A NEW WAY OF TUNING BEAM TRANSPORT LINES BY USING DIRECT SEARCH METHODS

JM. LAGNIEL, JL. LEMAIRE,
Laboratoire National Saturne, CEN-SACLAY (France)

ABSTRACT

Finding the best solution given by a set of parameters, in the shortest time, is the goal of any machine designer when it is setted up for the first time and for operating people who take care of the machine every day. As anyone knows, theoretical value of parameters just constitue an approach to pratical problems so we have developped a computer code, operating on-line to make the tuning process easier. The technique used has many advantages.

- It is well suited to solve problems dealing with several parameters (the code has provision for 50),

- There is no need for prior knowledge of the analytic relation ship between parameters and the quantity which is to be optimised,

- No differential calculus and manipulation of matrices or determinants are required. The algorithm used, given by R. Hooke and T.A. Jeeves fits with many optimisation problems encountered during accelerators operation.

INTRODUCTION

Every time an accelerator machine is restarted up, the whole set of parameters which correspond to the required tunning is sent to the equipments and most of the time the values, stored either on a computer disk or on a log book that correspond to the previous identical run, do no lead to the same operating result of the machine. For instance, beam intensity is rather twice low than twice high that before and it is necessary to have the beam back in the same conditions again in the shortest amount of time. At this stage a manual optimisation process run by the operators starts. So we have looked into a way of reducing time spent for starting up, in order to increase the total operating time dedicated to physics experiments and also to optimize the beam quality (transmission of a transport line) when the machine is on, without interrupting the overall operation. Evidently the computer plays an important role in order to achieve the goal of searching for the best solution of a given on-line problem. The scope of this report is limited to transport beam line tuning applications (matching and steering) but we have used successfuly this technique for all the part of the accelerator machine.

SEARCH FOR AN OPTIMUM SOLUTION

1 - STRATEGY

The strategy is to maximize (or minimize) a quantity that we call the criterion. For most of the cases that we are dealing with, the criterion will be a characteristic of the beam at a given location of a beam transport line. This is the result of a measurement on the beam which can be :

- either non destructive (position of the beam, given by position monitors if we are concerned with steering problems, beam size given by beam profilers if we are concerned with matching problems, beam losses given by loss detectors or beam intensity given by current transformers),

- or destructive (beam size given by collimators or flags, beam intensity given by faraday cups, beam emittance given by emittance meter or any scheme combinating different kind of measurements).

The criterion is a function J (Y) where Y is defined by its components $y_1, y_2, .. y_n$ which are the parameters under control. We look for a set of value of parameters located upstream of the beam measurement location which leads to an improvement of the choosen beam property. This parameter belong to a domaine $\mathcal{D} \subset \mathcal{R}^n$ necessarily limited by physical constraints.

The parameters can be bending magnet currents, quadrupole magnet currents, steering magnet currents, electrostatic lens voltages, electrostatic deflecting voltages, RF elements.

The constraints are the minimum and the maximum values that can be applied to the elements of a transport line.

Searching for the OPTIMUM SOLUTION (best solution) from a given starting point Y_0 means that we have to find a set of values $Y_1, Y_2, ..Y_n$ in order to reach the set Y^* which corresponds to the best set of the parameters, resulting from the move.

$$\text{so we have } J(Y_0) < J(Y_1) < \ldots < J(Y^*) = r^*$$

r^* is called the best revenu : this is the value of the criterion when $Y \equiv Y^*$.

From our experience, it may arise that relative maxima (minima) $\tilde{Y}$ exists and if such is the case, the search will stop

$$\text{and we have } J(Y^*) > J(\tilde{Y})$$

This is why we call optimax the very best solution Y^* which is identical to the mathematical solution when a mathematical model of the problem does exist. Indeed, depending upon the starting point this optimax may not be reached if there are relative optima. This limitation does not depend upon the choosen method of iteration and is well known by people concerned with optimisation problems. It can be only improved by taking a different starting point.

