SimEx, a start-to-end simulation platform for experiments at advanced light sources

Juncheng E

PaNOSC WP5 Development sprint

20th Apr., 2020

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Outline

- What is SimEx
- What has been done with SimEx
- What is going on with SimEx
- What is expected in the future

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

What is SimEx?

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

“From source to detector” at advanced light sources.

Photon Source

- *XFEL*
- *Synchrotron*
- *Optical Laser*

Photon propagation

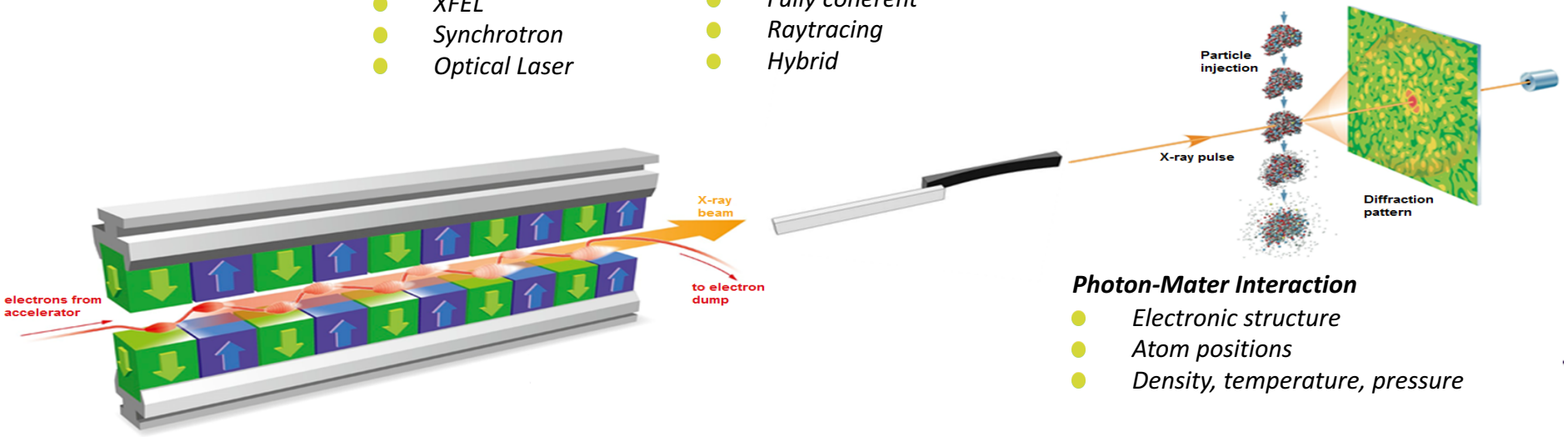
- *Fully coherent*
- *Raytracing*
- *Hybrid*

Signal generation

- *Spectroscopy*
- *Diffraction*
- *Inelastic scattering*

Detector response

- *Pixel detectors*



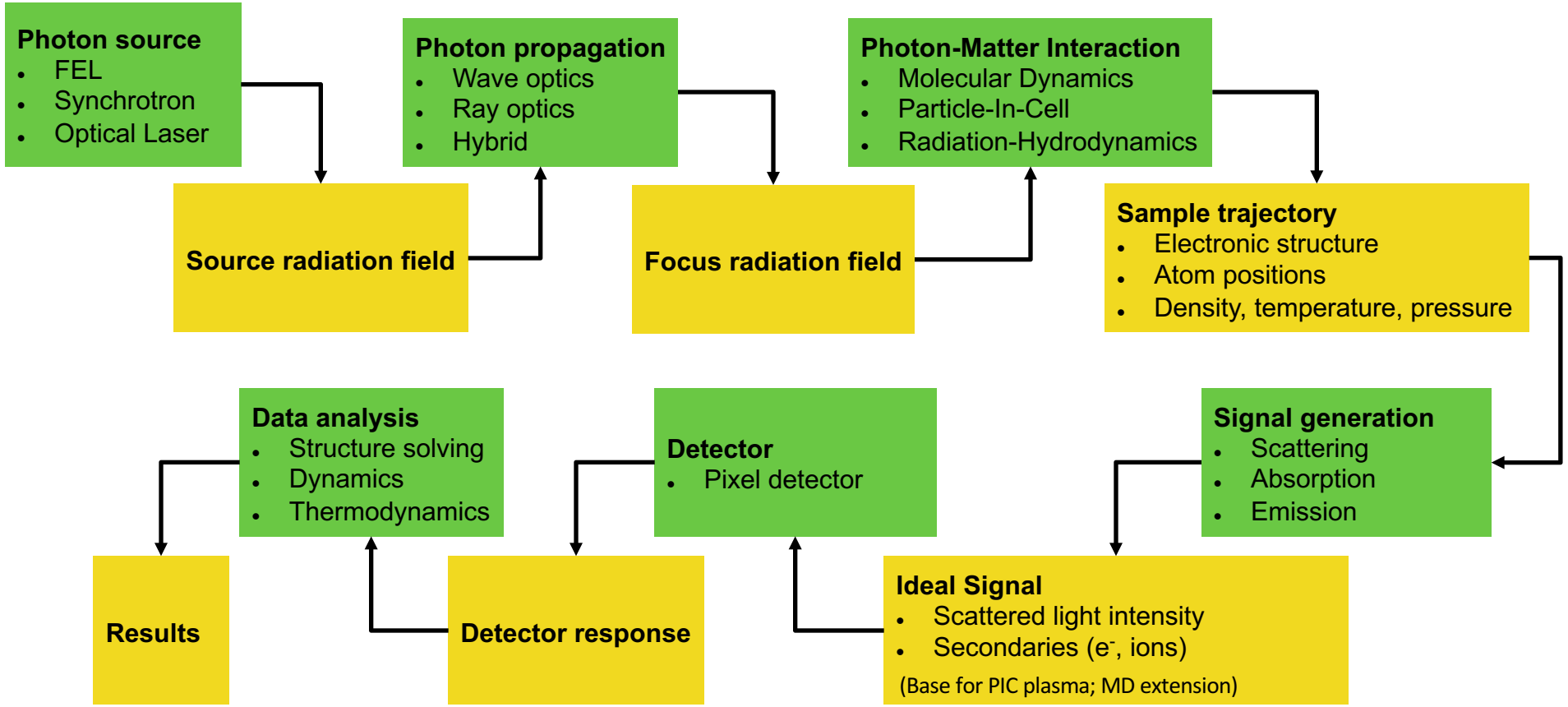
Photon-Mater Interaction

- *Electronic structure*
- *Atom positions*
- *Density, temperature, pressure*

Yoon et al. Scientific Reports **6** 24791 (2016) https://github.com/PaNOSC-ViNYL/SimEx

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No. 823852

SIMEX provides user interfaces and data formats for start-to-end photon experiment simulations



