



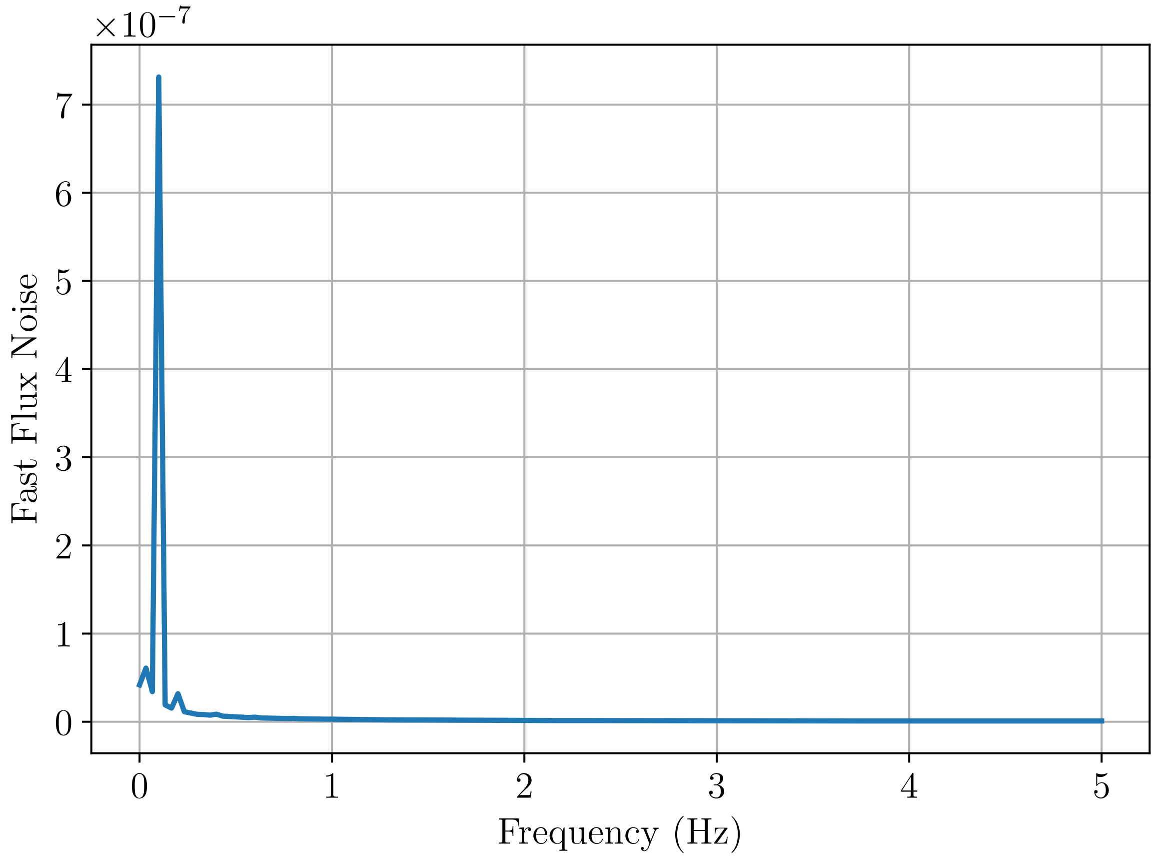
(a) Fast Flux Noise



(b) Thermal Flux Noise

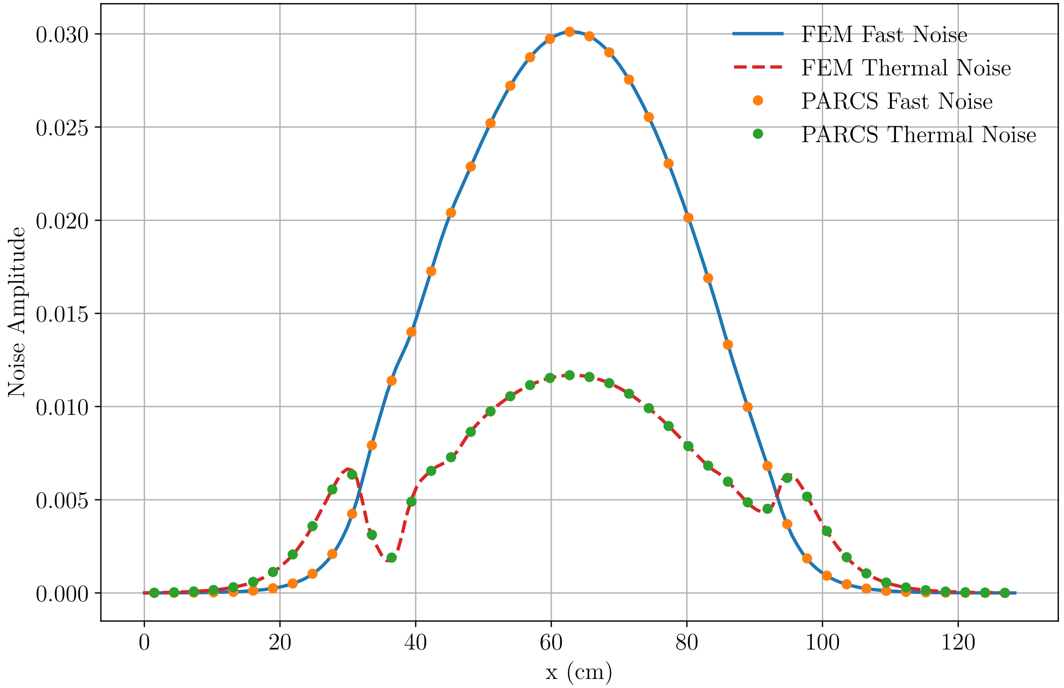
Fourier Transform

- If we make the Fourier, we can see that the induced noise is essentially monochromatic.
