

# Parallel cascade identification as a means for automatically classifying protein sequences into structure/function groups

Michael Korenberg<sup>1</sup>, Jerry E. Solomon<sup>2</sup>, Moira E. Regelson<sup>2</sup>

<sup>1</sup> Department of Electrical and Computer Engineering, Queen's University, Kingston, Ontario, K7L 3N6, Canada

<sup>2</sup> Center for Computational Biology, The Beckman Institute, California Institute of Technology, Pasadena, CA 91125

Received: 16 March 1998 / Accepted in revised form: 2 June 1999

**Abstract.** Current methods for automatically classifying protein sequences into structure/function groups, based on their hydrophobicity profiles, have typically required large training sets. The most successful of these methods are based on hidden Markov models, but may require hundreds of exemplars for training in order to obtain consistent results. In this paper, we describe a new approach, based on nonlinear system identification, which appears to require little training data to achieve highly promising results.

## 1 Introduction

Stochastic modeling of hydrophobicity profiles of protein sequences has led to methods being proposed for automatically classifying the sequences into structure/function groups (Baldi et al. 1994; Krogh et al. 1994; Regelson 1997; Stultz et al. 1993; White et al. 1994). Various linear modeling techniques for protein sequence classification, including a Fourier domain approach (McLachlan 1993), have been suggested but most have not shown impressive performance in experimental trials (about 70% in two-way classifications). A hidden Markov modeling approach (Baldi et al. 1994; Krogh et al. 1994; Regelson 1997) has been more effective in classifying protein sequences, but its performance may depend on the availability of large training sets and require a lengthy training time. This is because thousands of parameters might be incorporated into each hidden Markov model to obtain reasonable classification accuracy (Baldi et al. 1994; Regelson 1997). Hence, a potential drawback of the hidden Markov modeling approach to classifying proteins is the possible need of using large training sets (i.e., hundreds of exemplars) in order to obtain consistent results (Regelson 1997).

Here, in a pilot study, we propose classifying protein sequences by means of a technique for modeling dynamic nonlinear systems, known as parallel cascade identification (Korenberg 1991), and show that it is capable of highly accurate performance in experimental trials. This method appears to be equally or more discriminating than other techniques (Krogh et al. 1994; McLachlan 1993; Regelson 1997; Stultz et al. 1993; White et al. 1994) when the size of the training sets is limited. Remarkably, when only a *single* sequence from each of the globin, calcium-binding, and kinase families was used to train the parallel cascade models, an overall two-way classification accuracy of about 79% was obtained in classifying novel sequences from the families.

This paper is addressed to managers of large protein databases to inform them of an emerging methodology for automatic classification of protein sequences. It is believed that the proposed method can usefully be employed to supplement currently used classification techniques, such as those based on hidden Markov models. This is because, as detailed below, the new method appears to require only a few representative protein sequences from families to be distinguished in order to construct effective classifiers. Hence, for each classification task, the accuracy may be enhanced by building numerous classifiers, each trained on different exemplars from the protein families, and have these classifiers vote on new classification decisions (see Discussion). Because the proposed method appears to be effective while differing considerably from that of hidden Markov models, there are likely benefits from employing them together. For example, when the two methods reach the same classification decision, there may well be increased confidence in the result.

It is also hoped that individual scientists involved in various aspects of protein research will be interested in the new approach to automatically classifying protein sequences. For this reason, some detail is provided about the parallel cascade identification method, and also the construction of one particular protein sequence classifier (globin versus calcium-binding) is elaborated upon as an example.

Correspondence to: M. Korenberg  
(e-mail: korenber@post.queensu.ca  
Tel.: +1-613-5452931, Fax: +1-613-5456615)

## 2 System and methods

For managers of large protein databases, discussion of the modest requirements of computer memory and processing time is not crucial. However, the individual researcher who might wish to investigate this approach further may be interested in knowing that the protein sequence classification algorithm was implemented (in Turbo Basic source code) on a 90-MHz Pentium. Only 640 kilo-bytes of local memory (RAM) are actually required for efficient operation. Training times were only a few seconds while subsequent classification of a new protein sequence could be made in a fraction of a second.

Our data consisted of hydrophobicity profiles of sequences from globin, calcium-binding protein, and protein kinase families. The particular mapping of amino acid to hydrophobicity utilized is the ‘‘Rose’’ scale shown in Table 3 of Cornette et al. (1987), and the resulting profiles were not normalized. The protein sequences were taken from the Brookhaven Protein Data Base of known protein structures.

