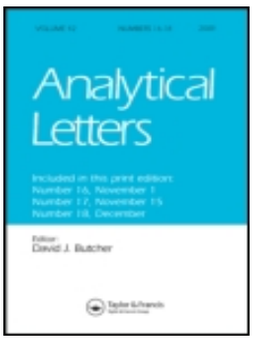

Analytical Letters

ISSN: 0003-2719 (Print) 1532-236X (Online) Journal homepage: http://www.tandfonline.com/loi/lanl20

A Wavelet-Based Genetic Algorithm for Compression and De-Noising of Chromatograms

Xueguang Shao , Fang Yu , Hongbing Kou , Wensheng Cai & Zhongxiao Pan

To cite this article: Xueguang Shao , Fang Yu , Hongbing Kou , Wensheng Cai & Zhongxiao Pan (1999) A Wavelet-Based Genetic Algorithm for Compression and De-Noising of Chromatograms, Analytical Letters, 32:9, 1899-1915

To link to this article: http://dx.doi.org/10.1080/00032719908542941

Published online: 18 Feb 2008.

ANALYTICAL LETTERS, 32(9), 1899–1915 (1999)

A WAVELET-BASED GENETIC ALGORITHM FOR COMPRESSION AND DE-NOISING OF CHROMATOGRAMS

Keywords: Wavelet, Genetic algorithm, Compression and de-noising, Chromatograms

Xueguang Shao[a*], Fang Yu, Hongbing Kou, Wensheng Cai, Zhongxiao Pan

[a]Department of Chemistry, Department of Applied Chemistry,
University of Science and Technology of China, Hefei, Anhui, 230026,
P.R.China.

ABSTRACT

A wavelet-based genetic algorithm using real-number coding and arithmetical crossover method in signal processing is described in this work. Due to the characteristic of the wavelet, an analytical signal can be represented by a finite linear combination of wavelet-based functions. Using a wavelet-based genetic algorithm to find the coefficients to such representation, an analytical signal can be reconstructed by the coefficients and the corresponding elementary

function. Therefore the method can be used to compress and de-noise analytical signals because the insignificant information such as noise will not be reserved in the reconstructed signal. Both simulated signals and experimental multicomponent chromatograms are successfully compressed and de-noised with the proposed algorithm.

INTRODUCTION

Wavelet transform is a high performance signal-processing technique[1-3] and has been used in the compression,[2] de-noising,[3] baseline correction,[4] and the resolution of multicomponent overlapping chromatograms[5]. The main advantage of the wavelet transform is that it decomposes a signal into fixed building blocks of constant shape but at different scales and positions, and each of building blocks represents the information at a different frequency. Therefore, the wavelet transform is a powerful tool for time-frequency analysis. Genetic Algorithms (GAs) were introduced by John Holland[6] in 1975, as a probabilistic search technique. Due to the advantages of global and parallel searching ability, GA has been applied to combinatorial and parameter optimizations[7-9].

In a general time-frequency decomposition, a signal is decomposed into a set of elementary functions, characterized by time and dimensions in the time-frequency plane. The analyzing signal can be represented by a linear combination of a given number of elementary functions with different time-frequency windows. In this study, wavelet-based elementary functions[6] were used to represent analytical signals, and a genetic algorithm was adopted to optimize the involved parameters, i.e., translation, dilation, and the corresponding coefficients. Therefore, a wavelet-based genetic algorithm was called for the proposed method. The advantage of the method is that it optimizes all the parameters for the wavelet-based functions simultaneously without any knowledge about the analyzing signal.

THEORY

Wavelet Analysis

Wavelet is defined as a family of functions which are derived by dilation and translation from a unique function $\psi(x)$. The $\psi(x)$ is called the basis of a wavelet and the corresponding wavelet family $\{\psi_{a,b}\}$ is given by

$$\psi_{a,b}(x)=\frac{1}{\sqrt{|a|}}\psi\left(\frac{x-b}{a}\right), \qquad a,b\in R, a\neq 0 \qquad (1)$$

where a and b denote the parameters to control the dilation and translation, respectively, and $\frac{1}{\sqrt{|a|}}$ is the normalization constant. The wavelet transform of a function $f(x) \in L^2(R)$ is defined by

$$Wf(a,b)=\langle f(x),\psi_{a,b}(x)\rangle=\frac{1}{\sqrt{|a|}}\int_{-\infty}^{+\infty} f(x)\psi\left(\frac{x-b}{a}\right)dx \qquad (2)$$

