Stochasticity, Nonlinearity and Forecasting of Streamflow Processes

Stochasticity, nonlinearity and forecasting of streamflow processes

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof.dr.ir. J.T. Fokkema,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen

op woensdag 24 mei 2006 om 10:00 uur

door

Wen WANG

Master of Science in geomorphology and Quaternary environment,
Nanjing University, China
geboren te Jiangyan, Jiangsu, China

Dit proefschrift is goedgekeurd door de promotor:
Prof. drs. ir. J.K. Vrijling

Toegevoegd promotor:
Dr. ir. P.H.A.J.M. van Gelder

Samenstelling promotiecommissie:

Rector Magnificus	voorzitter
Prof. drs. ir. J.K. Vrijling	Technische Universiteit Delft, promotor
Dr. ir. P.H.A.J.M. van Gelder	Technische Universiteit Delft, toegevoegd promotor
Prof. Z. W. Kundzewicz	Polish Academy of Sciences
Prof. L. Garrote	Universidad Politécnica de Madrid
Prof. dr. ir. N. van de Giesen	Technische Universiteit Delft
Prof. Y.F. Chen	Hohai University, China
Dr. P. Reggiani	WL|Delft Hydraulics

Prof. Liu Xinren (Hohai University) heeft als begeleider in belangrijke mate aan de totstandkoming van het proefschrift bijgedragen.

ISBN 1-58603-621-1

Keywords: *Streamflow; hydrological forecast; time series analysis; long memory; stationarity; autoregressive conditional heteroskedasticity; nonlinearity; chaos; predictability; artificial neural network; data-driven model*

Published and distributed by IOS Press under the imprint Delft University Press

Publisher & Distributor
IOS Press
Nieuwe Hemweg 6b
1013 BG Amsterdam
Netherlands
fax:+31-20-687 0019
email: info@iospress.nl

Distributor in the USA and Canada
IOS Press, Inc.
4502 Rachael Manor Drive
Fairfax, VA 22032
USA
fax: +1-703-323 3668
e-mail: sales@iospress.com

PRINTED IN THE NETHERLANDS

CONTENTS

Summary

Streamflow forecasting is of great importance to water resources management and flood defence. On the other hand, a better understanding of the streamflow process is fundamental for improving the skill of streamflow forecasting. The methods for forecasting streamflows may fall into two general classes: process-driven methods and data-driven methods. Equivalently, methods for understanding streamflow processes may also be broken into two categories: physically-based methods and mathematically-based methods.

This thesis focuses on using mathematically-based methods to analyze stochasticity and nonlinearity of streamflow processes based on univariate historic streamflow records, and presents data-driven models that are also mainly based on univariate streamflow time series. Six streamflow processes of five rivers in different geological regions are investigated for stochasticity and nonlinearity at several characteristic timescales (i.e., one day, one month, 1/3 month, and one year). But only the streamflows of the upper Yellow River in northern China are considered for forecasting.

Firstly, several important aspects of the stochasticity, including trend, seasonality, stationarity, and long-memory, are examined in this study.

In the context of global warming, how hydrological processes are impacted is an issue of wide concern. The analyses of two streamflow processes of the Yellow River show that there is no obvious trend in the average annual flow process of the upper reach at Tangnaihai (TNH) from 1956 to 2000, whereas the streamflow process recorded at Tongguan in the middle reaches exhibits significant declining trend. No significant decline is found in the precipitation processes, on the other hand, it is found that the lower the reaches of the Yellow River, the more significant the downward trend. This indicates that the impact of climate warming on the river flow processes of the Yellow River are far less significant than anthropogenic influences.

Stationarity is required for the construction of many types of models and for the application of many data analysis techniques. In the present study, ADF unit root test (Dickey and Fuller, 1979; Said and Dickey, 1984) and KPSS test (Kwiatkowski et al., 1992), which originate from the econometrics, are introduced to test for the nonstationarity in hydrological time series. It is found that the smaller the timescale of the streamflow process is, the more likely it tends to be nonstationary.

Seasonality is a common feature in hydrological time series. Before fitting a time series model, it is popular to deseasonalize the hydrological time series by subtracting the seasonal means and dividing by seasonal standard deviations. But it is shown that, the deseasonalization procedure can remove the seasonality in the mean and variance, but not the seasonality in the autocorrelation structure.

Many studies have shown that, many hydrological processes, especially streamflow processes, have long-memory property. This is also confirmed by the present study. Furthermore, the investigation of the long-memory phenomenon in streamflow processes at different timescales

shows that, with the increase of timescale, the intensity of long-memory decreases. Generally speaking, according to the test results, all daily flow series exhibit strong long-memory; 1/3-monthly flow series may be considered as weak long-memory processes; monthly series may be considered as short memory processes or at most processes of very weak long-memory.

Secondly, the nonlinearity of streamflow processes is investigated. It is well accepted that watershed systems are nonlinear. Correspondingly, the output of watershed systems, streamflow processes, may also exhibit nonlinearity. But there is no well-accepted methods to quantify the degree of nonlinearity. What is less-accepted or even controversial is the sources or the nature of the nonlineairty, such as whether hydrological processes are deterministic chaotic processes or not. In the present study, besides the general nonlinearity, two special types of nonlinearity are examined in more details, i.e., conditional heteroskedasticity and chaos.

