IPSJ Transactions on Bioinformatics Vol.7 24–29 (Oct. 2014)

[DOI: 10.2197/ipsjtbio.7.24]

Database/Software Paper

Signal Processing Algorithm Development for Mass++ (Ver. 2): Platform Software for Mass Spectrometry

Shin-ichi Utsunomiya[1,a)] Yuichiro Fujita[1] Satoshi Tanaka[1] Shigeki Kajihara[1]
Ken Aoshima[2] Yoshiya Oda[2] Koichi Tanaka[1]

Received: March 17, 2014, Accepted: June 26, 2014, Released: October 22, 2014

Abstract: Mass++ is free platform software for mass spectrometry, mainly developed for biological science, with which users can construct their own functions or workflows for use as plug-ins. In this paper, we present an algorithm development example using Mass++ that performs a new baseline subtraction method. A signal processing technique previously developed to correct the atmospheric substances in infrared spectroscopy was converted to adjust to the mass spectrum baseline estimation, and a new method called Bottom Line Tracing (BLT) was constructed. BLT can estimate a suitable baseline for a mass spectrum with rapid changes in its waveform with easy parameter tuning. We confirm that it is beneficial to utilize techniques or knowledge acquired in another field to obtain a better solution for a problem, and that the practical barriers to algorithm development and distribution will be considerably reduced by platform software like Mass++.

Keywords: mass spectrometry, platform, baseline subtraction, algorithm, signal processing

1. Introduction

Mass spectrometry is widely used in biological science for proteomics, metabolomics, transcriptomics, and genomics, including epigenetics. Various mass spectrometers are supplied by multiple instrument vendors, and each of them has its own commercial software. However, this software cannot always satisfy the users' demands, especially in biological science, as they often need special functions or workflows of their own. But, it is impossible for users to modify commercial software and it is also difficult for instrument vendors to modify their software to accommodate uncommon needs.

At the same time, there is free software being developed and supplied by academic projects [1], [2], [3] that can load data files with an open format for mass spectrometry (e.g. mzData [4], mzXML [5], or mzML [6]) instead of or in addition to the instrument's own data file format, and that can sometimes supply functions or workflows lacking in the commercial software. Since much of this is open-source software, users may create original functions or workflows of their own. In practice, however, it seems to be difficult for users to add or modify functions of the software by themselves. Therefore user reports about original function implementations using open-source software as a platform have rarely been found.

[1] Koichi Tanaka Laboratory of Advanced Science and Technology, Shimadzu Corp., Kyoto 604–8511, Japan
[2] Biomarkers and Personalized Medicine CFU, Eisai Product Creation Systems, Eisai Co. Ltd., Tsukuba, Ibaraki 300–2635, Japan
[a)] utu@shimadzu.co.jp

Mass++ is free platform software for mass spectrometry that can load open-format data files and data files from some instrument vendors. In addition, it has an architecture that enables users to add their own functions. Mass++ was developed in 2005 by a pharmaceutical company (Eisai) that uses a mass spectrometer, supported by the former Japanese public foundation project "CREST" [7]. In 2010, it was transferred to the current Japanese public fund project called the "FIRST" program and has been released as "Mass++ (Ver. 2)" [8]. Also, a mass spectrometer vendor (Shimadzu) has joined the development team together with the former developer. More than half of our project members were new users of Mass++.

In this paper, we will introduce an example of signal processing algorithm development on Mass++ from a new user's point of view. We will also present a new baseline subtraction method which is implemented on Mass++ as a plug-in. The method was originally developed to correct atmospheric constituents in infrared spectroscopy and then converted into a new baseline subtraction method for mass spectrometry.

2. Methods: Platform for Development

2.1 Mass++ Overview

Mass++ has data read, display, and analysis functions used to analyze data from mass spectrometers. It is utilized as a platform to develop signal processing algorithms (**Fig. 1**).

