

Purdue University
Purdue e-Pubs

ECE Technical Reports

Electrical and Computer Engineering

11-1-1993

A Selective Committee Architecture for Time Series Prediction and Pattern Classification

Antonio G. Thome

Purdue University School of Electrical Engineering

Manoel F. Tenorio

Purdue University School of Electrical Engineering

Follow this and additional works at: <http://docs.lib.purdue.edu/ecetr>

Thome, Antonio G. and Tenorio, Manoel F, "A Selective Committee Architecture for Time Series Prediction and Pattern Classification" (1993). *ECE Technical Reports*. Paper 248.

<http://docs.lib.purdue.edu/ecetr/248>

This document has been made available through Purdue e-Pubs, a service of the Purdue University Libraries. Please contact epubs@purdue.edu for additional information.

A SELECTIVE COMMITTEE
ARCHITECTURE FOR TIME
SERIES PREDICTION AND
PATTERN CLASSIFICATION

ANTONIO G. THOME
MANOEL F. TENORIO

TR-EE 93-40
NOVEMBER 1993



SCHOOL OF ELECTRICAL ENGINEERING
PURDUE UNIVERSITY
WEST LAFAYETTE, INDIANA 47907-1285

**A Selective Committee Architecture for
Time Series Prediction and
Pattern Classification**

**Antonio G. Thome
and
Manoel F. Tenorio**

**School of Electrical Engineering
Purdue University
West Lafayette, IN 47907-1285**

A Selective Committee Architecture for Time Series Prediction and Pattern Classification

Antonio G. **Thome** and **Manoel** F. Tenorio
Parallel Processing Laboratory
School of Electrical Engineering
Purdue University

Abstract:

In this report we describe a novel technique to generate a committee architecture for time series prediction. The algorithm, here named Selective Multiple Prediction Network, consists of three steps: a systematic partition of the input hyperspace, a selective training of many agents and a flexible combining strategy. Potentially uncorrelated agents are generated which improves the combination process. The proposed architecture is easily extended to the class of classification problems.

Key words: Committee Architecture, Team Prediction, Combining Predictions

1. INTRODUCTION

System Identification and Linear Prediction are two very important topics in the field of System Theory, and are widely applied to many diverse areas such as Signal Processing, Control, and Forecasting. System Identification is the process of estimating an unknown structure by the knowledge of only its input / output pairs. Linear Prediction, on the other hand, is the process of estimating a future system response based only on the knowledge of its present and past responses.

The emphasis on creating a predictor relies on the identification of underlying patterns, and on the estimation of the model parameters. Historical data analysis and pattern consistency, i.e., time series stationarity, are respectively the major resource and the major underlying assumption. It may be intuitive the understanding that more complex the problem the more difficult is the underlying patterns identification, and even more difficult is the estimation of a single model able to satisfactorily cover the entire problem.

A large class of prediction problems can be better solved by decomposing the original problem into several subproblems and then combining the multiple predictions. This approach leads to a higher accuracy than solving the problem with a single and global predictor. Our committee approach has a simple architecture (fig. 1) and consists basically of three steps as follows:

- decomposition of the original problem into several (ideally disjoint) subproblems;
- parallel estimation of the parameters for each model (agent); and
- combination strategies.

The first step, partition of the original problem, relies on the believe that for those complex problems, a better result may be achieved through a divide and conquer approach. A transformation is applied to the original time series and spacial similarities of the state-space vector are exploited by the use of an unsupervised clustering procedure.

In the second step, many agents are trained in parallel, on the subsets created by the clustering procedure. To each cluster there is a corresponding agent that can be seen as an expert on a

particular view of the underlying structure. Uncorrelated agents are expected to result from this training scheme.

Each agent provides its own prediction, and the network or committee final prediction is then obtained through the combination of the individual contributions. Three different combining strategies are proposed.

2. COMBINING TECHNIQUES

In a seminal paper [Bat69], Bates and Granger showed that a simple linear combination of distinct predictions generally outperforms the individual predictions. A stream of papers followed this initial work. Clemen and Winkler [Cle86] and Clemen [Cle89] provide excellent summaries and extensive bibliographic references.

The field has so far been dominated by works in statistical decision theory, with particular emphasis on optimal linear combination and on Bayesian inference. Recently, connectionist researchers have begun to show a strong interest in the subject in the form of network committees, agent teams, stacked generalization, and others. Some of the major contributions can be cited as follows:

- Simple averaging procedures can be found in works done by Doyle and Fenwick [Doy76], Makridakis [Mak82], Makridakis and Winkler [Mak83], and Kang [Kan86];
- Minimization of the error cross-covariance matrix can be found in Bates and Granger [Bat69], Newbold and Granger [New74], Winkler and Makridakis [Win83], and Kang [Kan86];
- Historical weights, where each agent is weighted by one minus the ratio of its own MSE to the overall team MSE, is found in Doyle and Fenwick [Doy76];
- Ordinary least square regression, either restricted to the weights adding up to one, for the case where the individual agents are known unbiased, and unrestricted in the contrary, can be found in Granger and Ramanathan [Gra84], Bopp [Bop85], Clemen [Cle86], and Trenkler and Hiski [Tre86];
- Bayesian methods, based on the posterior probability, are found in Bunn [Bun75/77/81], Winkler [Win81], Bordley [Bor82/86], Gupta and Wilson [Gup87], and Chen and Anandaligan [Che87].

