



Software for the data analysis of the arrival-timing monitor at SACLA

Kyo Nakajima,^{a,b*} Yasumasa Joti,^{a,b} Tetsuo Katayama,^{a,b} Shigeki Owada,^b
 Tadashi Togashi,^{a,b} Toshinori Abe,^{a,b} Takashi Kameshima,^{a,b} Kensuke Okada,^a
 Takashi Sugimoto,^{a,b} Mitsuhiro Yamaga,^{a,b} Takaki Hatsui,^{a,b} and Makina Yabashi^{a,b}

Received 7 June 2017

Accepted 19 November 2017

^aJapan Synchrotron Radiation Research Institute, 1-1-1 Kouto, Sayo-cho, Sayo-gun, Hyogo 679-5198, Japan, and^bSPring-8 Center, RIKEN, 1-1-1 Kouto, Sayo-cho, Sayo-gun, Hyogo 679-5148, Japan.

*Correspondence e-mail: kyo.nakajima@spring8.or.jp

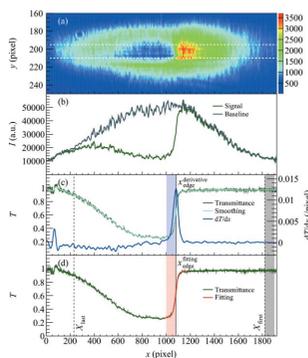
 Edited by V. Favre-Nicolin, CEA and
 Université Joseph Fourier, France

Keywords: X-ray free-electron laser;
 arrival-timing monitor; data analysis;
 software; SACLA.

X-ray free-electron laser (XFEL) pulses from SPring-8 Ångstrom Compact free-electron LASer (SACLA) with a temporal duration of <10 fs have provided a variety of benefits in scientific research. In a previous study, an arrival-timing monitor was developed to improve the temporal resolution in pump–probe experiments at beamline 3 by rearranging data in the order of the arrival-timing jitter between the XFEL and the synchronized optical laser pulses. This paper presents *Timing Monitor Analyzer (TMA)*, a software package by which users can conveniently obtain arrival-timing data in the analysis environment at SACLA. The package is composed of offline tools that pull stored data from cache storage, and online tools that pull data from a data-handling server in semi-real time during beam time. Users can select the most suitable tool for their purpose, and share the results through a network connection between the offline and online analysis environments.

1. Introduction

X-ray free-electron lasers (XFELs) (Emma *et al.*, 2010; Ishikawa *et al.*, 2012) have resulted in many achievements in a variety of fields such as biology, chemistry, materials science and physics. In particular, pump–probe experiments combined with synchronized optical lasers have provided fruitful results in ultrafast dynamics investigations (*e.g.* Tanaka *et al.*, 2013; Oura *et al.*, 2014; Matsubara *et al.*, 2016; Oloff *et al.*, 2016; Hartley *et al.*, 2017; Uemura *et al.*, 2017). Using SPring-8 Ångstrom Compact free-electron LASer (SACLA) (Ishikawa *et al.*, 2012), we developed an arrival-timing monitor (Sato *et al.*, 2015; Katayama *et al.*, 2016) to diagnose the relative arrival timing between the XFEL and the optical laser pulses generated by the synchronized femtosecond optical laser system (Tono *et al.*, 2013). In this scheme, the arrival timing is measured by using the diffracted branch (−1st order) generated with a transmission grating, while the transmitted X-ray beam (zero order) that contains the major part of the total intensity is transported to the user experiments. The correlation between the arrival timings of the −1st-order and zero-order branches shows an overall accuracy of 7.0 fs in root mean square (RMS) in the timing-monitoring scheme. Despite an arrival-timing jitter of about 250 fs (RMS), this accuracy enables a drastic improvement in the temporal resolution by rearranging measurement data in the order of the arrival timing. The arrival-timing monitor was released in experimental hutch 1 (EH1) of beamline 3 (BL3) in September 2015. Recently, a good agreement between the arrival timing obtained with the arrival-timing monitor and the THz streak



camera (Gorgisyan *et al.*, 2017) for two-color XFEL pulses with different energies was reported.

We developed dedicated software applications that have adequate high-speed performance to analyze the data of the arrival-timing monitor as a process during beam times. In general, analysis processes for wide-band data including two-dimensional images are classified into two steps: prompt data analysis (PDA) to extract useful parameters for experimental or analytical feedback to the experiments, and subsequent full data analysis (FDA). To achieve these processes shot-to-shot, together with many parameters to indicate beam properties or experimental conditions, a dedicated data acquisition (DAQ) and analysis system (Joti *et al.*, 2015, 2017) was provided at SACLA and worked successfully for many experiments.

In this paper, we describe the basic concepts of data analysis for the arrival-timing monitor. Software applications were implemented in the PDA environment at SACLA by associating with various application program interfaces (APIs) to retrieve raw data or store the results as mentioned below. The analysis of the data of the arrival-timing monitor can be performed as PDA in both the offline and online analysis environment. The PDA in the offline analysis environment provides the arrival timing of each shot for the FDA of the main experiments after data acquisition has been completed, in which the data are rearranged in the order of the arrival timing. On the other hand, the PDA in the online analysis environment provides the results in semi-real time during the experiments, that are used for monitoring of data accumulation under the consideration of the timing jitter, or data sorting in the FDA of the main experiments on both the offline and online analysis environment. Once the experimental system is designed and constructed, it may be possible to feed back the arrival timing into the operations during experiments. For example, users can remotely control the delay time of the optical laser pulses, which is determined by accurate positioning stages driven by pulse motors, from the user software application in the online analysis environment or another computer within the same network. By integrating the monitoring of the accumulation of the rearranged results in the order of the arrival-timing into the software application, time-resolved measurements over a period greater than the time scale of the arrival-timing jitter may be effectively performed by continuous data acquisition in discrete time-delay steps equivalent to the spread of the timing jitter.

2. Algorithm

2.1. Extraction of the arrival timing

The principles and optical components of the arrival-timing monitor have been described in detail (Katayama *et al.*, 2016). In this paper, therefore, we will start with an example of a frame image obtained from the arrival-timing monitor, as shown in Fig. 1(a). Although the size of the original image is 1920×540 pixels, with a pixel aspect ratio of 1:2, only a region meaningful for extraction of the arrival timing is displayed here. In the arrival-timing monitor, an XFEL pulse

with a temporal duration of <10 fs (Tono *et al.*, 2013) is focused onto a thin crystal of gallium arsenide (GaAs) with an incident angle of 45° , which affects the optical characteristics of the GaAs crystal and rapidly decreases the transmittance. At the same time, an optical laser pulse with a central wavelength of 800 nm and a temporal duration of 40 fs (Katayama *et al.*, 2016) from a synchronized Ti:sapphire laser system is focused on the crystal with a normal incident angle. The two pulses are overlapped temporally and spatially on the crystal. The intensity of the transmitted laser pulse, therefore, decreases in the region where the XFEL pulse passes through the crystal earlier than the timing of the laser pulse. The spatial distribution of the intensity of the transmitted laser pulse is measured by a charge-coupled device (CCD) camera from the back side of the crystal. By repeating this procedure for every shot, the arrival-timing jitter between the two pulses

