

OASYS: A Software for Beamline Simulations and Synchrotron Virtual Experiments

Manuel Sanchez del Rio^{1,a)} and Luca Rebuffi²

¹*European Synchrotron Radiation Facility, Grenoble (France)*

²*Elettra Sincrotrone Trieste, Trieste (Italy)*

^{a)}Corresponding author: srio@esrf.eu

Abstract. A modern synchrotron beamline requires an important simulation work for its design and optimization using software modelling tools. OASYS (OrAnge SYnchrotron Suite) is an open-source graphical environment for beamline simulation software used in synchrotron facilities.

The OASYS environment provides not only an intuitive and very-easy-to-use graphical interface, but also high flexibility and rapidity for interactive simulations. It allows to quickly define and compare multiple configurations in the same workspace to permit optimizing an X-ray instrument.

OASYS integrates in a synergetic way the most powerful open-source calculation engines available. It interfaces widely used simulation tools for X-ray Optics (e.g. SHADOW for ray tracing, and SRW for wave optics) that are completed with new tools. OASYS provides a mechanism to communicate among the different packages by sending and receiving encapsulated data. The final goal of the OASYS platform is the integration of different packages to completely model synchrotron virtual experiment, retrieving the parameters of the electron beam, calculating the radiation from magnetic structure, then transporting and optimizing the photon beam and eventually including models for interaction with materials to get instrumental functions, study analyzers and detectors and perform ab-initio simulations to support experimental data analysis.

THE PHILOSOPHY OF A VIRTUAL EXPERIMENT: OASYS GOAL

OASYS (ORange SYnchrotron Suite) is a new graphical environment for modelling X-ray experiments. The goal of the OASYS platform is to make available to the user a collection of different packages to model a synchrotron virtual experiment. From the scientific and algorithmic point of view, OASYS rely on existing well know codes and libraries that are available thanks to the open-source community. We rely on the open-source mechanism which permits the use of valuable tools developed in synchrotron facilities and made available to the community, and at the same time guarantees due credit to the authors and supporting institutions. Although it is possible to use these codes independently, the users generally get stacked because of installation problems, definition of input parameters, decoding output files and perform data visualization. OASYS integrates in a synergetic way these computational units in a modern and performant graphical environment that facilitates all these tasks, and more important introduces the concept of interoperability or, in other words, the ability of the different units to communicate and exchange information among them.

OASYS aims to simulating the complete chain of a synchrotron experiment, from the beginning to the end. Thus, the virtual experiment can be decoupled in different steps (see Figure 1:

1. Electron beam description and propagation
2. Photon source: the creation of a photon beam using magnetic structures (insertion devices, bending magnets) placed in the electron beam
3. Beamline optics: the description and effect of the individual optical elements or components (slits, mirror, crystals, gratings, etc.)
4. The interaction with the photon beam with a sample: The detection and analysis of the scattered radiation to get information on the instrumental function (using ideal samples) or on the sample itself via comparison with experimental data.

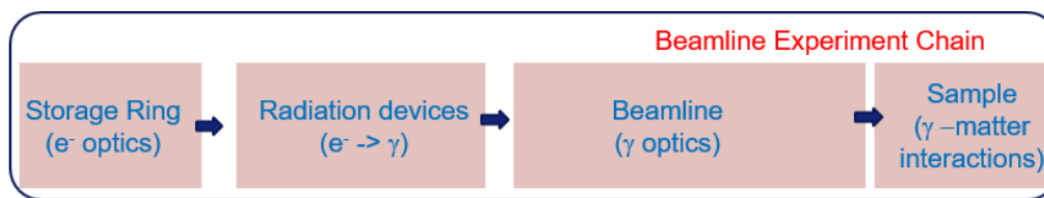


FIGURE 1. Schematic workflow of a synchrotron experiment where simulations cover the electron beam, the photon source, the beamline and finally the interaction of the photon beam with the samples.

