BNL-104870-2014-TECH

AGS/AD/Tech Note No. 454;BNL-104870-2014-IR

# BNL MAD: Harmonic Analysis Commands

J. Niederer

November 1996

Collider Accelerator Department

**Brookhaven National Laboratory**

**U.S. Department of Energy**

USDOE Office of Science (SC)

# DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or any third party's use or the results of such use of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof or its contractors or subcontractors. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

*For Internal Distribution Only*

Accelerator Division
Alternating Gradient Synchrotron Department
BROOKHAVEN NATIONAL LABORATORY
Upton, New York 11973

Accelerator Division
Technical Note

**AGS/AD/Tech. Note No. 454**

# BNL MAD: Harmonic Analysis Commands

J. Niederer

November 13, 1996

# BNL MAD: Harmonic Analysis Commands.

*J. Niederer*

AGS Department
Brookhaven National Laboratory

November 13, 1996

## 1. Introduction

Two methods of computing coefficients of a Fourier series for sets of particle orbit data points are provided by the **Vharm** command of the local BNL version of the MAD accelerator program. One is a conventional least squares fit of a sine and cosine series to the set of data points. The second method is a less conventional constrained amplitude subtraction of individual harmonics that is less sensitive to missing items or regions of the sampled data set. Input values can be irregularly spaced: spacing is not restricted. Both methods accept sets of measurements or program output expressed in vector format, at given locations S, with optional weights. Two data sets, nominally X and Y values at their respective sampled locations, can be treated by one command. A range of harmonics consistent with the number of data points may be selected. A fixed term is always computed, which for circular orbits is an overall average displacement due to off momentum of the beam. The fits are iterated to remove the biasing effect of ostensibly missing data upon the fixed term. In successive iterations, missing data points are computed from the fitted results of previous iterations. Output is to printed tables of inputs and fits, and optionally to an internal harmonic table which can be directed to histogram plots of harmonics, synthesis plots, and individual harmonics plots. Harmonic commands can also be applied to data sampled at monitors (or elsewhere) of Twiss generated orbits.

BNL MAD service facilities can be used to display and check results. One kind of fairly demanding test is to put a pattern composed of known harmonics on the corrector set, compute the orbit, and compare the resolution by the harmonics commands of the pattern recorded on the orbit monitors with the original pattern. Adding more orbit distortions can be a further challenge to the resolving power.

## 2. Constrained Fits

In the familiar problem, a pattern of I measurements Vi taken at locations Si over a distance Smax is to be fitted to a harmonic series:

$$V_i = A_0 + \sum_{j=jmin}^{j=jmax} A_j \cos(2\,\pi\, j\, S_i\,/\,S_{\max}) + \sum_{j=jmin}^{j=jmax} B_j \sin(2\,\pi\, j\, S_i\,/\,S_{\max})$$

When treated as an extremal problem, the set of Aj, Bj are found by minimizing the difference at each point between the measurement and the fitted function. The measurements may also be weighted individually, noted by Wi. The function to be minimized is:

$$\sum_{i=1}^{i=I} W_i\,(V_i - A_0 - \sum_{j=jmin}^{j=jmax} A_j \cos(2\,\pi\, j\, S_i\,/\,S_{\max}) - \sum_{j=jmin}^{j=jmax} B_j \sin(2\,\pi\, j\, S_i\,/\,S_{\max}))^2$$

This expression is differentiated with respect to each of the Aj, Bj, leading to a set of linear equations which can be solved conventionally for the Aj, Bj. We call this the general method here. The solutions can be simplified if the spacing between points is uniform, using orthogonal properties of the resulting sines and cosines.

The constrained method selects for each harmonic j an even (A) and odd (B) coefficient that produce the largest reduction of the above penalty function. Initially, the squares of the difference between each point of the data set and the computed contribution from harmonic Aj, Bj are summed to obtain the penalty. The program searches in an Aj, Bj space for this reduction. The Aj and Bj pairs must satisfy the relation

$$A_j^2 + B_j^2 < Constraint * Residual\_Amplitude^2$$

Trial sets of Aj and Bj amplitude pairs are cycled, over a grid that is reduced over succeeding passes. First an Aj is cycled over 20 intervals of da, where da is formed from the largest residual amplitude as:

$$da \ = \ \sqrt{Constraint} \ * \ Abs\,(R_{max)} \ / \ 20$$

Bj is taken as zero during this first cycle in Aj. Penalty functions of residuals are computed, and a best Aj candidate is selected from among those tried. The interval da is halved about the best Aj, the penalty calculation repeated for the two values Aj - da and Aj + da, and Aj further localized. Then Bj is similarly cycled in steps db over a range subject to the constraint, using the best Aj. This process is repeated a few times, soon converging. A final quadratic interpolation in the neighborhood of the best Aj, Bj pair is carried out. The contribution of this Aj and Bj is subtracted off each data point, reducing the Vi. The is process repeated for the next, and remaining harmonics. When finished, the Vi residuals are a measure of the fit over the given range of harmonics. The sum of the squares of the Vi, the penalty function, is now in a sense a least squares for the fit.

Somehow, this procedure works. For Booster orbits, there is always a clean minimum in both Aj and Bj as the cycling is carried out. The results do not cluster about the end points of the constrained ranges of A or B, as if a preferred result were outside of the range. The order in which the Aj, Bj are evaluated and subtracted off does not seem to matter. It is less sensitive to missing points, and does not encounter the artifacts of the usual Fourier analysis which tend to produce spurious values outside the edges of the region, and in the holes at the missing points. However, it is not at all clear how general it may be.

Several options are provided to help test the validity of the fits. In the permute option, the order of processing of the harmonics can be scrambled. The fits of a set of harmonics are repeated in a pattern which begins with a pivot harmonic, *jmain,* followed by the next larger one, then by the next lower one, until the range has been covered. The pivot is then advanced by one, and the next version of the pattern repeated. For low order harmonics, the method has appeared insensitive to the order in which harmonic contributions are subtracted. Under the drop mode option, the fits can be cycled over a pattern which drops one of the data points, performs the fit, restores the dropped point, and repeats with the next point dropped. A dropped point that leads to a markedly better fit may cast some doubt on the point. Results may have some minor sensitivity to the dropped point in the overall orbit averaging, but for the most part they do not have the wild swings often produced by the general least squares methods.

The constrained method is somewhat slower because of the relatively large amount of searching, especially if the permute or drop mode options are used. For reasonable search patterns, and with debugging code, a 24 point data set iterated 10 times takes about 1/6 second on a 100mips workstation.

The constrained method can behave dramatically better than others when applied to distorted data samples. In our tests, such as removing four adjacent data points, the original pattern was reproduced quite well, although it lacked some detail in the section missing data. Most other methods will distort the fitted function increasingly with the separation between data points. The conventional least squares method, achieving about the same or a little smaller least square with its solution, may generate large overshoots in the region among the missing points, and individual amplitudes can become much larger than any of the data points. Results over a span of a single period or less can be particularly unrealistic beyond the edges of the fitted region, as Fourier methods tend to shift effects from (unfitted) higher harmonics into the edge regions, a phenomenon called aliasing. Plots of the computed amplitudes along the orbits can be quite helpful in judging the quality of the fit.

## 3. The Vharm Command

The **Vector Harmonics** command analyzes data arrays expressed as vectors. Vectors are identified by name in the inputs. Several options describe the range of sampling and outputs. In the permute mode, the fits are repeated with different sequences for the subtraction of the several harmonic contributions. In the drop mode, the fits are repeated in a sequence which drops one input point at each step. The drop feature offers a way to examine potentially bad monitors, aid in data selection, and explore sensitivity of fits to particular monitors. Data selection can be obtained dynamically with the filter option and selective weighting.

### 3.1. Input Data Vectors

Xvector

The *Name* of a simple array of measurements, for example, a set of Bpm readings. Either X or Y (or both) must be given, and the lengths of the vectors (number of entries) must agree. The array of decimal values is defined as a MAD vector. Units should agree with those of the *Svectors* position data below. Units conversion can be arranged with the use of the *Convert* attribute noted below. Erratic input values may be rejected by the use of the *Filter* attribute noted below.

Yvector

Sxvector

The *Name* of a simple array of the decimal values of positions at which the above measurements were taken, also a MAD vector form. These positions need not be equally spaced. If both X and Y data as given above are taken at the same locations, only one *Svector* need be given. The number of entries in these vectors must agree with each other and with those of the measurement vectors.

*Svector* entries are not multiplied with the *Convert* feature. (Use the CMult operation of the **Vserve** command for this conversion if needed.)

Syvector

Xweights

The *Name* of a simple decimal array of weights, also a MAD vector form. Optional. The default of unit weights will be used if no weight vectors are given. If individual weights are used, each weight should be the inverse square of the nominal mean error for the measurement.

