

BEAMLINE-CONTROLLED STEERING OF SOURCE-POINT ANGLE AT THE ADVANCED PHOTON SOURCE*

L. Emery, G. Fystro, H. Shang, and M. Smith, ANL, Argonne, IL 60439, USA

Abstract

An EPICS-based steering software system has been implemented for beamline personnel to directly steer the angle of the synchrotron radiation sources at the Advanced Photon Source. A script running on a workstation monitors "start steering" beamline EPICS records, and effects a steering given by the value of the "angle request" EPICS record. The new system makes the steering process much faster than before, although the older steering protocols can still be used. The robustness features of the original steering remain. Feedback messages are provided to the beamlines and the accelerator operators. Underpinning this new steering protocol is the recent refinement of the global orbit feedback process whereby feedforward of dipole corrector set points and orbit set points are used to create a local steering bump in a rapid and seamless way.

INTRODUCTION

In order to give beamlines more convenient (and faster) control of their source points, APS has implemented in June 2016 direct source-point angle steering using EPICS in concert with a workstation-based server script.

The Advanced Photon Source (APS), like many other light sources, has been providing a steering request system to the beamlines to effect a local change in the source angle or (rarely) source position. Because the light source has a large number of beamlines, APS has implemented in 1995 a formal protocol between the beamlines and the accelerator control room using an intermediary floor coordinator (FC) contacted by pager and telephone (Fig. 1). Needless to say, the process of communicating back and forth by these means to iterate to a final steering condition took time and became too onerous to use frequently.

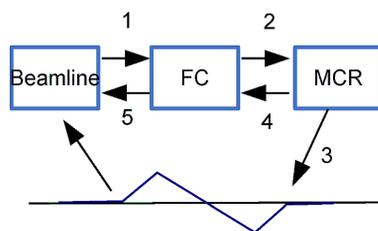


Figure 1: Original request protocol. Numbers indicate the telephone communication flow through FCs. Step 3 represent the actual source angle steering process for a particular beamline. Time to completion was 3 to 10 minutes.

* Work supported by the U.S. Department of Energy, Office of Science, Office of Basic Energy Sciences, under Contract No. DE-AC02-06CH11357.

In Jan 2016, a new (and somewhat bridging) steering request protocol was implemented, in which the beamlines communicated with the Main Control Room (MCR) operators directly using a web page form, reducing the turn-around time for steering requests to a few minutes.

Before the roll-out of the web page protocol a new algorithm for performing a local steering was developed and implemented, in which the switching between the "slow" global orbit feedback and a local bump feedback was replaced by the same global orbit feedback but with insertions of feedforward bumps for beam position monitor (bpms) set points and dipole corrector current set points. The design goal of the new algorithm was both to eliminate micron-step orbit changes of the original method and to make it possible to automate.

This feedforward algorithm was implemented in an operations GUI ("operator" steering) in Dec. 2015, permanently replacing a long-standing steering procedure, and then later, as a workstation-based server using EPICS ("beamline" steering) (Fig. 2) in June 2016. Both these methods are available for use now.

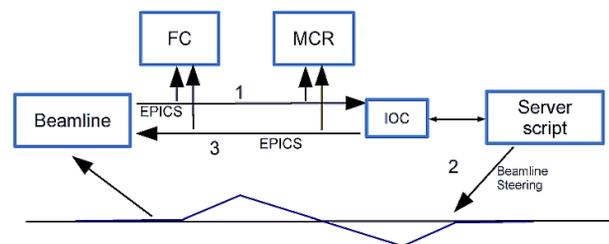


Figure 2: Direct steering protocol. The flow is through the EPICS control system. Time to completion is 5 to 20 seconds.

STEERING REQUEST BY WEB PAGE

Though superseded by the new direct steering some details of this protocol is reported here since some non-EPICS beamlines have not yet connected with the new steering PVs with an EPICS client. This protocol is used to request insertion device (ID) and bending magnet (BM) source angle steering in addition to two other user-run operational activities, beamline intensity optimization and x-ray bpm calibration, not yet implemented as a beamline-controlled activity.

A series of web pages were created for requests and responses between a beamline's console and an operator's console. The beamline staff types in an angle value and a direction for one or both planes. The beamline user presses a "Send" button, which causes an audible notification in the MCR's web page. After the MCR operators execute the

steering procedure and complete it, they press a “completed” button to notify the beamline.

Because the communication is done through the internal internet, the beamlines were registered with their workstation IP so that they would be allowed to send request only for their own beamline. The FCs can also send any request for all beamlines. The request data are archived in a web server, and statistics can be done on requests.

