An algorithm for baseline correction of MALDI mass spectra

Betsy Williams[*†] Shannon Cornett[‡] Anna Crecelius[‡] Richard Caprioli[‡]
Benoit Dawant[†] Bobby Bodenheimer[†]

†Department of Computer Science, Vanderbilt University, Box 1679 Station B, Nashville, Tennessee 37235 USA
‡Department of Biochemistry, Vanderbilt University Medical Center, 607 Light Hall, Nashville, Tennessee 37232 USA

ABSTRACT

Visualization and differentiation of proteins in tissue are problems of increasing interest in computational systems biology, bioinformatics, and image processing. A platform for generating such proteomic information is matrix-assisted laser desorption ionization mass spectrometry (MALDI-MS). In imaging MALDI-MS, spatial information and protein expression can be created. However, data from imaging MALDI-MS spectra require considerable signal processing to generate quantitative results and to provide input to later classification algorithms. To compare MALDI-MS spectra at different spatial locations (sample-to-sample comparisons) or classify parts of the spectra, a processing step called baseline correction is essential. This paper reports a robust algorithm for computing the baseline correction of MALDI-MS spectra. The algorithm requires few user inputs and is suitable for automatically processing a large number of spectra, as is the case when generating images. The results of our algorithm are validated on a dataset of spectra available for comparison purposes.

Categories and Subject Descriptors

I.4 [**Image Processing and Computer Vision]**

Keywords

Scientific visualization, baseline correction, mass spectrometry, MALDI

1. INTRODUCTION AND BACKGROUND

This paper reports an algorithm for the processing of Matrix Assisted Laser Desorption Ionization Mass Spectrometry (MALDI-MS) spectra. MALDI-MS allows direct measurement of the proteins within a biological tissue. Imaging MALDI-MS is a technique used to visualize the distribution of proteins within a tissue sample, and holds the promise, for example, of being able to differentiate tumors from normal tissue through the expression of proteins within these tissues [1].

MALDI-MS works, in brief, as follows. A chemical ``matrix'' is added to a biological tissues and allowed to crystallize. Caught within the crystalline structure of the matrix are large mass biomolecules, e.g., proteins and peptides. The sample is then bombarded with laser pulses, causing the matrix to vaporize and the proteins and peptides within the tissue to ionize. The ions, now in gaseous phase, are accelerated in an electric field and their mass calculated by a detector based on time of flight. The resulting time of flight calculations produce a spectrum of mass distribution of ions within the sample.

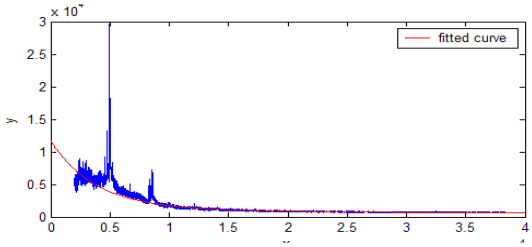

Figure 1. Example of MALDI MS mass spectrum before baseline correction from a sample of mouse brain. The red line is the baseline calculated by our algorithm.

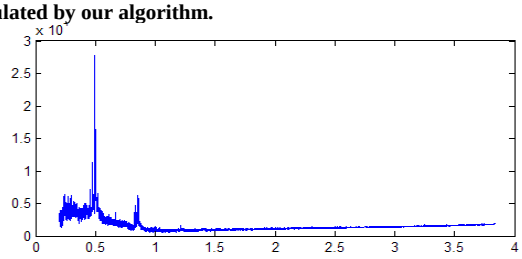

Figure 2. Example of MALDI MS mass spectrum after baseline correction using the baseline correction routine in the Data Explorer software. The overall intensity has decreased in low mass to charge ranges, but the spectrum seems to rise in the high mass to charge ranges.

Spectra resulting from the MALDI MS ionization process are noisy, as seen in the unprocessed mass spectrum in Figure 1. Large variations in intensity, even within the same tissue sample, make the quantification of the spectra difficult. This variance can be attributed to a number of factors, such as differences in application of the matrix and limitations in the detector [2][3]. One of the artifacts affecting the spectra is the baseline, which affects both the peak detection algorithms and sample-to-sample comparisons. The baseline is a mass-to-charge dependent offset on which the information-bearing component of the spectra is superimposed. Figure 1, which illustrates a typical MALDI MS spectrum, shows that the baseline appears as an exponential that decays with mass-to-charge value, also denoted m/z. The baseline hampers quantitative comparison of spectra and is an impediment to generating 2D and 3D MALDI images [1][4].