The discrete form of equation (1) and (2) can be written as

$$\psi_{j,k}(x)=a_0^{-j/2}\psi\left(a_0^{-j}x-kb_0\right) \qquad (3)$$

and

$$Wf(j,k)=\langle f(x),\psi_{j,k}(x)\rangle=2^{-j/2}\int_{-\infty}^{+\infty} f(x)\psi\left(2^{-j}-k\right)dx \qquad (4)$$

where $a=a_0^j$, $b=kb_0a_0^j$, j and k are integers and, a_0=2 and b_0=1 generally. Therefore, a discrete signal can be represented by

$$f(x)=\sum_j\sum_k Wf(j,k)\psi_{j,k}(x) \qquad (5)$$

A signal can be precisely approximated by a wavelet representation due to the characteristic of the wavelet. For low frequency components, the windows controlled by the dilation parameter j are wide in time domain and narrow in frequency domain, giving good frequency resolution, and for high frequency components, the windows are narrow in time domain and wide in frequency

domain, allowing good time resolution. Such a sufficient coverage of the time-frequency plane makes the method very effective for signal analysis.

Genetic Algorithm

A genetic algorithm is a stochastic search technique, based on simulations of the mechanisms of natural selection and natural genetics. It has been widely used in optimization problems, especially in the complex, multi-variable, and non-linear optimization problems which are difficult to solve by usual search methods. Similar to the natural evolution, the genetic algorithm is able to find the best chromosome from a population by operating genes on chromosomes to make them possess much higher fitness to the environment. To a practical problem, a set of candidate solutions constitute the population which is to be optimized, one candidate solution (individual) is presented as a set of parameters (genes) which are encoded in a chromosome as a numerical string. A fitness function is used to evaluate the quality of the individuals in population. Parent individuals are selected from the population according to their fitness values with the rule that individuals with higher fitness have higher probability of survival than the lower ones. Gene crossover and mutation are operated on the corresponding chromosomes to generate a new population with a certain probability, respectively.

The basic procedures of GAs generally include: (1)population initialization, (2)evaluation of each individual, (3)selection of parents based on the fitness values, (4)crossover and mutation. The whole procedure can be illustrated as Figure 1.

Crossover or recombination of genes between two parent individuals is implemented in various ways, such as single-point, two-point, and uniform crossover. The single-point and two-point crossover act in a similar way in which genes after or between the randomly selected point(s) will be exchanged to form new individuals, i.e., the children. Uniform crossover can be viewed as multi-

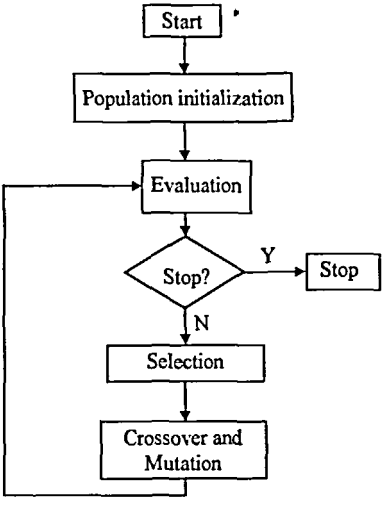

Figure 1. Flowchart of a genetic algorithm

point crossover in which both the points and the number of the points are determined by probability. Mutation will be applied to individuals by changing randomly selected genes according to a probability threshold.

Wavelet-based Genetic Algorithm for Signal Decomposition

The aim of this study is to represent a signal by a finite linear combination of elementary functions with arbitrary time-frequency windows in the following form

$$\hat{f} = \sum_{i=1}^{N} c_i \varphi_{a_i,b_i} \qquad (6)$$

where the number N is the number of elementary functions, φ_{a_i,b_i} is a set of elementary functions defined by dilation with a_i and translation with b_i, c_i is the corresponding coefficients, $\hat{f}$ can be regarded as an approximation of the original signal f. The genetic algorithm is applied to find all the parameters a_i, b_i, and c_i with the best fitness to the original signal.

