WavePropaGator: interactive framework for X-ray free-electron laser optics design and simulations[1]

Liubov Samoylova,[a]* Alexey Buzmakov,[b]* Oleg Chubar[c] and Harald Sinn[a]

[a]European XFEL GmbH, Albert-Einstein-Ring 19, Hamburg, 22761, Germany, [b]Institute of Crystallography, Leninskii prospekt 59, Moscow, 119333, Russian Federation, and [c]National Synchrotron Light Source II, Brookhaven National Laboratory, Upton, NY 11973, USA. *Correspondence e-mail: liubov.samoylova@xfel.eu, buzmakov@gmail.com

This article describes the WavePropaGator (WPG) package, a new interactive software framework for coherent and partially coherent X-ray wavefront propagation simulations. The package has been developed at European XFEL for users at the existing and emerging free-electron laser (FEL) facilities, as well as at the third-generation synchrotron sources and future diffraction-limited storage rings. The WPG addresses the needs of beamline scientists and user groups to facilitate the design, optimization and improvement of X-ray optics to meet their experimental requirements. The package uses the Synchrotron Radiation Workshop (SRW) C/C++ library and its Python binding for numerical wavefront propagation simulations. The framework runs reliably under Linux, Microsoft Windows 7 and Apple Mac OS X and is distributed under an open-source license. The available tools allow for varying source parameters and optics layouts and visualizing the results interactively. The wavefront history structure can be used for tracking changes in every particular wavefront during propagation. The batch propagation mode enables processing of multiple wavefronts in workflow mode. The paper presents a general description of the package and gives some recent application examples, including modeling of full X-ray FEL beamlines and start-to-end simulation of experiments.

1. Glossary

FEL: free-electron laser.

FFT: fast Fourier transform.

Git: a distributed revision control and source code management system (http://git-scm.com).

GitHub: a web-based Git repository hosting service (https://github.com/).

HDF5: Hierarchical Data Format. To view HDF5 files one can use the official free HDFView tool (https://www.hdfgroup.org/products/java/hdfview).

IPython Notebook: an interactive computational environment for combining code execution, rich text, mathematics, plots and rich media (http://ipython.org/notebook.html).

MPI: Message Passing Interface. A standardized and portable message-passing system designed by a group of researchers from academia and industry to function on a wide variety of parallel computers (http://www.mpi-forum.org/).

SASE: self-amplified spontaneous emission. The process whereby an electron beam passes through a long undulator, such that the initial random field of spontaneous radiation becomes amplified in intensity and enhanced in coherence characteristics. SASE FEL pulses are characterized by spiky structure and fluctuations in wavelength and FEL pulse energy.
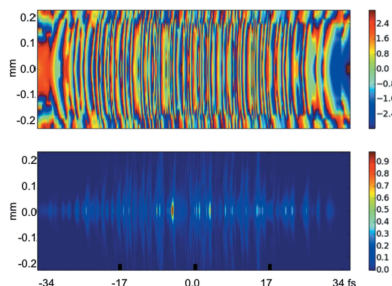
2. Introduction

The X-ray free-electron laser facilities (XFELs) that have emerged in recent years deliver radiation with a unique combination of properties, such as extreme peak field intensities, ultra-short X-ray pulse duration and a high degree of transverse coherence of the XFEL radiation, and thus establish a new specific combination of requirements for X-ray optics. Additional factors such as the varying beam size and divergence as a function of the electron bunch charge, the very large distances to the source, and, as a consequence, the large apertures required for the optics further complicate the environment.

Besides, the end stations at XFEL facilities have very versatile layouts, which can include multi-crystal monochromators, split-and-delay lines and units for pump–probe experiments (Roling *et al.*, 2014), monochromators for hard and soft X-ray self-seeding schemes (Amann *et al.*, 2012; Inagaki *et al.*, 2014), and various refocusing optics. For a correct optics design, one has to analyze and optimize all the factors that have an impact on the beam properties after propagation of the beam through the optics. Hard X-ray FEL wavefronts are subject to distortions caused by imperfections of the optics, in particular the effect of mirror surface errors, which have been measured experimentally at LCLS and SACLA (Rutishauser *et al.*, 2012; Kayser *et al.*, 2014). Moreover, characterizing focused XFEL intense wavefields is crucial for their use in single-shot diffractive imaging experiments (Loh *et al.*, 2013; Park *et al.*, 2013).

The wave optics formalism is a natural and reliable way to deal with interference effects that are unavoidable in coherent beam propagation (Bahrdt *et al.*, 2014; Chubar *et al.*, 2011). We present here the *WavePropaGator* (*WPG*) package, the goal of which is to provide a user-friendly software environment that any scientist involved in XFEL optics design can use, without having specific experience in numerical wave optics calculations. The package is built on top of the open-source *Synchrotron Radiation Workshop* (*SRW*) software (Chubar *et al.*, 2011), which is actively used for development of X-ray and infrared beamlines at synchrotron light sources. The *SRW* code is based on the principle of physical optics using FFT and asymptotic expansion based propagators and is capable of accurate modeling of fully and partially coherent synchrotron radiation emission and propagation through different types of optics. The software is able to calculate the propagation of SASE XFEL pulses, which requires much larger computing resources than propagation of steady state beams or longitudinally coherent pulses.

For the *WPG* framework we use Python structures, which provide the possibility for coding in self-explaining metalanguage, so that a complete beamline with tens of optical elements can be specified within one page of text. One can easily create and modify such scripts without going into implementation details. In addition, an experienced user can create propagators for new optics types in Python and include them in the beamlines. The *WPG* framework scripts can be run either within the *IPython Notebook* interactive environment, combining code execution, rich text, mathematics and plots in one file, or in batch file mode, as a Python script in Python or an IPython shell. The *WPG* also includes modules necessary for start-to-end simulations of XFEL experiments, which simulate propagation of the XFEL pulses (Manetti *et al.*, 2016) through beamline optics and provide an interface to codes modeling the interaction with the sample (*SingFEL*, *PMI*; Yoon *et al.*, 2016).

