



A novel clustering approach to positron emission particle tracking



Cody Wiggins^{a,*}, Roque Santos^b, Arthur Ruggles^b

^a University of Tennessee-Knoxville, Department of Physics and Astronomy, 1408 Circle Drive, Knoxville, TN 37996, United States

^b University of Tennessee-Knoxville, Department of Nuclear Engineering, United States

ARTICLE INFO

Article history:

Received 6 August 2015

Received in revised form

20 November 2015

Accepted 30 November 2015

Available online 18 December 2015

Keywords:

Positron emission particle tracking

G-means

Clustering

Multiple particle tracking

ABSTRACT

A novel approach to positron emission particle tracking is presented based on determining regions of space with high density of line of response crossing via clustering. The method is shown to be able to accurately track multiple particles in systems where the number of particles is unknown and in which particles can enter and leave the field of view of the scanning system. This method is explored in various environments and its parametric dependence is studied.

© 2015 Elsevier B.V. All rights reserved.

1. Introduction

Positron emission particle tracking (PEPT) is a measurement technique in which positron emission tomography (PET) technology is used to locate the position of a moving tracer [17]. The tracer is labeled using a positron-emitting isotope such as F-18, Ga-68, or Na-22. This positron will annihilate in the near vicinity of the tracer particle, emitting back-to-back, coincident gamma rays. By detecting these coincident gamma rays, one can draw a “line of response” (LOR) between the two detection sights. In theory, each of these LORs will pass near the location of the tracer. Thus, by examining the positions of thousands of these LORs, one can calculate the position of the tracer as it moves throughout the bore of the PET scanner.

The original and most prominent PEPT algorithm is that developed by Parker et al at the University of Birmingham (1993). In this method, a chronologically collected sequence of LORs is arranged into groups of N LORs, each group collected within a specific time interval. Using these N events, one can calculate the position of the tracer as the point in space that minimizes the sum of the distances from the tracer to each of the LORs. In this method, an iterative approach is used to remove LORs that are deemed corrupt (either from Compton scattering or random coincidence). Lines further than a preset distance from the calculated tracer position are rejected iteratively until only a fraction f of the initial N LORs remains.

This method has been shown to successfully track a single positron-emitting particle in a number of experiments for various applications [5,7,8,18,23]. This method has also been expanded to use

in tracking multiple particles in the case that the tracers are labeled with different, predetermined activities [24]. In this implementation, the Birmingham algorithm is set up to first find the particle with the highest activity. All LORs which are used to triangulate this position are then removed from the field of view and the process is repeated to find the particle with the second highest activity. This process is continued until all particles in the field of view have been located. This approach has been used to track up to three particles at once, but it is limited by the necessity to have sufficiently different activities on the tracers while not saturating the detector’s data acquisition system.

2. Line density algorithm

Another approach to both single and multiple PEPT has been developed by Bickell et al [6]. In this method, called the “line density algorithm”, a Cartesian, 3-dimensional grid is laid over the field of view (FOV) of the scanner. LORs are then collected over a preset time slice (usually a few milliseconds) of the total data set. The algorithm counts the number of LOR crossings at each point in the Cartesian grid. The voxel with the highest number of line crossings in this count matrix is then considered as the source of the gamma rays, and one-voxel-wide slices are taken in the x , y , and z -directions. Gaussian fits are applied to the count data for each of these slices with the centroid taken to be the position of the tracer and the uncertainty in each direction to be the full width at half maximum (FWHM) of each of these fits divided by the square root of the number of contributing LORs. Tests with this algorithm have shown to be comparable in accuracy to the Birmingham method previously described.

This algorithm can also be used to track multiple tracers when the initial positions of the tracers are known. By specifying the

* Corresponding author.

E-mail address: cwiggins2@vols.utk.edu (C. Wiggins).

regions of the FOV in which each particle is known to begin, the line density algorithm can identify local peaks in the count matrix and determine the location of each particle accordingly. Subsequent particle locations can be found by searching for particles in regions near each particle location from the previous time step. By examining the particle locations from multiple previous time steps, one can use each tracer's velocity and acceleration data to further refine each search region in the FOV.

3. A new algorithm

The previously described methods are limited in that they require *a priori* information of the number of tracers and their locations (in the case of the line density method). It is desired to have a measurement technique in which this information is not needed and in which particles are allowed to enter and leave the field of view of the scanner. Such a method is desirable for the case of fluid flow measurements in which a single test section can be studied as part of a recirculating flow loop. This paper explores a new adaptation of the line density algorithm in which no *a priori* information is required of the system, allowing for a variable number of tracers within the FOV.