Among the connectionists it can be cited the following recent contributions:

- The weighted majority algorithm, where Nick Littlestone and Manfred Warmuth [Lit91] use an weighted voting scheme to combine a team of agents. They claim that the algorithm is robust with respect to noise and that the misclassification bounds for the pool are closely related to the error bounds of the best algorithms in the pool.
- Uwe Beyer and Frank Smieja [Bey93] suggest a combination that takes into account a degree of selfconfidence, named reflection, where each model estimates the correctness of its own predictions.

- Partitions of the learning set are also implicit in Wolpert's stacked generalizers [Wol92], Schapire's boosting technique [Sch89], and Ersoy and Hong's self organizing networks [Ers89, Hon91].

- Mackay [Mac93], winner of the 1993 energy prediction contest, uses a simple averaged combination of the "k" best models ranked by their performance on the validation set. He observes that although the committee's validation error was significantly better than any single model, the same improvement did not extend to the test set. Mackay says, however, that in adaptive on-line schemes, the committee may always outperform a single model.

- Hybrid systems are found in [Zha92], winner of the protein prediction contest, and in [Yos90] where a combination of neural network agents and hidden markov models is suggested for a speech-recognition task.

3. THE COMBINING PARADIGM

Combining is theoretically no worse than any of the individual agents, which can be shown as follows:

Let A_α and A_β be two distinct agents working on information sets I_α and I_β , and let f_α^n and f_β^n be their corresponding predictions for time step n .

If the predictions are optimal with respect to their respective information sets, and hence, they can be written in terms of posterior expected values, i.e.

$$f_\alpha^n = E \{ X_n / I_\alpha \}, \quad (1)$$

and

$$f_\beta^n = E \{ X_n / I_\beta \}. \quad (2)$$

The optimal prediction, based on all possible information is then known to be

$$f^n = E \{ X_n / I_\Gamma \}, \quad I_\Gamma = I_\alpha \cup I_\beta. \quad (3)$$

This complete estimation problem is normally very complex and computationally expensive. A particular subset of $\{I_\Gamma\}$ that can be considered for example, is a linear combination of the individual predictions

$$C^n = \alpha_1 f_\alpha^n + \alpha_2 f_\beta^n. \quad (4)$$

It is expected that α_1 or α_2 should go to zero whenever f_α^n or f_β^n is optimal with respect to the global information set $\{I_\Gamma\}$. If neither one is optimal then α_1 and α_2 are expected to be different from zero. In general, C^n and f^n are not equal, which clearly indicates that the combination will not be optimal too, although a superior result to each of the original predictions is expected.

Although showing potential for performance improvement, combining techniques present some weak points. The combined performance is highly dependent on the estimation error cross-correlation, serial correlation, and bias. The most effective combinations are achieved with no

positive cross-correlation between individual model errors. When negative correlation occurs, which is quite rare, the gains can be spectacular. However, with high positive cross-correlation it is often difficult to achieve even a small improvement. Moreover, if an unstable optimization technique is used, the results may be even worse than those of using equal weights or of selecting the apparently best model.

4. A THEORETICAL ANALYSIS

Mean squared error (MSE), is the most straightforward way to combine multiple predictions from a connectionist point of view. Linear and even non-linear regressions can be easily implemented. Unconstrained regression with the use of an intercept (bias) can be shown to be theoretically the best alternative [Gra84].

For clarity, the following notation will be adopted:

Methods

A	unconstrained regression without bias
B	constrained regression
C	unconstrained regression with bias

Terminology

$x = (x_1, x_2, \dots, x_n)^T$; $n \times 1$ vector of values to be predicted

$f^j = (f_1^j, f_2^j, \dots, f_n^j)^T$; $n \times 1$ vector of predictions provided by the j^{th} agent

$F = (f^1, f^2, \dots, f^k)$; $n \times k$ matrix of predictions

$l = [1 \ 1 \dots \ 1]^T$; vector of "1's" of appropriate dimension

$\alpha = (\alpha_1 \ \alpha_2 \dots \ \alpha_k)^T$; $k \times 1$ vector of combining weights.

Method A: $X = F\alpha + e_c$

The mean squared optimization ($\min_{\alpha} (e_c^2)$), leads to the critical solution

$$\alpha^* = (F^T F)^{-1} F^T X. \quad (5)$$

The resulting optimal quadratic error Q_A will be taken as the standard for comparisons. The quadratic error is given by

$$Q_A = (X - X_A^*)^T (X - X_A^*) = X^T X - X^T F \alpha^*. \quad (6)$$

This unconstrained optimization scheme can be shown to be unbiased only if the individual models are unbiased and the combining weights add up to one. Let the prediction error be given by

$$e_c = X - x_A^* = X - F \alpha^*. \quad (7)$$

For the average error ($l^T e_c$) to equal zero, it is required that $(l^T X = l^T F \alpha^*)$.

Assuming every individual agent is unbiased, then

$$1^T(X - f^j) = 0, \quad \forall j$$

which, in matrix notation gives

$$1^T.X.1 = 1^T.F.$$

Multiplying both terms by (α^*) gives

$$(1^T.X).1\alpha^* = 1^T.F\alpha^*.$$

Hence, the sufficient conditions for an unbiased combination are:

- a) f^j , is unbiased $\forall j$;
- b) $1^T\alpha^* = 1$, i.e. the weights must sum to 1.

