Phase Space Tomography of an Electron Beam in a Superconducting RF Linac

Tianyi Ge*[1,†] and Jayakar Thangaraj[2,‡]

[1]*Department of Physics and Astronomy, University of California, Los Angeles, California 90095, USA*
[2]*Fermi National Accelerator Laboratory, Batavia, Illinois 60510, USA*

(Dated: August 20, 2015)

In this paper, different methods to reconstruct the phase space distribution of an electron beam from a superconducting rf linac are presented using ELEGANT input data. Results from computed tomography (CT) simulation, such as Algebraic Reconstruction Technique (ART) and Simultaneous Algebraic Reconstruction Technique (SART), both qualitatively and quantitatively agrees with the ELEGANT input. The SART algorithm is especially discussed in detail because of its easy implementation, speed, and accuracy to restore the original phase space. Further results on tomographic principles and simulation are shown.

FAST BEAMLINE

Over the years, the accelerator physics community has seen a growing interest in studying fundamental limitations to beam intensity and developing approaches to beam generation, acceleration and manipulation. The Fermilab Accelerator Science and Technology (FAST) incorporates a 1.3-GHz superconducting radiofrequency linac in conjunction with a photoinjector and a storage ring capable of storing electrons or protons. The FAST facility supports a broad range of extremely stable and high brightness electron beam experiments. Consisting of various diagnostic devices , the FAST beamline especially aims at determining the detailed characterization of high brightness electron bunches from the photoinjector. In this paper, various methods are presented to measure the transverse phase space distribution of electron beams from a superconducting rf linac. With a changing quadruple strength, a full range of 180° 1D projections of the beam are measured on the screen and a 2D phase space is accordingly reconstructed using Algebraic Reconstruction Technique (ART). With a changing quadruple strength K_1, the unknown beam will generate 180° 1D histograms of position x on the screen. The desired 2D phase space $x-x'$ transverse can be reconstructed from the position x data collected. The same technique can be extended to the reconstruction of the $y-y'$ transverse phase space as well as to the longitudinal $t-p$ longitudinal phase space.

ALGEBRAIC RECONSTRUCTION ALGORITHMS

Algebraic Reconstruction Techniques

With the output data of transverse $x-y$ from ELEGANT on a screen, we explored various tomographic methods to reconstruct the original phase space from the photojector. Especially, we investigate different iterative algorithms for solving linear equation systems and apply them to the reconstruction process. Algebraic Reconstruction Techniques (ART) have unique advantages in solving large linear equation systems as it can be implemented relatively easily. The illustration of ART is demonstrated in Figure 1. A $n \times n$ evenly-spaced grid of size δ is put on a 2D image. Each block is assigned with the scaled pixel value f_j of the image (0-255). In the ART simulation, we shoot a total number of m light rays, $P_1 \ldots p_m$, across the image for each angle θ_k which spans 180°. w_{ji} denotes the normalized weight coefficient of how much the jth cell with pixel value f_j contributes to the 2D reconstructed image when $i-$th ray penetrates the cell for each projection angle θ_k. Since the projection

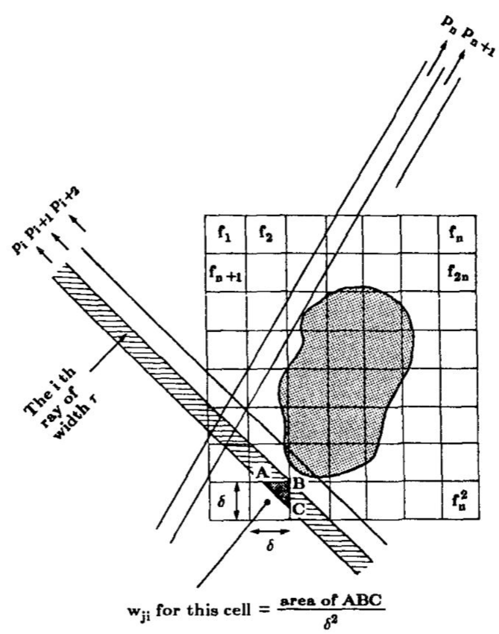

FIG. 1: Illustration of ART algorithm [1].

transformation is linear and we assume that no diffraction or bending occurs while light propagates throughout