We are conscious of the complexity, dimension and difficulty of such complete simulation in particular if one thinks on the large variety of beamlines and experimental techniques in use. At present, the power of OASYS resides in presenting a modular environment to allow simulations along the complete chain. OASYS now includes powerful tools to make simulations of points 2 (photon sources) and 3 (beamline optics). The ability to be extended to cover the electron beam characteristics and also the interaction with samples is also present and some results are obtained.

initio calculate the instrumental function of a synchrotron radiation beamline. This has been successfully tested with an X-ray powder diffraction (XRPD) beamline (Rebuffi and Scardi, 2014) used for line profile analysis, where complete control of the diffracted signal is necessary (Scardi et al., 2010). The synchrotron beam shape, divergence and energy distributions that result from the source characteristics and beamline optics contribute to broaden the diffraction peaks of the recorded diffractograms. The peak width dependence versus the 2θ angle (Caglioti et al., 1958; Sabine, 1987) is usually parameterized by Caglioti's equation (Scardi et al., 1994), where the full width at half-maximum (FWHM) of the instrumental peak profiles represented as pseudo-Voigt curves has the form $\text{FWHM}(\theta) = [W + V \tan \theta + U \tan^2 \theta]^{1/2}$, where W , V and U are Caglioti's parameters and θ is the diffraction angle.

OASYS MECHANISM (CANVAS, WIDGETS AND ADD-ONS). KEY TECHNOLOGIES.

A main feature of OASYS is to deliver a highly intuitive graphical interface provides high flexibility and rapidity for interactive simulations. OASYS presents an environment for synchrotron radiation virtual experiments. It is presented as a graphical workspace (canvas) that can be populate with applications (widgets) that that communicate among them and are picked up from menus. The applications (widgets) come from different simulation packages interfaced into OASYS and called add-ons. The different add-ons are optional packages that can be installed or uninstalled from OASYS itself depending on user's convenience. The interesting concept is that the users can add new add-ons depending on their needs. The widgets are connected via "wires" that are channels for exchanging information. The kind of information is "labelled" in such a way that two widgets can be physically connected if one is capable to emit and the other is able to receive signals that are compatible, i.e., with the same label. In this way the user builds the simulation in a form of workflow containing widgets linked by wires. Every widget has a multiple functionality: it holds the input parameters, triggers a calculation or a data flow and displays results. All these functionality is embedded in a single window that opens by double-clicking the widget. Figure 2 presents an example of the canvas and a window from a widget. The flowchart concept of the application naturally permits to chain the elements of a virtual experiment. Configurations represented by a single or multiple widgets can be copied, duplicated and changed very quickly to compare multiple configurations. OASYS integrates in a synergetic way the most powerful calculation engines available, that are summarised in the next section.

OASYS goal is to become an integration platform where different software packages can communicate by sharing information and results, allowing the user to perform multi-application, complex analyses and simulations easily and safely. To achieve this goal, a careful planning of the technologies to be used and a right selections of the software tools had been done. The boundary conditions imposed are to be open-source, work with open-source tools and be compatible with applications, libraries and packages developed in the academic context. For this reason Python has been selected as main language. It lives a very wide ecosystem with plenty of tools developed by a huge community.

The OASYS higher level workflow engine that is presented to the user is fully based on the software Orange REFERENCE, developed at the University of Ljubljana (Slovenia). Orange is a Python-based, comprehensive, component-based framework especially designed for data mining and machine learning applications. Orange is designed to simplify the assembly of data analysis workflows and crafting of data mining approaches from a combination of existing components. The Orange suite lumps together several data mining oriented services and applications into a workflow-