The BDS test method (Brock et al., 1996) is introduced to test for the existence of nonlinearity in streamflow processes. It is found that the shorter the timescale, the stronger the nonlinearity. All annual series are linear, whereas all daily streamflow processes are strongly nonlinear.

Besides those well-recognized physical sources, such as the mechanisms involved in the rainfall–runoff transformation, some other sources can be identified as viewed from the streamflow time series itself. Asymmetric seasonality in the mean and variance of raw (or log-transformed) streamflow processes and the seasonality in the variance of the pre-whitened streamflow processes (i.e., residual series) play a role in the exhibition of nonlinearity. The degree of nonstationarity has a significant impact on the test for nonlinearity. Nonstationarity or the stationarity of low significance level may give rise to positive results (i.e., the existence of nonlinearity) in nonlinearity test. The conditional heteroskedasticity is an important source of the nonlinearity of streamflow processes, so is long-memory if it is also viewed as a type of nonlinearity.

The existence of conditional heteroskedasticity is detected in the residual series from linear models fitted to the daily and monthly streamflow processes of the upper Yellow River. It is shown that the ARCH effect is fully caused by seasonal variation in the variance for monthly flows, but seasonal variation in variance only partly explains the ARCH effect for daily streamflow. To capture the ARCH effect in the daily streamflow processes so as to improve the estimate of forecast uncertainty, the ARMA-GARCH error model with seasonal standard deviations is proposed, in which an ARMA model is used to model the mean behaviour and a GARCH model to model the ARCH effect in the residuals from the ARMA model.

Whether or not hydrological processes are deterministic chaotic processes is a widely concerned and quite controversial issue in the last decade. No finite correlation dimension is found for all the streamflow series in the present study with correlation exponent method. Because the existence of finite correlation dimension is critical for verifying the existence of chaos, therefore, while nonlinear behaviour seemed to be present with different intensity at various time scales, the dynamics would not seem to be associable to the presence of low dimensional chaos. On the other hand, even if we found the evidence of the existence of chaos in a time series, it does not necessarily mean determinism. Experiments with three well-known chaotic systems (i.e., Henon map, Ikeda map, discretized Mackey-Glass flow) show that chaos could be stochastic.

Thirdly, the issue of streamflow forecasting is addressed with data-driven methods in the present study.

When forecasting streamflow processes, both the forecasting model (or method) and the model performance measure are needed. A seasonally-adjusted *CE* (*SACE*) is proposed to measure how good forecasts from a model are better than seasonal mean values, instead of the overall mean value as with popular model performance measure Coefficient of efficiency (*CE*). The measure *SACE* is more suitable than *CE* for evaluating seasonal processes.

Two groups of data-driven models are used for forecasting daily streamflows in the study: (i) ARMA-type models, including the ARMA(20,1) model, the ARFIMA(7,d,0) model, and the periodic AR (PAR) model; and (ii) ANN-type models, including the normal multi-layer perceptron (MLP) ANN model, the cluster-based ANN model, the threshold-based ANN model, the period-based ANN model (or, periodic ANN model, PANN) and its variations (i.e., soft PANN and hard PANN). Among these models, PAR and PANN are proposed for modelling daily streamflow processes in this study. They are fundamentally a group of AR models or ANN models. Each AR model or ANN model are fitted to a specific season partition, so that the seasonality in daily streamflow processes is better captured. Furthermore, to combine the strength of these dynamically different models, four forecast combination methods are adopted in this study: simple average method (SAM), rollingly-updated weighted average method, semi-fixed weighted average method, and modular semi-fixed weighted average method. The comparison of the forecast performances of these models and methods shown that, despite of the limitation of univariate streamflow time series, in terms of the *SACE* measure, satisfactory forecasts can be made for lead times of up to 5 days ($SACE > 0.8$). The overall performance of PANN is the best for one-day ahead forecasts. SAM generally performs best among the four competitive combination methods, and it outperforms all the five individual models for forecasts of up to 4 days ahead (except for the PANN model for one-day ahead forecasts). The ARFIMA model performs the best for long lead times (longer than four days).

Generalization is a critical issue in constructing an ANN model when the size of training data is not big enough. The comparison of two techniques to achieve generalization, i.e., the Bayesian regularization (BR) method and the cross-validated early stopping (CVES) method, shows that the BR method outperforms the CVES method in general for making one step ahead forecasts for univariate time series. In the case of building ANN models where multiple explanatory variables are included, the advantage of the BR method is also confirmed. Therefore, when the speed of training is not a major concern, the BR method is recommended for making streamflow forecasts.

Predictability is an important aspect of the dynamics of hydrological processes. However, very few studies are conducted about the issue of predictability in the literature, and they are conducted based on the multiple explanatory variables. A univariate time series based approach is proposed in the present study, in which the predictability of a process is defined as the predictable horizon for which the prediction is no better than the mean value for a stationary process or the seasonal mean value for a seasonal process. With such a definition, the predictability is easily comparable among different streamflow processes. Investigation of the predictabilities of a number of streamflow series at different basin scales shows that, in general, the larger the basin scale, the better the predictability.