Yweights

Xlabels

Optional. The *Name* of an optional simple list of data point label names. (List, Menu, or Xmenu form) These names will be used to label entries in least squares fit tables, and should be suitable for plotting the contents of monitor data vectors s well.

Ylabels

### 3.2. Necessary Parameters

Smax

A REQUIRED decimal number, given in the same units as the *Svectors* above, normally meters. This is the total length over which the input measurement set has been sampled. *Smax* defines the full period over which harmonics are to be fitted. It is the length of a circular orbit when a set of Bpm values is analyzed. Phases for each harmonic J at positions S(i) are computed as:

$$Theta(j,i) = 2. * \pi * J * S(i) / Smax$$

Constraint

A decimal number which constrains the choice of Aj and Bj coefficients of each trial according to

$$A_j^2 + B_j^2 < Constraint * Residual\_Amplitude^2$$

The supplied default is 1.2. In our trials, results were usually rather insensitive to values within a range of about .3 to 1.5.

Dropmode

Logical Flag: If True, the fitting will be iterated, dropping a different input reading on each successive pass. Output may be sent to the *Stacktab* Table, a stacked set of results from each pass. Computation time increases as the number of data points in the sample.

Permute

Logical Flag: If True, the fitting will permute the order of cycling harmonics. Computation time increases as the number of harmonics in the sample.

Filter

A decimal number which should be used to screen accelerator produced data such as monitor readings. Measurements greater than the absolute value of *Filter* will be rejected for fitting.

Type — The *Name* of the type of fit. Values of "COnstrained", "GEneral", or "BOth" are accepted. The default is the faster general least squares fit to coefficients.

Cycles(2) — 1. An integer giving the number of iteration cycles, which tend to adjust the fixed A0 term, and may influence others. Successive iterations make shake up the fitting, producing more attractive fits, which may however differ little in final least squares criteria from others. The proper value should be obtained by trial, viewing the listing which traces the penalty function, hopefully to some convergence. The default is set for 3, which is probably too small for most cases.

The method computes a fit once without the missing points, and subsequently includes values derived from the previously fitted function at the missing points to compute the next iteration. For the Booster, the contributions from the two missing monitors out of 24 tend to balance each other.

2. An integer giving the number of internal cycling passes used in searching the local grid for the best Aj, Bj pairs. Default is 3, which has proved adequate for Booster work.

Convert — An optional decimal number, that if present, is used as a conversion factor for all X and Y Vector input values. (Orbit measurement data)

### 3.3. Harmonic Selection Options

Harmonics are indexed here by the letter J. The lead term, J = 0, (A0), is always fitted, and is not included in the following J ranges.

Jmain — An optional integer. The number of the central harmonic to be fitted, if *Jmin* and / or *Jmax* are not given. *Jmin* and *Jmax* will be defined in terms of the number of measurements in the *Xvector* as

Jmax = Jmain + jx

Jmin = Jmain - jx

Nmeas > 2 * (Jmax - Jmin) + 1

Jmin — An optional integer. The lowest harmonic to be fitted. If given, *Jmax* will be constrained so:

Nmeas > 2 * (Jmax - Jmin) + 1

Jmax — An optional integer. The highest harmonic to be fitted. If given, and *Jmin* is not given, *Jmin* will be constrained so

Nmeas > 2 * (Jmax - Jmin) + 1

The program may reduce the range *Jmin, Jmax* actually used to agree with the number of data points accepted after filtering and non zero weight criteria have been applied. For these cases *Jmax* is adjusted downwards. The use of these J attributes provides alternatives, such as dropping the low order harmonics instead.

### 3.4. Output Options

The option of a simple histogram of fitted coefficients is attached to the **Vharm** command. More elaborate plotting of the fitted function may be done by including a **Pharmon** drawing command in the menu of a **Plcalls** screen plotting command.

Htable — The *Name* to be given to the internal table that receives the Fourier coefficients fitted to the measurement vectors. Other analyses and display programs need to refer to this table by its name. The table will be saved for possible further use only if an *Htable* name is given.

Plot — Logical Flag: If True, a screen histogram of the normalized amplitudes of each harmonic display will be presented. (SGI, or X Windows Version) This plotting needs a reference to a **Plotdef** command, and associated picture setup, such as given on the example

"Hplots.boos" call file.

Plotdef — The *Name* of a **Plotdef** command keyed to the Fourier coefficient histograms. A *Plotdef* is needed if the *Plot* option is selected.

Print — Logical Flag: If True, a table of harmonic coefficients will be printed.

Verbose — Logical Flag: If True, an extended trace of fitting progress will be printed. This trace can be helpful for evaluating sensitivity.

Stacktab — The *Name* of an internal table which is to receive repeated groups of coefficients, under the iteration options such as *Dropmode* and *Permute.* In effect it is a number of individual harmonic coefficient tables stacked together.

### 3.5. Others

/lsqc(2) — These two cells receive the final X and Y values of the least squares penalty function for the constrained type of fit. If more than one cycle has been performed, results from each cycle overwrite those of the previous, so only the last cycled values are preserved. As a part of each **Vharm** command data object in the data base, these values may be reached by any of the data logging, parameter change, and display facilities of the program.

/lsqg(2) — Similarly these two cells receive the values of final penalty functions for the general type of fit.

/status — An internal cell used to mark the command status after the last use of the command.

/stamp — An internal cell used to mark the last use of the command.

## 4. Plotting

Plots are available through the **Plcalls** plot manager command. Pictures may show individual harmonics and the buildup of the fitted function as harmonics are added. **Plcalls** is a Silicon Graphics based menu driven display, that can also be connected to orbit computations and display commands, and to utilities that enable parameter changes and comparisons. Curves computed from tables *(Htable* attribute) of harmonic coefficients produced by **Vharm** commands may be displayed by the **Pharmon** drawing command, which may be attached to the plot manager. Original orbit values in the measurement vectors, Twiss generated orbits, and orbits computed from the harmonic fits may also be displayed on the same picture. The program may also be compiled with an alternative X Windows based version for similar graphics.

## 5. Results

Several representative screen plots obtained by sampling data from AGS Booster orbits are shown in Figures 2 - 7. This particular data, from an off momentum orbit of 1993, was selected to test the extraction of the proper momentum from orbit data. These examples are intended to illustrate the methods rather than to offer conclusions from the particular data sampled. **Pharmon** command options display amplitudes and phases of individual harmonics, and the buildup in the form of curves of sums of harmonics Jmin --> N, as N is increased from Jmin through Jmax. Monitors are missing at D6 and F6, and the point from monitor C8 is rather obviously suspect, which further strains the resolving power. A comparison between fitting methods for a case with a major hole in the sampling distribution is also included. While the resulting Figures 2 and 3 are busy, they do show that even with this marginal data, the individual harmonics maintain their essential character. Correlated distortions do appear to cause a strong second and third harmonic in this particular set of readings. The spurious harmonics are weaker in similar curves run without the C8 entry, but such efforts to clean up the data are not considered further here. The seventh harmonic, basically an artifact of the fitting, is very small except in the generalized fit for the sample with readings deleted. In other tests, not presented here, a seventh harmonic put on the correctors, along with arbitrary displacements on a few lattice elements, was properly resolved.

In these Booster data samples, the harmonic analysis averages only broadly as the orbit is measured at essentially regular intervals, rather widely spaced with respect to dimensions of likely orbit distortions, The Booster is perhaps a worst case in that the number and spacing of monitors barely covers the range of orbit harmonics present in the machine. Details of finer structure are obviously limited by the number of sampling points. (Figure 7) A few X plane monitors between these intervals, perhaps by adding an X coordinate to several of the Y plane monitors, would be very helpful.

A subset of statements copied from a larger test file which applies the harmonics commands discussed here is appended. While complicated, as a library of useful commands the file is readily edited for new applications. This style permits tailoring of graphics from input statements, rather than hard coding the morass of details about what is to be drawn. The Figures which follow are generated from these inputs as well. Parts of the resulting PRINT file of the **Vharm** fits to this sample data are also included. Fitted coefficients are tabulated for the four cases of general and constrained, with and without the hole in the data. Also listed is the progress of the constrained fit during the iterations (5), and a playback of the fit to each of the data points, for the hole case (6).