### 2.1 Algorithm description

Consider a discrete-time dynamic (i.e., has memory) nonlinear system described as a ‘‘black box’’, so that the only available information about the system is its input  $x(i)$  and output  $y(i)$ ,  $i = 0, \dots, I$ . Parallel cascade identification (Korenberg 1991) is a method for constructing an approximation, to an arbitrary degree of accuracy, of the system’s input/output relation using a sum of cascaded elements, when the system has a Wiener or Volterra functional expansion. Each cascade path comprises alternating dynamic linear and static nonlinear elements, and the parallel array can be built up one cascade at a time in the following way.

A first cascade of dynamic linear and static nonlinear elements is found to approximate the input/output relation of the nonlinear system to be identified. The residue – i.e., the difference between the system and the cascade outputs – is treated as the output of a new dynamic nonlinear system, and a second cascade is found to approximate the latter system. The new residue is computed, a third cascade can be found to improve the approximation, and so on. These cascades are found in such a way as to drive the input/residue crosscorrelations to zero. It can be shown (Korenberg 1991) that any nonlinear system having a Volterra or Wiener functional expansion (Wiener 1958) can be approximated to an arbitrary degree of accuracy in the mean-square sense by a sum of a sufficient number of the cascades. For generality of approximation, it suffices if each cascade comprises a dynamic linear element followed by a static nonlinearity, and this cascade structure was used in the present work. However, additional alternating dynamic linear and static nonlinear elements could optionally be inserted into any cascade path.

If  $y_k(i)$  denotes the residue after adding the  $k$ th cascade, then for  $k \geq 1$ ,

$$y_k(i) = y_{k-1}(i) - z_k(i) , \quad (1)$$

where  $y_0(i) = y(i)$ . The parallel cascade output,  $z(i)$ , will be the sum of the individual cascade outputs  $z_k(i)$ . The (discrete) impulse response function of the dynamic linear element beginning each cascade can, optionally, be defined using a first-order (or a slice of a higher-order) crosscorrelation of the input with the latest residue (discrete impulses  $\delta$  are added at diagonal values when higher-order crosscorrelations are utilized). When constructing the  $k$ th cascade, note that the latest residue is  $y_{k-1}(i)$ . Thus, the impulse response function  $h_k(j)$ ,  $j = 0, \dots, R$ , for the linear element beginning the  $k$ th cascade will have the form

$$h_k(j) = \phi_{xy_{k-1}}(j) , \quad (2)$$

if the first-order input/residue crosscorrelation  $\phi_{xy_{k-1}}$  is used, or

$$h_k(j) = \phi_{xy_{k-1}}(j, A) \pm C\delta(j - A) , \quad (3)$$

if the second-order crosscorrelation  $\phi_{xy_{k-1}}$  is used, and similarly if a higher-order crosscorrelation is instead employed (Korenberg 1991). In (3), the discrete impulse function  $\delta(j - A) = 1$  if  $j = A$ , and equals 0 otherwise. If (3) is used, then  $A$  is fixed at one of the values  $0, \dots, R$ , and  $C$  is adjusted to tend to zero as the mean-square of the residue approaches zero. The linear element’s output is

$$u_k(i) = \sum_{j=0}^R h_k(j)x(i - j) . \quad (4)$$

Next, the static nonlinearity, in the form of a polynomial, can be best-fitted, in the least-square sense, to the residue  $y_{k-1}(i)$ . If a higher-degree (say,  $\geq 5$ ) polynomial is to be best-fitted, then for increased accuracy scale the linear element so that its output,  $u_k(i)$ , which is the input to the polynomial, has unity mean-square. If  $D$  is the degree of the polynomial, then the output of the static nonlinearity, and hence the cascade output, has the form

$$z_k(i) = \sum_{d=0}^D a_{kd}u_k^d(i) . \quad (5)$$

The new residue is then calculated from (1). Since the polynomial in (5) was least-square fitted to the residue  $y_{k-1}(i)$ , it can readily be shown that the mean-square of the new residue  $y_k(i)$  is

$$\overline{y_k^2(i)} = \overline{y_{k-1}^2(i)} - \overline{z_k^2(i)} , \quad (6)$$

where the bars denote the mean (time average) operation. Thus, adding a further cascade to the model reduces the mean-square of the residue by an amount equal to the mean-square of the cascade output.