Figure 2 presents an example of Mass++ main window, which has several child windows displaying mass spectrometer data in several ways. Developers can focus on developing algorithms since the necessary data read and display functions are already

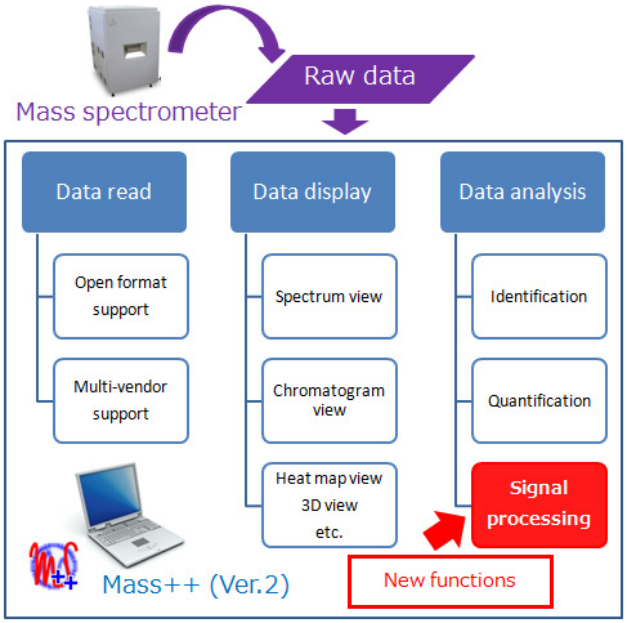

Fig. 1 Overview of Mass++ functions.

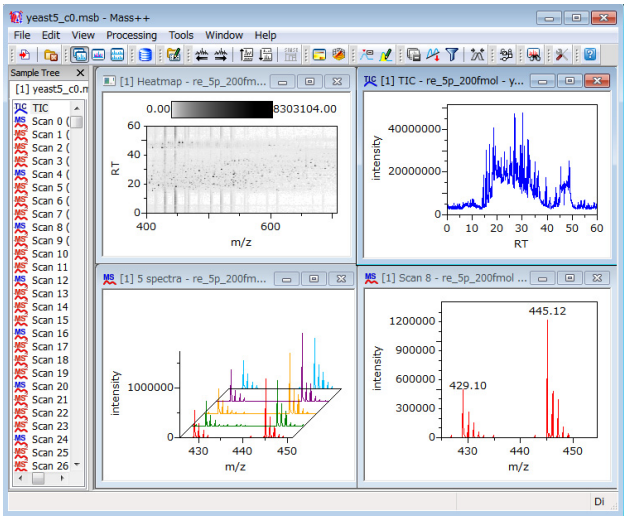

Fig. 2 Mass++ main window overview. Mass++ has a multiple document interface. The examples given are a heat map view for 2-dimensional data (upper left), a chromatogram view (upper right), a multiple mass spectra view (lower left), and a spectrum view with peak detection results (lower right).

provided in Mass++.

Mass++ runs on Microsoft® WindowsXP, 7 or later version. At this time, Mass++ is not open-source software but has an extensible plug-in architecture and an accessible license that encourage users to develop their own functions or workflows. The Mass++ Standard Development Kit (SDK) is supplied to help developers. It contains documents, sample source codes, and tools for developments including the Mass++ plug-in wizard that works with Microsoft® Visual Studio. According to the Mass++ (Ver. 2) license [9], users can develop and use their own plug-ins and have the right to sell, distribute, lend, or transfer them. C++ and C#.Net are supported for programming languages.

2.2 Mass++ Plug-in Construction

Figure 3 presents the Mass++ plug-in development workflow.

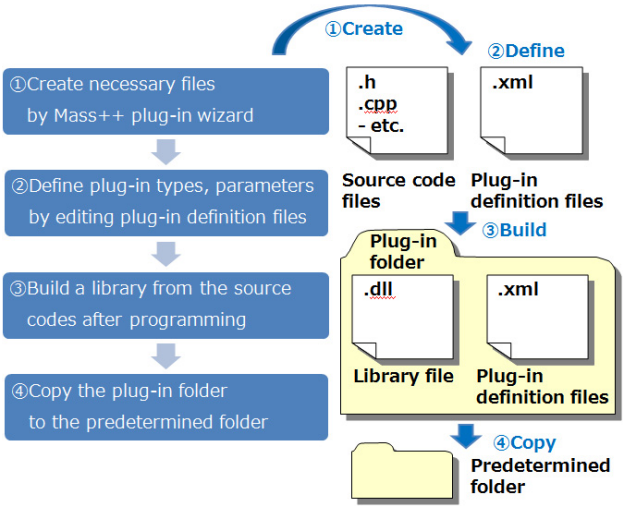

Fig. 3 Mass++ plug-in development workflow.

