# CANADIAN LIGHT SOURCE BEAM POSITION VISUALIZATION TOOL

M. A. Bree*, T. D. Batten, C. W. Stevens, J. M. Vogt

Canadian Light Source, Saskatoon, Canada

## Abstract

The Canadian Light Source orbit correction system acquires, collates, and publishes storage ring beam centroid position information from 48 beam position monitors (BPMs) at a rate of 1000 samples per second. We present a "Beam Position Visualization Tool" that computes and displays dynamic Fast Fourier Transforms (FFTs) and Cumulative Power Spectral Densities (CPSDs) for all BPMs in real-time using full resolution data. The computed FFTs and CPSDs can be plotted in various combinations and in waterfall plots that allow visualization of changes over long periods of time. In addition, correlations between all BPM channel combinations are computed and ranked. Data from any two BPM channels can be selected for plotting in two dimensions wherein correlations are visually apparent. Computed CPSDs are further binned and archived for further analysis. Preliminary results from the Beam Position Visualization Tool have proven useful in identifying storage ring beam position noise at the Canadian Light Source.

## INTRODUCTION

The Canadian Light Source (CLS) is a third generation synchrotron commissioned in 2005 that now supports 22 operational beamlines. The storage ring orbit correction (OC) system was upgraded in November 2021 to replace obsolete hardware and support new diagnostic capabilities [1]. The original Beam Position Monitors (BPMs) and associated Bergoz electronics are unchanged but new high speed D-tAcq analog-to-digital converters (ADCs) [2] feed beam position data via fiber optic cables to an OC host computer running Scientific Linux 7. The OC host uses the BPM data as input to the orbit correction algorithm to compute and send setpoints to the orbit corrector magnets (OCMs) over dedicated private Ethernet networks. The OC system uses the Experimental Physics and Industrial Control System (EPICS) [3] for configuration and control.

Many new features were built into the new OC system to facilitate specific diagnostic requirements such as hardware status. One such feature, called "beam position broadcast", publishes a data stream consisting of the beam positions read at all OC BPMs. Since the data stream protocol is connectionless and static, its usage by clients is completely decoupled from the OC system. Future diagnostic tools can be implemented using the data stream without modifying, interrupting, or otherwise interacting with the running OC system in any way.

_____

* michael.bree@lightsource.ca

This paper covers two general topics:

- *Beam Position Broadcast*, which publishes BPM data.

- *Beam Position Visualization Tool*, which is an example of a client that consumes the published BPM data.

## BEAM POSITION BROADCAST

The new OC system uses four D-tAcq ACQ435ELF ADCs to read BPM values of the X (horizontal) and Y (vertical) beam centroid positions at 48 locations around the CLS storage ring. The BPMs are read at 1000 Hz, and each reading (sample) is three bytes. Using four bytes per sample results in a data rate of 384,000 bytes/s (plus protocol overhead) to transmit all 96 channels.

Publishing the beam position data in some form allows clients external to the OC system to process the information as they see fit, independently of OC operation. The CLS uses the EPICS control system and one possible mechanism for publishing beam position data is via EPICS Process Variables (PVs). EPICS clients can connect to PVs via a TCP-based protocol and can receive (and transmit) data via PVs over the control system network. However, the 1000 Hz update rate is high for EPICS (although there are techniques to mitigate this). More important, though, was the desire to not burden the real-time OC host computer that publishes the BPM data stream with hosting EPICS PVs for full resolution BPM data; and to not require data consumers (clients) to support EPICS. Therefore, EPICS was not used to publish the BPM data stream.

### System Architecture

The User Datagram Protocol (UDP) [4] was chosen to publish the BPM data stream. UDP is a lightweight, connectionless protocol that minimizes the burden on the OC host computer which has as its primary task running the orbit correction algorithm and sending updates to the OCMs. A dedicated CPU core on the OC host was assigned to transmit the data stream on a configurable network interface. The data stream is transmitted on a private network to keep it off the control system network. It is sent to the broadcast address which allows any number of clients to independently receive it with no impact on the transmitting OC host. For example, the beam visualization tool described later in this paper is a data stream client that runs on a separate computer which is connected to the private network. The tool is a pure Python program that receives the BPM data stream directly, processes it, and publishes its results as EPICS PVs. The overall system architecture is illustrated in Fig. 1.