3. Coherent wavefront propagation

This section summarizes the basics of the calculation methods implemented in the *SRW* (Chubar & Elleaume, 1998; Chubar *et al.*, 2002, 2008, 2011) that are most relevant for FEL applications and are available through the *WPG* interface.

For small emission and observation angles, the propagation of the transverse components in free space from a point r_1 to a point r_2 can be described in terms of the Huygens−Fresnel principle in the frequency domain as a integral by integration over the plane Σ_1 perpendicular to the z axis (beam axis):

$$E_\perp(r_2, \omega) \simeq -\frac{i\omega}{2\pi c} \iint\limits_{\Sigma_1} E_\perp(r_1, \omega) \frac{\exp(-i\omega|r_2 - r_1|/c)}{|r_2 - r_1|} \, d\Sigma_1,$$

(1)

where ω is the angular frequency of the wave and c the speed of light. The wavefield at the observation plane can be written in a more general form as

$$E_\perp(x_2, y_2, \omega) \simeq \int\int dx_1 \, dy_1 \, \boldsymbol{K}(x_2, y_2, x_1, y_1, \omega) \, E_\perp(x_1, y_1, \omega),$$

(2)

where the kernel $\boldsymbol{K}(x_2, y_2, x_1, y_1, \omega)$ is given as

$$\boldsymbol{K}(x_2, y_2, x_1, y_1, \omega) \simeq -\frac{ik}{2\pi z} \exp\left\{\frac{ik}{2z}\left[(x_2 - x_1)^2 + (y_2 - y_1)^2\right]\right\}.$$

(3)

As shown in the following two sections, propagation through most X-ray optics elements can be presented in the form of the convolution integral (2). Using the Fourier optics modular approach, a complete beamline can be described as a set of propagators corresponding to individual optical components, which can be then numerically solved by means of a two-dimensional FFT algorithm.

The representation (3) also enables the analytical treatment of the quadratic phase term, allowing for free-space propagation simulations with a reduced sampling rate (Chubar *et al.*, 2008). This dramatically improves the technical feasibility and robustness of the Fourier optics method.

3.1. Thin optical elements

Many of the optical components of an X-ray beamline can be described as thin optical elements, *i.e.* as linear filters that change the wavefield amplitude and/or phase in the plane normal to the propagation direction. For thin optical elements the kernel (3) can be represented as

$$\boldsymbol{K}(x_2, y_2, x_1, y_1, \omega) = \boldsymbol{T}(x_1, y_1, \omega) \, \delta(x_1 - x_2) \, \delta(y_1 - y_2), \quad (4)$$

Table 1
Optical elements (propagators) available in the *WPG*.

The parameters are shown in ***bold italic*** and the names of the propagators in *italic*.

Optics	Propagator	Parameters	Comments
Free space	*Drift*	***Propagation distance***	In most cases the semi-analytical propagation is recommended, see §3.1.
Slits, apertures	*Aperture*	***Slit or obstacle***, ***shape*** (rectangular or circular), ***slit width***, ***height***, ***slit center coordinates***	For a circular aperture only slit width is used as aperture diameter.
Grazing-incidence plane elliptical mirror	*Mirror_elliptical*	***Orientation***, ***p*** and ***q***, distances to source and focus, ***incidence angle*** in the mirror center, ***misalignment angle***, ***mirror length***	See §3.3 for details.
Thin lens	*Lens*	***Focal lengths*** in horizontal and vertical plane, and ***lens center coordinates***	Can be used to model sagittal bending in a plane elliptical mirror or to approximate focusing with a spherical mirror.
Mirror surface error†	*WF_dist*	***Mesh*** and ***dimensions***	Phase screen approach is used to introduce wavefront distortions caused by residual surface height errors (Samoylova *et al.*, 2009).
Grazing-incidence VLS grating	*VLS_grating*	***Substrate*** (a mirror propagator), ***diffraction order number m***, ***grove density g_0*** in the grating center, ***g_k***, ***k = 1, ..., 4***, ***grove density polynomial coefficients***: $g = g_0 + g_1 x + g_2 x^2 + g_3 x^3 + g_4 x^4$	See §3.3 for details.
Crystal monochromator	*Xtal*	***Interplanar spacing*** of the selected Bragg reflection H, ***Fourier components of electric susceptibility***,‡ crystal ***thickness***, ***asymmetry angle***, ***deviation of the incidence angle*** from the rocking curve center	
CRLs	*CRLs*	Obligatory parameters: ***which focal plane*** (1 for horizontal, 2 for vertical, 3 for both), ***δ***, ***attenuation length***, ***shape*** (parabolic or spherical), horizontal and vertical ***aperture diameters***, ***tip radius***, ***number*** of CRLs (holes), ***lens thickness at apex***; optional parameters: a flat array/list of ***void center coordinates and radii***: [$x_1, y_1, r_1, x_2, y_2, r_2, ...$], ***initial photon energy***, ***final photon energy***	See also §3.3 on modeling voids in CRLs. Note: the energy range parameters are used only if the attenuation length and δ are arrays. Then the dispersion effect of CRLs is taken into account.

† The propagator should be used together with the `calculateOPD()` function, see Fig. 3.   ‡ For *0, H* and *−H* reciprocal lattice nodes for reflection *H*.

where $T(x_1, y_1, \omega)$ is a complex transmission function, and $\delta()$ is the δ function. For example, the transmission function for a thin lens is given by

$$T(x, y, \omega) = \exp\left\{-ik\left[\frac{(x-x_0)^2}{2f_x} + \frac{(y-y_0)^2}{2f_y}\right]\right\}, \quad (5)$$

where (x_0, y_0) are coordinates of the lens center, and f_x and f_y are focal distances in the horizontal and vertical planes. The transmission for a two-dimensional parabolic compound refractive lens (CRL) (Kohn *et al.*, 2003) is given by

$$T(x, y, \omega) = \exp\left\{-ik[\delta(\omega) - i\beta(\omega)]N(r/R + d)\right\}, \quad r < a, \quad (6)$$

where the complex refractive index of the lens material is $n = 1 - \delta + i\beta$, the CRL thickness depending on the distance from the lens center $r = (x^2 + y^2)^{1/2}$ is $t = N(r/R + d)$, R is the radius of the parabola tip, N is the total number of individual parabolic lenses, d is the minimum spacing between two parabolas and a is the lens aperture.

