ISTITUTO NAZIONALE DI FISICA NUCLEARE

Laboratori Nazionali di Frascati

SPARC/EBD-07/004
24 Luglio 2007

A Genetic Code able to compute the emittance value of a real beam by a Multiple Ellipse Slice Analysis of the transversal phase-space image

Alberto Bacci (INFN-MI)

Abstract

In this note an innovative method to analyze the trace space of a real beam is presented. This method directly works on the intensity distribution shown in the trace space, represented by a bi-dimensional function. A code has been developed using a Genetic Algorithm (GA) [1-2], that is able to manage different elliptical distributions in order to find out an analytical representation of the real beam and to calculate its rms emittance. The method strength is in representing beams composed by sub-beams of different density, such beams showing cross like form distribution or many ellipses opening in a bowtie like form, in trace spaces. It is also noteworthy the fact that a theoretical model of the beam is forced to represent the real beam, giving analytical tools to the user, like the twist parameters, which could be useful to understand the beam dynamics.
This code has been developed to analyze the results coming from the beam dynamic of the SPARC [3] photoinjector and it has produced good results in full agreement with other methods.
Trace space versus the longitudinal position of the beam, that has been analyzed, has been obtained by the movable emittance meter [4] and by using a dedicated algorithm [5].

PACS:11.30.Er,13.20.Eb;13.20Jf;29.40.Gx;29.40.Vj

Published by ***SIS-Pubblicazioni***
Laboratori Nazionali di Frascati

1 Introduction

A real beam that is well represented by an ellipse Fig. 1 in the projected phase space and shows boundaries well defined, has the emittance proportional to the ellipse area. Real beams usually do not have well defined boundaries and a method, for calculating the emittance, is to choose a specific density contour, in the phase space, that represents from the 50% (worst cases) up to the 98-99% (best cases) of the bunch charge (or integrated intensity). Inside this density contour and under certain conditions [6], such emittance satisfies the Liouville's theorem and is thus conserved.
Electrons Bunches produced by photoinjectors, in some cases, could be represent by the sum of many beams with different densities. Each beam shows its own dynamic and draw concentrical ellipses in the phase space, with different slopes and with covering areas of different intensities.
Such a beam has an associated emittance that is directly proportional to the total union area of the ellipses. Therefore the emittance could be evaluated finding the ellipses boundaries union and the rms value rightly from the distributions enclosed in the boundaries union. In the calculations of the rms emittance, every phase space point is weighted by its intensity value.
For the electron beam of SPARC, following theoretical considerations and in the beam line zone where the measures have been done, we look for the union of two elliptical distributions which, should represent approximately all the bunch charge, and for other distributions with intensity very low that could draw other elliptical areas in the phase space.
The code here presented has been optimized to find solutions from the union of eight ellipses (however it could work with any numbers of ellipses). For example in a bunch composed by eight sub-bunches, the code is able to provide evidence for all the eight elliptical distributions of the phase space. In the case where the sub bunches are less then eight, as in the SPARC experiment, the useless ellipses falls inside these useful ones. The useless solution are degenerate solutions that do not affect the calculated emittance. Each ellipse gives a contribution to the emittance value only for those portions of area that are not included in other ellipses.