- Done with fft_parcs_lines.py

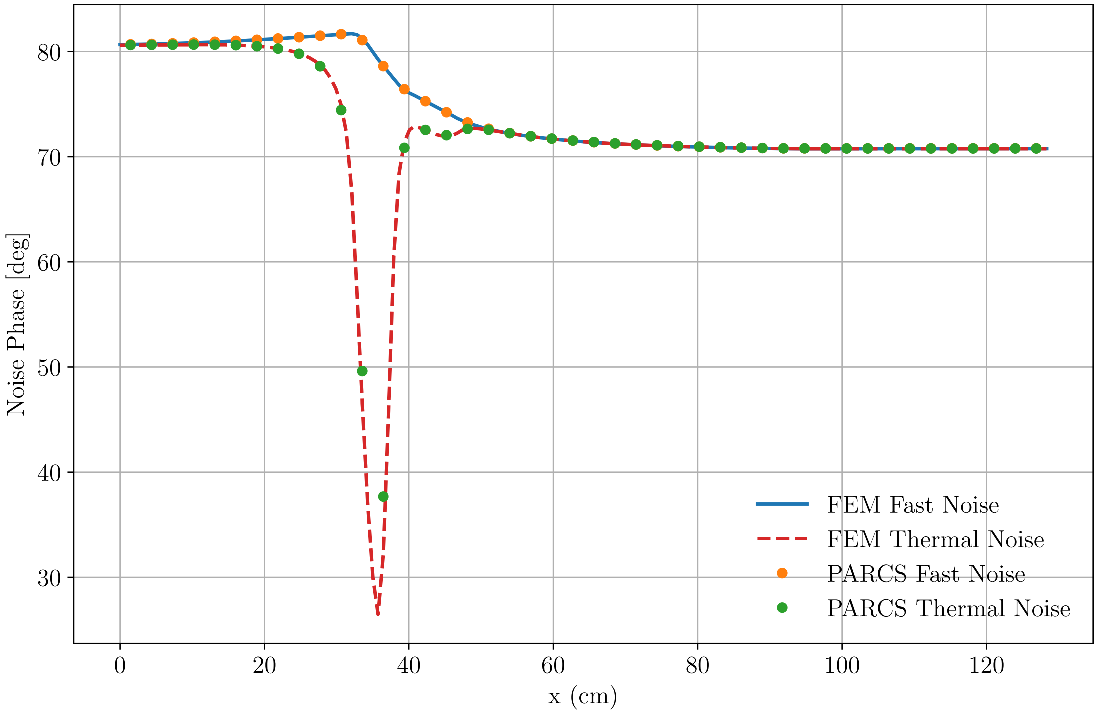


Fourier Transform at 0.1 Hz

- Results at the center line y= 64.173 cm
- **Uniform refinement.**
- Done with noise_time_results.py