3.2. Extended optical elements

The wave optics method that we use for simulation of grazing-incidence mirrors is based on the local stationary-phase approximation (Canestrari *et al.*, 2014). The kernel is presented as

$$\begin{aligned}K(x_2, y_2, x_1, y_1, \omega) \cong {} & G(x_1, y_1, \omega) \\ & \times \exp\left[(i\omega/c)\,\Lambda(x_2, y_2, x_1, y_1, \omega)\right] \\ & \times \delta[x_1 - \tilde{x}_1(x_2, y_2)]\,\delta[y_1 - \tilde{y}_1(x_2, y_2)], \quad (7)\end{aligned}$$

where G is a matrix function defining local transformations of amplitudes of electric field components, *i.e.* the propagation between input (x_1, y_1) and output (x_2, y_2) planes including the reflection from a mirror surface; $\tilde{x}_1(x_2, y_2)$ and $\tilde{y}_1(x_2, y_2)$ are scalar functions defining the transformation of coordinates for points in transverse planes before and after the optical element; and $\Lambda(x_2, y_2, x_1, y_1, \omega)$ is a scalar function defining the optical path between the points in the input and output planes. These functions can be found by using the asymptotic expansion of the Fresnel−Kirchhoff integral over the mirror surface, namely, the stationary phase approximation, which is Fermat's principle for mirror reflection expressed in mathematical form. The stationary phase point can be found by defining a ray from the point (x_1, y_1) in the input plane in the direction provided by the local gradient of the input radiation field phase, finding its intersection with the mirror surface, generating the specular reflected ray, and finding its intersection with the output plane at the point (x_2, y_2) behind the mirror. As a final step, the electric field components are re-calculated on the required rectangular mesh, using a two-dimensional interpolation.

For gratings (Canestrari *et al.*, 2014), an extra phase shift $\Delta\Phi(x_2, y_2, \omega)$ associated with the reflection should be taken into account:

$$\begin{aligned} \boldsymbol{K}(x_2, y_2, x_1, y_1, \omega) &\cong \boldsymbol{G}(x_1, y_1, \omega) \\ &\times \exp\big[(i\omega/c)\Lambda(x_2, y_2, x_1, y_1, \omega) + \Delta\Phi(x_2, y_2, \omega)\big] \\ &\times \delta[x_1 - \tilde{x}_1(x_2, y_2)]\,\delta[y_1 - \tilde{y}_1(x_2, y_2)]. \end{aligned} \quad (8)$$

This function describes the phase shift introduced by grooves in the transverse direction, the shift being perpendicular to the grooves in the output plane:

$$\Delta\Phi(x_2, y_2, \omega) = -2\pi m \tilde{N}(x_2, y_2, \omega), \quad (6)$$

where m is the diffraction order, and $\tilde{N}(x_2, y_2, \omega)$ is the effective groove number for a given intersection point. For a variable line spacing (VLS) grating with a groove density that varies along the 'longitudinal' direction of the grating surface according to the nth order polynomial $\rho(\tilde{x}) = \sum_{k=0}^{n} a_k \tilde{x}^k$, the effective number of grooves associated with the longitudinal position $\tilde{x}(x_2, y_2, \omega)$ on the grating surface is

$$\tilde{N}(x_2, y_2, \omega) = \int_0^{\tilde{x}} \rho(\tilde{x})\,\mathrm{d}\tilde{x} = \sum_{k=0}^{n} \frac{a_k}{k+1} \tilde{x}^{(k+1)}(x_2, y_2, \omega). \quad (10)$$

Propagation of the XFEL pulses through a perfect diffracting crystal can be described within the Fourier optics approach (Bushuev, 2008). For that, the crystal propagator known from dynamical diffraction theory is applied to wavefield Fourier amplitudes in reciprocal space. Afterwards the resulting wavefront in real space can be obtained with an inverse Fourier transformation. For implementation details see Sutter *et al.* (2014); the first practical applications of these *SRW* modules were described by Suvorov *et al.* (2015) and Chubar *et al.* (2016).

The available optical elements are listed in Table 1.

3.3. Imperfections in optics

High-performing X-ray mirrors must be able to preserve the wavefront of the incident radiation inside the focused spot on the sample. Deviation of the mirror surface from an ideal one results in the appearance of scattered waves, which form speckles (irregularities of the radiation intensity caused by irregularities of the wavefront) in the beam spot downstream. In particular, the XFEL radiation divergence is only several microradians, and the source-to-mirror, r_0, and the mirror-to-sample, r_1, distances range up to several hundred metres. As a result, the mirror surface errors at the longest spatial wavelengths, comparable to the mirror length, have the greatest effect on the quality of the reflected beam. The spatial frequencies responsible for scattering radiation inside the spot (damaging frequencies) for a flat mirror, *e.g.* horizontal offset mirrors at the European XFEL, can be estimated as (Yashchuk *et al.*, 2015)

$$\frac{\theta_0}{r_0 \delta\theta} < \nu_x < \frac{\theta_0 \delta\theta}{2\lambda}\left(1 + \frac{r_0}{r_1}\right), \quad (11)$$

where θ_0 is the incidence angle, $\delta\theta$ is the radiation divergence and λ is the X-ray wavelength. With the parameters of the SASE1 beamline at the European XFEL, the range of spatial frequencies that can produce speckles is $\nu_x \simeq 0.033$–0.3 cm^{-1}. These frequencies correspond to the spatial lengths $d_x \simeq 3.3$–30 cm. Lower spatial frequencies (spatial lengths longer than 30 cm) would lead to splitting of the beam. With the parameters of the SASE3 beamline, the range of spatial frequencies that will generate speckles is $\nu_x \simeq 0.02$–1.5 cm^{-1}. These frequencies correspond to the spatial lengths $d_x \simeq 0.67$–50 cm.

