BEAM-BASED ALIGNMENT FOR SEXTUPOLES VIA PARALLEL OPTIMIZATION

X. Huang*, SLAC National Accelerator Laboratory, Menlo Park, United States

Abstract

We propose a method to perform beam-based alignment measurement for multiple sextupoles simultaneously by minimizing the induced orbit shifts from sextupole modulation through parallel local orbit optimizations. The parallel optimization is made possible by using local orbit bumps as knobs and separating the induced orbit kicks through model-based response matrices. The approach reduces slow, multi-knob optimization problems into multiple fast, single-knob optimization problems. The method is demonstrated in simulation.

INTRODUCTION

Beam-based alignment (BBA) aims at finding the magnetic centers of magnets which can be used to define the orbit target. BBA is important for accelerator commissioning and operation as there are many benefits when the beam is steered through the magnet centers. Quadrupole BBA is often an important step in the commissioning of new accelerators and a routine task on operating machines. Sextupole BBA is less common presently. However, as misalignments in sextupoles cause linear optics and coupling errors which have big impact on the performance of low emittance storage rings, it may be more important to steer beam through sextupole centers in such machines.

Similar to quadrupole BBA, sextupole BBA can be done through measurements of the induced orbit shifts (IOS) from modulation of the strengths of the magnets. Parallel-BBA (PBBA), a method to simultaneously find the centers of multiple quadrupoles by correcting the induced orbit shifts has been proposed and demonstrated at SPEAR3 [1]. It has also been successfully tested at other facilities [2–4]. Another parallel method for quadrupole BBA, parallel-QMS (P-QMS), scans the orbits at the targeted magnet locations to find the beam orbits with zero induced kicks [5], using the measured IOS and the orbit response matrix to isolate the induced kicks by the magnets.

Both PBBA and P-QMS can be extended to sextupoles. In Ref. [6], the P-QMS method is applied to sextupoles by fitting the induced horizontal kicks to the horizontal or vertical orbits to parabolic curves from which the magnetic field centers can be found. The principle of PBBA is to minimize the IOS with orbit changes at the magnet locations. This method has been tested to sextupoles in experiments [5]. However, convergence can be very slow when applied to multiple sextupoles simultaneously since it becomes a nonlinear multi-dimensional optimization problem, unlike the linear case when PBBA is applied to quadrupoles.

* xiahuang@slac.stanford.edu

In this paper, we propose an approach to divide the nonlinear multi-dimensional problem into multiple 2-dimensional optimization problems that can be optimized on the machine in parallel. This approach can greatly expedite the convergence toward the optimum. We also propose a coding structure for implementing online optimization algorithms which not only facilitates parallelizing multiple instances, but also allows better control of the optimization process.

SEXTUPOLE BBA BY MINIMIZING IOS

The IOS is the orbit change before and after the targeted magnet group changes its strengths. The IOS can be accounted for with the induced kicks at the magnet locations due to the orbit offsets at these magnets,

$$\Delta\mathbf{x} = \mathbf{R}_{x,s}\Delta\boldsymbol{\theta}_{x,s}, \quad \Delta\mathbf{y} = \mathbf{R}_{y,s}\Delta\boldsymbol{\theta}_{y,s}, \tag{1}$$

where $\mathbf{R}_{x/y,s}$ are orbit response matrices from kicks applied at the magnet locations to the BPMs and $\Delta\boldsymbol{\theta}_{x/y,s}$ are such kicks due to modulating the strengths of the magnets. For sextupoles, the induced kicks and the orbit offsets are related through,

$$\Delta\theta_{x,s} = -[\Delta K_2 L]\frac{1}{2}\left((x_s - x_0)^2 - (y_s - y_0)^2\right) \tag{2}$$

$$\Delta\theta_{y,s} = [\Delta K_2 L](x_s - x_0)(y_s - y_0), \tag{3}$$

where $[\Delta K_2 L]$ is the change of integrated strength, (x_0, y_0) represents the sextupole field center and (x_s, y_s) the orbit at the magnet. When the beam passes through the sextupole field center, the induced kicks are zero in both planes.

The magnetic centers can be found by online optimization, using the beam orbits at the magnet as knobs. The objective can be the sum of squares of the IOS on all BPMs in both planes. It is a relatively simple problem if one sextupole is involved at a time (with only 2 knobs). However, for cases with multiple sextupoles powered by a common power supply, it becomes a challenging nonlinear optimization problem (with $2N_s$ knobs for N_s sextupoles). For most optimization algorithms, convergence becomes considerably slower as the dimensionality of the optimization problem increases. The slow convergence could inhibit the use of the method to realistic sextupole BBA problems.

