GPMS: A Genetic Programming Based Approach to Multiple Alignment of Liquid Chromatography-Mass Spectrometry Data

Soha Ahmed[1(✉)], Mengjie Zhang[1], and Lifeng Peng[2]

[1] School of Engineering and Computer Science, Wellington, New Zealand
{soha.ahmed,mengjie.zhang}@ecs.vuw.ac.nz
[2] Victoria University of Wellington, 600, Wellington 6140, New Zealand
lifeng.peng@vuw.ac.nz

Abstract. Alignment of samples from Liquid chromatography-mass spectrometry (LC-MS) measurements has a significant role in the detection of biomarkers and in metabolomic studies.The machine drift causes differences between LC-MS measurements, and an accurate alignment of the shifts introduced to the same peptide or metabolite is needed. In this paper, we propose the use of genetic programming (GP) for multiple alignment of LC-MS data. The proposed approach consists of two main phases. The first phase is the peak matching where the peaks from different LC-MS maps (peak lists) are matched to allow the calculation of the retention time deviation. The second phase is to use GP for multiple alignment of the peak lists with respect to a reference. In this paper, GP is designed to perform multiple-output regression by using a special node in the tree which divides the output of the tree into multiple outputs. Finally, the peaks that show the maximum correlation after dewarping the retention times are selected to form a consensus aligned map.The proposed approach is tested on one proteomics and two metabolomics LC-MS datasets with different number of samples. The method is compared to several benchmark methods and the results show that the proposed approach outperforms these methods in three fractions of the protoemics dataset and the metabolomics dataset with a larger number of maps. Moreover, the results on the rest of the datasets are highly competitive with the other methods.

1 Background

LC-MS is commonly applied to both proteomic and metabolomic experiments. In LC-MS proteomics analysis, the sample is subjected to proteolytic digestion which results in a mixture of peptides. The resulting fraction of peptides mixture is then separated by liquid chromatography [1]. The peptides are then eluted at different retention times and detected by the mass spectrometer after ionization based on their mass to charge ratios [2]. Therefore, the resulting spectrum is a 3D map, called LC-MS map, which consists of mass to charge ratio (m/z), retention time (RT) and ion intensity count (Int). LC-MS can be used for providing quantitative and qualitative information about the proteins in a biological sample

A.I. Esparcia-Alcázar et al. (Eds.): EvoApplications 2014, LNCS 8602, pp. 915–927, 2014.
DOI: 10.1007/978-3-662-45523-4_74

[2]. Such information is useful in several applications including system biology, functional genomics and biomarker detection. For these applications to be successful, ideally the m/z and RT of the same molecule at different spectra among the LC-MS replicate runs detected in the same LC-MS platform should be the same. However, this is not always the case. In particular, there is a large shift and sometimes distortion in RT between different runs [2]. In addition, the m/z values show smaller distortion which introduces ambiguity in peak matching in comparative analyses. Moreover, the variations in RT may show non-linear deviations and can be greater than predicted [1]. Therefore, an effective algorithm is required to address two main tasks, the first is to match the peaks arising from the same peptides at different runs within certain m/z and RT windows and the second is to find the correct transformation of the RTs in order to make comparison [3] between the intensity values effectively.

The methods for alignment of LC-MS spectra can be classified into two groups. The first group is the raw-based methods, which select the set of significant peaks from raw data and use these peaks as a reference for aligning the data. These methods can avoid the errors due to feature detection but they have high computational cost [4]. The second group is the peak-based methods where the alignment is done after extracting features and grouping corresponding features (peaks) from different LC-MS runs [2]. However, feature extraction and centroidization can introduce some errors [4]. Therefore, the quality of the alignment algorithm will depend mainly on the quality of these preprocessing paradigms.

Examples of raw-based methods include the hidden Markov Models (HMMs) approach presented in [5], where the alignment of RT and the normalization of the peak intensities were done at the same time. HMMs were used to represent the correct retention times and the parameters of the model were estimated using the maximum likelihood estimation. A star-wise manner alignment of either raw or feature maps was depicted in [1] in the open source platform *OpenMS*. In the first phase, features were matched together using pose clustering followed by linear regression to correct the retention time distortion. In the second phase, the dewarped maps were combined into a consensus map by using the nearest neighbor search. The RANdom SAmple Consensus (RANSAC) algorithm was used in the *MZmine2* [6] framework to find features that fit a non-linear model within a user supplied m/z and RT tolerances. A locally-weighted scatter plot smoothing regression method was used on all the points obtained from RANSAC. Genetic algorithms were used in [7] to predict the RT dewarping function.

