TWO DIMENSIONAL SCANS WITH ADAPTIVE DOMAIN CONTROL

Stefan Adam and Anton C. Mezger
Paul Scherrer Institute (PSI)
5232 Villigen, Switzerland

ABSTRACT

Methods to investigate the combined effect of two parameters being varied, can be helpful for online optimization of the beam setup in an accelerator complex. Compared to single parameter scans, twodimensional scans provide more insight into the way how parameter changes can modify the accelerator performance.
For onedimensional scans it is common practice to monitor the beam losses while scanning and to cut back the interval originally forseen when the beam losses exceed a critical value. In the program described in this paper, the principle of adaptating the interval to be scanned, is extended to the twodimensional case: The domain of allowable parameter variations is built up during the process of scanning. With this new method, twodimensional scanning is not anymore bound to the disadvantage of using extremely narrow scanning intervals, nor does it run the risk of producing too many beam shutoffs.

1. THE NAME OF THE GAME

The program for two-dimensional scanning presented in this paper was given the name "GEISS". This name is neither an acronym, nor does it point back to greek origins, it is simply the Swiss-German name for a goat. The name of this animal was felt to fit the idea of the program, for goats are curious and they are able to find distributed parts of herbs in the mountains without falling down a slope. A short name that is easy to remember can also help to make a program popular with the control crew.

2. THE ORIGINAL SCOPE OF THE PROGRAM

For the control system of the PSI accelerator complex[1),2)] a series of individual programs for onedimensional scanning had been summarized in a single scanning program of general applicability due to selectable parameter-files.

Some programs for twodimensional scanning were in use, but their replacement by a single program appeared to be difficult. These programs had differences in their functionality, in particular with respect to the shape of the twodimensional domain to be scanned. In addition to differences between individual programs, most of the twodimensional scanning programs had a common problem: when the scanning domain was fixed before the start of the scanning, the size of this domain had to be chosen carefully finding a compromise between using only small parameter variations and running the risk of a beam shutoff (interlock). The task to construct a generally applicable program for twodimensional scanning therefore started with the search for a solution to the problem of the scanning domain.

In one dimension the solution is evident: while the active variable is changed stepwise, some passive parameters that give an indication on beam losses are continuously monitored along with the main passive parameter recorded by the scan. If the beam losses become higher than a predefined critical level, the scanning stops, even if the parameter variation has not yet reached the end of the scanning range forseen at the start. Such a scheme of adapting the range while scanning had to be realized in the twodimensional case as well: the scanning domain had to be built-up online.

3. REALIZING THE BASIC FUNCTIONALITY

In most of the cases of twodimensional scanning for parameter optimisation in a accelerator, one can clearly distinguish between the primary and the secondary parameter to be varied e.g. parameter changes of RF devices are much more reproducible than changes of the current in a big magnet. The parameter that is more reproducible is considered as primary parameter, which is changed up and down many times, while the secondary parameter is changed by single steps only. The different role of the two active parameters is the reason why the program "Geiss" is based on a sequence of onedimensional scans which is quite an asymmetric way of treating the two scanning parameters.

The main framework of the program "Geiss" is based on the principle of extendability of a onedimensional scan (see fig. 1).

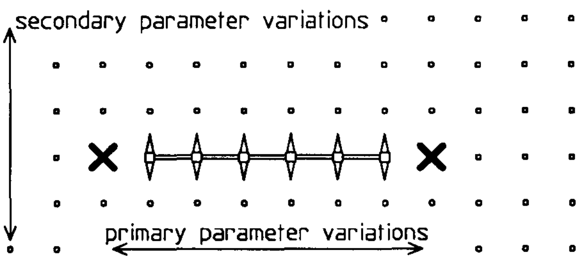

Fig. 1. **Candidates for domain extension.** The changes of the secondary parameter by one step up and down from a successfull point within a line segment of primary parameter scanning are the set of candidates for domain extension from this elementary subdomain.

A onedimensional scan that has been completed, varying the primary parameter, represents a horizontal line-interval in the twodimensional plane, hence an elementary subdomain where scanning has proven to be possible. This elementary subdomain cannot be extended to the left nor to the right hand side, because the onedimensional scan has been done with limit-checking. In the vertical direction however, from each point of this subdomain, an extension upwards and downwards is feasible. Such movements, which represent changes of the secondary parameter by one step up or down, while the primary parameter remains within the range of the scan-line, form the set of candidates for domain extension.

