

MAGNET SORTING ALGORITHMS*

D. Dinev

Institute for Nuclear Research and Nuclear Energy
72 Tzarigradsko chaussee, 1784 Sofia, Bulgaria

Abstract

Several new algorithms for sorting of the dipole and/or quadrupole magnets in synchrotrons and storage rings are described. The algorithms make use of a combinatorial approach to the problem and belong to the class of random search algorithms. They use an appropriate metrization of the state space. The phase-space distortion (smear) is used as a goal function. Computational experiments for the case of the JINR-Dubna superconducting heavy ion synchrotron NUCLOTRON have shown a significant reducing of the phase-space distortion after the magnet sorting.

1. INTRODUCTION

In assembling of an synchrotron or storage ring the following problem arises: whether to install the magnets in their locations around the ring in a random way or to try to find an optimum magnet sequence so as to get a better beam quality?

The magnet ordering procedure requires for the dipole or multipole field errors in each individual magnet to be preliminarily measured. The ordering algorithm can minimize the local orbit distortion, the magnitude of several isolated resonance driving terms etc. The sorting is practically cost-free and this is its most attractive feature.

The magnet sorting has been applied for the first time to the TEVATRON to minimize the strongest driving terms of nonlinear resonances driven by sextupoles-[1].

The present paper describes several new algorithms for optimum installing of the dipole and/or quadrupole magnets in synchrotrons and storage rings.

As a large number of harmonics contribute to the aperture limitations it is not enough to minimize few driving terms only. For that reason we have chosen the phase-space distortion (smear) as a figure-of-merit.

Let us have M magnets for installation at M consecutive locations around the accelerator ring.

Let us denote by $k=1,2,\dots,M$ the successive numbers of the magnets and by $j=1,2, \dots,M$ the successive numbers of the locations.

The arrangement of the magnets can be described mathematically by permutations

$$X=(k_1, k_2, \dots, k_M) \quad (1)$$

The permutations (1) compose the combinatorial space of permutations P_x . Its power is $M!$ i.e. is extremely large even for small machines so that a direct enumeration of all the variants is practically impossible.

Let $Q = Q(X)$ be the goal function, nonlinear in the general case. We will discuss the explicit expression of the goal function in the next chapter.

The problem of finding the minimum of the function $Q(X)$ over the points of the combinatorial space P_x is known as the 'nonlinear assignment problem' of the discrete programming-[2] and in cases of large dimension it proves to be a difficult task. That is why new approaches to the problem should be looked for.

2. GOAL FUNCTION

The magnet sorting could be implemented on the base of: a.) dipole component, b.) sextupole component, c.) octupole component of the nonsystematic magnet errors i.e. the goal function whose value is minimized during the sorting could be: a.) closed orbit, b.) magnitude of isolated resonance driving terms, c.) phase-space distortion (smear).

The random normal sextupole component of the dipole field is the most important parameter limiting the accelerator aperture.

In practice the location of the working point in the tune diagram is carefully chosen to be far from any dangerous nonlinear resonance. For that reason we have chosen the distortion of the linearity of the particle motion (smear) as goal function. In other word the magnet sorting will try to find such a magnet sequence that assure almost linear behaviour of the particles within the accelerator aperture.

The beam envelope distortion due to sextupole fields with normalized integral strengths $K'l$ is given by:

$$\begin{aligned} \Delta a_x^s &= \sum_j F_{xj} \cdot (K'l)_{kj} \\ \Delta a_y^s &= \sum_j F_{yj} \cdot (K'l)_{kj} \end{aligned} \quad (2)$$

* Work supported by NSF of Bulgaria under contract F-309

The beam envelope distortion due to octupole fields with normalized integral strength $K''l$ is given by:

$$\begin{aligned}\Delta a_x^o &= \sum_j G_{xj} \cdot (K''l)_{kj} \\ \Delta a_y^s &= \sum_j G_{yj} \cdot (K''l)_{kj}\end{aligned}\quad (3)$$

Functions F_x, F_y and G_x, G_y depend on β, ϵ, Q and μ and their explicit expressions can be found for example in-[3].