This is a preprint — the final version is published with IOP

12th Int. Beam Instrum. Conf.  IBIC2023, Saskatoon, Canada  JACoW Publishing

ISBN: 978-3-95450-236-3  ISSN: 2673-5350  doi:10.18429/JACoW-IBIC2023-MO3C05
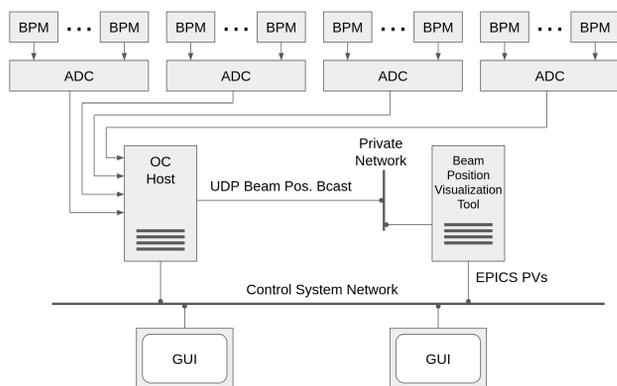
Figure 1: System Architecture. Each of the 4 ADCs supports 12 BPMs. The OC host transmits the BPM data stream on a private network on which multiple clients can receive it. Processed results are published as EPICS PVs on the control system network where they may be accessed by screens and graphical user interfaces (GUIs).

### BPM Data Stream Protocol

Each transmitted frame consists of a 20-byte header as shown in Fig. 2. Typically, clients read the header of each
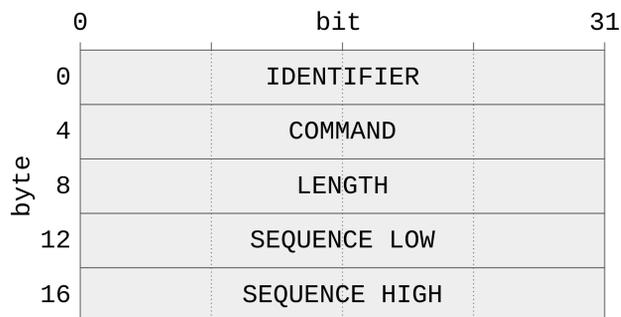


Figure 2: BPM data stream protocol frame header structure.

received frame, and immediately reject any frame that does not contain the 32-bit identifier. Next they can read the 32-bit length field which indicates the number of bytes in the frame and reject any frame with a size that does not match. They may optionally check the 64-bit sequence number as frame delivery and frame order is not guaranteed by UDP[1]. Finally, the 32-bit command is read and the frame processed accordingly. Currently, the protocol consists of two commands:

- **BPM_Position:** The frame contains sequential 32-bit numbers, each representing the calibrated voltage read by a BPM ADC.

- **BPM_Meta:** The frame contains a JavaScript Object Notation (JSON) string consisting of a human readable dictionary specifying the index, name, and volt-to-metre conversion factor for each channel. The dic-

---

[1] In 1.5 years of continuous operation no missing or out of sequence frames have been detected.

tionary also contains the current sampling rate. Such dictionaries are easily parsable by high level languages such as Python.

The OC host transmits a **BPM_Position** frame each time the BPMs are read (1000 Hz). Also, it sends a **BPM_Meta** whenever the meta data change, or once per second otherwise.

This protocol simplifies client implementation. For example, clients need not be EPICS aware as they would be had EPICS been chosen to publish the BPM data.

## BEAM POSITION VISUALIZATION TOOL

### Dynamic Fast Fourier Transforms

After the new OC system was commissioned, a project was launched to use the new BPM data stream to dynamically compute and display the Fast Fourier Transforms (FFTs) and Cumulative Power Spectral Densities (CPSDs) of all 96 BPM channels. A pure Python multi-threaded program reads the data steam and waits for the next **BPM_Meta** frame that when received allows the program to dynamically configure itself. It then sends subsequent BPM samples from **BPM_Position** frames to worker threads, one for each channel. By monitoring the frame sequence numbers, missing data can be detected and FFT computations suspended until the data stream resumes.
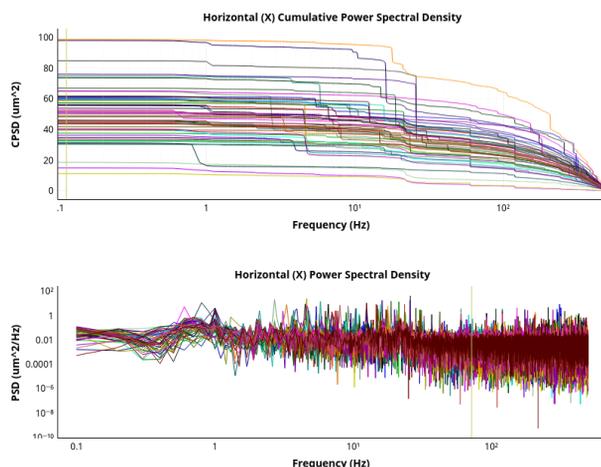