Calculators: python APIs to advanced simulation codes

Data interfaces based on open metadata standards: openPMD

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Interfaced simulation codes

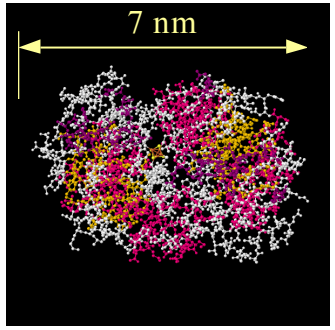
X-ray source	FEL	FAST	Yurkov, Schneidmiller (DESY)
	FEL	Genesis/Ocelot	S. Reiche / G. Geloni et al.
	Synchrotron	Oasys	L. Rebuffi, M. Sanchez-Rio
Propagation	coherent wavefront prop.	WPG/SRW	Samoylova, Buzmakov, Chubar
	x-ray tracing	Oasys	L. Rebuffi, M. Sanchez-Rio
X-ray photon matter interaction		XMDYN & XATOM	Z. Jurek et al. (CFEL)
	Atoms, molecules, clusters	MCMD	P. Ho et al (ANL)
		HF/LDA	H. Quiney et al. (U Melbourne)
Optical photon matter interaction	1D Rad-Hydro	Esther	Colombier et al. (CEA)
	2D Rad-Hydro	Multi2D	R. Ramiz et al.
	3D PIC	PIConGPU	M. Bussmann et al. (HZDR)
Signal generation	molecule, cluster scattering	SingFEL	C.H. Yoon (LCLS)
	Plasma SAXS	paraTAXIS	T. Kluge et al. (HZDR)
	Plasma Compton/Thomson	XRTS	G. Gregori, CFG
	Crystal diffraction	CrystFEL/pattern_sim	T. White et al. (CFEL)
	EXAFS	FEFF8L	J.J. Rehr et al. (U Washington)
	Diffraction (large systems)	GAPD	J. C. E, S. N. Luo (PIMS China)
Detector simulation	2D Pixel detectors	X-CSIT, Karabo	T. Rüter et al. (XFEL)
Analysis/Reconstruction	Pattern orientation	EMC	N.D. Loh (Singapore)
	Phasing	DM	

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

What has been done with SimEx

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

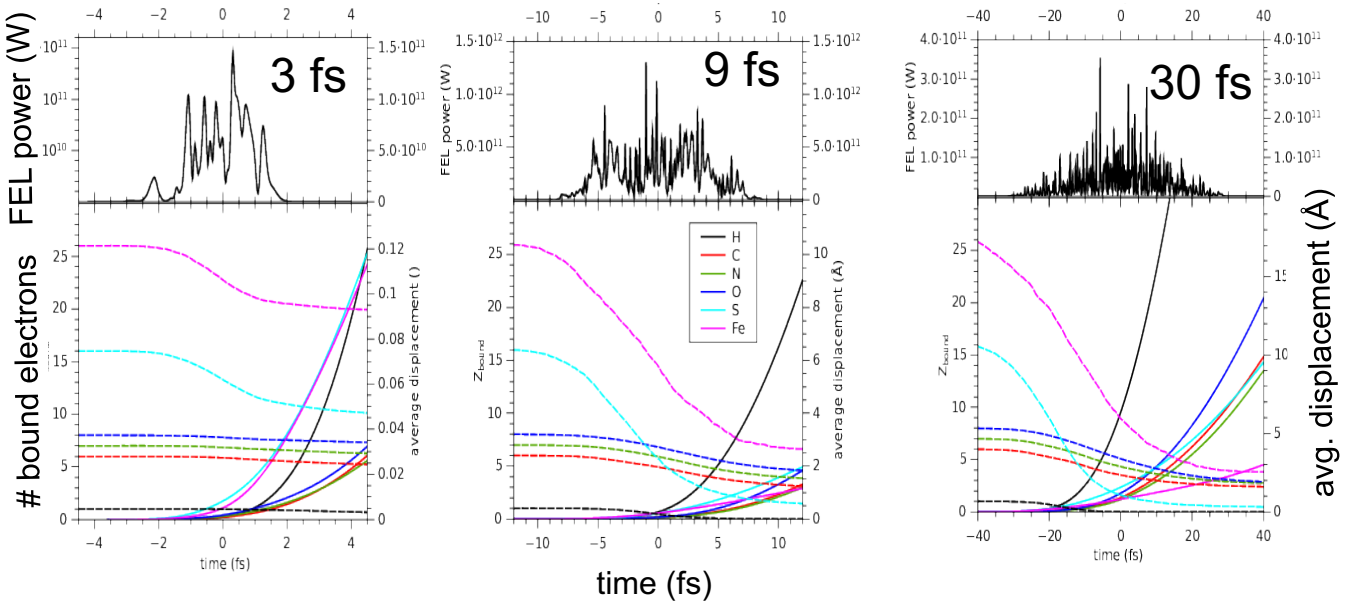
Identify optimal pulse duration for small protein imaging



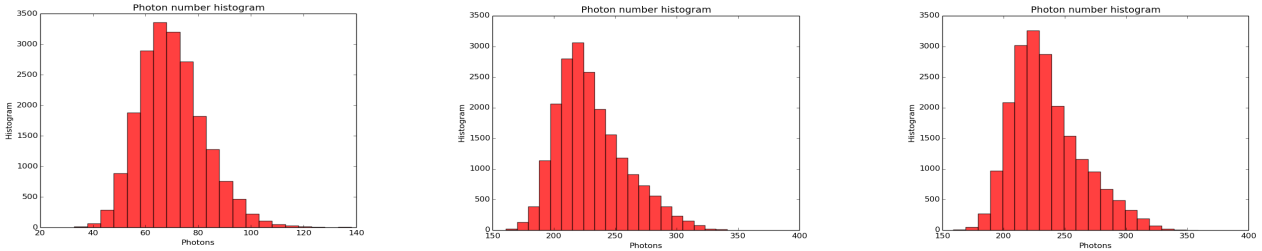
sample molecule 2NIP

Atom	τ_{Auger}(fs)
C	10.7
N	7.1
O	4.9
S	1.3
Fe	2.0

photons per pulse = $1 \cdot 10^{11}$ | $5 \cdot 10^{11}$ | $1 \cdot 10^{12}$



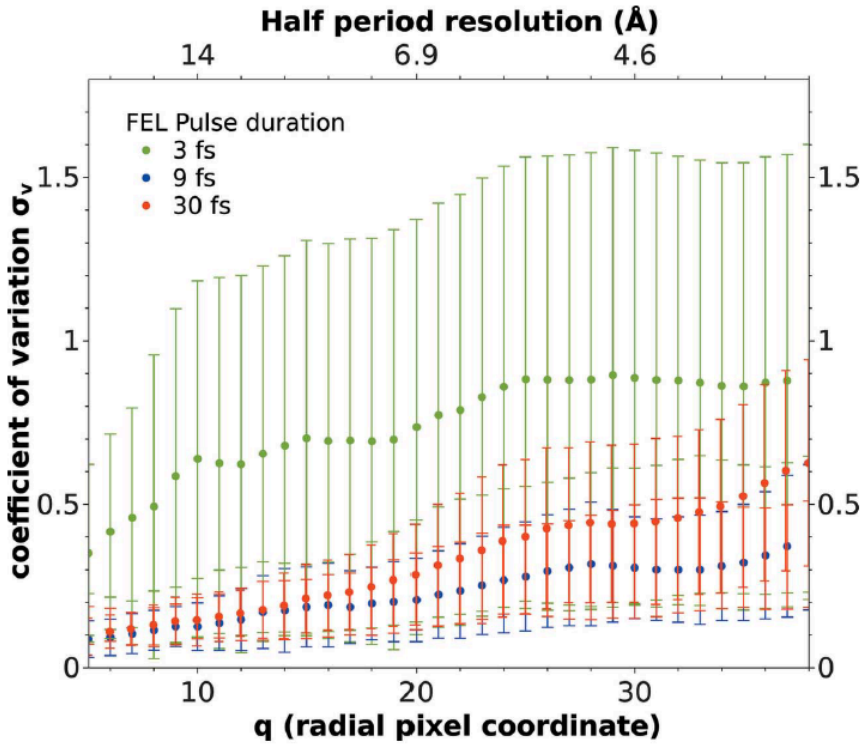
diffracted photons = 70+/-12 | 230+/-28 | 234+/-29