### Documentation

Unix troff typeset format

| | |
|---|---|
| **Host:** | **rapt.ags.bnl.gov** |
| This report: | /usr/disc2/jn/Docum+/Vharm.man |
| Plcalls: | /usr/disc2/jn/Docum+/Plcalls.man |
| Vharm Test File: | /usr/disc2/jn/Docum+/Vharm.ex |
| Vharm Results: | /usr/disc2/jn/Docum+/Vharm.pr |

```
          PRINT FILE - Booster Harmonic Fitting    (Edited)

   1.        General Method, for 22 Booster Orbit Points

RAW FOURIER STRENGTHS IN X, Y
   N      FREQ(N)    ABX(N)     AX(N)     BX(N)          ABY(N)     AY(N)     BY(N)
   0      0.00000   3.05889  -3.05889   0.00000         0.00000   0.00000   0.00000
   1      0.03114   1.68133   1.67280   0.16917         0.00000   0.00000   0.00000
   2      0.06228   1.18530  -1.15888   0.24883         0.00000   0.00000   0.00000
   3      0.09342   1.90317   1.86981  -0.35483         0.00000   0.00000   0.00000
   4      0.12456   2.57949  -2.48282   0.69954         0.00000   0.00000   0.00000
   5      0.15569   4.47817   4.01490  -1.98358         0.00000   0.00000   0.00000
   6      0.18683   1.86091   1.33022   1.30134         0.00000   0.00000   0.00000
   7      0.21797   1.26415   1.24470  -0.22093         0.00000   0.00000   0.00000
   8      0.24911   1.45056   0.17537   1.43992         0.00000   0.00000   0.00000


NORMALIZED FOURIER STRENGTHS IN X, Y
   N      FREQ(N)    ABX(N)     AX(N)     BX(N)          ABY(N)     AY(N)     BY(N)
   0      0.00000   0.68307  -0.68307   0.00000         0.00000   0.00000   0.00000
   1      0.03114   0.37545   0.37355   0.03778         0.00000   0.00000   0.00000
   2      0.06228   0.26468  -0.25879   0.05557         0.00000   0.00000   0.00000
   3      0.09342   0.42499   0.41754  -0.07923         0.00000   0.00000   0.00000
   4      0.12456   0.57601  -0.55443   0.15621         0.00000   0.00000   0.00000
   5      0.15569   1.00000   0.89655  -0.44294         0.00000   0.00000   0.00000
   6      0.18683   0.41555   0.29705   0.29060         0.00000   0.00000   0.00000
   7      0.21797   0.28229   0.27795  -0.04933         0.00000   0.00000   0.00000
   8      0.24911   0.32392   0.03916   0.32154         0.00000   0.00000   0.00000


   2.      General Method, for 18 Booster Orbit Points


RAW FOURIER STRENGTHS IN X, Y
   N      FREQ(N)    ABX(N)     AX(N)     BX(N)          ABY(N)     AY(N)     BY(N)
   0      0.00000  10.00887 -10.00887   0.00000         0.00000   0.00000   0.00000
   1      0.03114  11.43978  -8.72791  -7.39542         0.00000   0.00000   0.00000
   2      0.06228  12.28985  -4.43206 -11.46286         0.00000   0.00000   0.00000
   3      0.09342  11.53079   6.29038  -9.66386         0.00000   0.00000   0.00000
   4      0.12456   7.29351   6.60916  -3.08454         0.00000   0.00000   0.00000
   5      0.15569  12.06186  12.05369   0.44376         0.00000   0.00000   0.00000
   6      0.18683   9.40161   5.94817   7.28076         0.00000   0.00000   0.00000
   7      0.21797   5.07646   1.68715   4.78790         0.00000   0.00000   0.00000
   8      0.24911   4.53911  -1.51888   4.27745         0.00000   0.00000   0.00000

NORMALIZED FOURIER STRENGTHS IN X, Y
   N      FREQ(N)    ABX(N)     AX(N)     BX(N)          ABY(N)     AY(N)     BY(N)
   0      0.00000   0.81440  -0.81440   0.00000         0.00000   0.00000   0.00000
   1      0.03114   0.93083  -0.71017  -0.60175         0.00000   0.00000   0.00000
   2      0.06228   1.00000  -0.36063  -0.93271         0.00000   0.00000   0.00000
   3      0.09342   0.93824   0.51184  -0.78633         0.00000   0.00000   0.00000
   4      0.12456   0.59346   0.53777  -0.25098         0.00000   0.00000   0.00000
   5      0.15569   0.98145   0.98078   0.03611         0.00000   0.00000   0.00000
   6      0.18683   0.76499   0.48399   0.59242         0.00000   0.00000   0.00000
   7      0.21797   0.41306   0.13728   0.38958         0.00000   0.00000   0.00000
   8      0.24911   0.36934  -0.12359   0.34805         0.00000   0.00000   0.00000
```

3.     Constrained Method, for 22 Booster Orbit Points

```
RAW FOURIER STRENGTHS IN X, Y
   N       FREQ(N)    ABX(N)     AX(N)     BX(N)            ABY(N)     AY(N)     BY(N)
   0       0.00000   3.12466  -3.12466   0.00000           0.00000   0.00000   0.00000
   1       0.03114   1.74163   1.72798  -0.21759           0.00000   0.00000   0.00000
   2       0.06228   0.94677  -0.75396  -0.57265           0.00000   0.00000   0.00000
   3       0.09342   2.69081   2.64719  -0.48255           0.00000   0.00000   0.00000
   4       0.12456   2.52353  -2.31849   0.99638           0.00000   0.00000   0.00000
   5       0.15569   4.36602   4.03325  -1.67183           0.00000   0.00000   0.00000
   6       0.18683   1.95039   1.01054   1.66819           0.00000   0.00000   0.00000
   7       0.21797   0.64884   0.57710  -0.29657           0.00000   0.00000   0.00000
   8       0.24911   1.56170  -0.46891   1.48965           0.00000   0.00000   0.00000


NORMALIZED FOURIER STRENGTHS IN X, Y
   N       FREQ(N)    ABX(N)     AX(N)     BX(N)            ABY(N)     AY(N)     BY(N)
   0       0.00000   0.71568  -0.71568   0.00000           0.00000   0.00000   0.00000
   1       0.03114   0.39890   0.39578  -0.04984           0.00000   0.00000   0.00000
   2       0.06228   0.21685  -0.17269  -0.13116           0.00000   0.00000   0.00000
   3       0.09342   0.61631   0.60632  -0.11052           0.00000   0.00000   0.00000
   4       0.12456   0.57799  -0.53103   0.22821           0.00000   0.00000   0.00000
   5       0.15569   1.00000   0.92378  -0.38292           0.00000   0.00000   0.00000
   6       0.18683   0.44672   0.23146   0.38208           0.00000   0.00000   0.00000
   7       0.21797   0.14861   0.13218  -0.06793           0.00000   0.00000   0.00000
   8       0.24911   0.35769  -0.10740   0.34119           0.00000   0.00000   0.00000
```

4.         Constrained Method, for 22 Booster Orbit Points

```
RAW FOURIER STRENGTHS IN X, Y
   N       FREQ(N)    ABX(N)     AX(N)     BX(N)            ABY(N)     AY(N)     BY(N)
   0       0.00000   3.05902  -3.05902   0.00000           0.00000   0.00000   0.00000
   1       0.03114   1.68076   1.67223   0.16918           0.00000   0.00000   0.00000
   2       0.06228   1.18563  -1.15917   0.24911           0.00000   0.00000   0.00000
   3       0.09342   1.90298   1.86962  -0.35474           0.00000   0.00000   0.00000
   4       0.12456   2.58024  -2.48340   0.70024           0.00000   0.00000   0.00000
   5       0.15569   4.47756   4.01451  -1.98301           0.00000   0.00000   0.00000
   6       0.18683   1.86132   1.33044   1.30171           0.00000   0.00000   0.00000
   7       0.21797   1.26456   1.24520  -0.22043           0.00000   0.00000   0.00000
   8       0.24911   1.45075   0.17609   1.44002           0.00000   0.00000   0.00000

NORMALIZED FOURIER STRENGTHS IN X, Y
   N       FREQ(N)    ABX(N)     AX(N)     BX(N)            ABY(N)     AY(N)     BY(N)
   0       0.00000   0.68319  -0.68319   0.00000           0.00000   0.00000   0.00000
   1       0.03114   0.37537   0.37347   0.03779           0.00000   0.00000   0.00000
   2       0.06228   0.26479  -0.25888   0.05564           0.00000   0.00000   0.00000
   3       0.09342   0.42500   0.41755  -0.07923           0.00000   0.00000   0.00000
   4       0.12456   0.57626  -0.55463   0.15639           0.00000   0.00000   0.00000
   5       0.15569   1.00000   0.89658  -0.44288           0.00000   0.00000   0.00000
   6       0.18683   0.41570   0.29714   0.29072           0.00000   0.00000   0.00000
   7       0.21797   0.28242   0.27810  -0.04923           0.00000   0.00000   0.00000
   8       0.24911   0.32400   0.03933   0.32161           0.00000   0.00000   0.00000
```