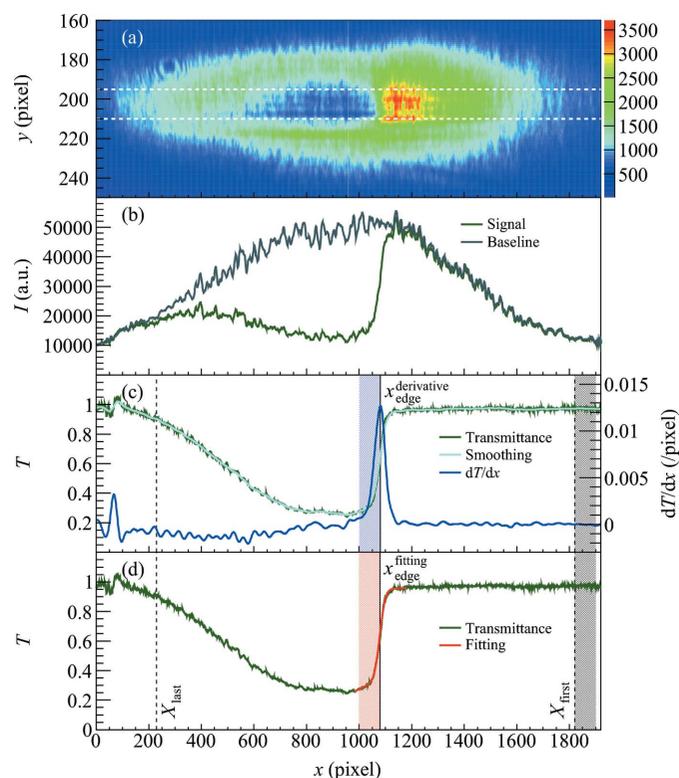


Figure 1

A typical image acquired by the arrival-timing monitor and extraction of the arrival timing. (a) Raw image around the focal area of an optical laser pulse. The two white dashed lines indicate the region of interest (ROI) specified for this image. (b) The green line is the one-dimensional intensity profile of the projection of the image in the ROI of (a), whereas the gray line indicates that of the baseline image obtained with only laser pulses. (c) Derivative profile (blue line) of the smoothed profile (light blue line) of the transmittance profile (green line) obtained by dividing the green line by the gray line in (b). The black line indicates the arrival timing determined as a peak position of the derivative profile. The horizontal width of this image corresponds to a range of approximately 5 ps of the arrival timing. The figure also shows time windows for quality checks. Dashed lines on the left and right side are x_{last} and x_{first} , respectively. The blue shaded region indicates the region around the arrival timing, whereas the gray shaded region indicates the region of the baseline. (d) The red line is a fitted line to the transmittance profile (green line). The red shaded region indicates the region around the arrival timing (black line).

is measurable as a shot-to-shot variation of the spatial distribution of the transmittance.

Fig. 1(b) shows a one-dimensional intensity profile of the projection of the image in the region of interest (ROI) indicated in Fig. 1(a), and that of a baseline image obtained with only the laser pulses. The ROI of this figure was roughly tuned by hand so that a decrease of the transmittance appears clearly. We subtracted the offset due to the dark signal from each frame by using the intensities in unilluminated areas. The configurations of the ROI and the unilluminated areas can be determined by users through a graphical user interface (GUI), described below, and are statically applied to all images in the subsequent analysis processes. The baseline profile in Fig. 1(b) was determined by averaging 1000 consecutive images. Users can store any number of images with only the laser pulses to use for determination of the baseline profile, under the consideration of the fluctuation of the images. The baseline profile needs to be reset by users when variations occur due to realignment of the optics or the drift of the laser so that it always characterizes the intensity of the optical laser pulse without being affected by the XFEL pulse. Fig. 1(c) shows a transmittance profile obtained by dividing the profile with the XFEL pulse by that without the XFEL pulse shown in Fig. 1(b). We can see the characteristic shape of transient transmittance changes caused by an ultrafast XFEL pulse as already described (Harmand *et al.*, 2013; Katayama *et al.*, 2016), that is, a rapid decrease in the transmittance due to an increase of the free-carrier density in the crystal resulting from X-ray-induced photoionization and the subsequent relaxation of that due to recombination of the electrons and the holes. A spike on the transmittance profile in the region near $x = 0$ is caused by a small amount of excess and deficiency on the one-dimensional intensity profile affected by the XFEL pulse. Although this appearance is not regular for the arrival-timing monitor, it is possible to exclude this kind of spike near both ends of the profile from extraction of the arrival timing by introducing the time window described below.

The arrival timing is determined as a peak position of the derivative profile of the smoothed transmittance profile in the range $X_{\text{last}} \leq x < X_{\text{first}}$, which can be specified by users (derivative method) as shown in Fig. 1(c). In this configuration, the position of the arrival timing has a larger value when the XFEL pulse arrives much earlier than the optical laser pulse. Users can choose a smoothing method from integrated options. The following methods are directly available in the analysis tools: (i) a Kernel smoother with the Nadaraya-Watson regression estimator using the given bandwidth, and (ii) a LOWESS smoother using the given smoother span, which are implemented in C++ as part of a modular scientific software framework, *ROOT* (Brun & Rademakers, 1997); (iii) a linear least-squares fit using cubic B-spline basis functions with uniform breakpoints in the GNU Scientific Library (GSL; <http://www.gnu.org/software/gsl/>, accessed: 28/08/2017) under the given number of B-spline coefficients; and (iv) a simple moving average using the given number of data points. The smoothed profile in Fig. 1(c) is estimated with a Kernel smoother with a bandwidth of 30 pixels, which is the same as

the default settings. The calibration factor to convert from pixel units to time units, which is approximately $2.6 \text{ fs pixel}^{-1}$ (Katayama *et al.*, 2016), is extracted from the correlation between the average position of the arrival timing and the optical delay time.

As a complementary method, the arrival timing is also determined using the results of the fitting for the transmittance profile (fitting method) as shown in Fig. 1(d). In this procedure, the transmittance profile around the arrival timing is fitted with the following function, similar to that used in an earlier demonstration of the diagnostic of the relative arrival timing between the optical laser pulses and the XFEL pulses performed at the Linac Coherent Light Source (LCLS) (Harmand *et al.*, 2013),

$$f(x) = \frac{a}{2} \left[\operatorname{erf} \left(\frac{x - x_0}{\sqrt{2}\sigma} \right) + 1 \right] + b_{\pm}x + c_{\pm}, \quad (1)$$

where fitting parameters are a , b_{\pm} , c_{\pm} , σ and x_0 . This function is composed of the error function and a first-order correction term responsible for the difference between the experimental profile and the error function. The position of the arrival timing is determined as x_0 having a different value in each shot due to the timing jitter. When the position of the arrival timing is found to be x_0^1 on the GaAs crystal, the delay time of the XFEL pulse to the optical laser pulse is relatively shorter than that of the position x_0^0 by $(x_0^1 - x_0^0)/(\sqrt{2}c)$. The standard deviation σ is assumed to be characterized by the convolution of the temporal duration of the incident light of the optical laser, the XFEL pulse and the physical processes induced by the XFEL pulse, which are not discussed here. The suffixes $-$ and $+$ represent the left and right sides of x_0 , respectively, generating the asymmetry of the fitting function around the arrival timing as shown in Fig. 1(d). The value of c_- is automatically determined under the condition of the continuity of the fitting function at the position of the arrival timing. The fitting is performed as the χ^2 (least-square) fit using the *MINUIT* minimization package implemented in C++ as part of *ROOT*. It should be noted that the fitting function is only valid in the vicinity of the arrival timing; the range of the data applied to the fitting must be limited around the arrival timing. The range is determined by specifying upper and lower limits relative to the position of the arrival timing previously determined by the derivative method. The initial values of the fitting parameters are automatically set by default as the position of the arrival timing by the derivative method for x_0 , the minimum value of the transmittance profile for c_+ , and 1 minus the default value of c_+ for a . The initial conditions and ranges of the fitting parameters can be refined by users through a GUI. In most case, the initial values of b_{\pm} corresponding to the global gradient around the arrival timing are enough to be set to 0. The fitting parameters determined by users here are statically applied to all images in the subsequent analysis processes.

