

---

# An Immunogenetic Approach to Spectra Recognition

---

**Dipankar Dasgupta**

Department of Mathematical Sciences  
The University of Memphis  
Memphis, TN 38152

**Yuehua Cao**

Departments of Mathematical  
Sciences and Chemistry  
The University of Memphis  
Memphis, TN 38152

**Congjun Yang**

Department of Mathematical Sciences  
The University of Memphis  
Memphis, TN 38152

## Abstract

The paper describes an immunogenetic approach to recognize spectra for chemical analysis. In particular, an immunological model for chemical reactions is introduced in which a population of specialists for each of the possible products was evolved using a genetic algorithm. Accordingly, a small well-trained specialist library is established and tested their recognition ability with real dataset (Raman Spectra). Our experiments produced very encouraging results in finding the correct products responsible for an input spectrum, epecially, for a composite spectrum in which there are multiple products physically mixed and it would be very difficult to interpret otherwise.

## 1. INTRODUCTION

The natural immune system protects the body from a large variety of bacteria, viruses, and other pathogenic organisms. It recognizes foreign cells and molecules by producing antibody molecules that physically bind with antigens (or antigenic peptides). In order for the antigen and antibody molecules to bind, their three-dimensional shapes must match in a lock-and-key manner. For every antigen, the immune system must be able to produce a corresponding antibody molecule, so that the antigen can be recognized and defended against. The antibody, therefore, can have a geometry that is specific to a particular antigen (specialist) or is capable of partial matching and capturing of a broad group of antigens (generalist). The primary role of this defense mechanism is to distinguish between the self (body cells and tissues) and the non-self (antigens). This discrimination is achieved in part by T-cells, which have receptors on their surface that can detect foreign proteins (antigens). During the generation of T cells, their receptors are evolved (from gene libraries) through a pseudo-random genetic rearrangement process. Then they undergo a censoring process, called negative selection, in the *thymus* where T

cells that react against self-proteins are destroyed, so only those that do not bind to self-proteins are allowed to leave the thymus. These matured T cells then circulate throughout the body to perform immunological functions to protect against foreign antigens. Moreover, it continually evolves such immune cells and other antibody molecules (in right proportion) in order to defend the body.

These immunological mechanisms have inspired the development of several computational models [4]. A brief survey of some of these models may be found elsewhere [5]. Forrest et al. [9] developed a negative-selection algorithm for change detection based on the principles of self-nonself discrimination. This algorithm works on similar principles, generating detectors randomly, and eliminating the ones that detect self, so that the remaining T-cells can detect any non-self. This self and non-self (computational) algorithm, the representative of a two-component model, appears to be very useful in many applications [6], but is not adequate for applications with multiple classes involved, of which each requires to be uniquely recognized.

The researchers have also been studying immunogenetic approaches (evolving antibodies using genetic algorithms) for more than a decade [4, 10]. Farmer et al. [7] compared the immune system with learning classifier systems. Bersini and Varela [1] used the recruitment mechanism of the immune system to accelerate the parallel and local hill climbing. In particular, they developed an IRM (Immune Recruitment Mechanism) and GIRM (Genetic IRM) to recruit a candidate from a certain population in the shape space. There exist other computation models emulating different immunological principles, for example, its ability to detect common patterns in a noisy environment [8], its ability to discover and maintain coverage of diverse pattern classes [19], and its ability to learn effectively, even when not all antibodies are expressed and not all antigens are presented [15]. In some studies, genetic algorithms have been used to model somatic mutation -- the process by which antibodies are evolved to recognize a specific antigen [16]. Hajela [12,13]

recently used a genetic search for immune network design in solving structural optimization problems. Other researchers investigated artificial immune systems for scheduling [11, 14]. Potter and De Jong [18] reported a method for concept learning in which a coevolutionary genetic algorithm was applied to the construction of an immune system whose antibodies can discriminate between examples and counter-examples of a given concept.

