

A GUI FOR SLOW ORBIT FEEDBACK USING MATLAB*

H. S. Kang[†], E. H. Lee, J. Choi

Pohang Accelerator Laboratory, Pohang, 790-784, Korea

Abstract

A GUI using MATLAB is coded for slow global orbit feedback that is based on a newly established EPICS environment in PLS. The GUI displays the response matrix, the SVD analysis, the orbit, and the correction kicks. The orbit feedback algorithm uses the SVD (singular value decomposition) method. The orbit feedback GUI gives a good convenience in the analysis of orbit feedback.

INTRODUCTION

MATLAB [1] is becoming popular in particle accelerator applications. MATLAB provides many mathematical tools such as matrix algebra, mathematical functions and graphic display, which enables accelerator physicist to realize physics and control algorithm in an easy way. So, MATLAB is widely used as high-level software for accelerator physics and control in many accelerator laboratories [2,3,4].

The PLS also adopted MATLAB as accelerator analysis and control software, and a slow global orbit feedback using MATLAB is now being prepared to control the slow orbit movement. The slow global orbit feedback system is designed to have a feedback bandwidth of below 1 Hz, and a feedback speed of a few Hz.

The slow global orbit feedback is based on a newly established EPICS environment. A GUI using MATLAB was coded for the feedback.

GLOBAL ORBIT FEEDBACK PROGRAM

Orbit Feedback GUI

Figure 1 shows the orbit feedback GUI. It has several displays; the response matrix, the SVD analysis, the inverse response matrix, the calculated feedback kick, and the real time orbit.

The orbit feedback algorithm uses the SVD method. The BPMs and correctors used in the feedback algorithm can be selected among total 108 BPMs and 70 corrector magnets at the window menu.

The response matrix is obtained by measuring all BPM readings after applying the same kick to each corrector magnet one by one. The response matrix R is represented by

$$Y=R \times X \quad (1)$$

where X is the vector of correction kick with the length of 70, and Y is the BPM Reading with the length of 108, and R is the response matrix with the size of 108 x 70.

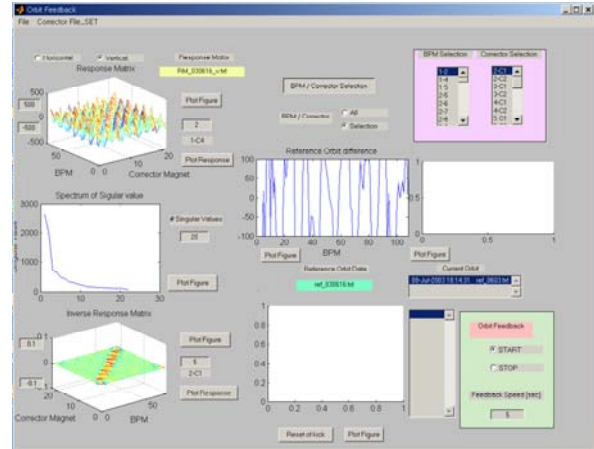


Figure 1: MATLAB GUI for slow global orbit feedback.

By SVD, the response matrix R can be decomposed into U, S, V

$$R=U \times S \times V^T \quad (2)$$

where, U (108*108) is the transform of the BPM readings, S is the diagonal matrix (108 * 70), V^T (70*70) is the square orthonormal eigenvector matrix, which transforms the corrector settings.

The inverse of R can be written as

$$R^{-1}=V \times inv(S) \times U^T \quad (3)$$

The present algorithm uses 70 BPMs and 22 corrector magnets. The available number of BPMs is changeable because some BPMs show suspicious behaviours; parasitic movement or large current dependence. The selected correctors are the ones located at the straight sections where the horizontal and vertical beta function is relatively high.

The ADC bit of BPM is only 12 bit. Thus, at the preliminary feedback test, the number of useful BPMs is about 70. But, it is expected that the number of available BPMs shall be limited when the ADC bit of BPM is improved to 16 bit so that the reading accuracy of BPM critically determines the feedback performance, and even a small current dependence of BPM affects the feedback performance.

Figure 2 shows the measured response matrix and its SVD eigenvalues, and the inverse response matrix. The number of correctors used in this figure is 70.

* Supported by the Ministry of Science and Technology of Korea.

[†]hskang@postech.ac.kr

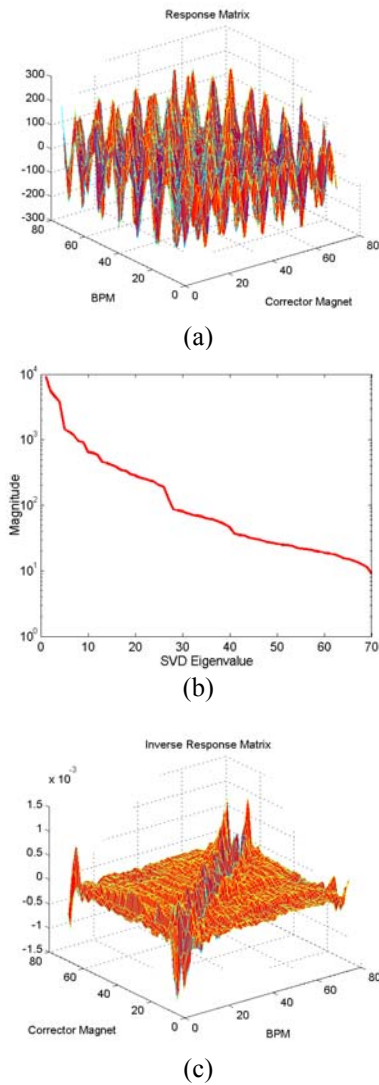


Figure 2: Response matrix, SVD eigenvalues, and inverse response matrix.