Samenvatting

Het voorspellen van rivierafvoeren is van groot belang voor watermanagement en hoogwaterbestrijding. Omgekeerd is een betere doorgronding van het afvoerhydrologisch proces fundamenteel om voorspellingen te verbeteren. Methoden voor het voorspellen van rivierafvoeren kunnen in twee klassen worden ingedeeld: procesgeoriënteerde en datageoriënteerde methoden. Analoog hieraan kunnen in het hydrologische proces ook twee categorieën onderscheiden worden, namelijk fysische methoden en statistische methoden.

Dit proefschrift maakt voornamelijk gebruik van statistische methoden voor analyse van stochastische en niet-lineaire effecten van hydrologische processen gebaseerd op univariate afvoergegevens, en presenteert datageoriënteerde modellen die tevens voornamelijk gebruik maken van univariate afvoergegevens. Zes hydrologische processen van vijf rivieren in verschillende geologische gebieden zijn stochastisch en niet-lineair onderzocht bij verschillende tijdschalen (per dag, maand, ⅓ maand en een jaar). Betreffende het voorspellen is alleen de afvoer van de Gele rivier in Noord China beschouwd.

Als eerste zijn in deze studie de verscheidene belangrijke aspecten van de stochasticiteit, inclusief trends, seizoensinvloeden, stationairiteit, en lange termijn aspecten onderzocht.

In de context van globale opwarming van de aarde, is op dit moment de impact ervan op het hydrologische proces een belangrijk onderwerp. De analyse van twee afvoerprocessen van de Gele rivier laten zien dat er geen verandering is in het gemiddelde jaarlijkse afvoerproces van de bovenloop bij Tangnaihai van 1956 tot 2000, terwijl het afvoerproces waargenomen bij Tongguan in het midden van het stroomgebied een significante afnemende trend laat zien. Er is geen significante daling gevonden in de neerslag, aan de andere kant wordt er gevonden dat, hoe meer benedenstroom van de Gele rivier, hoe duidelijker de neerwaartse trend wordt. Dit duidt erop dat de impact van de klimaatverandering op het rivierafvoerproces van de Gele rivier veel minder significant is dan antropogene invloeden.

Stationairiteit is vereist voor het maken van vele typen modellen en voor de applicatie van vele data analyse technieken. In deze studie de ADF eenheidswortel test (Dickey en Fuller, 1979; Said en Dickey, 1984) en de KPSS test (Kwiatkowski et al., 1992), die afstammen uit de econometrie, zijn geïntroduceerd als test voor niet stationairiteit van hydrologische tijdreeksen. Gevonden is dat hoe kleiner de tijdsschaal van het afvoerproces is des te meer instationairiteit er optreedt.

Seizoensinvloeden zijn een algemeen kenmerk van hydrologische tijdswaarnemingen. Voorafgaand aan modellering van een tijdswaarneming, is het gebruikelijk om de hydrologische gegevens te ontdoen van seizoensinvloeden door de seizoensgemiddelden af te trekken en te delen door de standaarddeviatie van de seizoenen. Aangetoond is dat deze procedure de seizoensafhankelijkheid van het gemiddelde en variantie kan verwijderen, maar niet de seizoensinvloed van de autocorrelatie.

Vele studies hebben aangetoond dat vele hydrologische processen, in het bijzonder rivierafvoer processen, langere termijn eigenschappen hebben. Dit is eveneens in deze studie bevestigd. Voorts laat onderzoek naar de langere termijnverschijnselen in rivierafvoer processen bij verschillende tijdschalen zien dat bij toename van de tijdsschaal de intensiteit van de langere termijn afneemt. Volgens onderhavige testresultaten kan in het algemeen gesteld worden dat alle dagafvoeren sterke langere termijnverschijnselen vertonen; ⅓ maandelijkse afvoerwaardes mogen als zwakke langere termijn processen worden beschouwd; maandelijkse waardes mogen als korte termijn processen of hoogstens als zeer zwakke langere termijn processen beschouwd worden.

Als tweede is de niet-lineairiteit van het afvoerproces onderzocht. Het is algemeen geaccepteerd dat stroomgebieden niet-lineair zijn. Als gevolg hiervan is de uitvoer van rivierafvoer processen eveneens niet-lineair. Maar er zijn geen algemeen acceptabele methoden om de graad van niet-lineairiteit vast te stellen. Nog minder geaccepteerd en zelfs controversieel is de oorzaak of de bron van de niet-lineairiteit, zoals of het hydrologische proces een deterministisch chaotisch proces is of niet. In deze studie zijn naast algemene niet-lineairiteit twee speciale types in dieper detail bestudeerd, namelijk conditionele heteroskedasticiteit en chaos.

De BDS onderzoeksmethode (Brock et al., 1996) is geïntroduceerd als test voor de existentie van niet-lineairiteit in afvoer processen. Gevonden is dat hoe korter de tijdschaal, des te sterker de niet-lineairiteit is. Alle jaarwaarnemingen zijn lineair, echter alle dagelijkse afvoerprocessen zijn sterk niet-lineair.