In this paper, we describe the use of an immunogenetic approach in the interpretation of chemical spectra. In section 2.1 some basic spectroscopic knowledge required to understand this method is introduced. The spectrum representation, specialist evolution and spectrum recognition are described in detail at the rest part of Section 2. In Section 3 the model is tested on a set of real-world problems and the results are presented and analyzed. Section 4 gives some concluding remarks and directions for future work.

## 2. THE PROBLEM AND THE PROPOSED APPROACH

Interpretation of a composite spectrum (like IR, UV-Visible, Raman, Mass, etc.) has been a difficult and very time-consuming task for chemists. This work has conventionally been performed manually with limited accuracy. Molecular calculations have provided some confirmatory information to aid the interpretation, but the situation did not improve much. There is a report in which neural network was attempted for rapid screening of large infrared spectral databases, where only spectra of pure chemicals were involved [17].

### 2.1 DESCRIPTION OF A BAND IN A SPECTRUM

Absorption spectra are plots representing the absorbance (A) or transmittance (T) as a function of frequency (or, more specifically, wavenumbers in  $\text{cm}^{-1}$ ) of the recorded radiation. Spectra are made up of a collection of bands deriving from the fundamental tones, combined tones and overtones, related to the normal vibrations of a molecule. Each band of a spectrum is characterized by the following parameters:

- (a) The position of the band maximum, most frequently expressed in wavenumbers,  $\nu$ ;
- (b) The intensity of the band:
  - (i) at the maximum,  $I_{\text{max}}$  ( $A_{\text{max}}$ ),
  - (ii) the integrated intensity  $I_{\text{int}}$  (absorption  $A_{\text{int}}$ ):

$$I_{\text{int}} = \int_{-\infty}^{+\infty} I(\nu) d\nu$$

- (c) The band half-width  $\rho_{\nu/2}$

The position of the band maximum ( $\nu_{\text{max}}$ ) is the most significant parameter, because it yields information on the frequency and hence the type of vibration. The above

information is introduced using absorption spectrum as an example, but it actually applies to almost all the spectroscopy family, including scattering spectrum and mass spectrum (trivially). So in this study, as an initial work in this field, we will use the positions of the band maxima to represent a spectrum. Other information such as the intensity of bands can easily be attached by a slight modification to the data structure. An example of a Raman spectrum [2] is shown in figure 1.

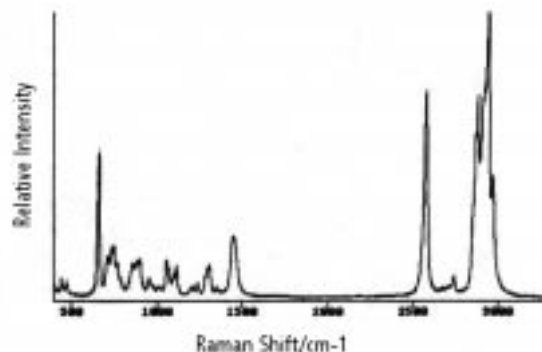


Figure 1: A sample Raman spectrum of 1-butanethiol.

### 2.2 REPRESENTATION OF THE SPECTRUM

Each spectrum is represented with a binary string where each bit in the string corresponds to a peak occurrence within an equal length of wavenumbers. The value of the bit is determined by the signal received at the detector: either 1 if there is a peak at that region of wavenumber or 0 if not. If the spectra domain has  $n$  wavenumbers and we represent the spectrum with a string of  $m$  bits, each bit has a coverage of  $n/m$  wavenumbers. In this bitstring universe, recognition takes place when the (antibody) bitstring and the (antigen) bitstring “match” each other as will be explained later. Using this representation, the above Raman spectroscopy can be expressed as the bit string as shown in figure 2:

- (a) The first half of the string ( $400 \text{ cm}^{-1} - 1700 \text{ cm}^{-1}$ )

1	0	1	0	1	1	1	0	1	0	1	0	0
.5	-.5	.5	-.5	.5	.5	.5	-.5	.5	-.5	.5	-.5	-.5

- (b) The second half of the string ( $1700 \text{ cm}^{-1} - 3000 \text{ cm}^{-1}$ )

0	0	0	0	0	0	0	0	1	0	1	1	1
-.5	-.5	-.5	-.5	-.5	-.5	-.5	-.5	1.5	-.5	.5	.5	.5

Figure 2 A binary representation of the spectrum displayed in Figure 1. A bit value ‘1’ means a peak occurrence and ‘0’ otherwise. The weight associated with each bit given under the string indicates peak properties.