MATLAB Channel Access

MATLAB Channel Access (MCA) - MATLAB interface to EPICS Channel Access (CA) client library – is the same one developed in SLAC [5]. EPICS IOCs of BPM and corrector magnet power supplies are PC IOC, but will be replaced with VME IOC in the near future.

BPMs and correctors are accessible as EPICS Process Variables (PV's). The MCA commands are MCAOPEN, MCAGET, MCAPUT, etc.

```
>> Handle=mcaopen('PVName');
```

Array PV's is also read or write and transferred into a MATLAB array. The command to read or write is

```
>> ArrayHandle = mcaopen('ArrayPVName');
>> ArrayData = mcaget('ArrayHandle');
>> mcaput(ArrayHandle, NewData);
```

Feedback Algorithm Analysis

The orbit feedback program can be used for the algorithm analysis of fast feedback. In the fast feedback, a very limited number of BPMs and correctors will be used. Shown in Figure 3 is an example of fast feedback analysis which is assumed to use 11 BPMs and 11 correctors. This figure shows the SVD analysis and the response amplitude of corrector to the BPM readings. Only 3 BPM readings are used to calculate the correction kick of corresponding corrector. It is important to select appropriate BPMs and correctors for the feedback to function well.

FEEDBACK TEST RESULT

The slow global orbit feedback using MATLAB was tested and the algorithm of the feedback was proved well. However, we concluded that the orbit feedback with the current hardwares for BPMs and corrector magnets can not be applied into the user service operation at this moment because the performance limitations of BPM and corrector power supplies make the orbit feedback worse.

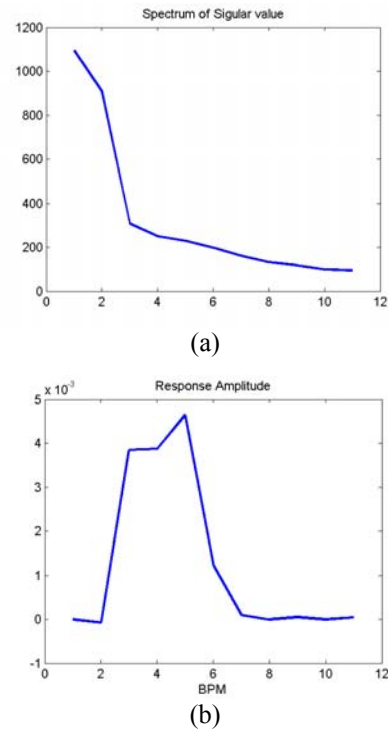


Figure 3: Analysis of feedback using 11 BPMs and 11 correctors: (a) SVD analysis and (b) response amplitude of corrector to the BPM readings.

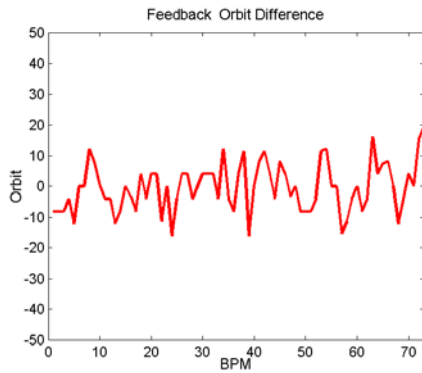


Figure 4: The feedback real-time orbit.

Bad BPMs are sorted out and about 70 out of 108 BPMs are used in the feedback test. And the vertical plane only was tested. The performance of the feedback using 22 correctors at the straight sections is same to using all correctors (70 ea). Both the 12-bit ADC of BPM and the 12-bit DAC of corrector power supplies make the orbit under feedback hunt for the reference orbit. Figure 4 shows the orbit under feedback, which is clearly limited by the BPM reading resolution of $5\text{ }\mu\text{m}$.

Using a 12-bit DAC power supply, the kick angle per 1 bit is $1\text{-}\mu\text{rad}$ and the calculated orbit rms in the vertical plane generated by feedback itself is $6.3\text{ }\mu\text{m}$. To overcome this quantization noise of power supply, it is highly requested to replace with 20-bit corrector power supply.

Synchronization problem among the corrector power supplies causes a big movement of the orbit every feedback turn; while setting the correction values for

corrector power supplies. After the correction is finished at each feedback turn, the orbit looks unchanged, of course, which still has a hunt problem as explained in Fig. 4.

CONCLUSION

The orbit feedback GUI using MATLAB gives a good convenience in the analysis of feedback. The appropriate choice of BPMs and correctors is possible with this analysis. The slow global orbit feedback using MATLAB can not be applied into machine operation until hardwares for BPMs and correctors are replaced with higher bits.

ACKNOWLEDGEMENT

The authors thank Andrei Terebilo in SLAC for the use of MATLAB channel access toolbox.

REFERENCES

- [1] MATLAB, The MathWorks, Natick, MA.
- [2] G. Portmann, "Slow Orbit Feedback at the ALS using MATLAB," PAC'99, New York, 1999, p. 2373.
- [3] J. Corbett et al., "Interactive Orbit Control in MATLAB," PAC'01, Chicago, 2001, p. 813.
- [4] J. Safraneck et al., "MATLAB-based LOCO," EPAC'02, Paris, 2002, p. 1184.
- [5] A. Terebilo, "Channel Access Client Toolbox for MATLAB," SLAC-PUB-9339.