Once spectra are corrected, we can begin to work on classification and segmentation algorithms that allow differentiation of cancerous and non-cancerous material. We can employ algorithms from machine learning, image processing, and scientific visualization to allow automatic results that operate quantitatively to perform this differentiation. Correcting the baseline is

therefore an important pre-processing component in the analysis of spectra. Several algorithms have been proposed for baseline correction, and some of them are provided by the manufacturers of MALDI MS instruments. However, in our experience, they suffer from a number of weaknesses. For example, Figure 2 shows the spectrum of Figure 1 after baseline removal with Data Explorer [5]. After correction, a residual baseline can still be observed in the low mass to charge ranges as well as an upward trend in the high mass to charge range.

Recently, Liu et al. have proposed an approach in which they first smooth the spectrum using a moving average to eliminate noise [6]. They then define the baseline as the convex hull of the smoothed spectrum. In contrast, our algorithm computes the baseline from the raw spectrum. Depending on the filter parameters and the spectrum, this technique may produce a baseline that is too low. Wagner et al. characterize the variation of the MALDI spectrum as near-exponential decay [7]. They then iteratively apply a local linear regression technique and vary smoothness based on the mass to charge ratio. This technique has the disadvantage that the smoothness parameters must be set specifically for each spectrum. It may also result in a baseline that subtracts an unacceptable amount from broad peaks in the spectrum.

2. EXPERIMENTAL PROCEDURE

In this section we present the baseline correction algorithm. First, we briefly show how the algorithm works on an example spectrum in Section 2.1. Then we discuss the specifics of the algorithm in two parts. Section 2.2 presents an algorithm that finds where useful quantitative data in the spectrum begins, which is the starting point for our baseline correction algorithm. Section 2.3 presents the main part of our baseline correction algorithm. The idea is to divide the spectrum into local areas, and then divide the local areas into windows. We then find the minima in these windows. The size of the local area adaptively changes to insure that minima lying on peaks are not selected. The baseline is then defined as a function approximating the set of minima. Since the baseline varies throughout the spectrum, a window-based algorithm allows the creation of a baseline locally fitted to the spectrum. Figures 3 and 4 show two separate MALDI MS images of a male C57Bl/6J mouse brain created by averaging over a mass to charge ranges 7050.24 to 7098.09, and 13875.7 to 14516, respectively. The pixel colors are assigned linearly based upon ion count, using a blue-cyan-yellow-red color scale. Dark blue represents the lowest intensity and red represents highest. Within the figures, the image on the left is computed using MALDI data that has not been baseline corrected, and the image on the right was constructed after baseline correction.

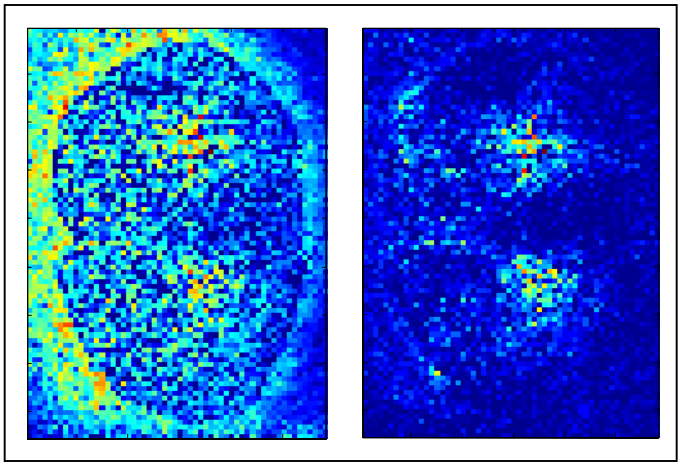

Figure 3. MALDI MS image of a male mouse brain created by averaging over a mass to charge range in each of the spectra in the dataset. The pixel colors are assigned linearly based upon ion count, using a blue-cyan-yellow-red color scale. Dark blue represents the lowest intensity and red represents highest. The image on the left is computed using MALDI data that has not been baseline corrected, and the image on the right was constructed after baseline correction