Naast goedherkenbare fysische bronnen zoals de mechanismen die bepalend zijn voor de regenwaterafvoer modellering, kunnen sommige andere bronnen geïdentificeerd worden gezien vanuit afvoertijdswaarnemeningen zelf. A-symmetrische seizoensinvloeden in het gemiddelde en variantie van ruwe (of log-getransformeerde) afvoerprocessen en de seizoensinvloed in de variantie van pre-whitened rivierafvoer processen (zoals residuele waarden) spelen een belangrijke rol in het vaststellen van niet-lineairiteit. De graad van instationairiteit heeft een significante invloed op het niet-linieairiteitsonderzoek. Niet-stationairiteit of stationairiteit van laag significantieniveau kunnen aanleiding geven tot positieve resultaten (namelijk het bestaan van niet-lineairiteit) in niet-linieaire onderzoeken. De conditionele heteroskedasticiteit is een belangrijke bron voor het niet-lineaire afvoerproces, zoals het langere termijnproces is als het eveneens beschouwd wordt als een niet-lineair type.

Het bestaan van de conditionele heteroskedasticiteit is gevonden in de residuele waarnemingen van lineaire modellen, vertaald naar dagelijkse en maandelijkse afvoerprocessen van de bovenloop van de Gele rivier. Gebleken is dat het ARCH-effect volledig veroorzaakt wordt door seizoensvariatie in de variantie van maandsafvoeren, maar dat dit slechts deels het ARCH-effect van dagelijkse afvoer verklaard. Ter bepaling van het ARCH-effect in de dagelijkse afvoerprocessen om zo een verbetering van onzekerheden te schatten, is het ARMA-GARCH foutenmodel met seizoensstandaarddeviaties voorgesteld, waarin het ARMA model is gebruikt ter modellering van het gemiddelde gedrag en het GARCH model ter bepaling van het ARCH effect in het residu van het ARMA model.

Of hydrologische processen nu wel of niet deterministisch chaotische processen zijn, is een breed beschouwd en vrij controversieel onderwerp het laatste decennium. In deze studie is met de exponent correlatie methode geen eindige correlatie dimensie gevonden voor alle afvoerwaarnemingen. Daar de existentie van eindige correlatie dimensie kritisch is voor het verifiëren van existentie van chaos, terwijl niet-lineair gedrag lijkt op te treden met

verschillende intensiteit bij verschillende tijdsschalen, lijkt de dynamica niet gecombineerd te kunnen worden met de aanwezige lage dimensionale chaos. Zelfs als er bewijs gevonden kon worden voor de existentie van chaos in de tijdreeksen, betekent het niet direct determinisme. Experimenten met drie goed bekende chaotische systemen (bijv. Henon map, Ikeda map, Discretized Mackey-Glass stroming) laten zien dat chaos stochastisch zou kunnen zijn.

Als derde is in deze studie het onderwerp van afvoervoorspelling gekoppeld aan de datageoriënteerde methoden.

Voor het voorspellen van afvoerprocessen zijn zowel het voorspelmodel (of methode) als uitgevoerde metingen nodig. Een aangepaste seizoensvoorspeller SACE is voorgesteld voor het meten in hoeverre de voorspellingen van een model beter zijn dan de seizoensgemiddelde waarden, ten opzichte van de overall gemiddelde waarde die gemeten is met het populaire schattingsmodel Coëfficiënt van Efficiëntie (CE). De SACE maat is beter geschikt dan de CE voor evaluatie van seizoensprocessen.

Twee groepen van datageoriënteerde modellen zijn gebruikt voor voorspelling van dagelijkse afvoeren in de studie: 1) ARMA-type modellen, inclusief het ARMA(20,1) model, het ARFIMA(7,d,0) model, en het periodiek AR (PAR) model; en 2) de ANN-type modellen, inclusief de normaal multi-lagen perceptron (MLP) ANN model, het clustergebaseerde ANN model, het drempelgebaseerde ANN model, het periodiekengebaseerde ANN model (of periodiek ANN model, PANN) en hun variaties (zoals soft PANN en hard PANN). Naast deze modellen zijn in deze studie PAR en PANN voorgesteld voor modellering van dagelijkse afvoerprocessen. Die maken fundamenteel deel uit van AR modellen of ANN modellen. Elk AR model of ANN model is gekoppeld aan een seizoensdeel, zodat seizoensinvloed van dagelijkse afvoerprocessen tot uiting komt. Om de sterke eigenschappen van deze dynamisch verschillende modellen te combineren zijn daarnaast in deze studie de volgende vier voorspelmethodes gehanteerd: Simple average method (SAM), rollingly-updated weighted average method, semi-fixed weighted average method, en de modular semi-fixed weighted average methode. De vergelijkbaarheid van de voorspelbaarheid van deze modellen en methodes laten zien dat, ondanks de beperking van univariate afvoertijdreeksen, binnen de SACE maat, toereikende voorspellingen gemaakt kunnen worden voor voorspelperioden tot aan 5 dagen (SACE > 0.8). De over het geheel genomen prestatie van PANN is het beste voor voorspellingen 1 dag vooruit. SAM presteert over het algemeen het best van de 4 competitieve combinatie methoden en het schakelt alle vijf individuele modellen voor voorspellingen tot 4 dagen vooruit uit (behalve het PANN model voor voorspellingen 1 dag vooruit). Het ARFIMA model presteert het beste voor lange lead times (langer dan 4 dagen).

