Development of alternative method for Booster TBT data Analysis

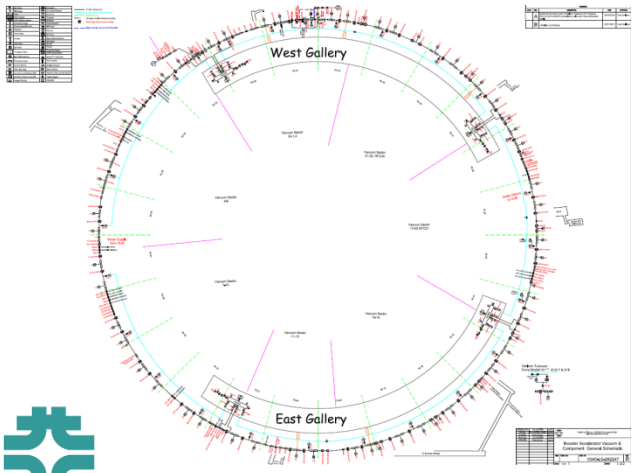
Alexey Boldyrev, MIPT

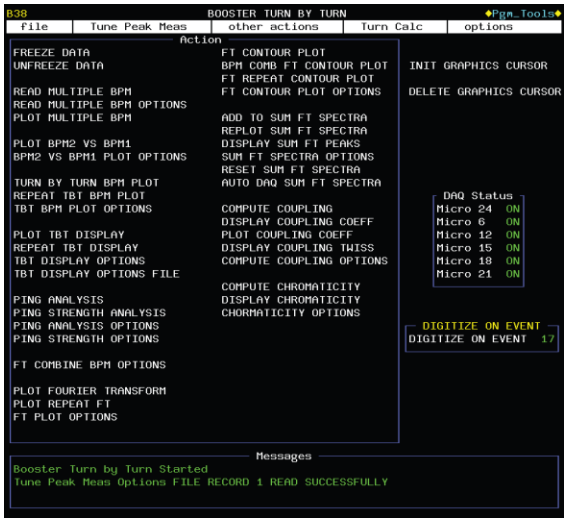
Supervisor: Eliana Gianfelice-Wendt, FNAL/APC

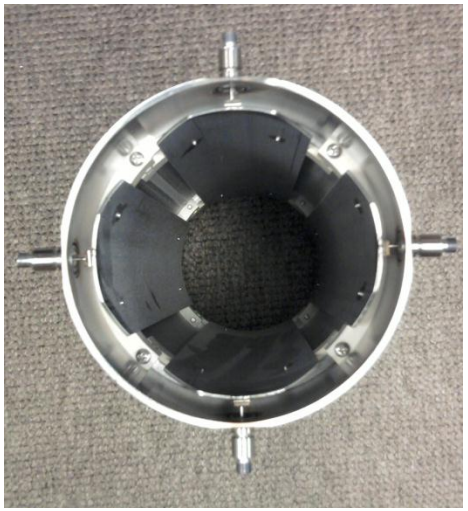
PARTI Summer Student Meeting
Fermi National Accelerator Laboratory
August 22, 2012

Fermilab Booster BPMs

- The Fermilab Booster - 474 m long machine accelerating protons from 0.4 to 8 GeV kinetic energy in 33 ms (about 22000 turns);
- B38 - Booster console application for analyzing the Turn-by-Turn (TBT) beam position measured by Beam Position Monitors (BPMs);
- 48 Booster BPMs measure both horizontal and vertical beam position.