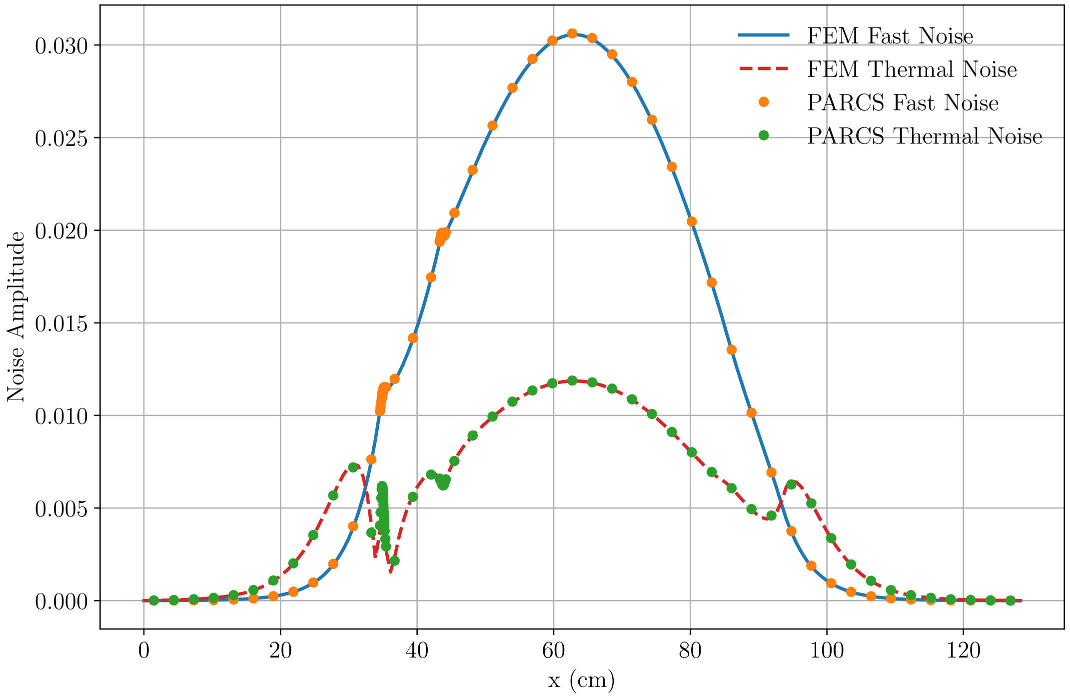
(a) Noise magnitude



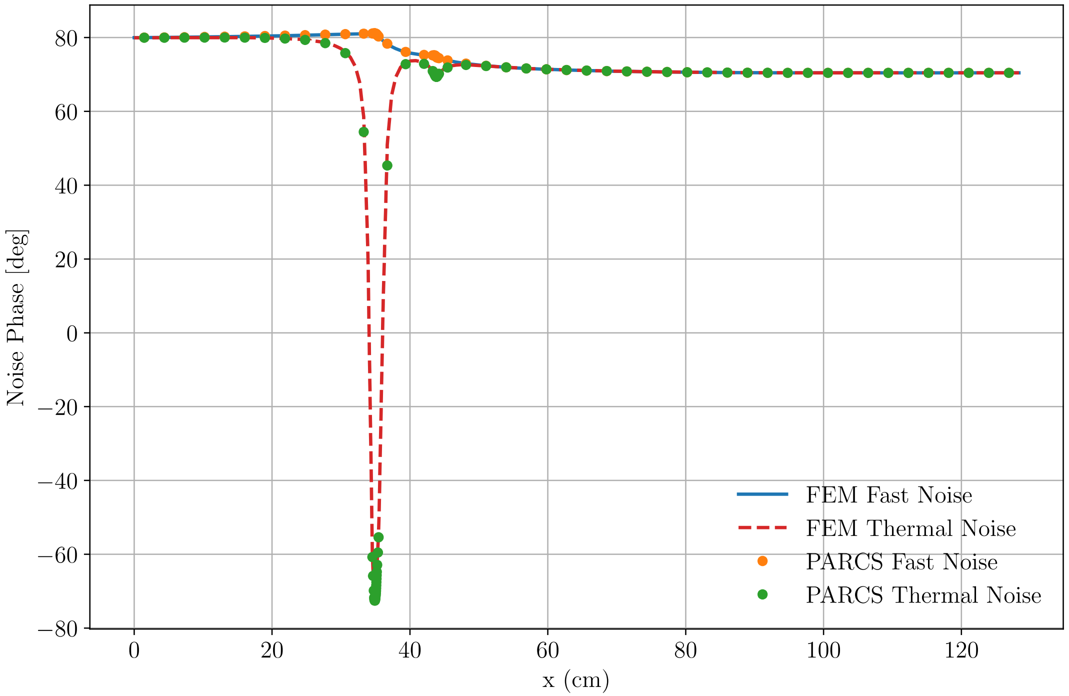
(b) Noise Phase

Fourier Transform at 0.1 Hz

- Results at the center line y= 64.173 cm
- **Local refinement.**
- Done with noise_time_results.py