If the coherent X-ray beam is reflected at grazing angle θ from a very smooth mirror surface with a height profile $h(x_\mathrm{m})$, the optical path differences introduced by the mirror's imperfections are much smaller than the X-ray wavelength λ:

$$(4\pi/\lambda)\big|h(x_m)\big|\sin\theta \ll 1 \quad \forall m, \quad (12)$$

where x_m is the coordinate along the mirror surface and m is the point number. In this case one can use the complex transmission function

$$\boldsymbol{T}(x, y) = \exp[(-i4\pi/\lambda)\,h(x, y)\sin\theta] \quad (13)$$

to analyze wavefront distortions induced by mirror surface residual height errors. Owing to the very small incidence angle the beam footprint is much larger along the propagation direction than in the sagittal direction. If the full width at half-maximum of the XFEL beam is about 0.5 mm, for a typical incidence angle 2 mrad, the beam footprint is about 25×0.5 mm. Since the impact of sagittal errors will be small, one can use here only one-dimensional metrology data to model the surface imperfections along the beam propagation direction.

The propagator of an imperfect CRL is simulated in a two-step approach. First, it uses the propagator for an ideal CRL as given in §3.2. Then, voids are simulated respecting a given void size distribution between a minimum and a maximum void diameter and are distributed in accordance with a given void density into a fictive cylinder of the same material, having the same aperture and thickness as the ideal CRL. In the next
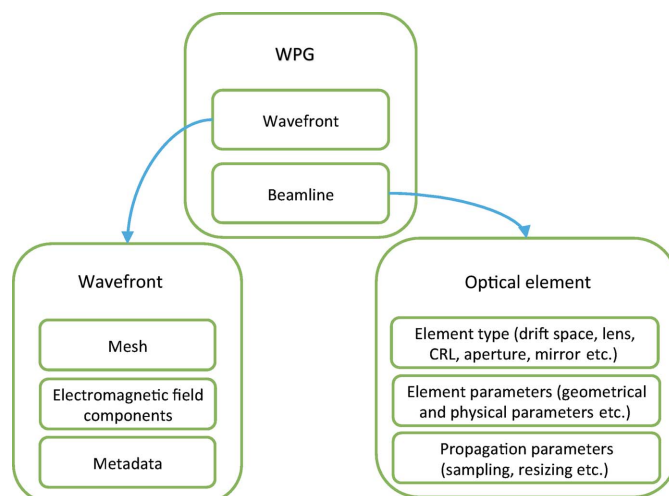

Figure 1
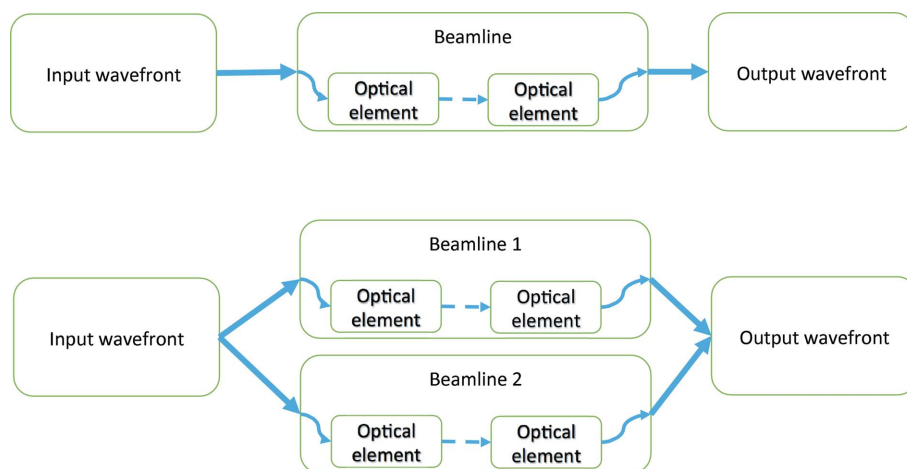General structure of the *WPG* framework.

Figure 2
Workflow chart: a wavefront propagates through a beamline, which is set up as a container of propagators. Top: a simple 'linear' workflow for a regular beamline; bottom: beamline with two branches, *e.g.* XFEL split-and-delay lines.

can be calculated only once, saved in a binary format supported by Python and later loaded from the hard drive. See also Roth *et al.* (2014), example 7 (https://github.com/ochubar/SRW), and the *WPG* tutorial (http://wpg.readthedocs.org/en/latest/tutorials.html).

3.4. Wavefront propagation setup

In the *WPG* framework the model of wavefront propagation is implemented using two classes, `Wavefront()` and `Beamline()` as shown in Fig. 1. To start the propagation one should define a wavefront, which can be a Gaussian steady state or time-dependent short pulse calculated using corresponding library functions, or an external XFEL pulse, *e.g.* from the X-ray FEL Photon Pulses Database (XPD; Manetti *et al.*, 2016), or any other external wavefield defined on a uniform Cartesian grid.

The beamline is defined as a container of propagators with suitable parameters. The basic workflow is shown in Fig. 2.