Figure 3: CPSDs and FFTs for all 48 horizontal BPM channels [1]. 60 Hz noise can be seen in the CPSDs for all channels. Other large spikes affect only individual BPMs or small groups of BPMs. Any combination of channels may be selected for plotting.

The Python `multiprocessing` module is used for each worker, ensuring true parallel processing. Each worker adds each beam position sample to 10,000 element First-In-First-Out (FIFO) memory buffers for FFT computations. This produces a FFT from 0.1 to 500.0 Hz at 0.1 Hz resolution. At some configurable interval computations are performed using the FIFOs as input and the resulting FFTs are published as EPICS waveform PVs.

The FFTs are integrated from the Nyquist frequency (500 Hz) back to DC resulting in CPSDs which are also published as EPICS waveform PVs.

A GUI (Fig. 3 and first presented in Ref. [1]) was developed to display the dynamically computed FFTs and CPSDs. The dynamic display makes it easy to see transient noise, and noise that affects single BPMs and multiple BPMs. For example, in Fig. 3, 60 Hz noise is discernible in all BPMs.

### Data Archiving

The CPSDs are further summed into five large frequency bins: 0.1-1 Hz, 1-10 Hz, 10-100 Hz, and 100-500 Hz. Each of these values is published at a configurable interval (currently once per second) as an EPICS scalar PV. This results in 480 PVs (i.e., 5 per channel) suitable for archiving.

### Dynamic Waterfall Plots

Another visualization technique is to treat each FFT for a BPM as a row in an image with pixel intensity corresponding to FFT amplitude. The resulting images illustrate noise as a function of time on the vertical axis and frequency on the horizontal axis. A new row is added to the top of the image each time a new FFT is calculated, and the bottom row (based on a configurable time span) is removed. Thus the image scrolls downward, leading to the *waterfall* name. Some interesting results are shown in Figs. 4 and 5. These images were collected with orbit correction in operation and with typical user-mode ring current. At the present time, sources of noise revealed by this new visualization tool are under investigation.
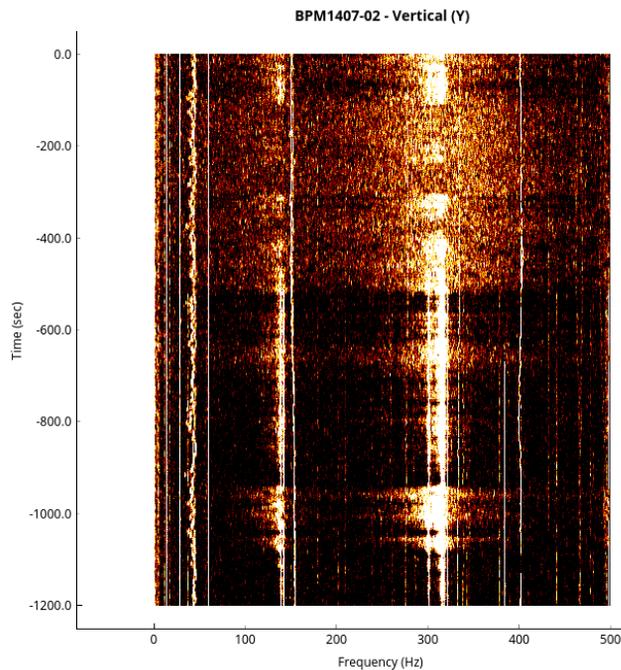


