DESY SUMMER STUDENT PROGRAM 2008

FREQUANCY MAP FOR PETRA III STORAGE RING

Manukyan Koryun
Supervisor: Alexander Kling

INTRODUCTION

Frequency map analysis (FMA) was introduced for the demonstration and understanding of the chaotic behaviour of the solar system that can be considered as a dynamical system with 3N degrees of freedom (DOF) where N is the number of planets. The method applies more generally to any Hamiltonian system or symplectic map (eventually with some small dissipation). It is particularly interesting for systems of DOF larger than 2, when a simple surface of section fails to provide a global view of the dynamics of the system. The application to particle accelerator dynamics was very natural, as the motion of a single particle in a storage ring is usually described in a surface of section of the beam by a symplectic map of dimension 4, or eventually of dimension 6 when the synchrotron oscillation is also taken into account. FMA has been applied to many machines, providing in each case a picture of the dynamics of the beam. Here we consider a FMA for the PETRA III storage ring obtained by the particle tracking program “Elegant”. We use the PETRA III bare lattice without insertion devises obtained from the MAD-X input file. A finally we compare the result obtained by elegant with results obtained with “SIXTRACK”.

FREQUENCY MAP

Frequency map analysis is not a perturbative theory, it is useful to describe its properties of a Hamiltonian close to integrable, where a rigorous setting can be derived. Let us thus consider a n–DOF Hamiltonian system in the form $H(I, \theta) = H_0(I) + \varepsilon H_1(I, \theta)$, where H is real analytic for canonical variables $(I, \theta) \in B^n \times T^n$, B^n is a domain of R^n and T^n is the n-dimensional torus. For $\varepsilon = 0$, the Hamiltonian reduces to $H_0(I)$ and is integrable. The equations of motion are then for all j = 1. . . n

$$\dot{I}_j = 0, \qquad \dot{\theta}_j = \frac{\partial H_0(I)}{\partial I_j} = \nu(I) \qquad (1)$$

The motion in phase space takes place on tori, products of circles with radii I_j , which are described at constant velocity $\nu_j(I)$. If the system is nondegenerate, that is if

$$\det\left(\frac{\partial \nu(I)}{\partial I}\right) = \det\left(\frac{\partial^2 H_0(I)}{\partial I^2}\right) \neq 0 \qquad (2)$$

the frequency map

$$F : B^n \rightarrow R^n$$
$$(I) \rightarrow (\nu) \qquad (3)$$

is a diffeomorphism (one to one smooth map) on its image Ω, and the tori are as well described by the action variables $(I) \in B^n$ or by the frequency vector $(\nu) \in \Omega$. For a nondegenerate system, the KAM theorem still asserts that for sufficiently small values of ε, there exists a Cantor set Ω_ε of values of (ν), satisfying a Diophantine condition of the form

$$|< k,\nu >| = |k_1\nu_1 + . . . + k_n\nu_n| > \kappa_\varepsilon/|k|^m \qquad (4)$$

for which the perturbed system still possesses smooth invariant tori with linear flow (the KAM tori). Moreover, these tori that survive on a totally discontinuous set of initial conditions are still properly ordered in some sense as, according to Poschel [22], there exists a diffeomorphism

$$\Psi : T^n \times \Omega \rightarrow T^n \times B^n; \qquad (\varphi, \nu) \rightarrow (\theta, I) \qquad (5)$$

which is analytic with respect to φ, C^∞ in ν, and on $T^n \times \Omega_\varepsilon$ transforms the Hamiltonian equations into the trivial system

$$\dot{\nu}_j = 0 \qquad \dot{\varphi}_j = \nu_j \tag{6}$$

If we fix $\theta \in T^n$ to some value $\theta = \theta_0$, we obtain a frequency map on B^n defined as

$$F_{\theta_0} : B^n \to \Omega \qquad I \to (\nu) = p_2\left(\Psi^{-1}(\theta_0, I)\right) \tag{7}$$

where p_2 is the projection on Ω ($p_2(\varphi, \nu) = \nu$). For sufficiently small ε, the torsion condition (2) ensures that the frequency map F_{θ_0} is still a smooth diffeomorphism.