5.      Progress of Iterations, Constrained Method, 18 Booster Orbit Points

```
    Harmonics Program.  Starting Value for WDsq =      753.100   Mean Dev. =       5.8508
Iteration  1,   Penalty      54.4274,     Change    -698.6727,   A0       -3.1020, Dav        -3.1020
Iteration  2,   Penalty      51.0878,     Change      -3.3396,   A0       -2.9829, Dav         0.1191
Iteration  3,   Penalty      50.6543,     Change      -0.4335,   A0       -3.0083, Dav        -0.0254
Iteration  4,   Penalty      50.4471,     Change      -0.2072,   A0       -3.0252, Dav        -0.0169
Iteration  5,   Penalty      50.3468,     Change      -0.1002,   A0       -3.0365, Dav        -0.0113
Iteration  6,   Penalty      50.2977,     Change      -0.0491,   A0       -3.0440, Dav        -0.0075
Iteration  7,   Penalty      50.2733,     Change      -0.0244,   A0       -3.0490, Dav        -0.0050
Iteration  8,   Penalty      50.2610,     Change      -0.0123,   A0       -3.0524, Dav        -0.0033
Iteration  9,   Penalty      50.2548,     Change      -0.0063,   A0       -3.0546, Dav        -0.0022
Iteration 10,   Penalty      50.2515,     Change      -0.0032,   A0       -3.0561, Dav        -0.0015
Iteration 11,   Penalty      50.2498,     Change      -0.0017,   A0       -3.0571, Dav        -0.0010
Iteration 12,   Penalty      50.2490,     Change      -0.0009,   A0       -3.0578, Dav        -0.0007
Iteration 13,   Penalty      50.2485,     Change      -0.0005,   A0       -3.0582, Dav        -0.0004
Iteration 14,   Penalty      50.2482,     Change      -0.0003,   A0       -3.0585, Dav        -0.0003
Iteration 15,   Penalty      50.2481,     Change      -0.0001,   A0       -3.0587, Dav        -0.0002
Iteration 16,   Penalty      50.2480,     Change      -0.0001,   A0       -3.0588, Dav        -0.0001
Iteration 17,   Penalty      50.2480,     Change       0.0000,   A0       -3.0589, Dav        -0.0001
Iteration 18,   Penalty      50.2480,     Change       0.0000,   A0       -3.0590, Dav        -0.0001
Iteration 19,   Penalty      50.2480,     Change       0.0000,   A0       -3.0590, Dav         0.0000
Iteration 20,   Penalty      50.2480,     Change       0.0000,   A0       -3.0590, Dav         0.0000
```

6.      Summary Table for X Fits.  Constrained Method, 18 Booster Orbit Points

```
 N                          S  V Measured    V Fitted  Difference     Penalty    Weight
 1  A2                   5.08       3.956       2.538       1.418    2.011989      1.00
 2  A4                  13.49      -8.694      -6.660      -2.034    4.136658      1.00
 3  A6                  21.90      -4.039      -6.090       2.052    4.209449      1.00
 4  A8                  30.30       1.900       3.224      -1.324    1.753799      1.00
 5  B2                  38.71       1.212       1.122       0.091    0.008239      1.00
 6  B4                  47.12      -4.886      -5.984       1.098    1.205077      1.00
 7  B6                  55.53      -7.631      -5.992      -1.638    2.683562      1.00
 8  B8                  63.93      -1.533      -2.734       1.201    1.442453      1.00
 9  C2                  72.34       2.786       2.704       0.082    0.006714      1.00
10  C4                  80.75       0.946       2.620      -1.673    2.800266      1.00
11  C6                  89.16      -7.147     -10.029       2.883    8.309948      1.00
12  C8                  97.56     -19.705     -16.500      -3.205   10.268968      1.00
13  D2                 105.97      -4.802      -7.361       2.559    6.547286      1.00
14  D4                 114.38      -1.615      -0.315      -1.299    1.688672      1.00
15  D6                 122.79       0.000      -0.456       0.000    0.000000      1.00
16  D8                 131.19      -0.625      -1.470       0.845    0.714320      1.00
17  E2                 139.60      -5.475      -4.418      -1.057    1.117659      1.00
18  E4                 148.01      -4.605      -5.405       0.801    0.641111      1.00
19  E6                 156.42      -2.491      -2.066      -0.425    0.180490      1.00
20  E8                 164.82      -0.376      -0.595       0.218    0.047646      1.00
21  F2                 173.24      -2.166      -1.944      -0.223    0.049672      1.00
22  F4                 181.64      -4.314      -4.551       0.238    0.056558      1.00
23  F6                 190.05       0.000      -4.717       0.000    0.000000      1.00
24  F8                 198.45       1.058       1.664      -0.606    0.367411      1.00
      Total WDsq =    50.247948   Mean Dev. =       1.5113   20 Iterations.
```

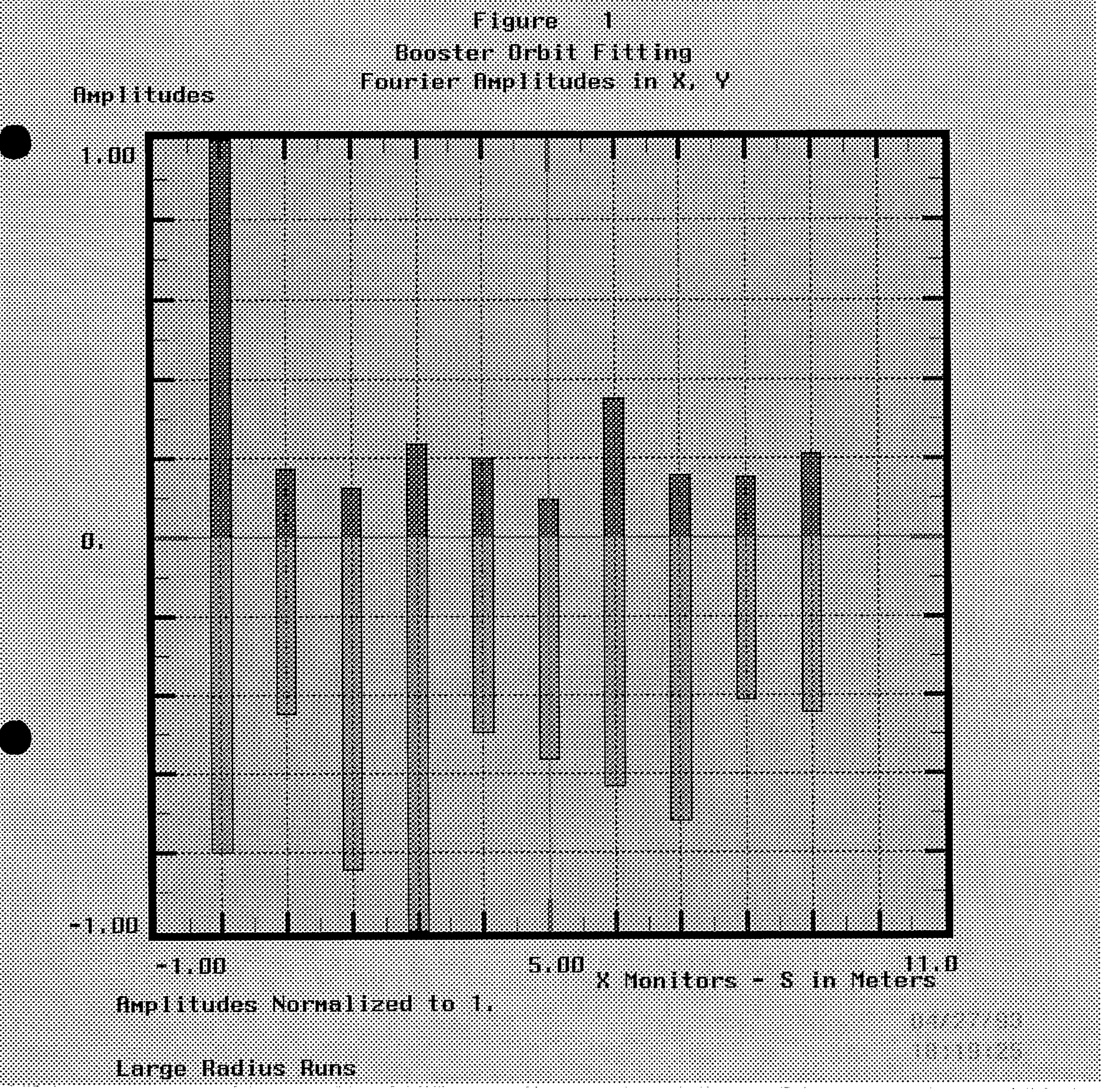

**Figure 1.**

A simple histogram of amplitude magnitudes normalized relative to the largest one, A0, using the *Plot* option. Data is from an AGS Booster orbit snapshot taken in April of 1993 with a large horizontal A0 coefficient corresponding to a large off momentum run. The normalization here is rather misleading, as it is dominated by the A0 term. Horizontal values are drawn above the center line, and vertical ones below. The actual Y amplitudes are small. The shading attempts to imitate the various colors of the original.