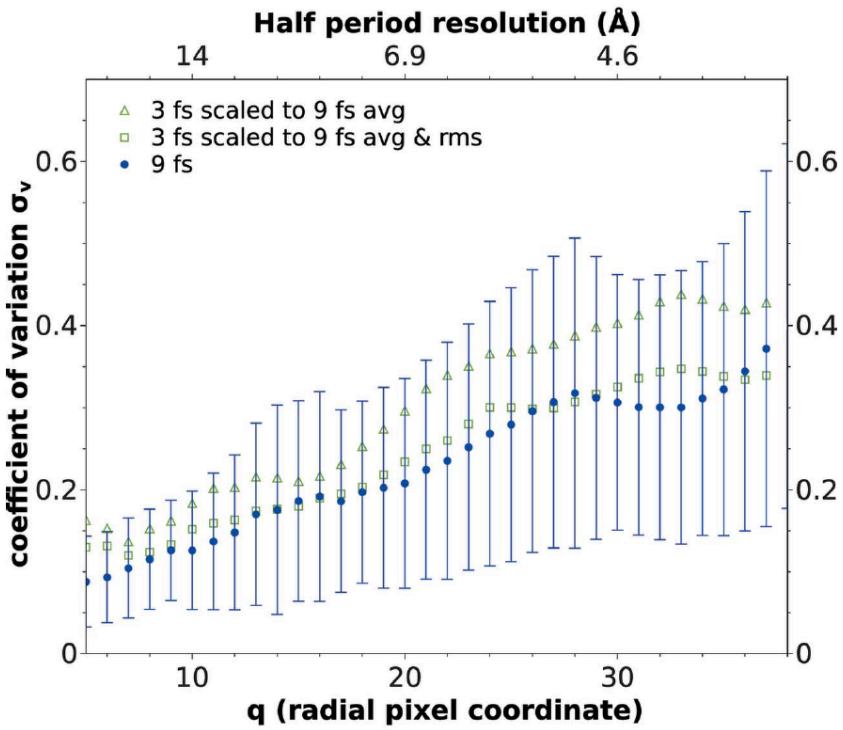


CFG et al., IUCrJ **4** (2017) Yoon et al. Scientific Reports **6** 24791 (2016)

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Identify optimal pulse duration for small protein imaging



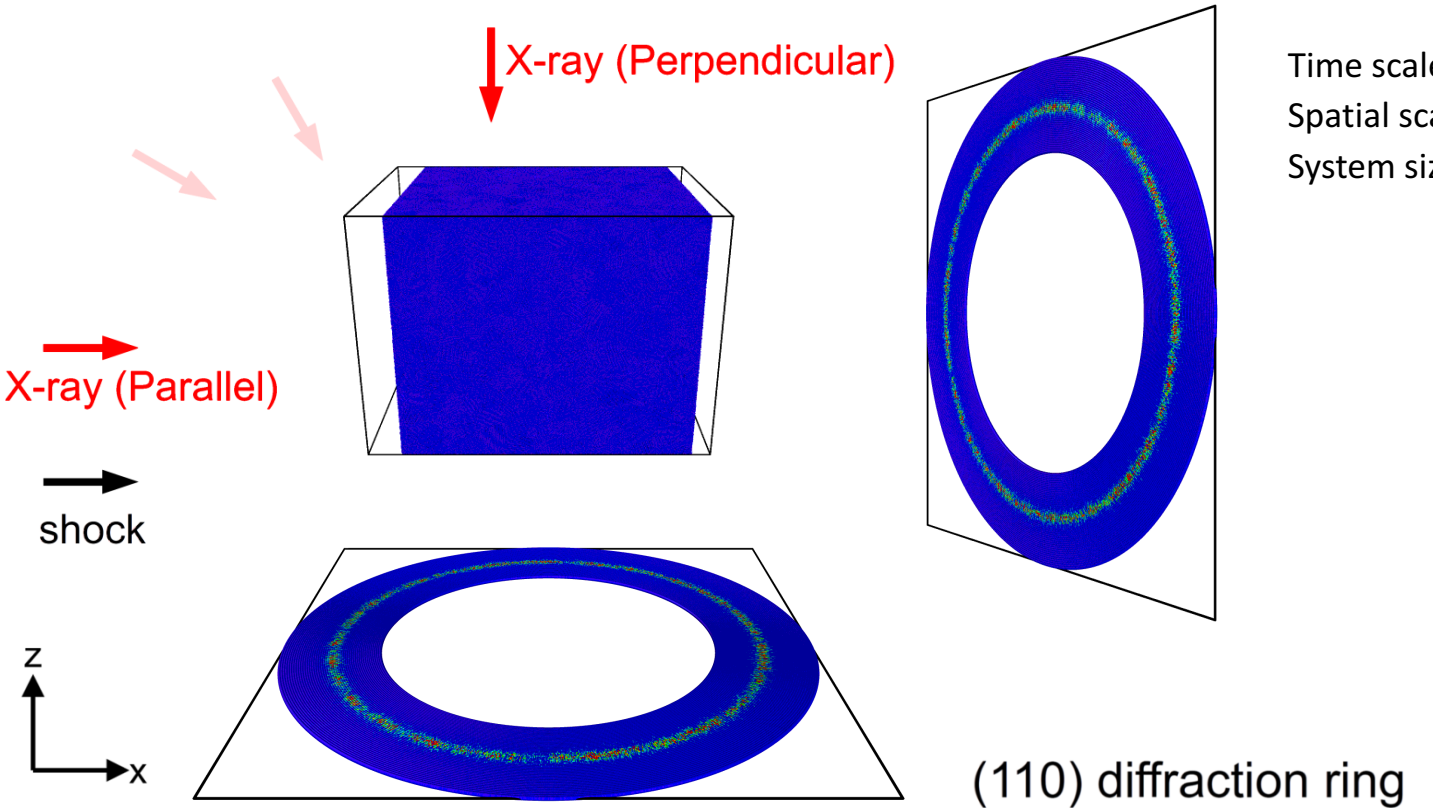


CFG et al., IUCrJ **4** (2017)

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Methodology validation example

- Deducing density and strength of nanocrystalline diamond under shock compression for advanced light sources



Time scale: ps
Spatial scale: nm
System size: 19 million atoms

Y.Y. Zhang, M.X. Tang, Y. Cai, J.C. E, and S.N. Luo, *Journal of Synchrotron Radiation* 26, 413 (2019).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

$$d_{\mathrm{m}}(hkl) = d_{\mathrm{P}}(hkl)[1+(1-3\cos^2\psi)\,Q(hkl)]$$

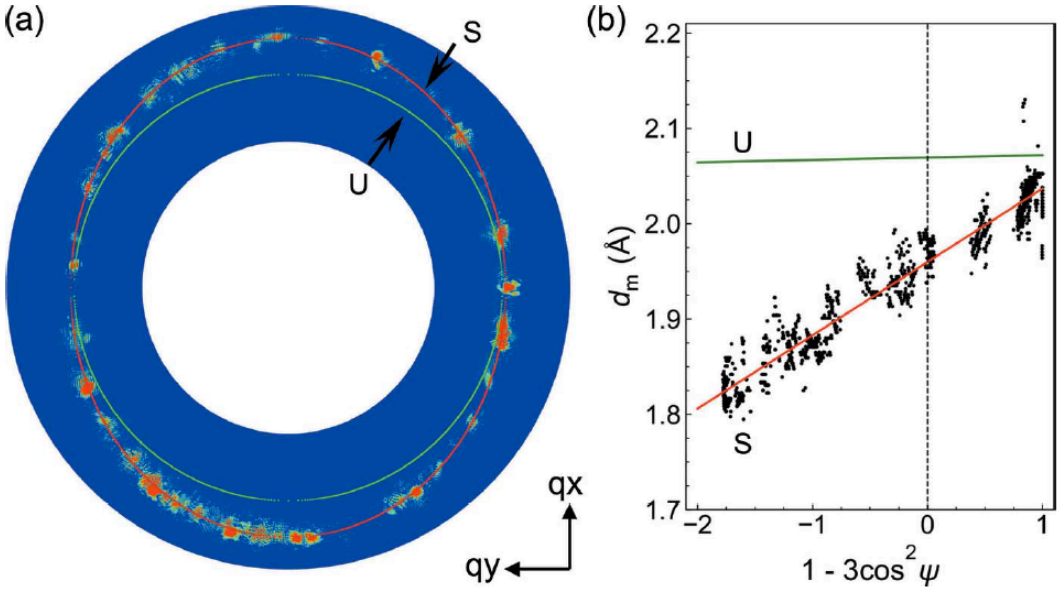
- d_{m}: measured d-spacing
- d_{P}: d-spacing under σ_P

$$Q(hkl) = \frac{t}{6}\{\alpha[G_{\mathrm{R}}^{X}(hkl)]^{-1} + (1-\alpha)[G(V)]^{-1}\}$$