STEERING BY FEEDFORWARD

Previously at APS local steering operations were conducted during an interruption of the 10-Hz “slow” global orbit correction, with a gap of a few seconds in global control and the possibility of small orbit steps. Both were running on an EPICS IOC connected to the 1.5 kHz real-time orbit feedback network [1] for access to the bpm readbacks and corrector set points. Local steering actions has now been integrated into the same global orbit correction by way of a feedforward vector bump in the beam position monitor (bpm) set points and corrector set points.

Under normal global orbit feedback operation, an sddscontrollaw process (an SDDS-compliant EPICS tool, [2]) in the orbit correction IOC cycles through the following steps: read a vector of bpm readback from the fast orbit feedback network, calculate a vector of bpm errors using a vector of bpm set points (reference orbit), multiply the error vector by a correction matrix to produce a vector of corrector current setpoint change, i.e., a delta vector. This delta vector is added to a vector of present corrector setpoints to make a vector of new corrector setpoints. The latter is an automated accumulator process in the IOC. This corrector setpoints vector is sent out to the orbit feedback network, completing the iteration cycle.

The delta vector of corrector can be loaded externally to the IOC as an EPICS array PV independently of the running status of orbit correction. This is possible because of the stand-alone accumulator process mentioned above.

Assuming that the loop is open and the whole bpm error vector is zero, one can implement a local bump by applying a 4-corrector closed-bump vector. The bpm error signal will remain zero if we also apply a vector of expected orbit change. See Figure 3 for a real example. Thus the vectors of bump corrector delta and bump bpm setpoint delta form a consistent feedforward bump pair.

We added a bpm setpoint delta vector in the IOC plus the associated accumulator process to permit the feedforward pair capability.

Since global feedback maintains a close-to-zero orbit error at all times, inserting such pair of vectors in between global orbit iterations does not interfere in any way the global orbit correction.

Since we are feedforwarding both bpm and corrector setpoints we encounter no serious problems of bump non-closure. Since global orbit correction proceeds uninterrupted, there is no problem with possible drift from, say, perturbing insertion devices in the ring.

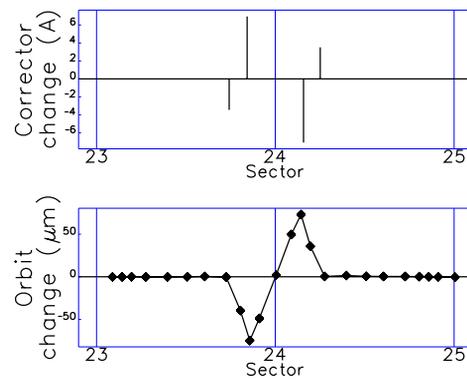


Figure 3: A pair of feedforward angle-bump vectors, one for corrector delta and one for bpm setpoint delta.

As with the old steering method we implemented several robustness features into the various processes to make sure that minimum conditions for valid operation are met. In global orbit correction, we check stored current, corrector limits, corrector range limits (i.e. how much a corrector changed since the start of the loop), bpm readback limits, whether some auxiliary workstation processes are running on the local network. EPICS alarms would alert operations of invalid conditions. For the steering process, the expected final local orbit is checked to see whether it would trip the orbit interlock or cause a corrector to reach its limit. In these cases, the steering process would not commence, status and error messages would be sent, and EPICS alarms set off.

The operator GUI to feedforward steering (“operator” steering) consists of selecting a beamline, entering values for proposed Δx , $\Delta x'$, Δy and $\Delta y'$, and buttons for previewing the orbit change and for launching the steering.

When the steering is launched, the GUI calculates the appropriate feedforward delta waveform pair for the selected beamline and sends them by EPICS array PV to the global orbit correction IOC. As mentioned earlier the delta vectors are accumulated asynchronously to the appropriate vector interleaved with the regular orbit correction cycles. The GUI is configured to limit the local orbit setpoint change in one step to about 20 μm , thus the waveforms are scaled accordingly, and several waveform pairs are sent at an interval of 1 second. The x and y planes are steering in sequence, though they could be simultaneous if we wanted to.

We still follow the policy of removing x-ray bpm from the global orbit correction before doing steering, since they would fall out of calibration for a new photon beam position. The removal is an automated procedure before the start of steering.

STEERING BY SERVER

ID, canted ID and BM sources can now be controlled by their respective beamlines using the same EPICS access security as for ID gap control. We allow only source angle steering for the beamline version of local steering. Only one source point steering is handled at a time for now, thus

a steering lock-out mechanism is implemented to prevent simultaneous requests.. Limits on steering angles and on accumulated steering angles are enforced to prevent operational problems. The beamline steering system may be disabled by MCR operators at any time when an unrelated operational problem occurs.