Users can analyze sample images with the analysis tools that are provided on the HPC system, described in the *Applications* section in detail together with an example of a set of the analysis parameters including the fitting parameters. The

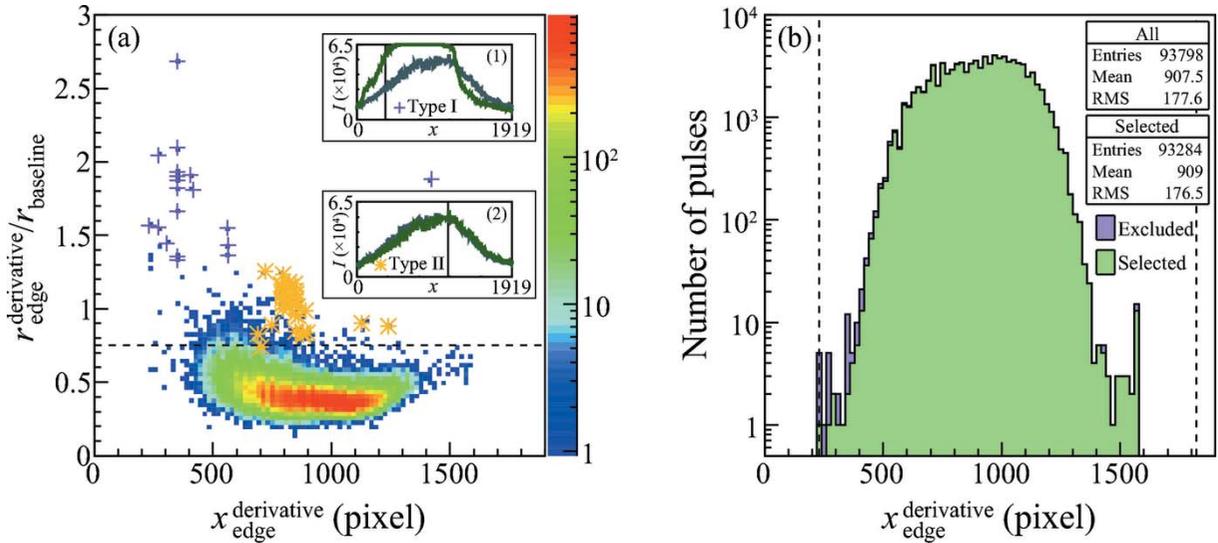


Figure 2

Distributions of results extracted from 93798 images of the arrival-timing monitor. (a) Two-dimensional histogram showing the distribution of $r_{\text{edge}}^{\text{derivative}}/r_{\text{baseline}}$ versus $x_{\text{edge}}^{\text{derivative}}$ (pixel). Magenta plus markers represent results from saturated profiles, which are similar to the example shown in inset (1) using the same manner as Fig. 1(b). Orange star markers represent results from blank profiles, which lack clear edges as shown for example in inset (2). (b) Arrival-timing distribution for all pulses. Histograms for selected values and excluded values are summed cumulatively. Selection is carried out by applying thresholds indicated by the dashed line in (a), $X_{\text{last}} < x < X_{\text{first}}$ in (b), $r_{\text{baseline}} > 0.4$ and $\Delta x_{\text{edge}} < 30$.

position of the spatial overlap of the XFEL pulses and the optical laser pulses may be different for every experiment in response to the alignment procedures. Moreover, the beam profiles are not completely similar to those of other experiments. Therefore, further refinement of the analysis parameters taking into account the characteristics of the images, especially the ROI and the smoothing parameter, is required for users to extract more stable results. In principle, the derivative method without the smoothing is obviously sensitive to local defects of the transmittance profile, whereas the fitting method can be expected to overcome this issue. The effects of the smoothing on the results of both the derivative and fitting methods are shown in Fig. 3, using a Kernel smoother as an example with the same data sets as used in Fig. 2. To extract the dependence of the arrival timing determined around a peak position of the derivative profile on the bandwidth of the Kernel smoothing from all the results, before applying the quality checks described in the next subsection, the 1σ spread of the Gaussian fit of shot-to-shot differences of the values of the arrival timing is shown in Fig. 3 so that invalid results mainly caused by missing the correct peak positions in the derivative method (especially in the case of low bandwidth) have little significance for this discussion. The fitting function is applied to the unsmoothed original profile so that the results of the fitting method, indicated with red plus markers, tend to be more independent of the smoothing than the derivative method, with blue circle markers. The decrease in the differences between the two methods becomes stable around a bandwidth of 30 pixels, which is determined as the default setting of the bandwidth in these data sets. The arrival timing is independently extracted with the two methods to verify the results from the correlation between them, as described below. The effects of the ROI selection are also discussed there.

2.2. Filtering of the results based on the quality checks

Although the settings of the analysis parameters including the ROI, smoothing and fitting are appropriately adjusted, invalid results may be produced as a result of the following possibilities during the experiments. In particular, this relates to the intrinsic fluctuations of the beam profiles. For example,

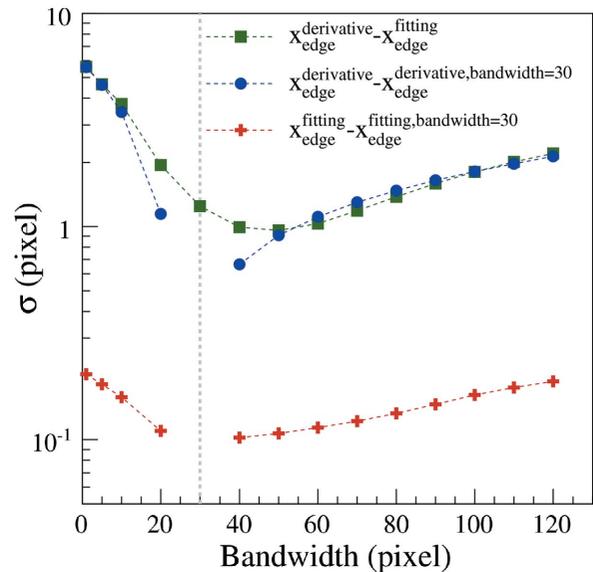


Figure 3

Effects of the Kernel smoother on arrival-timing values with the same data sets as used in Fig. 2. Green square markers represent standard deviations determined with a Gaussian fit of shot-to-shot differences between arrival-timing values by derivative and fitting methods. Blue circles and red plus markers represent standard deviations determined with a Gaussian fit of shot-to-shot differences of arrival-timing values by derivative and fitting methods, respectively. Differences are evaluated for results which are extracted for a bandwidth of 30 pixels, indicated by the gray dashed line.

the XFEL pulses with smaller intensities than expected may form unclear edges in the transmittance profiles, or the optical laser pulses with higher intensities may overflow the images resulting in no edges. It can also be considered that the variation or the drift of the arrival timing may bring the positions of the edges outside of the images. The loss of the edge positions from inside the images may also be caused by artificial operations of the arrival-timing monitor such as changing the optical delay.

Based on these considerations, we may need to identify some data that are not appropriately associated with the arrival timing and exclude them from the FDA. Filtering of the results is classified into four elements, as listed in Table 1, that are correlated to: the status of individual XFEL or optical laser pulses, the temporal overlap between them, and the shape of the resulting transmittance profile. Some of these pulses were previously excluded before extraction of the arrival timing by using the parameters on the database as described in the *Applications* section, whereas others were excluded by using quality checks of the results after extracting the arrival timing as described here. To satisfy this purpose, users can adjust time window and threshold conditions in order to regard the results as valid values of the arrival timing.

The most effective condition utilizes the decrease in the transmittance in the region of the left-hand side of the arrival timing, as shown in Fig. 1(c). This characteristic is quantified using the ratio of the average of the experimental transmittance profile, $r_{\text{edge}}^{\text{derivative/fitting}}$, in the region defined by $x_{\text{edge}}^{\text{derivative/fitting}} - \Delta x_{\text{baseline}} \leq x \leq x_{\text{edge}}^{\text{derivative/fitting}}$, to r_{baseline} in $X_{\text{baseline}}^0 \leq x \leq X_{\text{baseline}}^1$. Here, $x_{\text{edge}}^{\text{derivative}}$ is the arrival timing determined as a peak position of the derivative profile, and $x_{\text{edge}}^{\text{fitting}}$ is that determined by a fitted line to the transmittance profile. The value of r_{baseline} determines the shot-to-shot baseline level of the transmittance profile. For simplicity, the same width is applied to the two regions, $x_{\text{baseline}} \equiv X_{\text{baseline}}^1 - X_{\text{baseline}}^0$. These positions can be defined by users so that the region from X_{baseline}^0 to X_{baseline}^1 extracts the transmittance with only the optical laser pulse. Fig. 2(a) shows a two-dimensional histogram of $r_{\text{edge}}^{\text{derivative}}/r_{\text{baseline}}$ versus $x_{\text{edge}}^{\text{derivative}}$ of 93798 images together with 50 markers corresponding to the values that should be obviously excluded from the FDA.