- G_R^X: shear modulus under Reuss assumption
- $G(V)$: shear modulus under Voigt assumption
- α: weight factor

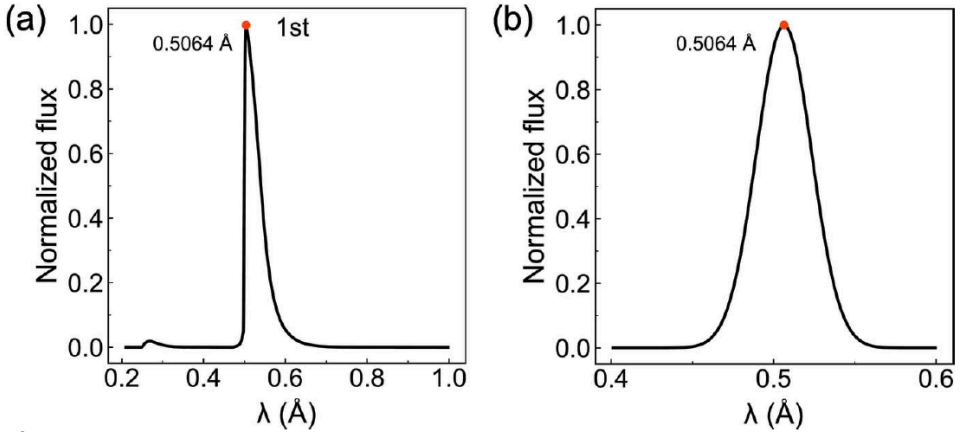
$$t = 6G\langle Q(hkl)\rangle f(x,\alpha)$$

- $f(x,\alpha) \approx 1.0$ as assumption
- G: arithmetic mean of $G(R)$ & $G(V)$

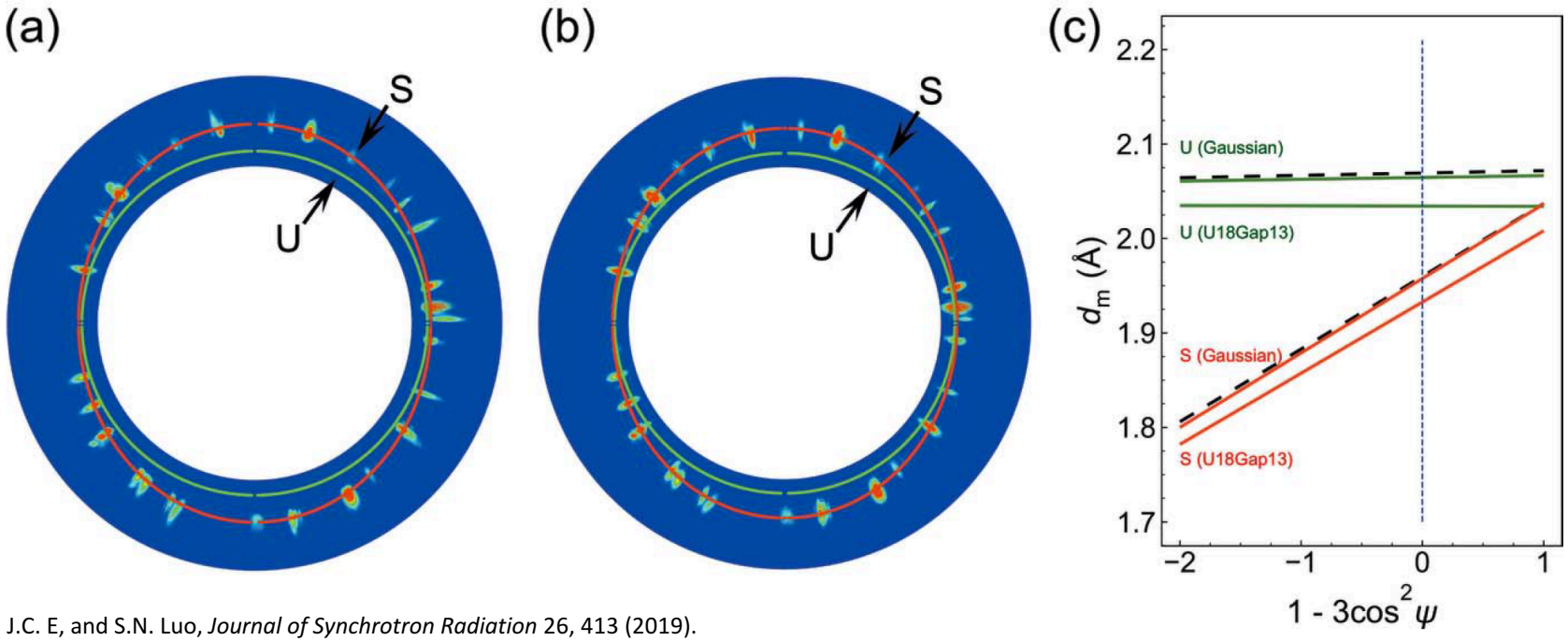


Singh, A. K. (1993). J. Appl. Phys. 73, 4278–4286.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852



- Slope: strength, not affected by the spectrum width or symmetry
- Intercept: density, affected by the asymmetry of the undulator x-ray spectrum