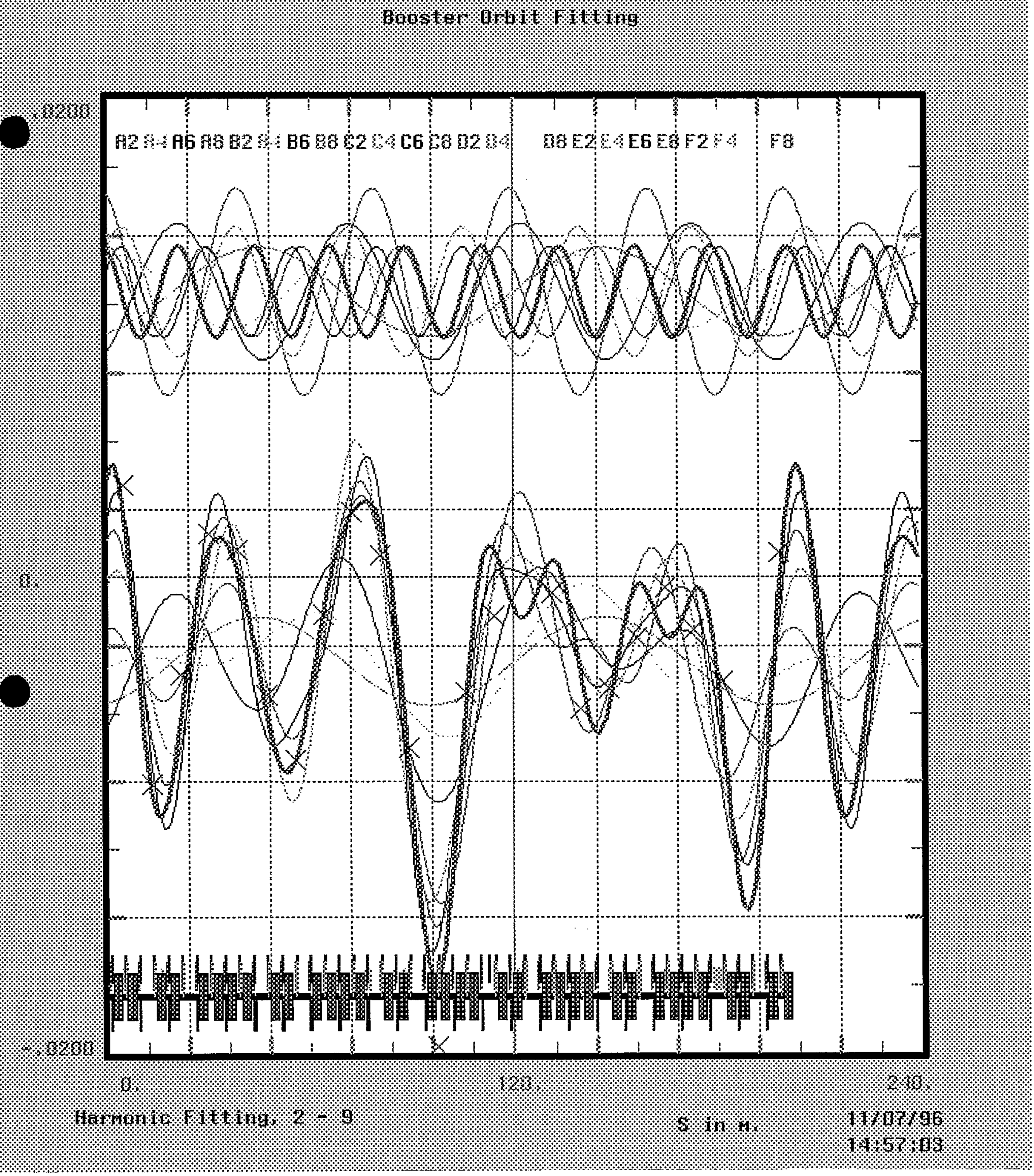

**Figure 2.**

An example harmonics analysis plot produced by the Pharmon drawing command from another set of fitted Booster off momentum orbit data. The upper set of curves display the amplitude of each individual harmonic. The lower series of curves shows the buildup of the fitted function, in the form of the sum of the first J harmonic amplitudes, one curve each for J = 2 through 9. (17 Coefficients from 22 Data Points) The heavier solid curve is the final result, for J = 9. Measurement data points are shown as crosses. Both general and constrained fits have very similar curves for this case. Both scales are in meters. The original in color distinguishes much better among the harmonics.

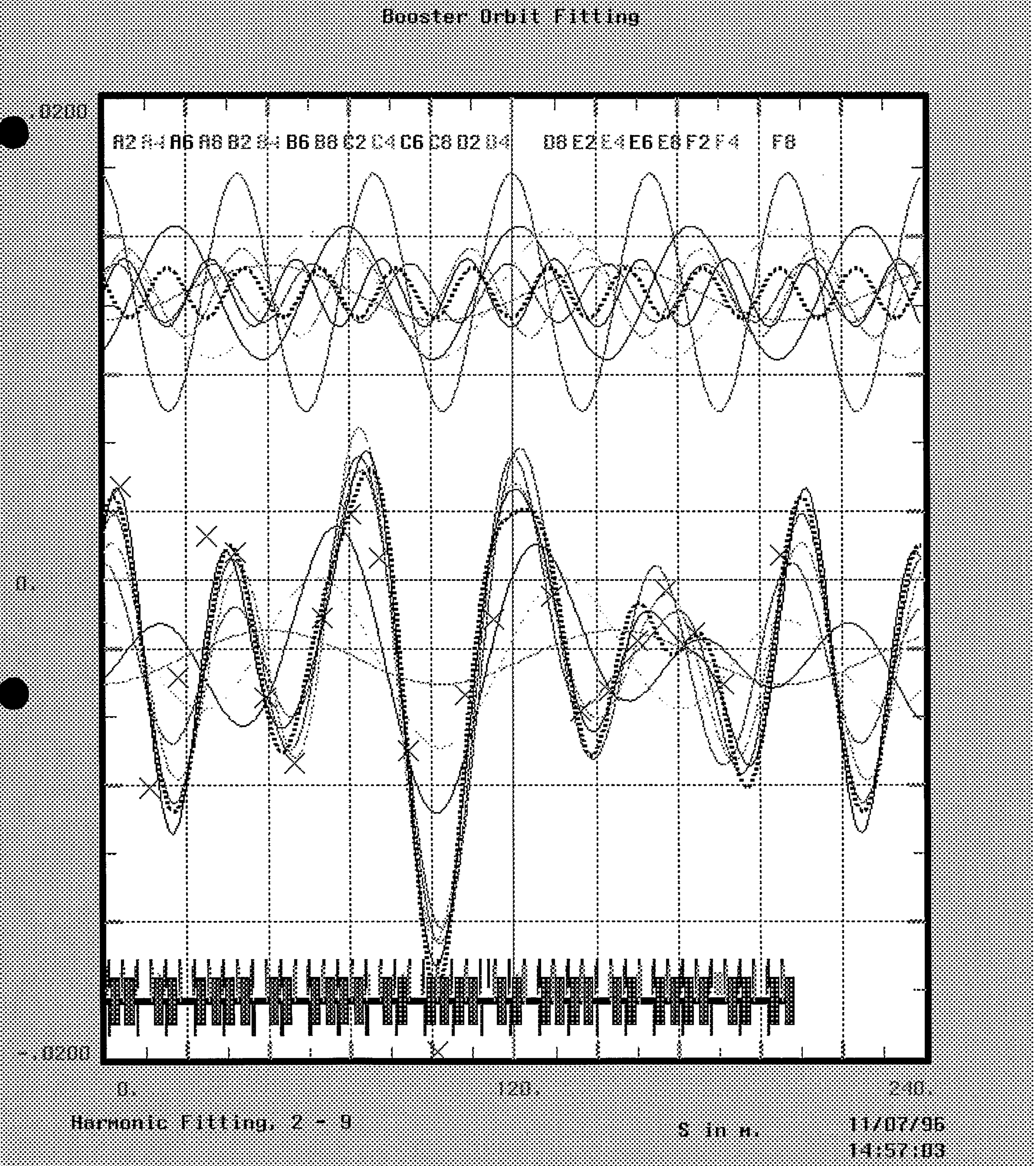

**Figure 3.**

Constraint method curves similar to those of Figure 2 showing the fitted function when a section of data is removed from the sample. Data points 2 - 5 (Monitors A4 - B2) have been removed. The region about the first six monitors is largely preserved, but there are also differences about the missing points D6 and F6, although probably consistent with the errors in the data. Curves are extended beyond the actual orbit boundary to look for artifacts of the fits.