Figure 4: Intermittent noise across a wide frequency spectrum.
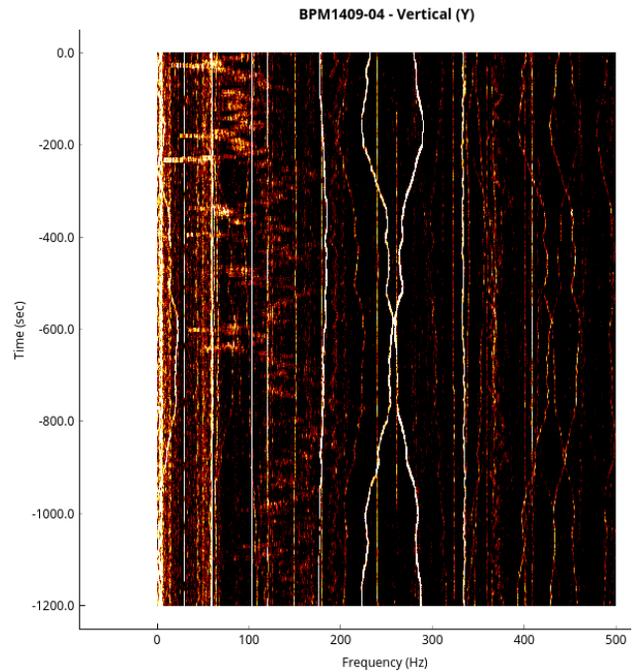


Figure 5: Narrowband noise that slowly drifts in frequency between 200 Hz and 300 Hz.

### Position Visualization

The tool uses the BPM data stream to plot data from pairs of BPM channels as pixels on an X-Y grid. The grid is divided into bins, and a bin "hit count" is incremented each time a measured position falls within it. As data is acquired, an image appears with pixel intensity proportional to the hit count, resulting in probability density function of beam position. An example is shown in Fig. 6.

Bin hits can be optionally aged out, and the pixel takes on the behaviour of a phosphorous screen with persistence. What appears is a dynamically changing image of beam position, updated at 1000 pixels per second. When the X and Y channels of the same BPM are chosen, the pixels represent the actual measured beam centroid position at the BPM. Examples are illustrated in Figs. 7 and 8.

### Channel Correlation

The tool computes the correlation between all combinations of the 96 beam position channels over sliding time windows of configurable duration. The pairs with the highest correlation are displayed on a GUI from which they can be selected for plotting.

### Machine Learning

The CLS is researching the potential application of machine learning techniques to orbit correction [5]. A Python client was written to gather and store full resolution beam position data from the BPM data stream for training the neural network. Users can now gather beam position data whenever and for whatever duration they want.

This is a preprint — the final version is published with IOP
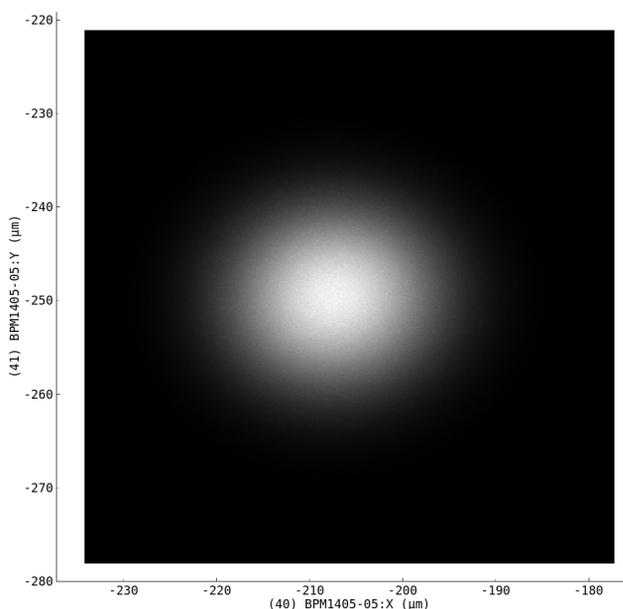
Figure 6: Image of beam position at BPM1405-05 with aging disabled. All recorded positions are kept and the pixel intensity reflects the total number of observations in the corresponding bin. The image represents the beam position probability density function over a period of time.
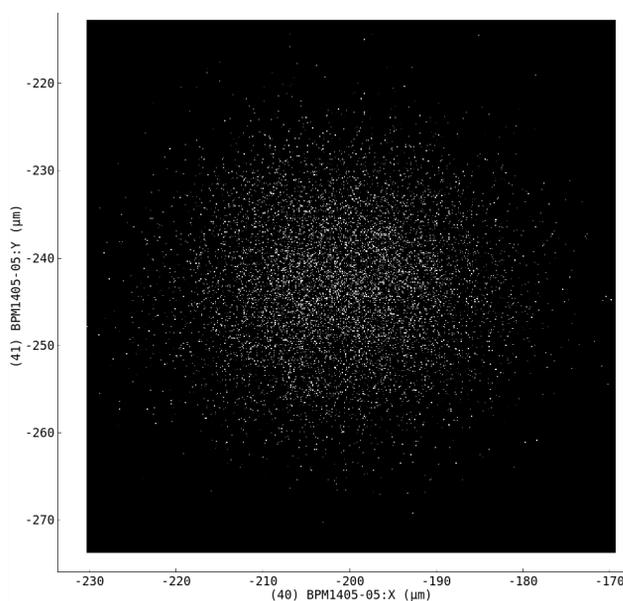


