ENLARGING DYNAMIC AND MOMENTUM APERTURE BY PARTICLE SWARM OPTIMIZATION

Zhenghe Bai[#], Lin Wang, Weimin Li, NSRL, Hefei, Anhui, CHINA

Copyright © 2011 by IPAC'11/EPS-AG — cc Creative Commons Attribution 3.0 (CC BY 3.0)

Abstract

Particle Swarm Optimization (PSO) is a computational intelligence algorithm for global optimization. Obtaining adequate dynamic and momentum aperture is crucial for high injection efficiency and long beam lifetime in low emittance electron storage rings. Different from nonlinear driving terms optimization, we have made direct optimization of dynamic and momentum aperture by PSO algorithm. It is critical to make criteria for comparison of dynamic and momentum aperture tracking results in the direct optimization procedure. Thus, in this paper we first propose a quantitative criterion of dynamic aperture. Then we apply PSO to the optimization of chromatic and harmonic sextupoles to find the optimal sextupole settings for enlarging the dynamic aperture. Taking the momentum aperture into consideration, we make joint optimization of dynamic and momentum aperture. Also, the momentum aperture has its quantitative criterion. In this paper we also propose some methods for reducing the computation time. As an example of application, the dynamic and momentum aperture of an FBA lattice studied in the design of the storage ring of Hefei Advanced Light Source were optimized, and the results have shown the power of PSO algorithm.

INTRODUCTION

The nonlinear optimization of dynamic aperture (DA) and momentum aperture (MA) is important for obtaining high injection efficiency and long beam lifetime in the lattice design for future light sources and possible upgrades of existing light sources. Minimizing the nonlinear driving terms is widely used for the nonlinear optimization. It has been successful in its application, but the setting of weight factors for the terms is based on the designers' experiences to some extent. Presently, Artificial Intelligence (AI) algorithms for example Genetic Algorithms are applied to nonlinear optimization for some light sources. In [1] we use PSO to optimize harmonic sextupoles for enlarging DA.

There are two important problems in the nonlinear optimization using AI algorithms. One is how to quantitatively describe the objective functions, and the other how to reduce the computation time. In this paper, we will address these two problems.

OPTIMIZATION OF DA

Quantitative Criterion of DA

Here we use the tracking program *elegant* [2] for the simulations of DA and MA. For the DA simulation, the command *bunched_beam* is used. In our cases, the number of tracking particles is 1600, and initially the particles are uniformly distributed in physical space. With so many tracking particles, one can find some small unstable islands, but the calculation consumes much time.

Our proposed quantitative criterion of DA is shown in Fig. 1. *A* is an unstable island where the tracking particles will be lost after a number of turns and *B* is a stable island where the tracking particles can survive. The quantitative criterion is described by the following formula:

$$D = P - k_1 \times |L - R| + k_2 \times E \quad (0 \le k_1 \le 1, k_2 \ge 0). \tag{1}$$

P represents the area of the orange pyramid-like shape; *L* represents the left area (*x*<0) of the *P*, and *R* the right; and *E* represents the area inside the pink half-ellipse.

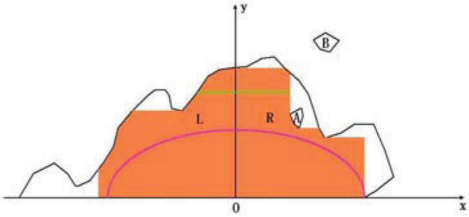

Figure 1: Schematic of the quantitative criterion of DA.

A pyramid-like shape is made in the DA boundary as the useful area of DA. From *x*=0 to both sides, the height of the pyramid-like shape decreases along the boundary. If the height is equal to a predefined value, the length of the pyramid-like shape will stop at this position. So the left "tail" of the DA is clipped off as shown in Fig. 1. We can restrict the height of the pyramid-like shape as the green line in Fig. 1. In such a case, the useful area of DA is the area of the pyramid-like shape below the green line.