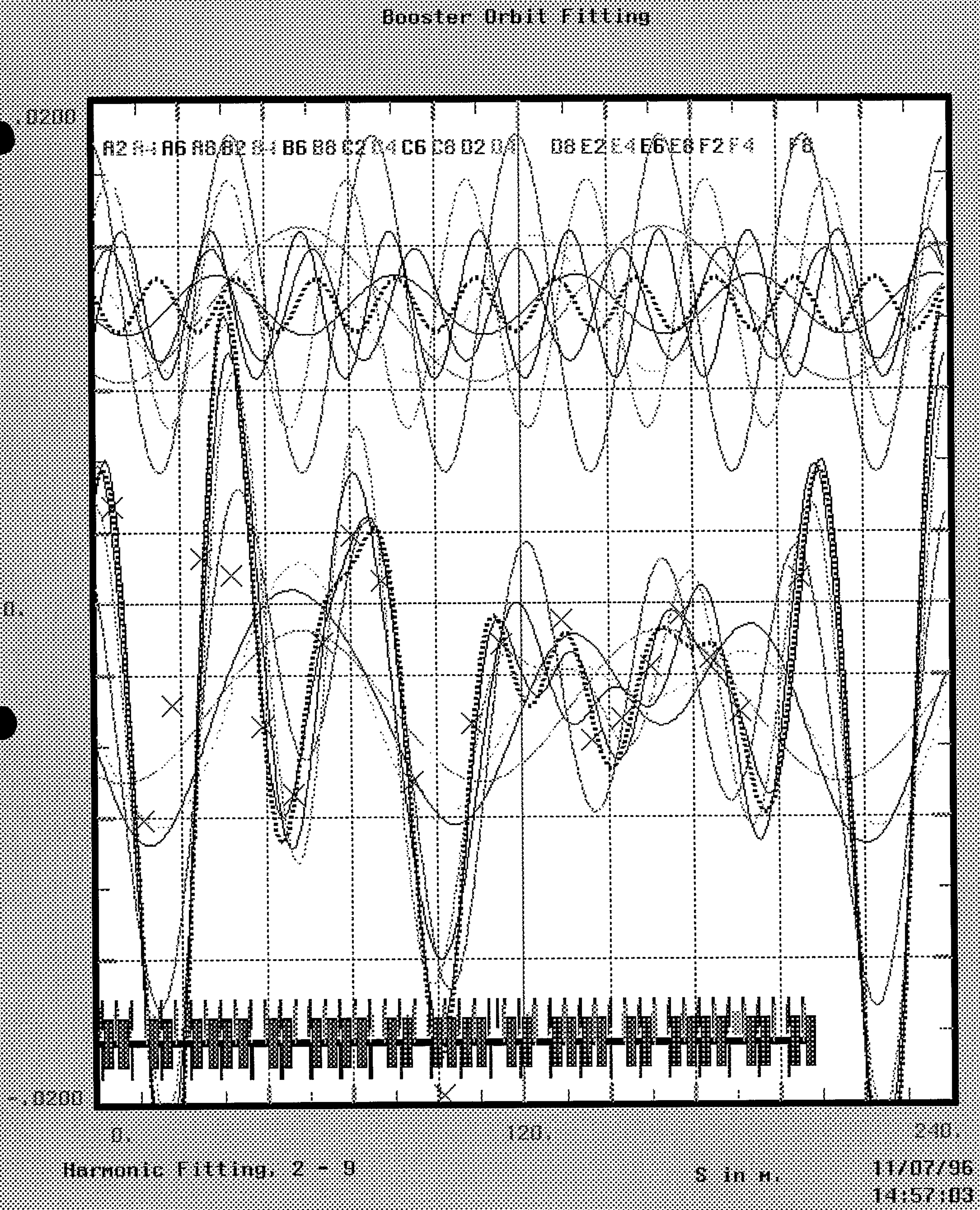

**Figure 4.**

Curves showing the general least squares fit for the case of points 2 - 5 removed from the data sample, corresponding to Figure 3 above. Individual amplitudes are much larger and there are substantial overshoots in the beginning region. This is one of the better examples; using J = 1 through 8 instead leads to a real mess.

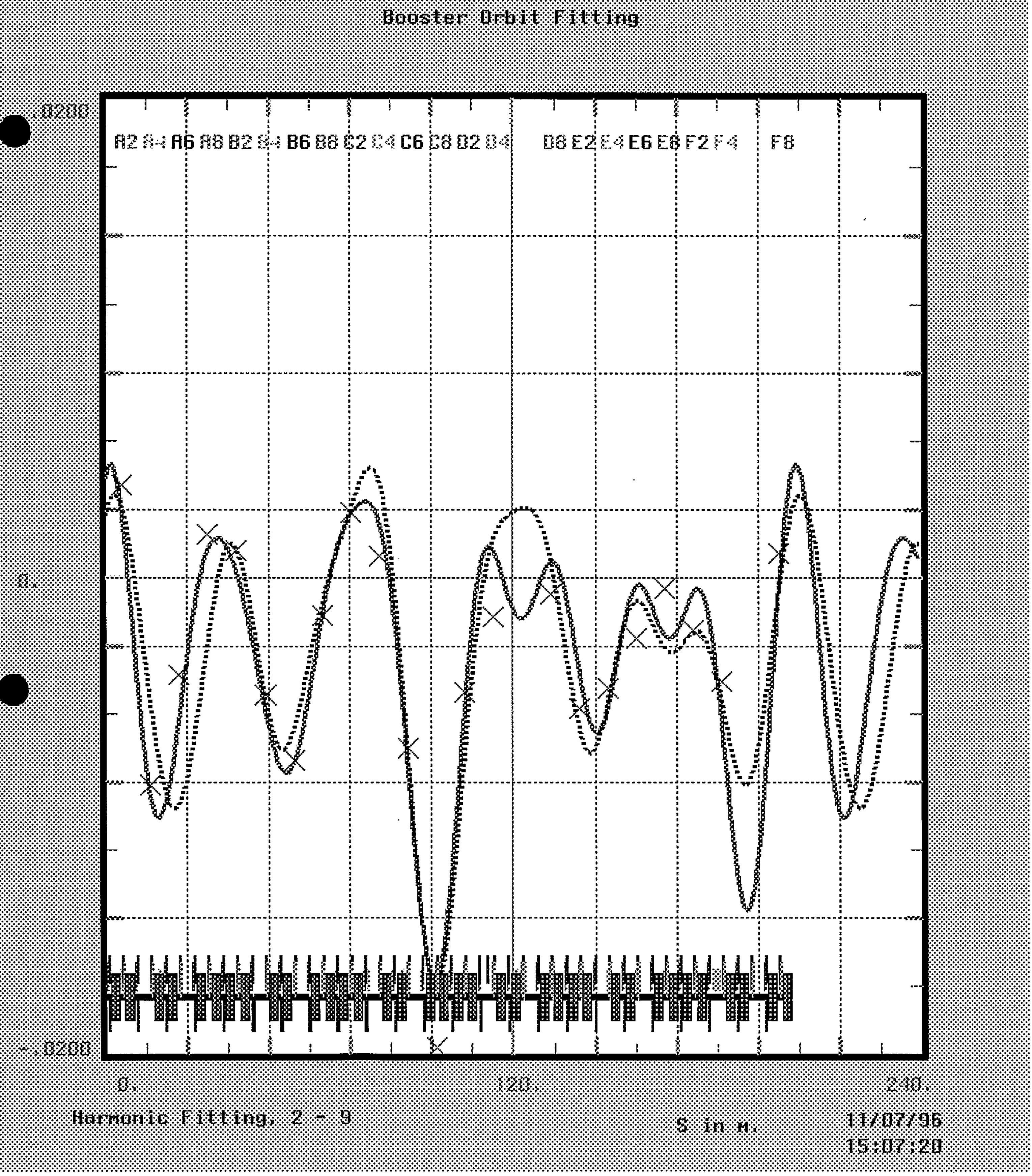

**Figure 5.**

The final synthesis curves from Figures 2 and 3, comparing the constrained type of fit for the case with all 22 points involved (solid curve) and with points 2 - 5 removed (dashed curve). Harmonics J = 0, 2 - 9 are included. The average deviation for these fits is about 1.2 - 1.5 mm, strongly influenced by point C8.