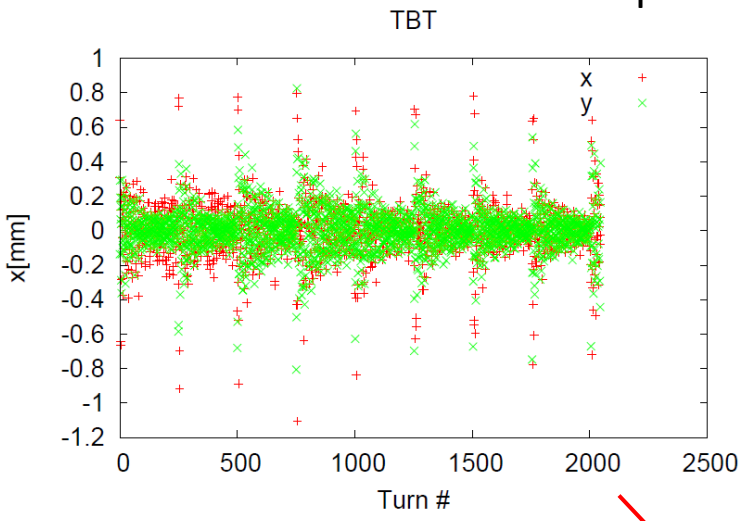






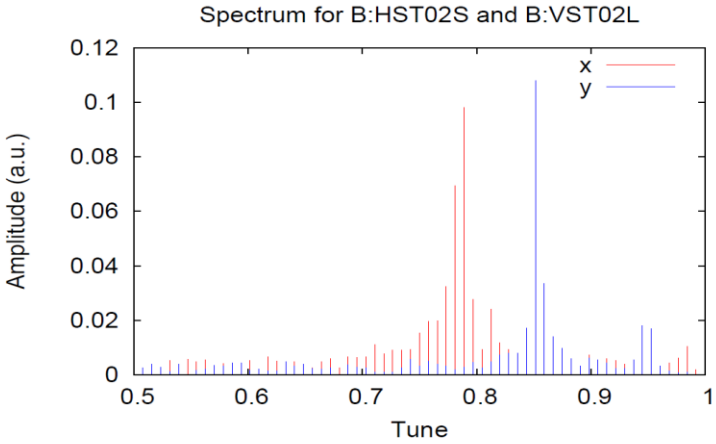
Continuous Fourier Transformation for TBT data

To decrease noise data from all BPMs were summed with according theoretical phase shifts for each BPM.

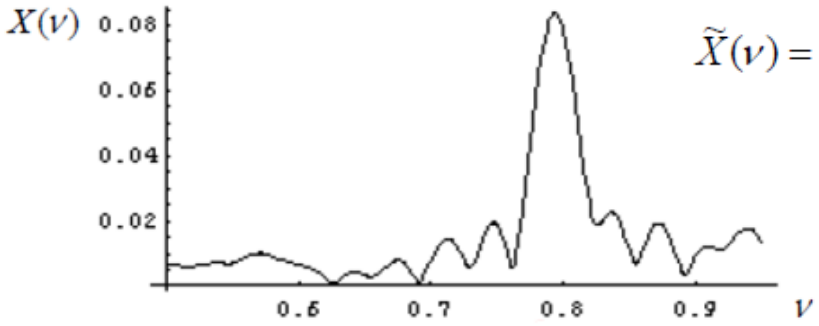


*fast decrement – small number of turns available

F.F.T.



"C.F.T."



$$\tilde{X}(\nu) = \frac{1}{N}\sum_{n=1}^{N} e^{-2\pi i\nu(n-1)} \sum_{k} w_k x_n^{(k)} e^{-i\varphi_x^{(k)} - is_k(\nu - Q_{x0})}$$

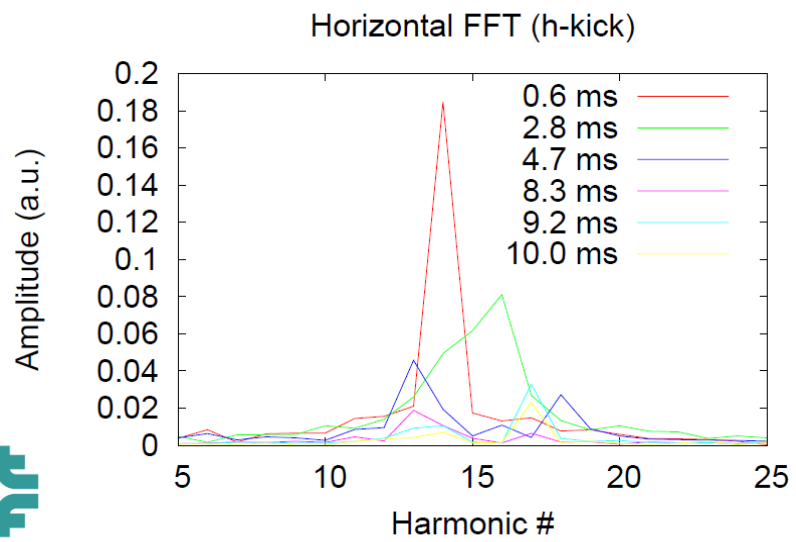
$$\tilde{x}_n = \sum_k w_k x_n^{(k)} \exp[-i\varphi_x^{(k)}]$$