3.4.1. Wavefront data. The structure of the HDF5 wavefront file, which is described in glossary file `wpg/glossary.py`, is used for mapping the Python wavefront structure to HDF5.

step, the paraboloids that are defining the CRL shape are removed, including the voids there. Finally, a plain void propagator is generated with the corresponding transmission. The passage of the wavefront through an imperfect CRL is thus calculated with two propagators, first for the ideal CRL, followed by the plane void propagator. To speed up the calculations the propagators for perfect and imperfect CRLs

```python
import wpg.optical_elements
from wpg import Beamline
from wpg.optical_elements import Empty, Use_PP
from wpg.optical_elements import Drift, Aperture
from wpg.optical_elements import Lens,Mirror_elliptical,WF_dist,calculateOPD
bl.append(Aperture(shape='r',ap_or_ob='a',Dx=lengthOM*thetaOM,Dy=range_xy),
        Use_PP(zoom=1., sampling_h=1./0.5, sampling_v=1.,semi_analytical_treatment=0))
bl.append(Mirror_elliptical(orient='x',p=z_M2, q=q_M2,thetaE=thetaOM, theta0=thetaOM,
                    length=lengthOM), Use_PP(semi_analytical_treatment=1))
wf_dist_m2 = WF_dist(1500,100,range_xy,lengthOM*thetaOM, )
calculateOPD(wf_dist_m2, os.path.join(mirror_data_dir,'mirror1.dat'),2,'\t','y',thetaOM, scale=5)
bl.append(wf_dist_m2,Use_PP())
bl.append(Drift(z_M3-z_M2),Use_PP(semi_analytical_treatment=0));
bl.append(Mirror_elliptical(orient='y',p=z_M3, q=q_M3,thetaE=thetaOM, theta0=thetaOM,length=lengthM3),
        Use_PP(semi_analytical_treatment=1))
wf_dist_m3 = WF_dist(100, 1500, range_xy, lengthM3*thetaOM, )
calculateOPD(wf_dist_m3, os.path.join(mirror_data_dir,'mirror2.dat'),2'\t''y',thetaOM, scale=5)
bl.append(wf_dist_m3,Use_PP())
bl.append(Drift(z_focus_M2-z_M3),Use_PP(semi_analytical_treatment=0));
width = 50.e-6    # slit width
dz_blades = 30e-2 # distance between blades
bl.append(Aperture(shape='r',ap_or_ob='o',Dx=50e-3,Dy=50e-3,x= (50e-3/2+width/2),y=0),Use_PP())
bl.append(Drift(dz_blades),Use_PP(semi_analytical_treatment=0));
bl.append(Aperture(shape='r',ap_or_ob='o',Dx=50e-3,Dy=50e-3,x=-(50e-3/2+width/2),y=0),
        Use_PP(zoom_h=0.9,sampling_h=0.9/1.0))
bl.append(Drift(z_M3-z_focus_M2),Use_PP(zoom_h=2.4,sampling_h=2.4/0.4));
bl.append(Aperture(shape='r',ap_or_ob='o',Dx=50e-3,Dy=50e-3,x=0,y=(50e-3/2+width/2)),Use_PP())
bl.append(Drift(dz_blades),Use_PP(semi_analytical_treatment=0))
zz = zz + dz_blades
bl.append(Aperture(shape='r',ap_or_ob='o',Dx=50e-3,Dy=50e-3,x=0,y=-(50e-3/2+width/2)),Use_PP())
bl.append(Drift(z_focus_M2-z_focus_M3-dz_blades),Use_PP(zoom_v=1.4,sampling_v=1.4/0.5))
print bl; bl.propagate(wf)
```

Figure 3
Example of a beamline definition: the SASE3 beamline at the European XFEL will include two horizontal offset mirrors (M1 and M2), a vertical focusing mirror M3, and horizontal and vertical clean-up slits.

The *WPG* wavefront HDF5 file contains the obligatory and optional sections described below.

The main groups include the following:

(*a*) *data* – contains two three-dimensional arrays of electromagnetic fields with horizontal and vertical polarizations, `arrEhor` and `arrEver`

(*b*) *params* – contains a Cartesian grid of wavefront data, geometrical parameters *etc.*

The following are optional groups, especially recommended for use in start-to-end simulations:

(*a*) *history* – contains a link to a parent wavefront, if it exists, and a hierarchy structure with parent wavefront parameters and a link to its data

(*b*) *info* – some information about the origin of the current wavefront

(*c*) *misc* – information for visual checking of wavefront parameters (intensity distribution *etc.*)

