SIMEX: Simulation of Experiments at Advanced Light Sources

C Fortmann–Grote[1], A A Andreev[2,3], R Briggs[4], M Bussmann[5], A Buzmakov[6], M Garten[5,7], A Grund[5,7], A Huebl[5,7], S Hauff[1], A Joy[8], Z Jurek[9,10], N D Loh[11], T Rüter[1], L Samoylova[1], R Santra[9,10,12], E A Schneidmiller[13], A Sharma[3], M Wing[8,13], S Yakubov[13], C H Yoon[14], M V Yurkov[13], B Ziaja[9,10,15], A P Mancuso[1]

[1]European XFEL GmbH, Holzkoppel 4, 22869 Schenefeld, Germany
[2]Max Born Institute, Berlin, Germany
[3]ELI ALPS, Szeged, Hungary
[4]European Synchrotron Radiation Facility ESRF, Grenoble, France
[5]Helmholtz–Zentrum Dresden–Rossendorf, Germany
[6]Institute of Crystallography, Russian Academy of Sciences, Moscow, Russia
[7]Technische Universität Dresden, Germany
[8]University College London, UK
[9]Center for Free Electron Laser Science, Deutsches Elektronen Synchrotron, Hamburg, Germany
[10]The Hamburg Center for Ultrafast Imaging, Hamburg, Germany
[11]Department of Physics, National University of Singapore, Singapore
[12]Department of Physics, University of Hamburg, Germany
[13]Deutsches Elektronen Synchrotron, Hamburg, Germany
[14]Linac Coherent Light Source, SLAC National Accelerator Laboratory, Menlo Park, USA
[15]Institute of Nuclear Physics, Polish Academy of Sciences, Krakow, Poland

E-mail: [1]carsten.grote@xfel.eu

Abstract. Realistic simulations of experiments at large scale photon facilities, such as optical laser laboratories, synchrotrons, and free electron lasers, are of vital importance for the successful preparation, execution, and analysis of these experiments investigating ever more complex physical systems, e.g. biomolecules, complex materials, and ultra–short lived states of highly excited matter. Traditional photon science modelling takes into account only isolated aspects of an experiment, such as the beam propagation, the photon–matter interaction, or the scattering process, making idealized assumptions about the remaining parts, e.g. the source spectrum, temporal structure and coherence properties of the photon beam, or the detector response. In SIMEX, we have implemented a platform for complete start–to–end simulations, following the radiation from the source, through the beam transport optics to the sample or target under investigation, its interaction with and scattering from the sample, and its registration in a photon detector, including a realistic model of the detector response to the radiation. Data analysis tools can be hooked up to the modelling pipeline easily. This allows researchers and facility operators to simulate their experiments and instruments in real life scenarios, identify promising and unattainable regions of the parameter space and ultimately make better use of valuable beamtime.

1. Introduction

Simulations of photon science experiments that cover the complete path of the photon from the source through beamline optics to the sample, its interaction with and scattering from the sample, and its detection have recently been used to model coherent diffraction imaging experiments of single biomolecules at Ångstrom resolution [1]. Single particle imaging (SPI) experiments (schematically depicted in Fig. 1a) are performed at x-ray free-electron lasers (XFELs) that meet the stringent requirements for ultra–short and intense x–ray pulses. Such start–to–end simulations allow us to investigate the feasibility and potential outcome of complex experiments under real–world conditions. Every part of the experiment is simulated such that imperfections in the sources spectral, temporal, and spatial structure, in the x–ray optical components, radiation damage to the sample, and detector limitations are all taken into account, allowing us to study how they affect the experimental observable and how these error sources correlate with each other.

The computer program `simex_platform` [2] was derived from the SPI simulation suite `simS2E` [1]. Besides SPI, `simex_platform` supports simulation of various types of experiments, involving various light sources (synchrotron sources, free-electron lasers (FELs), optical lasers), various beamline optics, samples, and diagnostic techniques. A python library provides standardized user interfaces to the simulation codes and their parameters. This approach was adopted from the Atomic Simulation Environment (ASE) [3, 4] which provides python objects (so called *Calculators*) that act as interfaces to ab–initio electronic structure codes. Users can embed their own codes into our simulation environment and run them under more realistic conditions compared to running them isolated with idealized parameters and initial conditions.

Some simulation codes (called *Backengines*) are readily shipped with `simex_platform`, in particular for SPI simulations. For other codes, we have developed standardized data formats (data interfaces) to integrate them into simulation workflows. Our software has modern and flexible deployment options including cmake, binary packages, and docker containers packed for various operating systems [5] and runs in parallel on high–performance computing clusters.