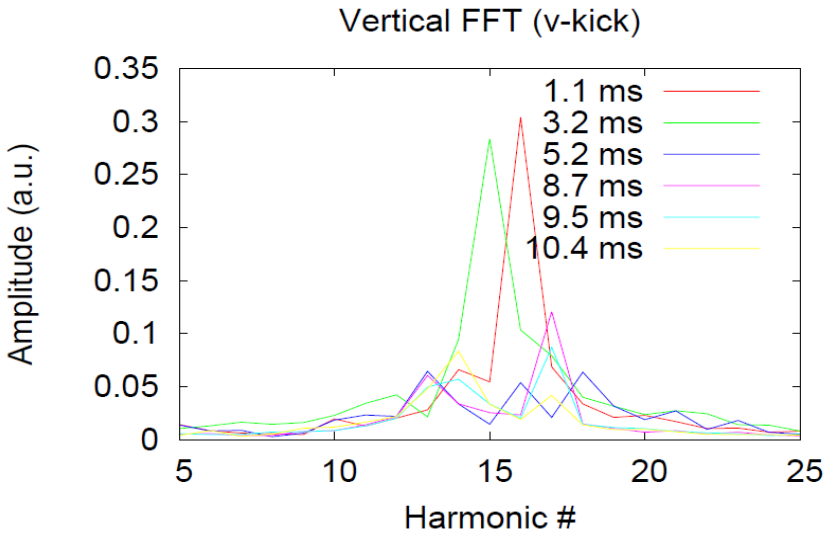
$$\varphi_x^{(k)} = \varphi_{x0}^{(k)} + s_k(\nu - Q_{x0})$$

Problems

- In the presence of coupling both transverse mode peaks are present in the spectra of oscillations;
- Existing algorithm may fail to identify tunes correctly in case of strong coupling, BPM noise or when the mode with lowest damping rate dominates the spectra of oscillations in both planes.

Example of a problematic point: which are the tunes at 9.5 ms?





New algorithm for tune identification

An alternative algorithm with real fit parameters for TBT data analysis which should improve tune identification has been proposed:

$$F^x{}_n \sim a_x e^{ig(\nu_x,\varphi_x,d_x,n)} + b_y e^{ig(\nu_y,\psi_y,d_y,n)}$$
$$F^y{}_n \sim a_y e^{ig(\nu_y,\varphi_y,d_y,n)} + b_x e^{ig(\nu_x,\psi_x,d_x,n)}$$

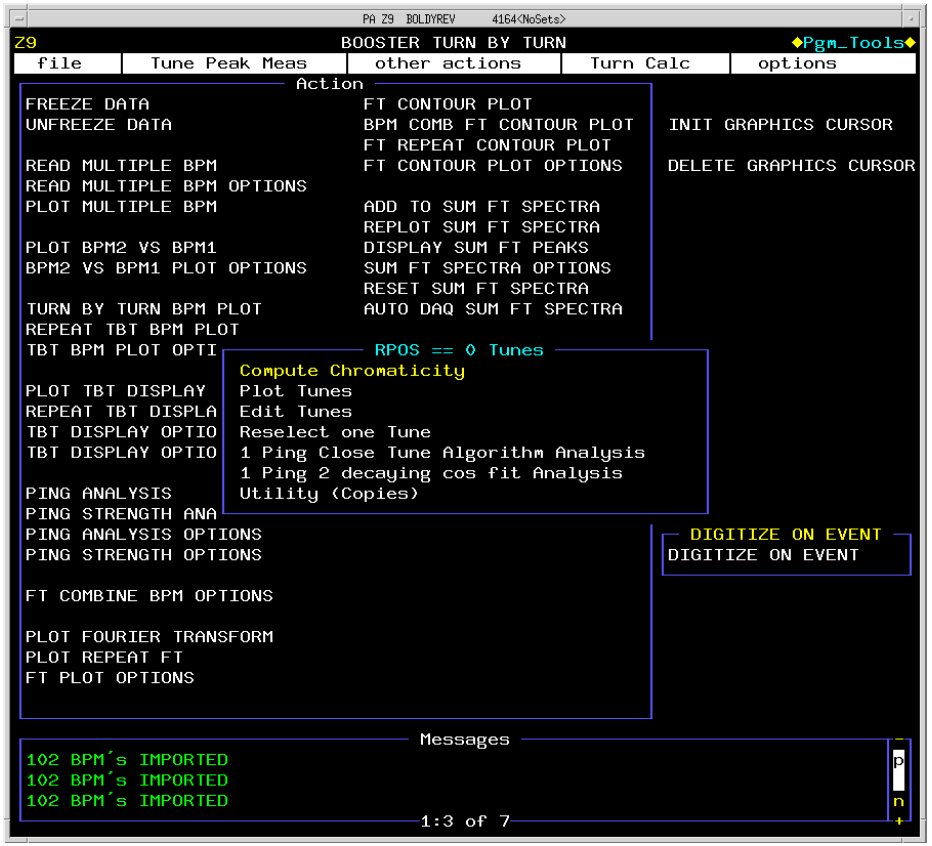
where $g = (2\pi n\nu + \varphi) + i * nd$ or $g = (2\pi n\nu + \psi) + i * nd$