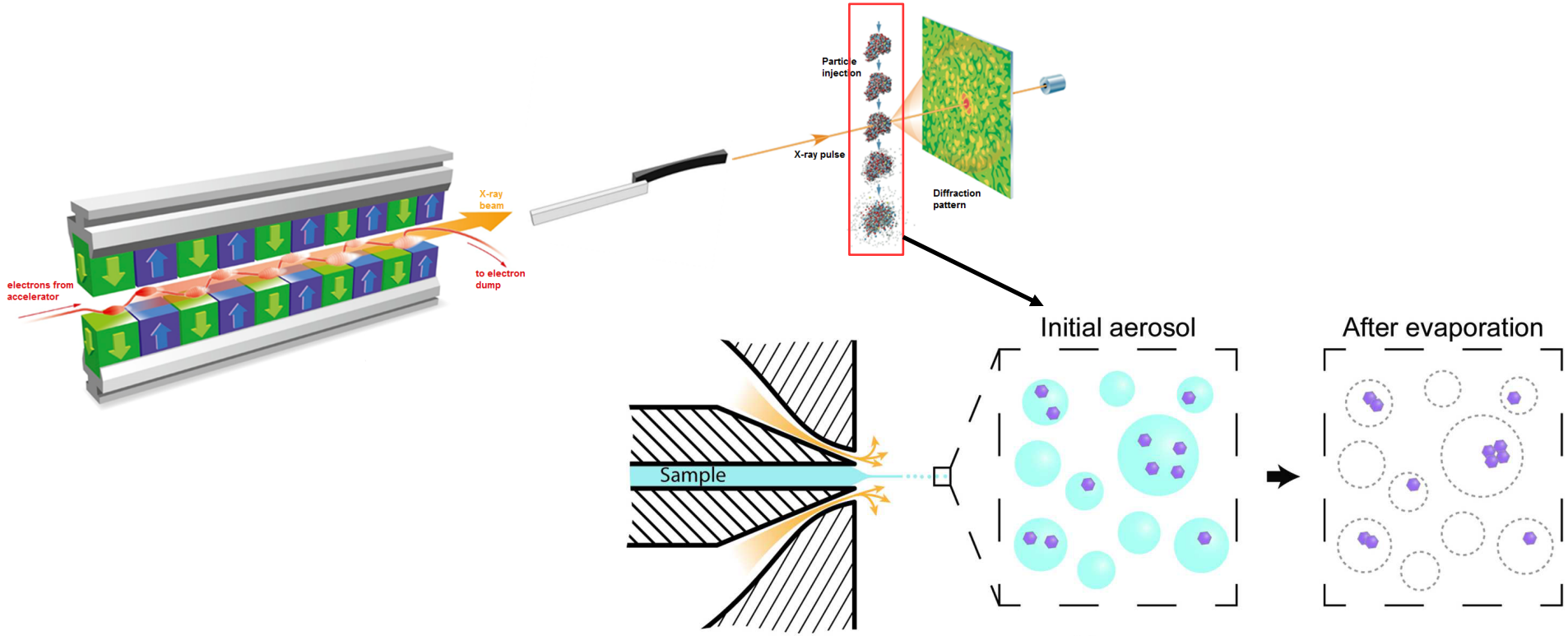
Y.Y. Zhang, M.X. Tang, Y. Cai, J.C. E, and S.N. Luo, *Journal of Synchrotron Radiation* 26, 413 (2019).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

What is going on with SimEx

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Solvated proteins are injected through the Gas Dynamic Virtual Nozzle

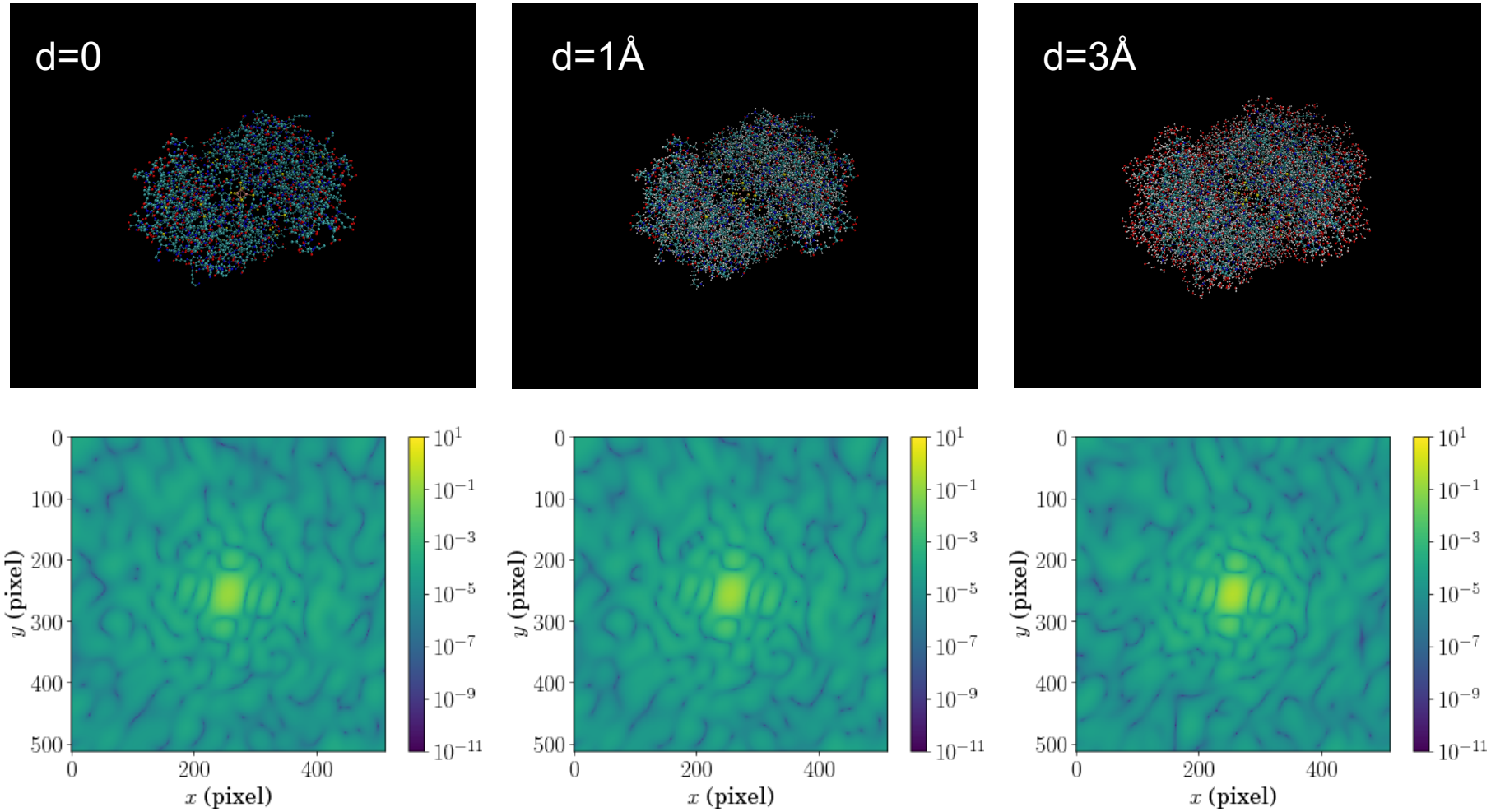


Up to which sample layer of thickness can people still successfully orient or phase the data?

Daurer et al., IUCrJ **4**, 251 (2017)

