

Modelling of River Discharges using Neural Networks derived from Support Vector Regression

K. Y. Choy C. W. Chan

Department of Mechanical Engineering
The University of Hong Kong, Hong Kong, China
mechan@hkucc.hku.hk

Abstract - Neural networks are often used to model complex and nonlinear systems, as they can approximate nonlinear systems with arbitrary accuracy and can be trained from data. Amongst the neural networks, Associative Memory Networks (AMNs) are often used, since they are less computation intensive, and yet good generalization results can be obtained. However, this can only be achieved if the structure of the AMNs is suitably chosen. An approach to choose the structure of the AMNs is to use the Support Vectors (SVs) obtained from the Support Vector Machines. The SVs are obtained from a constrained optimization for a given data set and an error bound. For convenience, this class of AMNs is referred to as the Support Vector Neural Networks (SVNNs). In this paper, the modelling of river discharges with rainfall as input using the SVNN is presented, from which the nonlinear dynamic relationship between rainfall and river discharges is obtained. The prediction of river discharges from the SVNN can give early warning of severe river discharges when there are heavy rainfalls.

I. INTRODUCTION

Data driven modelling and prediction methods have become quite popular for systems that are too complex or too difficult to be described mathematically. Although data driven approaches do not always lead to a better understanding of the physics or the dynamics of the system, they are extremely practical for modelling "data rich" but "theory weak" nonlinear systems. However, one has to be cautious in building and using these models, especially if they are working beyond the data range that has been used for producing them, or there are large outliers in the model.

Neural networks are popular for modelling nonlinear systems, since they have been shown to be universal approximators [1]. In practice, the generalization ability of neural networks depends on the choice of their structure. Associative Memory Networks (AMNs) are neural networks that store information locally using a set of basis functions and associated weights. Though lattice-based AMNs are simple to construct, they have the disadvantage that the number of basis functions in the network increases exponentially as the dimension of the input space increases, leading to the "curse of dimensionality" problem [2]. Not only the memory required to implement the network is large, computation cost in training the network can be excessive. The Radial Basis Function (RBF) networks belong to a class of AMNs for interpolation in an input space with large dimension [3]. An advantage of the RBF networks is that the centres of the basis functions can be

distributed over the input space, instead of on a lattice [4]. For the same number of basis functions, this configuration often leads to better generalization results than with centres on a lattice. Further, the choice of the structure of the networks is less dependent on the dimension of the input space. However, as the performance of the RBF networks depends on the choice of the centres of the basis functions, it is crucial that they are suitably chosen. Several techniques are available in the literature. A popular technique is the Adaptive Spline Modelling of Observation Data (ASMOD) algorithm [5] that searches for the "best" structure from a set of basis functions. The orthogonal-forward-regression routine involves selecting the structure of the network based on the error reduction ratio [6]. In [7], the basis functions are pruned by backward elimination using the orthogonal triangular decomposition to minimize a prediction risk criterion. In [8], a constrained minimization algorithm similar to that in the Support Vector Regression (SVR) [9] is proposed to select the number of basis functions and their respective centres for a given modelling error bound. This class of networks has been referred to as the Support Vector Neural Networks (SVNNs) [8]. The advantage of SVNN is that its structure is selected objectively by a constrained optimization.

In this paper, the SVNN is applied to model the river discharges of Fuji River with rainfall as its input. Fuji River is the third steepest river in Japan. The data set for the study consists of daily river discharges at Kitamatsumo gauging station, and daily rainfall data collected at 10 weather stations in and around the basin from January 1990 to December 1993. The SVNN is constructed from the given data, from which the temporal relationship between rainfall and river discharges is obtained. This result can be used to examine the dynamic effect of rainfall on the river discharges, and to forecast the river discharges from rainfall. Therefore an early warning of severe river discharges can be obtained when there are heavy and prolonged rainfalls.

The organization of this paper is as follows. A brief review of AMNs and Support Vector Machine are given respectively in Sections II and III. The derivation of Support Vector Neural Networks from the Support Vector Machine is presented in Section IV, together with the training algorithm and the training procedure. The modelling of the river discharges with rainfall as an input is described in Section V.