Method B: $X = F\beta + e_c$
 subject to
 $1^T\beta = 1$

Using Lagrange multipliers the optimization problem can be transformed into

$$\min_{\beta} \{ (X-F\beta)^T(X-F\beta) + 2\lambda(1^T\beta-1) \}. \quad (8)$$

The first order necessary conditions

$$F^T(X-F\beta) - \lambda 1 = 0$$

$$1^T\beta = 1$$

lead to the following critical values

$$\beta^* = \alpha^* - \lambda (F^TF)^{-1} 1$$

$$\lambda^* = (1^T\alpha^* - 1)(1^T(F^TF)^{-1})^{-1}.$$

Where $\alpha^* = (F^TF)^{-1}F^TX$ is the optimal weight vector from method A

The quadratic error

$$Q_B = (X - x_B^*)^T(X - x_B^*),$$

reduces, after some algebra, to

$$Q_B = Q_A + (h^*)^2 1^T(F^TF)^{-1} 1.$$

Which shows that $Q_B \geq Q_A$, i.e., unconstrained without bias is superior to the constrained approach.

Method C: $X = F\delta + 1^T\delta_0 + e_c$

MSE optimization yields the normal equations:

$$F^T(X - F\delta - \delta_0^T 1) = 0$$

$$1^T(X - F\delta - \delta_0^T 1) = 0.$$

This leads to the following critical values:

$$\delta^* = \mathbf{a}^* - (F^T F)^{-1} F^T 1^T \delta_0^*$$

$$\delta_0^* = \frac{1^T \mathbf{e}_A}{n - \Gamma},$$

where

$$\Gamma = 1^T F (F^T F)^{-1} F^T 1^T$$

\mathbf{e}_A is the error vector using method **A**.

The quadratic error $Q_c = (X - x_c^*)^T (X - x_c^*)$ reduces, after some algebra, to

$$Q_c = Q_A - \frac{(1^T \mathbf{e}_A)^2}{n - \Gamma} \quad n - \Gamma > 0$$

Thus $Q_c \leq Q_A$ and hence unconstrained MSE with bias is theoretically optimal among the three alternatives.

In summary it can be said that combining multiple predictions has an intuitive appeal. The subject has been thoroughly studied and theoretical analysis suggests it is no worse than the best individual model. Error cross-correlation and bias are the two main concerns for successful results. The major issue for committee models is therefore the generation or selection of agents, ideally uncorrelated or even better, complimentary biased. Requirements that are very difficult to be achieved in practice.

In the rest of this report, a new approach is described. It tries to generate distinct models by training similar networks on different subsets of the input vector space. These subsets are defined through an unsupervised clustering procedure which groups together those vectors with similar properties or spatial characteristics. The number of distinct cluster defines the number of networks, which can be seen as experts, each on a particular region of the input space.

5. THE SELECTIVE MULTIPLE PREDICTION NETWORK

To train agents over distinct subsets of the full training set is not a new idea. Wolpert [Wol92] uses arbitrarily selected partitions to train the first layer of generalizers; Schapire [Sch89] adopts a residual scheme in which every new agent is trained only on those vectors which previous agents have disagreed on. Ersoy's parallel, self-organizing, hierarchical neural networks [Ers89, Hong91], can also be seen as a kind of residual partition where new agents are trained on transformed versions of those samples rejected by previous agents. In SMP a different scheme to partition the input data set is used. A clustering algorithm is adopted to subdivide the original problem into sets of more homogeneous and easier subproblems, which may eventually lead to learning and prediction improvements.

The major motivation for the SMP approach comes from the observation that many time series problems present certain underlying spatial patterns. These patterns if captured and exploited

by the network certainly help to improve prediction accuracy. In this present implementation, any stationary signal is a potential candidate for the successful use of this approach.

The Selective Multiple Prediction Network (fig. 1) involves three distinct processing steps and a number of distinct agents working in parallel. These agents can form a hybrid or a homogeneous structure depending on how they differ from one another. In our studies we only considered homogeneous systems in which each agent is a neural network.

5.1 Processing Steps

Pattern matching, function approximation, and a combining strategy are the most important components of the selective multiple prediction task. Pattern matching involves feature selection and unsupervised learning; function approximation involves selective supervised learning, where each neural network is trained to become an expert on specific views of the entire environment; and the combining strategy generates the final prediction. Figure 2 shows a block diagram of these steps.

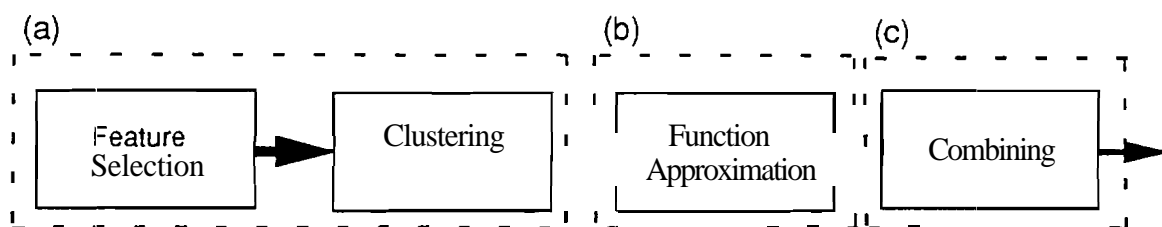


Figure 2- SMP Processing Steps block diagram, (a) pattern matching; (b) function approximation; (c) combining strategy.