The elementary functions φ_{a_i,b_i} are derived from a locally support wavelet-like basis function by dilation and translation. These elementary functions are not necessary independent each other. In this study, the following four elementary functions were investigated:

(1) B2-spline

$$\varphi(t)=\begin{cases}\frac{1}{2}t^2 & 0\le t<1\\ \frac{3}{4}-(t-\frac{3}{2})^2 & 1\le t<2\\ \frac{1}{2}(t-3)^3 & 2\le t<3\\ 0 & otherwise\end{cases} \tag{7}$$

(2)B3-spline

$$\varphi(t)=\begin{cases}\frac{1}{6}t^3 & 0\le t<1\\ -\frac{1}{2}(t-2)^3-(t-2)^2+\frac{2}{3} & 1\le t<2\\ \frac{1}{2}(t-2)^3-(t-2)^2+\frac{2}{3} & 2\le t<3\\ \frac{1}{6}(4-t)^3 & 3\le t<4\\ 0 & otherwise\end{cases} \tag{8}$$

(3) Marr function

$$\varphi(t)=\frac{1}{\sqrt{2\pi}}(1-t^2)\exp(-t^2/2) \tag{9}$$

(4) Morlet wavelet basis function

$$\varphi(x)=\cos(1.75x)\exp(-x^2/2) \tag{10}$$

For a given basis function, a real-valued triplet (a_i, b_i, c_i) for each elementary function is represented as genes, and a given number of the triplets are represented as a chromosome which approximates the given signal. A simple way to encode the parameters in the form of real number strings is adopted as the following definition:

```
typedef struct{
    float a;      //dilation factor
    float b;      //translation factor
```

```
    float c;        //coefficient
  }GENE;
typedef struct{
    GENE chrome[nChrome]; //nChrome = number of elementary
                          //functions in a chromosome
    float fitness;        // fitness of individual
  }CHROME;
```

The standard selection mechanism based on fitness scaling is used with a slight modification to ensure the diversity in the population. An elitist strategy is employed in which the best individual of the population always survives to the next generation.

Two methods for mutation, i.e., real-number adaptive mutation ratio method and dynamic adjustment of mutation probability, are used respectively. For the adaptive mutation, during the initial stage random mutation is used to ensure the diversity in the population. When the best fitness does not improve within 50 generations, big creep mutation giving a big disturbance around a real number will be used to enlarge the search space. When another 50 generations passed without any evolution, little creep mutation giving a little disturbance around a real number will be used to locally optimize the chromosome. The other mutation method is to adjust mutation probability during the evolution process, i.e. to start the algorithm with an initial mutation probability during the initial stage, to increase the probability gradually for each 50 generations without any evolution until a threshold, and then to decrease the probability gradually back to the initial value during the last stage.

Besides the standard single point crossover, arithmetical crossover is also implemented, which takes two (real-valued) parent genes, s and t, and calculates their offspring genes, s' and t', as a linear combination of the parents' string by

$$\begin{aligned} s' &= k \cdot s + (1-k) \cdot t \\ t' &= (1-k) \cdot s + k \cdot t \end{aligned} \qquad (11)$$

with the parameter $k \in [0,1]$. For each individual gene participating in the crossover, the parameter k is an uniformly random choice from the interval [0,1].

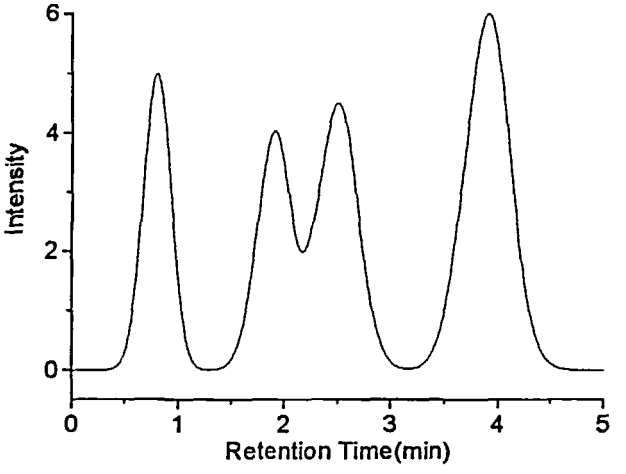

Figure 2 Simulated four component chromatogram with 512 data points

The evaluation function in this work is defined as the norm of the difference between the signal f and its approximation $\hat{f}$ divided by the signal norm $\|f\|$ to produce a relative error measurement as in equation (12).