The difficulty of slow convergence can be addressed by dividing the single optimization problem with multiple sextupoles into multiple optimization problems, each targeting one sextupole only, which can be carried out simultaneously in parallel. This is possible because the induced kicks by the different sextupole magnets can be separated out, using

$$\Delta\boldsymbol{\theta}_{x,s} = \left(\mathbf{R}_{x,s}^T\mathbf{R}_{x,s}\right)^{-1}\mathbf{R}_{x,s}^T\Delta\mathbf{x}, \tag{4}$$

and similarly for the y-plane. The response matrices $\mathbf{R}_{x/y,s}$ can be calculated with a lattice model.

Since the induced kicks from one sextupole are only determined by the local orbit, the magnet center can be found by steering the orbit (x_i, y_i) to minimize the induced kicks by the magnet, with an objective function

$$f_i = \Delta\theta_{x,i}^2 + \Delta\theta_{y,i}^2, \quad (5)$$

where subscript $i = 1, \cdots, N_s$ indicates the i'th sextupole. The original optimization problem with $2N_s$ knobs now becomes N_s independent problems, each with only 2 knobs.

These optimization problems can be performed on the machine in parallel, however, with the requirement that the evaluations of the objective functions are synchronized. This is necessary because the measurement of IOS for all sextupoles in the targeted group is done at the same time. The optimization algorithms need to be implemented in a way to accommodate synchronized function evaluations for multiple independent problems.

IMPLEMENTATION OF PARALLEL ONLINE OPTIMIZATION

There are two approaches to implement parallel online optimization that requires synchronized function evaluations. The first approach is a multi-thread, client-server framework. Each optimization problem is executed in a client thread which sends the trial solutions to the server thread and waits for the evaluation results. The server thread collects requests from all clients, combines them into one measurement, executes the measurement, and distributes the results back to the client threads. Although with this approach the implementation of the optimization algorithm does not need to change, the complexity of runtime control increases.

The second approach is to rewrite the optimization algorithm code for it to facilitate synchronized function evaluation in a single-thread, linear structure. This approach is more like the traditional tuning process and allows direct, human-in-the-loop control. However, changes to the optimization algorithms are necessary as the objective function is usually evaluated at multiple places in the code, corresponding to various different scenarios. The sequence of the evaluation scenarios differs according to the initial condition and is impossible to predict, except for population-based stochastic algorithms and algorithms that follow a prescribed sequence such as linear scans. Optimization algorithms such as Nelder-Mead simplex [7] and RCDS [8] are difficult to force synchronized evaluations for multiple optimization instances and thus need to be rewritten.

A generic structure for an optimization algorithm suitable for parallel execution in a single-thread may be shown as Fig. 1. The key features here are (1) there is only one function evaluation in one iteration; (2) the function evaluation is outside of the logic flow of the algorithm; (3) the algorithm defines a set of states, and, at each state, it generates the next trial solution and the next state based on previous

```
%s = state, c = config, d = data
i = 0;
while i<MaxEval:
    i = i + 1;
    [x,y,s] = FuncEval(x);
    [x,s,d] = NextState(x,y,s,d,c);
    if convergend:
        break;
    end
end
function [x,s,d]=NextState(x,y,s,d,c):
    d=updateData(d,x,y);
    switch s:
        case A: [x,s,d] = nextA(s,c,d);
        case B: [x,s,d] = nextB(s,c,d);
        ...
    end
end
```

Figure 1: Optimization algorithm structure that separates function evaluation from the algorithm logic flow. Add corresponding sets of variables and 'FuncEval' and 'NextState' lines for each optimization instance.

evaluation data points and algorithm configuration parameters (such as initial step size). The algorithm is embodied through the transitions between the states as prescribed in the function 'NextState'. As an example, the RCDS algorithm has states 'BracketUp' and 'BracketDn', indicating the process of determining the upper or lower limits that bracket the minimum, respectively. It has a state 'FillUp' for adding additional data points within the bracket if necessary and a state 'FindMin' for performing parabolic fitting and calculating the minimum. Similarly, the simplex method has states such as 'Reflection', 'Expansion', 'Contraction', and 'EvalSimplex' (as needed for the shrinking operation).