In the simple procedure used in the present study, the impulse response of the dynamic linear element beginning each cascade was defined using a slice of a crosscorrelation function, as just described. Alternatively, a nonlinear mean-square error (MSE) minimization technique can be used to best-fit the dynamic linear and

static nonlinear elements in a cascade to the residue (Korenberg 1991). Then, the new residue is computed, the minimization technique is used again to best-fit another cascade, and so on. This is much faster than using an MSE minimization technique to best-fit all cascades at once. A variety of such minimization techniques, e.g., the Levenberg-Marquardt procedure (Press et al. 1992), are available, and the particular choice of minimization technique is not crucial to the parallel cascade approach. The central idea here is that each cascade can be chosen to minimize the remaining MSE (Korenberg 1991) such that crosscorrelations of the input with the residue are driven to zero. Alternatively, various iterative procedures can be used to successively update the dynamic linear and static nonlinear elements, to increase the reduction in MSE attained by adding the cascade to the model. However, such procedures were not needed in the present study to obtain good results.

A key benefit of the parallel cascade architecture is that all the memory components reside in the dynamic linear elements, while the nonlinearities are localized in static functions. Hence, approximating a dynamic system with higher-order nonlinearities merely requires estimating higher-degree polynomials in the cascades. This is much faster, and numerically more stable than, say, approximating the system with a functional expansion and estimating its higher-order kernels. Nonlinear system identification techniques are finding a variety of interesting applications and, for example, are currently being used to detect deterministic dynamics in experimental time series (Barahona and Poon 1996; Korenberg 1991). However, the connection of nonlinear system identification with classifying protein sequences appears to be entirely new and surprisingly effective, and is achieved as follows.

Suppose that we form an input signal by concatenating one or more representative hydrophobicity profiles from each of two families of protein sequences to be distinguished. We define the corresponding output signal to have a value of  $-1$  over each input segment from the first family, and a value of  $1$  over each segment from the second. The fictitious nonlinear system which could map the input into the output would function as a classifier. Nothing is known about this nonlinear system save its input and output, which suggests resorting to a “black box” identification procedure. Moreover, the ability of parallel cascade identification to conveniently approximate dynamic systems with high-order nonlinearities can be crucial for developing an accurate classifier and, in fact, this approach proved to be effective, as is shown next.

### 3 Implementation

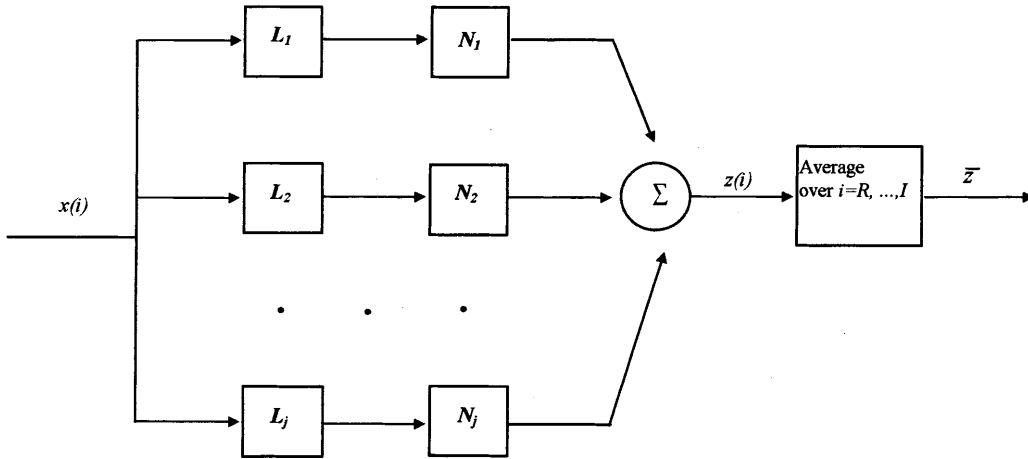
Using the parallel-cascade approach, we wished to investigate how much information about a structure/function family could be carried by one protein sequence in the form of its hydrophobicity profile. Therefore, we selected one protein sequence from the globin family (Brookhaven designation 1hds, with 572 amino acids),

one from the calcium-binding family (Brookhaven designation 1scp, with 348 amino acids), and one from the general kinase family (Brookhaven designation 1pfk, with 640 amino acids). These profiles are unusually long since they are multidomain representatives of their respective families, e.g., the average length of globin family proteins is about 175 amino acids. The lengthier profiles were selected to enable construction of sufficiently elaborated parallel cascade models. Alternatively, one could of course concatenate a number of profiles from the same family together, but we were interested in exploring the information content of single profiles.