The frequency analysis method and algorithms rely heavily on the observation that when a quasiperiodic function f(t) in the complex domain C is given numerically, it is possible to recover a quasiperiodic approximation of f(t) in a very precise way over a finite time span [−T, T], several orders of magnitude more precisely than by simple Fourier analysis. Indeed, let

$$f(t) = e^{i\nu_1 t} + \sum_{k \in Z^n} a_k e^{i\langle k,\nu\rangle t} \qquad a_k \in C \tag{8}$$

be a KAM quasiperiodic solution of an Hamiltonian system in $B^n \times T^n$, where the frequency vector (ν) satisfies a Diophantine condition (3). The frequency analysis algorithm NAFF will provide an approximation $f'(t) = \sum_{k=1}^{N} a'_k e^{i\omega_k t}$ of f(t) from its numerical knowledge over a finite time span [−T, T] . The frequencies ω'_k and complex amplitudes a'_k are computed through an iterative scheme. In order to determine the first frequency ω'_1, one searches for the maximum amplitude of $\phi(\sigma) = \left\langle f(t), e^{i\sigma t} \right\rangle$ where the scalar product $\langle f(t), g(t) \rangle$ is defined by

$$\langle f(t), g(t) \rangle = \frac{1}{2T} \int_{-T}^{T} f(t)\bar{g}(t)\chi(t)dt \tag{9}$$

and where χ(t) is a weight function. Once the first periodic term $e^{i\omega'_1 t}$ is found, its complex amplitude a'_1 is obtained by orthogonal projection, and the process is restarted on the remaining part of the function

$$f_1(t) = f(t) - a'_1 e^{i\omega'_1 t} \tag{10}$$

For a KAM quasiperiodic solution (8), the computed frequency ν_1^T converges very rapidly towards the true frequency ν_1 as

$$\nu_1 - \nu_1^T = \nu\left(\frac{1}{T^{2p+2}}\right) \tag{11}$$

where p is the order of the cosine window $\chi_p(t) = 2^p(p!)^2(1+\cos \pi t)^p/(2p)!$ used in (9). To construct numerically a frequency map, we will fix all initial angles $\theta_i = \theta_{i0}$, and for each initial action values $(I) = (I_1, \ldots, I_n)$, integrate numerically the trajectories over a finite time interval of length T . The fundamental frequencies (ν) are computed by the previous (NAFF) algorithm, for all initial actions (I), and we thus construct a correspondence :

$$F_{\theta_0}^T : B^n \to \Omega \qquad I \to (\nu) \tag{12}$$

that converges towards F_{θ_0} as T → +∞. This map will thus be regular on the set of regular trajectories, and whenever its appears to be non regular, it will reveal the existence of chaotic orbits.In practice, to study the dynamics of a beam, in a given surface of section corresponding to a starting location on the lattice, one can fix the two transverse momenta, and integrate the trajectories with a tracking code for a network of initial conditions spanning both horizontal and vertical directions [1].

We look at a given longitudinal position (typically s = 0) at the return map. The coordinates used are the canonical transverse positions (x; y) and momenta (x', y'). Given a set of initial conditions (x_0; y_0; $x'_0 = y'_0 = 0$) the particle trajectory is numerically computed over few thousands turns. For a surviving particle, we plot it in the configuration space defining the dynamic aperture and we compute its transverse tunes with the FMA over the first half turns and then again over the last half turns. The logarithmic tune difference gives a diffusion index ($\log(d\nu)$) coded by a colour.

PETRA III

Petra III is a high brilliance 3rd generation low emittance synchrotron radiation source. The storage ring is divided into 8 octants, each consisting of an arc and a long or a short straight section on either side. Seven arcs are formed from FODO cells with mirror symmetry in the middle of each octants arc and a missing magnet dispersion suppression scheme. One octant, NE to E, is currently reconstructed into a DBA lattice (from 1100 to 1400 in fig.2). Figure 2 shows the optic functions for the bare machine (without insertion devices) calculated bay elegant. The design parameters for PETRA III are given in Table 1 [2].