The RCDS algorithm with a structure as shown in Fig. 1 was previously implemented in the timer function for the AutoTuner program [9]. This gives the GUI program the ability to pause, resume, and stop the tuning process at any time. It has recently been rewritten for parallel online optimization. The simplex algorithm in this structure has also been implemented.

SIMULATION

An example on the SPEAR3 storage ring is used to demonstrate sextupole BBA with parallel online optimization. A group of 8 SF sextupoles from 4 lattice cells with a common power supply is selected. Since the two sextupoles from each cell are close to each other, there are not enough correctors to steer the orbit through them independently. The two sextupoles are considered one magnet and are given the same alignment errors in the simulation. The cells are far apart; the orbit can be independently controlled at the four groups of sextupoles. Therefore, the test case consists of 4 parallel optimization problems. The knobs for each problem are the horizontal and vertical orbits at the target BPM, which is

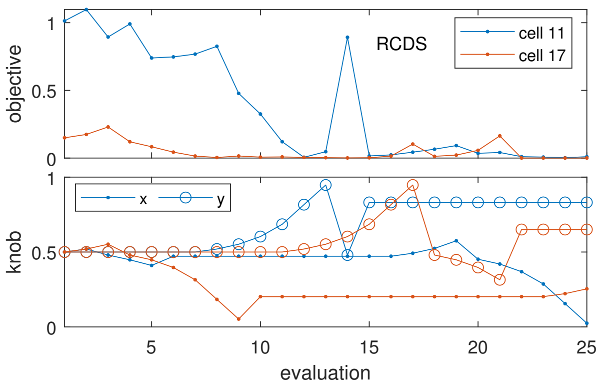

Figure 2: Evolution of objectives and knobs for two out of the four parallel optimization problems optimized with the RCDS algorithm.

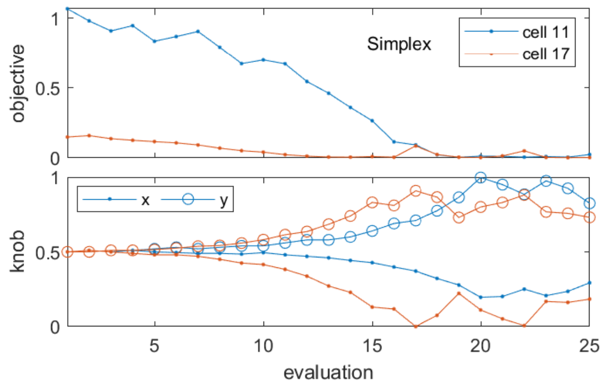

Figure 3: Evolution of objectives and knobs for two out of the four parallel optimization problems optimized with the Nelder-Mead simplex method.

located between the pair of sextupoles. The objective is the sum of squares of the induced kicks, as shown in Eq. (5).

In the simulation, random alignment errors on the order of a few hundreds microns are inserted to the lattice model for the 4 pairs of sextupoles. Orbit steering is done by making local bumps at the 4 BPMs with the help of the orbit response matrix. Reduced weights are applied to the 4 adjacent BPMs of each target BPM (two on each side) to reduce the strengths on the correctors. The steering range is ±1 mm for both planes. The strengths of the sextupole magnets are lowered by 20% for the IOS measurements. BPM noise with sigma of 1 µm is added to all orbit measurements.

Figure 2 shows the objective and knob values during a test run for two out of the 4 problems optimized with the RCDS algorithm. Within 15 evaluations, the objective functions of all 4 problems are reduced to near zero. Figure 3 shows the same two problems optimized with the simplex algorithm. The differences between the optimal orbits from the RCDS optimization and the sextupole centers are below 100 µm. When the beam is steered to the optimal orbit for each pair of sextupoles, the measured IOS is reduced to a very small level, as shown in Fig. 4.

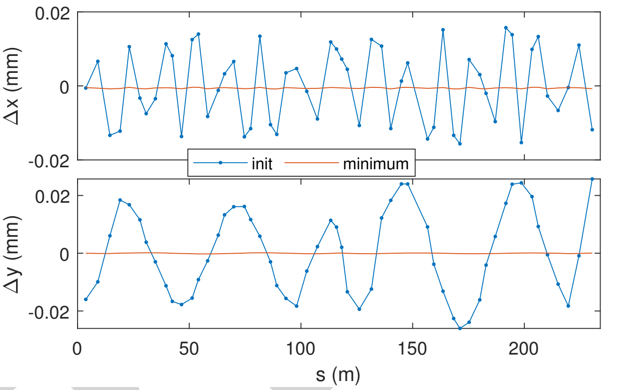

Figure 4: The measured induced orbit shifts before optimization and after the orbits are corrected to the best solutions found by the RCDS optimization.