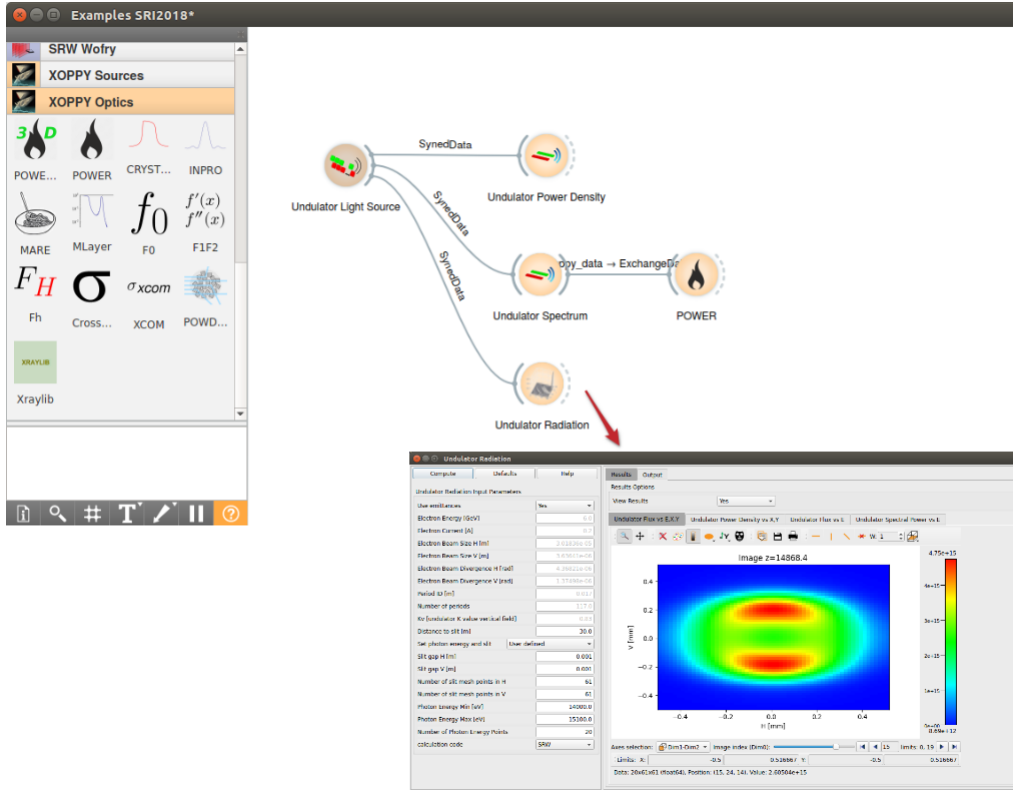


FIGURE 2. OASYS application showing on the left area the menu zone containing the installed add-ons, the canvas with some widgets and connections, and a window with the parameters and results of the “Undulator radiation” application which computes the flux of an undulators as a function of the horizontal and vertical coordinates and photon energy.

based interface that provides the interaction with the user and the communication mechanism among the different units. The OASYS higher lever interface derived from the Orange application naked from its application tools, menus and widgets. OASYS reuses the native Orange main window, canvas, and workflow infrastructure and prompts it to receive the packages we specifically developed for the synchrotron virtual experiments. In addition to Orange workflows, OASYS includes the software elements that are used by several applications (like exchangeable data, loop elements, python scripts, etc.) and installs some common dependencies used by all or several packages (add-ons). Orange widgets are based in Qt, thus all user-interface code implements Qt functionality. In OASYS all high level graphics are implemented using the library `silx` REFERENCE developed at ESRF. It contains the necessary applications for displaying 1D, 2D and 3D data and interactive tools for high-level visualization. Coming into more specific scientific applications, most synchrotron simulation codes need to access data for simulating matter-x-ray interactions. The material database for photon-matter cross sections used in OASYS is `xraylib` REFERENCE. `xraylib` provides access to some of the most cited databases of physical data in the field of X-rays. It is used to compute refraction indices, attenuation coefficients, cross sections, crystal structure factors, etc. OASYS also has `scipy` as a dependency, as it is commonly used for many scientific calculus, fitting, special functions, etc. In particular, the fundamental physical constants should not be directly coded into the applications, as some of them must be updated from time to time following the recommended values the Committee on Data of the International Council for Science (CODATA). OASYS uses the values available in `scipy.constants`.