Parameter	Value	Unit
Number of insertion devices	13	
Energy	6	Gev
Current	100	mA
Emittance $\varepsilon_x/\varepsilon_y$	1/0.01	nmrad
Circumference	2304	m

Table 1.Design parameters of PETRA III (including dumping wigglers)

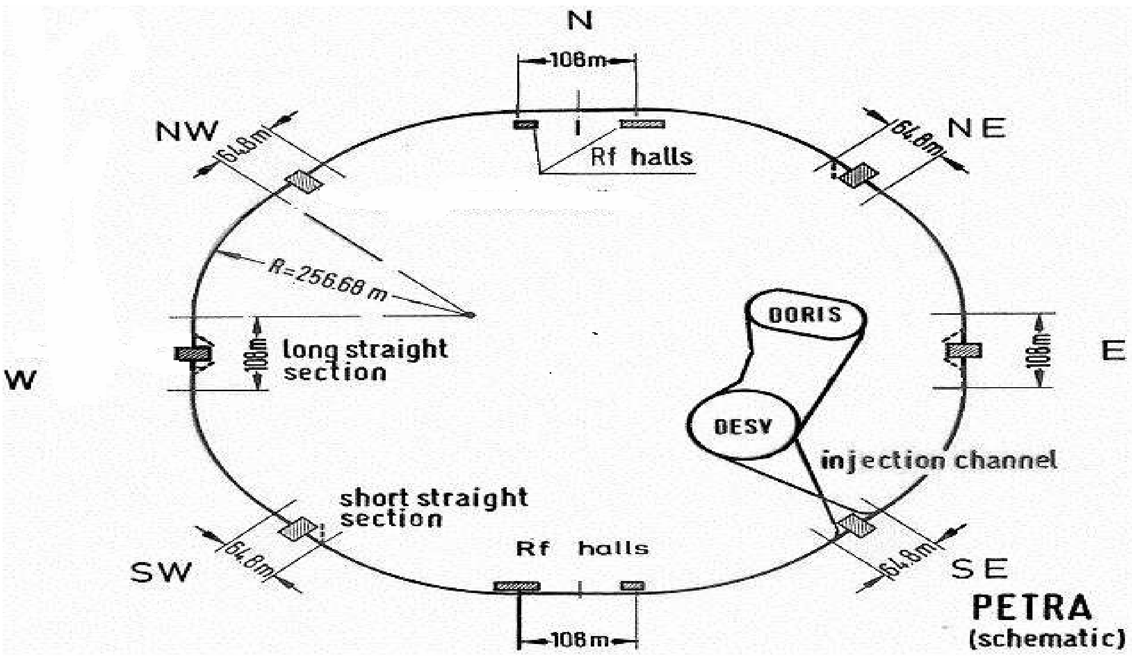


Fig.1: PETRA III schematic overview.