---

*Author

Method Name	Elapsed Time(s)	α	Error(%)	β	Error(%)
ELEGANT	≈ 0	61.3245	0	30.2357	0
SART	0.864	65.7396	7.20	24.2348	19.85
CIMMINO	2.197	64.1310	4.58	23.4886	22.31
CAV	2.411	64.1324	4.58	23.4892	22.31
DROP	2.056	64.4792	5.14	23.6570	21.76
ART	798.641	63.0229	2.77	23.0796	23.67

TABLE I: Measured elapse time of MATLAB internal execution and Twiss parameters from various tomographic algorithms for reconstructed $x - x'$ phase space.

the 2D image, our task reduces to find the solution of a linear system n equations with m unknowns.

$$P_i = \sum_{j=1}^{N} w_{ji} f_j \tag{1}$$

where P_i stands for the ith ray, w_{ji} the weight matrix, and f_j the pixel value of jth cell. The equation (1) has an expanded form.

$$\begin{aligned} w_{11}f_1 + w_{12}f_2 + \cdots + w_{1N}f_n &= P_1 \\ w_{21}f_1 + w_{22}f_2 + \cdots + w_{2N}f_n &= P_2 \\ &\vdots \\ w_{m1}f_1 + w_{22}f_2 + \cdots + w_{2N}f_n &= P_n \end{aligned} \tag{2}$$

ART can be implemented to extract the coefficients w_{ji} of the unknowns in the equation system and then treat them as matrix W. Therefore, the matrix form of the linear equation system to solve can be written as:

$$W\,x = P, \quad W \in \mathbb{R}^{n\times m} \text{ and } P \in \mathbb{R}^n \geq 0 \tag{3}$$

where the contribution of each cell w_{ji} must be positive. The weight matrix can be constructed from the following equation [1]:

$$w_{ji} = \frac{A_{ji}}{\delta^2} \tag{4}$$

where A stands for intersection area of the jth ray with ith cell and δ is the size of each cell. Moreover, P matrix can be experimentally collected from the projection data on the screen. The classical method of reaching the solution x of the equation system (3) is the Karczmar method [2], which proposes an initial guess x_0 and reiterates using the following algorithm. For $k = 0, 1, \ldots,$

$$x_{k+1} = x_k + \frac{P_i - \langle W_i, x_k\rangle}{||W_i||_2^2} W_i \tag{5}$$

where W_i denotes the ith row of matrix A and b_i denotes the ith projection data measured on the screen. The convergence condition is imposed by the following equation [2].

$$\mathbb{E}||x_k - x||_2^2 \leq \left(1 - k(W)^{-2}\right)^k \cdot ||x_0 - x||_2^2 \tag{6}$$

where $\mathbb{E}$ stands for the error, $k(W)$ the scaled condition number [2], x_0 the initial guess and x the solution. Since the classical ART method in equation (5) processes the row W_i of matrix W one by one, another method of solving equation (3) is proposed in this paper to process all n rows of matrix W at the same time in one iteration, therefore significantly reducing the running time.

Simultaneous Iterative Reconstruction Techniques

The Simultaneous Iterative Reconstruction Techniques (SIRT) are a group of iterative algorithms that process the entire equation system at the same time. The general SIRT group algorithm is implemented from the following algorithm in MATLAB [3].

$$x^{k+1} = x^k + SA^T M(b - Ax^k), \quad k = 0, 1, 2, \ldots, \tag{7}$$

where S and M are symmetric positive definite and depend on the ratio of W_i and $||b_i||_2^2$. The convergence condition for the SIRT group is the same as dictated by equation (6). The running time and Twiss parameters using ART and SIRT group algorithms are summarized in Table 1.