The fact of not converging towards the optimax can also be due to the possible strong dependance of the criterion on parameters not under control. For complicated problems such as the ones that we are dealing with, we do not take into account at one and the same time all the possible parameters. There still are uncontrolled parameters that the criterion depends on. Uncontrolled variations of the beam caracteristics, noise in the electronics of the measuring devices lead to fluctuations of the criterion not related to the parameters under control. One has to be aware of that, fluctuations have to be a lot less important than the variation due to the modification of the parameters during the optimisation process. If not the optimisation will be performed on an unstable criterion and will fail. To cope with this difficulty, we have improved the mathematics by calculating a mean value of the criterion over several acquisitions that correspond to one command of a parameter and by making a computer survey of the suspected parameters or devices that are likely to be instable.

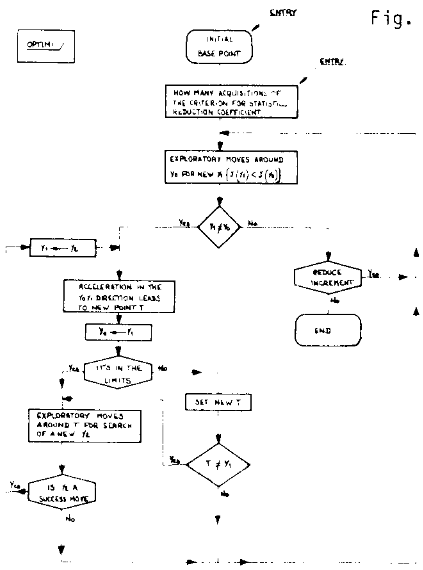


Fig. 1 : Flow diagram

2 - INPUT DATA FOR OPT2

The code is called OPT2 and in its present form it is a Fortran program run on a Mitra computer.

The computer control system developped around the SATURNE accelerator facility has a data base which enables one to know the state and the value of all the parameters and the beam caracteristics at each pulse. They are defined by a name of 6 characters. The link between the computer and the elements is made by the Camac standart. The use of the programm is made from a console display and inputs are as follow ;

input 1 - enter the name of the criterion

input 2 - enter the name of the other physical caracteristics or parameters that are to be surved at each cycle,

input 3 - enter the number of controlled parameters and their names,

input 4 - enter the initial values of specified parameters (can be present values that are automatically read or new different values that are to be sent on the key board),

input 5 - enter the different step value of every parameter,

input 6 - define the specified constraints, if not, limits of power supplies will be taken

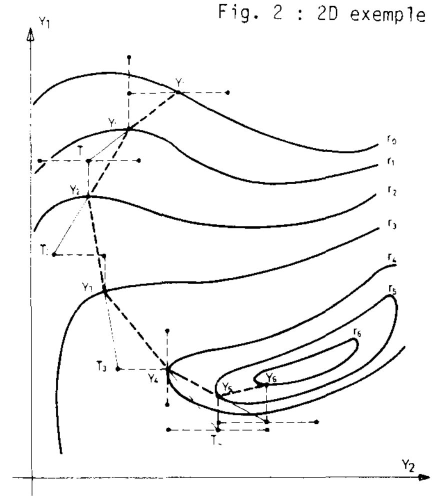
input 7 - enter the coeff CFP (see below) and the number of acquisitions for the criterion and let the program go.

This initialisation is very fast and on a display or/and a hard copier, the operator can follow what the code is doing. If something wrong happens, the program will stop and ask for restoring the initial values or wait for a new order to restart.

DESCRIPTION OF THE PATTERN SEARCH METHOD

The algorithm named "PATTERN SEARCH" by Hooke and Jeeves (ref. 1) was modified in order to deal with problems having constraints. The flow diagram is given on fig. 1 but we will go through a simple 2D exemple to explain the procedure (fig. 2). The dimension of the problem is equal to the total number of the choosen parameters and one has to keep in mind that a real problem will deal with n parameters, usually of the order of 10.

Fig. 2 : 2D exemple for Pattern Search



Y_0 is the initial guess so that a local search starts with the initial steps in order to improve the criterion. The r_m curves are the level curves for different revenus (the optimax is in the vicinity of r_6). The idea is to find a new point Y_1 which improves the criterion while the parameters stay inside the permitted domaine. The parameters are varied by an amount determined by the value of the steps defined by the user (local search around Y_0). After Y_1 has been found (best result around Y_0), point T location is calculated in the Y_0Y_1 direction using a coefficient CFT which correspond to the acceleration of the process.

$$T = Y_1 + CFT\ (Y_1 - Y_0)$$

CFT has been optimised once on several theoretical problems and we have drawn its influence on the fig. 3. The curve is rather flat but values around 0.8 will reduce time spent. We use the value of 0.8 in our code which is a good compromise. This coefficient could be optimised during the search in order to follow very curved ridges.