(*d*) *version* – version of the wavefront glossary

3.4.2. Beamline as a container of propagators. An example of the beamline definition is shown in Fig. 3.

3.4.3. Adjustment of propagation parameters. Our framework provides special tools to facilitate propagation parameter adjustment and make the process transparent for the user. In particular, the method `Use_PP()` takes none or several propagation parameters defined by the user for a given optical element as arguments (see Fig. 3) and provides the core library propagation function with a full propagation parameter set, using for the rest the default values. For instance, `bl.append(Drift(50.),Use_PP(semi_analytical_treatement=1))` introduces a 50 m drift in free space and sets up calculations using the semi-analytical processing described in §3. To tune the sampling, one can use the parameters `zoom` and `sampling` (Fig. 3). As in all numerical methods involving discrete Fourier transforms, the correct sampling of the signal in real and
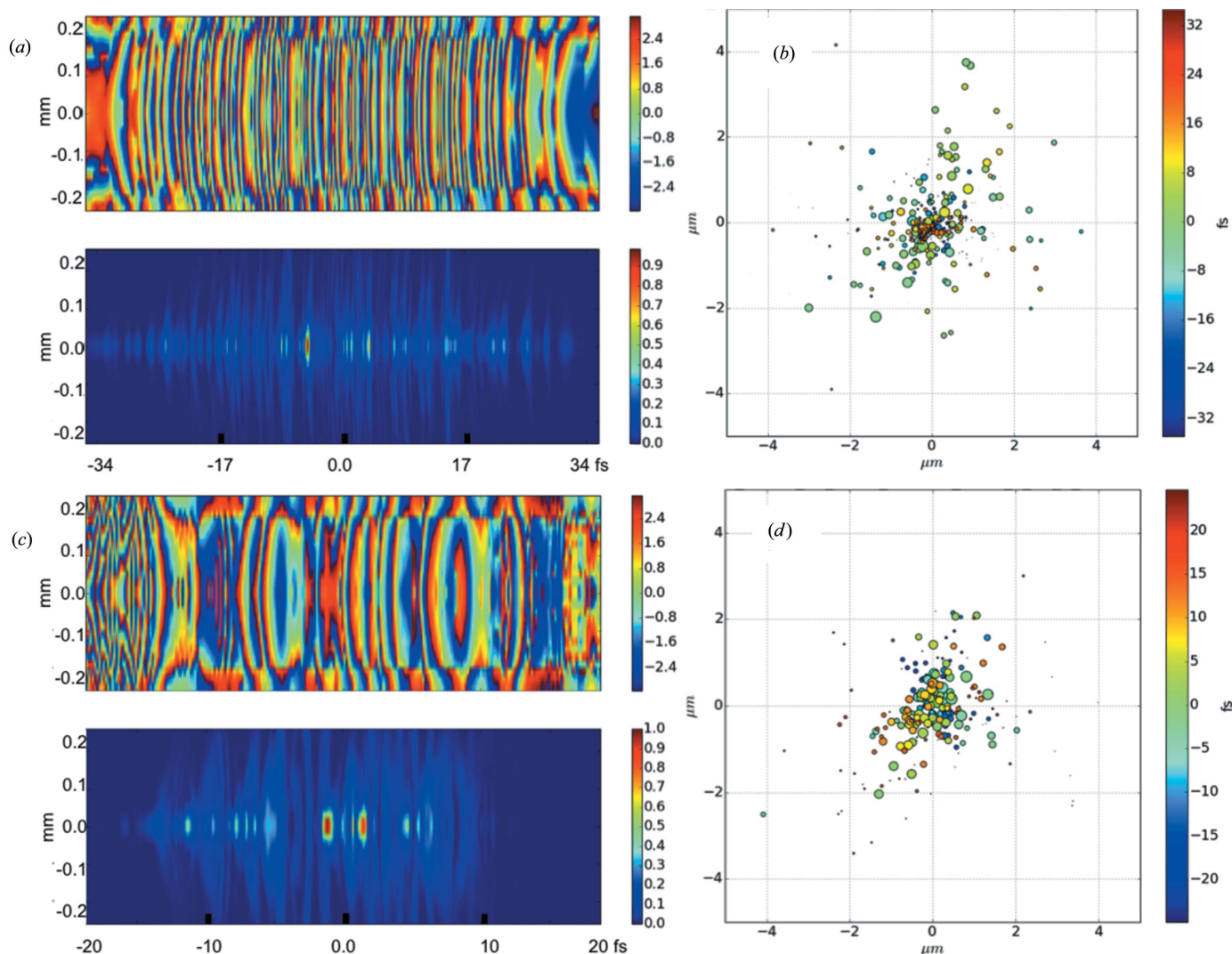

Figure 4
An example of three-dimensional wavefront visualization. For the SASE3 undulator with a defined maximum active segment length of 130 m, the simulated XFEL pulse data taken from the XPD (Manetti *et al.*, 2016) are used: photon energy 800 eV, electron bunch charge 250 pC and electron energy 14 GeV. The top row corresponds to an untapered undulator and pulse energy 8 mJ, and the bottom row to an undulator with optimized tapering (Schneidmiller & Yurkov, 2016) and pulse energy 17 mJ. (*a*), (*c*) Vertical cuts of the pulse phase and intensity; (*b*), (*d*) slice-to-slice shift of the pulse center of mass. The slice times are color coded and the intensities are represented by the size of the circles.

reciprocal space is critical for correct numerical calculations (see *e.g.* Potter, 1973; Goodman, 2004). To control the choice one can refer to the sampling in reciprocal Q space: the angular spectrum can be visualized using corresponding library utilities. See also the *WPG* documentation section with FEL beamline examples (http://wpg.readthedocs.org/en/latest/real_beamlines.html).

4. Examples

4.1. FEL pulse visualization

Fig. 4 shows the intensity and phase distribution of an XFEL pulse extracted from the XPD (Manetti *et al.*, 2016) and visualized by the *WPG* package at the exit of the SASE3 undulator with a maximum length of active segments of 130 m. The FEL data were simulated with the *FAST* code (Saldin *et al.*, 1999) for photon energy 800 eV, electron bunch charge 250 pC and electron energy 14 GeV. Note that the shift is much smaller than the spot size. One can easily observe that the pulse after the tapered undulator is shorter and more collimated.

4.2. Modeling the SCS SASE3 beamline

Fig. 5 shows the intensity distribution around the sample position of the SCS beamline at the European XFEL. The optical elements considered in this example are offset and distribution mirrors of the beam transport, vertical mirrors of the soft X-ray monochromator, clean-up slits in the vertical and horizontal intermediate foci, and Kirkpatrick–Baez micro-focusing mirrors close to the sample position.

Cut-off effects of all optical elements were considered, assuming here an idealized Gaussian beam profile as a source with a divergence corresponding to XFEL SASE3 radiation from electron bunch charge 100 pC and electron energy 17.5 GeV. The simulations took into account a slightly deteriorated performance of the beamline mirrors with respect to their specifications. The left panel of Fig. 5 shows the beam intensity around the focus position with the clean-up slits fully open. The vertical cuts correspond to (left to right) −6, 0 and 6 mm positions around the focus. Asymmetric wings are visible in the horizontal plane, while the focus size is slightly enlarged in the vertical plane. By closing the clean-up slits (right panel), the focus becomes more homogeneous and smaller in the vertical plane. As demonstrated in this example, clean-up slits can in principle be used to reduce the effect of profile distortions from upstream mirrors without losing too much intensity. However, damage effects on these slits have to be carefully monitored and make this equipment usable only for moderate X-ray beam intensities.