6D PHASE SPACE RECONSTRUCTION

Theory of Phase Space Reconstruction

The 6D phase space of beam passing through a quadruple undergoes an affine transformation as described by a

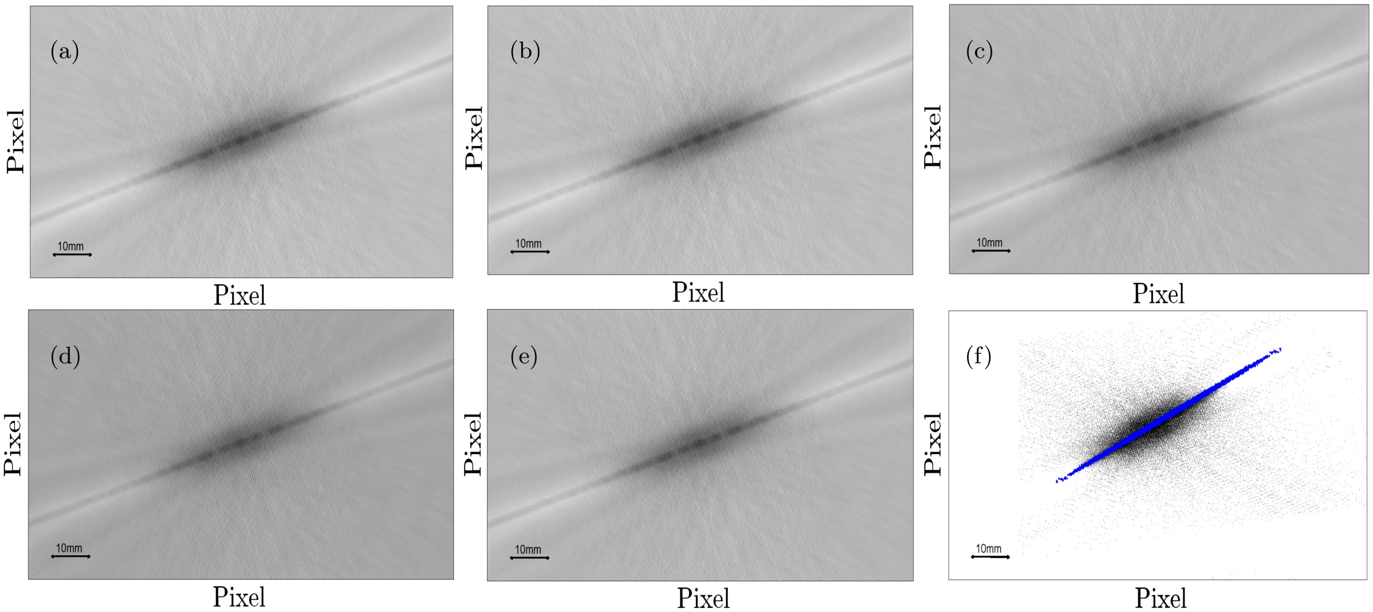

FIG. 2: (a): CAV method with running time 2.411 s. **(b):** CIMMINO method with 2.197 s. **(c):** DROP method with 2.056 s. **(d):** ART method with **784.641** s. **(e): SART** method with **0.864** s. **(f):** phase space contrast between ELEGANT input(**blue**) and **sart**(**black**) simulation.

transport matrix. For example, the element of incoming phase space $x_0 - x'_0$ is related to the outcoming $x - x'$ element by the following transformation [4]:

$$\begin{pmatrix} x \\ x' \end{pmatrix} = R \begin{pmatrix} x_0 \\ x'_0 \end{pmatrix} = \begin{pmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{pmatrix} \begin{pmatrix} x_0 \\ x'_0 \end{pmatrix} \tag{8}$$

and

$$\begin{pmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ k & 1 \end{pmatrix} \begin{pmatrix} e_1 & 0 \\ 0 & e_2 \end{pmatrix} \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \tag{9}$$

, where the transport matrix $\mathbf{R}$ can be decomposed into the combination of a shear, expansion, and rotation matrix. The scaling constant e_1 and rotational angle θ can be calculated from the ELEGANT output R_{11} and R_{12}.

$$e_1 = \sqrt{R_{11}^2 + R_{12}^2}, \quad \theta = \tan^{-1}\left(\frac{R_{12}}{R_{11}}\right) \tag{10}$$

The ELEGANT output R_{11} and R_{12} are generated from its input quadrupole strength K_1. The relation between K_1 and rotational angle θ in a single quadrupole setup is described by the following equation [5].

$$\theta = \tan^{-1}\left(\frac{\frac{1}{\sqrt{K_1}}\sin\left(\sqrt{K_1} l_1\right) + l_2\cos\left(\sqrt{K_1} l_1\right)}{-l_2\sqrt{K_1}\sin\left(\sqrt{K_1} l_1\right) + \cos\left(\sqrt{K_1} l_1\right)}\right) \tag{11}$$

, where l_1 is the length of the quadrupole, l_2 the drift length from the quadrupole to the screen, and K_1 the strength of the quadrupole. Because we aim to cover a range of 180° projections with one angle in every 10°, we insert the desired angles $\theta = 5°, 15°, 25°, \ldots, 175°$ into equation (11) and numerically solve for an array of K_1 accordingly. Nevertheless, since the limit of quadrupole strength K_1 can only go from -40 to 40 m^{-1} in experiment, we neglect such K_1 from our result and their corresponding rotational angle θ as well.