(a) Noise magnitude

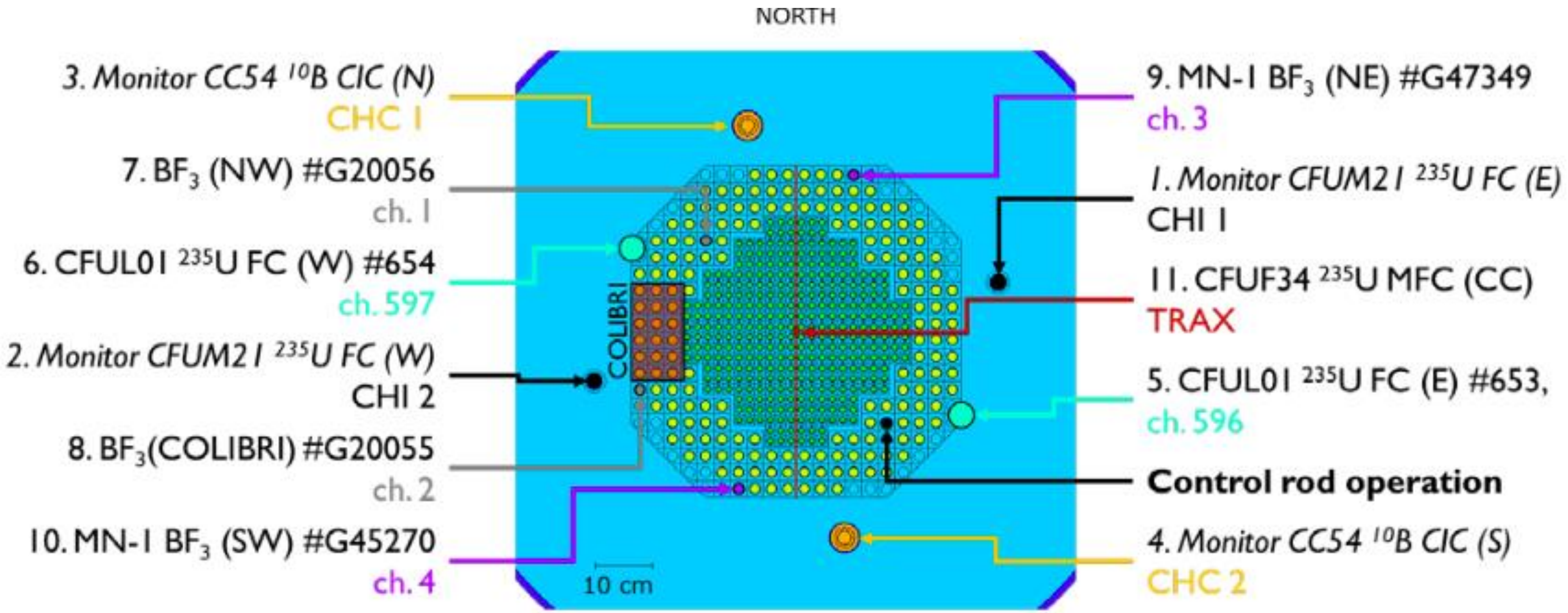


(b) Noise Phase

Detector Data

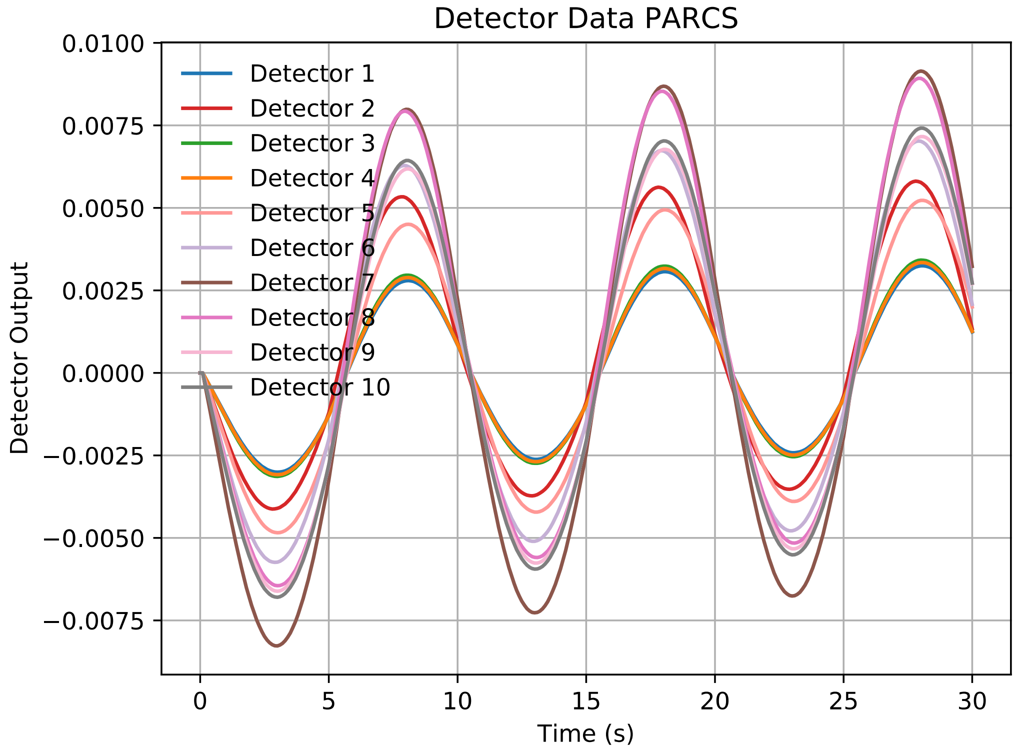
- Now, we can extract the detectors data from the models and compute the CPSD between detectors.