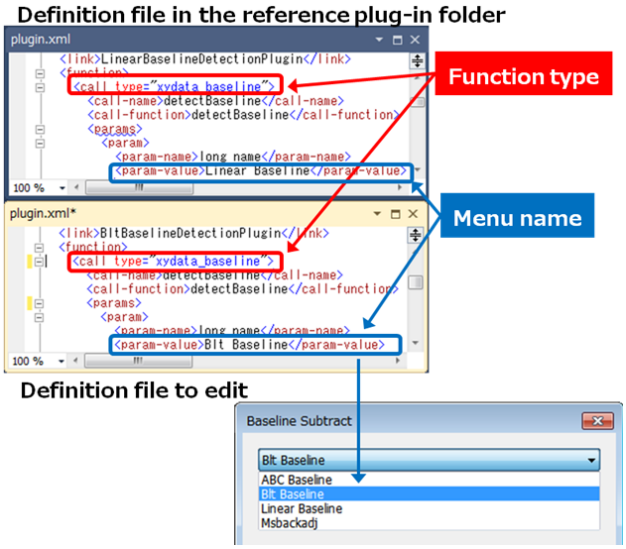

Fig. 4 Example of a Mass++ plug-in definition. Two definition files are depicted: one (the lower) is the file to edit in the new plug-in, and the other (the upper) is a reference file in the existing plug-in to examine or to copy from. Both plug-ins have the same function type "xydata-baseline" and are activated in the same "Baseline Subtract" dialog by selecting each plug-in menu entry.

At the first step in Fig. 3, the Mass++ plug-in wizard automatically sets up a project folder and prepares the necessary files and settings. In the next two steps, the newly created plug-in folder, which has a library file and plug-in definition files, is configured by the user. Finally, in the fourth step, after the new plug-in folder is stored in a predetermined Mass++ folder, the new function becomes available in Mass++.

In general, not negligible start-up times are required for software developers to become familiar with a platform or a framework, since they have to learn certain programming rules or functions unique to it. The situation was once the same for Mass++. Recently, however, the start-up time has been greatly reduced due to its plug-in wizard, as seen in Fig. 3. The characteristic and a little bit complicated work on the workflow is the plug-in definition at the second process while other works are quite simple or general in software developments. The developer should use the existing Mass++ plug-in definition file as a reference. An ex-

ample is given in **Fig. 4**. A number of items are described in the definition file "plugin.xml" in XML format. The important items are highlighted in the figure as follows. The plug-in function type defined by a "<call type = ...>" tag specifies the plug-in classification in Mass++, and the developed plug-in will be loaded along with any other plug-ins that have the same function type. In this example, the plug-in type is specified as "xydata_baseline," and the menu entry used to call the function appears among the "Baseline Subtract" dialog options. The menu string is defined by a "<param-value>" tag.

According to the plug-in architecture, already implemented Mass++ functions or workflows such as "Baseline subtraction" or "Quantitation" can be used as parent processes for a newly developed plug-in without editing their source codes. In this manner, a new signal processing method can be added and utilized on Mass++ by implementing each plug-in only.

3. Experiments: Signal Processing

3.1 Conventional Signal Processing in Mass Spectrometry

In general, the mass spectrum raw data acquired by an instrument is processed with a signal processing sequence as illustrated in **Fig. 5**. First, random or impulse noise in the raw data is reduced. Second, the baseline is subtracted and processed data is obtained as a result. Finally, peaks are detected in the processed data. After the signal processing, the peak list, which is composed of the detected peaks that indicate the mass values and quantities of the chemical constituents of the analyzed sample, will be further analyzed (e.g., identification, quantification, or other advanced analyses).