2. Concepts and structure of `simex_platform`

2.1. Simulation baseline

Currently, `simex_platform` supports simulation of photon experiments that follow a sequential pattern involving five steps:

(i) Photon source: generation of radiation e.g. by accelerated charged particles or optical resonators.

(ii) Photons propagate through a beamline consisting of apertures, slits, mirrors, gratings, and lenses.

(iii) The photons interact with the sample.

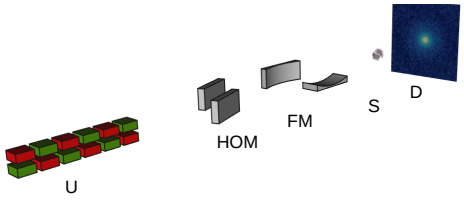
(iv) Scattered photons travel towards the detector.

(v) A detector registers incoming photons and produces a digital signal.

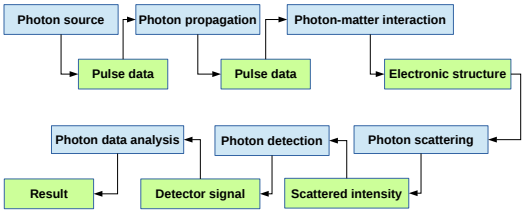
SPI, for which a schematic setup is shown in Fig. 1a is one example for a baseline application. The data flow between *Calculators* and intermediate data containers for which we have defined standardized data hierarchies and formats is shown in Fig. 1b.

2.2. Calculators

Each of the five simulation tasks in a baseline application is represented by a suitable *Calculator*. *Calculators* are organized in a lightweight abstraction scheme consisting of three abstraction layers as depicted in Fig. 2



(a) Schematic of a XFEL single–particle imaging experiment. Photons emitted from the undulator (U) of the XFEL are deflected by offset mirrors (HOM) and focussed by focussing mirrors (FM) onto the sample (S). Diffracted radiation is captured in the detector (D).



(b) Baseline workflow in `simex_platform`. Blue boxes represent *Calculators*, green boxes are data interfaces.

Figure 1.

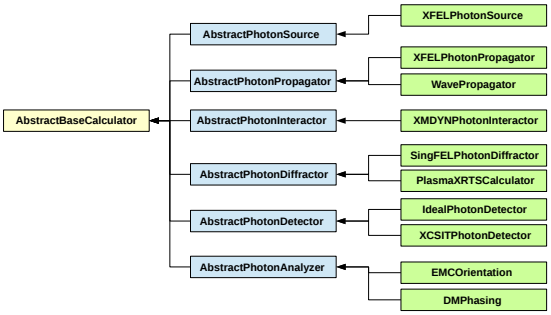


Figure 2. SIMEX *Calculators* and their relationships

The top level class *AbstractBaseCalculator* is a pure virtual object. It declares virtual members that a deriving *Calculator* must implement; in this way we ensure basic functionality of the *Calculator*, in particular that *Calculators* can exchange data and status information with each other. These virtual members are a data container, where the data manipulated by the *Calculator* is stored, together with corresponding query and assignment methods, furthermore methods for *Backengine* execution as well as input-output (IO) utilities. The intermediate level (blue boxes) provides abstract base classes for each block of the simulation workflow: AbstractPhotonSource, AbstractPhotonPropagator, AbstractPhotonInteractor, AbstractPhotonDiffractor, and AbstractPhotonDetector. A sixth abstract *Calculator*, AbstractPhotonAnalyzer is provided to ease integration of post–processing data analysis codes.

These classes are mainly responsible for declaring a list of variable names expected from the previous *Calculator* in the workflow and a list of variable names provided for the next *Calculator* in the workflow. Specialized *Calculators* adjust these lists according to their needs.

The third and final abstraction layer (green boxes) implements the interfaces to the simulation codes (*Backengines*) that perform the actual numerical work on the input data. They convert parameters and input data into suitable formats for the *Backengine*, before issuing the system or library calls, gather the results after the calculation has finished, and write them to an interface data file.

2.3. Interfaces

Start–to–end simulations in `simex_platform` require that any two subsequent *Calculators* employed in the simulation pipeline can communicate data amongst each other. The data source has to write the data in a format that the ensuing data sink can handle and interpret correctly. In `simex_platform`, we chose the Hierarchical Data Format (hdf5) [6] as the underlying format for all simulation data files. Data consistency in the workflow is realized through the aforementioned mechanisms of exchanging information about the required and provided data sets among the *Calculators*. Recently, we have started to adopt a more general approach of defining inter–calculator interfaces based on the open standard for particle–mesh data `openPMD` [7]. In particular, the transparent handling of physical units in `openPMD` makes it possible to implement generic unit conversion mechanisms.