In this work, the above string representation is used to define the immunological terms in the following manner:

**Self:** a set of starting materials **R** before chemical or photochemical reactions.

**Non-self:** a set of products due to reactions.

**Antigen:** any of the products **P**.

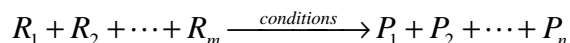
**Antibody:** any evolved population which uniquely recognize one and only one of the products, and it should have no response to the spectrum of the starting materials **R** or other products.

**Matching:** an antigen and antibody are said to “match” (in hamming space) if the similarity between the antigen and antibody string exceeds the set threshold **T**.

**Matching-function:** a function **f** to measure how well two spectra match.

### 2.3 THE SPECIALIST EVOLUTION

A general form of a chemical reaction looks like this:



Where  $R_i$  ( $1 \leq i \leq m$ ) are reactants and  $P_j$  ( $1 \leq j \leq n$ ) are all the possible products. Each reactant or product has a specific spectrum that identifies it. By using binary representation introduced above, each of the reactants  $R_i$  and products  $P_j$  is encoded into a unique string.

The next step is to evolve a population of specialists for each of the products. To do this, we are using product  $P_k$  ( $1 \leq k \leq n$ ) as an antigen, exposing it to a randomly generated initial population, and then keeping only those which match  $P_k$  very well and eliminating the rest population. We call the population matching  $P_k$  very well as the pre-antibody  $\sigma_{pre}$ . The reason for this name lie in the fact that some members of this population may also match a reactant string or other product strings as well. To uniquely recognize a product antigen, it is necessary to expose the evolved antibodies to the reactants and other product environment. So we need to put this population into a pool consisting of  $\{R_i | 1 \leq i \leq m\} \cup \{P_j | 1 \leq j \leq n, j \neq k\}$  for purification, this time we only want to keep those unmatched strings  $S_k$ , which are the trained specialist uniquely recognizing  $P_k$ . Those strings whose matching function value exceeds the threshold should be removed from the population of antibodies. Using the similar censoring approach as the nature does (T-cell maturation), we could evolve a population of specialists for every product.

In our implementation, the determination of initial population size is based upon the actual necessity. Genetic operators (crossovers and mutations) are applied to the population in the usual way to generate a

population of better fitness. Elitism is used to preserve good individuals and keep sufficient diversity in the next generation. The number of generations depends on the real-valued activation threshold that represents the extent of similarity required to initiate an immune response (positive product recognition). Rather than simply using the number of matching bits as the fitness function, we assigned a weight to each bit, with the spectroscopic band  $b_k$  which characteristically defines  $P_k$  being assigned a significant higher weight  $W_k$  (see Figure 2). In general, to initialize an immune response, the matching function must satisfy:

$$F = \sum_{i=1}^n b_i W_i(h_i, c_i) \geq T$$

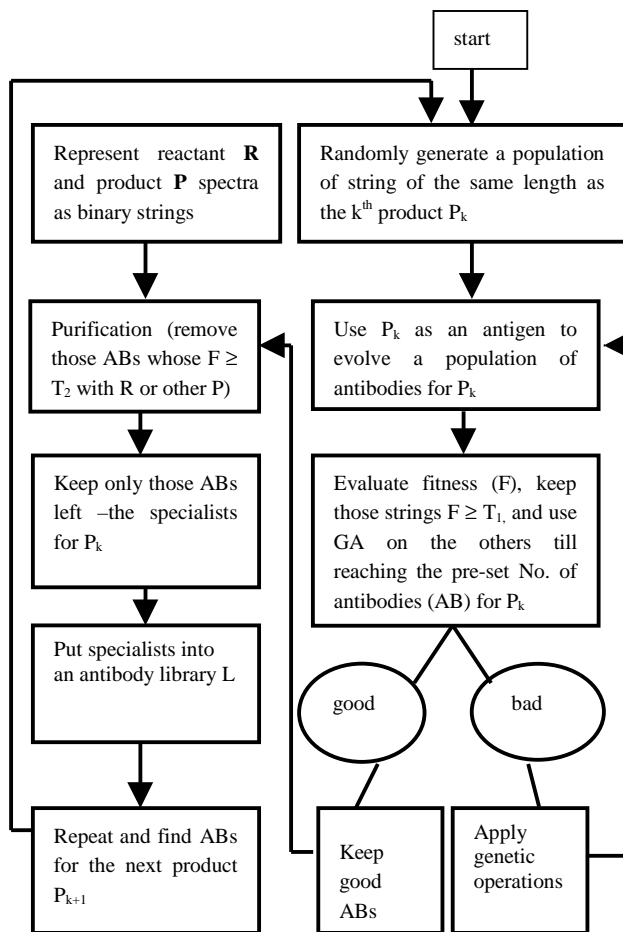
Where,  $b_i$  is the  $i^{th}$  bit value of the  $n$  bits in the string, either 0 when no peak at that interval or 1 when there is a peak at that interval;  $T$  is the set threshold for initiating an immune response;  $W_i(h_i, c_i)$  is the weight of the  $i^{th}$  bit, also a function of both the presence of a peak  $h_i$  (0.5 if there is also a peak in  $P_k$  at this position and  $-0.5$  otherwise) and characteristic property ( $c_i = 1$  if yes,  $c_i = 0$  if not)

$$W_i(h_i, c_i) = h_i + c_i$$

The inclusion of negative values in the domain of  $h_i$  allows the matching function to take into consideration a penalty for the appearance of unexpected peaks, thus populations with peaks at unwanted position are not encouraged during evolution. This could avoid the occurrence of false positive.

Next, we purposefully set the value for a characteristic peak twice more important than the presence of a normal peak. A typical example is the peak at  $2578 \text{ cm}^{-1}$  as shown in Figure 1, which is attributed to the S-H vibration and safely identifies that free thiol molecules are contained in the compound(s) responsible for this spectrum [3]. This is a strategy to converge the population towards having the preference to include this peak in their strings (this is a desirable property) during evolution and preserve a sufficient population of antibodies for a particular product. Last but not least important, the appropriate choosing of the threshold value provides this model a noise-tolerant feature. As we know, in spectroscopy some effects like the random noise and baseline fluctuations can be eliminated from the input data, but other effects like the frequency shift (usually small) and bandwidth variation can not be canceled from the experimental spectra and make the assignment troublesome. These problems, however, can be easily resolved by choosing an appropriate threshold. It is also a fundamental advantage of a genetic algorithm (GA) over deterministic methods. Figure 3 shows the algorithmic

steps schematically in evolving specialists for product identification.



**Figure 3:** A Flow chart illustrating the proposed immunogenetic approach for evolving specialists.

A library of specialists is then created to perform the central administration of spectrum recognition, to which specialists of each product are added. Extra power is introduced by admitting specialists for a product having NOT been encountered. These specialists are functionally similar to the innate antibodies of human being. The more comprehensive or the more diverse the library, the more powerful it will be in performing the recognition task.

### 2.4 SPECTRUM RECOGNITION

The efforts in the previous sections are aimed to establish a specialist library. In this section we are more concerned with the utilization of this library. The recognition in this application is the automatic process to find all possible products responsible for the observed spectrum, analog to the antibody's recognizing antigen in the natural immune system. There are situations where a spectrum obtained at

certain condition may not be any of the known products. In this case this model will treat it as a new antigen, and then follow the same algorithm to evolve a population of specialists for it (as given in figure 1). In general, if there are  $m$  such unknown spectra, the  $i^{th}$  will be named as unknown species- $i$  ( $1 \leq i \leq m$ ).