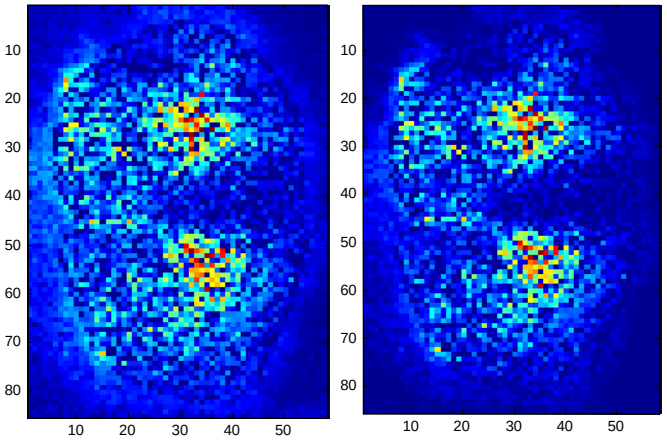

Figure 4. MALDI MS image of a mouse brain created by averaging over a different mass to charge in each of the spectra in the dataset. This range includes a mass of a singly charged myelin basic protein. The pixel colors are assigned linearly based upon ion count, using a blue-cyan-yellow-red color scale. Dark blue represents the lowest intensity and red represents highest. The image on the left is computed using MALDI data that has not been baseline corrected, and the image on the right was constructed after baseline correction.

2.1 Spectrum Example

In this section we discuss how the baseline program works for a section of a spectrum from our dataset. As mentioned previously, the baseline program operates on the spectrum locally by dividing a local area into windows and fitting a polynomial through local median points. The local area adaptively expands if it contains medians lying on peaks, i.e., too many medians above the fitted polynomial. We quantify this statement in step 5 of Section 2.3. For this example, we started with a local area of 1000 at a mass to charge ratio of 20,200. Figure 5 shows the initial local area of the algorithm highlighted in red. The local area is divided into 20 windows and the medians in each of these windows are chosen as candidate baseline points as shown in Figure 6. After selecting these local medians, a polynomial is fitted through the points. The polynomial resulting from the first iteration of this program is shown in Figure 7 and Figure 8. The polynomial is too high, which is a result of too many local medians lying above the polynomial. To remedy this situation, the local area is increased until a percentage of the local medians are below the polynomial or lie with in a distance epsilon from the polynomial. Figure 9 shows an intermediate result of the algorithm. The local area no longer expands when enough medians are found that are less than distance epsilon above the polynomial. Medians that lie above distance epsilon from the polynomial are discarded. The remaining median points are added to a list of baseline points. The calculated baseline is shown in Figure 10. The algorithm then proceeds through the rest of the spectrum in a similar manner.

2.2 Finding the Offset

Since the low mass to charge ranges of the spectra are populated with chemical noise, the area is considered noise. Typically a spectrum will begin with an ion count of zero, then increase in intensity before beginning to decrease again. Useful quantitative data begins roughly in the middle of the initial rise and fall of the spectrum, or the first area of highest intensity. We present an algorithm that finds this mass to charge value, a marker that indicates where we start dividing our spectrum into local areas and extracting the minima used to find our baseline function. The only input into this algorithm is the window width, *w*. The starting point is found using the following algorithm:

1. Define a window size of fixed width, *w*. This *w* m/z wide window will be stepped across our spectrum. In our results we have found that a window value of 1000 mass to charge values works well, since it is large enough to average out noise in the signal.
2. Initialize the location of the first mass to charge value in the window, *w_loc* spectrum to the beginning of spectrum.
3. Set current intensity, *curr_int* equal to the average intensity of mass to charge values in this beginning window.
4. Move the window *w* mass to charge values, *w_loc=w_loc+w*.
5. Set previous intensity, *prev_int*, equal to the current intensity, *curr_int*. Set current intensity, *curr_int*, equal to the average intensity of the window.
6. Repeat steps 4 and 5 until *prev_int* > *curr_int*.

Now we have found an area where the window of highest mass to charge value exists. We next move the window one mass to charge value at a time to find the window with the highest average.

7. Define the area to search for the window of highest mass to charge intensity, *startsearch=w_loc*-2**w*, and *endsearch=w_loc*.
8. Create a new window of width *w*, and start it at the location *startsearch*.
9. Move the window one mass to charge value and continually calculate the average until window reaches *endsearch*. Update the location of the window of highest average.
10. Calculate the median of the window of highest average and use it as the starting point for our baseline algorithm.