Most of these approaches for alignment of LC-MS data focus on solving the pairwise alignment problem, which produces somehow suboptimal results for multiple alignment problems.

Genetic programming (GP) is an evolutionary algorithm which solves a given problem by automatically evolving computer programs (functions) [8]. Initially, GP starts with random programs which are then modified using different genetic operators such as crossover and mutation based on Darwin evolution theory [8]. GP has been successfully used for alignment and forecasting of time series data

[9] and achieved good results. In particular, GP is well known for symbolic regression which provides great potential for aligning LC-MS data. However, GP has not been used for the alignment of LC-MS datasets.

1.1 Goals

The overall goal of this study is to develop a GP based method for multiple alignment of LC-MS peak maps which can correct the distortion of RT in multiple maps simultaneously. The proposed method is composed of two main phases, the first is to match the peaks across multiple maps and the second is to find the best dewarping function for the RT of the matched peaks. The method is tested on one protoemics dataset and two metabolomics datasets and compared against five benchmark algorithms. Specifically we will perform the following:

- develop an appropriate peak matching approach across multiple LC-MS maps with different number of peaks;
- design a GP method to perform multiple-output regression;
- model the terminal set of GP to perform multiple regression simultaneously; and
- investigate whether the new GP method outperforms the conventional alignment methods on these datasets.

1.2 Organisation

The rest of the paper is organised as follows. Section 2 describes the proposed approach and the new GP method. The experimental design, the datasets description and preprocessing are presented in Section 3. Section 4 reports the experimental results along with the discussions. The conclusions and future work are presented in Section 5.

2 The Alignment Approach

The objective of the alignment of LC-MS maps (we refer to each sample or run as a map) is to produce a consensus map which contains matching peaks of the same molecules from each map after transformation of RTs. In other words, the aim is to produce peak lists which have similar m/z and RT values in order to perform comparison of intensity values effectively.

The alignment approach proposed here works with peak data which has a much smaller amount of data than the raw maps. Therefore, it can be used to develop faster dewarping techniques. Figure 1 shows the overview of the proposed alignment approach which starts with taking the peak lists as inputs. The main aim of alignment is to find the possible transformations that maps the RT points of one map (reference map) $(r_1, r_2, ..., r_n)$ to the corresponding points of the other maps $(m_1, m_2, ..., m_x)$. To achieve this objective, the most matched partners must be detected by the peak matching approach which is used as an intermediate step to allow GP to search for the optimal transformation. The peak lists which have different number of peaks are passed to the peak matching

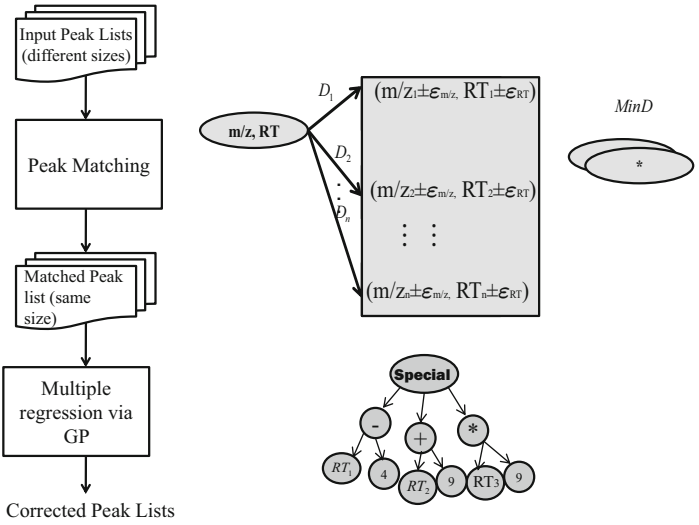

Fig. 1. Overview of the alignment approach

phase to detect the matched peak lists between the reference map and the other maps $((r_1, m_1), (r_2, m_2)....(r_n, m_n))$.

For pairwise alignment, GP can be used directly to evolve the transformation function. However, the multiple alignment of multiple maps requires a different structure of the evolved programs of GP to determine the transformation of the multiple maps. Therefore, a new GP multi-branch tree approach is developed for correcting RTs of multiple maps simultaneously. Finally, GP outputs the corrected peak lists. The two phases of the alignment approach are described below. For presentation convenience, the new approach is called GPMS.