The values of weights in matching function used in the recognition phase should be of difference from the one used during evolution (see the weight function in section 2.3). Modification to the matching function is the removal of the penalty of an unexpected peak on a spectrum. It is especially necessary for a composite spectrum, which could be recognized by multiple different antibodies. In particular, here,  $h_i = 0.5$  if there is also a peak in  $P_k$  at this position and  $h_i = 0$  (instead of -0.5 in the evolution phase) otherwise.

The following is the proposed algorithm for the spectrum recognition:

---

#### Algorithm for Recognition (NewStr)

---

- 0: Input string NewStr  $S$ ;
  - 1: Expose  $S$  to the specialist library  $L$ ;
  - 2: Find all those antibodies ( $AB$ ) whose binding energy  $F_2$  with  $S$  exceeds  $T_2$  (the recognition threshold);
  - 3: Check whether there exist unassigned peaks  $(S - \Sigma AB)$ ;
  - 4: If yes, name it unknown- $j$  ( $U_j$ ), and evolve a population of  $AB$  for  $U_j$ , and then add these  $AB$  of  $U_j$  to  $L$ ;
  - 5: Return  $AB_i$  and  $U_j$ .
- 

With of the establishment of the specialist library, the recognition capacity of this approach increases with the size the library. Whenever a product is recognized for the first time, a copy of it is reserved as a new specialist for that product. Therefore, when it appears a second time, it can be easily recognized by the antibodies created during the first appearance. Consequently, this approach provides a learning methodology for pattern recognition and its memory capability grows more and more powerful with its increasing recognition experiences.

### 3. EXPERIMENTAL DETAILS & RESULTS

Without lose generality, we used Raman spectra for the experiment simply for its availability in our laboratory,

popularity worldwide and maturity as a powerful analytical tool. Five known composite spectra were used as antigens to test whether and how well they can be recognized. There are a total of 20 different product spectra collected [3], of which the five mixtures are composed. Each product maintains a certain number of specialists ( $\sigma_{AB}$ ). No matter which one of these specialists has a positive reaction towards the input spectrum, it will return the same molecular formula. The number of different formula returned represents the number of different products responsible for the spectrum. In this section, we give experimental results using several parameters including  $T_1$  (threshold during evolution),  $T_2$  (threshold during recognition), and the number of antibodies (specialists) maintained for each product spectrum. Table 1 shows these 20 products as well as their peak densities that are used in the experiments.

**Table 1** Information for 20 products that are used in our experiments to form different mixtures.

Product	Molecular Formula	Peak Density (%)
1	(CH <sub>3</sub> ) <sub>2</sub> SO	3.00
2	C <sub>6</sub> H <sub>5</sub> CH <sub>2</sub> CH <sub>2</sub> SH	7.00
3	(C <sub>2</sub> H <sub>5</sub> ) <sub>2</sub> O	3.00
4	CCl <sub>4</sub>	1.33
5	HS(CH <sub>2</sub> ) <sub>4</sub> SH	5.00
6	CH <sub>3</sub> COCH <sub>3</sub>	4.67
7	C <sub>6</sub> H <sub>5</sub> COOH	4.67
8	CH <sub>3</sub> CH <sub>2</sub> OH	3.33
9	COOH CH <sub>2</sub> SH	4.67
10	O-NH <sub>2</sub> C <sub>6</sub> H <sub>5</sub> SH	7.00
11	C <sub>4</sub> H <sub>9</sub> SO <sub>3</sub>	5.67
12	C <sub>6</sub> H <sub>5</sub> SO <sub>3</sub>	4.33
13	CH <sub>3</sub> (CH <sub>2</sub> ) <sub>17</sub> SH	5.00
14	CH <sub>3</sub> (CH <sub>2</sub> ) <sub>2</sub> SH	7.00
15	trans-FC(O)SCH <sub>3</sub>	3.33
16	(CF <sub>3</sub> ) <sub>2</sub> C=NH	6.33
17	NaCOO(OH)CHCH(OH)COOK	6.67
18	C <sub>6</sub> H <sub>5</sub> S CH <sub>3</sub>	7.67
19	Cl CH <sub>2</sub> COOC <sub>2</sub> H <sub>5</sub>	13.7
20	F CH <sub>2</sub> CONH <sub>2</sub>	4.67