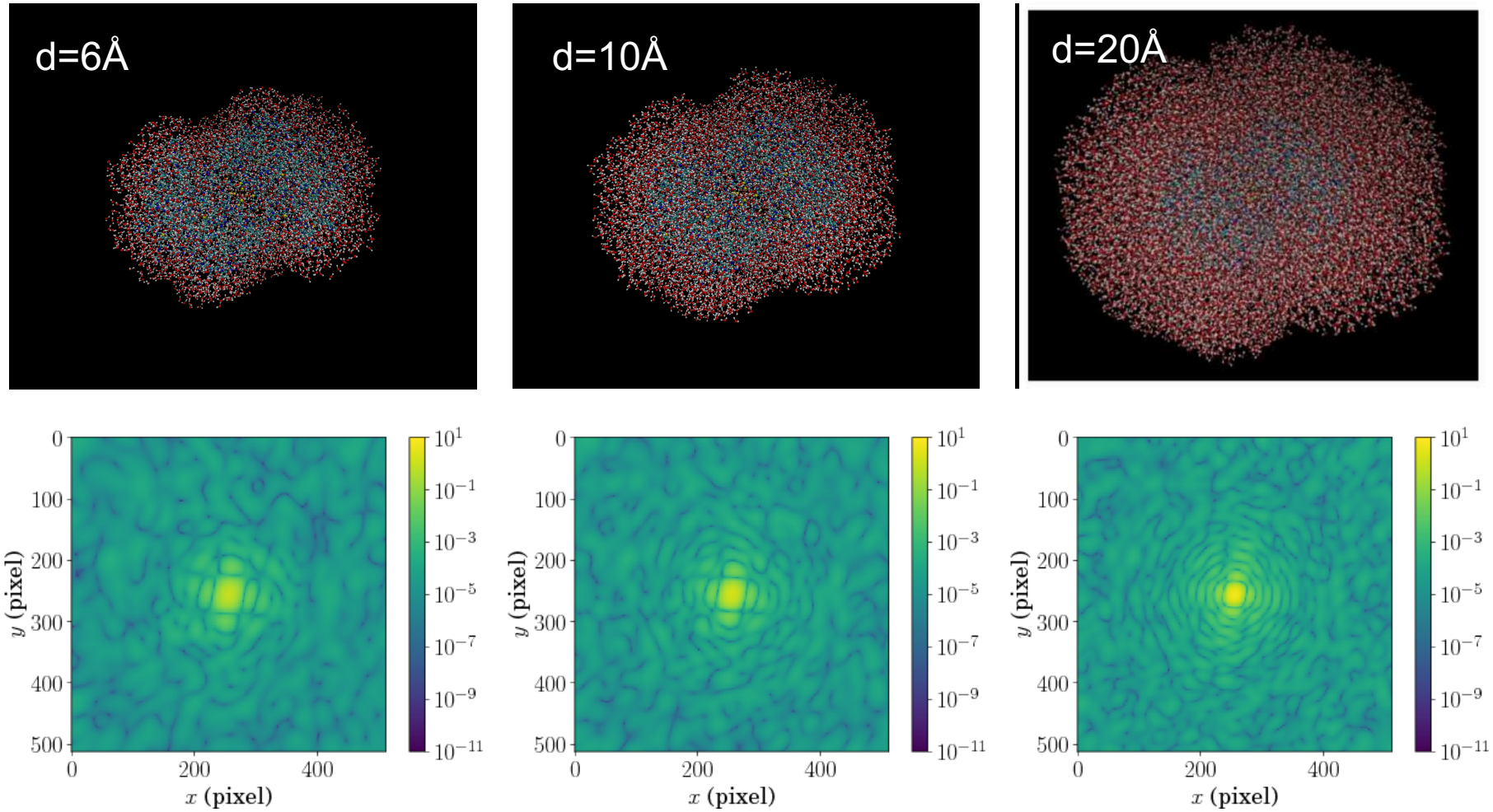
This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Solvated proteins are injected through the Gas Dynamic Virtual Nozzle



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Solvated proteins are injected through the Gas Dynamic Virtual Nozzle



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Simulated Serial Crystallography

- Diffraction simulation jupyter notebook

In [1]:
```
from timeit import default_timer as timer
import os,shutil
from SimEx import *
```

```
initializing ocelot...

WARNING: esther_execute could not be imported. This is most probably due to Esther not being installed or not found.
Expect RunTimeErrors when attempting to run the EstherPhotonMatterInteractor.backengine().
```

Data path setting ¶

In [ ]:
```
data_path = '../../../data/simulation/'
out_path = "xstal/xstal_diffr100"
# Remove previous results
shutil.rmtree(data_path+out_path,True)
os.chdir(data_path)
os.getcwd()
```

Configure photon beam properties

In [ ]:
```
beamParam = PhotonBeamParameters(
    photon_energy = 4972.0 * electronvolt,  # photon energy in eV
    beam_diameter_fwhm=130e-9 * meter,      # focus diameter in m
    pulse_energy=100e-3 * joule,           # pulse energy in J
    photon_energy_relative_bandwidth=0.003, # relative bandwidth dE/E
    divergence=0.0 * radian,                # Beam divergence in rad
    photon_energy_spectrum_type='tophat',   # Spectrum type. Acceptable values are "tophat", "SASE", and "twocolor")
    )
```

Configure detector geometry

In [ ]:
```
panel = DetectorPanel(ranges={"fast_scan_min" : 0, "fast_scan_max" : 999,
                              "slow_scan_min" : 0, "slow_scan_max" : 999},
                      pixel_size=220.0e-6*meter,
                      energy_response=1.0/electronvolt,
                      distance_from_interaction_plane=0.13*meter,
                      corners={"x" : -500, "y": -500},
                      saturation_adu=1.e6,
                      )
detector_geometry = DetectorGeometry(panels=panel)
```

Set diffractor parameters and pass beam & detector parameters to diffractor

In [ ]:
```
diffParam = CrystFELPhotonDiffractorParameters(
    sample='3WUL.pdb',                      # Looks up pdb file in cwd, if not found, queries from RCSB pdb mirror.
    uniform_rotation=True,                  # Apply random rotation
    number_of_diffraction_patterns=100,     #
    powder=False,                           # Set to True to create a virtual powder diffraction pattern (unested)
    intensities_file=None,                  # File that contains reflection intensities. If set to none, use uniform int
ensity distribution
    crystal_size_range=[1e-7,1e-7],         # Range ([min,max]) in units of metres of crystal size.
    poissonize=False,                       # Set to True to add Poisson noise.
    number_of_background_photons=0,         # Change number to add uniformly distributed background photons.
    suppress_fringes=False,                 # Set to True to suppress side maxima between reflection peaks.
    beam_parameters=beamParam,                   # Beam parameters object from above
    detector_geometry=detector_geometry,       # External file that contains the detector geometry in CrystFEL notatio
n.
    )
```

Launch simulation

In [ ]:
```
diffractor.backengine()
diffractor.saveH5()
```

Analysis

In [ ]:
```
analysis = DiffractionAnalysis(diffractor.output_path)
analysis.plotPattern()
```

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Simulated Serial Crystallography

- Diffraction simulation jupyter notebook

In [1]:
```
from timeit import default_timer as timer
import os,shutil
from SimEx import *
```

```
initializing ocelot...

WARNING: esther_execute could not be imported. This is most probably due to Esther not being installed or not found.
Expect RunTimeErrors when attempting to run the EstherPhotonMatterInteractor.backengine().
```

Data path setting ¶

In [ ]:
```
data_path = '../../../data/simulation/'
out_path = "xstal/xstal_diffr100"
# Remove previous results
shutil.rmtree(data_path+out_path,True)
os.chdir(data_path)
os.getcwd()
```

Configure photon beam properties

In [ ]:
```
beamParam = PhotonBeamParameters(
    photon_energy = 4972.0 * electronvolt,  # photon energy in eV
    beam_diameter_fwhm=130e-9 * meter,      # focus diameter in m
    pulse_energy=100e-3 * joule,           # pulse energy in J
    photon_energy_relative_bandwidth=0.003, # relative bandwidth dE/E
    divergence=0.0 * radian,                # Beam divergence in rad
    photon_energy_spectrum_type='tophat',   # Spectrum type. Acceptable values are "tophat", "SASE", and "twocolor")
    )
```

Configure detector geometry

In [ ]:
```
panel = DetectorPanel(ranges={"fast_scan_min" : 0, "fast_scan_max" : 999,
                              "slow_scan_min" : 0, "slow_scan_max" : 999},
                      pixel_size=220.0e-6*meter,
                      energy_response=1.0/electronvolt,
                      distance_from_interaction_plane=0.13*meter,
                      corners={"x" : -500, "y": -500},
                      saturation_adu=1.e6,
                      )
detector_geometry = DetectorGeometry(panels=panel)
```

Set diffractor parameters and pass beam & detector parameters to diffractor

In [ ]:
```
diffParam = CrystFELPhotonDiffractorParameters(
    sample='3WUL.pdb',                      # Looks up pdb file in cwd, if not found, queries from RCSB pdb mirror.
    uniform_rotation=True,                  # Apply random rotation
    number_of_diffraction_patterns=100,     #
    powder=False,                           # Set to True to create a virtual powder diffraction pattern (unested)
    intensities_file=None,                  # File that contains reflection intensities. If set to none, use uniform int
ensity distribution
    crystal_size_range=[1e-7,1e-7],         # Range ([min,max]) in units of metres of crystal size.
    poissonize=False,                       # Set to True to add Poisson noise.
    number_of_background_photons=0,         # Change number to add uniformly distributed background photons.
    suppress_fringes=False,                 # Set to True to suppress side maxima between reflection peaks.
    beam_parameters=beamParam,                   # Beam parameters object from above
    detector_geometry=detector_geometry,       # External file that contains the detector geometry in CrystFEL notatio
n.
    )
```