The selection of relevant features is the first and one of the most important steps. In univariate time series this selection process can be thought of in terms of defining different embedding dimensions, i.e., the number of past values to be used in the model. Feature selection [The89, Hsu93, Lap86] is generally a very time consuming and complex task. Here we favored the use of a spread ratio measure r_s (eqn 10) to select those possible embeddings leading to a more consistent unsupervised partition of the input space. The model for a time series is generally expressed as

$$Y = f(X) + \epsilon, \quad (9)$$

where

$$X = [x(t) \ x(t-\tau) \ x(t-2\tau) \ \dots \ x(t-(m-1)\tau)]^T$$

$$Y = x(t+T)$$

τ is the sampling period

m is the embedding size

T is the prediction horizon (lead time).

The spread ratio measure is defined as

$$r_s = \text{mean} \left(r_{\sigma}^i \ r_f^i \right),$$

where

$$r_{\sigma}^i = \frac{\text{mean}(\sigma_1^2)}{\sigma^2}, \quad (10)$$

is the data consistency term,

σ_i^2 is the outcome variance for the input vectors belonging to cluster i

σ^2 is the outcome variance for the whole training vectors, and

$$r_F^i = \frac{\text{mean}(d_w^i)}{\text{mean}(d_B^i)}, \quad (11)$$

is the Fisher discriminator term,

$$d_w^i = \frac{1}{n_i} \sum_{j=1}^{n_i} \|X_j - V_i\|^2, \quad \{\forall j / X_j \in \text{Cluster } i\}$$

$$d_B^i = \frac{1}{c-1} \sum_{j \neq i} \|V_j - V_i\|^2.$$

Once the embedding m is determined, the time series can be rewritten as a collection of input vectors X , also known as state vectors, and their corresponding outcome Y . By ignoring the time dependence among these vectors, the regression problem can be viewed as a pattern association of pairs of vectors

$$X^1 \rightarrow Y^1$$

$$X^2 \rightarrow Y^2$$

$$X^n \rightarrow Y^n.$$

This transformation is the key to instance-based approaches [Aka91, Hsu93] in which the outcome prediction for the current state vector is based on a number of past state vectors found using a look-up search.

Here, in the selective multiple prediction approach, a similar transformation is applied to the time series to subdivide the original problem into a set of more homogeneous subproblems. The fuzzy locally sensitive clustering, described in [Tho93a], and also a K -means algorithm were used to partition the training set for several different embeddings. The two previously mentioned performance criteria were then applied to identify those embeddings resulting in more consistent and better shaped partitions. The cluster centroids of the best partition were then taken as class representatives for the pattern matching and selective function approximation phases.

Each cluster defines an associated agent that is trained only on those samples that are classified to the corresponding partition. Three strategies for learning and combining the individual predictions to form the committee prediction were evaluated. The first and simplest one, named winner-take-all, selects a single agent at every time step to perform the prediction. The selected agent is the one whose corresponding cluster centroid is closest to the current input vector. The second approach, called *full* committee, is at the other extreme, where all agents are taken into consideration. Each agent contributes and is trained on a percentage of the final prediction. The percentages or weights add up to one and correspond to the degree of membership of the

current input vector in each cluster. The third approach, called *windowed-committee*, is in between the two others, since it takes into consideration a subset of the available agents. A temporal window is used as a selection criterion to induce time continuity or time similarity. The combination of spatial and temporal similarities has special appeal in time series applications.

5.2 - Learning Procedures

The Quickprop algorithm [Fah88] with adaptive region of nonlinearity, as described in [Tho93b], was used in all experiments. All training data sets were 1500 or more samples long, and the agents were trained in parallel accordingly to each combining strategy. Batch training with a fixed number of epochs upper bounded at 1000, was used.

case a) Winner-take-all procedure

- step 1- classify current input vector
- step 2- select winning agent
- step 3- estimate desired outcome
- step 4- train selected agent through error backpropagation

case b) Full-committee procedure

- step 1- compute degree of membership for current input vector

$$\mu_i = \frac{\exp\left(-\sqrt{(X-V_i)^T(X-V_i)}\right)}{\sum_{i=1}^c \exp\left(-\sqrt{(X-V_i)^T(X-V_i)}\right)};$$

- step 2- estimate the outcome for all agents in parallel

$$\hat{y}_i = f(X, W_i); \quad i=1,c$$

- step 3- generate the committee prediction by combining the individual outcomes

$$\hat{y} = \sum_{i=1}^c \mu_i \hat{y}_i;$$

- step 4- train each agent by backpropagating its contribution to the overall error

$$e_T = y_d - \hat{y},$$

and

$$e_i = \mu_i e_T$$

case c) windowed-committee procedure

- step 1- classify current input vector
- step 2- select winning agent

step 3- insert a tag of winner at the head of the time-window queue (FIFO) and eliminate the oldest tag

$$WD = [Tg_0 \ Tg_{-1} \ \dots \ Tg_{-k+1}], \quad \text{window of size } k$$

step 4- estimate the outcomes for the agents belonging to the time-window queue

$$\hat{y}_i = f(X, W_i), \quad i=1, k$$

step 5- combine individual outcomes

$$\hat{y} = \sum_{i=1}^k \lambda_i \hat{y}_i,$$

where

$$\lambda_i = \frac{\beta^i}{\sum_{i=1}^k \beta^i} \quad 0 < \beta \leq 1$$

and

$$\sum_{i=1}^k \lambda_i = 1, \quad \text{is the time weight decay}$$

step 6- train each agent by backpropagating its contribution to the overall error

$$e_T = y_d - \hat{y}$$

and

$$e_i = \lambda_i e_T.$$

Observe that the number of different agents within the window varies from 1 to k; the smoother the time series is, the smaller this number will be. For classification problems this dynamic window can be thought of as a subset of the full set of agents which provide a normalized degree of membership above a certain threshold, as for example $\frac{\mu_i}{\max(\mu)} > T$.