Veralgemenisering is een kritisch punt bij het maken van een ANN model wanneer de grootte van de training dataset niet groot genoeg is. De vergelijking van twee technieken om veralgemenisering te bereiken, bijv. de Bayesiaanse regularisatie (BR) methode en de cross-validated early stopping (CVES) methode, laat zien dat de BR methode de CVES methode overtreft in het algemeen door het maken van voorspellingen 1 stap vooruit, voor univariate tijdreeksen. Als ANN modellen gebouwd worden waar meervoudig verklaarbare variabelen worden meegenomen, wordt het voordeel van de BR methode ook bevestigd. Als de snelheid van training geen hoofdzaak is wordt daarom de BR methode aanbevolen voor het maken van rivierafvoer voorspellingen.

Voorspelbaarheid is een belangrijk aspect van de dynamica van hydrologische processen. Er zijn echter weinig studies bekend over dit onderwerp en zij zijn voornamelijk gebaseerd op meervoudige verklarende variabelen. In dit proefschrift een aanpak voor univariate

hydrologische tijdreeksen is voorgesteld, waarin de voorspelbaarheid van een proces gedefinieerd wordt door de voorspelbare tijdsduur, die op zijn beurt weer gedefinieerd wordt door het moment waarop de voorspelling niet beter is dan de gemiddelde waarde van het stationaire proces of de seizoens gemiddelde waarde van het seizoensproces. Met een dergelijke definitie is de voorspelbaarheid eenvoudig vergelijkbaar tussen verschillende afvoerprocessen. Onderzoek van de voorspelbaarheid in een aantal stroomgebieden van verschillende grootte toont aan dat in het algemeen hoe groter het stroomgebied is, hoe beter de voorspelbaarheid is.

Chapter 1 Introduction

Streamflow forecasting is of great importance to water resources management and flood defence. Short-term streamflow forecasting is crucial for flood defence; medium-range forecasting is highly beneficial for reservoir operation; long-range forecasting for more than one month is helpful for water resources management and planning. On the other hand, a better understanding of the streamflow process is fundamental for improving the skill of streamflow forecasting. One way of obtaining better knowledge of streamflow processes is to analyze historic streamflow records.

This thesis focuses on the analysis of stochasticity and nonlinearity of streamflow processes, and presents data-driven models mainly based on univariate streamflow time series for forecasting the streamflows of the upper Yellow River in northern China, especially medium-range streamflows on the daily timescale.

1.1 Streamflow Forecasting: Review

There are a variety of available methods for forecasting streamflows, which may fall into two general classes: process-driven methods and data-driven methods. The process-driven methods conceive a streamflow process as the output of a watershed system in the view of system theory, and mathematically approximate the internal physical processes of the watershed system that govern the streamflow process based on some understanding of those physical processes. By contrast, data-driven methods are fundamentally black-box methods, which mathematically identify the connection between the inputs and the outputs, without considering the internal physical mechanism of watershed system of interest.

1.1.1 Process-Driven Models

There are generally two types of process-driven models for streamflow forecasting: the rainfall-runoff models (lumped, semi-distributed, or distributed); and the low flow recession model. Rainfall-runoff models are applicable to modeling streamflow processes that driven by different runoff sources, especially the cases where the streamflow processes are dominated by rainfall processes, whereas low flow recession models describes the streamflow processes resulting from the drainage from the groundwater storage or other delayed sources, and consequently are usually only applicable to the recession period following the flood season.

1.1.1.1 *Low flow recession model*

Low flow recession analysis is a useful forecasting technique describing the river flow recession process during the low-flow period. The recession flow in a natural river system can be defined as the flow resulting from the drainage from the groundwater storage or other delayed sources (Hall, 1968). The recession of streamflow discharge can be reflected by a recession curve. The quantitative expression for the recession curve can be derived in several ways. Mishra et al. (2003) developed a recession flow forecasting model for the Blue Nile River using a non-linear catchment storage–outflow algorithm. However, there is a lack of consistency in obtaining recession characteristics due to the high time variability found in recessions, which has limited a wider use (Tallaksen, 1995).

1.1.1.2 *Rainfall-runoff model*

Rainfall-runoff models depict rainfall-runoff processes based on the combination of some physically-based equations and empirical equations. To physically modeling the streamflow processes, two essential sub-processes should be taken into account: rainfall-runoff transformation and channel routing. Correspondingly, a typical conceptual rainfall-runoff model is composed of a module simulating the process of rainfall to runoff transformation and another module simulating the process of channel routing.

While conceptual rainfall-runoff models are widely used for real-time and short-term flood forecasting, there are also many instances of applying conceptual models to modeling medium- to long-range rainfall-runoff processes. For example, the Pitman model, first developed in 1973 (Pitman, 1973), has become one of the most widely used monthly time-step rainfall–runoff models within southern Africa (Hughes, 2004). When being applied to real-time and short-term flood forecasting, the basic time units of input and output data of conceptual models usually range from 1 hour to 1 day. When they are applied to streamflow forecasting at large timescales (e.g., weekly or monthly), we should either aggregate the short-timescale forecasts to make large-timescale forecasts, or more commonly, change the timescale of input variables. For example, when being applied to 10-day or monthly streamflow forecasting, the time units of the inputs usually are 10 days or a month. Besides the change of input timescales, sometimes the model structure should be modified also. For instance, in conventional Xinanjiang model (Zhao and Liu, 1995), the runoff is separated into three components: the immediate runoff, the surface runoff and the groundwater runoff. When we use Xinanjiang model to forecast monthly streamflows, only two components should be considered, namely, the fast-responding runoff (surface flow and inter-soil flow) and the slow-responding runoff (groundwater runoff) (Liu, 1997). Correspondingly, the parameters for free reservoir to inter-soil and to groundwater as well as the adjustment parameter about inter-soil are not necessary.