II. ASSOCIATIVE MEMORY NETWORKS

Consider a nonlinear dynamic system given by,

$$y(k) = f(\mathbf{X}(k)) + e(k), \quad (1)$$

where $\mathbf{X}(k) = [y(k-1), \dots, y(k-m_y), u(k), \dots, u(k-m_u)]^T$, $y(k)$ and $u(k)$ are the output and the input; $e(k)$ is a white noise with normal distribution, $e(k) \sim N(0, \sigma^2)$; m_y and m_u are the known orders of the system; $f(\cdot)$ is a well-defined but a-priori unknown nonlinear function. Suppose the nonlinear function $f(\cdot)$ is approximated by an AMN. For an AMN with the centres of the basis functions evenly distributed on an m -dimensional lattice, the number of multivariate basis functions, or the number of weights of the RBF network is given by $n = \prod_{i=1}^m n_i$, which increases exponentially as m increases. This leads to the well-known "curse of dimensionality" problem [10]. Scattered partitioning of the input space can be used to overcome this problem by reducing the number of multivariate basis functions in a RBF network [4]. Without loss of generality, these centres can be chosen to coincide with the input data, $\mathbf{X}(k)$, giving

$$y(k) = \frac{\sum_{i=1}^n \theta_i \mu_i(\mathbf{X}(k))}{\sum_{i=1}^n \mu_i(\mathbf{X}(k))}, \quad (2)$$

where θ_i is the i^{th} weight; and μ_i is the i^{th} Gaussian basis function given by,

$$\mu_i(\mathbf{X}(k)) = \exp\left[-\frac{\|\mathbf{X}(k) - \mathbf{X}'(i)\|^2}{2\gamma^2}\right], \quad (3)$$

where γ and $\mathbf{X}'(i) = [x'_1(i), \dots, x'_m(i)]^T$ are respectively the spread and the centre of the Gaussian function. The centres $\mathbf{X}'(i)$ are chosen from the input data $\{\mathbf{X}(k)\}$, and γ is chosen to ensure a complete coverage of the input space. The i^{th} multivariate basis function is a product of the univariate basis functions for the input variables,

$$\mu_i(\mathbf{X}(k)) = \prod_{j=1}^m \mu_{i,j}(x_j(k)), \quad (4)$$

where $\mu_{i,j}(x_j(k)) = \exp\left[-(x_j(k) - x'_j(i))^2 / 2\gamma^2\right]$. From (3), $\mu_i(\mathbf{X}'(i)) = 1$ for $1 \leq i \leq n$.

III. SUPPORT VECTOR MACHINE

The Support Vector Machine (SVM) [9] for estimating the nonlinear function $f(\cdot)$ is

$$f(\mathbf{X}(k)) = \sum_{i=1}^N (\bar{\alpha}_i - \underline{\alpha}_i) K(\mathbf{X}(k), \mathbf{X}(i)) + b, \quad (5)$$

where N is the length of data; $\bar{\alpha}_i$ and $\underline{\alpha}_i$ are the Lagrange multipliers and b is the bias. Let $\phi(\mathbf{X})$ and \mathbf{w} be respectively the nonlinear regressor and the associated weights. The kernel $K(\mathbf{X}(k), \mathbf{X}(i))$ is given by the linear dot product of the nonlinear regressors, $\phi(\mathbf{X}(k))$ and $\phi(\mathbf{X}(i))$,

$$K(\mathbf{X}(k), \mathbf{X}(i)) = \phi^T(\mathbf{X}(k))\phi(\mathbf{X}(i)) \quad (6)$$

It is assumed that $K(\mathbf{X}(k), \mathbf{X}(i))$ in (6) is symmetric and satisfies the Mercer's condition [11]. The parameters $\bar{\alpha}_i$, $\underline{\alpha}_i$ and b in (5) are estimated by minimizing the regularized risk $L'(\mathbf{w}, b)$, which is a linear combination of the model complexity and the empirical risk,

$$L'(\mathbf{w}, b) = \frac{1}{2} \mathbf{w}^T \mathbf{w} + \frac{C}{N} \sum_{i=1}^N |y(i) - f(\mathbf{X}(i))|_e, \quad (7)$$

where C is the regularization constant; ϵ is the given error bound; $\mathbf{w}^T \mathbf{w} / 2$ represents the model complexity; and $|\cdot|_e = \max\{0, |\cdot| - \epsilon\}$, the ϵ -insensitive loss function. If the magnitude of \cdot is less than ϵ , then $|\cdot|_e$ is zero. The minimization of (7) can be transformed to the constrained minimization [12]:

Minimize:

$$L(\bar{\alpha}, \underline{\alpha}) = \frac{1}{2} \sum_{i,j=1}^N (\bar{\alpha}_i - \underline{\alpha}_i)(\bar{\alpha}_j - \underline{\alpha}_j) K(\mathbf{X}(i), \mathbf{X}(j)) - \sum_{i=1}^N (\bar{\alpha}_i - \underline{\alpha}_i) y(i) + \sum_{i=1}^N (\bar{\alpha}_i + \underline{\alpha}_i) \epsilon, \quad (8)$$

$$\text{subject to } \sum_{i=1}^N (\bar{\alpha}_i - \underline{\alpha}_i) = 0 \quad (9)$$

and the Lagrange multipliers $\bar{\alpha}_i, \underline{\alpha}_i \geq 0$ satisfies,

$$0 \leq \bar{\alpha}_i, \underline{\alpha}_i \leq \frac{C}{N}. \quad (10)$$

In the SVM, each pair of $\{\bar{\alpha}_i, \underline{\alpha}_i\}$ is associated with a kernel having a centre at $\mathbf{X}(i)$, which is retained if $\bar{\alpha}_i - \underline{\alpha}_i \neq 0$, but is removed if both $\bar{\alpha}_i$ and $\underline{\alpha}_i$ are zero. The centres $\mathbf{X}(i)$ of the retained kernels are referred to as the Support Vectors (SVs) [9]. Let

(i) n be the number of SVs,

(ii) $\alpha_j = \bar{\alpha}_i - \underline{\alpha}_i$,

(iii) $K_j(\mathbf{X}(k)) = K(\mathbf{X}(k), \mathbf{X}(i))$,

where $j = 1, \dots, n$, and $\mathbf{X}(i)$ is the j^{th} SV. Rewriting (5) gives,

$$f(\mathbf{X}(k)) = \sum_{j=1}^n \alpha_j K_j(\mathbf{X}(k)) + b. \quad (11)$$

$\{K_j(\mathbf{X}(k))\}$ are Gaussian functions given by (3) with their centres given by the SVs. The model given by (11) is referred to as the Support Vector Regression (SVR) [13], which is the minimum of $L(\bar{\alpha}_i, \underline{\alpha}_i)$ for a given error bound ϵ and C , to be specified by the users.

IV. SUPPORT VECTOR NEURAL NETWORKS

Since the SVR given by (11) can be considered as a two-layer neural network linear in its weights, it is intuitive to reformulate it as a RBF network given by (2) using normalized basis functions, since the kernels of the SVR do not necessarily form a partition of unity [8]. Following [8], the SVR (11) is transformed to the neural network (2),

$$y(k) = \frac{\sum_{i=1}^n \theta_i K_i(\mathbf{X}(k))}{\sum_{i=1}^n K_i(\mathbf{X}(k))} = \sum_{i=1}^n \theta_i N_i(\mathbf{X}(k)), \quad (12)$$

where $N_i(\mathbf{X}(k)) = K_i(\mathbf{X}(k))/\lambda(\mathbf{X}(k))$ is the normalized basis function. For convenience, (12) is referred to as the Support Vector Neural Network (SVNN) [8]. The properties of the SVNN are that the modelling errors have a zero mean and a variance smaller than the square of the error bound ϵ^2 . In contrast, the output of the SVR is biased and the variance of the modelling errors is greater than the SVNN [8].

A. Training Algorithm of SVNN

As the output of the SVNN is linear in the weights of the network, the weights can be estimated by the least-squares method. Rewriting (12) in matrix form gives,

$$y(k) = \mathbf{B}^T(\mathbf{X}(k))\boldsymbol{\theta}, \quad (13)$$

where $\mathbf{B}(\mathbf{X}(k)) = [N_1(\mathbf{X}(k)), \dots, N_n(\mathbf{X}(k))]^T$ and $\boldsymbol{\theta} = [\theta_1, \dots, \theta_n]^T$. The least squares estimate of $\boldsymbol{\theta}$ is obtained by minimizing the cost function,

$$V(\boldsymbol{\theta}) = \frac{1}{N} \sum_{k=1}^N [y(k) - \hat{y}(k)]^2, \quad (14)$$

where $\hat{y}(k)$ is the estimate of the output $y(k)$ computed by

$$\hat{y}(k) = \mathbf{B}^T(\mathbf{X}(k))\hat{\boldsymbol{\theta}}. \quad (15)$$