5.3 - SMP Properties and Drawbacks

SMP provides a powerful architecture to deal with complex real world problems. A set of specialized networks are used, rather than a single large network which must accommodate all aspects and underlying dynamics of the problem. Specialization, team cooperation, and truly parallel operation are the key issues in SMP. Robustness, complexity and learning effort reduction, and prediction accuracy improvement are the major goals.

Major properties:

- transforms complex problems into a set of more homogeneous and easily treated subproblems;
- uses smaller individual networks which reduces dimensionality problems and improves learning time;

- exploits spatial and temporal similarity of the input vector which is intuitive and appealing for many real world time series applications;
- combines instance based with parametric approaches without the memory and recall time overhead of the former:
- adopts either hybrid or homogeneous structure, with a flexible combining strategy;
- generates potentially distinct agents by training them on different partitions of the training set.

Observed Drawbacks:

- requires large training sets to avoid situations where an agent is trained on a very small number of patterns;
- requires frequent full retraining and input space partitioning if applied to non-stationary time series;
- may present overfitting problems since the number of training epochs is set equal for all agents and each agent has its own distinct training set. The training sets may have different sizes and degrees of complexity.

6. EMPIRICAL RESULTS

The Mackey-Glass chaotic time series was chosen for this benchmark due to its common use among connectionist researchers. The purpose behind the use of Mackey-Glass time series was not to show improvements of current estimates that are already at practical limits. Further improvement is of little practical value. Rather, our purpose was to use a chaotic system defined by a continuous orbit, which by nature does not have clearly definable clusters in the state space. If a reasonable prediction can be attained with this technique, functions that are clearly decomposable into multiple mappings can therefore, more easily be dealt with.

The Mackey-Glass equation was first proposed as a model of white blood cell production [Mac77] and subsequently popularized in the nonlinear field due to its richness in structure [Far82]. It is a time-delayed differential equation stated as follows:

$$\frac{\delta x}{\delta t} = \frac{ax(t-\Delta)}{[1+x^c(t-\Delta)]} - bx(t). \quad (9)$$

Which in discrete time domain can be rewritten as:

$$x(t+1) = \frac{ax(t-\Delta)}{[1+x^c(t-\Delta)]} - (b-1)x(t). \quad (10)$$

The constants are often taken as: $a= 0.2$, $b= 0.1$, and $c= 10$. The delay coefficient Δ determines the nature of the chaotic behavior displayed by the time series. This chaotic behavior, as studied in [Far82], is presented in table 1. There are two values of Δ (17, 30) that are commonly used for benchmarking predictions. Choosing $\Delta = 17$ yields a chaotic behavior, and a strange attractor with the fractal dimension ≈ 2.1 . Even for the same Δ , different initial values " $x(0)$ " generate different dynamics. Choosing $\Delta = 30$ yields a strange attractor with the fractal dimension ≈ 3.5 . The phase space of this system is infinite dimensional.

At $\Delta = 17$, $x(t)$ appears quasiperiodic and the power spectrum is broadband with numerous spikes due to the quasiperiodicity. Figure 3 shows the phase portrait and the time dynamics of the Mackey-Glass time series when generated with the standard values of parameters a , b , and c , $\Delta = 17$, $x(0)= 0.9$. For prediction it is commonly taken an embedding of six ($m=6$), a sampling rate of 6 ($\tau=6$), and a lead time (prediction horizon) of 6 or 85 ($T= 6$ or 85).

In Mackey-Glass benchmarks, it is commonly avoided to draw conclusions based solely on direct numerical comparisons with other published results. This is because of the differences that can arise from the use of different integrators, initial conditions, sampling rate, and transient elimination. In our study, all results are reported in terms of *Nrmse*.

Table 1 - Mackey-Glass dynamics as a function of Δ

$\Delta < 4.53$	a stable fixed point attractor
$4.53 < \Delta < 13.3$	a stable limit cycle attractor
$13.3 < \Delta < 16.8$	period of limit cycle doubles
$\Delta > 16.8$	chaotic attractor characterized by Δ

6.1 Input Space Partition

According to Takens [Tak81], a chaotic time series $x(t)$ can be predicted T time steps in the future by using only m number of equally spaced past samples of the time series itself. The prediction value is then obtained as follows:

$$x(t+T) = \mathcal{F} \{ x(t), x(t-\tau), x(t-2\tau), \dots, x(t-(m-1)\tau) \} \quad (11)$$

where \mathcal{F} , under suitable assumptions, is a nonlinear continuous function. The choice of an embedding scheme for a benchmark means the determination of the three parameters T , m and τ for the time series. In our experiments we adopted the most widely used values, i.e. $m= 6$, $\tau= 6$ and $T= 6$ and 85.