When applying conceptual rainfall-runoff models to forecast medium and long-range streamflows, weather information is of crucial importance. In some applications, historical precipitation data are utilized as the input to conceptual rainfall-runoff models. In the traditional Ensemble Streamflow Prediction (ESP) procedure (Day 1985), a hydrological model is driven with observed precipitation and temperature data up to the beginning of the forecast to estimate basin initial conditions. Then precipitation and temperature data for the same date from every other year in the historical record are used to produce ensemble forecasts of streamflow. The British Columbia Hydro and Power Authority in Canada uses a partly distributed conceptual hydrological model to forecast the seasonal inflow from January to August to the Mica project on the Columbia River (Druce, 2001). The hydrological model is run up to the time of forecast with observed meteorological inputs to obtain the current status of the basin, then historical precipitation data are used as inputs to forecast future seasonal inflows. With the improvement of the accuracy of weather forecasting, the forecasted medium- and long-range weather data are more and more commonly used in streamflow forecasting. For instance, the European Flood Forecasting System (EFFS) which is developed collaboratively by several European countries uses the downscaled medium-range ensemble precipitation predictions from the European Centre for Medium Range Weather Forecasting (ECMWF) as inputs to water balance and rainfall-runoff models so as to make 1- to 10-day ahead streamflow forecasts (De Roo et al., 2003); Wood et al. (2002) downscaled monthly ensemble climate forecasts produced by the global spectral model (GSM) to daily time step forecasts, then used the downscaled forecasts as input to a semi-distributed Variable Infiltration Capacity (VIC) hydrological model to produce long-range streamflow

forecasts; Similarly, Tucci et al. (2003) used a general circulation model (GCM) to obtain seasonal rainfall forecasts, and then input the forecasted rainfall into a distributed hydrological model to forecast streamflows up to 5 months ahead; Yang et al. (2005) combined the HBV rainfall-runoff model with long-term weather outlook to make 10-day streamflow forecasts during the dry season.

One important issue in using the medium- and long-range weather forecasts is how to downscale the forecasted atmospheric variables (especially the precipitation) so as to meet the requirement of rainfall-runoff models. Werner et al. (2005) compared two sets of forecasts with the Ensemble Streamflow Prediction (ESP) component of the National Weather Service River Forecast System (NWSRFS): a control run based on historically observed temperature and precipitation and an experimental run based on medium-range (1-14 day) forecasts (MRF) of temperature and precipitation. Their study showed that using forecasted weather data was generally superior to using historical weather data, but there were situations when the downscaled MRF outputs actually degraded the forecast.

1.1.2 Data-Driven Methods

The use of data-driven models has the advantage of representing arbitrarily complex processes based on mathematical criteria. They are easy to apply for different conditions because the modeling and forecasting procedure is usually analogous. Furthermore, the analysis of the structure and parameters of data-driven models can sometimes provide useful information on the dynamics of the phenomenon of interest. Since on the one hand, more and more data are available nowadays with the development of modern measuring techniques, and the computational capability is more and more powerful with the development of computer techniques; on the other hand, accurate knowledge about the physical mechanisms underlying streamflow at a particular location is still not possible so far, therefore, data-driven modeling techniques gain more and more popularity in the field of hydrology in the last decade (e.g., Solomatine, 2002).

1.1.2.1 *Regression model*

Regression analysis, including simple regression and multiple regressions, is one of the oldest and most frequently used methods in streamflow forecasting. It has the advantage that it is comparatively simple and can be easily implemented. Many early regression methods used graphical techniques (Linsley et al., 1958). Later on, statistical techniques such as principle components (e.g., Marsden and Davis, 1968), multiple regression (e.g., Zuzel et al., 1975; Nash and Foley, 1982; Stedinger et al., 1989; Garen, 1992), nonparametric regression (Smith, 1991) were introduced. Some variations of the regression model are available in the field of streamflow forecasting, such as the linear perturbation model (LPM) (Nash and Barsi, 1983). The index-variable method and storage-accounting method (Lattenmaier and Wood, 1993) are also essentially regression models. Lots of applications of regression methods can be found in the literature. Tangborn and Rasmussen (1976) suggested that basin storage S_t could be estimated as a linear function of basin precipitation from the beginning of an account period (generally about the beginning of the water year) up to the forecast data, less runoff. The forecast runoff for the period T is then a linear function of S_t. In 1980s', the Water Master Plan Project in cooperation with Cairo University and the MIT developed the multivariate stepwise linear regression model for making monthly flow forecasts for the Aswan reservoir (Curry and Bras, 1980; Water Master Plan Project, 1983). British Columbia Hydro uses a multiple linear regression program that includes snowpack, precipitation and temperature to make seasonal inflow volume forecast (Fast, 1990). The regression method is still used operationally nowadays in many cases. The forecast of the potential snowmelt runoff of the

two basins located in south-eastern Alberta, Canada are based on regression methods currently (Mahabir et al., 2003). Huo et al. (2001) forecasted Inflow into the Sanmen Gorge reservoir of the Yellow River during non-flooding period with regression models using antecedent precipitation and streamflows.