In our experiment, each bit represents 10 cm<sup>-1</sup> because it would be very occasional that two peaks occur within 10

cm<sup>-1</sup>. Note that the peak density is calculated using the number of peaks to divide the total number of intervals. We first investigated the effect of the number of specialists (antibodies) maintained for each product on recognition, Table 2 outlines the results of our experiments with various specialist sets,  $\sigma_{AB} = 3, 5, \text{ and } 10$ , respectively. The second column shows the formation of the mixtures using different products given in Table 1. The rest three columns show the products found in the mixture by our method. Obviously, the output quickly approaches the true value when  $\sigma_{AB}$  is increased from 3 to 5. When  $\sigma_{AB} = 10$ , all five mixtures are recognized with 100% accuracy, so other tests followed will adapt this setting. It is noted that the smaller the  $\sigma_{AB}$ , the fewer the number of products recognized. It is particularly exemplified by the fact that nothing returned when  $\sigma_{AB} = 3$  for mixture C.

**Table 2** Effect of the size of specialist set,  $\sigma_{AB}$  on the performance of this model. Here  $T_1 = 0.9, T_2 = 0.99$ .

Mixture	Actual composition	Products found in the composition		
		$\sigma_{AB} = 3$	$\sigma_{AB} = 5$	$\sigma_{AB} = 10$
A	8 and 20	8	8, 20	8, 20
B	1, 4 and 16	4	1, 4, 16	1, 4, 16
C	1, 13 and 15	null	1, 15	1, 13, 15
D	3, 4, 6 and 7	3, 4	3, 4	3, 4, 6, 7
E	4, 8, 10, 11 and 12	4	4, 8, 11, 12	4, 8, 10, 11, 12

Table 3 shows the results of each of the five tests and compares them with the actual chemical composition. Clearly  $T_2$  is very critical for the correct recognition. When  $T_2 = 0.99$ , the program correctly recognizes all the five mixture spectra. However, with further increasing of  $T_2$  to 0.999, some (more than half in most cases) possible products are ruled out. It means that this threshold is too high to find all the products, but it is useful when someone wants to know which product is exactly involved. On the other hand, when  $T_2$  is decreased to 0.95, wrong results began to appear. It is found that most of the mis-identified products share a common characteristic, i.e., their peak densities are relatively low. Normally, if the peak density of a product spectrum is less than 4.67, as in this example, it has a better chance to be positive toward an input spectrum (antigen) if  $T_2 \leq 0.95$ . We also tried the situation when  $T_2 = 0.90$ , in which not

only there were more false positives, but the results changed even for the same mixture between two runs. Overall, 0.99 is a nicely trained value for  $T_2$ .

**Table 3** Effect of the recognition threshold ( $T_2$ ) on the performance of this model.  $T_1 = 0.9$  and  $\sigma_{AB} = 10$ ; the underlined products indicates false positives.

Mixture	Actual composition	Products found		
		$T_2 = 0.99$	$T_2 = 0.999$	$T_2 = 0.95$
A	8 and 20	8, 20	8	8, 20, <u>1</u> , <u>3</u> , <u>4</u>
B	1, 4 and 16	1, 4, 16	4	1, 4, 16, <u>3</u>
C	1, 13 and 15	1, 13, 15	15	1, 13, 15, <u>3</u> , <u>4</u> , <u>8</u>
D	3, 4, 6 and 7	3, 4, 6, 7	3, 4, 6, 7	3, 4, 6, 7, <u>1</u> , <u>8</u> , <u>12</u>
E	4, 8, 10, 11 and 12	4, 8, 10, 11, 12	4, 8	4, 8, 10, 11, 12, <u>1</u> , <u>3</u> , <u>7</u>

While the performance of this method is very sensitive to  $T_2$ ,  $T_1$  behaves otherwise. The final set of experiments shown in Table 4 compares the results for different  $T_1$  values, while keeping  $T_2$  constant at 0.95. It is observed that no significant improvement was achieved as regard to the accuracy of the outputs when  $T_1$  changes from 0.9 up to 0.999. The number of false positive remained unchanged even when  $T_1$  varied from 0.99 to 0.999. These results verify our arguments above.