$$Detector\ Output = \phi_2(\vec{r}_{dtc})$$

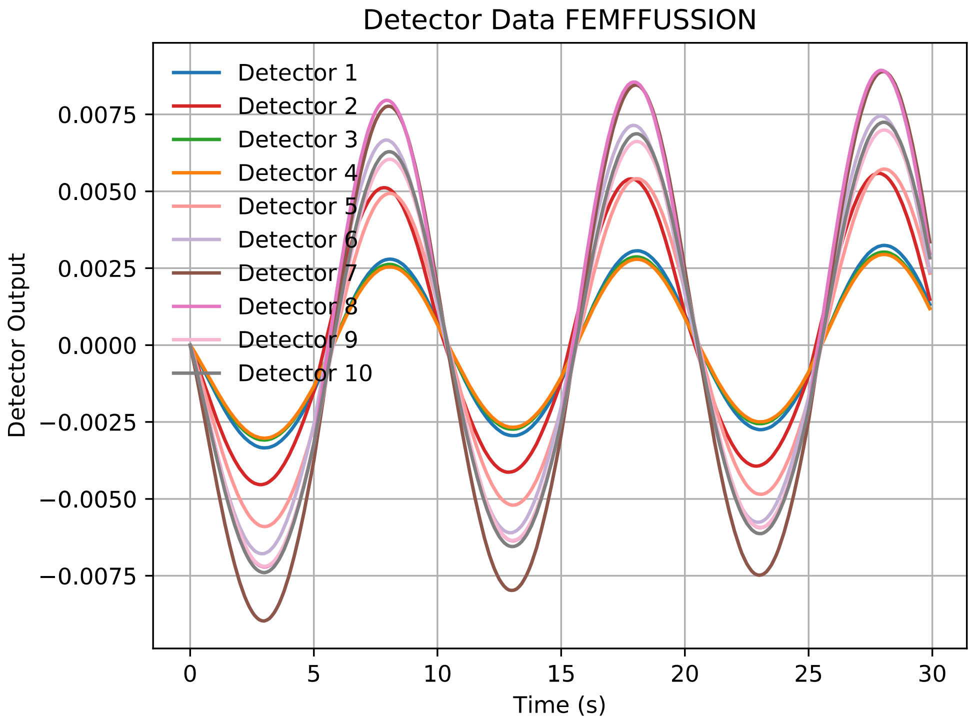


Detector Data

- Results at the center line y=
- **Uniform refinement.**
- Done with get_detector_data.py