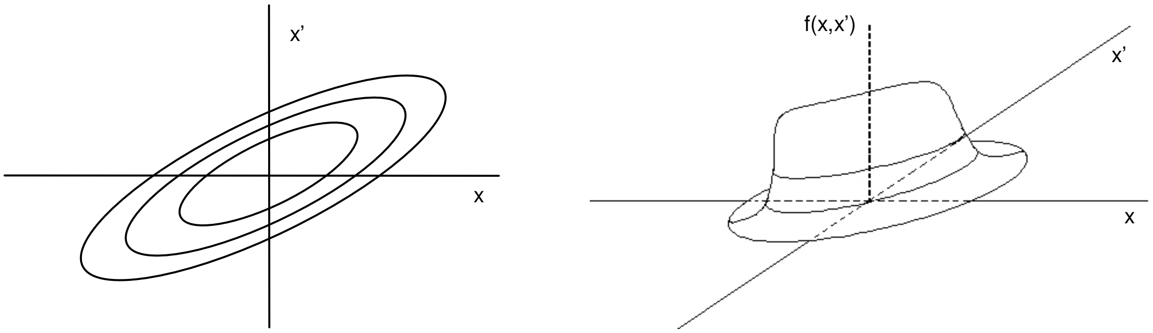

Figure 1: Isodensity contours of the beam. The emittance associate with any elliptical contour is the area of the contour divided by π

2.1 The code GMESA

The GMSA (Genetic Multi Slice Analyzer) code have been developed by using a Genetic algorithm that is the code's core. Genetic algorithms are particularly suitable to solve problems that have nonlinear character and where it is not possible to treat each parameter as a variable which can be fixed independently from all other ones. The correct representation of a real beam, in the projected phase space, by using the union of N analytical ellipses is no easy work. It has been evaluated that a good representations, of the real beam in the trace space, could be performed with eight analytical ellipses; this involves the computation of 26 parameters; 24 for the ellipses, keeping in mind that any ellipse is characterized by three Twist parameters α, β, γ and 2 for the centre of the beam (centroid). As explained in the introduction, the code has find out the boundaries, of the union of the ellipses, as well as possible, so the user that runs the code has to decide, to use as input parameter, the portion of the bunch charge that has to be kept into account. This portion defines a specific density contour and usually the higher is such percentage the better is the beam quality, because the halo is low and the measures could be performed with high contrast. SPARC data have been analyzed with a percentage closer to 95% that is a very high value. When the percentage of the bunch charge is fixed the code has to find the best set of the 26 parameters that better describes the real beam and that represents the best solutions. These 26 parameters are linked together with strongly non linear behaviour.

One set of 26 parameters is considered a chromosome and every single parameter of the chromosome, is considered a gene. The code starts generating a random population of 24 chromosomes and each random chromosome represents a possible solution of the problem of describing the real beam like the union of analytical ellipses.

This starting process, that is the first generations of chromosomes, is very important because it fixes the genes ensemble (or parameters ensemble) from which all the successive chromosome generations are created. Then new genes could be introduced only by mutation, as explained in the next paragraphs.

In order to have a good first generation the process has been developed in a very accurate way.

Each chromosome is composed by eight ellipses, that are defined by the Twist parameters α, β, γ and by the common centre of the ellipses that is the centroid of the bunch. Three Twist parameters to set one ellipse and two coordinates to set the centre of the ellipses mean that, every chromosome has 26 genes. During the generation the ellipses have to be sampled as uniform as possible, inside the phase-space matrix. This sampling is very important and if it is not well done the performance of the code is deteriorated. To perform a very good sampling, the "shuffled uniform" random number generator has been used, so that the property of random numbers turns out to be superior to that in default subroutines generally used by common compilers.

For computational reasons the emittance equation (1) has been written in the following way:

$$gx^2 + bx'^2 + 2axx' = 1 \qquad (1)$$

where $gb - a^2 = \frac{1}{\varepsilon^2}$, $g = \frac{\gamma}{\varepsilon}$ and $a = \frac{\alpha}{\varepsilon}$. ε is the emittance associated with any two-dimensional elliptical trace phase area ($\varepsilon = \frac{area}{\pi}$) Fig. 2.

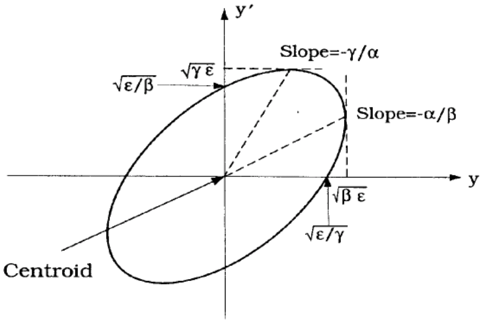

Figure 2: RMS ellipse

Following the equation 1 when a phase-space point (x, x') falls into the ellipse we obtain $gx^2 + bx'^2 + 2axx' < 1$, on the other hand ,when a point falls out to the ellipse boundaries $gx^2 + bx'^2 + 2axx' > 1$.