These abnormal results are visually identified by eye scanning all profiles for features of inappropriate intensity profile shape without any other filtering options. Inappropriate shapes can be separated into two types of intensity profile, referred to as type I and type II,

Table 1
Classification of filtering of pulses before or after extraction of arrival timing.

Elements	Reasons	Filtering methods	Parameters
XFEL pulse	Shutter closed Beam down	Exclusion before extraction using parameters on SyncDB	Open/closed state Available/unavailable state
Optical laser pulse	Shutter closed Low intensity Saturated intensity	Exclusion after extraction using quality checks	Open/closed state r_{baseline} $r_{\text{edge}}^{\text{derivative}}/r_{\text{baseline}}$
Temporal overlap	Out of range		$x_{\text{last}}, x_{\text{first}}$
Transmittance profile	Lack of clear edge False edge		$r_{\text{edge}}^{\text{derivative}}/r_{\text{baseline}}$ Δx_{edge}

as shown in the insets of Fig. 2(a). Type I has large values of $r_{\text{edge}}^{\text{derivative}}/r_{\text{baseline}}$ caused by the saturated laser intensity. Type II has slightly smaller values because the blank profiles lack clear edges. We confirmed that these results are effectively excluded by using a threshold value for $r_{\text{edge}}^{\text{derivative}}/r_{\text{baseline}}$. As the region of the spatial overlap of the XFEL and the optical laser pulses is limited on the images, the values of $r_{\text{edge}}^{\text{derivative}}/r_{\text{baseline}}$ heavily depend on the configuration of the ROI. Fig. 4(a) shows distributions of $r_{\text{edge}}^{\text{derivative}}/r_{\text{baseline}}$ for various center positions and heights of the ROI with the same data sets as used in Fig. 2. By setting a ROI with a height as small as 5–15 pixels to an appropriate position, we can obtain significantly small values of $r_{\text{edge}}^{\text{derivative}}/r_{\text{baseline}}$ corresponding to sharp edges, which provide efficient exclusion of the invalid values using a threshold value for $r_{\text{edge}}^{\text{derivative}}/r_{\text{baseline}}$. On the other hand, Fig. 4(b) shows the 1σ spread of the distribution of shot-to-shot differences of the values of the arrival timing by referring to those which are extracted for a ROI with a height of 5 pixels at the center position of $x_{\text{center}}^{\text{ROI}} = 0$. When the large height is applied, the values of the arrival timing are stably extracted with a difference within 2 pixels for all the positions shown in this figure. In the case of the small height, however, the position of the ROI is required to be optimized appropriately to include the region of the spatial overlap.

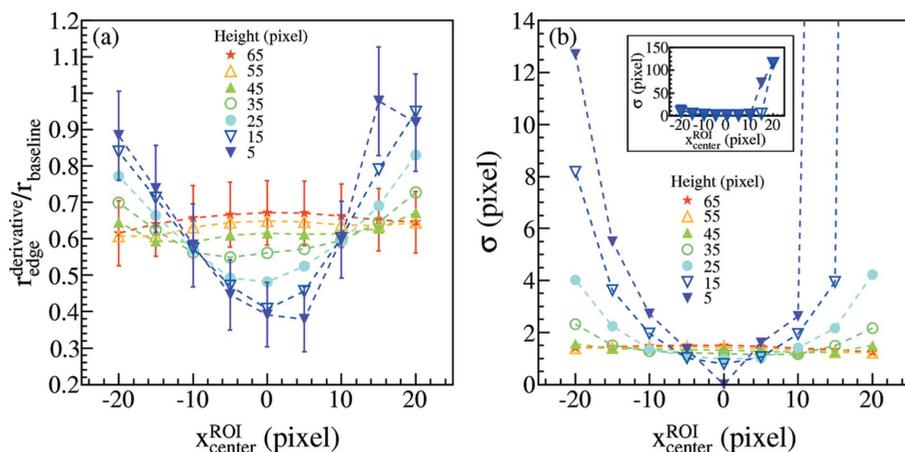


Figure 4
(a) Values of $r_{\text{edge}}^{\text{derivative}}/r_{\text{baseline}}$ and (b) 1σ spread of the Gaussian fit of shot-to-shot differences of values of arrival-timing for various center positions and heights of the ROI with the same data sets as used in Fig. 2. The size of the original image is the same as for Fig. 1(a). In (a), standard deviations for two height conditions are shown for reference. In (b), differences are evaluated for results which are extracted for $x_{\text{center}}^{\text{ROI}} = 0$ and a height of 5 pixels.

Furthermore, three additional conditions are applicable, *i.e.* using (i) r_{baseline} to discriminate the laser intensity; (ii) $\Delta x_{\text{edge}} \equiv |x_{\text{edge}}^{\text{fitting}} - x_{\text{edge}}^{\text{derivative}}|$ to select the differences between the results from the two methods; and (iii) a straightforward time window defined by $X_{\text{last}} \leq x_{\text{edge}} \leq X_{\text{first}}$. Fig. 2(b) shows typical distributions of the selected or excluded results of the derivative method using these conditions. In spite of over-exclusion for $r_{\text{edge}}^{\text{derivative}}/r_{\text{baseline}}$, 99.5% of the results can be used for the FDA owing to the overall stability of the timing-monitoring scheme including the fluctuation of the beam profiles and the timing synchronization, and also the operation of the arrival-timing monitor.

3. Applications

3.1. Overview of the data analysis environment at SACLA

A software package for the PDA of the arrival-timing monitor (*Timing-Monitor Analyzer, TMA*) consists of four tools listed in Table 2: offline *TMA* with a command user interface (CUI) and with a graphical user interface (GUI) (*TimingMonitorAnalyzerCUI* and *TimingMonitorAnalyzerGUI*, respectively), and online *TMA* with a CUI and with a GUI (*TMAOnlineCUI* and *TMAOnlineGUI*, respectively). The four tools use the same algorithm to extract the arrival timing from image data of the arrival-timing monitor so that the results are fully compatible and users can select an appropriate tool for their task. The CUI tools may be useful in cases when users desire to execute the analysis processes from their own scripts after the analysis parameters are optimized. This is described below in detail. The GUIs are built using a Java programming language and combined with the components of the CUI tools in C/C++ languages.

All of the tools operate with the various C APIs, which are provided for the analysis environment at SACLA as shown in Fig. 5. The analysis environment at SACLA consists of two networks separated by firewalls. From a network point of view, one is the online analysis environment corresponding to the inside of the experimental hall and the other is the offline analysis environment corresponding to the outside of the hall.

The offline analysis environment is formed from the high-performance computing (HPC) system connected to various data storage units. Image data stored during an experiment are transferred to the archive storage after a few months of the experiment and still available after that. Users can be logged into the HPC system from the user terminals in the facility and from outside the facility *via* a virtual private network (VPN) connection, so that the offline analysis using stored data on the HPC system can be performed both during and after experiments.

Table 2

Comparison of data analysis tools contained in the software package *TMA*.

		Offline <i>TMA</i>		Online <i>TMA</i>	
User interface		GUI	CUI	GUI	CUI
Operation Environment		Multiple processes on compute nodes in HPC system		Multiple processes on online analysis server	
Input data Import method		Pulling from storage system by <i>DataAccessAPI</i> or HDF5 format files converted by <i>DataConvert</i>		Pulling from memory of data-handling server by <i>OnlineAPI</i>	
Availability		Run data for which data acquisition was completed		Real-time data, not depending on the run	
Calculation Analysis algorithm		Common			
Parameter adjustment		YES	–	–	–
Configuration file		YES (Save/Load)	YES (Load)	YES (Load)	YES (Load)
Output results					
Save results	File	YES	YES	†	†
	UDB	–	–	YES	YES
Statistical displays		YES	‡	YES	‡

† Creatable by dump tools of *TMA*. ‡ Creatable by visualization tools of *TMA*.