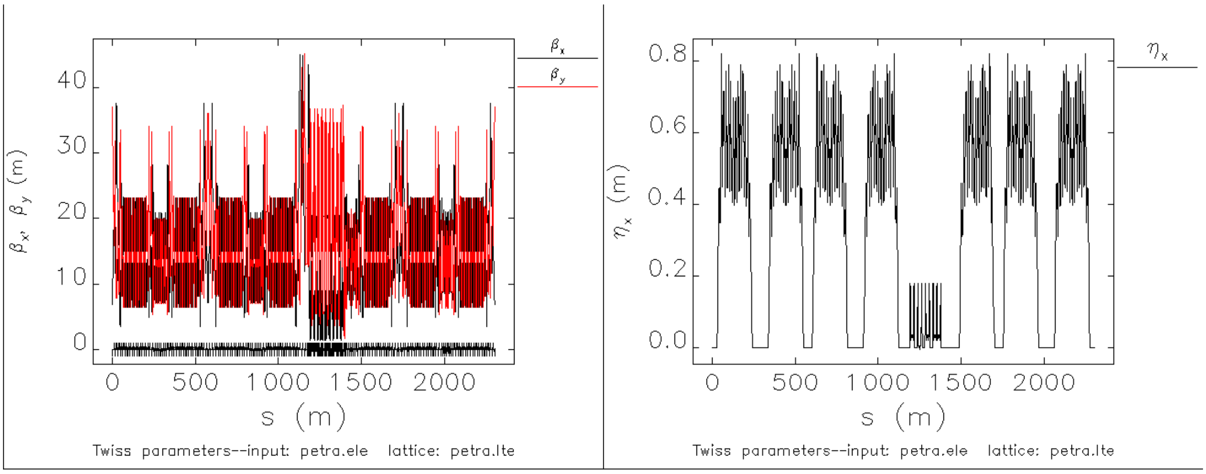


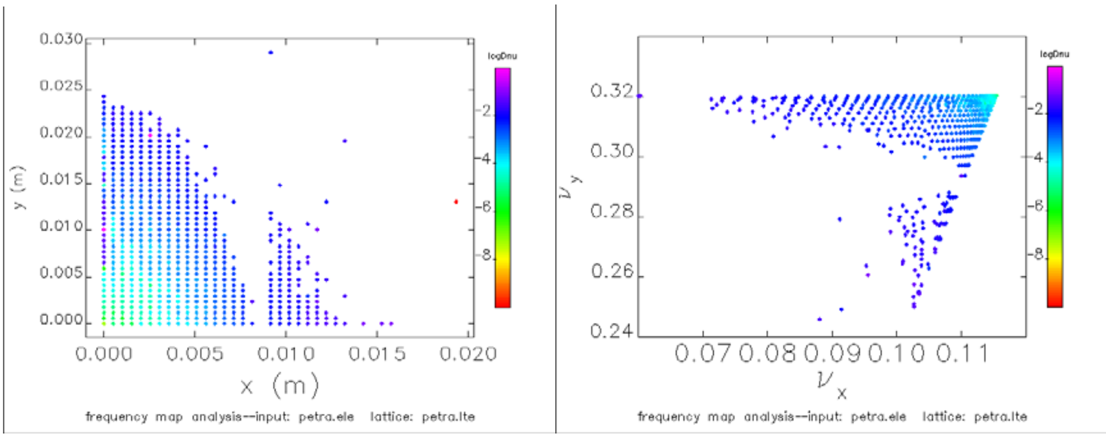
Fig.2: Beta functions and dispersion for PETRA III storage ring computed with elegant.

ELEGANT

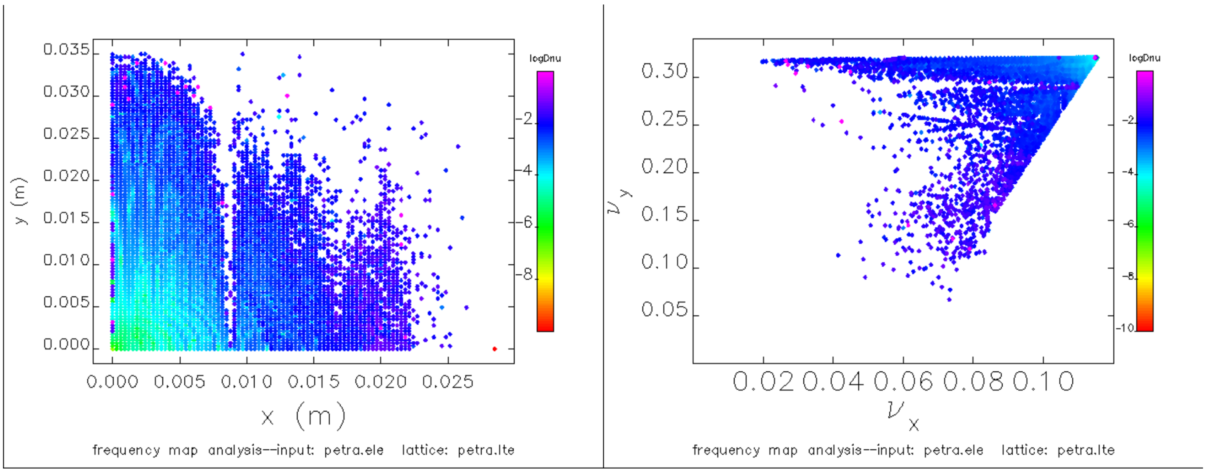
Elegant (ELEctron Generation ANd Tracking) is a 6-D accelerator simulation code that does tracking with matrices or using sympletic elements, optimization, synchrotron radiation, scattering, etc. elegant is SDDS-compliant (Self Describing Data Sets). For all its complexity, elegant is not a stand-alone program. For example, most of the output is not human-readable, and elegant itself has no graphics capabilities. These tasks are handled by a suite of post-processing programs that serve both elegant and other physics programs. These programs, collectively known as the SDDS Toolkit provide sophisticated data analysis and display capabilities. They also serve to prepare input for elegant, supporting multi-stage simulation. Elegant, written entirely in the C programming language, uses a variant of the MAD input format to describe accelerators, which may be either transport lines, circular machines, or a combination thereof. Program execution is driven by commands in a namelist format.
Elegant tracks in the 6-dimensional phase space (x, x', y, y', s, δ), where x (y) is the horizontal (vertical) transverse coordinate, primed quantities are slopes, s is the total, equivalent distance travelled, and δ is the fractional momentum deviation. The main input file for an elegant run consists of a series of namelists, which function as commands. Most of the namelists direct elegant to set up to run in a certain way. A few are "action" commands that begin the actual simulation. Each namelist has a number of variables associated with it, which are used to control details of the run. These variables come in three data types: (1) long, for the C long integer type. (2) double, for the C double-precision floating point type. (3) STRING, for a character string enclosed in double quotation marks. All variables have default values. STRING variables often have a default value listed as NULL, which means no data; this is quite different from the value "", which is a zero-length character string. long variables are often used as logical flags, with a zero value indicating false and a non-zero value indicating true[3].
Our simulation consists of next steps.