Since the size of the phase-space matrix is known, it is possible to make a parameterization of a, b and g such that all the random ellipses are generated inside the matrix. It is very important to choose a good parameterization, because different parameterization give different ellipses sampling, more or less uniform, that could affect the code's performance.

Another point that strongly affects the generation of the first population of chromosomes, and also all the code's runs, is the size of the phase-space matrix. These matrices often are oversized with respect to the information kept inside, thus causes that the ellipses ensemble to be much too wide, therefore the code tries many no good solutions wasting cpu time. This problem has been solved in the following way: the code starts finding the zones of the data matrix that do not keep information (corresponding to the frame of the matrix) and discharges them. This process is made by a binning along the two dimensions and by the analysis of the average intensity falling in every bin versus the bin position, as shown in the Fig. 3. The behaviour of the average intensity along the two dimensions gives the necessary information to resize the matrix just around the image of the electron bunch. Such process could reduce the matrix size by a worth percentage making the code speeder and the analysis more accurate.

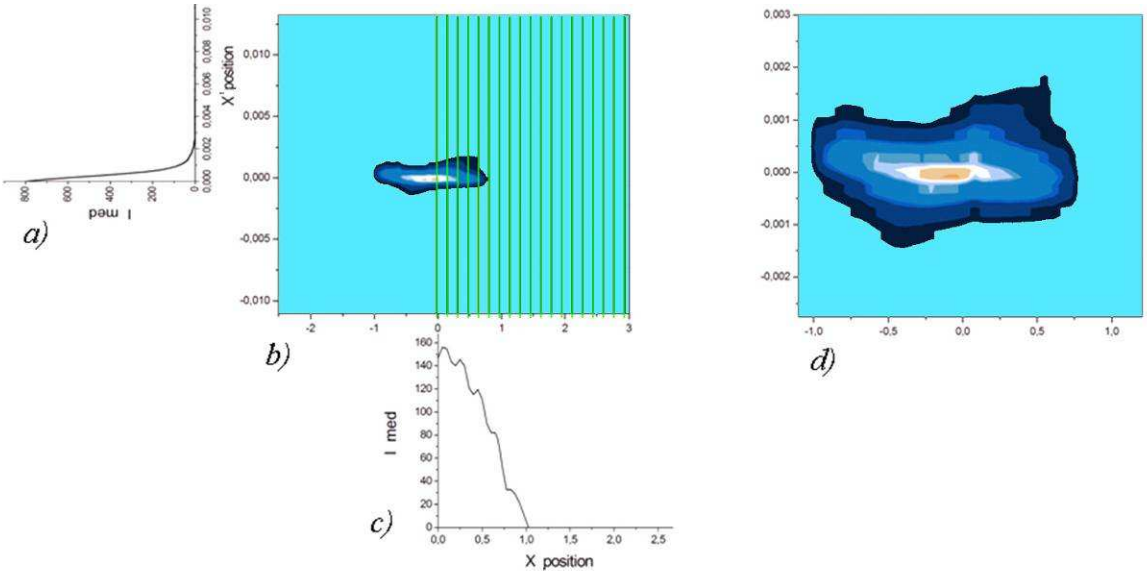

Figure 3: Downsizing process of the phase-space matrix. *a*) Behaviour of the average intensity along the *x'* dimension, resulting from the binning. *b*) Example of the binning process along x dimension. *c*) Behaviour of the average intensity along the *x'* dimension. *d*) The new phase-space matrix that will be used by the code. The frame, which don't keep information, has been discharged.