It is apparent that the signal processing plays a considerable role because its results influence the following analyses. Therefore, several methods have been investigated and established for each process in Fig. 5. A comprehensive work surveying these has been reported [10] .

3.2 Baseline Subtraction in Mass Spectrometry

In general, it is easy to estimate the baseline for a spectrum like the example in Fig. 5, whose waveform characteristics vary only gradually. However, it is difficult to estimate suitable baseline for a spectrum with a rapidly changing waveform, and the performance depends significantly on the algorithm used.

Figure 6 presents a noise-reduced spectrum for raw data acquired by MALDI-ISD [11] measurement of a synthesized nucleic acid sample. As seen in the figure, the target spectrum has a steep slope and a number of ion peaks. In order to estimate the baseline, we first tried linear interpolation [10], a well-known conventional method, but this failed as the estimated baseline waveform was discontinuous. We then tried another algorithm called "msbackadj" that estimates the baseline value for each small segment, composes the baseline curve interpolating those values, and adjusts the baseline considering the peak signals. These two algorithms take a same strategy to deal with the variation of the signal characteristics however the later has more parameters and function types.

The "msbackadj" function is supplied in MATLAB [12], which is popular commercial software for mathematical analyses and

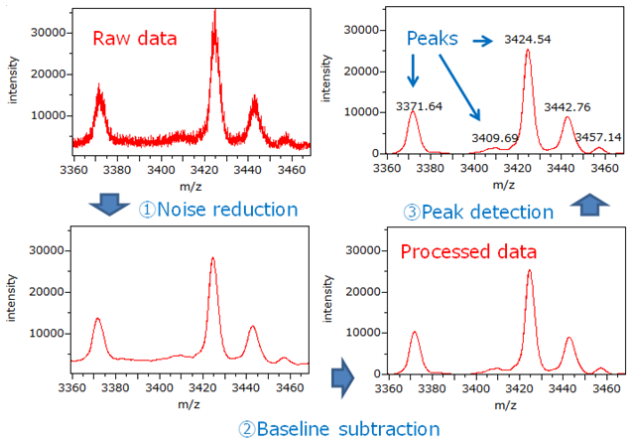

Fig. 5 Signal processing for a mass spectrum. The horizontal axis [m/z] of each mass spectrum indicates atomic mass unit divided by the charge number of the ions, while the vertical axis indicates signal intensity of the ion current.

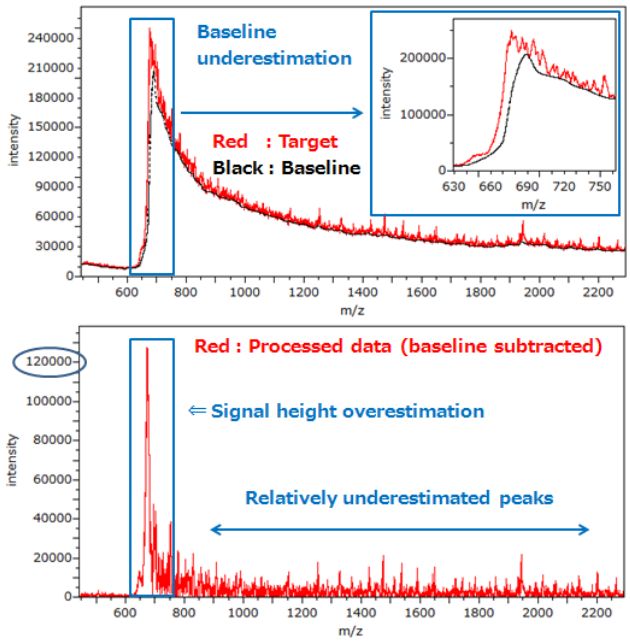

Fig. 6 Baseline subtraction by msbackadj. Upper: Target spectrum (noise reduced by Gaussian filter) and the baseline estimated by msbackadj. Lower: Processed data with baseline subtracted. The signal height was overestimated near the steep slope and the peaks out of that range were relatively underestimated.

some of their library functions (including msbackadj) are freely-available from other software. We implemented a Mass++ plug-in to use msbackadj in the MATLAB library and carefully tuned the parameters to adjust for the target in Fig. 6.