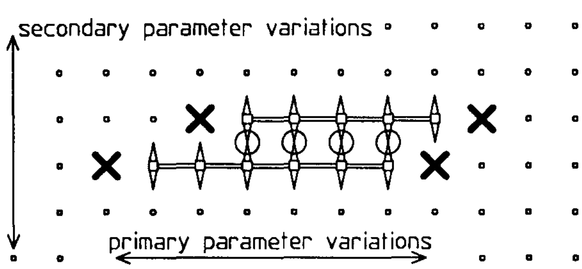

Fig. 2. **Cancellation of candidates pointing towards each other.** Extension candidates that point upwards from the lower scan-line cancel with their counterpart pointing down from the upper scan-line.

During the build-up of the domain in the twodimensional scanning process the program tries to extend the known subdomain using candidates for domain extension. If the elementary movement of domain extension leads to excessive beam losses, the parameter is stepped back, this candidate is eliminated and another candidate has to be tried. If a candidate for extension is successful, it becomes the starting point for a new horizontal (primary parameter) scan.

Two adjacent horizontal scan lines always have some pairs of extension candidates that cancel each other (see fig. 2). These pairs have to be eliminated from the list of candidates after every onedimensional scanning, except for the very first. Candidates of domain extension that have no counterpart from the neighbouring scan, but which point to an endpoint of a neighbouring scan-line have to be eliminated as well because this endpoint, in the previous scan, has proven to lie outside the allowed region. After these bookkeeping operations are done, another attempt to extend the domain via an extension candidate follows. When the list of extension candidates has become empty, the process of twodimensional scanning is complete.

The main working cycle of this twodimensional scanning can be summarized as follows:

A) initialisations
B) do the first primary parameter scan
C) cycle, while there are extension candidates:
   1) try candidates (secondary parameter steps)
      a) discard candidate upon warning
      b) else take as new starting point
   2) do a primary parameter scan (warning limited)
   3) update list of candidates (add new, cancel pairs)
D) display the scanning results

4. FUNCTIONALITY EXTENSIONS

The basic functionality described in the previous section is not sufficient to get a useful program. An elementary additional requirement is that in spite of the automatic domain control based on loss warnings the twodimensional scanning range must be confined. Absolute minima and maxima of the active parameters have to be given and the program has to treat these absolute limits in a different way than the loss warning limits.

The aim to avoid beam interlocks requires that beween two primary parameter scans, when both active parameters are varied, to move to the starting point of the next extension candidate, this travelling path should remain inside the allowed domain. To make this "goodpath" option work, the program has to maintain a table describing the valid domain. This table records all the successful scans sequentially including their vertical positions and their horizontal limits. It is independent of the candidate list which gets empty at the end while the domain table steadily grows.

Another important entry in this domain table is the "father" pointer indicating from which scan line the extension link originated that lead to the present scan line. These pointers bring a tree organisation into the domain table. Travelling inside the domain from one scan line to another corresponds to motions along branches of this tree. The tree of scan lines has an unusual structure. Compared to tree structures used for commercial data processing it has extremely few branching nodes. Most of the nodes in this tree have just one father and one son, but if a branching occurs, there is no principal limit to its multiplicity. The strange structure of the tree of scan lines reflects the fact that this tree is an image of the topology of the domain (see fig. 3).

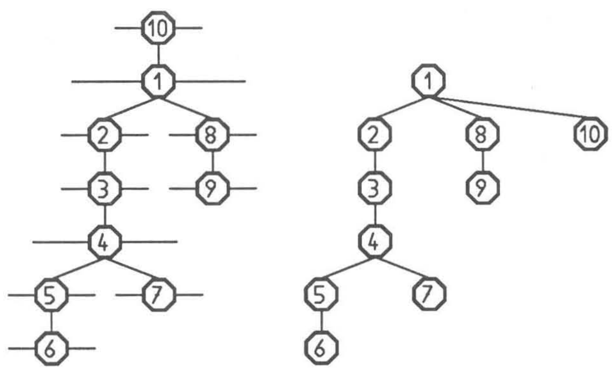

Fig. 3. An example of a domain with a complicated shape and of the corresponding tree structure describing the topology of the interconnections between scan lines.