On the other hand, the online analysis environment is formed from the online analysis servers, which are available only from the user terminals in the experimental hall, connected to not only disk space on a network attached storage (NAS) between the two analysis environments but also data-handling servers that transfer image data from the camera system to the cache storage. In the online analysis environment, users can obtain online data and perform analysis in semi-real time during the experiments. From a network security point of view, direct network access from the offline analysis environment to the devices on the online analysis environment is strictly prohibited with the firewalls so that the experiments are not unexpectedly interrupted from outside the experimental hall. The results of the online analysis can be stored on the NAS or the user database described below so that these results can be retrieved from the offline analysis environment during and after the experiments. Only the users in their beam times can use both of the analysis environments at the same time, taking advantage of combinations of these features.

Various parameters related to image data are regularly inserted into the event-synchronized database (SyncDB), which is based on a relational database management system and synchronized with the XFEL pulses. These parameters include, for example, the XFEL pulse properties such as the pulse energy and the beam position that are measured by the beam diagnostic instruments (Tono *et al.*, 2013) at the beamline, and the experimental conditions such as the optical delay and the sample position that are controlled by the accurate positioning stages driven by the pulse motors. The primary SyncDB is replicated to the other SyncDBs located in both the offline and online analysis environments, so that data on the SyncDB can be retrieved from either environment.

Even if users perform their own analysis in either the offline or online analysis environment they can directly gather offline or online data from their analysis programs without expert knowledge of the DAQ system at SACLA or the complexity of data conversion into intermediate files. For this purpose, we developed dedicated APIs for users (Joti *et al.*, 2017). These APIs consist of three groups that satisfy different purposes.

The first one is the *DataAccessAPI* for pulling offline data from the storage system and the experimental parameters from the SyncDB. When we store image data to the cache storage, these are saved as a set of a series of data files for a certain period called a run, in which each pulse is identified by a tag number, a unique number incrementally assigned during the whole operation period of SACLA. Data on the SyncDB and image data on the storage system can be correlated with the tag numbers. In addition to the method of gathering these experimental data directly from the analysis applications *via* the *DataAccessAPI*, the facility has provided an application for pre-processing prior to the analysis, *DataConvert* (Joti *et al.*, 2015, 2017), in the offline analysis environment. This application gathers image data and relevant experimental parameters for specified tags in a run from the storage system and the SyncDB, respectively, then converts them into an

HDF5 (The HDF Group, 2006) format file (.h5). Although *DataConvert* has also been developed by implementing the *DataConvertAPI*, the HDF5 format files provide convenience for data portability to users. The offline tools of *TMA* were also designed to run on the HPC system using the *DataAccessAPI* to gather image data in runs for which data acquisition was completed.

The second API is the *OnlineAPI* for pulling online data from the data-handling server in semi-real time while not depending on the runs. The online tools of *TMA* were designed to run on the online analysis servers using the *OnlineAPI*. Thus, only the users in their beam times can use the online tools exclusively without conflicting with other users logging into the HPC system.

Finally, the third API is the *UserDatabaseAPI* (*UDB-API*) for recording/extracting the PDA data associated with each tag number to/from user databases (UDBs), which are also supported by the facility and based on a relational database management system. Whereas the SyncDB is mainly intended to store raw data concerning the status of the devices including the XFEL equipped in the facility, the UDB is designed to store processed data that are created by users through their analysis for each pulse. To achieve this purpose by themselves,

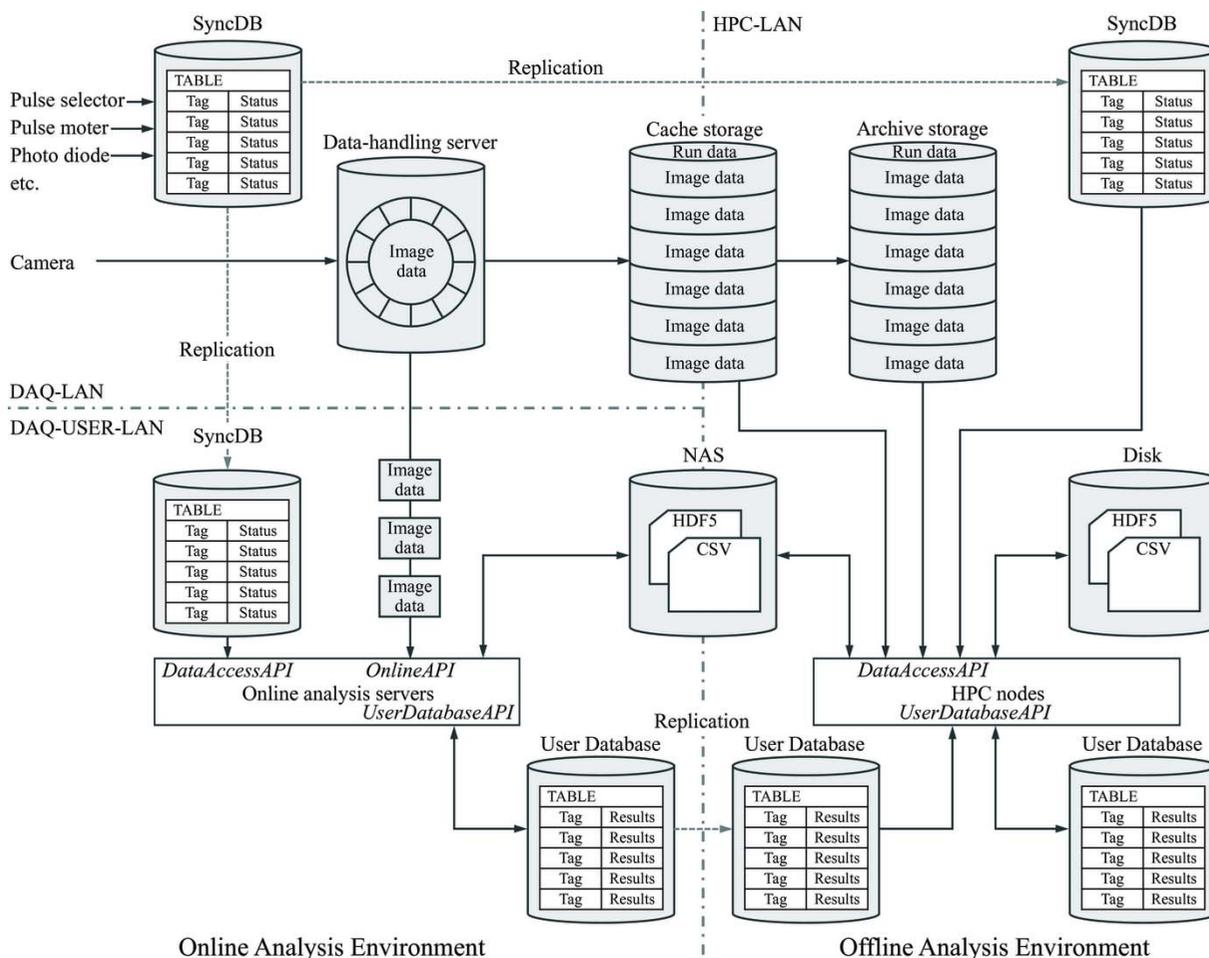


Figure 5 Data flow of the analysis environment at SACLA. The flow of image data branches to online and offline analysis environments, which are separated by firewalls, whereas analysis results can be shared through a network connection between these environments.

the principal investigator of each experimental group can create new tables to insert data in a previously assigned region on the UDB with the *UDB-API*. The UDB in the online analysis environment is only available from the online analysis environment; however, data on it are replicated to the UDB in the offline analysis environment. The replicated UDB allows retrieval of the results, that are inserted in the online analysis environment at a repetition rate of 60 Hz, within one second with asynchronous replication across the network between the analysis environment. Users logging into the HPC system can individually insert/retrieve data to/from another region of the UDB in the offline analysis environment, that is isolated from the replication.

3.2. Offline TMA

The offline *TMA* with the GUI provides three main functions: (i) to adjust execution parameters and save them in a configuration file; (ii) to submit the analysis processes to the other compute nodes; and (iii) to visualize the results with various plots. On the other hand, the offline *TMA* with the CUI covers only the second function based on a specified configuration file. Each function is described in more detail below.