The least-squares estimate of the weights $\hat{\boldsymbol{\theta}}$ that minimizes (14) is

$$\hat{\boldsymbol{\theta}} = [\boldsymbol{\Phi}^T \boldsymbol{\Phi}]^{-1} \boldsymbol{\Phi}^T \mathbf{Y}, \quad (16)$$

where

$$\boldsymbol{\Phi} = \begin{bmatrix} N_1(\mathbf{X}(1)) & \dots & N_n(\mathbf{X}(1)) \\ N_1(\mathbf{X}(2)) & \dots & N_n(\mathbf{X}(2)) \\ \vdots & & \vdots \\ N_1(\mathbf{X}(N)) & \dots & N_n(\mathbf{X}(N)) \end{bmatrix}, \text{ and} \quad (17)$$

$$\mathbf{Y} = \begin{bmatrix} y(1) \\ y(2) \\ \vdots \\ y(N) \end{bmatrix} \quad (18)$$

B. Training Procedure of SVNN

The training procedure for the SVNN is summarized below.

- Step 1** Normalize the input-output data such that they are in the range of 0 and 1, and choose the spread γ of the Gaussian function given by (3) to give a good coverage of the input space.
- Step 2** Select ϵ and C , and obtain the SVs of the SVM by minimizing $L(\bar{\alpha}_i, \underline{\alpha}_i)$ given by (8) subject to the constraints (9) and (10).
- Step 3** Obtain $\mathbf{B}(\mathbf{X}(k)) = [N_1(\mathbf{X}(k)), \dots, N_n(\mathbf{X}(k))]^T$ from (12) using the SVs, and compute the least-squares estimate of the weights $\hat{\boldsymbol{\theta}}$ by (16).
- Step 4** Evaluate the modelling errors of the SVNN. Choose another ϵ , and repeat Steps 3 and 4.

From experience, the choice of C does not have too much effect on the optimization result. However, the choice of ϵ has a direct impact on the number of SVs, as it governs the precision of the approximation. Therefore, it is necessary to choose a few ϵ and obtain the corresponding SVNNs. The choice of the 'best' one is a compromise between the variance reduction in the modelling errors, and the number of SVs used.

V. RESULTS

A. The modelling data

Fuji River, which runs through mountainous terrain, is the third steepest river in Japan. The basin is located between the latitudes $35^\circ 07' - 35^\circ 46' \text{ N}$, and longitudes $138^\circ 12' - 138^\circ 39' \text{ E}$, and the river originates at an elevation of 2685 m. The catchment area of the river is 3570 km^2 over a length of 128 km [14]. The data set used in the study consists of daily river discharges at Kitamatsuno gauging station, which is 10.7 km from the river mouth with a catchment area of 3540 km^2 , daily rainfall data collected at 10 weather stations in and around the basin (Kamikuishiki, Nakatomi, Kawaguchiko, Yamanaka,

Nanbu, Ooizumi, Nirasaki, Kofu, Katsunuma and Ootsuki) for the period from January 1990 to December 1993. There are 1461 sets of daily river discharges data $y(k)$ (m^3/s) and daily rainfall data $u(k)$ (mm) from each of its surrounding weather stations. Following the modelling procedure described in Section IV, these data are normalized with values within 0 and 1, giving, which are denoted by $\underline{y}(k)$ and $\underline{u}(k)$. The input $\mathbf{X}(k)$ of the SVNN is selected by choosing first m_y and m_u , based on the autocorrelation functions $\rho_y(\lambda)$ of $y(k)$ and the cross-correlation functions $\rho_{yu}(\lambda)$ between $y(k)$ and $u(k)$,

$$\rho_y(\lambda) = \frac{1}{(N-\lambda)\sqrt{r_y r_y}} \sum_{k=\lambda+1}^N [y(k) - \bar{y}][y(k-\lambda) - \bar{y}], \quad (19)$$

$$\rho_{yu}(\lambda) = \begin{cases} \frac{1}{(N-\lambda)\sqrt{r_y r_u}} \sum_{k=\lambda+1}^N [y(k) - \bar{y}][u(k-\lambda) - \bar{u}]; & \lambda \geq 0 \\ \frac{1}{(N+\lambda)\sqrt{r_y r_u}} \sum_{k=1}^{N+\lambda} [y(k) - \bar{y}][u(k-\lambda) - \bar{u}]; & \lambda < 0 \end{cases}, \quad (20)$$