1. run_setup-Define global simulation parameters and output files.
2. run_control-Set up simulation steps and passes.
3. bunched_beam- Set up beam generation.
4. frequency_map-Compute and output frequency map.

Fig.3 Dynamic aperture and frequency map for PETRA III



a) Grid size 60*60 number of turn 8000 (dumping time)



b) Grid size 100*100 number of turns 1000(1/8 dumping time)

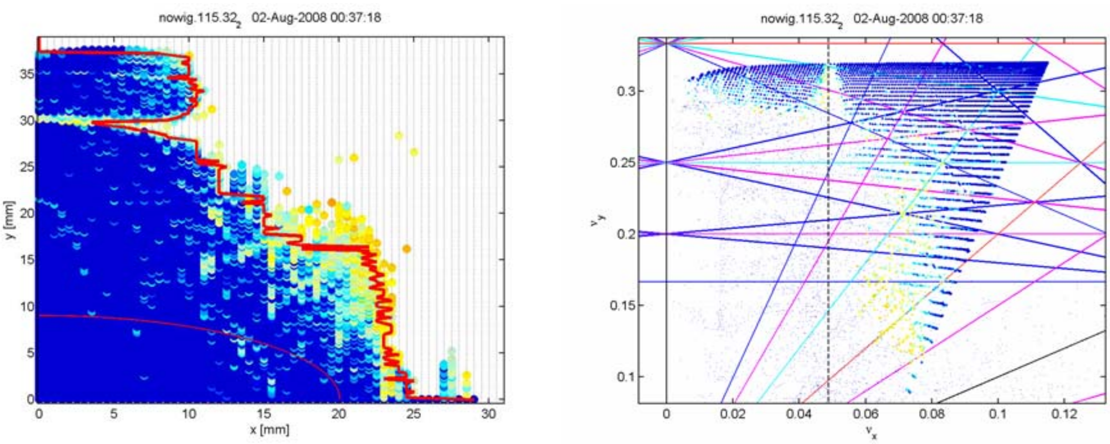


Fig. 4 Dynamic aperture and frequency map for PETRA III obtained by "SIXTRACK"

Reference

[1]. J.Laskar, frequency map analysis and particle accelerators, proceedings of the 2003 Particle Accelerator Conference

[2]. PETRA III technical design report

[3]. User's Manual for elegant