We will define the goal function of the magnet sorting as:

$$Q = \text{Max} \{ \Delta a_x^s, \Delta a_y^s, \Delta a_x^o, \Delta a_y^o \} \quad (4)$$

For large machines one must install the magnets in groups due to the limited magnet storage capacity. Let all the magnets be divided to p groups containing $M_p = M/p$ units. In this case we will apply the multi-step optimization. The optimization process could be described as follows. As a first step we will install the magnets of the first group with numbers from 1 to M_p at the first M_p locations in the ring. We will use for that a goal function of type (4) but now the summation over j is from 1 to M_p . Let $\Delta a_{x1}^{s*}, \Delta a_{y1}^{s*}, \Delta a_{x1}^{o*}, \Delta a_{y1}^{o*}$ be the beam envelope distortions corresponding to the optimum arrangement of the magnets of the first group. During the second step we will install the magnets of the second group with numbers from (M_p+1) to $2M_p$. We will take into account the results obtained during the installation of the first group of magnets i.e. the goal function for the second step of optimization process should be:

$$Q_2 = \text{Max} \{ \Delta a_{x1}^{s*} + a_{x2}^s, \Delta a_{y1}^{s*} + a_{y2}^s, \Delta a_{x1}^{o*} + a_{x2}^o, \Delta a_{y1}^{o*} + a_{y2}^o \} \quad (5)$$

The multi-step optimization process goes on until the last p -th group of M_p magnets is installed.

4. SORTING ALGORITHMS

As have been said above the perturbations X -(1) constitute a combinatorial space P_x . The points in this space are all possible permutations X of M elements $\{1, 2, \dots, M\}$. Let us introduce a metric in the P_x space in the following way: the distance $r(X, Y)$ between the point X and the point Y is assume to be equal to the minimum number of transpositions (pair interchanges) necessary to bring point X to the point Y . It can be shown that all the metric properties are fulfilled under such a definition. The metrization of the P_x space allow us to develop algorithms for finding of the optimum magnet sequence.

4.1. Controled Random Search

The algorithm belongs to the class of random search algorithms.

In contrast to the simple random search when at each iteration i one chooses (randomly) a point X_i from the whole combinatorial space P_x and after that determines the smallest value in the set $\{Q(X_i)\}$ in the present algorithm at each iteration i the region of investigation $S(X_{i-1}) \subset P_x$ is shrinked

We can divide $S(X_{i-1})$ in two parts that do not intersect:

$$S(X_{i-1}) = S^+ \cup S^-, \quad S^+ \cap S^- = 0 \quad (6)$$

where

$$S^- = \{Y: Y \in S(X_{i-1}), Q(Y) < Q(X_{i-1})\} \quad (7)$$

If X_{i-1} is the point X_{\min} of global minimum of $Q(X)$ S^- will be empty, otherwise its power is $\neq 0$.

The ratio of shrinking of the region $S(X_{i-1})$ should be such that $S^+ \approx S^-$ i.e. the probability to find a point $Y \in S^-$ should remain large enough.

The algorithm consists in the following steps.

Step 1. Let Q_0 be the smallest upper limit of the goal function $Q(X)$ for $X \in P_x$. Let R_0 be the initial radius ($R_0 \leq (M-1)$) as the largest distance between two arbitrary points of P_x is equal to $(M-1)$. Let N_0 be the initial number of iterations.

Step 2. Let $X \in P_x$ be a point chosen by random sampling with uniform probability distribution function. Let $Q=Q(X)$ be the value of the goal function in the point X . Let $\delta = Q/Q_0$.

Step 3. Let $R = \delta^\alpha R_0$ be the current radius. The value of the parameter α should be experimentally adopted to the investigated goal function. Let $S_R(X)$ be a sphere centred in the point X with radius R . Check whether $R \leq 1$. According to our definition of the metric in the combinatorial space P_x this is the smallest possible radius. If we have reached $R \leq 1$ limit the algorithm stops and we assume the current point X to be the point of minimum X_{\min} . If $R > 1$ we set $N_i = \delta^\alpha N_0$, $n=0$ and go to the step 4.

Step 4. Chose a point $Y \in S_R(X)$ by random sampling with uniform probability distribution. Set $n=n+1$.

Step 5. Check whether $Q(Y) < Q$. If yes go to step 7, if not go to step 6.

Step 6. Check whether $n > N_i$. If yes stop the algorithm and set $X_{\min} = X$, if not return to step 4.

Step 7. Set $X=Y$, $Q=Q(Y)$ and return to step 3.

When the current point X approaches the minimum X_{\min} the number of points $Y: Q(Y) < Q(X)$ diminish i.e. in order to keep the balance between the powers of the sets S^- and S^+ we must shrink the region $S_R(X)$. Shrinking $S_R(X)$ we simultaneously reduce the

number of points for investigation i.e. the necessary CPU time. With proper choice of the shrinking speed the algorithm can avoid the local minima and stops at the global minimum.