After the first population of chromosomes has been created, the genetic optimization process begins. That is 24 chromosomes generate a new population of 24 son chromosomes by crossover of the single individuals in an iteration way from generation to generation Fig 4. Two parent chromosomes generate two son chromosomes Fig. 5.

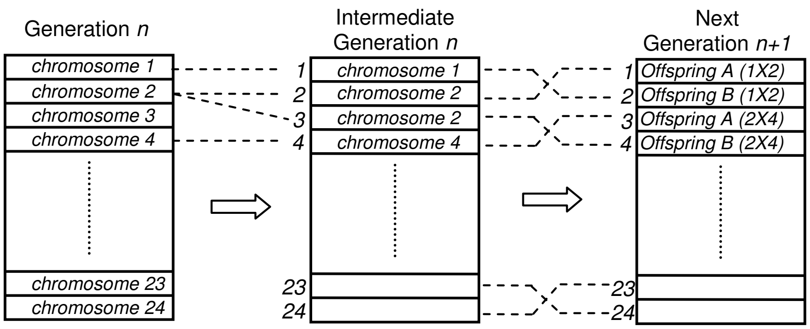

Figure 4: A schematic representation of the iteration genetic evolution. Each generation is broken down into a selection phase and then into a recombination phase. The process is iterated until the production of a good solution (or good chromosome)

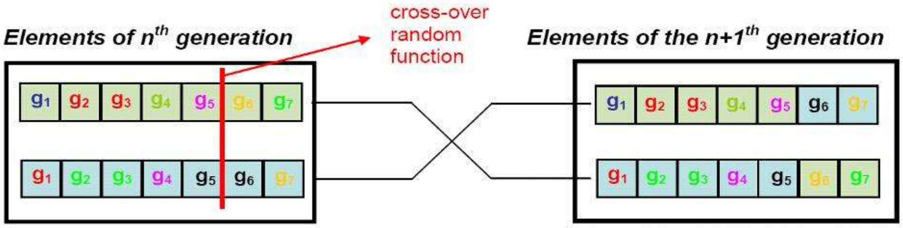
Elements of n^th generation
cross-over random function
Elements of the n+1^th generation
g1 g2 g3 g4 g5 g6 g7
g1 g2 g3 g4 g5 g6 g7
g1 g2 g3 g4 g5 g6 g7
g1 g2 g3 g4 g5 g6 g7

Figure 5: Scheme of the chromosomes crossover. At the left the parents, with the genes g_n. Different alleles are represented by different colours. At the right the sons after the crossover.

The reproduction of the chromosomes is obtained by crossing them over two by two, exchanging stochastically their genes. The evolution toward better individuals is regulated by a fitness function $F_{fitness}$ that evaluates the goodness of each chromosome and allocates a larger reproductive opportunity to the best ones. Each generation is never worse than the preceding one because the best chromosome is always reproduced in the successive generations.

The probability that a couple, with i≠j, produces, by crossing, an individual of the new generation is

$$P_{ij} = \frac{F^{i}_{fitness} \cdot F^{j}_{fitness}}{\sum_{i,j} F^{i}_{fitness} \cdot F^{j}_{fitness}} \qquad (2)$$

In this way, the couples whose partners have larger fitness function consequently have a larger probability to reproduce themselves. During each crossing over there exists a small probability that a gene undergoes a mutation. This expedient prevents the possibility that the iteration stops on a local solution and cannot evolve further.

In the code GMESA, the fitness function is defined in the following way:

$$F_{fitness} = \frac{I}{A_d} \qquad (3)$$

where A_d is the ellipses area union and I is the sum of the intensity (directly proportional to the charge) of the phace-space points that fall into the area A_d.

The gene mutation process, that happens with a very low probability, during each chromosome crossover, has been developed in a very accurate way. Such process is a very important point of a genetic optimization; it's the only way to introduce new genes in an offspring.

The GMESA code have four different mutation types:

1) A gene changes his value in a random way during the crossover.

