Point-to-Curve method of distance determination for use in linearization and particle identification

L. W. May, Z. Kohley, S. Wuenschel, K. Hagel, R. Wada, and S. J. Yennello

Particle identification in multi-detector arrays is done in various different ways. In the analysis of data taken on the NIMROD-ISiS array at the Cyclotron Institute at Texas A&M, a linearization technique is generally used in order to particle ID the various fragments picked up by the detector [1]. An important step in this linearization is determining the proximity of specific data points to lines in the data corresponding to specific elements. A new method of distance determination was developed in an attempt to improve upon this process of the linearization.

The new method, known as the Point-to-Curve method, was developed as a way to determine the exact and absolute distance between any given data point and a user-generated line defining an element on a ΔE-E plot. Previous methods had used a simple tracing in either the horizontal or vertical direction from the point until finding the nearest elemental line. Due to the known issue of the vertical and horizontal methods having trouble with data sets of very vertical or horizontal curvature, respectively, later iterations of the analysis code used a method that combined these ideas by searching along a 45° line. The Point-to-Curve method was created to provide the shortest distance calculation between a data point and any given curve representing an element.

The Point-to-Curve method uses the parameterization of the equation for the curve into a vector in terms of x and y values. For instance, a curve denoted by the equation $y = 3x^2-6x+10$ would become the vector $r = (x, 3x^2-6x+10)$. The value for the point in question, $p = (x_1, y_1)$, is then subtracted from the vector r giving $r-p = (x-x_1, 3x^2-6x+10-y_1)$. Since $r - p$ is a measure of the deviation in x and y between the point and the curve, this value can be squared by taking the dot product $(r - p) \bullet (r - p)$. This equation is minimized yielding points on the curve, one of which will be the minimum for calculating the distance between the point and curve. The minimization is done by setting the derivative of the dot product equal to zero and solving for x. These values can be both real and imaginary, but as imaginary values give non-physical results, these values can be discarded. Since several possible real roots can be given as solutions, constraints can be placed on the output such that only the value within the desired range will be found. The one remaining solution is the x-intersection of the curve and the line drawn from the point to the curve with the smallest possible distance. This x-intersection value can then be used to find the y-intersection value from the equation of the curve and the distance formula applied to the intersection values and the value of the point to give the actual distance between the point and the curve.

The computationally intensive step in this method is in the factorization of the polynomial. Polynomial factorization has been demonstrated to be an NP-complete problem, meaning that the time it takes to factor the polynomial scales exponentially with the size of the polynomial. This is not a problem for polynomials of order 4 or less, since these can be solved analytically using known algorithms. A problem could arise if high order functions are used to model the line. However, for the types of plots common for ΔE-E and CsI fast-slow data this is not prohibitive as the use of spline fits allows the order of the resulting polynomial to be kept low enough (no greater than 5 or 6) that the time to factor the polynomial is quite low. The factoring algorithm used in this case is that found in the GSL library packages linked in the ROOT analysis package.