2.1 Peak Matching

The first phase of the approach is to identify the significant matching peaks across all maps. The criteria for peak matching is the distance between the m/z and RT the reference map and the other maps. The procedure for peak matching is as follows:

1. Randomly select a map from the dataset as a reference map $R = (r_1, r_2, ..., r_n)$.
2. For each peak $(m/z_i, RT_i, Int_i)$ in the reference map, find the list of peaks in the next map $M = (m_1, m_2, ..., m_n)$ within a predefined m/z $(m/z_i \pm \varepsilon_{m/z})$ and RT $(RT_i \pm \varepsilon_{RT})$ tolerances and with the same charge.
3. Select the nearest neighbor (1-NN) peak from the list of peaks in the current map with respect to m/z, RT and Int, and add the two peaks as significant peaks of the reference and current maps into the consensus map. The distance between the peaks is measured using the Euclidean distance between m/z, RT and Int. More weight is given to m/z due to the fact that RT and Int

are much more tolerable than m/z. The Euclidean distance is given by:

$$ED = \sqrt{(W_1^2 * (R_{m/z} - M_{m/z})^2 + W_2^2 * (R_{RT} - M_{RT})^2 + W_3^2 * (R_{Int} - M_{Int})^2)}$$

where ED is the Euclidean distance between the two peaks of the reference (R) and the current (M) maps and W_1 =0.7, W_2=0.2 and W_3=0.1.

4. Mark the selected peak on the current map as a processed peak so that it will not be selected again as a nearest neighbor to another peak.
5. Repeat step 2- 4 on all the maps until all the peaks in all maps are processed. If there is no corresponding peak found in half of the maps, all significant peaks related to this peak are removed from the significant peak lists.

After identifying the matching peaks across all maps, the list of matching pairs is passed to GP to correct the RT values.

2.2 GP Multi-Branch Regression for Multiple Alignment

Unlike most of the previous RT alignment algorithms, our GP method corrects RTs of all maps simultaneously. The main advantage of this regression GP technique is that it can work efficiently. Another advantage is not having the requirement of a specific *gold standard* reference map for alignment of the rest of the maps. In other words, any map can be selected as a reference to align the rest of the maps. In this approach, we use the tree-based GP [10] for this task but we modified the tree structure as multi-branch tree. In the multi-branch GP approach, each individual is composed of several branches and each branch is responsible for evolving a part of the solution [10,11]. The final solution is integrating all these partial solutions through a special node which represents the root node [12,13]. The number of children of the special node is equal to the number of maps to be aligned. The children of the root node are the functions. The function node can also take other function nodes as its children. The terminal nodes of each branch are the RTs of a specific map and a random constant. The same branch cannot contain RTs from different maps. The structure of the multiple-output regression tree is shown in Figure 2.

In the rest of the section, we will describe terminal set, function set and the fitness function of the new GP method.

2.3 Terminal and Function Sets

An LC-MS sample is a 3D map composed of the m/z values, RTs and the intensity counts (Ints). The objective here is to correct the RTs of all maps to the corresponding RTs of the reference map. Therefore, the terminal set is composed of the RTs of N maps. We consider each input to GP as N RTs dimensions (equal to number of maps). For example, if we have three maps, each input to the terminal set is composed of three RT variables. We also used a random generated constant in the range of [-10,10] in the terminal set. Hence, our terminal set is composed of RTs values of all maps and random constants values. The function set used for this problem is $F = \{+, -, \times, \%, cos\}$, where

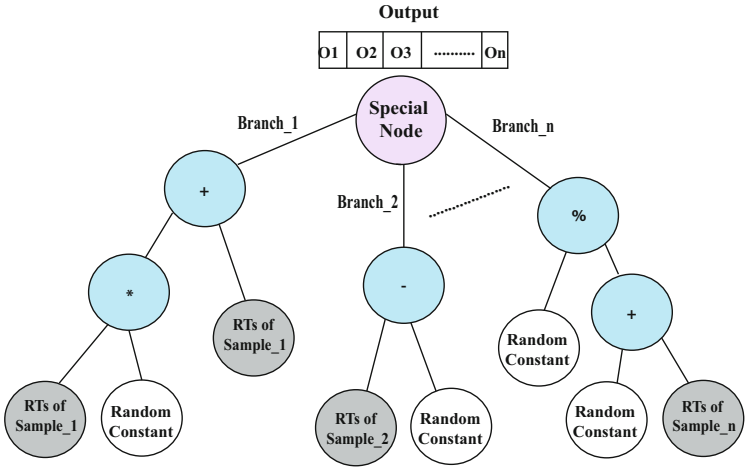

Fig. 2. Tree structure in the Multiple Alignment GP

% is the protected division operator which returns zero if the division is by zero. The aim of using *cos* operator is to evolve non-linear function for prediction and regression of the complex RTs deviations. The outputs (O_i) of each map are collected by the special node which is the root of the tree.