The symmetry of DA is important, and the absolute difference of *L* and *R* describes the asymmetry to a certain extent. The pink half-ellipse is the largest half-ellipse inscribed into the DA boundary, with a predefined ratio of the *x*-half-axis length to the *y*-half-axis length.

Our recommended value for the coefficient k_1 is between 0 and 0.5, and k_2 between 0 and 3.

Standard and Constrained PSO Algorithms

Particle Swarm Optimization (PSO) was proposed by Kennedy and Eberhart in 1995. In PSO, each particle updates its velocity and position according to:

$$v_{id} = w \times v_{id} + c_1 \times r_1 \times (p_{id} - x_{id}) + c_2 \times r_2 \times (p_{gd} - x_{id}),$$
$$x_{id} = x_{id} + v_{id}, \tag{2}$$

where x_{id}, v_{id}, and p_{id} are respectively the position, velocity and best position so far of the i^{th} particle in the d^{th} dimension, p_{gd} is the global best position so far, *w* is the inertia weight, c_1 and c_2 are two learning factors, and r_1 and r_2 are two uniformly distributed random variables in [0, 1]. In our cases, *w*=0.729, c_1=c_2=1.49445.

*Work supported by NSFC (10875127 and 10979045)

[#]baizheng@mail.ustc.edu.cn; wanglin@ustc.edu.cn

In our optimization of DA, the objective function is $-D$, and the PSO algorithm minimizes the objective function. The constraints are corrected chromaticity settings. The constraint violations are treated as an additional objective function. In our nonlinear optimization, both the corrected horizontal and vertical chromaticities (C_x, C_y) are set to be in the range [0, 4].

For two solutions with constraint violations less than or equal to ε, the one having lower value of $-D$ is better. If both values of constraint violations are greater than ε, the one having smaller constraint violations is better. If the value of the constraint violations of one solution is less than or equal to ε, and that of the other solution is greater than ε, the one having smaller constraint violations is better.

Optimization Strategies

The DA calculation has a special characteristic. For one solution, if its several-turn DA is good, its hundreds-of-turn DA may be good. But if its several-turn DA is bad, its hundreds-of-turn DA must be bad. Based on this, we proposed the "More Turns and Fewer Turns" (MTFT) strategy for reducing the computation time.

At some iteration of the PSO algorithm, consider *D_best* to be the best quantitative value of DA in the swarm found so far, which is tracked, for example, for hundreds of turns. For one particle, its DA is first tracked, for example, for only several turns. If the quantitative value *D_i* of this tracked DA is less than the value of *D_best* times k_a (*D_i*<k_a×*D_best*), we assign the value of *D_i* times k_b (k_b×*D_i*) as the hundreds-of-turn DA quantitative value of this particle, where k_a is a positive constant about 1 and k_b is a constant in the range $0<k_b<1$, and k_a times k_b is less than 1 ($k_a \times k_b<1$). If *D_i* is greater than or equal to k_a×*D_best*, the particle' DA is then tracked for hundreds of turns to get its hundreds-of-turn DA quantitative value.

Now look at Equation 2. There are two important quantities, p_{gd} and p_{id}, which direct the motion of the i^{th} particle. If at some iteration the particle's DA is first tracked for fewer turns, and its quantitative value is less than that of the DA corresponding to p_{id}, there is no need to track the particle's DA for more turns. The particle's position will not replace its previous best position p_{id} and, of course, also not replace the swarm's previous best position p_{gd}. Thus, the equation of motion for the particle will not change whether the particle's DA is tracked for fewer turns or more turns.

Consider that the DA of particle i is only tracked for fewer turns (because *D_i*<k_a×*D_best*), and its assigned value for more turns is k_b×*D_i*. First we can simply prove that the assigned value is less than *D_best*. Thus, the particle's position will not replace p_{gd}. That also means that the swarm's best DA is always tracked for more turns and its quantitative value is precisely calculated. Then we consider the quantity p_{id}. If k_b×*D_i* is greater than the quantitative value of the DA corresponding to p_{id}, the particle's position will replace p_{id}. But k_b×*D_i* is an assigned value, and its corresponding DA is not tracked for more turns. Thus k_b×*D_i* is not so accurate, and then some perturbations are introduced into the PSO algorithm. But in AI algorithms, to escape from local minima, perturbation methods are usually used. So the above perturbations in PSO are not as bad as they may seem.