Launch simulation

In [ ]:
```
diffractor.backengine()
diffractor.saveH5()
```

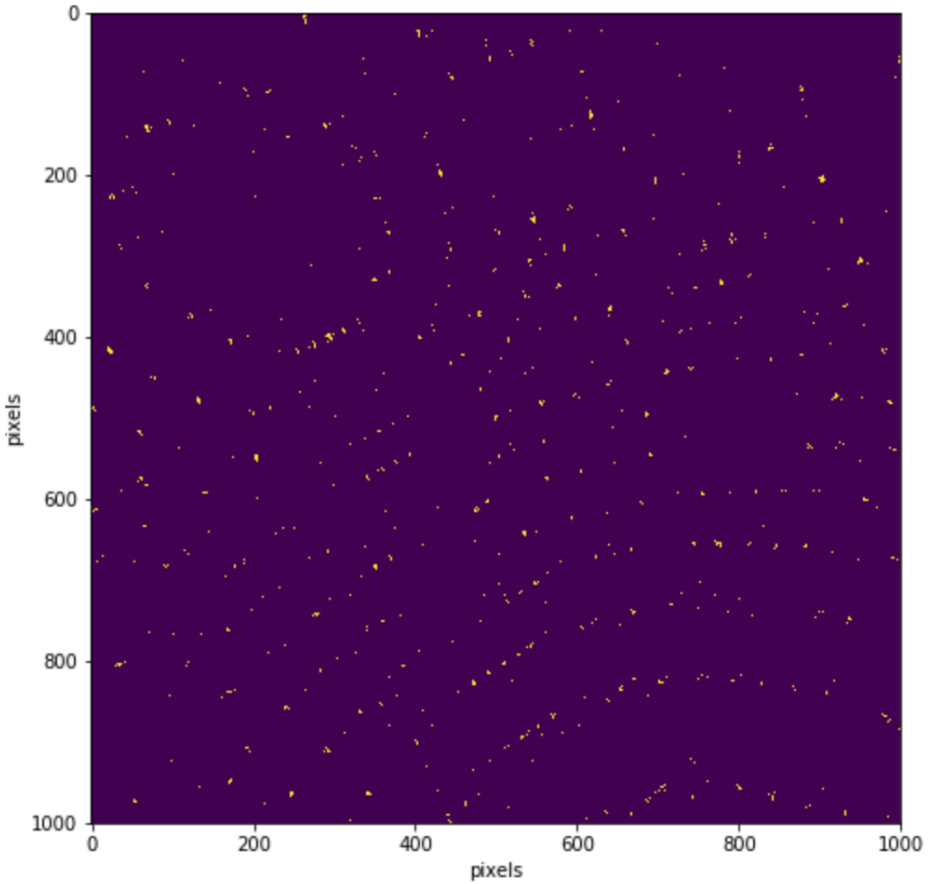
Analysis

In [ ]:
```
analysis = DiffractionAnalysis(diffractor.output_path)
analysis.plotPattern()
```

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Simulated Serial Crystallography

- Multiple diffraction patterns can be generated

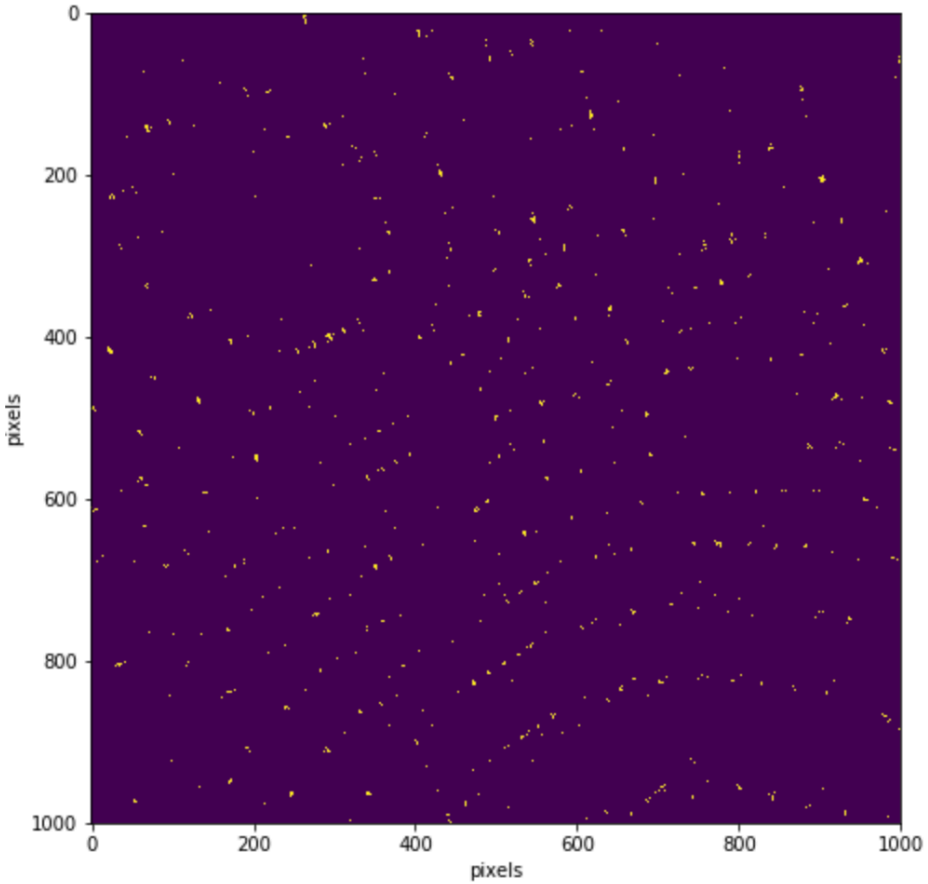


One example diffraction simulation result

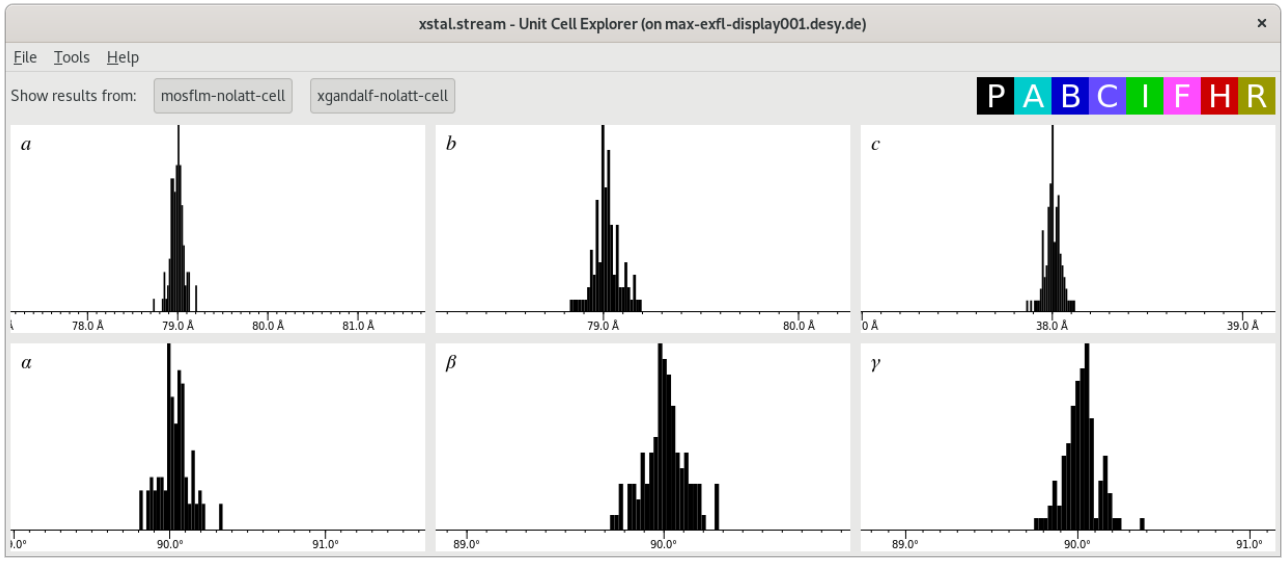
This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Simulated Serial Crystallography