Explanatory variables used in regression analysis usually include local factors (such as discharges at the upstream gauging stations, the precipitation and temperature, snow cover in the watershed area), and some geophysical quantities (such as ground temperature, sea surface temperature and atmospheric circulation index). The most important issues should be determined for applying regression methods include: (1) which factors have significant correlation with the streamflow processes under study; (2) at which timescale the selected factors have the most significant correlation with the streamflow process; (3) how long is the lag time that the streamflow response to the predictors? While real-time and short- to medium-range discharges are associated with local factors and initial conditions of the watershed, the long-range streamflows are commonly related to some remote geophysical quantities. The linkage between streamflow processes and sea surface temperatures (SSTs) as well as ENSO events have been intensively studied in the last decade (e.g., Hastenrath, 1990; Mechoso and Perez-Iribarren, 1992; Kahya and Dracup, 1993, 1994; Simpson and Cane, 1993; Marengo, 1995; Amarasekera et al., 1997; Piechota et al., 1998; Gutierrez and Dracup, 2001; Shrestha and Kostaschuk, 2005). Eltahia (1996) discovered that 25% variability of annual runoff of the Nile River is related to the SST in some region of the Pacific. Piechota et al. (1999) found that there is a significant lag relationship between El Nino and streamflow in the Northwest Pacific, and using this lag relationship, they proposed to extend the prediction of spring-summer runoff in the Columbia River basin from 1- to 3-month lead time to 3- to 7-month lead time. Hamlet and Lettenmaier (1999) incorporated the ENSO and Pacific Decadal Oscillation (PDO) climate signals into the extended streamflow prediction forecasting approach to make long-range streamflow forecast. Dettinger et al. (1999, 2000, 2002) made long-range streamflow forecasting in the United States according to El Nino and La Nino events. Whitaker et al. (2001) showed a significant relationship between the natural variability of the Ganges annual flow and the ENSO index, and proposed a statistical model that combined all these indicators to forecast the annual flow of the Ganges with a forecasting lead-time of 1 year. Eldaw et al. (2003) found that long-range forecasting of the Blue Nile River flows with lead times over 1 year was possible with a high degree of explained variance by using SSTs in a few regions in the Pacific Ocean. They used multiple linear regression model and principal component analysis to forecast the Blue Nile flows based on SSTs and the Guinea precipitation in previous year. Some researchers in China have investigated the relationship between SSTs/ENSO events and long-term streamflows of the upper Yellow River. For instance, Peng et al. (2000) found that there is good relationship between SST of northern Pacific, and the annual runoff of the upper Yellow River is closely related to ENSO events. Based on this, they fitted a regression model using SST in some oceanic regions as predictors to make annual runoff prediction. Besides the influences of SSTs and ENSO events, some other geophysical factors have also been related log-term streamflows. Tang (1992) proposed that the annual runoff of the Yellow River was closely related to the seismic field. Cai and Wang (1996) found a good relationship between the annual runoff of the headwaters of the Yellow River and the spatial pattern of ground temperature. Tomasino et al. (2004) coupled information from solar activity and from atmospheric circulation indices to make seasonal discharge predictions.

In most applications in streamflow forecasting, linear regression models are used. To make regressions more flexible, nonlinear regression models may be applied. One widely used nonlinear regression model is the threshold regression (TR) model, in which a separate

regression is fitted to a specific group of data according to some threshold values. A derivative of the TR model is the so-called model-tree model, with which a number of regression models are fitted in a hierarchical way. It has recently been used for rainfall-runoff forecasting problems (e.g., Solomatine and Dulal, 2003; Solomatine and Xue, 2004).

1.1.2.2 *Time series model*

Time series analysis plays an important role in hydrological research. It is used for building mathematical models to generate synthetic hydrological records, to forecast hydrological events, to detect trends and shifts in hydrological records, and to fill in missing data and extend record (Salas, 1993). Time series models available for streamflow forecasting can be roughly divided into two categories according to the number of time series involved in modelling: univariate models and models incorporating exogenous variables.

The most popular univariate models are the ARMA (autoregressive moving average) model and its derivatives, including ARMA, AR, ARIMA (autoregressive integrated moving average), SARIMA (seasonal ARIMA), PARMA (periodic ARMA), TAR (threshold AR), and ARFIMA (fractionally integrated ARMA) model, etc. The AR model is very commonly used for forecasting annual flows (e.g., McLeod et al., 1977). Lu et al. (1996) used an AR (3) model to forecast the annual runoff of Danjiangkou Reservoir in China. Because the ARMA model is built under the assumption that the series is stationary, whereas the hydrological time series with a timescale less than a year (e.g., monthly flow) usually exhibit strong seasonality, it is not applicable directly to such type of hydrological time series. In stead, three types of model can be employed for such series (Hipel and McLeod, 1994): seasonal ARIMA (SARIMA) model; deseasonalized ARMA model; and periodic ARMA (PARMA) model (including PAR model). These three types of model have been widely used in monthly or quarter-monthly streamflow forecasting (e.g., McKerchar and Delleur, 1974; Thmoposon et al., 1985; Noakes et al., 1985; Yurekli et al., 2005), and sometimes in daily (or even at shorter timescales) streamflow forecasting (normally as a benchmark model) (e.g., Kang et al., 1993; Abrahart and See, 2000). Time series forecasts have the property that they approach the long-term mean as the forecast lead-time increases. Bender and Simonovic (1994) suggested that, in general, SARIMA models appear to be more flexible for natural inflows with low upstream storage and high variability, while deseasonalized ARMA models may be better suited to natural inflow systems that have a large storage capacity, lower variability, and greater response lags to precipitation events. In recent years, the long-memory property of streamflow processes drew a lot of attention. Stochastic processes with long-memory can be described by the ARFIMA (autoregressive fractionally integrated moving average) model. Montanari et al. (2000) used an ARFIMA model to simulate monthly flow of the Nile River at Aswan. Following the idea of PARMA model, Ooms & Franses (2001) developed a periodic ARFIMA model to simulate monthly streamflow.