2.4 Fitness Function

For function approximation tasks, the performance can be measured as an error between the predicted and the real target values. As we have multiple outputs, each output corresponds to RTs of one map in the dataset, we calculate the sum of errors between the multiple outputs (which are the estimated outputs of the genetic programs) and the reference map output. The root mean square error (RMSE) is used as a fitness function. Thus the GP framework is to minimize the fitness so that the generated programs lead to minimum error between the RTs to be predicted. The RMSE fitness function is given by:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}\sum_{j=1}^{M}(RT_{ij} - \hat{RT}_{ij})^2}{N}}$$

where N and M are the number of maps and the number of RTs to be corrected in each map respectively. RT_{ij} is the i^{th} real RT value of the j^{th} map while $\hat{RT}_{ij}$ is the i^{th} estimated RT value of the j^{th} map by the GP program.

3 Experimental Design

3.1 Data Sets

We tested the proposed approach on one proteomics dataset (P_1) and two metabolomics datasets (M_1, M_2) obtained from the Open Proteomics Database

(OPD) [14] and Lange et al. [1]. Dataset P_1 contains two LC-MS runs with six different fractions and it originates from an *E.coli* sample. For this dataset each fraction is composed of pairs of LC-MS runs. The dataset was analyzed using LC/MS/MS with an ESI ion trap mass spectrometer (ThermoFinnigan Dexa XP Plus). It was exported into mzXML centroided mode and preprocessed using TOPP tools [15] to produce the peak lists which consist of the m/z, RT, intensity values and ignoring the charge states. The numbers of peaks in each fraction run were between 400 to 5800. A partial ground truth was produced using the first fraction of the dataset by linking the LC-MS spectra to the MS/MS of the SEQUEST search. More details about the steps for datasets preparation, analysis, preprocessing and parameters optimisation can be found in [1]. For the two metabolomics datasets, Arabidopsis thaliana leaf tissues were analyzed using two different LC-MS setups. An API QSTAR Pulsar i (Applied Biosystems/MDS Sciex) was used to produce 44 spectra for the M_1 dataset and a MicrOTOF-Q (Bruker Daltonics) to produce 24 spectra for the M_2 dataset. Peak extraction was done using XCMS software [16] resulting in 4000 to 17600 peaks in each spectrum. The ground truth was generated in the same study by selecting the high confident peaks. Those were the peaks found in more than four runs, having the same RT and also showing a high correlation in their peak shapes.

3.2 Genetic Operators and Parameters

The initial populations of GP are generated using the ramped half-and-half method. Each population consists of 1000 individuals in order to reduce the early convergence probability. The tournament selection method is used to select the individuals which can perform well for reproducing the new generations. The size of the tournament is set to 5. The standard crossover and mutation are used here with ratios of 80%, 19% respectively. Elitism is also used with a ratio of 1%. The depth of each individual is kept between 2 and 8. Each evolutionary process stops at the maximum generation 30 unless a perfect error of zero is found. The process is repeated for 30 independent runs. The random seed for each of the 30 runs in each set of experiments are all different. The peak matching phase parameters are as follows: the m/z tolerance and RT tolerance are set to 1.5, 100 respectively for dataset P_1 for all the fractions. For datasets M_1, M_2 the m/z tolerance and RT tolerance are set to 0.011, 20 respectively for both of them. Those parameters were selected after several tuning and they achieved the best results for our method. The GP implementation used in our experiments is the Evolutionary Computing Java-based (ECJ) package [17]. Table 1 describes the run time parameters used in the experiments.