4.3. Module for start-to-end simulation of XFEL experiments

The *prop* module numerically propagates the FEL wave-field data from the undulator exit, through the beamline optics, to the sample position using the Python module *multiprocessing*. Each file from the working directory is processed by one free CPU core. On modern PCs or servers this makes it possible to significantly speed up the calculations.
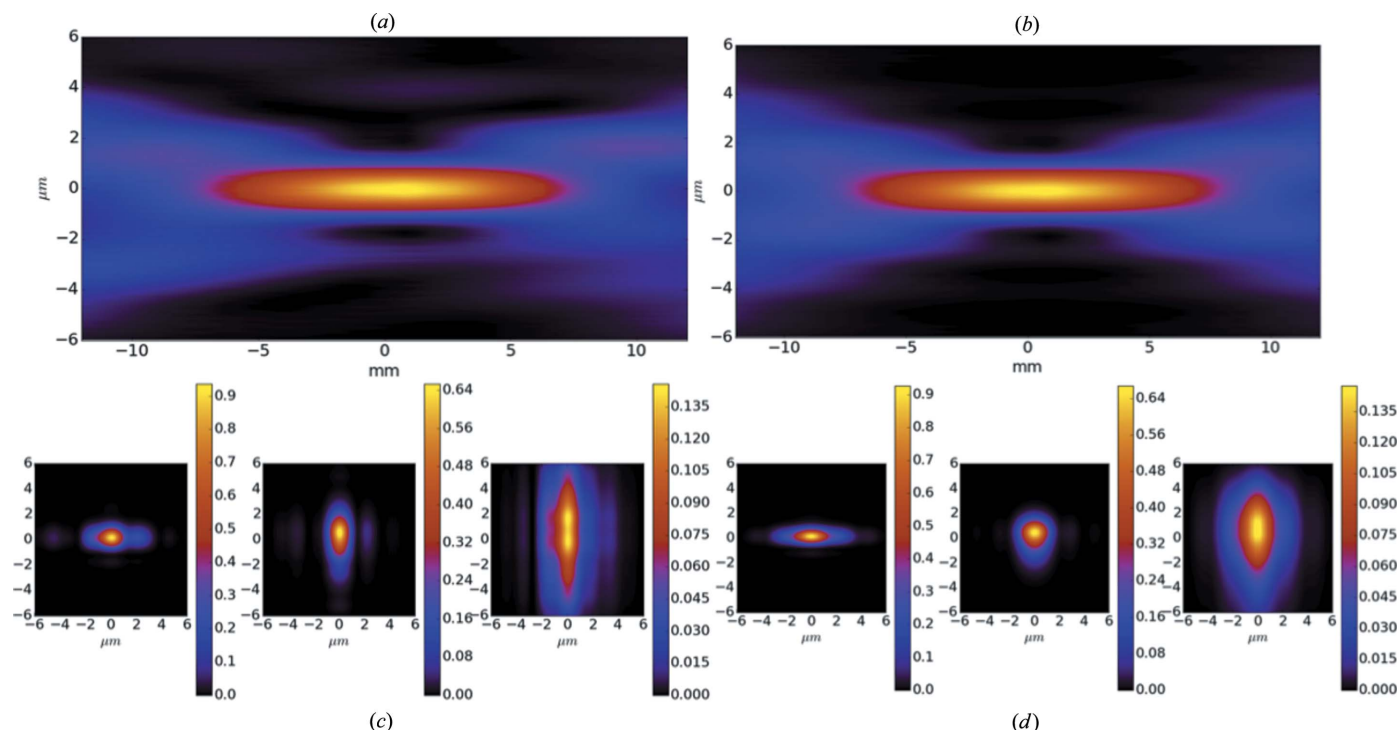
Figure 5
Intensity distribution around the sample position of the SCS instrument. Left panel (*a*), (*c*): the clean-up slits in the vertical and horizontal focus are fully open. Right panel (*b*), (*d*): both clean-up slits are closed to a gap of 50 μm. The cuts perpendicular to the optical axis in (*c*) and (*d*) correspond to −6, 0 and 6 mm positions around the focus.

However the memory load by every process can become a bottleneck: the required memory for processing of a short 30 fs XFEL pulse can exceed 8 Gb for a beamline including submicrometre focusing optics, such as the Kirkpatrick–Baez mirror system of the SPB-SFX instrument at the European XFEL (Bean *et al.*, 2016). The module was successfully used for modeling of a single biomolecule imaging experiment at the SPB instrument with the multiphysics simulation framework *simS2E* (Yoon *et al.*, 2016).

5. Software availability and documentation

The *WPG* framework runs reliably under Linux, Microsoft Windows 7 and Apple Mac OS X. Using IPython as a web front-end enables the users to run the code on a remote server as well as on their local personal computers. One can use popular Python libraries [such as *SciPy* (https://www.scipy.org/), *NumPy* (http://www.numpy.org/) and *Matplotlib* (http://matplotlib.org/)] for pre- and post-processing as well as for visualizing the simulation results.

The wavefronts are saved in HDF5 format for eventual further processing and start-to-end simulations of experiments. The HDF5 format allows for keeping the calculation history within a single file, thus facilitating communication between various scientific groups and cross-checking with other simulation results. The *WPG* source code (https://github.com/samoylv/WPG) together with guidelines for installation and application examples (https://wpg.readthedocs.org/en/latest/) are open source and available on the web. The installation includes building the C++ written *SRW* library from its sources.

The installation procedure for Linux, OS X and Windows 7 is described at https://wpg.readthedocs.org/en/latest/wpg.html#getting-started and includes installation of Python, *Numpy*, *Scipy*, *Matplotlib*, *h5py* (http://www.h5py.org/) and the *IPython Notebook*.