2) The Twist parameter a, tied to the correlation x, x' of the beam, changes the sign. This means that the ellipse invert the slope.

3) One of the three Twist parameters reverses the position in all the eight ellipses of the chromosome.

4) The eight ellipses reverses the position into the chromosome.

The mutation probability, of all kinds of mutations, is dynamically managed and it increase when the code stops on a solution.

An expedient to force the optimization has been inserted in the code. After many generations, when the process improves very slowly or is stopped on a solution, that could not to be the best one, a new first population is generate by the best chromosome. The code samples the genes, of the new starting population, always in a random way, in a narrow reng around to the genes of the best chromosome. This expedient is iterated every n-steps if the best solution does not reach the demands.

In GMESE the necessary number of generations that starts this forcing optimization is 10^4 generations, which is the average number of steps to reach a good representation of a phase-space of the SPARC beam.

Understanding when the code has reached a good representation of the phase-space is delicate issue. For clear images the analysis is fast and are needed few steps, but when the analysis is dirty or however complicate, the code needs much more steps to reach a good solution. So a convergence parameter has been introduced in order to cope with any situation. When the user starts the code and decides the portion of charge that has to be kept in consideration, just after the downsizing of the matrix, the code makes an estimation of the beam emittance assuming a quasi uniform density ideal beam. This estimation of the emittance, relative to a percentage of charge lower then 100%, could be represented like an inner ellipse of the quasi uniform distribution. Therefore is possible to find a value (*i_cut*) of intensity of the phase-space points, named intensity cut, below which one no points falls into the ellipse Fig. 6.

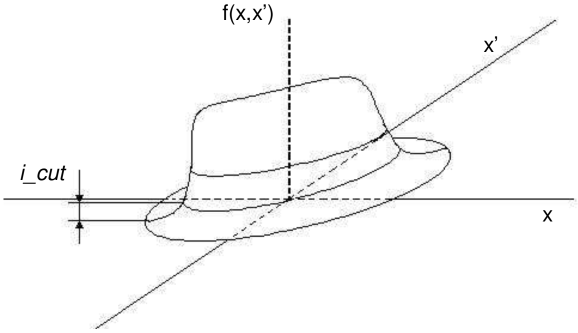

Figure 6: A quasi uniform density ideal beam f(x,x') distribution in the transverse phase-space and the intensity cut value *i_cut* concerning an arbitrary density contour.

In the image of a real beam, which is absolutely dirtier than an ideal beam, many phase-space points have an intensity lower than the intensity cut value; counting all these points, finally is possible to define the convergence parameter, which is the area directly proportional to the number of counted points. If the beam is ideal, such area is zero, whereas with the beam far from the ideal one, this area increase its size. When the code start a value is fixed, quite low, that is compared with the convergence parameter, every 2000 steps (or generations). If the parameter is lower the process is stopped, or else it carries on until the condition is reached. Whereas if the code do not reach the condition, the compared value is increased, by a fixed quantity, every 10^4 steps. The convergence parameter gives also an estimation of the confidence error of the method. Such parameter, which is an area, is directly proportional to the difference between the emittance of a ideal beam, quasi uniform distribution, and that one computed for the real beam, which can be only higher because to catch the same intensity of the ideal case, the ellipse could be only bigger.

2.2 Analytical test

During the development of the GMESA code, a few tests have been made. A phase-space image has been generated by three uniform elliptical distributions Fig. 7. A phase-space like this is a worth test case, because it shows the ability of the code to compute the rms emittance; in fact for an uniform distribution (a hyperellipsoid in n-dimensional space), the rms-emittance could be also analytically calculated from the geometrical emittance $\varepsilon_{rms} = \frac{\varepsilon_{geo}}{4}$.