In the case of the offline *TMA*, extraction of the arrival timing for each tag number is executed as an assembly of multiple processes on the compute nodes in the HPC system [Fig. 7(a)]. This takes the form of *MapReduce* programming with a *K MapReduce* (Matsuda *et al.*, 2013) library. In the *Map* step, each process applies the analysis algorithm to a frame image for each tag number and extracts the arrival timing. These processes are ordinarily assigned to four compute nodes with ten processes on each node. Then, the *Reduce* step gathers the results from all nodes participating in the *Map* step and saves them into a CSV format file in the order of the tag numbers. Table 3 shows the execution speed of the offline *TMA* with the CUI for 26 runs, each containing 50000 images with 1920×540 pixels. Experimental conditions of these runs are described in more detail in the *Verification* section below. The execution speed is, on average, 167 Hz. This is faster than the frame rate of 60 Hz and is stable for all runs. The multiple processes are basically submitted to the compute nodes that are available to all users; however, users can request to separate some nodes to be available exclusively during their beam times. In this case, users can select a job class assigned to them by the GUI of the offline *TMA*. Although the separated nodes are immediately returned to the original setting after their beam times, they can submit the multiple processes to the public nodes for the remaining PDA of the arrival-timing monitor.

The offline *TMA* supports two methods to retrieve image data: direct acquisition from the storage system using the *DataAccessAPI* as described above, or creation of the HDF5 format files by *DataConvert* for the specific tags. In both cases, users can apply an additional selection of tags to extract the arrival timing by referring some parameters on the SyncDB, as

Table 3

Processing rate of the offline *TMA* with the CUI on the HPC system.

Number of nodes	Number of runs	Average	Minimum	Maximum	Standard deviation
4	26	166.70 Hz	158.43 Hz	171.23 Hz	2.48 Hz

listed in Table 1. For example, users can apply the open or closed status of the pulse selector in the beamline. After clicking the ‘Execute’ button on the GUI or running the command from the terminal, a single loop of the analysis processes is executed for every run number in a series of runs specified by users in the manner described below: the results are combined in units of the run and saved in CSV-format files. Storage data files contained in a run become available after data acquisition for the run has been completed. If data specified by the run numbers already exist, extracting the arrival timing will start immediately after the process is started. On the other hand, when data specified by the run numbers are not available from the storage system, the analysis processes will wait until each run has been completed. Thus, automatic continuous analysis following data acquisition is possible for each new run during the beam time.

By using the GUI of the offline *TMA*, users can specify various execution parameters including the run numbers for extracting the arrival timing, the run numbers for baseline data, the analysis parameters and output directories for the results. The one-dimensional intensity profile for the baseline is created by averaging all images contained in one or more runs specified here. Among them, any changes to the analysis parameters from the GUI are immediately applied to a single image for a specified tag number. Then, the display of the front panel is updated [Fig. 6(a)]. Consequently, users can optimize these parameters by referring to the visualized results. When the analysis processes are submitted for all images later, the same parameter set that is optimized here will be globally applied.

The offline *TMA* with the CUI uses the execution parameters in a configuration file that is specified by users as an argument of the command. As the analysis parameters should be carefully optimized using a certain amount of data in storage to avoid influence of the fluctuation of the beam profile, the tools except for the offline *TMA* with the GUI are designed to apply the analysis parameters optimized through the offline *TMA* with the GUI. The analysis parameters are saved in a configuration file together with the baseline profile that was applied when optimizing the parameters. The file format of the configuration file is HDF5 from a standpoint of portability so that it can be easily shared with users. Note that the analysis parameters that are applied to the analysis processes are automatically saved to an HDF5 file by the offline tools or a table on the UDB by the online tools. These values can be referred to by users later as a part of the execution logs. In addition to a configuration file, users have only to specify a range of the run numbers in the command line. The results are saved in CSV-format files in the same manner.

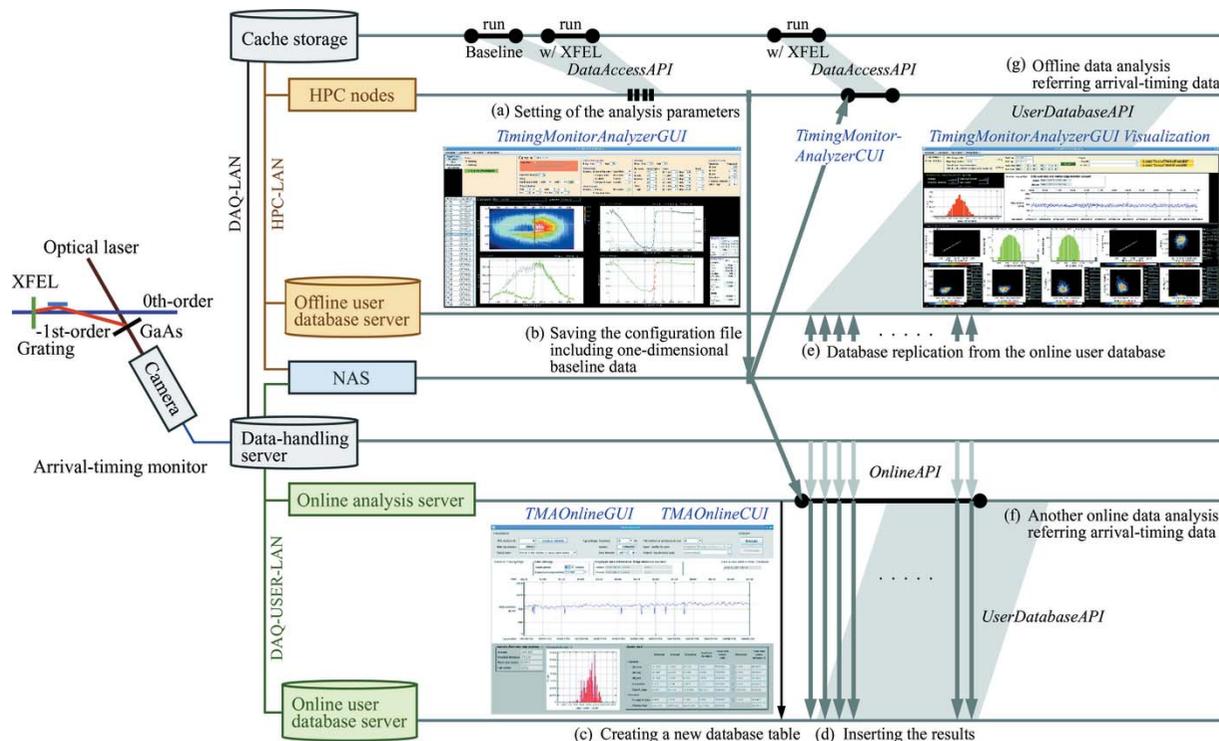


Figure 6 Sequential steps of PDA for the arrival-timing monitor, and the results in online and offline analysis environments. (a) Analysis parameters are optimized on the HPC system using data in storage. (b) Optimized parameters including adopted baseline data are saved into the configuration file in NAS between offline and online environments. (c) Online tools create a new database table in online user database (UDB) and (d) extract results based on configuration and store them in the table. (e) Results are replicated in the offline UDB immediately. Thus, results can be referred from other analysis processes in the (f) online or (g) offline environments in semi-real time.

In the case of the offline *TMA* with the GUI, users can check the status of the analysis by statistical information displayed for each run. On the other hand, to give priority to high-speed execution of only extraction of the arrival timing, the CUI tool omits additional processes such as creating various graphical plots, and then the statistical information is not displayed by the tool itself. After the analysis is completed, however, this information is available by loading log files of the CUI tool from a visualization function provided by the offline *TMA* with the GUI.