3.3 Benchmark Algorithms

We compared our approach with previous published results of five publicly available benchmark algorithms for alignment of LC-MS maps which are: msInspect [18], MZmine [19], SpecArray [20], XAlign [21] and XCMS [16]. msInspect [18] works in a star-wise manner which aligns all maps with respect to a specific

Table 1. GP run time parameters

Parameter	Value
Initialization method	Ramped Half-and Half
Initial tree Depth	2
Maximum tree depth	8
Generations	30
Mutation probability	19%
Crossover Rate	80%
Elitisim	1%
Population Size	1000
Selection type	Tournament
Tournament Size	5
m/z tolerance	1.5, 0.011,0.011 for P_1, M_1, M_2 respectively
RT tolerance before correction	100, 20, 20 for P_1, M_1, M_2 respectively

reference map, which is the map with minimum number of peaks. The process starts with the selection of the most intense peak within a certain RT tolerance and the removal of the rest of the peaks. After that, pairing the remaining peaks with peaks of similar m/z is performed. Smoothing spline regression is used for dewarping and finally divisive clustering is used to obtain the consensus map. The main disadvantage of this approach is the removal of less intense peaks which might cause the loss of many important peaks. MZmine [19] works by scoring the similarity of all features against a master list and if the score is "good enough" the feature is assigned to the best matched row. MZmine does not perform any transformation of RT. SpecArray [20] schema works as pairwise alignment and combine the pairwise aligned maps into a consensus map until all maps are aligned. SpecArray is not applicable to a dataset with a big number of maps. XAlign [21] also works in a star-wise manner and selects the most intense peaks within a user defined m/z and RT tolerance, the map with the minimum difference to the average RTs is chosen as a reference map. After dewarping the RT, the features with high correlation coefficient are selected to form the consensus map. XCMS [16] works as a multiple alignment approach where peak matching is performed in the first phase by using a fixed interval bin and using kernel density estimation to determine the distribution of the features. Boundaries of regions with features that have similar RTs are selected. Finally non-linear regression is used to correct RTs.

3.4 Performance Evaluation

The performance of the proposed approach is measured through the precision (PR) and recall (RE) measures. Precision is the probability that a found item is relevant, which is in our case the percentage of the correctly aligned peaks among all the peaks aligned by the approach.

$$PR = \frac{\text{Number of correctly aligned peaks}}{\text{Total number of peaks aligned}}$$

Whereas, recall is the probability that a relevant item is found (the percentage of the correctly aligned peaks among the peaks in the ground truth [22]).

$$RE = \frac{\text{Number of correctly aligned peaks}}{\text{Total number of peaks in the ground truth}}$$

The harmonic mean of the precision and recall is measured through the F-measure [22].

$$\text{F-measure} = \frac{2*\text{PR}*\text{RE}}{\text{PR}+\text{RE}}$$

Precision and recall of alignment were calculated using the evaluation script provided by Lang et al. [1].

4 Results and Discussions

4.1 Effectiveness Performance

GPMS is initially tested for the pairwise alignment on P_1 which is available in six different fractions. P_1 shows a large deviation in RT values which is a challenge for the alignment tool to correct the RT. Tables 2 and 3 show the results of the five conventional approaches compared to our approach notated as GPMS. As shown in Tables 2 and 3, GPMS achieved much better performance than msInspect and SpecArray in all the three datasets. GPMS outperformed all other methods in three fractions of P_1. For the first fraction (00), the mean of the 30 runs of GPMS is better than msInspect by 44 % in terms of precision, 30% in terms of recall and 38% in terms of F-measure. For the other approaches GPMS improves the precision by 1-25%, the recall and F-measure by 1-21%. For fraction (20), GPMS achieves similar performance as XCMS and has the third rank after MZmine and XAlign. GPMS performs better than msInspect, SpecArray and XCMS for fraction 40. Furthermore, our new method is the third best after MZmine and XAlign for the same fraction. For fractions (60) and (100), GPMS outperforms all other methods in terms of precision (which reaches 1.00 for fraction (100)) and F-measure. The proposed method has the best recall in fraction (60) while in fraction (100) it has the third best recall after Xalign and XCMS. Finally for fraction (40), the performance of GPMS was slightly better to XCMS and it is the second best after MZmine. In general, for P_1 the proposed method outperforms the other methods in three fractions, the second best in two fractions and third best in one fraction.