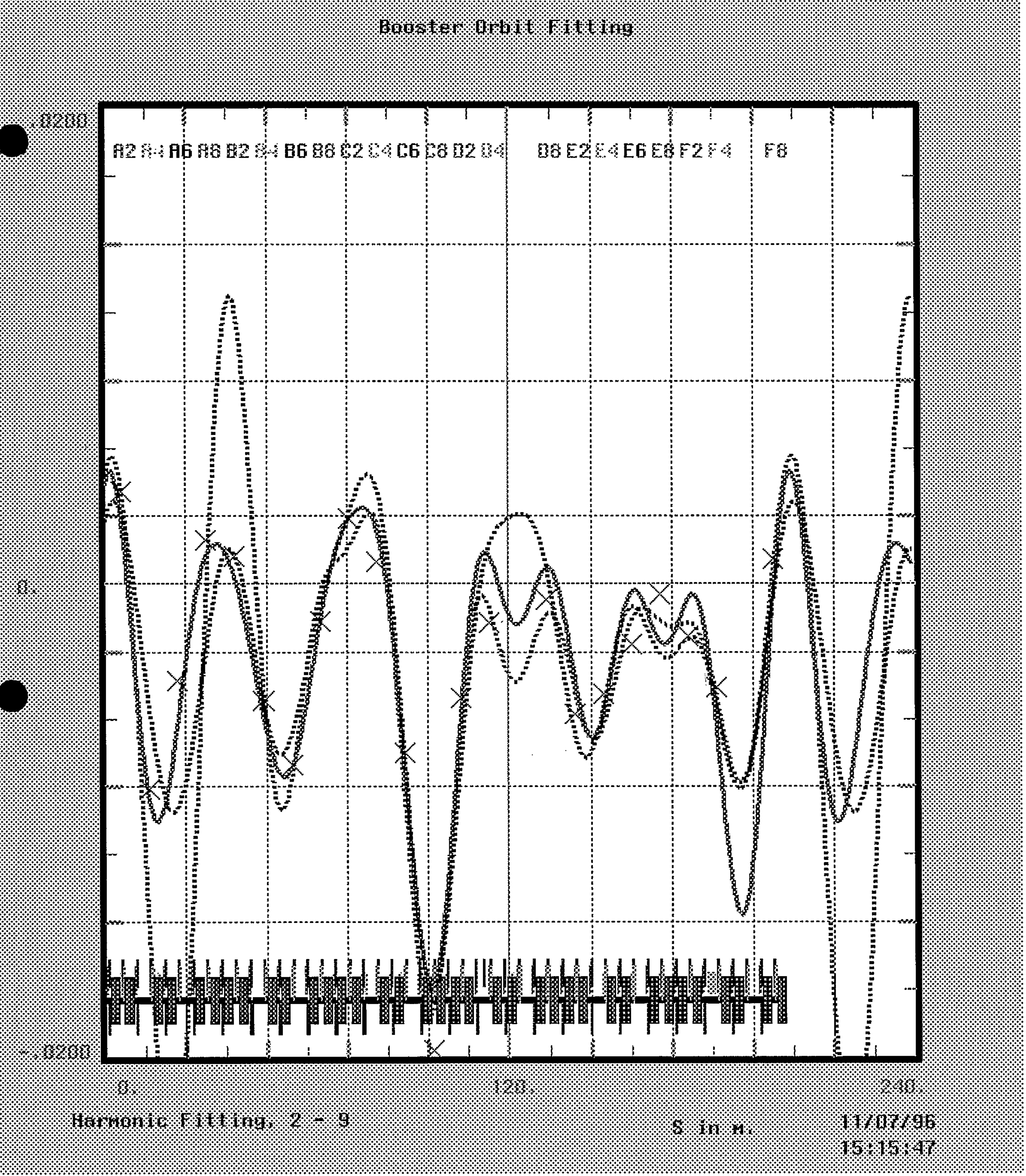

**Figure 6.**

A repeat of Figure 5, adding the fit with the general least squares method of Figure 4 to the same data with the same four data points removed. Harmonics J = 0, 2 - 9 are included. While the case is perhaps somewhat artificial, this kind of display rather forcefully points out nonsense fits. Data seems to be somewhat inconsistent here in the D and E regions.