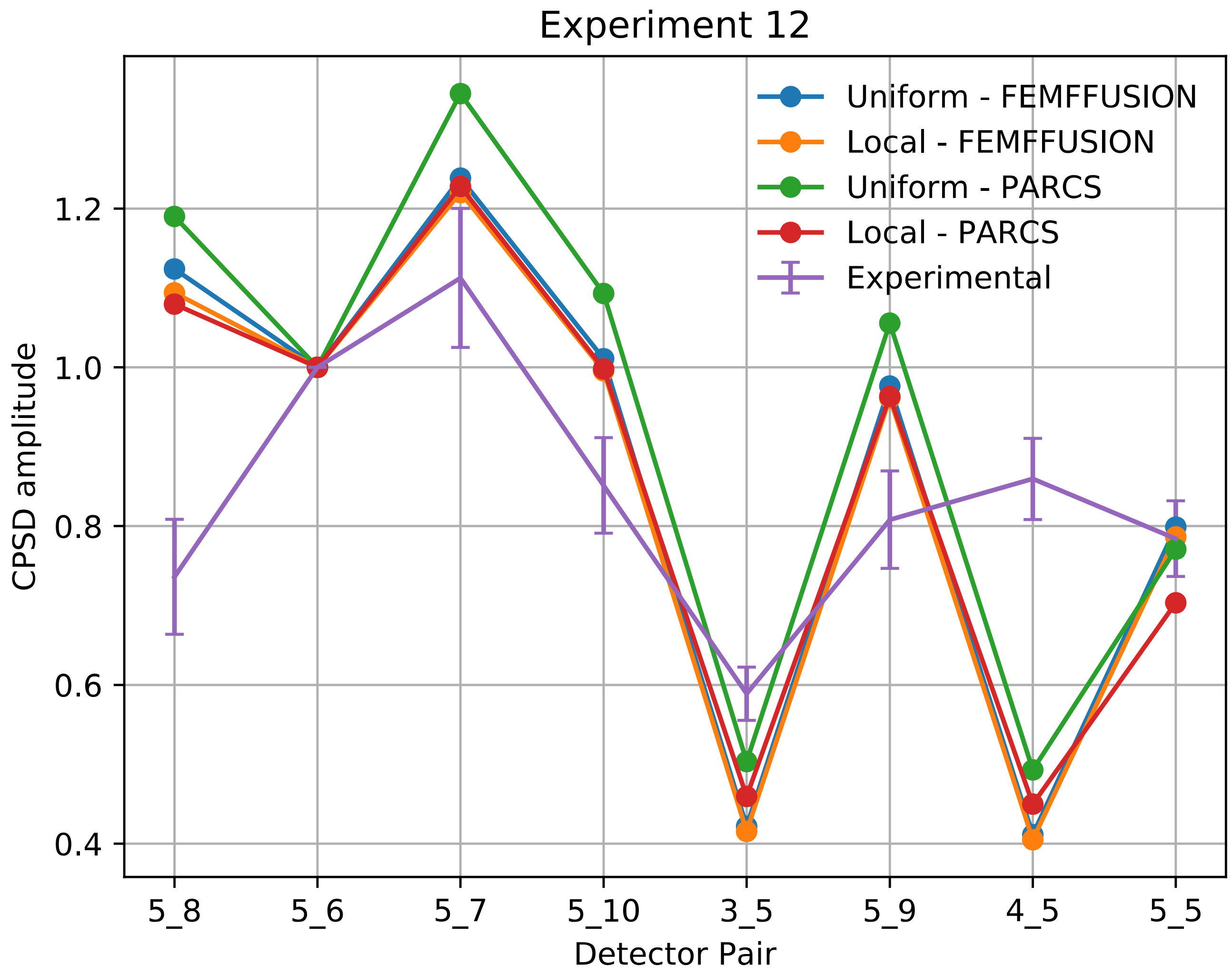
(a) PARCS



(b) FEMFFUSION

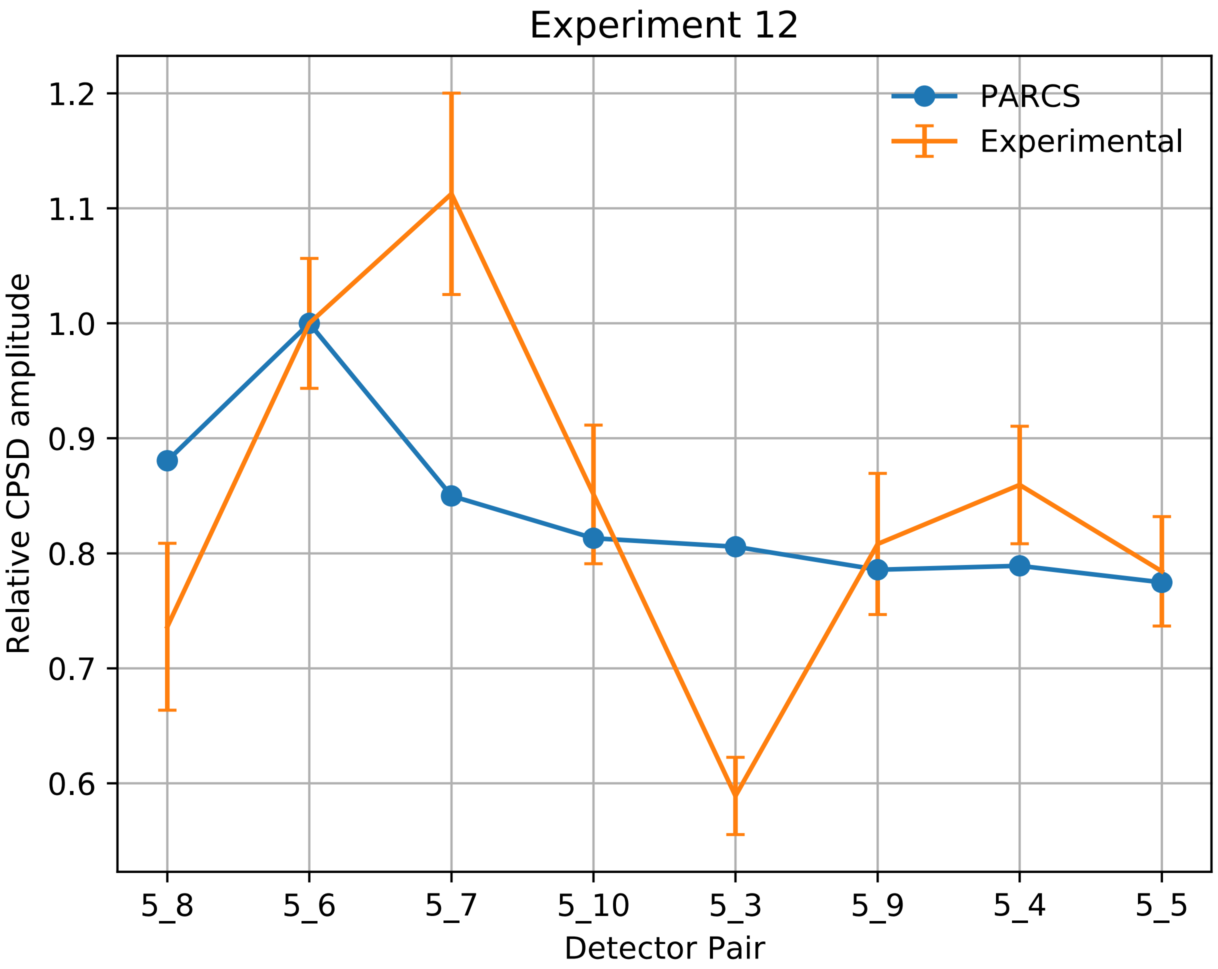
CPSD of Experiment 12