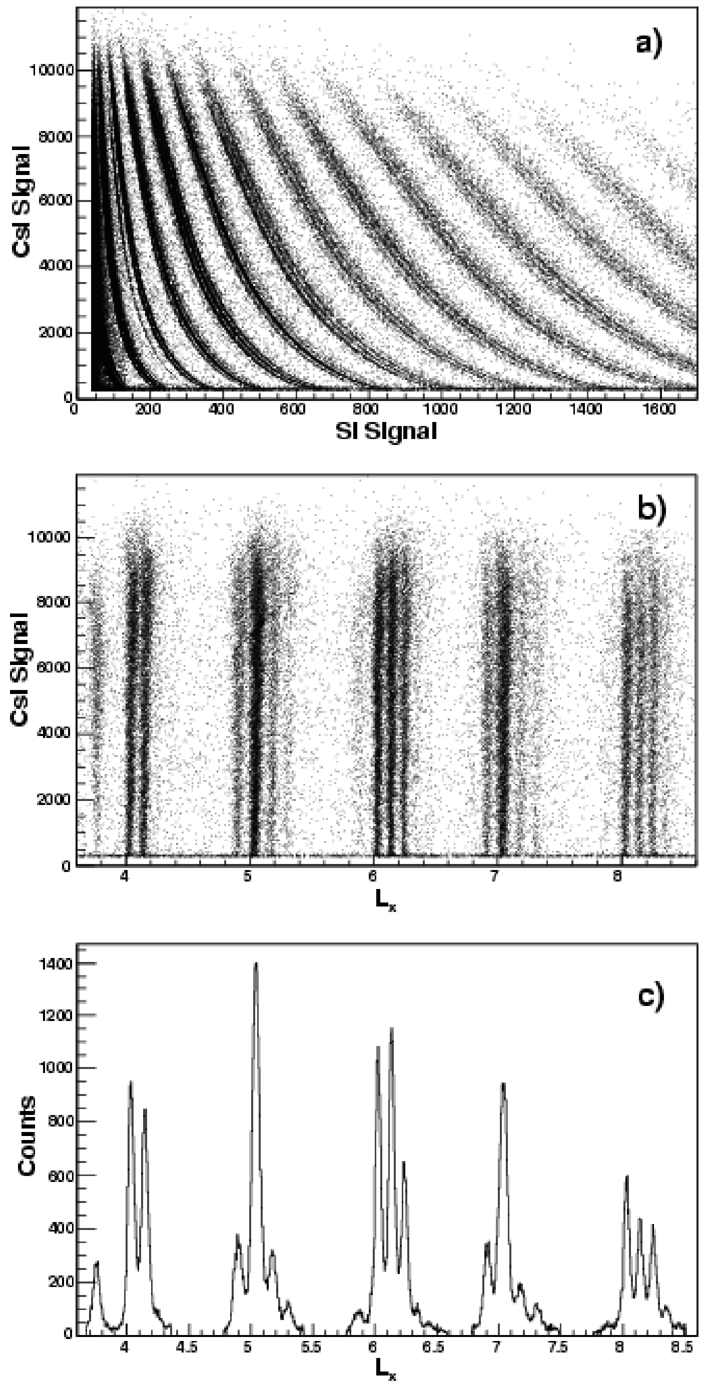

FIG. 1. An example step-through of the linearization process where a) A Si-CsI ΔE-E plot, b) a 2D plot of linearized x-axis versus the CsI signal and c) a 1D projection of the center plot with linearized x-axis versus counts calculated with the Point-to-Curve method. The linearized x-axis shows isotopes for Beryllium through Oxygen (Z=4-8).

While simulations of the Point-to-Curve method have shown that it indeed does return the shortest distance from a point to the line (compared to the horizontal, vertical and 45° line methods), current testing analysis on data of a 35MeV/A ^{70}Zn beam on a ^{70}Zn target taken using the NIMROD-ISiS array comparing the four methods gave interesting results [2]. A step-by-step breakdown of the linearization process can be seen in Fig. 1. The data showed no difference (within errors) between the horizontal, the 45° line or Point-to-Curve methods. The vertical method had great difficulty in analyzing the data due to the very vertical nature of the data set. While there is no difference in this particular data set (as seen from Table I), one can see that in deciding whether to use the horizontal or vertical methods, a decision must be made based on the curvature of the data. The Point-to-Curve method was developed for the purpose or performing equally well over all portions of the curvature of data so that one does not have to worry about the curvature of a data set ahead of time but rather can just use a single distance determination method for all data.

TABLE I. Peak sigmas with errors and percent contamination values with errors for the four distance calculation methods used on the ^{9}Be and ^{10Be} peaks.

	Peak	σ	σ error	% contamination	% contamination error
Vertical method	^{9}Be	0.01974	0.00039	41.861	1.173
	^{10}Be	0.06951	0.00054	16.225	0.798
Horizontal method	^{9}Be	0.01687	0.00028	3.908	0.408
	^{10}Be	0.02760	0.00044	4.067	0.399
45° line method	^{9}Be	0.02590	0.00028	3.835	0.407
	^{10}Be	0.02755	0.00045	4.015	0.399
Point-to-Curve method	^{9}Be	0.02568	0.00027	3.905	0.358
	^{10}Be	0.02764	0.00036	4.061	0.360

[1] S. Wuenschel *et al*, Nucl. Instrum. Method Phys. Res. **A604**, 578 (2009).
[2] Z. Kohley, private communication.