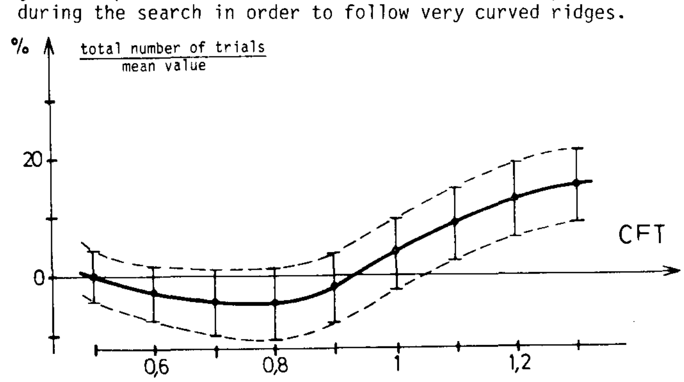


Fig. 3 : optimisation of the coefficient CFT acceleration process

If T is outside the permitted domaine, there will be a modified CFT coefficient that will put the point on the boundary and the direction of the search is saved.

The next step is a local search around T using the same algorithm as for Y_0. If the guess is not a success, the search will start again from the last Y instead of T. If it is a success, there will be an acceleration process to calculate a new T_2 point in the Y_1 Y_2 direction. The procedure is a sequence of acceleration to follow the ridge from Y_0 to Y_4 when it is wide and not curved to much and automatic deceleration when the curvature exists from Y_4 to Y_6.

The variation of the parameters is determined by the initial values of the steps and if the search is not a success after having made all the possible trials, the user will be asked to reduce the steps or to stop. This is the case when the ridge becomes narrower and narrower. The program divides the initial steps by a coefficient CFP which has been also optimised as for the CFT coefficient. On fig. 4 one sees that a reduction of a factor 5 is good enough. In fact, both CFT and CFP coefficients do depend on the specific problem that the user is dealing with and could be optimised at the same time. But it is a question of saving time as much as possible and so far we prefer to use the coefficients calculated in advance.

Notice also that the last successful direction is always kept into memory so that we can reduce the total amount of trials up to 10 % according to the problem.

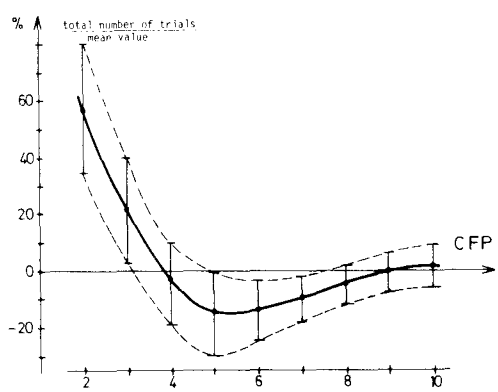


Fig. 4 : optimisation of the coefficient CFP reduction of the step

CONCLUSION

From our experience, this method has the following advantages.

The speed of convergence is comparable to the speed of the fastest algorithms we know about (Rosenbrook). It is a lot more faster than the well-known algorithms (steepest gradient or combined methods).

Commands and acquisitions are directly performed via the computer. It is an appreciable gain of time when the number of parameters is large compared to the operator procedure.

The program is very simple to use from the point of view of software ; the code can be used on any part of the accelerator machine (ref. 2).

This method is very well suited for on-line optimisation. No assumption either on emittances or on linearity of forces are made. There is no need for describing the transport line elements. The only requirement is that the acquisition devices have to be very reliable to avoid the instabilities on the criterion not related to the parameters.

The user has an important part of responsability on the success or the failure of the optimisation process : he is the only one who decides what the best parameters to perform the optimisation are.

REFERENCES

1 - "Direct Search" solution of numerical and statistical problems
R. HOOKE, TA. JEEVES,
Journal of the Association for Computing Machinery, vol. 8, n° 2, 1961

2 - On-line optimization code used at Saturne
JM. LAGNIEL, JL. LEMAIRE,
1ère Conférence on Computing in accelerator design and operation,
Berlin, Septembre 1983