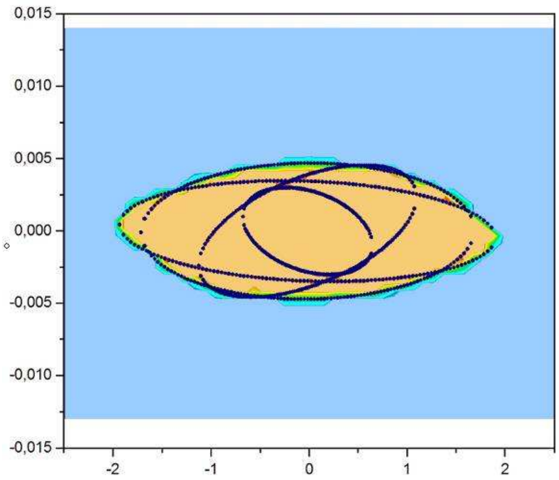

Figure 7: Phase-space test image composed by three uniform elliptical distributions

With GMESA have been done two analysis, one considering the 100% of the bunch charge and another one considering the 90% of the charge. In Fig. 8 is shown the ellipses placement of the best chromosome. It is worth showing that all the ellipses of the test case have been identically reproduced. It has to be underlined that the algorithm used to produce the test case works with an elliptical distribution generator different from that used in GMESA.

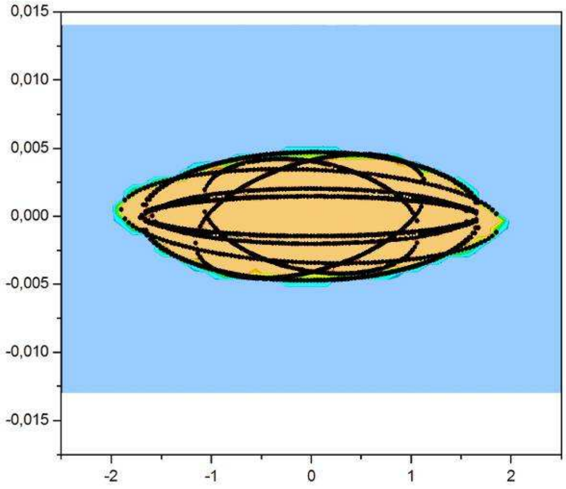

Figure 8: The eight ellipses representing the best chromosome in the test case analysis.

The result of the first test case, reported in Tab. 1, shows that GMESA has been able to reproduce the correct value, of the geometrical and rms emittance with very small difference due to the difference in the collected charge. If the value of emittance is rescaled from 99.12% to 100% the difference becomes really negligible. The value of emittance could be linearly scaled with the collected charge because the test beam is uniformly charged. Scaling the emittance again it is possible to show how the second test case gives a correct estimation of the emittance. These preliminary tests have been done with few genetic iterations, so the fraction of charge collected has small differences from that imposed from the user. With longer simulation the fraction of charge collected is really close to that imposed by the user.

	Test case	**GMESA 100%**	**GMESA 90%**
collected charge		99.12%	91.72%
ellipses union area [mm-mrad]	8.493	8.400	7.747
rms emittance [mm-mrad]	2.132	2.112	1.954

Table 1: Two preliminary test cases results.

Many other tests have been done directly on some particularly interesting phase-spaces, where the images shown more ellipses with boundaries well defined. In Fig. 9 is relevant the good agreement between the genetic solution, the union of the eight analytical ellipses, and the projected distribution of the real bunch. The analysis has been performed collecting a fraction of charge equal to 95% of the total one.

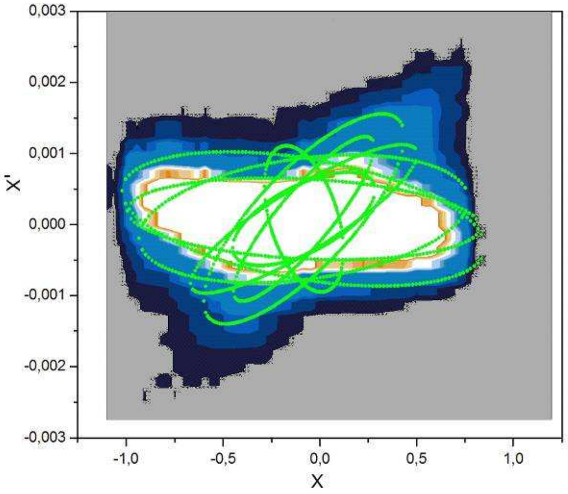

Figure 9: An image of the real beam, used as a test during the code developing.