$$F = \frac{\|f - \hat{f}\|}{\|f\|} = \frac{\left\{\sum_i |f(i) - \hat{f}(i)|^2\right\}^{1/2}}{\left\{\sum_i |f(i)|^2\right\}^{1/2}} \qquad (12)$$

The value F is to be minimized by the algorithm.

DATA PREPARATION AND CALCULATION

Figure 2 shows the simulated four-component chromatogram by Gaussian equation,

$$f(t) = \sum_{j=1}^{n} c_j \exp\left(-\frac{(t - t_{0,j})^2}{2\sigma_j^2}\right) \qquad (13)$$

where $f(t)$ is the simulated chromatogram with 512 discretely sampled data points, t is retention time, n is the component number, c_j and t_{0j} are the

concentration and position of component *j*, respectively. The width at half height of the simulated peaks can be obtained by $2\sigma\sqrt{2\ln(2)}$.

Experimental chromatograms were obtained on an HPLC system of Spectrasystem FL2000(Spectra-Physics, USA) and Shimadzu LC-6A (Shimadzu, Japan), respectively. The column was packed with ODS silica of 10μm (250×5mm, Shimadzu), and the post-column reaction agent was delivered by a LC-6A pumps (Shimadzu). The samples are mixed rare earths. The experimental conditions and sample preparation are the same as our previous works[11,12].

A computer program was written in C++ language and implemented on a Pentium-266. In all calculations, the population size is 50, the maximum generation number is 5000 (for the last experimental chromatogram, 10000 was used), the crossover probability is 0.9. For the adaptive mutation, the mutation probability is 0.033, the big and little disturbance of a real number are 20% and 1%, respectively. For dynamic mutation, the initial mutation probability is 0.01, and it will increase gradually (0.01 for every 50 generations without evolution) until it reaches 0.05. Whenever the probability reached 0.05, it will decrease gradually (0.01 for every 50 generations without evolution) until its initial value. The span for parameters *a*, *b*, and *c* are respectively controlled as the following:

$$a \in [0.001, 1.2]$$

$$b \in [t_{\min} - 0.75a, t_{\max} + 0.75a]$$

$$c \in \left[f_{\min} - \frac{\Delta f}{2}, f_{\max} + \frac{\Delta f}{2} \right]$$

where $t_{\min}$ and $t_{\max}$ are the minimum and maximum of the retention time, $f_{\min}$ and $f_{\max}$ are the minimum and maximum of the chromatographic signal, and Δf is the difference between $f_{\min}$ and $f_{\max}$.

RESULTS AND DISCUSSION

Effect of Genetic Operators on the Algorithm

In order to improve the performance of the genetic algorithm, results obtained

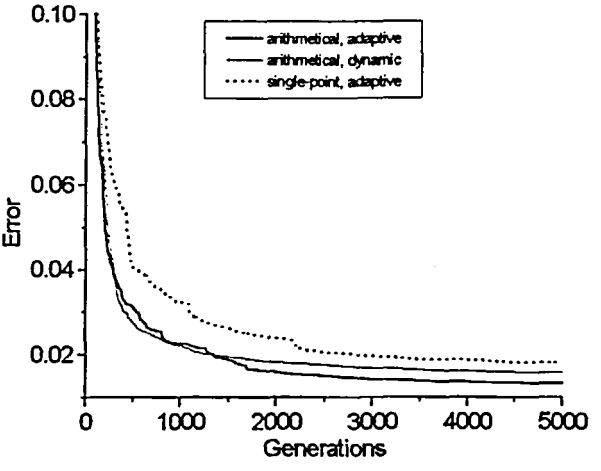

Figure 3 The error curves vs. generation averaged over 5 runs

with different crossover and mutation methods in the genetic algorithm were compared. Table 1 lists the fitness obtained with different crossover and mutation methods, 10 elementary functions from B2-spline basis for the chromatogram in figure 2. Figure 3 shows the comparison between the error curves vs generation for three different methods in Table 1. From Table 1, it is clear that the arithmetical crossover with adaptive mutation is the best method. From figure 3, the evolution procedure for each method can be investigated. It is clear that the arithmetical crossover with adaptive mutation method is much better than the single-point crossover with adaptive mutation method, and slightly better than the arithmetical crossover with dynamic mutation method only during the final period.