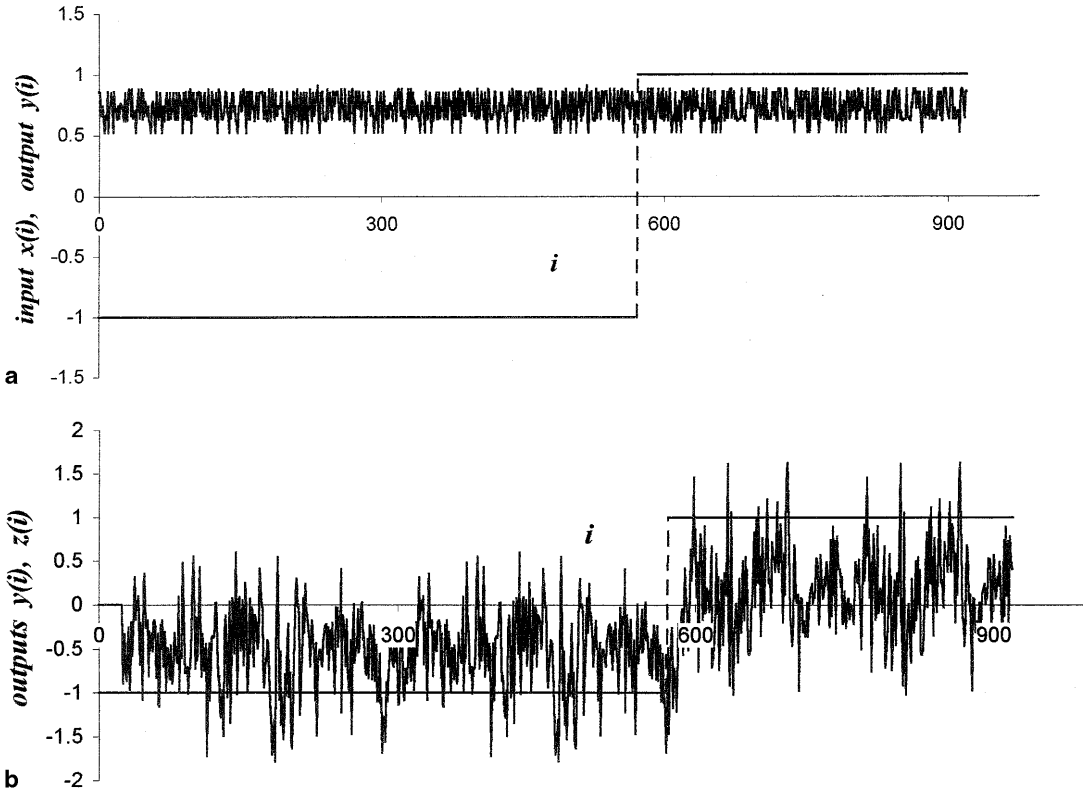
Only two-way (i.e., binary) classifiers were constructed in the present work; a multistate classifier can readily be realized by an arrangement of binary classifiers. To build a parallel cascade classifier to distinguish between, say, globin and calcium-binding protein families, begin by splicing together the two selected profiles from these families (forming a 920-point training input). Define the corresponding training output to be  $-1$  over the globin portion and  $1$  over the calcium-binding portion of the input. The system to be constructed is shown in block-diagram form in Fig. 1, and comprises a parallel cascade model followed by an averager. Figure 2a shows the input and corresponding output used to train the globin versus calcium-binding classifier.

The input output data were used to build the parallel cascade model, but a number of basic parameters had to be chosen. These were the memory length of the dynamic linear element beginning each cascade, the degree of the polynomial which followed, the maximum number of cascades permitted in the model, and a threshold based on a correlation test for deciding whether a cascade’s reduction of the MSE justified its addition to the model. These parameters were set by testing the effectiveness of corresponding identified parallel cascade models in classifying sequences from a small verification set.