4.2. Random Search Making Use of the Decrease Vector

The algorithm belongs to the class of random search algorithms. The idea of the method is at each iteration to restrict the number of checked points i.e. points $Y_j \in P_x$ at which we calculate the values of the goal function $Q_j = Q(Y_j)$ and after that compare these values for finding the minimum to a unite sphere (sphere with radius $R=1$). In fact this unit sphere will be a unit neighbourhood $S_1(X_{i-1})$ of the best point X_{i-1} found at the previous iteration.

An important point is that we will calculate the socalled decrease vector instead of the goal function values themselves. By definition the decrease vector is

$$\Delta(X, Y) = Q(X) - Q(Y) \quad (8)$$

where:

$$R(X, Y) = 1$$

It is an analogue to the gradient in the discrete case. In many cases it is much easier to calculate the decrease vector than the goal function. Thus for the goal function under consideration (4) the decrease vector for Δa_x^s if we have interchanged $k_p \leftrightarrow k_q$ is:

$$\Delta_1 = F_{xp}((K'l)_{k_p} - (K'l)_{k_q}) + F_{xq}((K'l)_{k_q} - (K'l)_{k_p})$$

Similar expressions can be written for $\Delta a_y^s, \Delta a_x^o, \Delta a_y^o$.

The algorithm consists in the following steps.

Step 1. Let $X \in P_x$ be a point chosen by random sampling with uniform probability distribution. Let

$$Q(X) = \text{Max}(\Delta a_x^s(X), \Delta a_y^s(X), \Delta a_x^o(X), \Delta a_y^o(X))$$

Set $i=0$ (iteration counter).

Step 2. Let $S_1(X)$ be a sphere centred in the point X with radius $R=1$. By random sampling with uniform probability distribution choose a point $Y \in S_1(X)$. Calculate $\Delta_1(X, Y); \Delta_2(X, Y); \Delta_3(X, Y); \Delta_4(X, Y)$

$$Q(Y) = \text{Max}(\Delta a_x^s + \Delta_1, \Delta a_y^s + \Delta_2, \Delta a_x^o + \Delta_3, \Delta a_y^o + \Delta_4) \quad (9)$$

Set $i=i+1$.

Step 3. Check whether $Q(Y) < Q(X)$? If yes chose the point Y as the new current 'best' point ($X:=Y$) and set:

$$\Delta a_x^s(X) := \Delta a_x^s(X) + \Delta_1$$

$$\Delta a_y^s(X) := \Delta a_y^s(X) + \Delta_2$$

$$\Delta a_x^o(X) := \Delta a_x^o(X) + \Delta_3$$

$$: \Delta a_y^o(X) := \Delta a_y^o(X) + \Delta_4$$

$$Q(X) = Q(Y)$$

$$i = 0$$

If $Q(Y) > Q(X)$ check whether $i < N_{\text{iter}} = (M-1)M/2$? If yes- return to step 2; if not-stop the iterations and set $X_{\text{Min}} = X, Q_{\text{Min}} = Q(X)$.

In the algorithm we have used that the number of points in unit neighbourhood of X is $(M-1)+(M-2)+\dots+1=(M-1)M/2$.

4. COMPUTATIONAL EXPERIMENTS

As a computational experiment we have studied the case of the JINR-Dubna superconducting heavy ion synchrotron NUCLOTRON-[4]. This synchrotron has 8 superperiods with 32 FODO cells and 96 dipole magnets. The betatron frequencies are $Q_x = 6.71$ and $Q_y = 6.87$.

The rms error levels have been taken as:

$$\sigma(K'l) = 0.09 \text{ m}^{-2}$$

$$\sigma(K''L) = 0.2 \text{ m}^{-3}$$

according to results of the magnet measurements.

Given the distribution of the sextupole and octupole errors in the dipole magnets we were able to reduce by magnet sorting the phase-space distortion to more than one order of magnitude.

REFERENCES

- [1]. H. Edwards. Proceedings of the 12th Int. Conf. on High Energy Accelerators, Fermilab, 1983.
- [2]. H. Taha. Integer Programming: Theory, Applications and Computations. Academic Press, New York, 1975.
- [3]. B. Autin. Nonlinear Betatron Oscillations. AIP Conference Proceedings N 153, p.288, NY, 1987.
- [4]. B. V. Vasilishin et al. Design of Nuclotron Magnet Structure. Preprint JINR-Dubna, 9-86-512, Dubna, 1986.