Finally we had obtained the result seen in Fig. 6, but the baseline was underestimated and consequently the processed data and the peak heights were overestimated at the region near the steep slope. They would absolutely degrade the quantification accuracy and also might bring out the degradation of the peak identification because of the relative underestimation of the peak heights out of that region. Moreover, the parameter tuning was somewhat difficult for "msbackadj." Not negligible time was required to understand and tune the parameters. Three parameters were tuned while three calculation types (each of which has two to five options) were untouched and remained set to their default values.

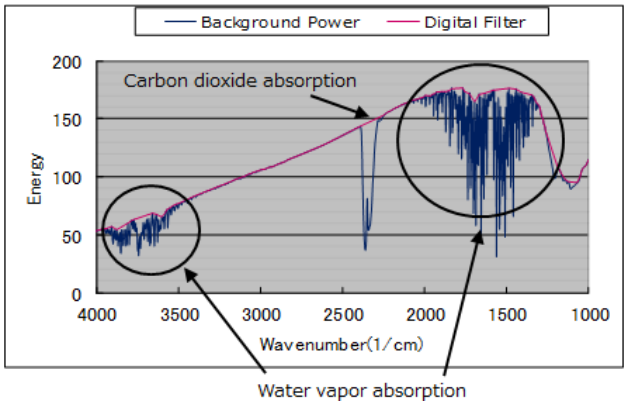

Fig. 7 Envelope estimation by the digital filter in infrared spectroscopy. A background power spectrum measured with an empty sample cell has numerous absorption troughs due to water vapor and carbon dioxide constituents in the atmosphere. An envelope is estimated by the digital filter to calculate the absorbance of the atmospheric constituents.

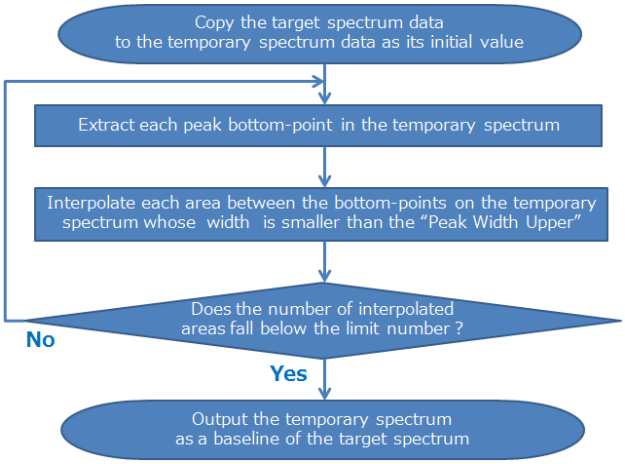

Fig. 8 BLT algorithm flowchart. "Peak Width Upper" is a parameter that users should tune to adjust for the target spectrum. The limit number is a threshold number including zero that detects the saturation of the iterative estimation.

It seems that the strategy of an algorithm that divides the target data into small segments cannot adapt sufficiently to rapid changes in the waveform characteristics. The same results will be expected for other methods that have the same strategy.

3.3 BLT: A New Baseline-subtraction Method

In order to create an easy-to-operate, precise baseline subtraction method, we employed a technique that had been previously developed for infrared spectroscopy [13].

In infrared spectroscopy, absorptions of water vapor or carbon dioxide by the atmosphere sometimes overlap with the sample's absorptions, and the analytical accuracy is reduced. To solve this problem, we developed a correction method for precisely estimating the absorption of atmospheric constituents based on the background spectrum measured with an empty sample cell. The practical key point of this solution is a digital filter that estimates the envelope of the background spectrum, which has complicated, sharp, trough waveforms caused by the absorptions of atmospheric constituents. An example is given in **Fig. 7**.