Using the above parameters, many distinct partitions of a training set with 700 samples were evaluated. A K-means clustering algorithm was used for several values of c (number of

clusters), and the performance criterion r_s (eqn 10) was evaluated for each resulting partition. The results as shown in figure 4, indicates a systematic partition improvement as the number of clusters increase. Other observation is that the quality of the partition deteriorates as the lead time T goes further in the future. This is because of the chaotic nature of the series.

6.2 Learning

Winner-,take-all,full-committee and windowed-committee schemes were evaluated on different partition sizes for lead times of 6 and 85. Each model (neural network structure) was defined with a single hidden layer (5 units for the T=6 case and 7 units for the T=85 case) and one output unit, hyperbolic tangent with ARON was adopted for all units. Tables 2 and 3 show some of the obtained results.

Table 2 - Mackey-Glass Committee Training Performance in Nrmse

Committee	Num. Clusters	T= 6	T= 85
WTA	09	.1962	.4309
WTA	23	.0773	.3403
Windowed	09	.1904	.4233
Windowed	23	.0728	.3194
Full-committee	23	.1221	.3752

Table 3 - Mackey-Glass WTA-Committee Training/Prediction for T= 85 (Cls - cluster number, No - cluster training Nrmse, cluster testing Nrmse; partition size of 23 clusters)

Cls	No	Ns	Cls	No	Ns	Cls	No	Ns
01	.1182	.1117	09	.2285	.2742	17	.1551	.2677
02	.5304	.3842	10	.6292	.6807	18	.3656	.3961
03	.2167	.2912	11	.0660	.0712	19	.5488	.6979
04	.7403	.1614	12	.7720	.6555	20	.3548	.4264
05	.1727	.1471	13	.2011	.2296	21	.5299	.4234
06	.3123	.1949	14	.2516	.1834	22	.2752	.2823
07	.3374	.3371	15	.3698	.3582	23	.5284	.3587
08	.2821	.2876	16	.1447	.1548	WTA	.3403	.3694

The architecture of SMP provides the flexibility to customize and individually tune each Agent. Therefore, in this experiment for example, Agents with poor training performance could, in the WTA scheme, be selected for individual retraining and final accuracy may eventually improve.

6.3 Committee Prediction

Table 4 and figure 5 show the committee prediction results for a partition size of 23 clusters. Observe that the prediction provided by the WTA scheme shows very good performance on

turning points, with almost no lag, with may be of great interest for some real world applications.

Table 4 - Mackey-Glass Committee Prediction Comparison

Committee	T = 6	T = 85
WTA	.0825	.3694
Windowed	.0835	.3653
Full	.1123	-

7. CONCLUSIONS

The Selective Multiple Prediction Network provides a very flexible and powerful architecture to handle those more complex problems, where a single and global model is very unlikely to exist. **Decomposing** the original problem into more homogeneous subproblems leads to **potentially** uncorrelated and simpler Agents. Less demanding training effort, and customized tuning **according** to the requirements of each subproblem are some of the characteristics of this approach. This proposed architecture can also be seen as a structure to combine neural networks (prediction module) with more sophisticated schemes of expert systems (selection module:)

8. REFERENCES

- [Bat69] Bates, J. M. and Granger, C.W.J., 1969, "The combination of forecasts", *Opl Res Q.*, vol 20, pp. 451-468.
- [Bey93] Beyer, U. and Smieja, F., 1993, "Learning from examples, agent teams and the concept of reflection"
- [Bop85] Bopp, A., 1985, "On combining forecasts: some extensions and results", *Management Science*, 31, pp. 1492-1498.
- [Bor82] Bordley, R.F., 1982, "The combination of forecasts: a bayesian approach", *Journal of the Operational Research Society*, 33, pp. 171-174.
- [Bor86] Bordley, R.F., 1986, "Linear combination of forecasts with an intercept: a Bayesian approach", *Journal of Forecasting*, vol 5, pp. 243-249.
- [Bun75] Bunn, D.W., 1975, "A bayesian approach to the linear combination of forecasts", *Operational Research Quarterly*, 26, pp. 325-329.
- [Bun77] Bunn, D.W., 1977, "A comparative evolution of the outperformance and minimum variance procedures for the linear synthesis of forecasts", *Opl. Res. Q.*, vol 28, 3, pp. 653-662.
- [Bun81] Bunn, D.W., 1981, "Two methodologies for the linear combination of forecasts", *J. Opl. Res. Soc.*, vol 32, 1, pp. 213-222.