- Multiple diffraction patterns can be generated **and analyzed**



One example diffraction simulation result



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Continue developing

PaNOSC-ViNYL / SimEx

Unwatch 19

Code | Issues 65 | Pull requests 0 | Actions | Projects 1 | Wiki | Security | I

Gaussian source fixes #190

Merged CFGrote merged 12 commits into develop from gaussian_source_fixes 5 days ago

Conversation 0 | Commits 12 | Checks 0 | Files changed 167

CFGrote commented 6 days ago • edited Member

adding tests and some bugfixes.

test files moved to zenodo, will be downloaded before tests are run.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

What is expected in the future

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Interfaced simulation codes

X-ray source	FEL	FAST	Yurkov, Schneidmiller (DESY)
	FEL	Genesis/Ocelot	S. Reiche / G. Geloni et al.
	Synchrotron	Oasys	L. Rebuffi, M. Sanchez-Rio
Propagation	coherent wavefront prop.	WPG/SRW	Samoylova, Buzmakov, Chubar
	x-ray tracing	Oasys	L. Rebuffi, M. Sanchez-Rio
X-ray photon matter interaction		XMDYN & XATOM	Z. Jurek et al. (CFEL)
	Atoms, molecules, clusters	MCMD	P. Ho et al (ANL)
		HF/LDA	H. Quiney et al. (U Melbourne)
Optical photon matter interaction	1D Rad-Hydro	Esther	Colombier et al. (CEA)
	2D Rad-Hydro	Multi2D	R. Ramiz et al.
	3D PIC	PIConGPU	M. Bussmann et al. (HZDR)
Signal generation	molecule, cluster scattering	SingFEL	C.H. Yoon (LCLS)
	Plasma SAXS	paraTAXIS	T. Kluge et al. (HZDR)
	Plasma Compton/Thomson	XRTS	G. Gregori, CFG
	Crystal diffraction	CrystFEL/pattern_sim	T. White et al. (CFEL)
	EXAFS	FEFF8L	J.J. Rehr et al. (U Washington)
	Diffraction (large systems)	GAPD	J. C. E, S. N. Luo (PIMS China)
Detector simulation	2D Pixel detectors	X-CSIT, Karabo	T. Rüter et al. (XFEL)
Analysis/Reconstruction	Pattern orientation	EMC	N.D. Loh (Singapore)
	Phasing	DM	

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Interfaced simulation codes

X-ray photon matter interaction		XMDYN & XATOM	Z. Jurek et al. (CFEL)
	Atoms, molecules, clusters	MCMD HF/LDA	P. Ho et al (ANL) H. Quiney et al. (U Melbourne)

GROMACS, molecular dynamics package mainly designed for simulations of proteins, lipids, and nucleic acids.

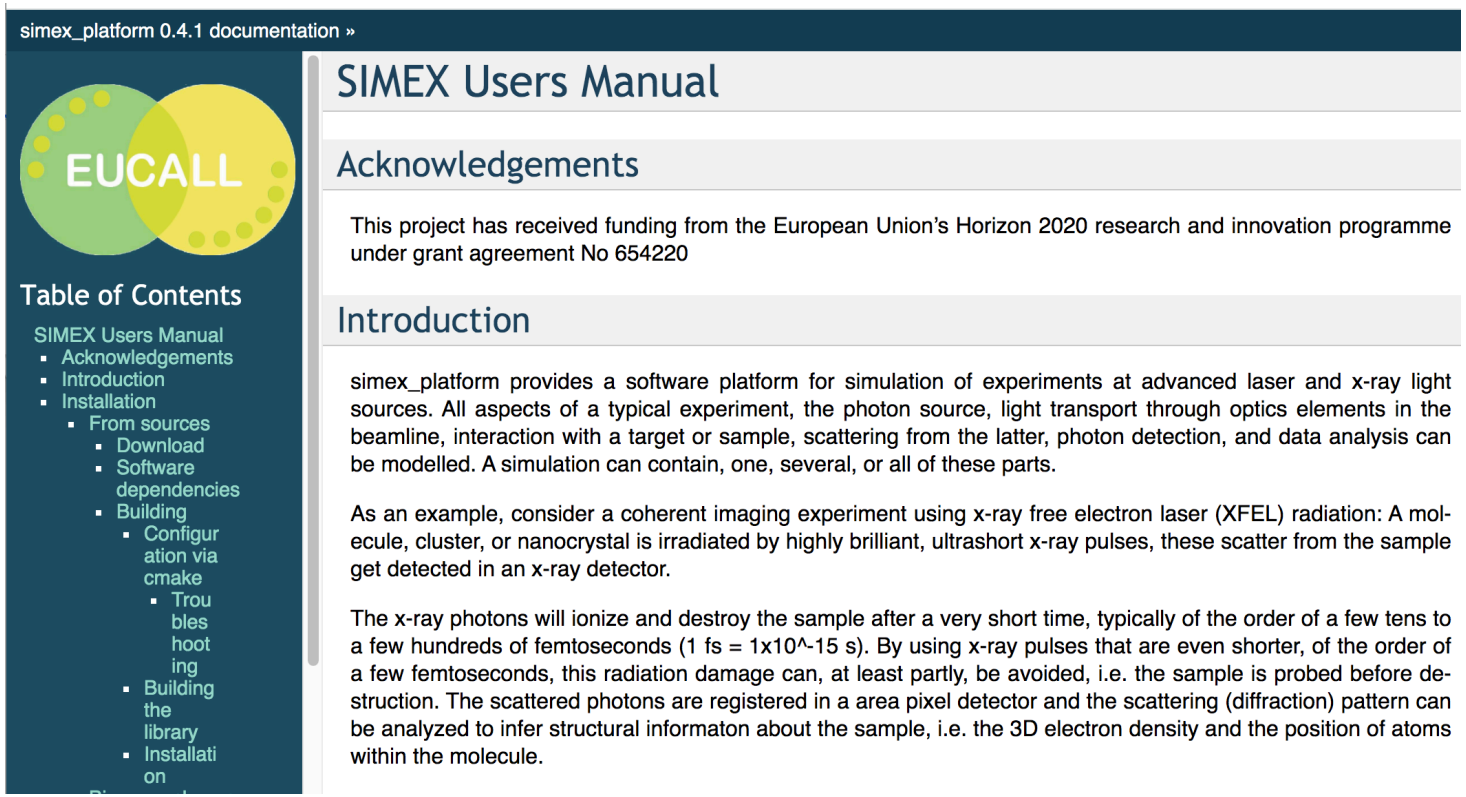
Cretin, radiation transfer code

Radiation damage

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Documentation and refactor

- Installation instruction
- Docstrings
- Tutorials



https://panosc-vinyl.github.io/SimEx/

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

Some questions for the API documentation

- Parameter specification:
  - Link
  - **Serialize parameters** or **dump**?

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852

panosc
photon and neutron
open science cloud

Thank you

Juncheng.e@xfel.eu

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 823852