We converted the digital filter algorithm into a mass spectrum baseline estimation that would smooth the peak waveforms of the ion currents in a mass spectrum instead of the trough waveforms of the absorptions in an infrared spectrum. **Figure 8** is a flowchart of the algorithm called Bottom Line Tracing (BLT). As seen in the figure, the bottom-point extraction and interpolation steps are iterated until sufficient smoothing is achieved.

From a user's point of view, the data-processing method should minimize the variable parameters. In infrared spectroscopy, the water vapor and carbon dioxide absorbance characteristics of the background spectrum are qualitatively known, and therefore all processing parameters can be fixed to constant values and users are free from parameter tuning. In general mass spectrometry, however, since mass spectrum characteristics are not consistent, "Peak Width Upper" in Fig. 8 is retained as a variable parameter. BLT parameter tuning is quite easy compared with other methods because there are fewer parameters and their meaning is simple. "Peak Width Upper" is the only parameter to be tuned and should be set roughly to the largest possible width of the sample peaks

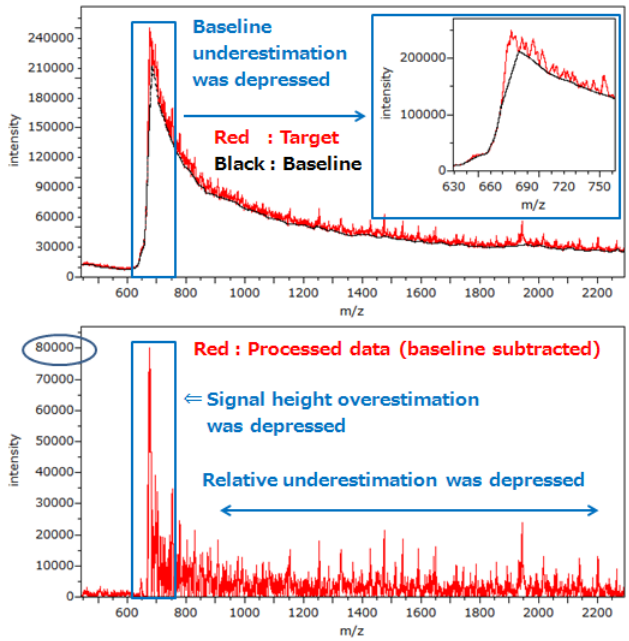

Fig. 9 Baseline subtraction by BLT. Upper: Target spectrum (noise reduced by Gaussian filter) and the baseline estimated by BLT (Peak Width Upper = 40Da). Lower: Processed data with baseline subtracted. The height overestimation and relative underestimation were depressed.

in the spectrum, based on visual observation.

Baseline subtraction by BLT in the previously mentioned example is depicted in **Fig. 9** where the baseline underestimation and the processed data overestimation were depressed and improved near the steep slope, compared to the msbackadj result in Fig. 6. Therefore higher identification or quantification accuracy can be expected with BLT. The parameter tuning was simple and easy because only one parameter, "Peak Width Upper," needs to be tuned and it had a simple meaning as described before.

Taking a logical point of view to the algorithm in Fig. 8, at the beginning of the estimation a roughly smoothed temporary spectrum is obtained and then is gradually transformed to have a more agreeable waveform through iterative estimations. As a result of this strategy, the algorithm is capable of achieving a suitable base-

line that precisely follows each slope over a wide range using BLT.

4. Discussion

4.1 Effect of Baseline Subtraction: BLT

The newly developed baseline subtraction method, BLT, has the following features as previously demonstrated.

- A suitable baseline can be estimated even when the target spectrum has rapidly changing waveform characteristics.
- Parameter tuning is simple and easy because the parameter meaning and the response are straightforward.

From an application point of view, baseline subtraction may improve the quantification accuracy and stability of mass spectrometry. In drug discovery and clinical diagnostics, mass spectrometry quantification plays an important role for biomarker research or validation. **Figure 10** presents an example of biomarker research and the baseline subtraction capability for improvement.