- [Bun89] Bunn, D., 1989, "Forecasting with more than one model", *Journal of Forecasting*, vol 8, pp. 161-166.
- [Che90] Chen, L. and Anandalingam, G., 1990, "Optimal selection of forecasts", *Journal of Forecasting*, vol 9, pp. 283-297.
- [Cle86] Clemen, R. T., 1986, "Linear constraints and the efficiency of combined forecasts", *Journal of Forecasting*, vol 5, pp. 31-38.
- [Cle89] Clemen, R. T., 1989, "Combining forecasts: a review and annotated bibliography", *International Journal of Forecasting*, vol 5, pp. 559-583.
- [Doy76] Doyle, P. and Fenwick, I.A., 1976, "Sales forecasting using a combination of approaches", *Long-Range Planning*, 9, pp. 60-64.
- [Ers89] Ersoy, O. K. and Hong, D., 1989, "Parallel, self-organizing hierarchical neural networks", Technical Report - TR-EE-89-56, School of Electrical Engineering, Purdue University, IN.
- [Fah88] Fahlman, S. E., 1988, "An empirical study of learning speed in back-propagation networks, TR, CMU-CS-88-162.
- [Far82] Farmer, D., 1982, "Chaotic attractors of an infinite-dimensional dynamical system", *Physica*, vol 4D, pp. 366-393.
- [Gra84] Granger, C.W.J. and Ramanathan, R., 1984, "Improved methods of combining forecasts", *Journal of Forecasting*, vol 3, pp. 197,-204.
- [Gup87] Gupta, S. and Wilton, P.C., 1987, "Combination of economic forecasts: an extension", *Management Science*, 33, pp. 356-372.
- [Hon91] Hong, D. and Ersoy, O. K., 1991, "Parallel, self-organizing neural networks", Technical Report - TR-EE-91-13, School of Electrical Engineering, Purdue University, IN.
- [Hsu93] Hsu, W., 1993, "Nonlinear and self-adapting methods for prediction", Ph.D. Thesis, School of Electrical Engineering, Purdue University, IN.
- [Kan86] Kang, H., 1986, "Unstable weights in the combination of forecasts", *Management Science*, vol 32, 6, pp. 683-695.
- [Lap87] Lapedes, A. and Farber, R., 1987, "How neural nets work", *Proc of IEEE, Denver Conference on Neural Nets*.
- [Lit91] Littlestone, N. and Warmuth, M.K., 1991, "The weighted majority algorithm", TR UCSC-CRL-91-28, University of California, Santa Cruz, CA.
- [Mac77] Mackey, M.C. and Glass, L., 1977, "Oscillation and chaos in physiological control systems", *Science*, pp. 197-287.
- [Mac93] Mackay, D., 1993, "Bayesian non-linear modeling of the energy prediction competition", University of Cambridge, Cambridge, United Kingdom.

- [Mak82] Makridakis, S. at al., 1982, "The accuracy of extrapolation (time series) methods: results of a forecasting competition", *Journal of Forecasting*, 1, pp. 111-153.
- [Mak83] Makridakis, S. and Winkler, R.L., 1983, "Averages of forecasts: some empirical results", *Management Science*, vol 29, 9, pp. 987-996.
- [New87] Newbold, P. and Granger, C.W.J., 1974, "Experience with forecasting univariate time series and the combination of forecasts", *Journal of the Royal Statistical Society, Series A*, 137, pp. 131-149.
- [The89] Therrien, C., 1989, *Decision Estimate and Classification*, John Willey & Sons, NY.
- [Tho93a] Thome, A.G. and Tenorio, M.F., 1993, "A fuzzy locally sensitive method for cluster analysis", Submitted to *IEEE Transactions on Fuzzy Systems*.
- [Tho93b] Thome, A.G. and Tenorio, M.F., 1993, "Accelerated Learning through a Dynamic Adaptation of the Error Surface", Submitted to *NN magazine*.
- [Tre86] Trenkler, G. and Liski, E.P., 1986, "Note: linear constraints and the efficiency of combined forecasts", *Journal of Forecasting*, 5, pp. 197-202.
- [Win81] Winkler, R.L., 1981, "Combining probability distributions from dependent information sources", *Management Science*, vol 27, 4, pp. 479-488.
- [Win83] Winkler, R.L. and Makridakis, S., 1983, "The combination of forecasts", *Journal of the Royal Statistical Society, Series A*, 146, pp. 150-157.
- [Wol92] Wolpert, D.H., 1992, "Stacked generalization", *Neural Networks*, vol 5, pp. 241-259.
- [Zha92] Zhang, X., Mesirov, J.P and Waltz, D.L., 1992, "Hybrid system for protein secondary structure prediction", *Journal of Molecular Biology*,