There are two major components that work together. One is a series of beamline steering PVs residing in a single soft IOC, and the other, a workstation-based server written in Tcl that monitors the values of the steering PVs.

The beamline steering controls are accessed through an EPICS MEDM window (Fig 4) or from a script. A beamline

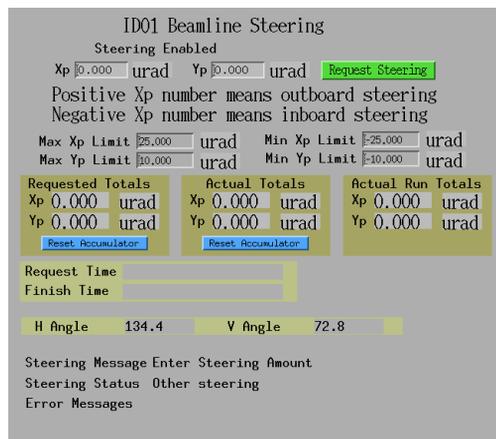


Figure 4: MEDM window of beamline steering controls.

staff enters values for the X_p ($\Delta x'$) and Y_p ($\Delta y'$) PVs, and then press the "Request Steering" button (sending a value of 1 to a PV). Some logic is performed in the IOC to check the limits on the angle request and on the accumulated angle steered. The workstation server, watching all of the beamlines "start" PVs, gets ready to launch the steering. But first, the server performs a lock-out procedure preventing the other beamlines from pressing their "start" buttons, until the present steering is complete.

The server then performs the local steering as described in the "operator" steering above. The same library calls are made, the same limits and checks are applied. The accumulated amount of steering are recorded in EPICS PVs. A steering status and a time stamp PV are available. To ensure a valid recovery file for the whole SR, we archive a new one at every steering.

Beamlines that do not operate generally with EPICS were able to run this steering on a local workstation running EPICS client.

A multi-threaded version of the server script can be written. It has been demonstrated with the operator interface version of steering on several workstations that simultaneous steering does not perturb the beam unduly.

There is no "undo" button if the beamline makes a steering mistake. The beamline should simply do the opposite steering. The beamline can now easily keep track of its history, since the steering requests and accumulators are PVs that are archived continuously.

2: Photon Sources and Electron Accelerators

A05 - Synchrotron Radiation Facilities

There are other steering-related operational procedure that are not yet implemented in server mode. Intensity Optimization is a simplex optimizer run by operators that attempts to maximize the readback of a beamline photon intensity monitor while searching the best angle steering in the x and y planes. If desired the optimization could in the future be initiated by the beamline through an EPICS command as well.

The other procedure is that of the x-ray bpm re-calibration. Implementing this on a server would require more thought, as human intervention is sometimes required to check the validity of results.

We ran the beamline steering server for one run starting in June. It was utilized a large number of time on a few beamlines. About half the beamlines did not need steering at all.

SOFTWARE PRINCIPLES FOLLOWED

As with our other workstation-based processes and archivers, the server software and libraries are written in Tcl/Tk. Libraries are written to avoid code duplication, which served us well in creating two version of feedforward steering, one for the operator interface and one for the beamline server. All data files followed the SDDS file protocol [3], which means that the powerful SDDS toolkit can be used for data processing, for example for obtaining the feedforward vectors of local bumps from lattice optics. For the moment all temporary files are kept for debugging any possible problems. SDDS-compliant EPICS tools are used to send vector data to the EPICS control system. It was fortuitous that the vector accumulator in the orbit correction IOC ran independently from the feedback (i.e. the principle of having many processes running independently but collaboratively), otherwise we would not have seen as clearly the opportunity of slipping the feedforward into the feedback process.

CONCLUSIONS

Over the years the stability of source points at APS have much improved, and fewer request for steering are being made at every start-up. Nevertheless, in response to a request from beamline participants of our beam stability working group, the accelerator division have streamlined the steering request process in two phases, leading to a system where the beamlines has direct control over the steering of their respective source point.

REFERENCES

- [1] F. Lenkszus *et al.*, "Integration of Orbit Control with Real-Time Feedback", in *Proc. PAC 2003*, Portland, OR, USA, paper TOPB008 (2003).
- [2] H. Shang *et al.*, "New Features in the SDDS-Compliant EPICS Toolkit", in *Proc. PAC 2003*, Portland, OR, USA, paper FPAG004 (2003).
- [3] R. Soliday *et al.*, "New Features in the SDDS Toolkit", in *Proc. PAC 2003*, Portland, OR, USA, paper FPAG005 (2003).