THE BEAMLINE SIMULATION TOOLS

The first step before the construction of any X-ray system, such as a synchrotron beamline, is an accurate conceptual design of the optics. The beam should be transported to a given image plane (usually the sample position) and

its characteristics should be adapted to the experimental requirements, in terms of flux, energy bandwidth, beam divergence, focal size, time structure, etc. Today, optical design relies more and more on computer simulation and optimization. OASYS integrates different simulation strategies via the implementation of advanced simulation tools for X-ray Optics. We can classify these applications into three groups: i) toolboxes, or applications used for calculating characteristics of sources like spectra or optical elements such as material data, reflectivities, refraction indices, etc. and then simulation of the complete beam using either a ii) ray tracing approach based on geometrical optics valid for mostly incoherent X-ray beams and iii) wavefront propagation useful for beamlines where coherent effects are important and implementing physical optics methods.

The toolboxes (XOPPY and X-ray server)

The ultimate beamline modelling perform a complete simulation including every component of the beamline. For that it is necessary to define every single parameter of the source, select the appropriated insertion device, define all optical elements including dimensions and positions, etc. In other words, a quite precise beamline description is needed. However, in a first stage of the design, the scientist or engineer would like to check different possibilities of undulators, mirror coatings, monochromator crystal reflection, etc. For that OASYS makes available different small programs to calculate the individual response of each optical element and characteristics of a particular source, like emitted flux and power. This concept of toolbox has been imported from the code XOP REFERENCE developed in the past which became very popular in many synchrotron facilities. Indeed, all applications of XOP have been ported to python for being integrated into XOPPY, the python version of XOP and available as an OASYS add-on.

XOPPY lumps together several codes to simulate X-ray sources. They calculate synchrotron radiation sources (spectral emission, angular and spatial distributions, etc.) for undulators, wigglers and bending magnets. For undulators, Undulator Spectrum computes the spectral characteristics (emission versus photon energy), Undulator Power Density calculates the emission versus spatial coordinates (x,y) in a screen. In Undulator Radiation the radiation is calculated versus three independent variables (x,y, photon energy). The TC and TC-SLIT calculate tuning curves (emission versus gap variation). These applications interface well known undulator codes (US, URGENT, SRW, pySRU). Other applications calculate emission for wigglers (WIGGLER, WS), Bending Magnets (BM) and X-ray tubes (TUBE_W, TUBES). The POWER application can be attached to the source spectrum to calculate the transmission and absorption of some elements (mirrors and filters). These applications are often used to calculate the power deposited and the absorption of the elements, a require input for further finite element analysis to compute thermo-mechanical deformation of the elements. The results of these calculations in terms of surface deformation can be re-injected in OASYS for studying the degradation of the beam, using ray tracing or wave optics simulations. Other applications for reflectivity of optical elements (mirrors, crystals, multilayers) complete the XOPPY package.

Other add-ons incorporate additional functionality for OASYS. In particular for crystals: the add-on XRAY-SERVER implements a front-end of Sergey Stepanov routines available in its widely used web server REFERENCE.

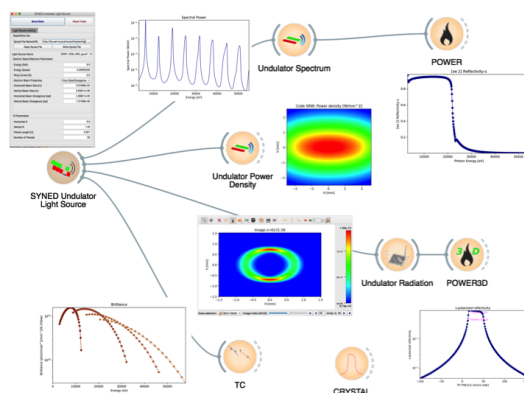


FIGURE 3. XOPPY.

Optics for incoherent optics. Ray-tracing and its environment.

The ray-tracing technique has been for decades the choice for simulating synchrotron beamlines. Most of the effects that are limiting the performances of X-ray optical elements like source emittance, aberrations, errors in the optical surfaces, etc. can be efficiently modelled by ray tracing. Some particularities of the X-ray optics as compared with visible optics makes impossible that standard commercial optics tools not adequate to synchrotron radiation simulations because, for example, they lack of a database for refraction indices in the X-ray domain, cannot handle very grazing incident optics and lack of models for perfect crystal diffraction. The code

tt shadow3 REFERENCE is available in OASYS and via the ShadowOUI add-on. This new interface for shadow3 makes a quantum gap in user-friendliness from the old interface in XOP making it more intuitive and incorporating also widgets that permit to define in a very easy way compound optical systems, like CRLs or transfocators. Moreover, ShadowOUI also contains two major new contributions that are introduced here: the Hybrid model and DABAM.