3 Application of GMESE to the SPARC data

The SPARC photoinjector is an innovative electron GUN for LINAC accelerators and it has been designed in order to produce very high brightness electron beams. This Gun has to be able to produce an emittance curve that show a double minimum, with a very low emittance value, in the following drift. The emittance curve is build by more measures along the drift. These measure have performed with the movable emittance meter. Each single measure represent a phase-space image (or a point of the curve), which could be analyzed with GMESA to take out the value of the emittance.

With GMESA have been analyzed different emittance curves; in Fig. 10 is shown one of the more significant curve, which has been analyzed also with two other methods, a pepper pot algorithm based on profile filtering measured from beam images acquired from the movable emittance meter, and the code TEAM [7]. The agreement of the three different methods, based on different algorithms, is remarkable.

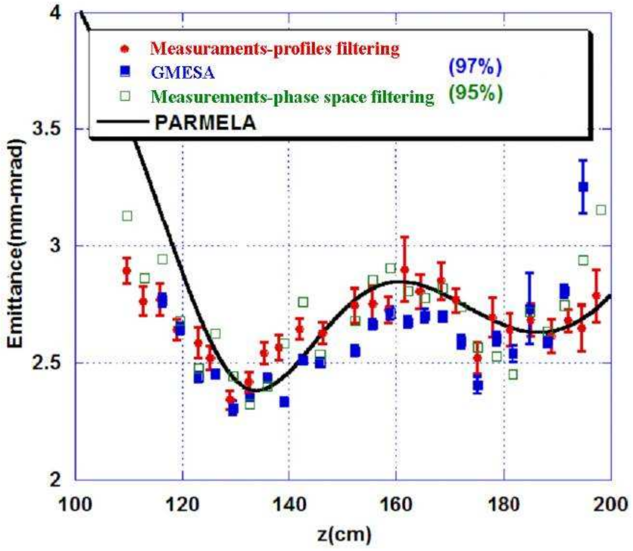

Figure 10: One of the more significant curve analyzed with GMESA compared with two other algorithms and a simulation of Parmela [8] about the dynamic of the electrons beam.

In Fig. 11 are shown some of the phase-space analyzed by GMESA and the analytical ellipsis solutions that follow the phase-space evolution of the beam for some Z position (referred to the figure 10)