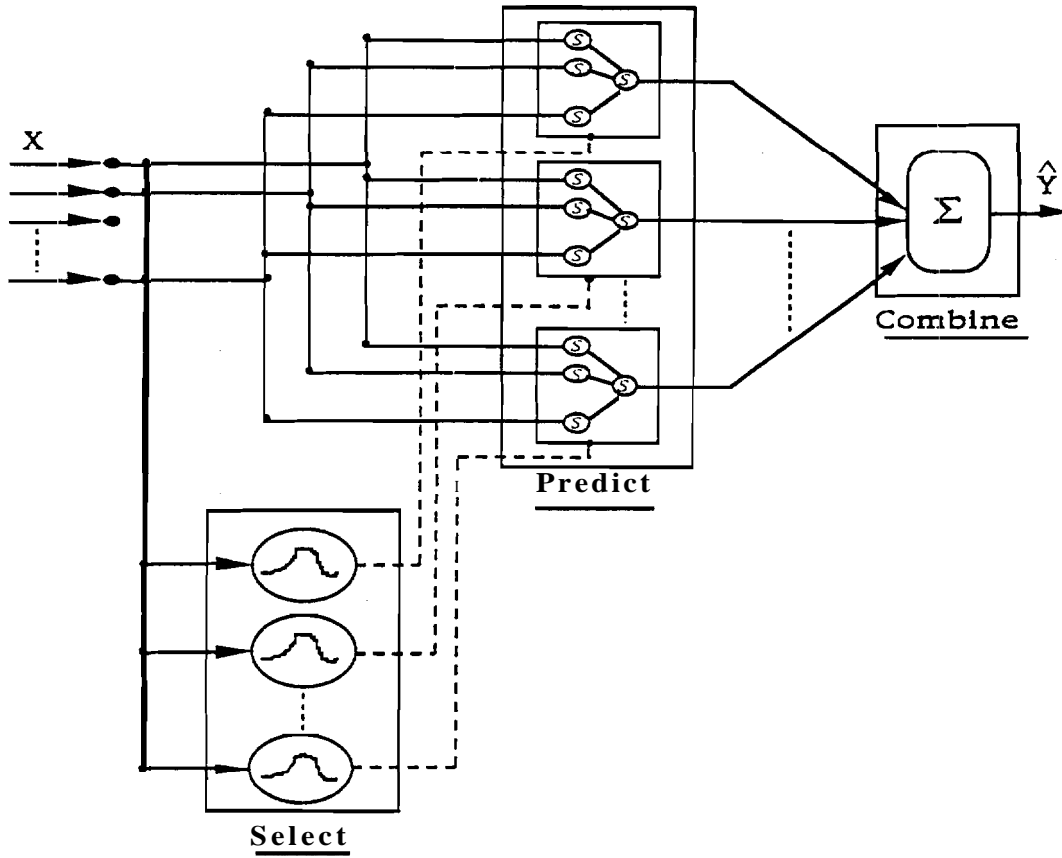


Figure 1 - SMPN architecture

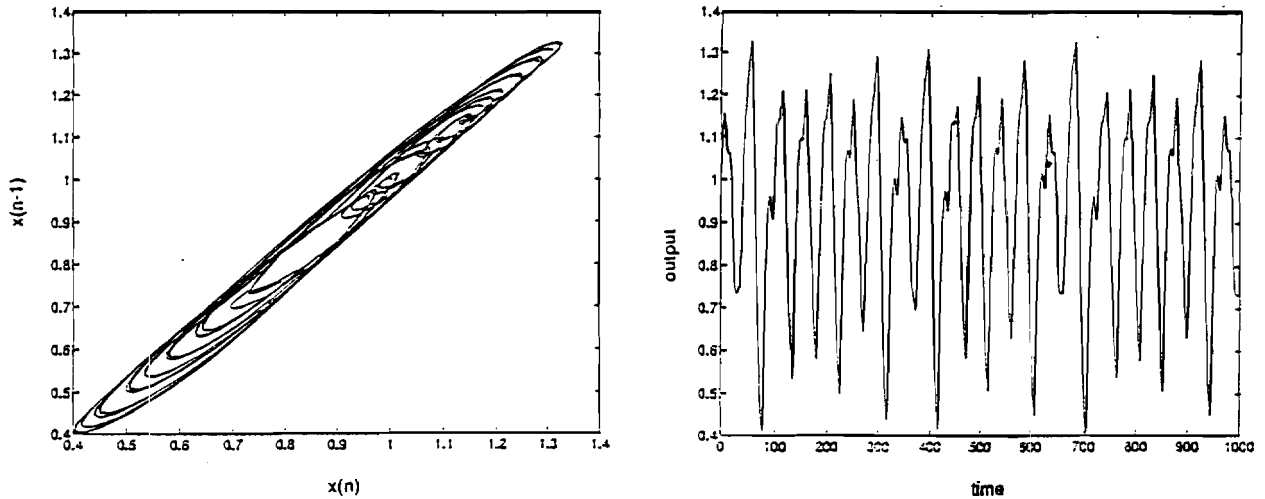


Figure 3 - The Mackey-Glass Time Series, (a) phase portrait, and (b) time dynamics (parameters settings: $a = .2$, $b = .1$, $c = 10$, $A = 17$, $\tau = 6$, and $x(0) = .9$)

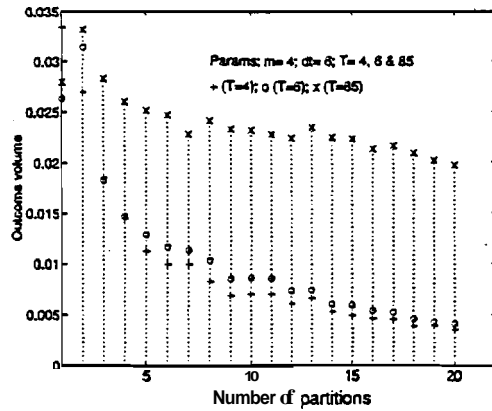


Figure 4 - Mackey-Glass Cluster Analysis (number of partitions x average variance of the desired output within partitions)

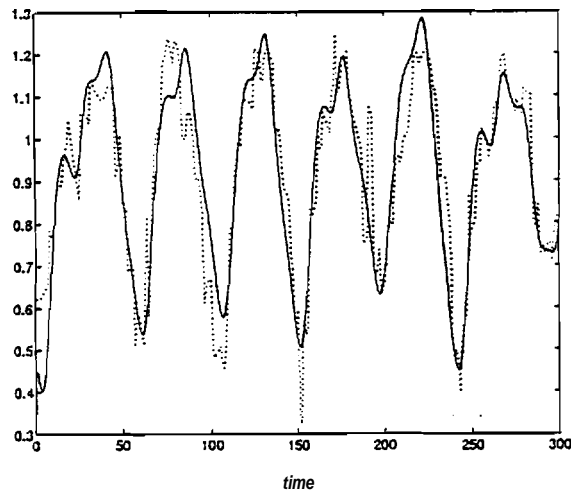


Figure 5 - WTA **Committee** Prediction on Mackey-Glass Time Series T=85 (23 Agents each with 7 units in the hidden layer and one output unit)