Effect of the Basis Functions on the Reconstructed Results

Table 2 shows the results, in which the fitness is calculated by equation (12), obtained with B2-spline, B3-spline, Marr, and Morlet basis functions,

Table 1 Results with different genetic operators*

Crossover type	Mutation type	Best fitness	Mean fitness
Arithmetical	Adaptive	0.0100	0.0133
Arithmetical	Dynamic	0.0126	0.0159
Single-point	Adaptive	0.0116	0.0181

*: Results over 5 runs were sampled.

Table 2 Results using different basis functions*

Function	best fitness	mean fitness
B2-spline	0.0100	0.0133
B3-spline	0.0107	0.0114
Marr	0.5959	0.6137
Morlet	0.1611	0.1733

*: Results over 5 runs were sampled.

respectively. 10 elementary functions are used in the calculation. From both the best fitness and mean fitness in the table it can be seen that B2- and B3-spline are superior to the other two functions. On the other hand, due to the relatively simple and explicit analytical form, B-spline functions are easy to evaluate and manipulate, and consume less CPU-time than the others. Therefore B2-spline was chosen in the following discussion because B2 is simpler than B3-spline. The result of approximation using B2-spline is shown in figure 4 by dot line, the fitness of which is 0.0100. It can be seen that the reconstructed chromatogram is almost the same as the simulated one.

Effect of the Number of Elementary Function on the Reconstructed Results

Table 3 tabulates the fitnesses by equation (12) obtained with B2-spline function as basis function but a different number of elementary functions. From

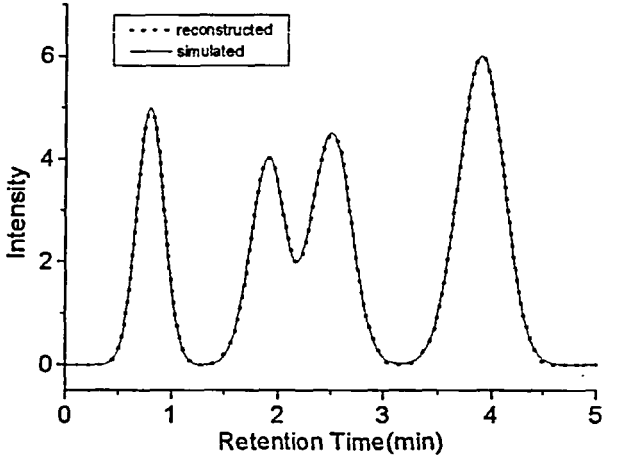

Figure 4 Comparison between simulated and reconstructed chromatograms

Table 3 Results using different number of elementary functions*

number of functions	compression ratio	best fitness	mean fitness
5	34.1:1	0.0211	0.0282
10	17.1:1	0.0100	0.0133
15	11.4:1	0.0081	0.0119
20	8.5:1	0.0123	0.0146
25	6.8:1	0.0190	0.0411
30	5.7:1	0.0875	0.1027

*: Results over 5 runs were sampled.

both the best fitness and the mean fitness in the table, it can be seen that there is no significant difference when the number of functions is between 10-20 with 15 being the best. Considering that the compression ratio becomes smaller and the computation becomes slower with the increase of the number due to the increase of the optimizing parameters, 10 or 15 should be the best value for the number of

Table 4 Results obtained from the simulated chromatograms with different level of noise*

noise level	best fitness	mean fitness	RE**
5%	0.0375	0.0378	0.0183
10%	0.0704	0.0714	0.0237
20%	0.1391	0.1402	0.0382
30%	0.2048	0.2059	0.0581
40%	0.2700	0.2707	0.0741
50%	0.3299	0.3306	0.0960

*: Results over 5 runs were sampled.