where $r_y = \frac{1}{N} \sum_{k=1}^N [y(k) - \bar{y}]^2$; $r_u = \frac{1}{N} \sum_{k=1}^N [u(k) - \bar{u}]^2$; λ is the

lag; $N = 1461$; $\bar{\bullet}$ denotes the mean of \bullet . The cross-correlation functions between the river discharges and the rainfall data from each of the ten weather stations are computed. As the cross-correlation functions for 5 out of 10 weather stations are much smaller than the other 5, they are not included in the study. Therefore, the average rainfall is computed from the remaining 5 weather stations, and is used to compute the cross-correlation functions between the river discharges and the rainfall. From the autocorrelation and the cross-correlation functions shown in Figs. 1 and 2, $\rho_y(1)$, $\rho_y(2)$, $\rho_{yu}(0)$ and $\rho_{yu}(1)$ are much larger than the other functions. Therefore, m_y is chosen to be 2 and m_u to be 1, giving $\mathbf{X}(k) = [y(k-1), y(k-2), u(k), u(k-1)]^T$.

The first 1000 normalized data are used as training data. To reduce the computing time, 53 data are further selected for finding the SVs, and are selected as follows. Since most of the training data are within the range (0, 0.1), this range is divided into 10 equal intervals and 3 data are randomly drawn from each interval, giving 30 data. The remaining 23 data include the smallest one and all those in the range [0.1, 1].

B. Modelling using the SVNN

The spread γ of the Gaussian function is chosen to be 1, C to be 300, and ϵ to be 0.08, 0.09 and 0.1, from which three SVRs are obtained, and the number of SVs in the respective SVR is 12, 9 and 8. The SVR with 9 SVs corresponding to $\epsilon = 0.09$ is shown in Fig. 3 with the SVs marked by circles. From these SVs, $\mathbf{B}(\mathbf{X}(k))$ and Φ are computed, and the least squares

estimate of the weights $\hat{\theta}$ is then computed from (16). From the estimated weights, the estimated output $\hat{y}(k)$ is computed from (15). The mean of the modelling errors of all trial models is approximately zero, and the variance of the modelling errors for each SVR are: 76.52², 88.68², and 91.96². The SVNN with structure determined by the SVR with $\epsilon = 0.09$ is preferred, since the reduction in the variance of the modelling errors for the one based on $\epsilon = 0.08$ is achieved by the addition of 3 more SVs.

The modelling errors for the selected SVNN are shown in Fig. 4. The SVNN is then used to predict the river discharges for the periods 1001 to 1461. The variance of the prediction errors, which are shown in Fig. 5, is 69.15², much smaller than the variance of the modelling errors. This is because there are a few large outliers in the modelling errors that lead to a larger variance.

C. Intervention

The outliers in the modelling errors are arisen from the way the data are collected. As the rainfall data are aggregated over a day, and the river discharges are measured at some specific time of the day, it is possible that a heavy rainfall may take place after the river discharges are measured. To obtain a more reliable model, it is necessary to remove these outliers. Since there are 6 outliers from the modelling errors as shown in Fig. 4, 6 intervention variables are used. Each one is a zero vector, except in the position where the outlier occurs, which is set to 1. These vectors are adjoined to Φ , and the number of parameters in the SVNN becomes 15. The magnitude of these outliers is given by the least squares estimate of the parameters associated with the intervention variables. Now, the normalized river discharges are adjusted by subtracting these parameters at the period when the outliers occurred. The modelling procedure is repeated with $\gamma = 0.7$, $\epsilon = 0.09$, and $C = 300$. A SVR with 10 SVs is obtained. The SVs shown in Fig. 6 are those data as marked by circles.

With the new 10 SVs, the SVNN is re-estimated using the least squares method and the original data with the intervention variables. The variance of the modelling errors is 44.40², and from Fig. 7, there are no obvious outliers in the modelling errors.

D. Relationship between Rainfall and River Discharges

To analyze the effect of rainfall on river discharges, 2 cases are simulated from the SVNN. In case 1, the rainfall $\underline{u}(1)$ is set to some values between 0 and 1, say 0.7, whilst the rainfall in the following days are set to zero. $\underline{y}(-1) = \underline{y}(0) = 0.012$, the mean of the river discharges for periods when there was no rainfall. The estimated output for the SVNN with intervention is shown in Fig. 8. In case 2, the rainfall $\underline{u}(1)$ is fixed at 0.4,

whilst $\underline{u}(2)$ is set to some values between 0 and 0.8 and the rainfall in the following days are set to zero. $\underline{y}(-1) = \underline{y}(0) = 0.012$, same as before. The estimated output for the SVNN with intervention is shown in Fig. 9.