As seen in the figure, the small target peaks at the foot of a larger non-target peak often fluctuate and their heights become unstable. In order to quantify the target signals precisely, it is necessary to eliminate any non-target signal influence. Baseline subtraction may resolve this problem in some cases, adjusting for the signal characteristics as seen in Fig. 10. With BLT, users can easily get a desirable baseline as demonstrated.

4.2 Impact of the Platform Software: Mass++

The advantages of utilizing Mass++ as a platform to develop signal-processing algorithms are as follows.

- For expert engineers from other fields, the start-up time is significantly reduced by a platform that supplies the functions necessary for mass spectrometry data analysis.
- Multiple algorithms can be compared and evaluated for the same data on the same platform.
- Superior algorithms or methods can easily be distributed on a free platform.

The algorithms or methods developed by our project are bundled with Mass++ as its plug-ins however the source code is not open except for some sample code.

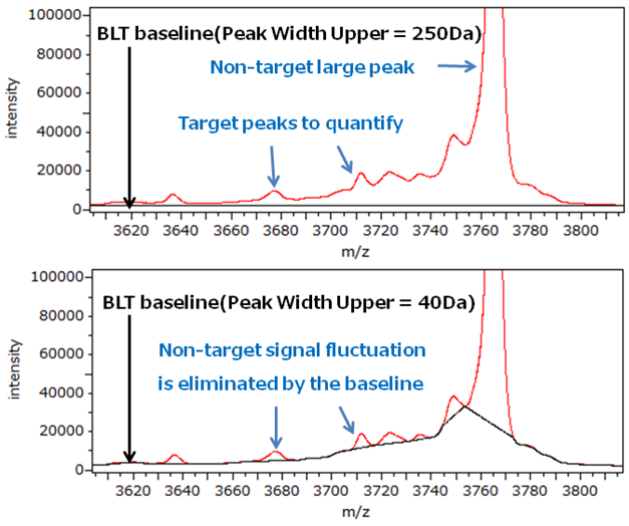

Fig. 10 Example of biomarker research and baseline subtraction. Upper: Peaks at the foot of a large peak often fluctuate, and their heights become unstable. Lower: Baseline subtraction may eliminate the non-target signal fluctuation in the target signal.

Currently, Mass++ is not open-sourced as mentioned before. One reason is for intellectual property protection; another is based on our project's policy that software should be maintained by the original developers as their own responsibility. The latter policy may change in the near future.

For an algorithm development like this, users do not need the platform source code. They can debug their original source code through Mass++ operations by using the Mass++ SDK without the platform source code. But for software development utilizing graphical user interfaces, it may be desirable for the platform source code to be opened for reference. Users can construct a new plug-in for a workflow that consists of multiple functions utilizing already implemented plug-ins as child processes by reading documents in the Mass++ SDK at this time, however, the platform source code will surely be a great help for them.

5. Conclusion

In this paper, we have reported two accomplishments:

- An example of algorithm development for Mass++, which is free platform software for mass spectrometry, was introduced. Efficient development was confirmed, and easy distribution of superior methods is expected through using this platform.
- A new baseline subtraction method for a mass spectrum called BLT, was developed. With BLT, it is easy to estimate a suitable baseline even for a spectrum which has rapidly changing characteristics. Consequently, improvements in quantification and identification accuracy can be expected.

We confirm that it is beneficial to utilize techniques or knowledge acquired in another field or for other devices to solve problems in bioinformatics or related instrumented data analysis. The practical barriers to software development will be reduced by platform software like Mass++ .

Mass++ (Ver. 2) can be downloaded from the FIRST ms3d project website (http://www.first-ms3d.jp/english/). The baseline subtraction method BLT will also be bundled.

Acknowledgments This work is granted by the Japan Society for the Promotion of Science (JSPS) through the "Funding Program for World-Leading Innovative R&D on Science and Technology (FIRST Program)," initiated by the Council for Science and Technology Policy (CSTP).

We particularly acknowledge the help of Mr. Koichi Kojima and Mr. Naoki Kaneko, who provided their experimental data used in these studies.