- CPSD amplitude at f=0.1 Hz, normalized at CPSD_56



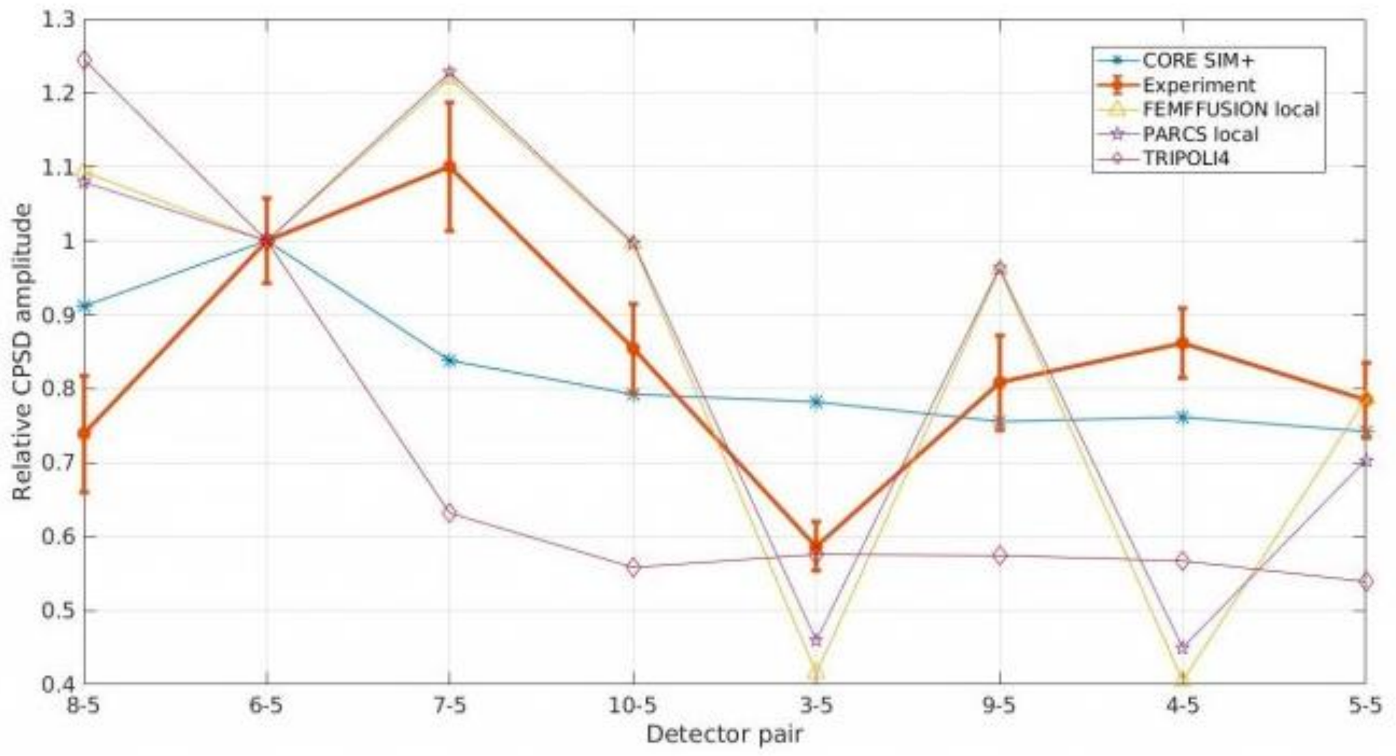
New results of Experiment 12

- Now with relative noise!