For datasets M_1 and M_2 which contain 44 and 24 maps respectively, the challenge for the alignment approach on these complex metabolomics datasets is to assign the most suitable matches and to correct the RT distortion across multiple maps. SpecArray did not manage to produce any results for these complex alignment tasks. As shown in Table 3, GPMS appears to be more powerful in aligning a large number of maps as in the dataset M_1 (44 maps). For M_1, it has better performance than other methods by 1-31% in terms of precision and 2- 49% with respect to F-measure. This suggests that the proposed method can be more powerful for multiple map alignment. The performance of GPMS outperforms msInspect in terms of precision by 41.87%, XCMS by 1% and it is equal to XCMS for M_2. In terms of recall, it is much better than msInspect and SpecArray. GPMS is better than msInspect by 53% and it outperforms SpecArray which did not manage to achieve results in terms of F-measure. Overall, the

Table 2. Proteomics dataset P_1 alignment results

Fraction	Measure	msInspect	MZmine	SpecArray	XAlign	XCMS	GPMS		
							Min	Max	Mean ±St.Dev.
00	Precision	0.38	0.81	0.61	0.82	0.58	0.82	**0.83**	**0.83±0.003**
	Recall	0.52	0.75	0.61	0.82	0.62	0.82	**0.83**	**0.82±0.004**
	F-measure	0.44	0.78	0.61	0.82	0.60	0.82	**0.83**	**0.82±0.004**
20	Precision	0.45	**0.88**	0.62	0.85	0.80	0.80	0.82	0.81±0.0100
	Recall	0.56	**0.87**	0.62	0.85	0.81	0.80	0.80	0.80±0.0000
	F-measure	0.50	**0.87**	0.62	0.85	0.80	0.80	0.81	0.81±0.0060
40	Precision	0.48	**0.90**	0.75	0.87	0.80	0.83	0.84	0.84±0.002
	Recall	0.63	**0.87**	0.75	**0.87**	0.81	0.81	0.81	0.81±0.0
	F-measure	0.54	**0.88**	0.75	0.87	0.80	0.82	0.82	0.82±0.003
60	Precision	0.54	0.84	0.71	0.87	0.75	**0.91**	**0.91**	**0.91±0.000**
	Recall	0.73	0.79	0.71	0.87	0.78	**0.92**	**0.92**	**0.92±0.000**
	F-measure	0.62	0.81	0.71	0.87	0.76	**0.91**	**0.91**	**0.91±0.005**
80	Precision	0.57	**0.94**	0.74	0.90	0.88	0.90	0.90	0.90±0.000
	Recall	0.70	**0.92**	0.74	0.90	0.89	0.89	0.89	0.89±0.0000
	F-measure	0.63	**0.93**	0.74	0.90	0.88	0.90	0.90	0.90±0.0040
100	Precision	0.56	0.92	0.77	0.96	0.96	**1.00**	**1.00**	**1.00±0.000**
	Recall	0.82	0.94	0.77	**0.96**	**0.96**	0.94	0.94	0.94±0.000
	F-measure	0.67	0.93	0.77	0.96	0.96	**0.97**	**0.97**	**0.97±0.000**

Table 3. Metabolomics datasets M_1 and M_2 alignment results

Fraction	Measure	msInspect	MZmine	SpecArray	XAlign	XCMS	GPMS		
							Min	Max	Mean ±St.Dev.
M_1	Precision	0.46	0.74	-	0.70	0.70	**0.77**	**0.77**	**0.77±0.003**
	Recall	0.27	0.89	-	0.88	**0.94**	0.89	0.91	0.9±0.004
	F-measure	0.34	0.81	-	0.78	0.80	**0.83**	**0.83**	**0.83±0.001**
M_2	Precision	0.47	**0.84**	-	0.79	0.78	0.79	0.79	0.79±0.001
	Recall	0.23	**0.98**	-	0.93	**0.98**	0.90	0.90	0.90±0.000
	F-measure	0.31	**0.90**	-	0.85	0.87	0.84	0.84	0.84±0.001

performance of GPMS is the second best with respect to precision, third best with respect to recall and F-measure in M_2. In general, GPMS is among the top two methods or even performs best (00, 60, 100 of P_1, M_1).

4.2 Efficiency Performance

Another comparison is done in terms of the run time of each of the methods and the results are shown in Table 4. For all the datasets, GPMS average run time is much better than all other approaches. The computational cost (in terms of time) of GPMS is more lower than the rest of methods, which represents another advantage of GPMS. For all the datasets, GPMS improves the efficiency by an order of magnitude than the rest of the methods except for XCMS. GPMS is also more efficient than XCMS in terms of computational time for P_1 and M_2. Moreover, the efficiency of GPMS for M_2 in one of the runs is also better than XCMS.