#### 4. CONCLUSIONS

The natural immune system uses learning, memory, and associative retrieval to solve recognition and classification tasks. Its learning takes place through recruitment mechanism which is partly an evolutionary process similar to the biological evolution. Various recognition and response mechanisms of the immune system have inspired the development of some useful computational models. This paper introduces an immunogenetic approach for the detection of products from an input spectrum with adjustable confidence. It is particularly useful in identifying compositions from chemical spectra, which has been a difficult task for chemists. Compared with deterministic spectrum detection approaches, this

method is more flexible and noise-tolerant. However, the capability of this model cannot function beyond the spectroscopy itself, and the combination with other measurements is not only helpful, but sometimes is also necessary, to clench the exact products. We showed that with a well-established specialist library and a well-chosen threshold, our approach could find all the possible products responsible for an input spectrum just within 1 second.

**Table 4.** Variation of the threshold ( $T_1$ ) during the evolution of specialists on the performance of this method.  $T_2 = 0.95$  and  $\sigma_{AB} = 10$ ; the underlined are false positives.

Mixture	Actual composition	Product found		
		$T_1 = 0.99$	$T_1 = 0.999$	$T_1 = 0.95$
A	8 and 20	8, 20	8	8, 20, <u>1</u> , <u>3</u> , <u>4</u>
B	1, 4 and 16	1, 4, 16	4	1, 4, 16, <u>3</u>
C	1, 13 and 15	1, 13, 15	15	1, 13, 15, <u>3</u> , <u>4</u> , <u>8</u>
D	3, 4, 6 and 7	3, 4, 6, 7	3, 4, 6, 7	3, 4, 6, 7, <u>1</u> , <u>8</u> , <u>12</u>
E	4, 8, 10, 11 and 12	4, 8, 10, 11, 12	4, 8	4, 8, 10, 11, 12, <u>1</u> , <u>3</u> , <u>7</u>

The utility of this model is expected to safely extend to other spectra, such as IR, UV-visible and Mass spectroscopy. In a bulk solution, Using transmission spectroscopic techniques, it can be programmed for automatic product detection in organic synthesis, which facilitates the optimization of reaction conditions towards the best possible yield based on the in-situ product makeup detection mechanism. However, for surface spectra (e.g. SERS, SERI) [20], this model should be equally useful.

#### Future work:

This work for spectrum representation is a simplified abstraction of the spectrum in the real world. The inclusion of the additional spectroscopic information like peak area in the data structure will definitely increase its ability of discrimination. For example, using integer representation instead of binary representation will allow the peak intensity information to be considered. It is also true for some other spectroscopic considerations. To be practically useful, a comprehensive collection of

specialists for a broad range of chemicals should be generated, and training and testing on some of them should be applied to work out an appropriate threshold. Further work will also study the antigenic feature extraction properties of the natural immune system to develop an improved pattern recognition methodology.