In the description of the basic functionality a symbolic program step requires "to choose a candidate", but it leaves open how this choice is implemented. Several strategies for selecting the next extension candidate have been tested. Aiming at a minimal number of non-scanning steps, mainly for the secondary parameter, the following scheme appeared to be quite favourable:

The highest priority is given to domain extension candidates starting from a scan line located as near as possible to the present scan line (wrt. to chronological order). If this criterion is still ambiguous, the selection strategy tries to use the same direction for the extension of the domain as for the previous scan. With these two criteria there is still no priority difference among a group of adjacent, parallel extension candidates; in that case the middle one is selected.

5. TEST FUNCTIONS

The program "Geiss" is organized in a way that all scanning and moving operations in the twodimensional space of parameter variations are requested by calling a single function. This central moving and measuring function can therefore be replaced by a test function that gives a similar response to the calling program, but does not vary any real parameter. The requests to change active parameters are just used to update the coordinates of the current point. These coordinates then serve as independent variables in a mathematical formula calculating the value of the passive parameter to be passed back. A loss warning is simulated by the test function when the passive parameter gets larger than a critical value.

Several of these test functions have been used where each one is conceived to demonstrate a different part of the program functionality. The "fourdip" function was the first of those test functions, it was made to test the ability of the program "Geiss" to find the boundaries of a non convex domain (see fig. 4). The functions "nautilus" (with a spiral shaped domain) and "snake" primarily checked the "goodpath" option, requiring that parameter movements must remain inside the valid domain. The "sta–logo" test and also the "snake" are examples that define domains where the connections between scan lines have a complicated topology.

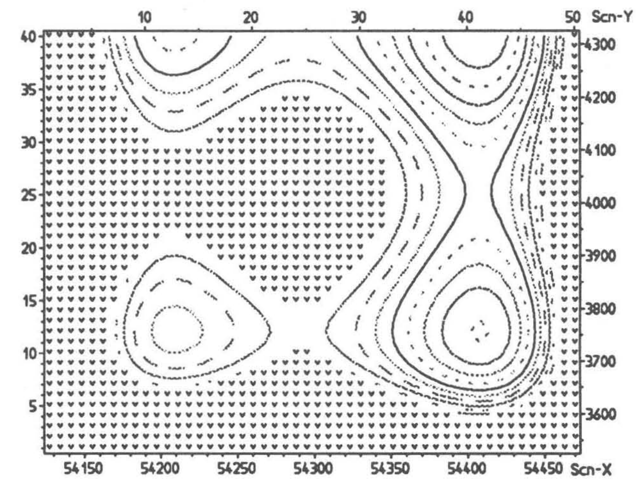

Fig. 4. The domain and the contour lines of a twodimensional simulated scan using the test function "fourdip".

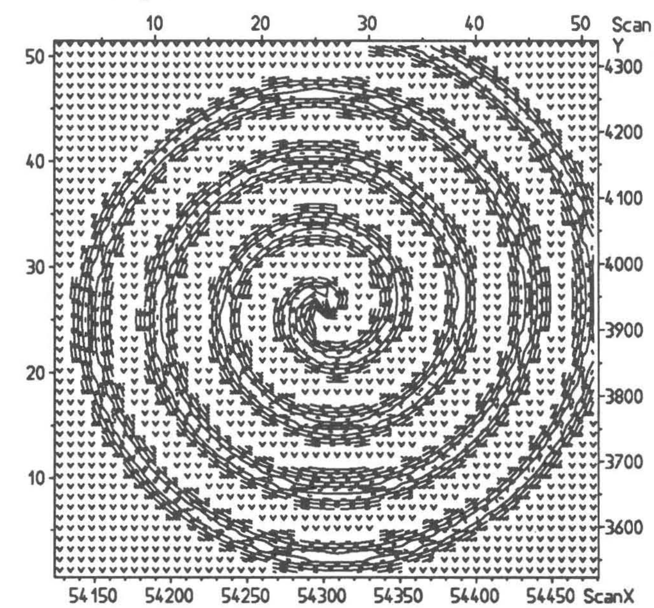

Fig. 5. **A scan with the "nautilus" test function.** Running the program "Geiss" with this test function checks that parameter motions remain inside the valid domain.