Table 4. Comparison of run time of GPMS with other approaches (in seconds)

Dataset	msInspect	MZmine	SpecArray	XAlign	XCMS	GPMS		
						Min	Max	Mean ±St.Dev.
P_1	60	40.2	111	69	54	**4.1**	**9.8**	**6.1±1.20**
M_1	720	1200	-	3060	54	**36.34**	64.92	64.92±4.97
M_2	2160	2640	-	2100	348	**81.10**	**94.20**	**87.37±3.23**

(SPE T_0 (- (- T_1 9.05) (cos T_1)))

Input		Output	
T_0	T_1	T_0	T_1
1263.95	1271.96	1263.95	1263.89
1307.84	1315.58	1307.84	1307.09
1708.72	1717.28	1708.72	1708.10

(a)

(SPE T_0 (+ T_1 17.56))

Input		Output	
T_0	T_1	T_0	T_1
182.95	165.425	182.95	182.98
111.45	94.12	111.45	111.68
455.08	438.12	455.08	455.68

(b)

Fig. 3. (a) An evolved model for fraction (00) with some examples of inputs and outputs of the model. (b) An evolved model for fraction (100).

4.3 Interpretation of the Evolved Regression Models

Some examples of the evolved regression models are shown below:

Figure 3 shows some examples of the evolved models for fractions (00) and (100). SPE refers to the special node which is the root node collecting the multiple outputs of the tree. T_0 refers to the RTs of the first map while T_1 refers to the RTs of the second map. The first map (T_0) is selected as the reference map in which the RTs of both maps should be corrected according to it. The dewarping functions of both inputs are determined simultaneously through the multiple branches. As shown in Figure 3 (a), GP managed to determine the correct amount of shift for the RTs of the second map (T_1) through a non linear dewarping model in the second branch of the tree. The RTs of first map (T_0) (the first branch of the tree) is kept the same as it has been selected as the reference map. Some examples are shown in the same figure where the inputs to the models and the mapped outputs after correction shows that GP has successfully aligned the maps with respect to the reference map. The evolved model for fraction (100) is shown in Figure 3 (b) where the GP dewarping function has managed to correct the distortion of RTs through a linear function. Examples of inputs and outputs of fraction (100) are also shown in Figure 3 (b).

5 Conclusions and Future Works

In this paper, we propose a new method for multiple alignment of LC-MS peak data. The proposed method has two phases. In the first phase, the partner peaks across multiple maps are detected in order to form the matched peak lists. In the second phase, the matched peak lists are passed to GP to perform the correction of RTs of all maps simultaneously. The new GP approach is depicted by dividing the tree into multiple branches, in which each branch produces the output dewarping function of each map with respect to the reference map. The proposed GP-based method (GPMS) was tested on one protoemics dataset of six different fractions and two metabolomics datasets. The results show that GPMS achieves better precision, recall and F-measure than five other LC-MS benchmark alignment methods for three fractions of the protoemics dataset and

one metabolomic dataset which has larger number of maps. This suggests that GPMS is more powerful in multiple alignment of LC-MS data. The proposed method also shows very competitive results in the rest of the datasets. GPMS in general is always either the best or among the two top methods for these datasets. Furthermore, the proposed GP method is much more efficient in terms of computational time than the benchmark methods.

Although very preliminary, this paper represents the first work of GP for multiple alignment of LC-MS data, and the competitive results of the proposed method encourages us to do further investigation in this direction in the future.

For future works, we will consider merging a clustering scheme to the first phase of the approach. This will relate to another interesting but challenging research direction, i.e. using GP for peak matching through a clustering approach which can match the partner peaks better.