## REFERENCES

1. H. Bersini and F. J. Varela. The immune recruitment mechanism: A selective evolutionary strategy. *In proceedings of the fourth International Conference on Genetic Algorithms*, pages 520--526, San Diego, July 13-16 1991.
2. Y. Cao and Y. S. Li, Constructing Surface Roughness of Silver for Surface Enhanced Raman Scattering by Self-assembled Monolayers and Selective Etching Process. *Appl. Spectrosc.*, in press.
3. Y. Cao and Y. S. Li, Spectra were collected with a spectrometer system consisting of a Spex Model 1403 double monochromator and a Hamamatsu R928-07 photomultiplier tube (PMT) held at  $-30^{\circ}\text{C}$  by a thermoelectrically refrigerated chamber (Product for Research, Model TE 117-RF). A Lexel 3000 laser ( $\text{Ar}^+$ ) equipped with a Spex Model 1405 tunable filter was used for sample excitation at 514.5 nm. A bandpass  $4\text{ cm}^{-1}$  was set for all the experiments (unpublished).
4. D. Dasgupta (editor). *Artificial Immune Systems and Their Applications*, Springer-Verlag, 1999.
5. D. Dasgupta and N. Attoh-Okine. Immunity-based systems: A survey. *In proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, pages 363--374, Orlando, Florida, October 12-15 1997.
6. D. Dasgupta and S. Forrest. Novelty Detection in Time Series Data using Ideas from Immunology. *In proceedings of the 5<sup>th</sup> International Conference on Intelligent Systems*, Reno, June 19-21, 1996.
7. J. D. Farmer, N. H. Packard, and A. S. Perelson. The immune system, adaptation, and machine learning. *In Physica D*, 22:187--204, 1986.
8. S. Forrest, B. Javornik, R. Smith, and A. S. Perelson. Using genetic algorithms to explore pattern recognition in the immune system. *Evolutionary Computation*, 1(3):191--211, 1993.
9. S. Forrest, A. S. Perelson, L. Allen, and R. Cherukuri. Self-Nonself Discrimination in a Computer. *In proceedings of the IEEE Symposium on Research in Security and Privacy*, pages 202--212, Oakland, CA, 16-18 May 1994.
10. S. Forrest and A. S. Perelson. Genetic algorithms and the immune system. *In proceedings of the Parallel Problem Solving from Nature*, Springer-Verlag, Berlin (Lecture Notes in Computer Science), 1991.
11. T. Fukuda, K. Mori and M. Tsukiyama. Immune Networks using Genetic Algorithm for Adaptive Production Scheduling. *In 15th IFAC World Congress*, Vol.3, pp.57--60, 1993.
12. P. Hajela and J. Lee. Constrained Genetic Search Via Schema Adaptation: An Immune Network Solution. *Structural Optimization*, Vol 12, No. 1, pp. 11-15, 1996.
13. P. Hajela, J. Yoo and J. Lee. GA Based Simulation of Immune Networks - Applications in Structural Optimization. *In Journal of Engineering Optimization*, 1997.
14. E. Hart, P. Ross and J. Nelson. Producing robust schedules via an Artificial Immune system. *In the proceedings of the IEEE International Conference on Evolutionary Computation*, 1998.
15. R. Hightower, S. Forrest, and A.S. Perelson. The evolution of emergent organization in immune system gene libraries. *In proceedings of the Sixth International Conference on Genetic Algorithms*, Pittsburg, 1995.
16. R. Hightower, S. Forrest, and A.S. Perelson. The Baldwin effect in the immune system: learning by somatic hypermutation," *In Adaptive Individuals in Evolving Populations*, Addison-Wesley, Reading, MA, pp. 159-167 (1996).
17. C. Klawun and C. L. Wilkins. Neural Network Assisted Rapid Screening of Large Infrared Spectral Databases *Anal. Chem.* 67, 374, 1995.
18. M. A. Potter and K. A. De Jong. The Coevolution of Antibodies for Concept Learning. *In proceeding of the Parallel Problem Solving from Nature (PPSN)*, Amsterdam, 1998.
19. R. E. Smith, S. Forrest, and A. S. Perelson. Searching for diverse, cooperative populations with genetic algorithms. *In Evolutionary Computation*, 1:2, pp. 127-149, 1993.
20. X. M. Yang, D. A. Tryk, K. Hashimoto and A. Fujishima, Examination of the Photoreaction of p-Nitrobenzoic Acid on Electrochemically Roughened Silver using Surface-Enhanced Raman Imaging (SERI) *J. Phys. Chem. B* 102, 4933, 1998.