This set comprised 14 globin, 10 calcium-binding, and 11 kinase sequences, not used to identify the parallel cascade models. It was found that effective models were produced when the memory length was 25 for the linear elements (i.e., their outputs depended on input lags  $0, \dots, 24$ ), the degree of the polynomials was 5 for globin versus calcium-binding, and 7 for globin versus kinase or calcium-binding versus kinase classifiers, with 20 cascades per model. A cascade was accepted into the model only if its reduction of the MSE, divided by the mean-square of the previous residue, exceeded a specified threshold divided by the number of output points used to fit the cascade (Korenberg 1991). For globin versus calcium-binding and calcium-binding versus kinase classifiers, this threshold was set at 4 (roughly corresponding to a 95% confidence interval were the residue-independent Gaussian noise), and for the globin versus kinase classifier the threshold was 14. For a chosen memory length of 25, each parallel cascade model would have a settling time of 24, so we excluded from the identification those output points corresponding to the first 24 points of each distinct segment joined to form the input. The choices made for memory length, polynomial degree, and maximum number of cascades ensured that



**Fig. 1.** Use of a parallel cascade model to classify a protein sequence into one of two families. Each  $L$  is a dynamic linear element with settling time (i.e., maximum input lag)  $R$ , and each  $N$  is a static nonlinearity. The protein sequence in the form of a hydrophobicity profile  $x(i)$ ,  $i = 0, \dots, I$ , is fed to the parallel cascade model, and the average model output  $\bar{z}$  is computed. If  $\bar{z}$  is less than zero, the sequence is classified in the first family, and if greater than zero in the second family



**Fig. 2.** **a** The training input and output used to identify the parallel cascade model for distinguishing globin from calcium-binding sequences. The input  $x(i)$  was formed by splicing together the hydrophobicity profiles of one representative globin and calcium-binding sequence. The output  $y(i)$  was defined to be  $-1$  over the globin portion of the input, and  $1$  over the calcium-binding portion. **b** The training output  $y(i)$  and the calculated output  $z(i)$  of the identified parallel cascade model evoked by the training input of (a). Note that the calculated output tends to be negative (average value:  $-0.52$ ) over the globin portion of the input, and positive (average value:  $0.19$ ) over the calcium-binding portion

there were fewer variables introduced into a parallel cascade model than output points used to obtain the model. Training times ranged from about 2 s (for a threshold of 4) to about 8 s (for a threshold of 14).

In more detail, consider how the classifier to distinguish globin from calcium-binding sequences was ob-

tained. Using the training input and output of Fig. 2a, we identified a parallel cascade model via the procedure (Korenberg 1991) described above, for assumed values of memory length, polynomial degree, threshold, and maximum number of cascades allowable. We then tested the identified model for its ability to differentiate

between globin and calcium-binding sequences from the small verification set. We observed that the obtained models were not good classifiers unless the assumed memory length was at least 25, so this smallest effective value was selected for the memory length.

For this choice of memory length, the best globin versus calcium-binding classification resulted when the polynomial degree was 5 and the threshold was 4, or when the polynomial degree was 7 and the threshold was 14. Both these classifiers recognized all 14 globin and 9 of 10 calcium-binding sequences in the verification set. In contrast, for example, the model found for a polynomial degree of 7 and threshold of 4 misclassified one globin and two calcium-binding sequences. Hence, to obtain the simplest effective model, a polynomial degree of 5 and threshold of 4 were chosen. There are two reasons for setting the polynomial degree to be the minimum effective value. First, this reduces the number of parameters introduced into the parallel cascade model. Second, there are less numerical difficulties in fitting lower-degree polynomials. Indeed, extensive testing has shown that when two models perform equally well on a verification set, the model with the lower-degree polynomials usually performs better on a new test set.

It remained to set the maximum number of cascades to be allowed in the model. Since the selected memory length was 25 and polynomial degree was 5, each dynamic linear/static nonlinear cascade path added to the model introduced a further  $25 + 6 = 31$  parameters. As noted earlier, the training input and output each comprised 920 points, and we excluded from the identification those output points corresponding to the first 24 points of each segment joined to form the input. This left 872 output points available for identifying the parallel cascade model, which must exceed the number of (independent) parameters to be introduced. Hence at most 28 cascade paths could be justified, but to allow some redundancy, a maximum of 20 cascades were allowed. This permitted 620 parameters in the model, slightly more than 70% of the number of output points used for the identification. While normally such a high amount of parameters is inappropriate, it could be tolerated here because of the low-noise “experimental” conditions. That is, well-established hydrophobicity profiles were spliced to form the training input, and the training output, defined to have the values  $-1$  and  $1$ , was precisely known as explained above.

In this way, we employed the small verification set to determine values of memory length, polynomial degree, threshold, and maximum number of cascades allowable, for the parallel cascade model to distinguish globin from calcium-binding sequences. Recall that the training input and output of Fig. 2a had been used to identify the model. Figure 2b shows that the calculated output of the identified model, when stimulated by the training input, indeed tends to be negative over the globin portion of the input, and positive over the calcium-binding portion.