3.1. Line density with G-means clustering

A new algorithm is proposed for both single and multiple PEPT that is based on finding clusters within the count matrix obtained within the line density method. The method begins in the same way as the traditional line density algorithm, dividing the data into time slices and counting the number of LOR crossings at each point in a Cartesian grid for each of these time slices. The next step is to filter the data before beginning the clustering process. First, a check is made to determine whether there is any particle(s) in the scanner's field of view or not. A user-input threshold is set such that if there is no grid element with a number of LOR crossings greater than or equal to this number, the time step is skipped. If this thresholding condition is met, the maximum number of LOR crossings at a given grid point is found. Then a preset fraction of this maximum value is subtracted from each point in the count grid, with any point having fewer counts than this fraction being set to zero. This fraction is determined by the user based on knowledge of each particle's activity relative to the background, and is usually set somewhere in the range of 0.25–0.5. The remaining data can thus be viewed as points in \mathbf{R}^3 with a multiplicity corresponding to the number of line crossings at each point. This data can be grouped into k clusters with the centroid of each cluster being taken to be the location of a tracer particle. However, such clustering can be quite difficult if the number of clusters (and thus particles) k is not known prior to calculation. Due to the isotropic distribution of positron emission about each tracer, it is expected that each cluster of LOR crossings should be normally distributed about the true particle position, and this feature can be used to discriminate between true and false clusters. For this reason, the method of G-means clustering is used.

3.2. G-means clustering

Gaussian-means (G-means) clustering is an adaptation of the well-established k -means algorithm [14] that allows clustering when the number of natural clusters in a dataset is unknown. In this method, developed by Hamerly and Elkan [9], principal component analysis and goodness-of-fit testing are used to determine the number of clusters k in a data set as well as their locations. The method begins by performing a k -means clustering of the dataset with $k=1$ (or a higher number if any *a priori* information is known) and subsequently splitting or accepting each cluster based on its adherence to a Gaussian fit. In this way, it grows the number of clusters in the dataset until it

reaches the number of natural clusters. In the case of our PEPT study, calculations are always started with a $k=1$ clustering (i.e. a universal centroid calculation). As such, this method is deterministic and does not risk the potential false convergence error of k -means caused by poor initialization.

The splitting process is performed based on a statistical testing of each cluster for normality. If the data appears to be normally distributed, the cluster is accepted. If it does not, the cluster is split into two. The statistical test is performed based on the adherence of the cluster to a one-dimensional Gaussian fit. This process is described as follows.

Consider a data set S of points in d -dimensional space (in the case of this method, $d=3$) with the data already divided into k clusters, $X_j \subset S$, where $j \in \mathbf{J} = \{1, 2, \dots, k\}$. Now consider a specific cluster, X_m , where $m \in \mathbf{J}$, containing n points. Power iteration [2] is performed on the covariance matrix for X_m and used to identify its main principal component and the corresponding eigenvalue λ . One then initializes two daughter centroids, c_1 and c_2 , along the main principal component, a distance $\pm \sqrt{(2\lambda/\pi)}$ from the centroid of X_m , and runs k -means on the cluster with $k=2$ and c_1 and c_2 as the initial cluster centers. After c_1 and c_2 converge on new values, c_1' and c_2' , one defines the line $v = c_1' - c_2'$. A one-dimensional projection of each data in X_m is taken along v such that $x'_i = \langle x_i, v \rangle / \|v\|$. This new data set X_m' is a one dimensional representation of the data in X_m , and is then transformed so that it has mean 0 and variance 1.

A one-dimensional Anderson–Darling (A–D) test [1] is then performed on the data in X_m' to test if it is normally distributed. For each of the n values $x'_i \in X_m'$, let $z_i = F(x'_i)$, where F is the $N(0,1)$ cumulative distribution function. Then the A–D statistic, A^2 , is defined as

$$A^2(Z) = -\frac{1}{n} \sum_{i=1}^n (2i-1) [\ln(z_i) + \ln(1-z_{n+1-i})] - n$$

It has also been shown [21] that for data sets where the mean and variance are estimated from the data itself, a correction must be applied to the A–D statistic as follows:

$$A_*^2(Z) = A^2(Z) \left(1 + \frac{4}{n} - \frac{25}{n^2} \right)$$

In the case of this method, if the A–D statistic is below a given critical value the original cluster X_m is accepted. If it is not, the cluster is split, and a k -means clustering of the entire data set S is performed with $k'=k+1$, and centroids initialized at the daughter centers, c_1' and c_2' , and the centroids of the remaining centroids X_n , ($n \neq m$). This process is continued until all clusters pass the A–D test.

3.3. G-means example

Fig. 1 shows an example of the splitting of clusters achieved by the G-means algorithm. The example uses actual 3-dimensional data acquired from a MicroPET P4 preclinical PET scanner [22] with three activated particles placed near the scanner's center of FOV. With three particles, it can be seen that the filtered, line density dataset has three natural clusters. In this case, the critical value for the A–D test is arbitrarily taken to be 20, based on the experience¹.

First, the universal centroid is acquired. In the first run G-means, the daughter centers are found, and the A–D statistic for the cluster is calculated as 143.2, thus the split is accepted. The daughter centers and A–D statistic are then found for the lower cluster, and it is found that $A_*^2 = 84.25$, meaning that this cluster is

¹ More precise critical values for the A–D test exist [21] but were derived under the assumption that the data set being tested is continuous. In the case of our discrete data, they were found to be too restrictive. It has been determined through experience that the optimal A–D critical values for our PEPT studies are between 10 and 25.

Download English Version:

<https://daneshyari.com/en/article/1822235>

Download Persian Version:

<https://daneshyari.com/article/1822235>

[Daneshyari.com](https://daneshyari.com)