VI. CONCLUSIONS

In this paper, the modelling of river discharges of Fuji River using the Support Vector Neural Networks is described. The ability of the SVNN to predict river discharges is demonstrated using the test data. From the SVNN, the nonlinear relationship between rainfall and river discharges is obtained, from which river discharges can be forecasted from rainfall. This relationship is useful to provide an early warning of severe river discharges when there are heavy and prolonged rainfalls.

ACKNOWLEDGMENT

We wish to thank Dr. Huaxia Yao for kindly providing the river discharges data of Fuji River and the rainfall data.

REFERENCES

- [1] K. Hornik, M. Stinchcombe and H. White, "Multilayered feedforward networks are universal approximators", *Neural Networks*, vol. 2, pp. 359-366, 1989.
- [2] C.J. Harris, M. Brown, K.M. Bossley, D.J. Mills, and M. Feng, "Advances in neurofuzzy algorithms for real-time modelling and control," *Eng. App. AI*, vol. 9(1), pp. 1-16, 1996.
- [3] M.J.D. Powell, "The theory of radial basis functions approximation in 1990," in *Advances in numerical analysis, vol. 2: Wavelets, subdivision algorithms and radial basis functions*, W.A. Light, Ed. New York: Oxford University Press, 1992, pp. 105-210.
- [4] J.-S.R. Jang, C.-T. Sun, and E. Mizutani, *Neuro-fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*. Prentice Hall, 1997.
- [5] T. Kavli and E. Weyer, "On ASMOD - an algorithm for building multivariate spline models," in *Neural Networks Engineering in Control*. Springer-Verlag, 1995, pp. 83-104.
- [6] S. Chen and S.A. Billings, "Neural networks for nonlinear dynamic system modelling and identification," *Inter. J. Control*, vol. 56, pp. 319-346, 1992.
- [7] X. Hong and S.A. Billings, "Givens rotation based fast backward elimination algorithm for RBF neural network pruning," *IEE Proc. Control Theory App.*, vol. 144(5), pp. 381-384, 1997.
- [8] W.C. Chan, C.W. Chan, K.C. Cheung, and C.J. Harris, "On the modelling of nonlinear dynamical systems using support vector neural networks," *Eng. App. AI*, vol. 14, pp. 105-113, 2001.
- [9] V. Vapnik, *The nature of statistical learning theory*. New York: Springer-Verlag, 1995.
- [10] M. Brown and C.J. Harris, *Neurofuzzy Adaptive Modelling and Control*. Prentice Hall, 1994.
- [11] C. Cortes and V. Vapnik, "Support vector networks," *Mach. Learn.*, vol. 20, pp. 273-297, 1995.
- [12] D.P. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods*. New York: Academic Press, 1982.
- [13] B. Schölkopf, K. Sung, C.J.C. Burges, F. Girosi, P. Niyogi, T. Poggio, and V. Vapnik, "Comparing Support Vector Machines with Gaussian kernels to Radial Basis Function classifiers," *IEEE. Trans. Signal Processing*, vol. 45(11), pp. 2758-2765, 1997.
- [14] A.W. Jayawardena, K. Takeuchi, and B. Machbub, *Catalogue of Rivers for Southeast Asia and the Pacific*, vol. II. A UNESCO-IHP publication, 1997, pp. 285.