References

1. Lange, E., Gröpl, C., Schulz-Trieglaff, O., Leinenbach, A., Huber, C.G., Reinert, K.: A geometric approach for the alignment of liquid chromatography-mass spectrometry data. Bioinformatics **23**(13), 273–281 (2007)
2. Vandenbogaert, M., Li-Thiao-Te, S., Kaltenbach, H., Zhang, R., Aittokallio, T., Schwikowski, B.: Alignment of LC-MS images, with applications to biomarker discovery and protein identification. Proteomics **8**(4), 650–672 (2008)
3. Lange, E., Tautenhahn, R., Neumann, S., Gropl, C.: Critical assessment of alignment procedures for LC-MS proteomics and metabolomics measurements. BMC Bioinformatics **9**(1), 375–394 (2008)
4. Heidi Vhmaa, Ville R. Koskinen, W.H.: PolyAlign: A versatile LC-MS data alignment tool for landmark-selected and automated use. International Journal of Proteomics, pp. 1–10 (2011)
5. Listgarten, J., Neal, R., Roweis, S., Wong, P., Emili, A.: Difference detection in LC-MS data for protein biomarker discovery. Bioinformatics **23**(2), 198–204 (2007)
6. Pluskal, T., Castillo, S., Villar-Briones, A., Oresic, M.: MZmine 2: Modular framework for processing, visualizing, and analyzing mass spectrometry-based molecular profile data. BMC Bioinformatics **11**, 395 (2010)
7. Palmblad, M., Mills, D.J., Bindschedler, L.V., Cramer, R.: Chromatographic Alignment of LC-MS and LC-MS/MS Datasets by Genetic Algorithm Feature Extraction. Journal of the American Society for Mass Spectrometry **18**(10), 1835–1843 (2007)
8. Poli, R., Langdon, W.B., McPhee, N.F.: A field guide to genetic programming. Lulu Enterprises, UK Ltd. (2008)
9. Ahalpara, D.P.: Improved forecasting of time series data of real system using genetic programming. In: Proceedings of the 12th Annual Conference on Genetic and Evolutionary Computation, GECCO 2010, pp. 977–978. ACM, New York (2010)
10. Smart, W.D., Zhang, M.: Probability based genetic programming for multiclass object classification. In: Proceedings of the 8th Pacific Rim International Conference on Artificial Intelligence, pp. 251–261 (2004)
11. Rodríguez-Vázquez, K., Oliver-Morales, C.: Multi-branches Genetic Programming as a Tool for Function Approximation. In: Deb, K., Tari, Z. (eds.) GECCO 2004. LNCS, vol. 3103, pp. 719–721. Springer, Heidelberg (2004)

12. Zhang, Y., Zhang, M.: A multiple-output program tree structure in genetic programming. In: Proceedings of The Second Asian-Pacific Workshop on Genetic Programming, pp. 1–12 (2004)
13. Defoin Platel, M., Vérel, S., Clergue, M., Chami, M.: Density Estimation with Genetic Programming for Inverse Problem Solving. In: Ebner, M., O'Neill, M., Ekárt, A., Vanneschi, L., Esparcia-Alcázar, A.I. (eds.) EuroGP 2007. LNCS, vol. 4445, pp. 45–54. Springer, Heidelberg (2007)
14. Prince, J., Carlson, M., Lu, R., Marcotte, E.: The need for a public proteomics repository. Nat. Biotechnol. **22**, 471–472 (2004)
15. Kohlbacher, O., Reinert, K., Gropl, C., Lange, E., Pfeifer, N., Schulz-Trieglaff, O., Sturm, M.: TOPP-the OpenMS proteomics pipeline. Bioinformatics **23**(2), 191–197 (2007)
16. Smith, C., Want, E., O'Maille, G., Abagyan, R., Siuzdak, G.: XCMS: processing mass spectrometry data for metabolite profiling using nonlinear peak alignment, matching, and identification. Anal. Chem. **78**(3), 779–787 (2006)
17. White, D.R.: Software review: the ECJ toolkit, 65–67 (2012)
18. Bellew, M., Coram, M., Fitzgibbon, M., Igra, M., Randolph, T., Wang, P., May, D., Eng, J., Fang, R., Lin, C., Chen, J., Goodlett, D., Whiteaker, J., Paulovich, A., McIntosh, M.: A suite of algorithms for the comprehensive analysis of complex protein mixtures using high-resolution LC-MS. Bioinformatics **22**(15), 1902–1909 (2006)
19. Katajamaa, M., Miettinen, J., Oresic, M.: MZmine: Toolbox for processing and visualization of mass spectrometry based molecular profile data. Bioinformatics **22**, 634–636 (2006)
20. Li, X., Yi, E., Kemp, C., Zhang, H., Aebersold, R.: A software suite for the generation and comparison of peptide arrays from sets of data collected by Liquid Chromatography-Mass Spectrometry. Molecular & Cellular Proteomics: MCP **4**(9), 1328–1340 (2005)
21. Zhang, X., Asara, J., Adamec, J., Ouzzani, M., Elmagarmid, A.: Data preprocessing in liquid chromatography/mass spectrometry-based proteomics. Bioinformatics **21**(21), 4054–4059 (2005)
22. Voss, B., Hanselmann, M., Renard, B., Lindner, M., Kthe, U., Kirchner, M., Hamprecht, F.: Sima: simultaneous multiple alignment of lc/ms peak lists. Bioinformatics **27**(7), 987–993 (2011)