6. Future steps

In the near future, we plan to enhance the start-to-end simulation possibilities by employing the MPI technology through the *mpi4py* module and running the propagation of multiple pulses on a cluster. One should, though, keep in mind that transferring the input and resulting pulse data through the network can also become a bottleneck. Thus the efficiency strongly depends on the cluster configuration and the ratio of calculation time for one pulse and network data transfer rate. In addition, we plan for the *WPG* to support full interoperation with a new framework, based on client–server architecture and using JavaScript and Python for a rich graphical user interface. This is currently under development in a project (Bruhwiler *et al.*, 2014) sponsored by the US Department of Energy.

7. Conclusions

Knowledge of temporal, spatial, spectral and coherence properties of the radiation from X-ray FELs is of key importance for planning user experiments. We have developed the *WPG* software package as a new framework for wave optics simulations and have used it successfully for X-ray FEL optics beamline design and experimental data analysis. The software is also used as a part of the multiphysics simulation framework *simS2E* for source-to-experiment simulations of a single-particle imaging experiment, employing pulse data from the XPD as an input. For the upcoming commissioning of the European XFEL, the *WPG* will be available as a versatile tool capable of simulating all relevant optics at the facility.

References

Amann, J. *et al.* (2012). *Nat. Photon.* **6**, 693–698.
Bahrdt, J., Flechsig, U., Grizolli, W. & Siewert, F. (2014). *Proc. SPIE*, **9209**, 920908.
Bean, R., Aquila, A., Samoylova, L. & Mancuso, A. (2016). *J. Opt.* **18**, 074011.
Bruhwiler, D., Chubar, O., Nagler, R., Krzywinski, J. & Boehnlein, A. (2014). *Proc. SPIE*, **9209**, 92090Z.
Bushuev, V. A. (2008). *J. Synchrotron Rad.* **15**, 495–505.
Canestrari, N., Bisogni, V., Walter, A., Zhu, Y., Dvorak, J., Vescovo, E. & Chubar, O. (2014). *Proc. SPIE*, **9209**, 92090I.
Chubar, O., Berman, L., Chu, Y. S., Fluerasu, A., Hulbert, S., Idir, M., Kaznatcheev, K., Shapiro, D., Shen, Q. & Baltser, J. (2011). *Proc. SPIE*, **8141**, 814107.
Chubar, O., Couprie, M.-E., Labat, M., Lambert, G., Polack, F. & Tcherbakoff, O. (2008). *Nucl. Instrum. Methods Phys. Res. Sect. A*, **593**, 30–34.
Chubar, O. & Elleaume, P. (1998). *EPAC 98, 6th European Particle Accelerator Conference*, pp. 1177–1179. Geneva: JACoW.
Chubar, O., Elleaume, P., Kuznetsov, S. & Snigirev, A. (2002). *Proc. SPIE*, **4769**, 145–151.
Chubar, O., Geloni, G., Kocharyan, V., Madsen, A., Saldin, E., Serkez, S., Shvyd'ko, Y. & Sutter, J. (2016). *J. Synchrotron Rad.* **23**, 410–424.
Goodman, J. W. (2004). *Introduction to Fourier Optics*, 3rd ed. Green Village: Roberts and Company Publishers.
Inagaki, T. *et al.* (2014). *FEL2014, Proceedings of the 36th International Free Electron Laser Conference*, TUC01, pp. 603–608. Geneva: JACoW.
Kayser, Y., Rutishauser, S., Katayama, T., Ohashi, H., Kameshima, T., Flechsig, U., Yabashi, M. & David, C. (2014). *Opt. Express*, **22**, 9004–9015.
Kohn, V., Snigireva, I. & Snigirev, A. (2003). *Opt. Commun.* **216**, 247–260.
Loh, N. D. *et al.* (2013). *Opt. Express*, **21**, 12385.

Manetti, M., Samoylova, L., Schneidmiller, E., Sinn, H., Szuba, J., Wrona, K. & Yurkov, M. (2016). *XPD: XFEL Photon Pulses Database*, https://in.xfel.eu/xpd/.

Park, H. J. *et al.* (2013). *Opt. Express*, **21**, 28729.

Potter, D. E. (1973). *Computational Physics.* New York: Wiley.

Roling, S., Zacharias, H., Samoylova, L., Sinn, H., Tschentscher, Th., Chubar, O., Buzmakov, A., Schneidmiller, E., Yurkov, M. V., Siewert, F., Braun, S. & Gawlitza, P. (2014). *Phys. Rev. ST Accel. Beams*, **17**, 110705.

Roth, T., Helfen, L., Hallmann, J., Samoylova, L., Kwaśniewski, P., Lengeler, B. & Madsen, A. (2014). *Proc. SPIE*, **9207**, 920702.

Rutishauser, S., Samoylova, L., Krzywinski, J., Bunk, O., Grünert, J., Sinn, H., Cammarata, M., Fritz, D. M. & David, C. (2012). *Nat. Commun.* **3**, 947.

Saldin, E. L., Schneidmiller, E. A. & Yurkov, M. V. (1999). *Nucl. Instrum. Methods Phys. Res. Sect. A*, **429**, 233–237.

Samoylova, L., Sinn, H., Siewert, F., Mimura, H., Yamauchi, K. & Tschentscher, T. (2009). *Proc. SPIE*, **7360**, 1–9.

Schneidmiller, E. A. & Yurkov, M. V. (2016). *J. Mod. Opt.* **63**, 288–292.

Sutter, J., Chubar, O. & Suvorov, A. (2014). *Proc. SPIE*, **9209**, 92090I.

Suvorov, A., Cunsolo, A., Chubar, O. & Cai, Y. Q. (2015). *Opt. Express*, **23**, 31607–31618.

Yashchuk, V. V., Samoylova, L. V. & Kozhevnikov, I. V. (2015). *Opt. Eng.* **54**, 025108.

Yoon, C. H., Yurkov, M. V., Schneidmiller, E. A., Samoylova, L., Buzmakov, A., Jurek, Z., Ziaja, B., Santra, R., Loh, N. D., Tschentscher, T. & Mancuso, A. (2016). *Sci. Rep.* **6**, 24791.