A test hydrophobicity profile input to a parallel cascade model is classified by computing the average of the resulting output post settling time (i.e., commencing the

averaging on the 25th point). The sign of this average determines the decision of the binary classifier (see Fig. 1). More sophisticated decision criteria are under active investigation, but were not used to obtain the present results. Over the verification set, the globin versus calcium-binding classifier, as noted, recognized all 14 globin and 9 of the 10 calcium-binding sequences. The globin versus kinase classifier recognized 12 of 14 globin, and 10 of 11 kinase sequences. The calcium-binding versus kinase classifier recognized all 10 calcium-binding and 9 of the 11 kinase sequences. The same binary classifiers were then appraised over a larger test set comprising 150 globin, 46 calcium-binding, and 57 kinase sequences, which did not include the three sequences used to construct the classifiers. The globin versus calcium-binding classifier correctly identified 96% (144) of the globin and about 85% (39) of the calcium-binding hydrophobicity profiles. The globin versus kinase classifier correctly identified about 89% (133) of the globin and 72% (41) of the kinase profiles. The calcium-binding versus kinase classifier correctly identified about 61% (28) of the calcium-binding and 74% (42) of the kinase profiles. Interestingly, a blind test of this classifier had been conducted since five hydrophobicity profiles had originally been placed in the directories for both the calcium-binding and the kinase families. The classifier correctly identified each of these profiles as belonging to the calcium-binding family. Out of the 506 two-way classification decisions made by the parallel cascade models on the test set, 427 ( $\approx 84\%$ ) were correct, but the large number of globin sequences present skews the result. If an equal number of test sequences had been available from each family, the overall two-way classification accuracy expected would be about 79%. The two-way classification accuracies for globin, calcium-binding, and kinase sequences (i.e., the amount correctly identified of each group) were about 92%, 73%, and 73%, respectively. These success rates cannot be directly compared with those reported in the literature (Krogh et al. 1994; Regelson 1997) for the hidden Markov model approach because the latter attained such success rates with vastly more training data than required in our study (see Discussion).

Using  $2 \times 2$  contingency tables, we computed the chi-square statistic for the classification results of each of the three binary classifiers over the larger test set. The null hypothesis states that the classification criterion used by a parallel cascade model is independent of the classification according to structure/function. For the null hypothesis to be rejected at the 0.001 level of significance requires a chi-square value of 10.8 or larger. The computed chi-square values for the globin versus calcium-binding, globin versus kinase, and calcium-binding versus kinase classifiers are  $\approx 129$ ,  $\approx 75$ , and 12.497, respectively, indicating high statistical significance.

How does the length of a protein sequence affect its classification? For the 150 test globin sequences, the average length ( $\pm$  the sample standard deviation  $\sigma$ ) was 148.3 ( $\pm 15.1$ ) amino acids. For the globin versus calcium-binding and globin versus kinase classifiers, the average length of a misclassified globin sequence was

108.7 ( $\pm$  36.4) and 152.7 ( $\pm$  24) amino acids, respectively, the average length of correctly classified globin sequences was 150 ( $\pm$  10.7) and 147.8 ( $\pm$  13.5), respectively. The globin versus calcium-binding classifier misclassified only six globin sequences, and it is difficult to draw a conclusion from such a small number, while the other classifier misclassified 17 globin sequences. Accordingly, it is not clear that globin sequence length significantly affected classification accuracy.

Protein sequence length did appear to influence calcium-binding classification accuracy. For the 46 test calcium-binding sequences, the average length was 221.2 ( $\pm$  186.8) amino acids. The average length of a misclassified calcium-binding sequence, for the globin versus calcium-binding and calcium-binding versus kinase classifiers, was 499.7 ( $\pm$  294.5) with seven sequences misclassified, and 376.8 ( $\pm$  218) with 18 misclassified, respectively. The corresponding average lengths of correctly classified calcium-binding sequences were 171.2 ( $\pm$  95.8) and 121.1 ( $\pm$  34.5), respectively, for these classifiers.

Finally, for the 57 test kinase sequences, the average length was 204.7 ( $\pm$  132.5) amino acids. The average length of a misclassified kinase sequence, for globin versus kinase and calcium-binding versus kinase classifiers, was 159.5 ( $\pm$  137.3) with 16 sequences misclassified, and 134.9 ( $\pm$  64.9) with 15 misclassified, respectively. The corresponding average lengths of correctly classified kinase sequences, for these classifiers, were 222.4 ( $\pm$  126.2) and 229.7 ( $\pm$  141.2), respectively.