2.3 Calculating the baseline

A baseline should not remove peak information from the spectrum. Suppose that a spectrum is divided up into windows of fixed width, and one point is selected from each window using a constraint such as the median. The baseline is defined as the function that best approximates this set of points. Problems arise when the windows that divide the spectrum lie on peaks. The resulting baseline will be too high in areas where points chosen to calculate the baseline are located on peak information. To overcome this problem, we employ an adaptively expanding window. The window expands according to mass to charge ranges. Therefore, the number of data points in a window can vary since the number of data points as a function of the mass to charge values is not linear. The inputs to the algorithm are the local area, *init_loc_area*, and number of windows, *num_wind*. The

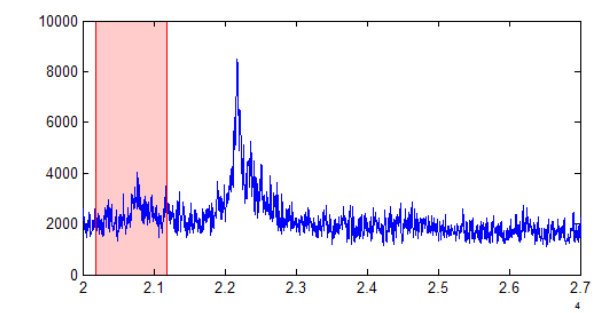

Figure 5. A sample area of spectrum taken from our sample dataset. This figure shows the initial local area chosen by the algorithm.

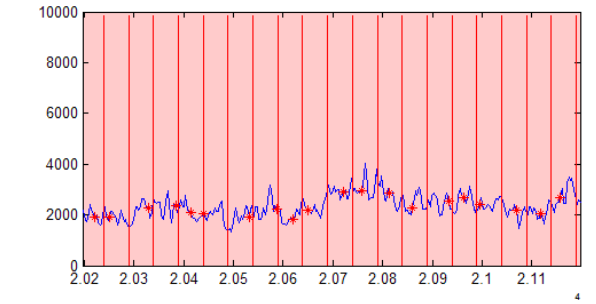

Figure 6. View of the red highlighted local area of Figure 5. The vertical lines represent the division of the local area into windows. In each window a median is selected and shown with a red asterisk.

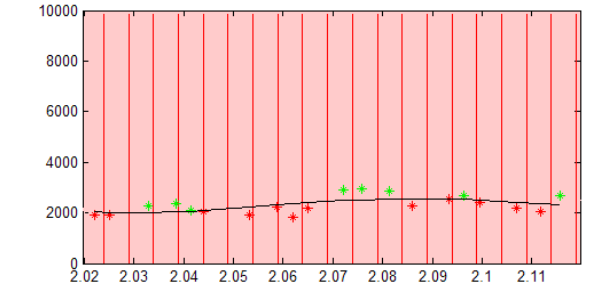

Figure 7. After picking the medians within the windows, a polynomial, shown in black, is fitted through the data. Medians greater than epsilon above the polynomial are shown in green, the rest are shown in red.

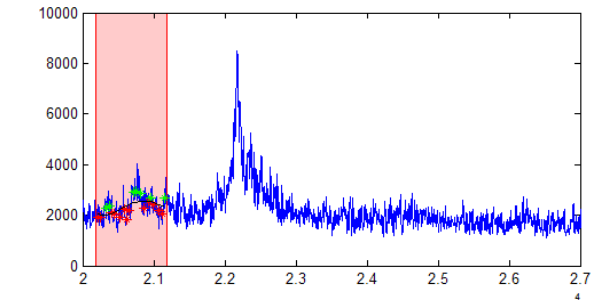

Figure 8. This figure shows the polynomial in black fitted through the medians of the first iteration (same as figure 5). This figure shows what a small portion of the spectrum after the local area looks like.

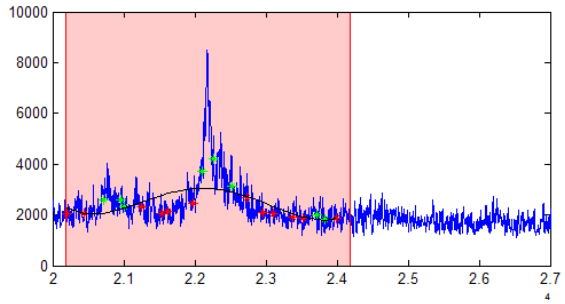

Figure 9. This is an intermediate result of the algorithm. The local area has expanded, but there are still too many median points more than epsilon above the polynomial. Median points epsilon above the polynomial are shown in green, the rest are shown in red.