**: RE is calculated by equation (12) using the reconstructed and the simulated chromatogram without noise.

the elementary functions. But it should be noted that the suitable number of the elementary functions should be related with the complexity (the number of peaks) of analyzing signal. It should be larger with the increase of the complexity of signal.

Effect of Noise Level on the Reconstructed Results

In order to investigate the effect of noise level in the analyzing signal, signals with different levels of noise were prepared by adding random noise into the simulated chromatogram in Figure 2. Generally the noise in an HPLC data is heteroscedastic, but for simplicity and the reason that the type of noise should not influence the results of the method, random noise was adopted in this study. The intensity of the added noise in percentage of the intensity of the signal are, respectively, 5%, 10%, 20%, 30%, 40%, and 50%. Table 4 shows the fitness obtained with the 15 B2-spline based elementary functions, and figure 5 shows the comparison between the reconstructed chromatogram (with the fitness being 0.0377 and the RE being 0.1391) and simulated chromatogram with 20% noise. The dash line at the bottom of figure 5 is the residual which is not recorded in the reconstructed chromatogram, i.e., the noise filtered out by the method.

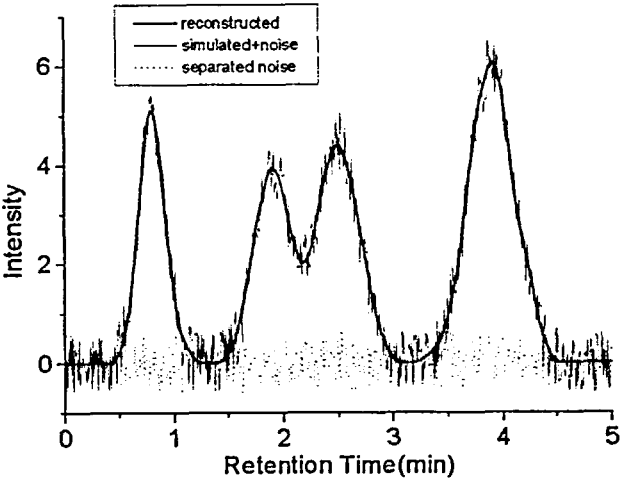

Figure 5 Comparison between the original and reconstructed chromatogram

From the table it is clear that not only did the fitness became larger and larger with the increase of noise level, which can be explained by the increase of the residual, the RE, which represents the difference between the reconstructed signal and the real analyzing signal, also increased. This indicates that the noise level will affect the results of the method. But from the values of RE in the table, it can be seen that the reconstructed signal will remain the main information of the analyzing signal. Figure 5 will also show us the conclusion.

Compression and De-nosing of Experimental Chromatograms

At first an experimental chromatogram with a great level of noise was investigated by the methods according to the parameters optimized above using 15 and 20 B2-spline based elementary functions. Table 5 shows the results and the comparison between the experimental chromatogram and the reconstructed chromatogram with fitness being 0.0708 from 20 functions is shown in figure 6. The dash line in the figure is the noise filtered out by the method. From the table

Table 5 Results obtained from the experimental chromatograms*

number of functions	compression ratio	best fitness	mean fitness
15	53.0:1	0.0721	0.0812
20	39.7:1	0.0708	0.0732

*:Results over 5 runs were sampled.

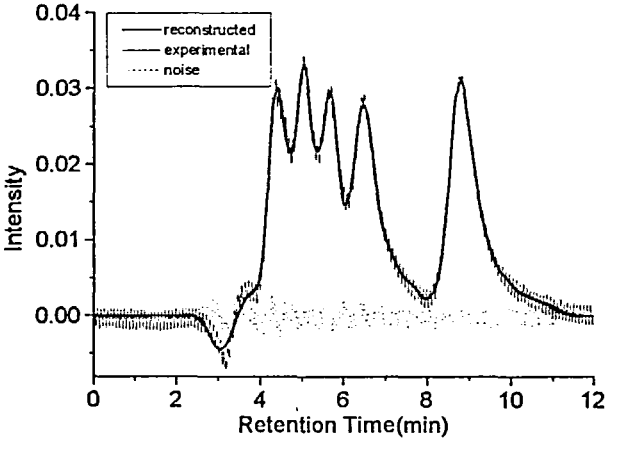

Figure 6 The experimental (2384 points) and the reconstructed chromatograms

and the figure, it can be seen that both the compression and the de-noising for the experimental chromatograms are satisfactory.