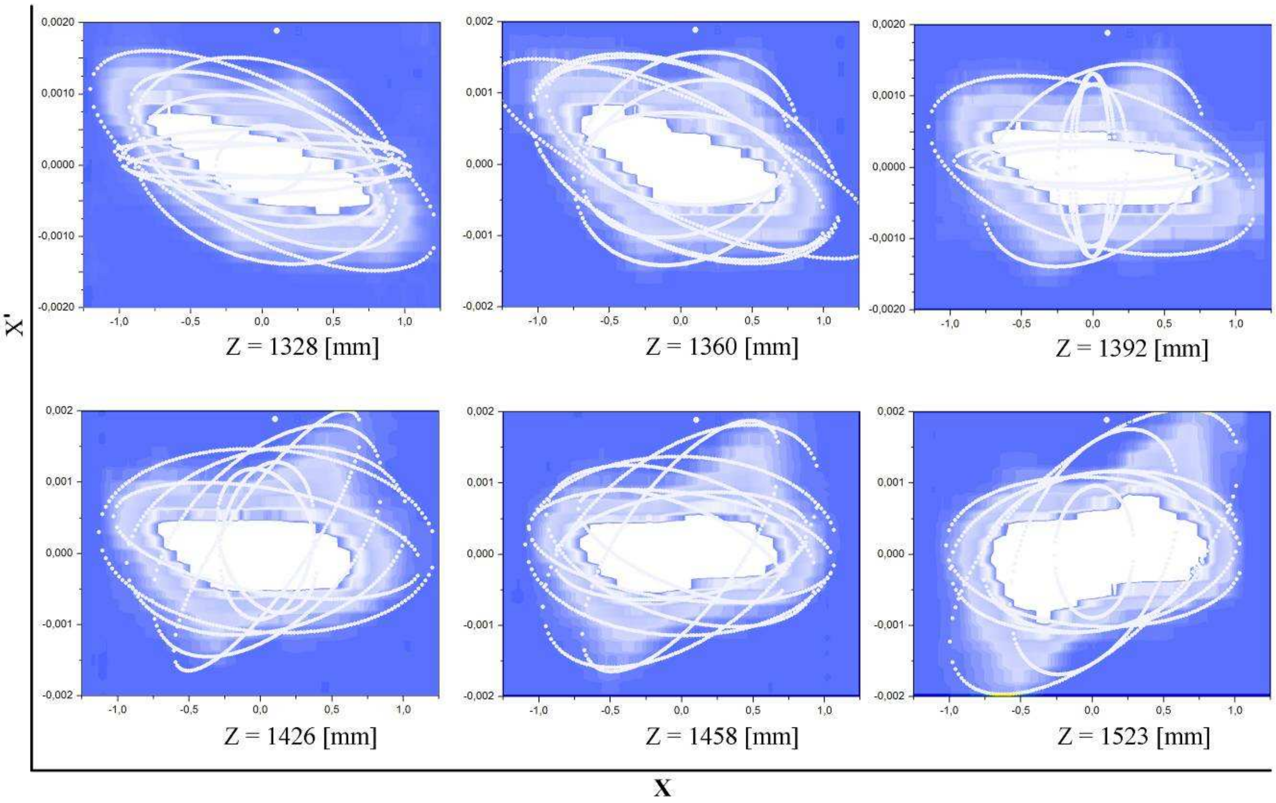

Figure 11: Some GMESA solutions overlapped to the real beam phase-spaces

4 Conclusion

The results obtained by GMESA have shown a strong agreement with other methods providing a confirmation of the algorithm and of the physical meaning of data.

During GMESA developing technical problems have been solved, that have given a robust structure to the code. Step by step GMESA has grown up more stable and independent from the user interaction; it can manage input data (phase spaces matrixes) of any kind of dimension and can accept NaN (Not a Number) input data converting them to computable values.

From first tests the code seem to be able to analyze the raw, not interpolated, phase space images coming from the pepper pot algorithm, but more case have be yet analyzed.

The next step that is under development is attributing to each ellipse a fraction of the total bunch charge. This upgrade will permit to GMESA to perform a real tomography of the beam.

Acknowledgments

A special thank goes to Vittoria Petrillo, Luca Serafini and Andrea Renato Rossi from which I received valuable suggestions to develop the code.

I thanks also all the SPARC data analysis group with which I worked exchanging many information and ideas

References

[1] L. Davis, Genetic Algorithms and Simulated Annealing. London, U.K.: Pittman, 1987.
[2] K. A. DeJong, “An analysis of the behavior of a class of genetic adaptive systems,” Ph.D. dissertation, Univ. Michigan, 1975.
[3] M. Ferrario and al, SLAC-PUB 8400.
[4] L. Catani and al, Review of Scientific Instruments 77, 93301 (2006).
[5] D. Filippetto, Sparc Note SPARC/EBD-07/002.
[6] Lawson, “The Physics ok Charged Particle Beams”.
[7] A. Cianchi and al., Sparc Note SPARC/EBD-07/003.
[8] L. M. Young, Los Alamos National Laboratory report LA-UR-96-1835.