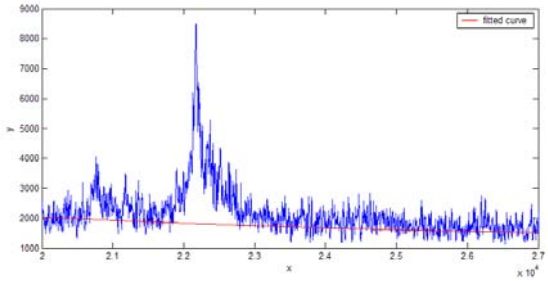

Figure 10. This figure shows the resulting baseline in red. The baseline is calculated from the "kept" medians.

algorithm is as follows:

1. Set the local area, *loc_area*, equal to initial local area size, *init_loc_size*. The local area is a range that is as small as possible under the criteria of Step 5 below, but which does not lie with in the spread of a single peak. In our experiments, *init_loc_size* was set to 1000. In our experience, a local size of 1000 works well and represents a reasonable tradeoff between precision and speed.
2. Define local window size, *loc_wind*, equal to *loc_area* divided by the number of windows, *num_wind*. This allows us to divide the local area into *num_wind* windows. In our results, we set the number of windows, *num_wind*, equal to 20.
3. Divide the local area into *num_wind* windows of width *loc_wind*. Extract the mass to charge value, *x*, and the intensity, *y*, of the medians in the window, (x_i, y_i), where *i=1:num_wind*.
4. Fit a fourth order polynomial, *P*, to the medians using a least squares technique to solve a system of linear equations at the points (x_i, y_i). A fourth order polynomial is a compromise between computational speed and smoothness.
5. Let $y'_i = P(x_i)$, i.e., the values of the polynomial *P* at x_i. Count the number of points at which $y_i - y'_i > \varepsilon$, i.e., count the number of medians computed in Step 3 that are above the polynomial by a pre-computed tolerance ε. The value of ε is not of great bearing; we used 5% away from the fitted polynomial.
    a. If the counted number of points exceeds 25% of the total number of points in that local area (*num_wind*), then discard all of the median points found in the local area. Increase the size of the local area by *init_loc_size* and start again at Step 2. The local area will expand until less than 25% of the points lie distance ε above the locally fitted polynomial.
    b. Otherwise, add the median points less than distance ε above the polynomial to the set of baseline points, and continue to Step 6.

    25% is used because it represents a value that ignores points lying on peaks without compromising speed. The algorithm is reasonably robust to the exact value of this parameter.
6. Reset the *local_area* to *init_loc_size*. Move to the next local area. Repeat steps 2 through 5 until the end of the spectrum is reached.
7. Calculate the baseline of our spectrum by fitting an exponential of the form $y = ae^{bx} + ce^{dx}$ to the set of data points using a least squares technique. This function is smooth and allows enough flexibility to fit through the data points.

If the end of the spectrum is reached the local area can still be expanded. The only median points that a polynomial will be fit

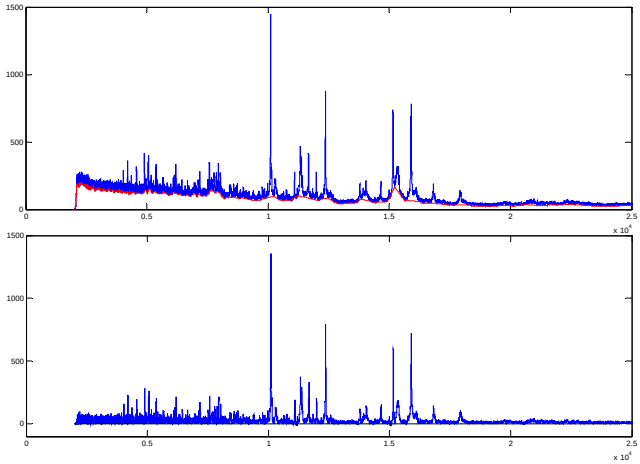

Figure 11. The top spectrum shows mass spectrum with the baseline calculated by linear interpolation seen in red. We see at the mass to charge value of around 15,000, the baseline rises up into the peak. By having a baseline at this point, important peak information is lost after baseline subtraction as seen in the second spectrum.

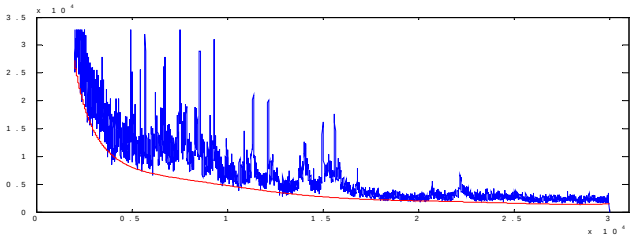

Figure 12. Example of spectrum shown in blue and the resulting baseline shown in red found by fitting polynomial through the minimum of a window.