An experimental chromatogram with 15 peaks was also investigated by the method. Figure 7 shows the comparison between the experimental and the reconstructed chromatogram. The fitness of the reconstructed chromatogram is 0.0649. The compression ratio is 7.5:1 because 40 elementary functions are used in calculation. It is clear that all the peaks remained in the reconstructed chromatogram except for the small peaks caused by the injection disturbance.

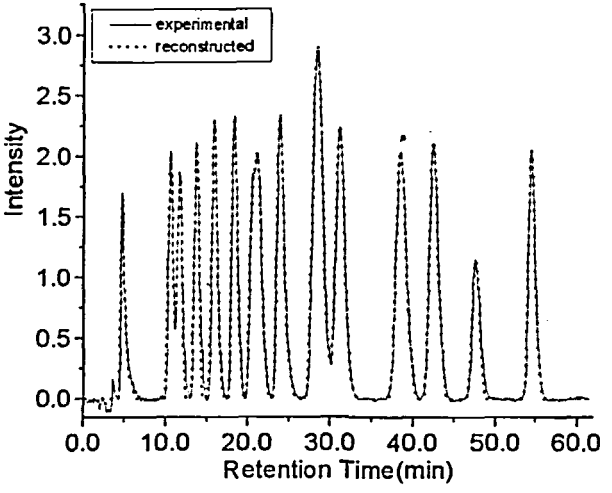

Figure 7 The experimental (900 points) and the reconstructed chromatograms

CONCLUSION

A wavelet-based genetic algorithm was developed to approximate analytical signals as a linear combination of wavelet-like elementary functions. Due to the characteristic of the wavelet-like functions, the method is able to represent signals in very high efficiency. Because only the significant information of the signals are recorded in the elementary functions, the method can also be used for de-noising. From the results of both simulated and experimental chromatograms, it was proven that the method can represent chromatographic signals by only a few elementary function. The method may be a high performance method for compression and de-noising of analytical signals.

ACKNOWLEDGEMENT

The work was supported by the National Natural Science Foundation of China (NSFC).

REFERENCES

1. M. Bos and J.A.M. Vrielink, *Chemom. Intell. Lab. Sys.*, **23**:115 (1994).
2. F.T. Chau, T.M. Shih, J.B. Gao and C.K. Chan, *Appl. Spectrosc.*, **50**:339 (1996).
3. V.J. Barclay, R.F. Bonner and I.P. Hamilton, *Anal. Chem.*, **69**:78 (1997).
4. Z. Pan, X. Shao and H. Zhong, et al, *Chinese J. Anal. Chem.*, **24**:149 (1996).
5. X. Shao, W. Cai, P. Sun, M. Zhang and G. Zhao, *Anal. Chem.*, **69**:1722 (1997).
6. J.H. Holland, *Adaptation in Natural and Artificial Systems.* University of Michigan Press: Ann Arbor MI, U.S.A. 1975
7. M. Srinivas and L.M. Patnaik, *IEEE Trans. System, Man Cybernetics*, **24**(4):656-667 (1994).
8. C.B. Lucasius and G. Kateman, *Chemom. Intell. Lab. Sys.*, **19**:1-33 (1993).
9. C.B. Lucasius and G. Kateman, *Chemom. Intell. Lab. Sys.*, **25**:99-145 (1994).
10. M.M. Lankhorst and M.D. van der Laan, Wavelet-based signal approximation with genetic algorithms, *lankhors@cs.rug.nl*
11. Z. Pan, X. Shao, H. Zhong, W. Liu, H. Wang and M. Zhang, *Chinese J. Anal. Chem.*, **24**(2):149-153 (1996).
12. X. Shao, W. Cai and P. Sun, *Chemom. Intell. Lab. Sys.*, **43**:147-155 (1998).

Received: January 27, 1999
Accepted: March 5, 1999