References

[1] Sturm, M. et al.: OpenMS — An open-source software framework for mass spectrometry, *BMC Bioinformatics*, Vol.9, No.163 (2008).
[2] Kessner, D. et al.: ProteoWizard: Open source software for rapid proteomics tools development, *BIOINFORMATICS Application Note*, Vol.24, No.21, pp.2534–2536 (2008).
[3] Deutsch, E.W. et al.: A guided tour of the Trans-Proteomic Pipeline, *Proteomics*, Vol.10, pp.1150–1159 (2010).
[4] Orchard, S. et al.: Autumn 2005 Workshop of the Human Proteome Organisation Proteomics Standards Initiative (HUPO-PSI) Geneva, September, 4-6, 2005, *Proteomics*, Vol.6, No.3, pp.738–741 (2006).
[5] Pedrioli, P.G.A. et al.: A common open representation of mass

spectrometry data and its application to proteomics research, *Nature Biotechnology*, Vol.22, pp.1459–1466 (2004).
[6] Turewicz, M. and Deutsch, E.W.: Spectra, Chromatograms, Metadata: mzML-The Standard Data Format for Mass Spectrometer Output, *Proteomics Methods in Molecular Biology*, Vol.696, pp.179–203 (2011).
[7] Tanaka, S. et al.: Mass++: universal & plug-in style software for mass spectrometer, *Proc. 56th ASMS Conference on Mass Spectrometry and Allied Topics*, MP157 (2008).
[8] Utsunomiya, S. and Tanaka, S. et al.: Mass++ : A platform for mass spectrometry to construct suitable software to achieve user's own purposes, *Proc. 61st ASMS Conference on Mass Spectrometry and Allied Topics*, MP18-360 (2013).
[9] Mass++ Software License Agreement, available from ⟨http://www.first-ms3d.jp/english/mass2-license⟩.
[10] Yang, C. et al.: Comparison of public peak detection algorithms for MALDI mass spectrometry data analysis, *BMC Bioinformatics* 2009, Vol.10, No.4 (2009).
[11] Hardouin, J. et al.: Protein sequence information by matrix-assisted laser desorption/ionization in-source decay mass spectrometry, *Mass Spectrometry Reviews*, Vol.26, pp.672–682 (2007).
[12] MathWorks Documentation Center, available from ⟨http://www.mathworks.com/help/bioinfo/ref/msbackadj.html⟩.
[13] Utsunomiya, S. and Takeuchi, S. et al.: Development of new technique for water vapor correction on FTIR, *The 51st Annual Conference of Japan Society for Analytical Chemistry Book of Abstract*, Vol.19 (2002).

Shin-ichi Utsunomiya is a signal processing expert who has been engaged in algorithm and software development for various instruments. He developed some signal-processing algorithms for mass spectrometry on Mass++.

Yuichiro Fujita is an information technology engineer who has been investigating analysis software for mass spectrometry. He implemented external software functions including msbackadj on Mass++.

Satoshi Tanaka is a software developer who has been engaged in developing Mass++ since its first construction. He designed or developed user interfaces, core functions of the platform, and plug-in wizards for Mass++.

Shigeki Kajihara is a general manager of the Software Development Group in the FIRST ms3d project[*1]. He is also the team leader of the Mass++ (Ver. 2) development in Kyoto.

Ken Aoshima is a senior scientist who has been engaged in the multi-OMICS data analysis/mining effort for drug discovery. He is also the team leader of the Mass++ (Ver.2) development in Tsukuba.

Yoshiya Oda was the research director of the former CREST project[*2], in which Mass++ was initially developed. He is also the director of the Mass++ (Ver. 2) development in Tsukuba.

Koichi Tanaka is the core researcher of the FIRST ms3d project[*1], in which Mass++ (Ver. 2) was developed. He is also the director of the Mass++ (Ver.2) development in Kyoto.

(Communicated by *Yoichi Takenaka*)

[*1] Development of the next generation mass spectrometry system, and contribution toward drug discovery and diagnostics (2010–2014)
[*2] Development of Quantitative Metabolomics and Integration of Metabolome Data with Proteome Data (2005–2009)