Thus, sequence length may have affected classification accuracy for calcium-binding and kinase families, with average length of correctly classified sequences being shorter than and longer than, respectively, that of incorrectly classified sequences from the same family. However, neither the correctly classified nor the misclassified sequences of each family could be assumed to come from normally distributed populations, and the number of misclassified sequences was, each time, much less than 30. For these reasons, statistical tests to determine whether differences in mean length of correctly classified versus misclassified sequences are significant will be postponed to a future study with a larger range of sequence data. Nevertheless, the observed differences in means of correctly classified and misclassified sequences, for both calcium-binding and kinase families, suggest that classification accuracy may be enhanced by training with several representatives of these families. Two alternative ways of doing this are discussed in the next section.

#### 4 Discussion

The most effective current approach for protein sequence classification into structure/function groups uses hidden Markov models, a detailed investigation of which was undertaken by Regelson (1997). Some of her experiments utilized hydrophobicity profiles (Rose scale, normalized) from each of which the 128 most significant power components were extracted to repre-

sent the corresponding protein sequence. The families to be distinguished, namely globin, calcium-binding, kinase, and a "random" group drawn from 12 other classes, were represented by over 900 training sequences, with calcium-binding having the smallest number, 116. Successful classification rates on novel test sequences, using trained left-to-right hidden Markov models, ranged over 92–97% for kinase, globin, and "random" classes, and was a little less than 50% for calcium-binding proteins (Table 4.30 in Regelson 1997). These results illustrate that, with sufficiently large training sets, left-to-right hidden Markov models are very well suited to distinguishing between a number of structural/functional classes of protein (Regelson 1997).

It was also clearly demonstrated that the size of the training set strongly influenced generalization to the test set by the hidden Markov models (Regelson 1997). For example, in other of Regelson's experiments, the kinase training set comprised 55 short sequences (128–256 amino acids each) represented by transformed property profiles, which included power components from Rose scale hydrophobicity profiles. All of these training sequences could subsequently be recognized, but none of the sequences in the test set (Table 4.23 in Regelson 1997), so that 55 training sequences from one class were still insufficient to achieve class recognition.

The protein sequences in our study are a randomly selected subset of the profiles used by Regelson (1997). The results reported above for parallel cascade classification of protein sequences surpass those attained by various linear modeling techniques described in the literature. A direct comparison with the hidden Markov modeling approach has yet to be done based on the amount of training data used in our study. While three protein sequence hydrophobicity profiles were used to construct the training data for the parallel cascade models, an additional 35 profiles forming our verification set were utilized to gauge the effectiveness of trial values of memory length, polynomial degree, number of cascades, and thresholds. However, useful hidden Markov models might not be trainable on only 38 hydrophobicity profiles in total, and indeed it is clear from Regelson (1997) that several hundred profiles could sometimes be required for training to obtain consistent results.

Therefore, for the amount of training data in our pilot study, parallel cascade classifiers appear to be comparable to other currently available protein sequence classifiers. It remains open how parallel cascade and hidden Markov model performance compare using the large training sets often utilized for the latter approach. However, because the two approaches differ greatly, they may tend to make their classification errors on different sequences, and so might be used together to enhance accuracy.

Several questions and observations are suggested by the results of our pilot study so far. Why does a memory length of 25 appear to be optimal for the classifiers? Considering that hydrophobicity is a major driving force in folding (Dill 1990) and that hydrophobic-hydrophobic interactions may frequently occur between amino

acids which are well separated along the sequence, but nearby topologically, it is not surprising that a relatively long memory may be required to capture this information. It is also known from autoregressive moving average (ARMA) model studies (Sun and Parthasarathy 1994) that hydrophobicity profiles exhibit a high degree of long-range correlation. Further, the apparent dominance of hydrophobicity in the protein folding process probably accounts for the fact that hydrophobicity profiles carry a considerable amount of information regarding a particular structural class. It is also interesting to note that the globin family in particular exhibits a high degree of sequence diversity, yet our parallel cascade models were especially accurate in recognizing members of this family. This suggests that the models developed here are detecting structural information in the hydrophobicity profiles.