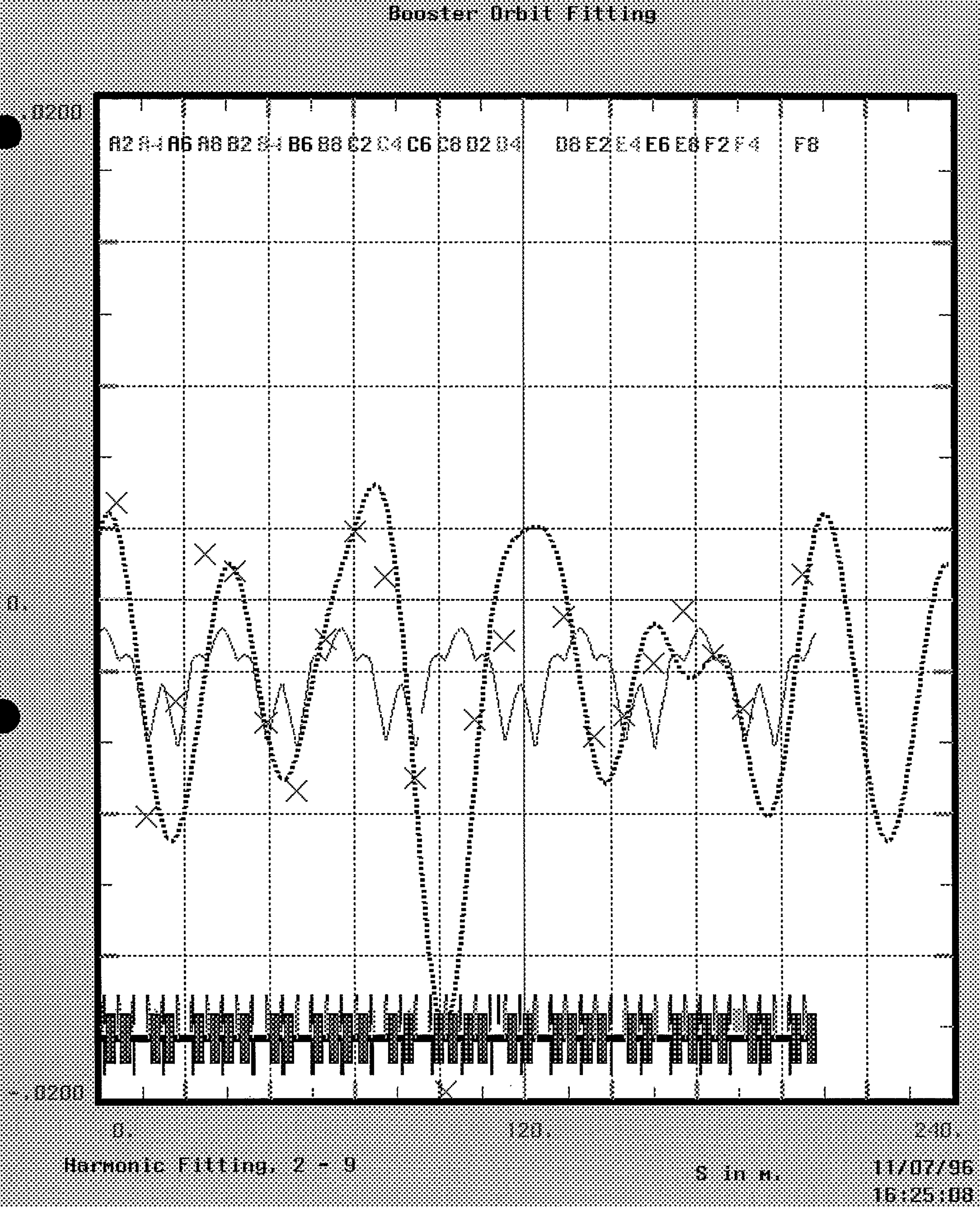

**Figure 7.**

A repeat of the constrained fit with four points removed of Figure 3, along with a Twiss calculation of the orbit for this off momentum. (˜.002) This shows the lack of detail resulting from the monitor placements.

```
  !          Harmonics Command Library for Booster Runs  (Partial)
  !          November 8, 1996
  !          File = "/usr/disc2/jn/Docum+/Vharm.ex"
  !      From:       " /usr/disc2/jn/Booster+/Hruns.boos"

!   Lattice Names of Bpms for Labelling Outputs.
    Xlab  Xmenu  = A2, A4, A6, A8, B2, B4, B6, B8, C2, C4, C6, C8,          &
                   D2, D4, D6, D8, E2, E4, E6, E8, F2, F4, F6, F8

!   Booster Orbit Data (Old, ~ April 1993)
  AX_30.1  Vector,                                                          &
  3.956113   -8.693634   -4.038608    1.899860   1.212470   -4.886150       &
 -7.630552   -1.532651    2.786204    0.946376  -7.146722  -19.704966       &
 -4.802379   -1.614516      44444.   -0.625031  -5.475378   -4.604772       &
 -2.491268   -0.376475   -2.166401   -4.313535     44444.    1.058067

 !*************************************************************************
!      Harmonic Analysis Commands for Machine Data, Vector Inputs.
  Store
  Boos.ccc     Vharm       Xvector = Ax_30.1,        SXvector = SXboos       &
                           Xlabels = Xlab,           Cycles   =  20,3,       &
                           Print,                                            &
                           Type    =  Con,                                   &
                           Htable  = "Hc_table",     Stacktab = "Sc_table",  &
                           Jmin    =  2,             Jmax     =    9,        &
                           Filter  = 99.,            Smax     = 201.78

!    Alternate Harmonic Analysis, "General" Rather Than "Constrained" Fit.
  Boos.ggg     Copy (Boos.ccc),                                              &
                           Type    =  Gen,           Htable   = "Hg_table"

!     Harmonic Copies for Demonstrating Effect of Missing Monitors.
    Boos.cct  Copy(Boos.ccc),  Htable = "Hc_hole",  Xweights= Wxx.
    Boos.ggt  Copy(Boos.ggg),  Htable = "Hg_hole",  Xweights= Wxx.
!     Weights, Used to Kill Off a Section of Monitors in Demos.
      Wxx.   Vector,  1., 4 ^ 0., 19 ^ 1.

 !*************************************************************************
 !  Commands for Drawing Harmonics for Machine Orbit Data.
 Hc.Pharx    Pharmon          Htable = "Hc_table", Curvedef = Cdef3c,      &
                              Convert= .001,       Smax     = 201.78,      &
                              Ypos   = Yyused * .80,  Noamp
 Hg.Pharx    Copy(Hc.Pharx)   Htable = "Hg_table", Curvedef = Cdef3g,      &
                              Ypos   = Yyused * .80,  Noamp
 Hct.Phar    Copy(Hc.Pharx)   Htable = "Hc_hole",  Curvedef = Cdef3ct,     &
                              Ypos   = Yyused * .80,  Noamp
 Hgt.Phar    Copy(Hc.Pharx)   Htable = "Hg_hole",  Curvedef = Cdef3gt,     &
                              Ypos   = Yyused * .80,  Noamp

 !*************************************************************************
 !  Curve, Pendefs for Distinguishing among Harmonics Curves, Plots.
 !  Indexed by Harmonic Number. Colors, Line Widths, Line Styles.
 !    ( Incomplete, Some Parts Are on Plot File: "Hplots.boos" )
 Cdef3c      Copy(Cdef3)      Pen      = Pdata3c,    Capts = Capts3
 Capts3c     Copy(Capts3)     Suppress
 Pdata3c     Pendata          Masks    = C3_maskj,   Colors     = C3_colc,  &
                              Styles   = C3_stylc,   Widths     = C3_widc
 C3_colc     Mvector,         N1 = 30, 0, 7, 2, 3, 4, 5, 6, 1, 0, 1
 C3_stylc    Mvector,         N1 = 30, 10 ^ 0
```

```
C3_widc      Mvector,         N1 = 30,  9 ^ 1, 3
!            Mask for All Curves.
C3_maskc     Mvector,         N1 = 30, 10 ^ 1
!            Mask for Final Curve Only.
C3_maskj     Mvector,         N1 = 30,  9 ^ 0, 1

Cdef3ct      Copy(Cdef3)      Pen        = Pdata3ct,   Capts = Capts3
Pdata3ct     Pendata          Masks      = C3_maskj,   Colors      = C3_colct,   &
                              Styles     = C3_styct,   Widths      = C3_widct
C3_colct     Mvector,         N1 = 30, 0, 7, 2, 3, 4, 5, 6, 1, 0, 0
C3_styct     Mvector,         N1 = 30,  9 ^ 0, 12
C3_widct     Mvector,         N1 = 30,  9 ^ 1, 3

Cdef3g       Copy(Cdef3)      Pen        = Pdata3g,    Capts = Capts3g
Capts3g      Copy(Capts3)     Y          = - Yyused * .5, Suppress
Pdata3g      Copy(Pdata3)     Masks      = C3_maskk,   Colors      = C3_colg,    &
                              Styles     = C3_stylg,   Widths      = C3_widg
C3_colg      Mvector,         N1 = 30, 0, 1, 2, 3, 4, 5, 6, 4, 0, 4
C3_stylg     Mvector,         N1 = 30, 10 ^ 0
C3_widg      Mvector,         N1 = 30,  9 ^ 1, 3
!            Mask for All Curves.
C3_maskg     Mvector,         N1 = 30, 10 ^ 1
!            Mask for Final Curve Only.
C3_maskk     Mvector,         N1 = 30,  9 ^ 0, 1

Cdef3gt      Copy(Cdef3)      Pen        = Pdata3gt,   Capts = Capts3g
Pdata3gt     Copy(Pdata3)     Masks      = C3_maskk,   Colors      = C3_colgt,   &
                              Styles     = C3_stygt,   Widths      = C3_widgt
C3_colgt     Mvector,         N1 = 30, 0, 1, 2, 3, 4, 5, 6, 4, 0, 4
C3_stygt     Mvector,         N1 = 30,  9 ^ 0, 12
C3_widgt     Mvector,         N1 = 30,  9 ^ 1, 3

!     Commands to Plot Machine Monitor (Bpm) Vector.
H1.Pvecx     Copy(T1.pvecx)     Vector  = Ax_30.1,   Ccolor   = 0,               &
                                Convert = .001,      Labels   = Xlab

!    Lattice Schematic Drawing Command. Vertical Sizes of Magnet Symbols.
T.Schema     Pschema    Curvedef    = Cdef2,                                     &
                        Y0          = Yyused * .06,                              &
                        Y1          = 0.,    0.,     0.,    -17.,   -17,         &
                                      5.,   -5.,     5.,     -5.,    10.,        &
                        Y2          = 0.,   25.,   -25.,     17.,    17.,        &
                                     25.,  -25.     20.,    -20.,    30.
!    Twiss Orbit Drawing Command.
T1.orbx     Porbit      Curvedef    = CdefB1,  Table    = "*Twiss.1"

!************************************************************************

!    Plot Definitions.  Scales, and Picture Element Name.
"Pdef.tx"    Plotdef       Plist  = "Txorbits", x1 =    0.,   x2 = 240.,         &
                           Window =  PirisP,     y1 = - .02,  y2 =   .02
!     (Partial) List of Picture Elements - Curvedefs, Borders, Scales,
!                     Axes, Labels, Titles, Drawing Commands.
 "Txorbits" Plist     = (Cdef2,        CdefB1,                                   &
                         Cdef3c,       Cdef3g,   Cdef3ct,  Cdef3gt,              &
                         Bp1,    Bp2,    Bp3,    Bp4,    Bp5x, Bp6x, Bp7,        &
                         Bp8,    Bp9,    Bp10,           Bp12, Bp13, Bp14,       &
                         Bp15,   Bp16,   Bp17,           Capts3,                 &
                         T.schema,       T1.orbx,                                &
                         Hc.pharx,       Hg.pharx,     Hct.phar,    Hgt.phar,    &
                        H1.pvecx,     DrawC5)
 Endstore
!***********************************************************************
!   Typical Subroutines, Can Be Executed from Plcalls Menu.

    Compute Harmonics from Booster Bpm Vectors
  S.BHarm  Subroutine
     Boos.ccc
     Boos.ggg
     Boos.cct
     Boos.ggt
  Endsubroutine
 !    Define Open Vector to Get Bpms from Twiss Runs.
        Vtwx1  Vector,  24 ^ 0.

 !     Subroutine Command Group to Prepare Twiss Orbit.
   S.tww1  Subroutine
 !    Run Twiss for Orbit.
    FTW.1     Ftwiss, save,  deltap = pdelta, Savefile = "*Twiss.1"
 !    Move Orbit Snapshot to Vector = Vtwx1 (List Is Names of Bpms)
    M_twx1    DtoV   Vname  = Vtwx1,  List = L_Xbpm,   Param = "/X",       &
                     Clear
 !     Harmonic Analyzers for Twiss Runs. Snapshots at Monitors.
    T1.Vharm  Vharm  Xvector    = Vtwx1,        SXvector  = SXboos         &
                     Print,                                                &
                     Type       = Gen,          Htable    = T1_Table,      &
                     Jmin       =  1,           Jmax      =   9,           &
                     Cycles     =  3,           Filter    = 99.            &
                     Smax       = 201.78,       Convert   = 1000.
   Endsubroutine
 !  List of Subroutines to Be Included in Plcalls Menus.
 .comms     Xmenu      S.tww1,   S.Bharm

**********************************************************************
 !  Display Manager Definition. Attach Command Lists, Plotdef, etc.
    T.plx   Plcalls    Menu          = T_Xmenu,     Plotdef  = "Pdef.tx",    &
                       Inspect       = Pl_Insp,                              &
                       Commands      = T.comms,                              &
                       Labels(1,1) = "All",       Labels(1,2) = "Cycle",     &
                       Labels(1,3) = "Pick",      Labels(1,4) = "Clear",     &
                       Labels(1,5) = "Redraw",    Labels(1,6) = "Commands",  &
                       Labels(1,7) = "Sliders",   Labels(1,8) = "Changes"

 !     Do Harmonic Fits.  (Can Be Done from Menu Also)
  S.Bharm
 !     Execute Displays from Menus on Screen, Twiss, Fits, etc.
  T.plx

**********************************************************************

 Return

 STOP
```