3.3. Online *TMA*

In the case of the online *TMA*, continuous extraction of the arrival timing for each tag number is executed as an assembly of multiple processes on a dedicated online analysis server in each beamline [Fig. 7(b)]. It adopts master/slave programming in a Message-Passing Interface (MPI) with

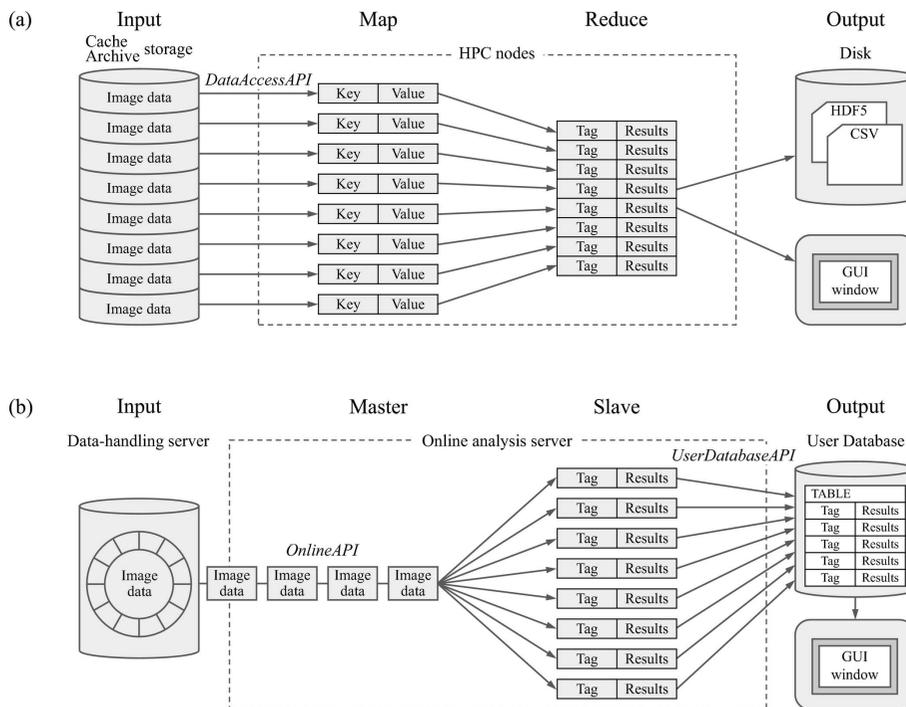


Figure 7 Schematic drawing of data flow in the offline *TMA* (a) in the form of MapReduce programming, which pulls image data from storage system, and (b) the online *TMA* in the form of master/slave programming, which pulls online data continuously.

an MPICH (MPICH, 2015) library. The master process continuously acquires a frame image for each tag number from a data-handling server using the *OnlineAPI* and provides it to the slave processes. The slave processes extract the arrival timing from them and store the results in the UDB with the tag numbers. Table 4 shows the system usage of the slave processes per 30 tags when a single CPU core was used for the data acquisition of the frame images with 1920×540 pixels at a repetition rate of 30 Hz. Thirty CPU cores were assigned to the parallel processes to extract the arrival timing. The processes per 30 tags consumed the resources for 9.4% of the execution time on average. Distributions of the usage indicate that the processes have enough speed to follow the XFEL pulses in semi-real time.

The online *TMA* uses the execution parameters, including the baseline profile, that are adjusted by the offline *TMA* with the GUI. These parameters are shared with a configuration file saved in the NAS between the offline and online analysis environments [Fig. 6(b)]. Users have only to specify a few parameters associated with execution of the online processes, such as the repetition rate of the data acquisition and the total number of tags in a series of analyses, by the GUI or arguments in the command line. The repetition rate of the data acquisition, that is set to the DAQ system by the facility staff based on requests of users, currently cannot be acquired with the user-level API in the online analysis environment, so the value needs to be set manually.

In the case of the online *TMA* with the GUI, users can create a new table in the UDB and specify it to save the results [Fig. 6(c)], or select a table that already exists in the UDB. In both of the online *TMA*s, after the execution is started, the analysis processes will start immediately for the tags that are currently available for analysis [Fig. 6(d)]. The results can be retrieved in other analysis programs using the *UDB-API* in semi-real time, not only from the UDB in the online analysis environment [Fig. 6(f)] but also from the replicated UDB on the offline analysis environment [Fig. 6(e)]. Thus, the results are easily shared from the offline analysis processes. Furthermore, to convert the results in the UDB to other formats, the offline *TMA* with the GUI provides functions to save the results into CSV- or HDF5-format files.

In the case of the online *TMA* with the GUI, users can check the status of the analysis by statistical information and a trend graph of the arrival timing. The results of the online *TMA* are replicated in the offline UDB immediately so that they can be retrieved from visualization tools of the offline *TMA* with the GUI. The graphical displays can also be created [Fig. 6(g)].

4. Verification

To determine the consistency of the results from the offline and online tools, and to collect performance parameters such as execution speed and stability, we executed the two tools in the analysis environments at SACLA simultaneously. The measurements were performed using the arrival-timing monitor in EH1 of BL3 at SACLA, which was released to user

Table 4

System usage of slave processes per 30 tags of the online *TMA* with the CUI on online analysis server.

Number of cores	Number of tags	Average	Minimum	Maximum	Standard deviation
30	108000	9.40%	6.88%	13.27%	0.76%

experiments. XFEL pulses with a central photon energy of 10 keV and optical laser pulses with a central wavelength of 800 nm were synchronized at a repetition rate of 30 Hz. The frame images with 1920×540 pixels from the arrival-timing monitor were acquired for over 12 h, corresponding to the time of a single shift of the user experiments.

The images were continuously stored in 26 runs, each containing 50000 pulses. Extraction of the arrival timing for each new run was successfully performed by the offline *TMA* with the GUI on the HPC system during the beam time. Before extracting the arrival timing, 35174 missing pulses, mainly owing to beam down, were excluded as an additional selection using the parameters on the SyncDB, described in §3.2. Finally, 1264826 pulses were selected to extract the arrival timing, and only 13813 pulses corresponding to 1.09% were excluded by the quality checks described in §2.2 under the same conditions as those used to apply the selections in Fig. 2(b), resulting in the arrival timing for 1251013 pulses that are available in the PDA.

While frame images were being stored and analyzed, extraction of the arrival timing for each tag number was executed by a prototype of the online *TMA* with the CUI in an online analysis server using the same configuration as the offline tool. The online processes were successfully executed for 1296000 pulses (Fig. 8), corresponding to 12 h and a repetition rate of 30 Hz, that are continuously acquired from a data-handling server, while a few tens of tags between consecutive runs are normally discarded without being stored in storage due to the specification of the DAQ system. The number of pulses processed by the online tool still includes a few missing pulses owing to beam down in this period. A single CPU core was used for data acquisition from the data-handling server and 29 CPU cores for the parallel processes to extract the arrival timing, thus the average execution time of the parallel processes was 109 ms per tag, which was equivalent to 267 Hz with the entire 29 processes, and was maintained stably.

Taking into account the total tag numbers entered into the offline and online analysis tools, and their time periods not being identical, the compatibility of the results obtained by the two tools was investigated. As a result, 1251861 pulses from which the arrival timing was extracted by both of the analysis tools within the same periods lead to identical results without missing online data on the semi-real time processes, indicating that the two methods are compatible between the results. Note that this comparison was performed between the results that were obtained by applying almost the same, but not completely the same, configuration as that used in the verification for the offline analysis tool described above, to both of the tools.