6. RUNNING EXPERIENCE AND IMPROVEMENTS

A very early version of the program was tried out with real beam during machine development and was successful in finding the allowed domain avoiding interlocks. After some more test with real beam a wishlist for improvements came up. An urgent requirement was to inhibit multiple scanning of the same domain. In some scanning experiments, the fact that beam losses are not reproducible, and therefore the beam loss limits are varying, made the program scan the same parts many times. This fault was cured by restricting the scanning range not only to the absolute parmeter limits, but to the part of the horizontal line that has not been scanned before. The "diagonal band" test function was programmed to check the multiple scanning inhibition offline. It uses a simple mathematical function, but includes a random function component.

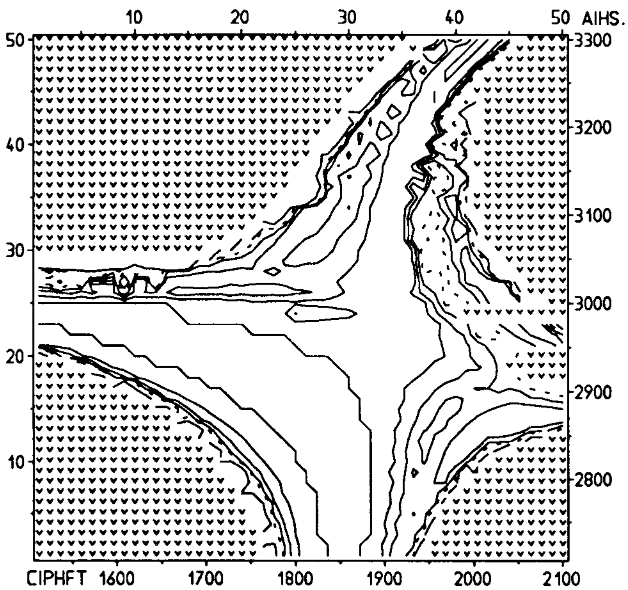

Fig. 6. **A twodimensional scan of flattop phase and main magnet excitation recording extraction losses.** The strange shape of the domain where scanning is possible reveals the advantage of a program that finds this domain automatically.

As twodimensional scanning can take quite some time, a way to abort the running program was introduced. Beam interlocks can occur any time, also in the middle of a running scan. The scan then stops and the operator can decide whether he wants the program to exit or to continue. A special option allows to continue scanning, but avoid the region having been scanned at the moment of the interlock. In studies of extraction losses the contour lines were concentrated at the boundary of the scanned domain giving no information on the loss structure inside the area. This drawback could be avoided using a new procedure to select the contour levels. It makes a statistics on the distribution of measured values and divides the integral curve – number of measured points vs. measured parameter values – into equal slots wrt. to the number of measured points in order to get the set of parameter values for contouring.

In order to reduce the total scanning time without loosing much information an option was introduced that can diminish the number of tries to extend the domain. An entry in the list of extension candidates is deleted by this option when the number of adjacent candidates is smaller than a critical width that can be specified.

An example of a twodimensional scan result with real beam is shown in figure 6.

7. THE USER INTERFACE

Before a twodimensional scan can be started, a long list of options and parameter values must be available to the program. This list includes e.g. the device names for active, passive and loss monitoring parameters, the absolute limits, the critical values for warnings as well as step size and time delay for the active parameters. If the operator would have to type in all this data for each measurement, the program "Geiss" would never be used. Complete sets of these running parameters are stored in individual parameter files, the operator has just to select one of the sample cases from a list and tell the program to start scanning.

Using the multi-window techniques of the OSF-motif (Open Software Foundation) display system on the VAX Workstation, the operator could be given a good overwiew and full control over the activities of the program. Selecting a predefined case, changing critical values, and changing device lists can easily be done with pointing to items in well separated scroll windows. Start and stop of the program and continuation after an interlock can be steered with click buttons.

The continuous display of the domain accumulated so far and of the current parameter values (position in the domain) had been forseen for testing only, but as twodimensional scanning can be quite slow this display was kept for the application case. It steadily informs the operator on what the scan is doing.

8. CONCLUSIONS

The twodimensional scanning program "Geiss" has become a good tool for beam optimisation. The method to build up the valid domain while scanning has proven to work well and makes the program much more useful than previous ones with a fixed scanning domain. The window based user interface and the online graphic display are essential to the success of this program.

9. REFERENCES

1) Blumer, T. et al., "The PSI Accelerator Control System: A Status Report." Proc of the Int. "Conf. on Accelerator and Large Experimental Physics Control Systems", Vancouver BC, Canada, 1989
2) Stammbach, Th., "Experience with the High Current Operation of the PSI Cyclotron Facility." this conference.