Instead of tracing the phase space for every particle, we discretized our projection image $x - y$ into a square matrix of fixed size, 656×656 for $x - x'$, and extracted the summation $p_m(x, \theta_k)$ of pixel value from each column. Then, we followed [4] to assemble and rescale the projection file $p_m(x, \theta_k)$, for $k = 1, 2, ..., n$ for n is the total number of projections taken for $x - x'$ [4]. The method is described in the following equation.

$$p_r(x, \theta_k) = e_1 p_m(x, \theta_k) \tag{12}$$

, where $p_m(x, \theta_k)$ is the summation $p_m(x, \theta_k)$ of pixel value from each column and $p_r(x, \theta_k)$ is the rotation-only matrix. Lastly, we applied ART, SART and several other algorithms to the assembled data $p_r(x, \theta_k)$. We used different methods in the previous section to reconstruct the $x - x'$ 2D phase space from our experimental data and compared them to the ELEGANT input. The result of 2D phase space reconstruction is presented in Figure 2. The initial reconstruction using the ART algorithm in **(d)** took around 13 minutes for MATLAB to run. However, by using the SART algorithm in **(e)**, we successfully reduced the running time to less than 1 s. Moreover, the SART reconstruction well agrees with the ELEGANT input in **(f)**.

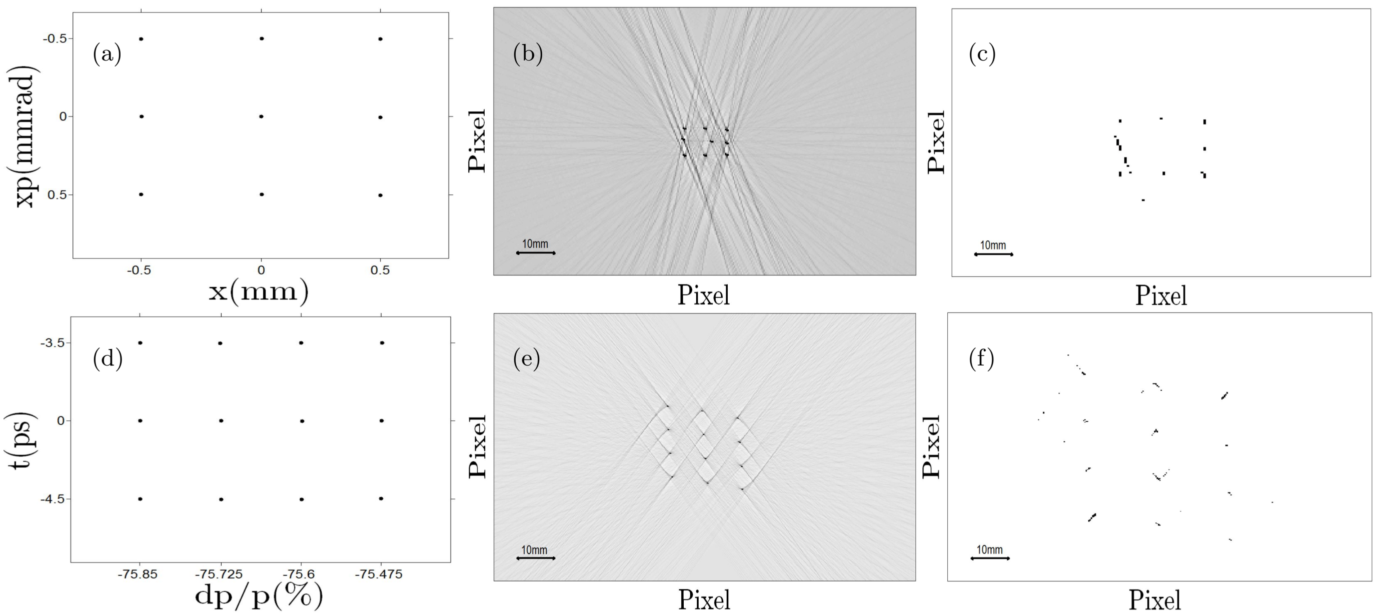

FIG. 3: (a): ELEGANT input of 3×3 pixel in $x - x'$. **(b):** Reconstruction of 3×3 pixel in $x - x'$ using SART. **(c):** Reconstruction of 3×3 pixel in $x - x'$ using SART with background reduction. **(d):** ELEGANT input of 4×3 pixel in $t - p$. **(e):** Reconstruction of 4×3 pixel in $t - p$ using SART. **(f):** Reconstruction of 4×3 pixel in $t - p$ using SART with background reduction.

Transverse and Longitudinal Phase Space Reconstruction in SART

In order to test the accuracy of SART and conduct future calibration, we input 3×3 pixels into ELEGANT and aim to reconstruct these pixels by using SART algorithm. In the simulation of $x - x'$ case, we took a total number of 18 projections and assembled these data according to the steps illustrated from the last subsection. We followed the almost same procedure to reconstruct the $t - p$ phase space but some modifications. We replaced all the R_{11} and R_{12} in the previous subsection with R_{65} and R_{66} from ELEGANT. We also input 4×4 pixels into ELEGANT and aim to reconstruct these pixels by using SART algorithm. The reconstruction results of both $x - x'$ and $t - p$ are demonstrated in Figure 3.

CONCLUSION