If the number of the more turns is *N_M*, and the number of the fewer turns is *N_F*, using the MTFT strategy, theoretically the computation time can be reduced by a factor of *N_M*/ *N_F* at most. But in some our optimizations, from the point of view of the iteration number, it is more than *N_M*/ *N_F*. That means that using the strategy, at the same iteration the number of the particles satisfying the constraints is usually smaller than not using the strategy (because of the perturbations). So it can speed up the iteration process. Empirical study shows that the performance of unconstrained PSO is not sensitive to the population size. We assume that the performance of constrained PSO is not sensitive to the number of the particles satisfying the constraints. Thus the iteration number becomes important. So, using the strategy, the computation time can be reduced much more. And, as analysed and assumed above, in our DA optimization, we found that the PSO algorithm using our proposed MTFT strategy can get as good results as not using the strategy.

The global version of the PSO converges very fast, but is easily trapped into local minima. The local version of the PSO can overcome this problem. Here we use the local version to optimize the DA. In our local version, most local bests satisfy the condition *D_i*⩾k_a×*D_best*, and their corresponding dynamic apertures are tracked for more turns. So perturbations are mainly on particles' personal bests, not on their "leaders" local bests.

To make some relatively good solutions as good "attractors" at the beginning of the algorithm, we proposed the pre-initialization strategy. Before initialization, the same size population as in initialization is repeatedly initialized lots of times. In this process, the dynamic apertures of the solutions satisfying the chromaticity constraints are tracked for only one turn or several turns. At the end of the process, some solutions having better objective function values than others are selected. The number of these best solutions in pre-initialization is equal to the number of local bests. Then, at initialization, the positions of these best solutions are respectively assigned to the positions of the same number of particles. The process of pre-initialization is quick.

Application to an FBA Lattice

As an example of application, we have optimized the DA of an FBA lattice studied in the design of the storage ring of Hefei Advanced Light Source (HALS). The optical functions of one period are shown in Fig. 2. The lattice has 20 super-periods, with emittance of 69pm•rad at 1.5GeV.

In our optimization related to DA, we choose 50 or 100 turns as the more turns, and 4 or 5 turns as the fewer turns. In our experience, this is a good choice if we want to reduce the computation time. After optimization, we can check the obtained optimal solutions with detailed

Copyright © 2011 by IPAC'11/EPS-AG — cc Creative Commons Attribution 3.0 (CC BY 3.0)

tracking and analysis. The coefficient k_b is always 0.8, and the other k_a is slightly greater than or equal to 1.

First we use seven variables of chromatic sextupole strengths to optimize the DA. k_a is set to 1. We optimized the DA without lattice errors, using the local version of PSO with a population size of 10000 and 30 iterations, on a common PC computer (2.50GHz CPU). At pre-initialization, the population was repeatedly initialized 200 times. After optimization, two sextupole strengths are basically zero. So we can employ five families. The variable range for optimal solutions is obtained.

Then we add two harmonic sextupoles to further optimization. So there are seven variables. The searching range of chromatic sextupole strength is set to the range obtained above. To reduce the computation time, k_a is increased to 1.1. The local version of PSO with a population size of 400 ran for 75 iterations on the same PC computer. At pre-initialization, the population was repeatedly initialized 30 times. Lattice errors were included in this further optimization. The DA of one of obtained optimal solutions is shown in Fig. 3 (left plot (a)).

Even though with 1600 tracking particles, either of these two works was done in only three days on the PC computer. It strongly shows the power of our proposed MTFT strategy for reducing computation time.

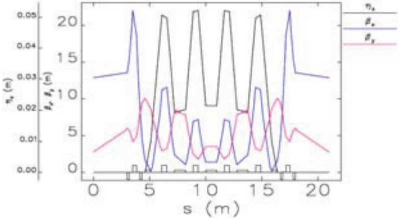

Figure 2: Optical functions of one period of the FBA lattice.