The particle swarm optimization (PSO) method was also tested for the example. Figure 5 shows the evolution of the objective functions. Each problem has a population of 4 and was run for 7 generations.

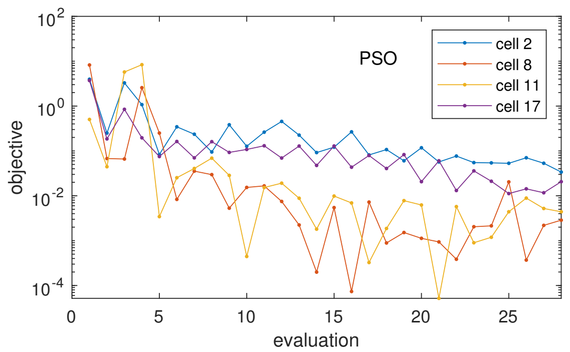

Figure 5: Evolution of the objective functions optimized with the PSO algorithm.

SUMMARY

We propose using parallel online optimization to find magnetic centers of multiple sextupoles simultaneously. In parallel online optimization, multiple optimization instances run independently, but the function evaluations are synchronized. The RCDS and Nelder-Mead simplex algorithms are rewritten in a framework that facilitates parallel online optimization by evaluating the objective only once in every iteration and using proper transitions between the various states to realize the logic.

The horizontal and vertical orbits at the sextupoles are the tuning knobs, while the sums of squares of the horizontal and vertical induced kicks arising from sextupole modulations are used as the objective function.

The approach has been demonstrated in simulation with 4 sextupole pairs on the SPEAR3 storage ring, using RCDS, simplex, and PSO algorithms.

REFERENCES

[1] X. Huang, "Simultaneous beam-based alignment measurement for multiple magnets by correcting induced orbit shift", *Phys.*

Rev. Accel. Beams, vol. 25, no. 5, p. 052802, May 2022. doi:10.1103/PhysRevAccelBeams.25.052802

[2] M. S. et al., "Test of parallel beam-based alignment at nsls-ii", in *Proc. IPAC'24*, Nashville, TN, pp. 3237–3239, May 2024. doi:10.18429/JACoW-IPAC2024-THPC83

[3] S. Liuzzo, L. Carver, L. Valle, N. Carmignani, S. White, and T. Perron, "Parallel beam-based alignment for the EBS storage ring", in *Proc. IPAC'24*, Nashville, TN, USA, pp. 1278–1281, Jul. 2024. doi:10.18429/JACoW-IPAC2024-TUPG27

[4] C. Goffing *et al.*, "Experimental validation of parallel quadrupole beam-based alignment at kara", *Journal of Physics: Conference Series*, vol. 3094, no. 1, p. 012041, Sep. 2025. doi:10.1088/1742-6596/3094/1/012041

[5] X. Huang, "Parallel quadrupole modulation for fast beam-based determination of magnet centers", in *Proc. IPAC'24*, Nashville, TN, pp. 3341–3344, May 2024. doi:10.18429/JACoW-IPAC2024-THPG37

[6] X. Huang, "Beam-based alignment simulations for the future circular collider electron lattice", in *Proc. IPAC'24*, Nashville, TN, pp. 308–311, May 2024. doi:10.18429/JACoW-IPAC2024-MOPG08

[7] J. A. Nelder and R. Mead, "A simplex method for function minimization", *The Computer Journal*, vol. 7, no. 4, pp. 308–313, Jan. 1965. doi:10.1093/comjnl/7.4.308

[8] X. Huang, J. Corbett, J. Safranek, and J. Wu, "An algorithm for online optimization of accelerators", *Nucl. Instrum. Methods Phys. Res, Section A*, vol. 726, pp. 77–83, 2013. doi:10.1016/j.nima.2013.05.046

[9] X. Huang and T. Zhang, "AutoTuner: A General Graphical User Interface for Automated Tuning", in *Proc. IPAC'18*, Vancouver, Canada, Apr.-May 2018, pp. 4939–4942. doi:10.18429/JACoW-IPAC2018-THPML116