The values of the variable parameters is found by minimizing the function:

$$F_n = [\Re(F^x{}_n - \tilde{x}_n)]^2 + [\Im(F^x n - \tilde{x}_n)]^2 + \frac{{a_{x0}}^2}{{a_{y0}}^2}\{\Re[F^y{}_n - \tilde{y}_n]^2 + \Im[F^y{}_n - \tilde{y}_n]^2\}$$

Main Goals

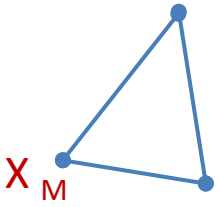
- Write a C++ routine resorting to a suitable minimization package for performing the fit;
- Embed it into B38.



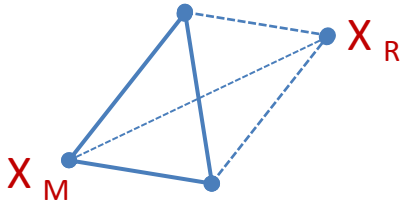
Nelder-Mead algorithm

- Simplex-method
- Does not use gradients
- Applied to noisy and nonsmooth functions

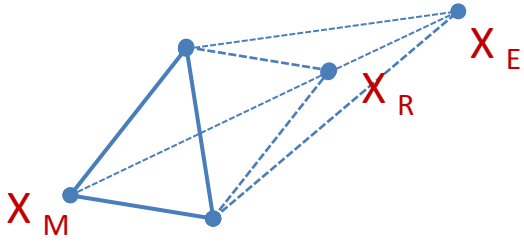
Simplex deformations:



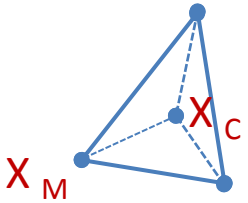
Initial



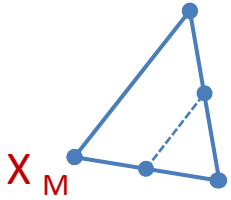
Reflection



Expansion



Contraction



Multi-Contraction

Variables constraining

Till beam is stable:

$\Phi_{x,y}, \psi_{x,y} \in [-\pi, \pi]$;

$\nu_{x,y} \in [0.5, 1]$;

$a_{x,y}, b_{x,y} > 0$;

$d_{x,y} > 0$

```
 Estimate of minimizing value X*:

              ax        10.3967
              bx        9.06848
             d_x      0.0160238
              qx       0.839712
           phi_x        1.57997
           psi_x        3.14116
              ay        5.68827
              by        7.44748
             d_y        1.23872
              qy        1.85984
           phi_y      -0.119833
           psi_y       -12.7483

 F(X*) = 5505.65

 Number of iterations = 14616
 Number of restarts =   13

Tune qy is not in range!
New tune qy = 0.859838

Phase psiy is not in a range!
New phase psiy = -0.181885
```

Return INFINITY value when parameters are out of range – leads to excess of iterations;

After a minimum is found, the variables are moved to the wished range of values. This is done for clarity purposes, being yet suitable for the tunes.

Stability of algorithm

- Starting point is obtained from CFT;
- Method non-sensitive for deviation from starting point:

  Amplitudes 30%,

  Phases 40%,

  Decrements 100%,

  Tunes 30%;
- For deviation from starting point more than is noted above tunes values are changed to 0.2% and 1.7% for ν_x and ν_y respectively.

Conclusions

- Suitable algorithm for function minimization was found;
- Algorithm was realized and tested in off-line program;
- Tests showed that algorithm is stable for initial point deviation;
- Got familiar with MECCA (Fermilab procedure for compiling and administrating machine applications) and ACNET (Fermilab Accelerator Control System). Write a subroutine for B38 based on this algorithm (in progress).