3. Applications

As a first development milestone `simex_platform` supports simulation of three different baseline applications: single particle imaging at the European XFEL, coherent x–ray scattering from high power laser excited plasmas and x–ray probing of warm dense matter produced by high energy laser shock compression.

3.1. Single particle imaging

Femtosecond coherent x–ray pulses are generated in the SASE1 beamline of the European XFEL. The self-amplification through stimulated emission (SASE) process is simulated with the code `FAST` [8], which gives the electric field distribution at the exit of the undulator section in a plane perpendicular to the beam direction as a function of time (discrete time slices). `FAST` is not public, we use precomputed datasets publicly available from a web database [9], instead.

The pulses are then propagated through the beamline and the SPB–SFX instrument's [10] focussing optics [11] to the sample interaction point. Propagation is modelled with the python library `WPG` [12, 13] with realistic parameters for the various optical components in the beamline. The output is again the electric field distribution as a function of time in the sample interaction plane transverse to the beam direction.

The ultra–short and intense x–ray pulses interact with the sample primarily via photo–ionization processes. Secondary Auger electrons quickly launch an ionization cascade creating a nano–plasma that starts to expand on timescales of 10–100 fs after the first photons hit the sample. These processes are calculated by means of Molecular Dynamics (MD) simulations with the code `XMDYN` [14], coupled to a Monte Carlo (MC) engine that describes the photon–matter interaction processes in a stochastic manner. Rates and cross sections are read from tabulated Slater–Hartree–Fock calculations using the code `XATOM` [14, 15]. We save the entire sample trajectory (atom positions and scattering form factors). We note here in passing that earlier simulations of SPI described in Ref. [16, 17] completely neglect radiation damage.

Scattering from the biomolecule is simulated in the far field approximation with the code `SingFEL` [18]. `SingFEL` utilizes the output from the photon–matter interaction code to calculate the instantaneously scattered light intensity by multiplying the intensity at the sample with the appropriately weighted form factors and geometry factors such as the detector solid angle. The instantaneous diffraction is integrated over the pulse duration and saved as a 2D array representing the pixel area detector.

The detector response to the incoming diffracted radiation can be calculated with the x-ray camera simulation toolkit (`X-CSIT`) software [19]. It describes particle generation, charge transport, and electronic signal processing. This can be used to simulated various detector effects like variations in pixel performance, non–linear gain, electronic noise, and counting statistics.

The detector simulation concludes the simulation pipeline. In SPI, diffraction patterns are submitted to a sequence of data post–processing algorithms to reconstruct the electron density

from the measured diffraction data. `simex_platform` provides interfaces and *Calculators* for orientation and phasing algorithms described in [20].

3.2. Scattering from hot dense plasmas

Hot (few hundreds of eV) and dense (near solid density and beyond) plasmas are generated during the interaction of ultra–intense ($> 1 \times 10^{17}\,\mathrm{W/cm^2}$) ultra–short (of the order 10 fs) optical laser pulses with solid targets (e.g. metal foils). The plasma is characterized by strong density and temperature gradients and plasma instabilities. These features can be characterized by coherent x–ray scattering [21]. The photon source and propagation are described in the same way as for the SPI example above (Sec. 3.1). The short–pulse optical laser–plasma interaction is modelled with `PIConGPU` [22, 23], an open source, explicit, relativistic 3D Particle In Cell (PIC) code employing finite difference time domain techniques [24] to solve the Maxwell equations coupling the laser field and the plasma particles. For x–ray scattering calculations, we describe the radiation by a photon distribution, which we convert from the field distribution delivered by the propagation *Calculator*. The simulation tracks the photons through the plasma volume simulated by `PIConGPU` using the software `paraTAXIS`. The distribution of scattered photons across the detector plane can then be fed into the detector simulation as described in Sec. 3.1.

3.3. X–ray absorption and radiography in warm dense matter

High energy (few tens of J) pulses of optical laser light impinge on a solid target surface creating a shock wave travelling through the target. Pulse shaping and multiple shock compression can create density, pressure, and temperature conditions similar to planetary interiors [25], also referred to as warm dense matter (WDM). The target is probed by x–ray absorption spectroscopy and radiography to measure the thermodynamic conditions before, during, and after traversal of the shock wave.

In this case, the laser–matter interaction is modelled with radiation–hydrodynamic codes, solving the partial differential equations of radiation transport and hydrodynamic plasma motion. Two codes are currently interfaced, the 1D code "Esther" [26] and the 2D code "MULTI2D" [27]. The most important variables considered in these codes are those associated with the extreme conditions generated by the high–power lasers: pressure (density), temperature, and velocity. Feedback from these hydrocode outputs are crucial in the design and implementation of laser shock experiments with x–ray interactions.