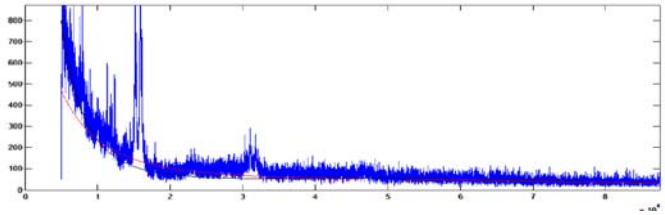

Figure 13. The baseline resulting from fitting a polynomial through the medians is shown in red. The black line is where our algorithm puts the baseline. By comparison, our method preserves more peak information.

through are those lying on the spectrum, i.e., less points are used in the polynomial fitting calculation because not all of the local windows lie within the scope of the spectrum. However, this is generally not a problem as there are few peaks lying in the high mass to charge ranges.

3. COMPARISON TO OTHER METHODS

There are simpler methods for baseline correction found in practice, but not described in the literature. We describe three of them here, with their shortcomings. One method to determine a baseline is to linearly interpolate the local minima found in "windows" of a varying width in the spectrum. The lower m/z values have smaller windows, and the larger m/z values have larger windows. The varying window accounts for the greater variance found at the low m/z values. The problem with this algorithm is that it produces a baseline that is not monotonic and smooth. Moreover, subtracting the baseline found by this algorithm can sometimes cause the subtraction of important peak information as shown in Figure 11.

Another method of baseline subtraction is to use the minima in window, fit a polynomial to the data instead of using linear interpolation. Figure 12 shows the results of this algorithm. It produces a more desirable baseline because it is smooth, but fails to account for noise found in the spectra. We desire a baseline that is smooth and discards some of the noise found in the spectrum.

To correct for this noise, the medians can be used instead of using the minimum. The results of this algorithm, as shown in Figure 13, show a baseline that is smooth and subtracts noise found in the spectrum. However, it sometimes subtracts important peak information from the spectrum. For instance, if a window is directly on a peak, then the median point in the window will be on a peak.

4. RESULTS AND DISCUSSION

There is no published metric for establishing the correctness of baseline correction algorithm. To validate our algorithm, we prepared a canonical data set of twenty mass spectra (labeled A through T) representing spectra typically obtained from a MALDI MS device. Both the offset algorithm of Section 2.2 and the baseline algorithm succeeded robustly on all datasets.

Figures 14-19 show three spectra from our dataset before and after baseline correction. Notice the intensity of the spectra in the figures. Sample C in Figure 14, Sample E in Figure 16, and Sample G in Figure 18 have the respective maximum intensities of 1500, 3250, and 600. Our baseline correction algorithm works well for all of these spectra, using the default parameters described previously.

The calculated baseline in Figure 14 (sample spectrum C), shown in red, demonstrates how the algorithm preserves peak information, while subtracting some of the noise. Note the baseline at the mass to charge values of approximately 10,000, 27,000, and 42,000. Peak information is preserved in these areas. Figure 15 shows spectrum C after baseline correction.

Sample E in Figure 16 has a quickly decreasing baseline with a sharp turn in the low mass to charge range. The algorithm is able to find this baseline while preserving all of the important peak data.

Sample G in Figure 18 shows a spectrum that is quite noisy. The algorithm is robust to this amount of noise and maintains the peak information found around the mass to charge ratio of about 15,000. The spectrum after baseline correction is shown in Figure 19.

The results for the full set of twenty spectra can be found at http://people.vanderbilt.edu/~betsy.williams/MALDI. Also available is the MATLAB code implementing this algorithm. The baseline is typically computed in under 2 seconds using a spectra contains 70,000 points. If faster performance is desired, the program be easily be coded in C.

5. CONCLUSION

We have presented a general purpose baseline correction algorithm. Our automatic algorithm is robust and only requires a few user inputs. The algorithm works well if the user inputs are defined as the set of parameters discussed in this paper.

The issue of how to judge the best algorithm is still open. Visual inspection is commonly used. In our work we calculated the baseline for spectra from a variety of tissue, and found that our algorithm worked well for all of the spectra.

Our future work includes a better method of characterizing the noise underlying the MALDI MS process. We would also like to improve the validation of the technique by calculating the spectrum on a wider variety of examples.

6. REFERENCES

[1] Stoeckli M, Chaurand P, Hallahan DE, Caprioli RM. Imaging mass spectrometry: A new technology for the analysis of protein expressions in mammalian tissues. *Nat Med.* 2001; 7: 493-496.