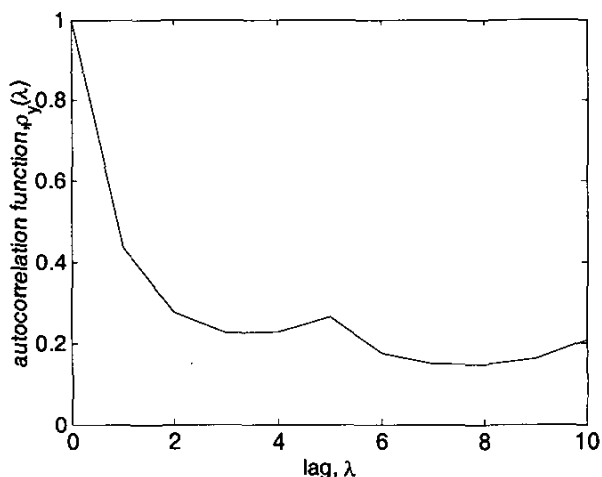


Fig. 1. Autocorrelation function of river discharges data

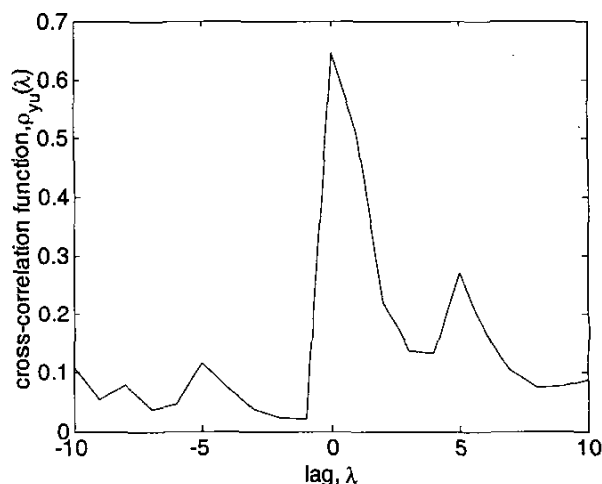


Fig. 2. Cross-correlation function between river discharges data and rainfall data

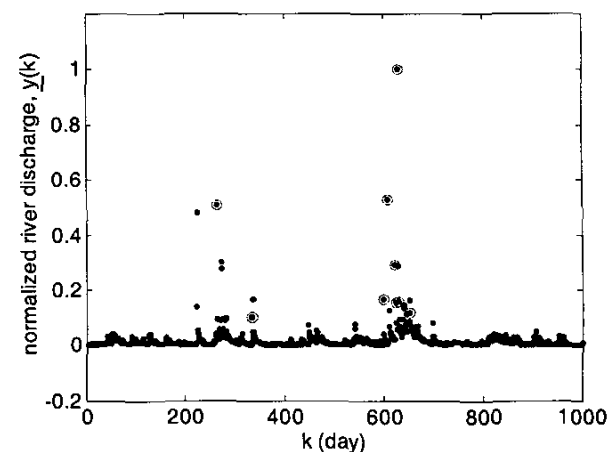


Fig. 3. The 1000 training data and 9 SVs

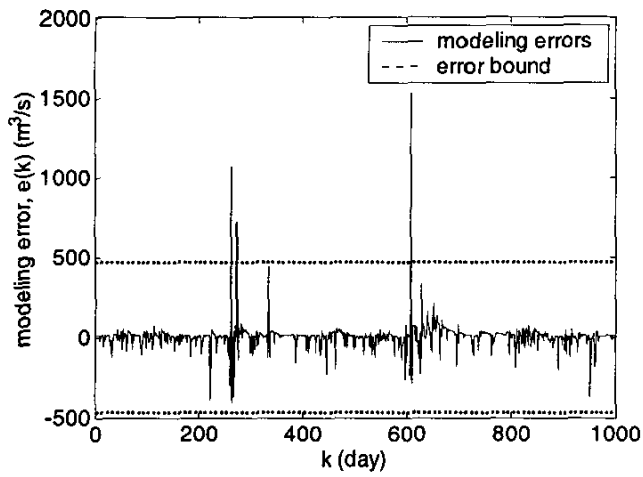


Fig. 4. The modeling errors of the SVNN (9 SVs)