By measuring the x–ray absorption as a function of position (radiography) and photon energy (x-ray absorption finestructure spectroscopy (XAFS)), the shock front structure, shock propagation dynamics, and ionic structure of the target can be monitored as a function of time by varying the delay between optical pump pulse and the x–ray probe.

The XAFS signal is a measure of the crystalline structure or liquid near order in the shock compressed target making it possible to identify structural phase transitions induced by the shock. XAFS can be simulated by combining first principle electronic structure methods with linear response theory. A well known implementation is the non–open source code FEFF [28].

Radiography can be used to image e.g. the shock front during dynamical compression. The Oasys [29] framework has the capability to calculate radiographs from simulated density profiles.

4. Summary and Outlook

We have presented in this work an open–source platform for simulation of experiments at advanced laser light sources. Our python library provides user interfaces and data format conventions that enable simulations of entire experiments from source to detector. *Calculator* interfaces to various simulation codes, both open and closed source, already exist, particularly for SPI simulations. For other applications, x–ray scattering from short–pulse laser excited

plasmas and warm dense matter diagnostics, we have collected the simulation codes and user interfaces will be provided in the next development steps.

Data format definitions are derived from the openPMD standard to facilitate communication between subsequent *Calculators*. These data interfaces will also enable us to perform simulations that go beyond the baseline scheme. Examples are simulation of coherent light sources based on laser–plasma accelerated electron beams and the generation of ion beams through laser–plasma interaction, their interaction with matter, and medical applications.

Acknowledgements
CFG, RB, AH, and SY acknowledge support from the European Cluster of Advanced Laser Light Sources (EUCALL) project which has received funding from the *European Unions Horizon 2020 research and innovation programme* under grant agreement No 654220. MW acknowledges support from the Alexander–von–Humboldt Foundation.

References

[1] Chun Hong Yoon et al., in: *Scientific reports* 6 (2016), p. 24791.
[2] URL: `https://github.com/eucall-software/simex_platform`.
[3] S. R. Bahn et al., in: *Comput. Sci. Eng.* 4.3 (2002), pp. 56–66.
[4] URL: `https://wiki.fysik.dtu.dk/ase/`.
[5] Sergey Yakubov et al., in: *NOBUGS*, 2016.
[6] The HDF Group, http://www.hdfgroup.org/HDF5/, 1997.
[7] Axel Huebl et al., 2015, URL: `http://www.openpmd.org`.
[8] E. L. Saldin et al., in: *Nucl. Instrum. Methods Phys. Res., Sect. A* 429.1 (1999), p. 233.
[9] URL: `https://in.xfel.eu/xpd/`.
[10] Adrian Mancuso et al., Technical Report, European XFEL GmbH, 2013.
[11] Richard J Bean et al., in: *Journal of Optics* 18.7 (2016), p. 074011.
[12] Liubov Samoylova et al., in: *J Appl Cryst* 49.4 (2016), pp. 1347–1355.
[13] URL: `https://github.com/samoylv/WPG`.
[14] Zoltan Jurek et al., in: *J Appl Cryst* 49.3 (2016), pp. 1048–1056.
[15] Sang-Kil Son et al., in: *Phys. Rev. A* 83.3 (2011), p. 33402.
[16] Kartik Ayyer et al., in: *Structural Dynamics* 2.4, 041702 (2015).
[17] S. Serkez et al., in: *ArXiv e-prints* (2014), p. 1407.8450.
[18] URL: `www.github.com/eucall-software/singfel`.
[19] A. Joy et al., in: *Journal of Instrumentation* 10.04 (2015), pp. C04022–C04022.
[20] Ne-Te Duane Loh et al., in: *Physical Review E* 80.2 (2009), p. 026705.
[21] T. Kluge et al., in: *Physics of Plasmas* 23.3 (2016), p. 033103.
[22] M Bussmann et al., in: *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, New York, New York, USA, 2013, p. 1.
[23] URL: `https://github.com/ComputationalRadiationPhysics/picongpu`.
[24] Kane Yee, in: *IEEE Transactions on Antennas and Propagation* 14.3 (1966), pp. 302–307.
[25] Y Ping et al., in: *Phys. Rev. Lett.* 96.25 (2006), p. 255003.
[26] J. P. Colombier et al., in: *Phys. Rev. B* 71.16 (2005), p. 165406.
[27] R. Ramis et al., in: *Computer Physics Communications* 180.6 (2009), pp. 977–994.
[28] John J. Rehr et al., in: *Phys. Chem. Chem. Phys.* 12 (2010), pp. 5503–5513.
[29] M. Sanchez del Rio et al., in: *Proc. SPIE 9209* (2014), p. 9209X.