[2] Bucknall M, Fung KYC, Duncan MW. Practical quantitative biomedical applications of MALDI-TOF mass spectrometry. *J Am Soc Mass Spectrom*. 2002; 13: 1015-1027.

[3] Krutchinsky AN, Chait BT. On the nature of chemical noise in MALDI mass spectra. *J Am Soc Mass Spectrom.* 2002; 13: 129-134.

[4] Crecelius A, Cornett DS, Williams B, Bodenheimer B, Dawant B, Caprioli RM. Developing 3-D Imaging Mass Spectrometry. Proc of the 51st ASMS Conf on Mass Spectrom and Allied Topics. June 2003.

[5] Data Explorer, Version 4.4, Applied Biosystems.

[6] Liu Q, Krishnapuram B, Pratapa P, Liao X, Hartemink A, Carin L. Identification of differentially expressed proteins using MALDI-TOF mass spectra. *Asilomar Conf on Signals, Systems and Computers*. November 2003.

[7] Wagner M, Naik D, Pothen A. Protocols for disease classification from mass spectrometry data. *Proteomics.* 2003; 3: 1692-1698.

[8] Horn DM, Zubarev RA, McLafferty FW. Automated reduction and interpretation of high resolution electrospray mass spectra of large molecules. *J Am Soc Mass Spectrom*. 2000; 11: 320-332.

[9] Jarman, KH, Daly DS, Anderson KK, Wahl KL. A new approach to automated peak detection. *Chemometrics and Intell Lab Sys.* 2003; 69: 61-76.

[10] Yanagisawa K, Shyr Y, Xu BJ, Massion, PP, Larsen PH, White BC, Roberts JR, Edgerton M, Gonzalez A, Nadaf S, Moore JH, Caprioli RM, Carbone DP. Proteomic patterns of tumour subsets in non-small-cell lung cancer. *Lancet*. 2003; 362: 433-39.

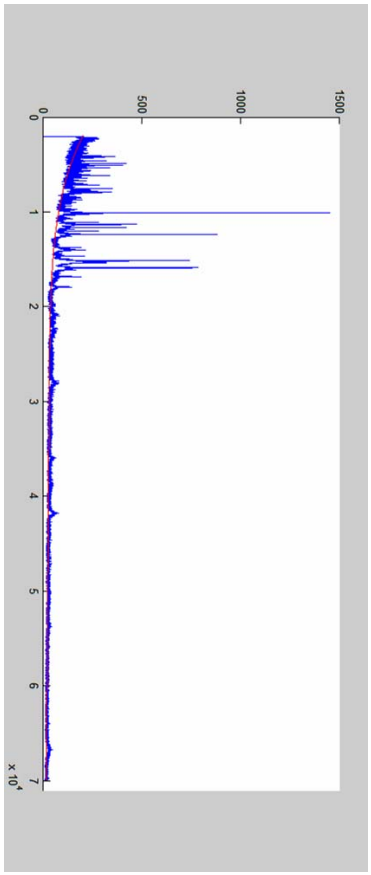

Figure 14. Sample C before baseline correction.

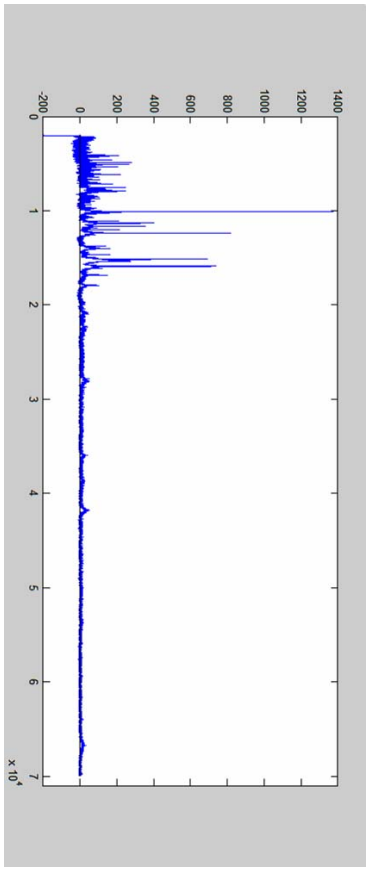

Figure 15. Sample C after baseline correction.