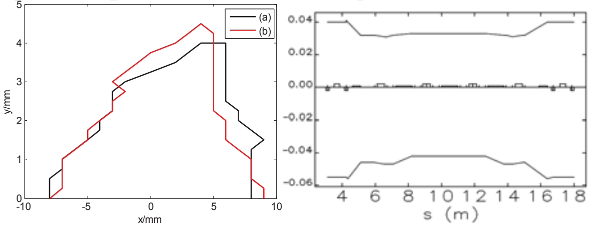

Figure 3: Left: DAs obtained in DA optimization (a) and in joint optimization (b); Right: transverse MA. (with errors)

JOINT OPTIMIZATION OF DA AND MA

In the lattice design, we want to find the lattice not only having good DA but acceptable MA as well. Our proposed quantitative criterion of MA is described by:

$$M = M_p + |M_n| + k_3 \times M_{\min} \quad (k_3 \geq 0), \tag{3}$$

where M_p represents the minimum value of MA along the lattice in the positive momentum deviation direction; $|M_n|$ represents the minimum absolute value of MA in the negative direction; M_{min} represents the minimum between M_p and $|M_n|$. Let M_e be the enough value of MA. If the value of M_p or $|M_n|$ is greater than M_e, the value of M_e is assigned to M_p or $|M_n|$. In our case, the coefficient k_3=0.5.

For one solution, if its DA is not good, we will not choose this solution even though its MA is good. So, in such a case, it is not necessary to calculate the MA. Based on the algorithm above, we introduced another objective function $-M$ to describe MA for the joint optimization of DA and MA. For one solution, if its DA quantitative value is less than a predefined value D_p, its value of $-M$ will be set to a positive constant. If its DA quantitative value is greater than or equal to D_p, its MA will be calculated by tracking. For two solutions in the second case, if they have different values of $-M$, the one having lower value of $-M$ is better; and if they have the same value of $-M$, the one having lower value of $-D$ is better. The predefined value D_p can be decided according to the results of the previous optimization of DA, defining the search region of interest. That is to say, that MA is only calculated by tracking in the region of good DA with quantitative value greater than or equal to D_p.

Here we also use the MTFT strategy to reduce DA computation time. The value of D_best is set to D_p when D_best is greater than D_p. Because $k_a \times k_b < 1$, we can simply prove that if one solution's MA is calculated by tracking, its DA must be tracked for more turns.

In this joint optimization, many parameters are the same as those in the latter case of the previous DA optimization. Lattice errors are also included. Here we only consider the transverse MA, and it is tracked for several tens of turns. After optimization, the optimal solutions are checked with detailed tracking and analysis. M_e is set to 4%. To quickly find optimal solutions, we used the global version of PSO. The algorithm with a population size of 600 ran for 30 iterations on the PC computer, and it consumed about one week.

The optimal solutions obtained in the joint optimization have better momentum apertures compared to those obtained in the previous DA optimization without MA optimized. In the positive direction, the values of M_p of the optimal solutions obtained in the joint optimization are larger by about 1% than those obtained in the previous DA optimization. In the negative direction, the MA values are all basically large enough with values beyond −4% for both cases. The DA and transverse MA of one of obtained optimal solutions are shown in Fig. 3 (left plot (b) and right plot). Then we lowered the value of D_p to simultaneously optimize DA and MA again. We found that the MA can be further improved at the price of the reduction of DA. With our just set up parallel computer, we will perform full 6D tracking for many turns for accurate determination of MA.

REFERENCES

[1] W. Fan, L. Wang and Z. Bai et al., "Physics issues in diffraction limited storage ring design", SCIENCE CHINA Physics, Mechanics & Astronomy (will be published).

[2] Borland, "elegant: A Flexible SDDS-Compliant Code for Accelerator Simulation," APS LS-287, 2000.

Copyright © 2011 by IPAC'11/EPS-AG — cc Creative Commons Attribution 3.0 (CC BY 3.0)