Figure 7: Image of beam position at BPM1405-05 with aging enabled. That is, position observations are discarded after a configurable period of time. The dynamic image is updated at 1000 pixels per second.

## FUTURE WORK

The data stream could be expanded to include full rate orbit corrector magnet (OCM) setpoints. This would include potential commands such as **OCM_Setpoint** and **OCM_Meta**. These are analogous to the **BPM_Position** and **BPM_Meta** commands described above except that
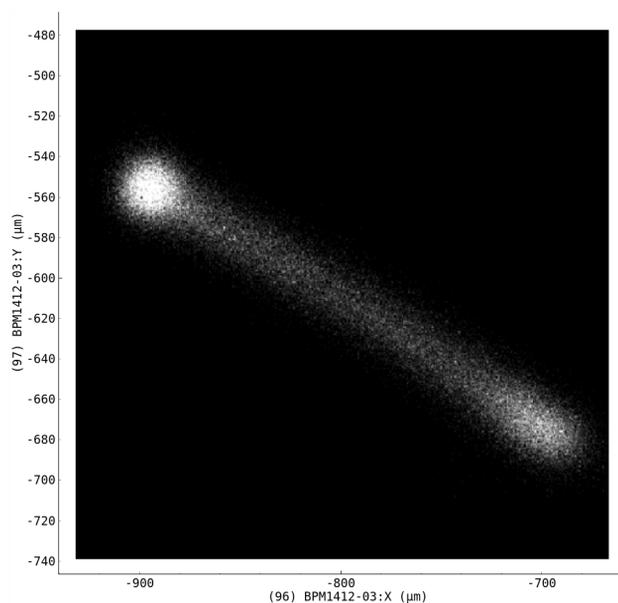


Figure 8: Beam motion at BPM1412-03 during injection with ring current at 3.4 mA.

they would describe the OCM setpoints computed by the OC system and the OCM metadata respectively. This additional data, along with the beam position information, has the potential to facilitate external tools to characterize the performance of the complete OC system. There are many possibilities for analysis of the streamed data. For example unexpected beam behaviour such as clustering or deviation from other reference measurements could be detected and flagged.

## CONCLUSION

The beam position broadcast feature has facilitated new visualization and analysis tools after the deployment of and without affecting the in-production orbit correction system. The extremely simple protocol allows anyone with interest to receive and analyze the BPM data stream (e.g., with simple Python) with no risk or impact on the orbit correction system.

The results have provided new insights and views of the CLS beam position characteristics that are still being evaluated. The archived CPSD values have proven useful for correlating changes in beam noise to other events, including modifications to the orbit correction algorithm. Likewise data gathered from the BPM data stream has facilitated studies such as machine learning initiatives.

## ACKNOWLEDGEMENTS

# REFERENCES

[1] T. D. Batten, M. A. Bree, and J. M. Vogt, "Orbit Correction Upgrade at the Canadian Light Source", in *Proc. 11th Int. Beam Instrum. Conf. (IBIC'22)*, Kraków, Poland, Sept. 2022, pp. 485-487.
`doi:10.18429/JACoW-IBIC2022-WEP34`

[2] D-tAcq ACQ435ELF 32 Channel Simultaneous Digitizer,
`https://www.d-tacq.com/acq435elf.shtml`

[3] Experimental Physics and Industrial Control System (EPICS),
`https://epics.anl.gov`

[4] User Datagram Protocol (UDP),
`https://www.ietf.org/rfc/rfc768.txt`

[5] S. Saadat and M. J. Boland, "A New Approach for CLS Future Orbit Correction System Driven by Neural Network", presented at IBIC'23, Saskatoon, Canada, Sept. 2023, paper MOP036, this conference.

02 Beam Position Monitors