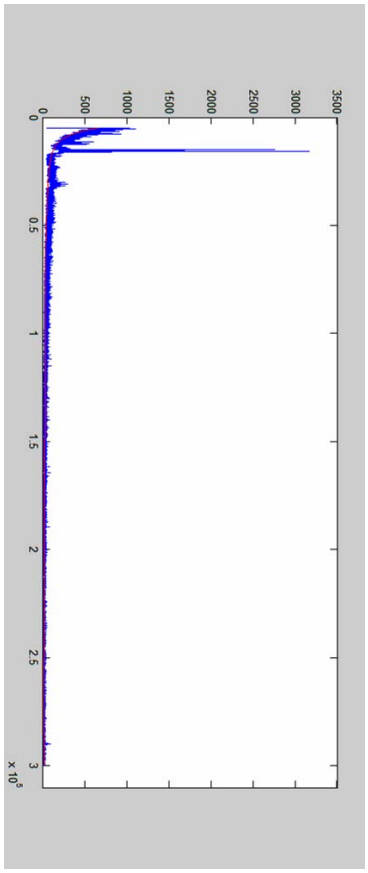

Figure 16. Sample E before baseline correction.

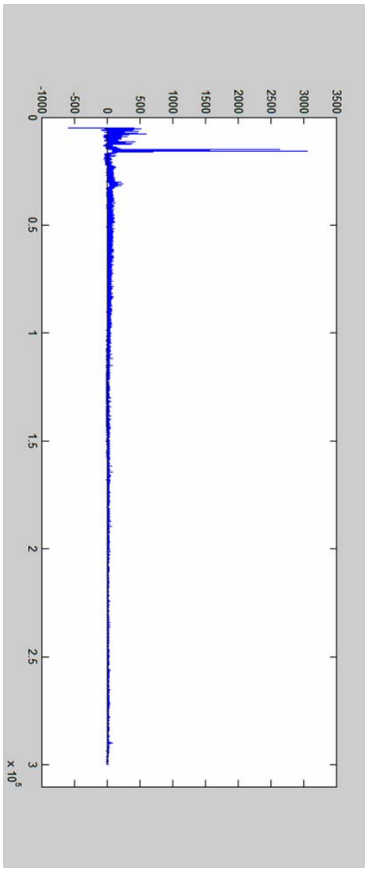

Figure 17. Sample E after baseline correction.

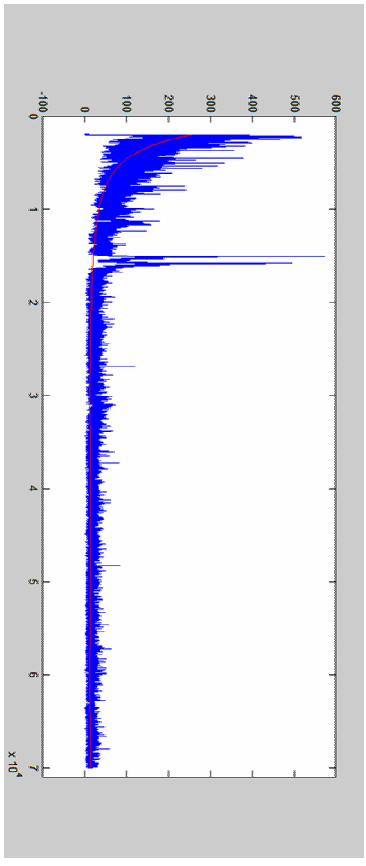

Figure 18. Sample G before baseline correction.

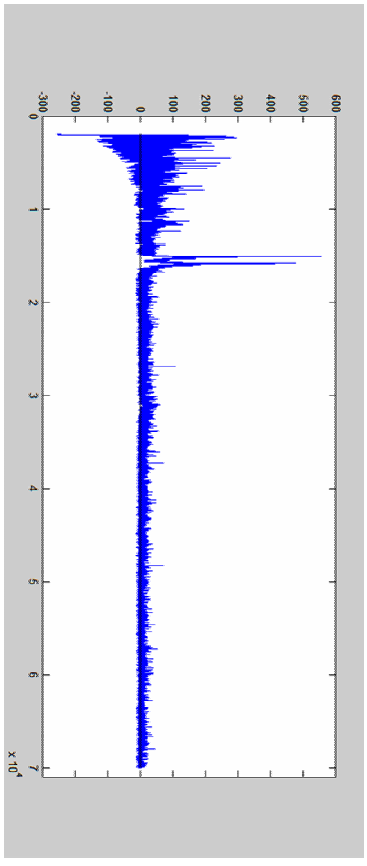

Figure 19. Sample G after baseline correction.