The hybrid method REFERENCE combines ray-tracing and wavefront propagation. One major application of the hybrid method is its ability to assess the effects of figure errors on the performance of focusing mirrors. The hybrid method provides accurate and quick evaluation of the expected mirror performance making it a useful tool for designing diffraction-limited focusing beamlines.

An open-source DATAbase for Metrology (DABAM) containing measured profiles of a large collection of X-ray mirrors has been created and interfaced in OASYS. It makes available metrology data (mirror heights and slopes profiles) that can be used with simulation tools for calculating the effects of optical surface errors in the performances of a synchrotron beamline. A typical case is the degradation of the intensity profile at the focal position in a beamline due to mirror surface errors. This database aims to provide to the users of simulation tools the data of real mirrors.

Optics for coherent optics. Wavefront propagation.

Towards simulations for partial coherence optics

OASYS availability, customization, extensibility

In the initial phase of OASYS development, we first focused our attention on importing python-based APIs into its environment, but recently we dedicated our efforts on its final goal: to define a uniform and exchangeable description of the real world set up that is tailored to the synchrotron world and is still flexible enough to allow particularities for different algorithms and physical approaches. This is implemented by creating an object-oriented framework library, providing the glossary for the definition of light sources and optical components, together with a set of dedicated (so-called) widgets, the active elements of the OASYS graphical user interface (GUI) [11]. This common layer upon OASYS APIs has been called SYNED (SYNchrotron Elements Dictionary), and it is the base for allowing user to create workspaces with several APIs simulating the same beamline without data misalignment or redundancy, thus separating the description of the real world from details of the calculation algorithms and allowing users to easily benchmark their calculations by using different products for the same simulation.

After implementing these common definitions and data structures, we work on the creation of a software platform able to combine different APIs, by exchanging data and calculation results. The final purpose is to optimize running different calculations and making comparisons of the results. The first integration framework library we developed is dedicated to wave optics and it is called WOFRY (Wave Optics FRamework in pYthon), a generalization (or abstraction) of a wavefront propagation software tool, combined with the SYNED definition of a synchrotron radiation beamline.

Existing OASYS API like ShadowOui [11] and the wavefront propagation tool WISE [12] and XOPPY (XOP [13] in PYthon) has been refactored to be compatible with SYNED. The SRW OASYS prototype API has been implemented according to both WOFRY and SYNED frameworks. UML [14] class diagrams, implementation details and examples of both framework are shown in the following paragraphs.

The ultimate purpose of OASYS is to integrate in a synergetic way the most powerful calculation engines available to perform virtual experiments in a synchrotron beamline, from the electron emission to the sample scattering. For X-ray Optics, OASYS integrates different simulation strategies via the implementation of adequate simulation tools (e.g. ray tracing and wave optics packages). It provides a language to make them to communicate by sending and receiving encapsulated data.

OASYS is a modern and user-friendly graphical environment for modelling X-ray experiments, where several top-level optical simulation tools have already been successfully integrated. We showed by examples how the OASYS

environment allows the users to perform complex calculations in an intuitive and easy way, increasing the level of affordability of the optical design. OASYS and the available APIs (Add-Ons) is continuously developed since 2013 and upgraded both in term of number of functionalities and in robustness.

Wave optics methods are computationally more expensive, as usually one has to finely grid the phase space. Hybrid methods permitting to switch from one description to another and vice versa would be ideal. The present trend [1] is the co-existence of software tools that allow treating the same system from two points of view. The integration of these two approaches into a single computer environment is a challenge that will certainly be developed in the near future. This paper describes the recent developments in ray tracing, mainly in connection with the SHADOW [2] code. At the end of the paper, some ideas of integration of ray tracing and wave optics are discussed, and some collaborative effort is on progress.