The various techniques used in this paper enable us to reconstruct and study the original 3D phase space of beam. In addition to the phase space tomography, we developed a much faster algorithm SART, reducing the running time from 13 minutes for ART down to less than 1 s. The reconstructed 3D phase space qualitatively reflects the phase space of incoming beam. The calculated Twiss parameters using SART also agrees with our ELEGANT input in a quantitative manner. Nevertheless, the main limitation of ART or SART algorithm is that a range of 180° must be taken in order to obtain a less noisy phase space. In our simulation, we pick 18 angles, one in each 10°, in the range of 180°.

Several features can be improved in our SART algorithm. First, in order to accurately describe a more complicated beam in experiment, we can add more quadrupoles in our ELEGANT simulation and revise our codes accordingly. Second, emittance calculation is neglected in our paper because our simulation is far different from input, which we believe is due to the missing conversion factor in our codes. Third, a quantitative agreement of Twiss parameters between input and simulation depend on background reduction. However, for different input, we don't have a fixed level to reduce the background. We aim to develop an algorithm to reduce the background in the future such that the calculation of Twiss parameters from simulation can be as close as possible to ELEGANT input.

---

† T. Ge `getianyi@ucla.edu`
‡ J. Thangaraj `jtobin@fnal.gov`

[1] A. C. Kak and Malcolm Slaney, *Principles of Computerized Tomographic Imaging*, ISBN 978-0-898-71927-7, Society of Industrial and Applied Mathematics (2001).

[2] T. Strohmer, R. Vershynin, "A randomized Kaczmarz algorithm with exponential convergence", Journal of Fourier Analysis and Applications, Volume 15, Issue 2 , pp 262-

278 (2009).

[3] P. Hansen, M. Hansen, "AIR Tools A MATLAB package of algebraic iterative reconstruction methods", Journal of Computational and Applied Mathematics, Volume 236, Issue 8 , Pages 21672178 (2012).

[4] D. Xiang et al, "Transverse phase space tomography using a solenoid applied to a thermal emittance measurement", Phys.Rev.ST Accel.Beams 12 (2009).

[5] D. Stratakis, "Tomographic measurement of the phase space distribution of a space-charge-dominated beam", Ph.D. dissertation, University of Maryland, College Park,Chap. 4 (2009).

ACKNOWLEDGEMENT

I am grateful to my mentor Jayakar Thangaraj for guiding me through this research project. In addition, I would like to thank Lee Teng Internship for funding my project and Eric Prebys for organizing the internship.

All data analyses were performed in *Matlab R2013a*. Instructions regarding how to use *Matlab* can be found on the *Matlab* official website `http://www.mathworks.com`.