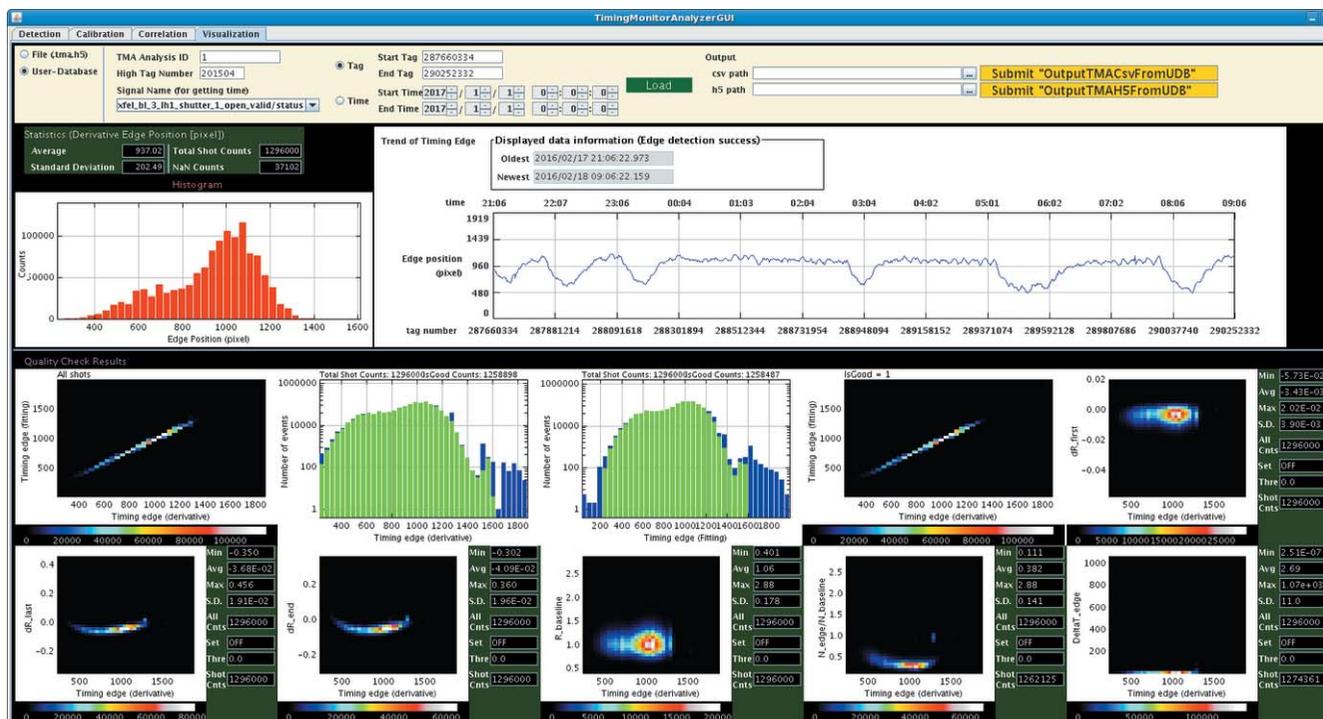


Figure 8 Display of the results of the online *TMA* with the CUI for 12 h. Results of arrival timing are indicated in (top left) histogram and (top right) trend graph. A summary of the quality checks is shown by various histograms in the bottom panel.

For these reasons, the number of results available for the comparison is slightly different from that of the previous verification.

5. Conclusions and outlook

TMA, a software package for data analysis of the arrival-timing monitor at SACLA, has been developed as part of the offline and online analysis environments. Following the first version of the offline *TMA* with a GUI, which was released with the apparatus of the arrival-timing monitor in September 2015, the CUI version and the online *TMA* were successively released by the end of 2016. Extraction of the arrival timing for over 12 h was successfully demonstrated for each run by the offline *TMA* and in semi-real time by the online *TMA*.

By sharing the configuration file through a network connection between the offline and online analysis environments, users can easily execute the CUI tools from their own scripts without being bound by the GUI tools. Additionally, all *TMA* tools are compatible between results. Thus, users can select the most suitable tool for their purpose from the offline or online methods.

Based on the appropriate performance of the analysis processes, a more convenient way to write/read the results will be provided, such as a function to store the results of the offline tools in the UDB, or dedicated APIs by which users can retrieve individual elements of the results from the UDB without interpreting an internal data structure applied by the *TMA* when inserting the results. To provide a simplified

method for discriminating shot-to-shot characteristics of the XFEL or optical laser pulses corresponding to the results by the online tools in the same manner as the offline tools, there are plans to add a process that follows the semi-real time processes to gather the values of the parameters from the SyncDB and discriminate the pulses based on the settings of the selection criteria in the configuration file. A brief flag indicating whether or not the result is applicable to the FDA is attached by the process to each result on the UDB. The flags are stored in fields different from those for the results of the arrival timing so that users can find out the results of the discrimination promptly and independently of the extraction of the arrival timing.

Acknowledgements

The authors would like to acknowledge suggestions and feedback provided by users that contributed to improving *TMA*. The contribution of engineering and technical staff at SACLA involved in the development of *TMA* is also gratefully acknowledged. The experiments for the arrival-timing monitor were performed at BL3 of SACLA with the approval of the Japan Synchrotron Radiation Research Institute (JASRI) (Proposal Nos. 2015A8069, 2015B8066, 2016A8068, and 2016B8087).

References

Brun, R. & Rademakers, F. (1997). *Nucl. Instrum. Methods Phys. Res. A*, **389**, 81–86.
 Emma, P. *et al.* (2010). *Nat. Photon.* **4**, 641–647.

- Gorgisyan, I. *et al.* (2017). *Opt. Express*, **25**, 2080–2091.
- Harmand, M., Coffee, R., Bionta, M. R., Chollet, M., French, D., Zhu, D., Fritz, D. M., Lemke, H. T., Medvedev, N., Ziaja, B., Toleikis, S. & Cammarata, M. (2013). *Nat. Photon.* **7**, 215–218.
- Hartley, N. J. *et al.* (2017). *Appl. Phys. Lett.* **110**, 071905.
- Ishikawa, T. *et al.* (2012). *Nat. Photon.* **6**, 540–544.
- Joti, Y., Kameshima, T., Yamaga, M., Sugimoto, T., Okada, K., Abe, T., Furukawa, Y., Ohata, T., Tanaka, R., Hatsui, T. & Yabashi, M. (2015). *J. Synchrotron Rad.* **22**, 571–576.
- Joti, Y., Nakajima, K., Kameshima, T., Yamaga, M., Abe, T., Okada, K., Sugimoto, T., Hatsui, T. & Yabashi, M. (2017). *Synchrotron Radiat. News*, **30**, 16–21.
- Katayama, T., Owada, S., Togashi, T., Ogawa, K., Karvinen, P., Vartiainen, I., Eronen, A., David, C., Sato, T., Nakajima, K., Joti, Y., Yumoto, H., Ohashi, H. & Yabashi, M. (2016). *Struct. Dyn.* **3**, 034301.
- Matsubara, E., Okada, S., Ichitsubo, T., Kawaguchi, T., Hirata, A., Guan, P. F., Tokuda, K., Tanimura, K., Matsunaga, T., Chen, M. W. & Yamada, N. (2016). *Phys. Rev. Lett.* **117**, 135501.
- Matsuda, M., Maruyama, N. & Takizawa, S. (2013). *Proceedings of the 2013 IEEE International Conference on Cluster Computing (CLUSTER)*, 23–27 September 2013, Indianapolis, IN, USA.
- MPICH (2015). *MPICH*, <http://www.mpich.org> (accessed 28/04/2017).
- Oloff, L. *et al.* (2016). *Sci. Rep.* **6**, 35087.
- Oura, M. *et al.* (2014). *Trans. Mater. Res. Soc. Jpn*, **39**, 469–473.
- Sato, T., Togashi, T., Ogawa, K., Katayama, T., Inubushi, Y., Tono, K. & Yabashi, M. (2015). *Appl. Phys. Express*, **8**, 012702.
- Tanaka, Y., Ito, K., Nakatani, T., Onitsuka, R., Newton, M., Sato, T., Togashi, T., Yabashi, M., Kawaguchi, T., Shimada, K., Tokuda, K., Takahashi, I., Ichitsubo, T., Matsubara, E. & Nishino, Y. (2013). *J. Ceram. Soc. Japan*, **121**, 283–286.
- The HDF Group (2006). *HDF5*, <http://www.hdfgroup.org/> (accessed 28/04/2017).
- Tono, K., Togashi, T., Inubushi, Y., Sato, T., Katayama, T., Ogawa, K., Ohashi, H., Kimura, H., Takahashi, S., Takeshita, K., Tomizawa, H., Goto, S., Ishikawa, T. & Yabashi, M. (2013). *New J. Phys.* **15**, 083035.
- Uemura, Y. *et al.* (2017). *Chem. Commun.* **53**, 7314–7317.