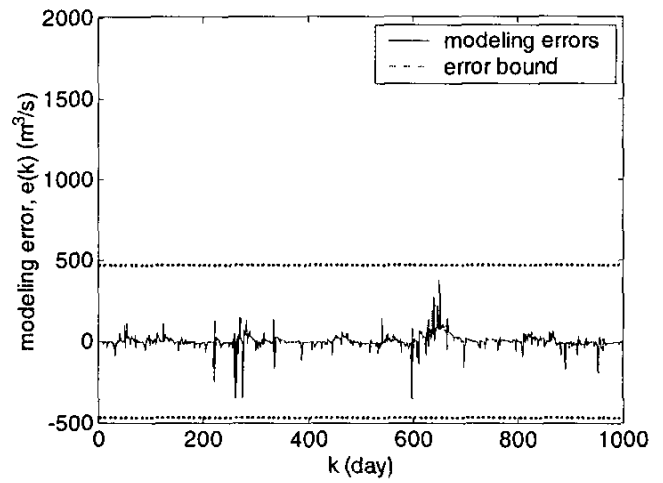


Fig. 7. The modeling errors of the SVNN with intervention

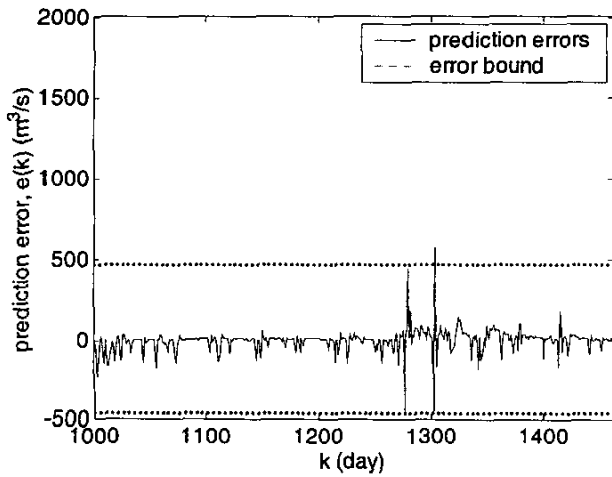


Fig. 5. The predictions errors of the SVNN (9 SVs)

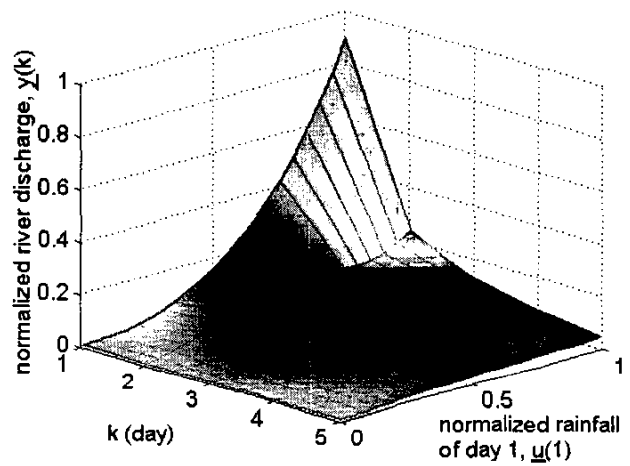


Fig. 8. River discharges as a function of rainfall

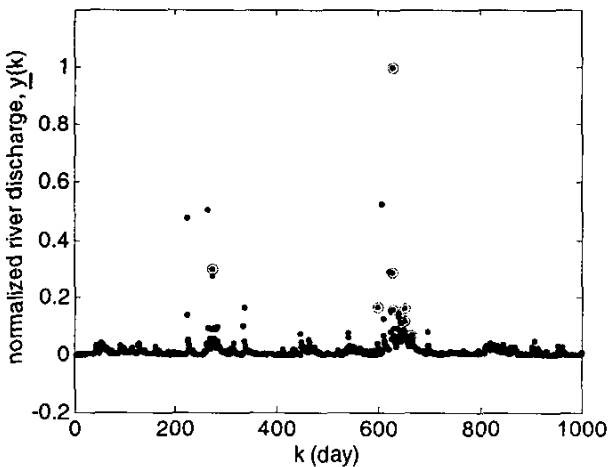


Fig. 6. The 1000 training data and 10 SVs (intervention)

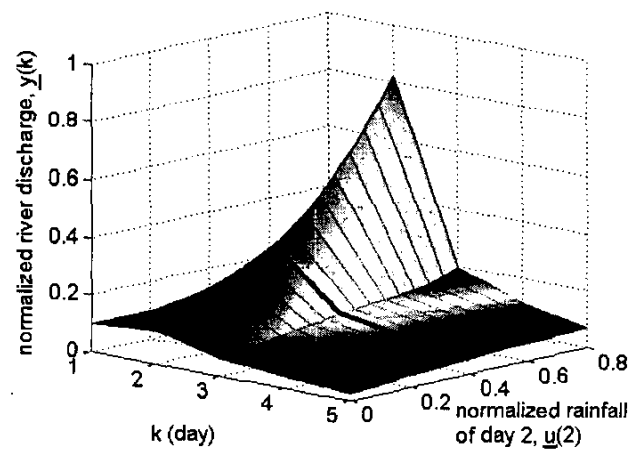


Fig. 9. River discharges for 2 consecutive raining days