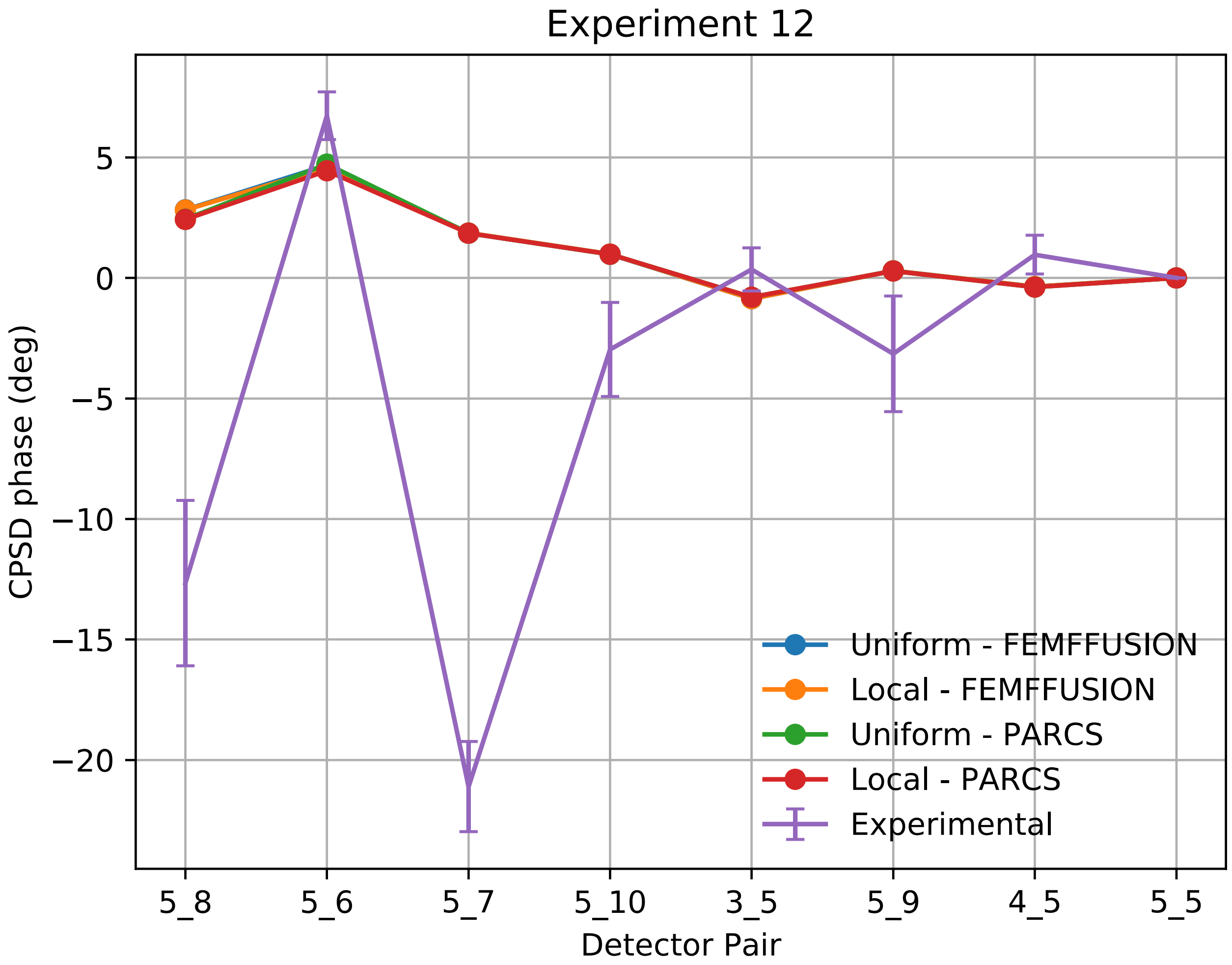


CORE SIM+ Results

!



Phase Comparison



Ideas to converge (between modellers)

- Converge on similar 2D numerical benchmark.
- In COLIBRI and AKR:
  - Use of the same mesh and cross sections.
    - A set for homogenized 2-group diffusion approximation.
    - A set for transport (MC and SN).
  - Unify amplitudes and frequency of the vibrations.
    - Use of the <u>measured</u> amplitudes.
  - Position of the detectors in the computational grid.

  - **Relative noise** or absolute noise.

Thank you