In future work, we will construct multi-state classifiers, formed by training with an input of linked hydrophobicity profiles representing, say, three distinct families, and an output which assumes values of, say,  $-1$ ,  $0$ , and  $1$  to correspond with the different families represented. This work will consider the full range of sequence data available in the Swiss-Prot sequence data base. We will compare the performance of such multi-state classifiers with those realized by an arrangement of binary classifiers. In addition, we will investigate the improvement in performance afforded by training with an input having a number of representative profiles from each of the families to be distinguished. An alternative strategy to explore is identifying several parallel cascade classifiers, each trained for the same discrimination task, using a different single representative from each family to be distinguished. It can be shown that, if the classifiers do not tend to make the same mistakes, and if each classifier is right most of the time, then the accuracy can be enhanced by having the classifiers vote on each decision. To date, training times have only been a few seconds on a 90-MHz Pentium, so there is some latitude for use of lengthier and more elaborate training inputs, and/or training several classifiers for each task.

The advantage of the proposed approach is that it does not require any a priori knowledge about which features distinguish one protein family from another. However, this might also be a disadvantage because, due to its generality, it is not yet clear how close proteins of different families can be to each other and still be distinguishable by the method. Additional work will investigate, as an example, whether the approach can be used to identify new members of the CIC chloride channel family, and will look for the inevitable limitations of the method. For instance, does it matter if the hydrophobic domains form alpha helices or beta strands? What kinds of sequences are particularly easy or difficult to classify? How does the size of a protein affect its classification? We began an investigation of the latter question in this paper, and it appeared that sequence length was a factor influencing the accuracy of

the method in recognizing calcium-binding and kinase proteins, but was less evidently so for globins. This suggested that using further calcium-binding and kinase exemplars of differing lengths in training the parallel cascade classifiers may be especially important to increase classification accuracy.

The present work appears to confirm that hydrophobicity profiles store significant information concerning structure and/or function as was observed by Regelson (1997). Our work also indicates that even a single protein sequence may reveal much about the characteristics of the whole family, and that parallel cascade identification is a particularly efficient means of extracting characteristics which distinguish the families. We are now exploring the use of parallel cascade identification to distinguish between coding (exon) and non-coding (intron) DNA or RNA sequences. Direct applications of this work are both in locating genes and increasing our understanding of how RNA is spliced in making proteins.

*Acknowledgements.* Supported in part by grants from the Natural Sciences and Engineering Research Council of Canada. We thank the referees for their astute comments on the manuscript.

## References

- Baldi P, Chauvin Y, Hunkapiller T, McClure MA (1994) Hidden Markov models of biological primary sequence information. *Proc Natl Acad Sci USA* 91:1059–1063
- Barahona M, Poon C-S (1996) Detection of nonlinear dynamics in short, noisy time series. *Nature* 381:215–217
- Cornette JL, Cease KB, Margalit H, Spouge JL, Berzofsky JA, DeLisi C (1987) Hydrophobicity scales and computational techniques for detecting amphipathic structures in proteins. *J Mol Biol* 195:659–685
- Dill KA (1990) Dominant forces in protein folding. *Biochemistry* 29:7133–7155
- Korenberg MJ (1991) Parallel cascade identification and kernel estimation for nonlinear systems. *Ann Biomed Eng* 19:429–455
- Krogh A, Brown M, Mian IS, Sjolander K, Haussler D (1994) Hidden Markov models in computational biology – applications to protein modeling. *J Mol Biol* 235:1501–1531
- McLachlan AD (1993) Multichannel Fourier analysis of patterns in protein sequences. *J Phys Chem* 97:3000–3006
- Press WH, Teukolsky SA, Vetterling WT, Flannery BP (1992) *Numerical recipes in C: the art of scientific computing*, 2nd edn. Cambridge University Press, Cambridge
- Regelson ME (1997) Protein structure/function classification using hidden Markov models. Ph.D. Thesis, The Beckman Institute, California Institute of Technology, Pasadena
- Stultz CM, White JV, Smith TF (1993) Structural analysis based on state-space modeling. *Protein Sci* 2:305–314
- Sun SJ, Parthasarathy R (1994) Protein-sequence and structure relationship ARMA spectral-analysis – application to membrane-proteins. *Biophys J* 66:2092–2106
- White JV, Stultz CM, Smith TF (1994) Protein classification by stochastic modeling and optimal filtering of amino-acid-sequences. *Math Biosci* 119:35–75
- Wiener N (1958) *Nonlinear problems in random theory*. MIT Press, Cambridge, Mass