When exogenous inputs are included, we can develop ARMAX models or transfer function noise (TFN) models. Thompstone et al. (1985) compared deseasonalized ARMA model, PAR model, TFN model and a concept model, and found that TFN model performs better than competitive models. Awadallah1 & Rousselle (2000) use sea-surface temperature as exogenous input variables to develop a TFN model to forecast summer runoff of the Nile River. Because more information is used for making forecast when considering exogenous inputs, usually TFN models can make better forecast than univariate ARIMA models. A special type of TFN model is intervention model, which is capable of modeling the disturbance of some exogenous factors. Kuo and Sun (1993) develop an intervention model,

which takes the influence of the typhoon into consideration, based on AR (1) model to simulate and forecast 10-day average discharge of the Danshui River in Taiwan.

The models mentioned above are mostly linear models. Because streamflow processes, especially daily streamflow processes are commonly accepted as nonlinear, some nonlinear models are also applied to streamflow forecasting. One type of commonly used nonlinear time series model is threshold autoregressive (TAR) model (Tong and Lim, 1980). Astatkie et al. (1997) proposed a nested threshold autoregressive (NeTAR) model to describe daily streamflow processes. In fact, PARMA and PAR models can be considered as special type of TAR model, which use the season as the threshold value instead of using any observed value.

1.1.2.3 *Artificial neural network model*

The artificial neural network (ANN) model is a data-driven method with a flexible mathematical structure which is capable of identifying complex non-linear relationships between input and output data sets without the necessity of understanding the nature of the phenomena. ANNs have gained more and more popularity for hydrological forecasting in the last decade (e.g., Maier and Dandy, 2000; Dawson and Wilby, 2001). In one of the early applications involving streamflows, Kang et al. (1993) used ANNs and autoregressive moving average models to predict daily and hourly streamflows. This preliminary study concluded that ANNs are useful for forecasting streamflows. Many following studies have confirmed the superiority of ANN models over or comparableness to the traditional statistical and/or conceptual techniques in modeling the hydrological process (e.g., Raman and Sunilkumar 1995; Dibike and Solomatine, 2001; Tokar and Markus 2000; Birikundavyi et al., 2002).

The most popular type of ANN is the multi-layer perception (MLP) model optimised with backpropagation algorithm. It is also the most widely type of ANN in streamflow forecasting. Hsu et al. (1995) applied ANN models to make daily streamflow forecasts. Markus (1997) made monthly streamflow forecasts with MLP ANN models for several rivers, and compared the performance of ANN models with other models. Jain (1999) used MLP ANN model to make monthly flow forecasting. Zealand et al. (1999) used MLP ANN models to make one to four weeks ahead streamflow forecasting. Sajikumar and Thandaveswara (1999) demonstrated the use of a special type of MLP ANN model, namely a temporal back propagation neural network, to make monthly rainfall-runoff modelling. Birikundavyi et al. (2002) investigated the performance of ANN models for 7-day-ahead daily streamflow forecasting and showed that the ANNs outperformed a conceptual rainfall-runoff model for up to 5-day-ahead forecasts. Tawfik (2003) applied ANN model to predict the Nile River inflows into the Aswan reservoir for the months of July, August, and September. Kisi (2004) used MLP ANN models to make monthly flow forecasts.

Some other types of ANNs have also been applied to streamflow forecasting problem, but less frequently. Radial basis function (RBF) model has been used by many researchers for streamflow forecasting (Fernando and Jayawardena, 1998; Dibike and Solomatine, 2001; Dawson et al., 2002). Chang and Chen (2001) applied a counter-propagation fuzzy-neural network, which is a fusion of a neural network and fuzzy arithmetic, to forecast one-hour-ahead streamflow. Ballini et al. (2001) applied a neuro-fuzzy network to the problem of seasonal streamflow forecasting. Moradkhani et al. (2004) explored the applicability of a Self Organizing Radial Basis (SORB) function to one-step ahead forecasting of daily streamflow. To capture more efficiently the nonlinearity in streamflow processes, modular neural network (MNN), hybrid neural networks, and threshold (or so called range-depend) neural network can be applied (e.g., Zhang and Govindaraju, 2000; Hu et al., 2001). Zhang and Govindaraju

(2000) examined the performance of MNNs in predicting monthly discharges over three medium-sized watersheds. Hu et al. (2001) applied range-depend ANN, which could be considered as a special case of MNN, to forecast annual and daily discharges.

A frequently encountered problem during the application of ANNs to various water resource problems is the limitation of the data sets required for the training stage of ANNs, which prevents ANNs from learning input and output data sets within different ranges, thus decreasing the prediction capability during the testing stage. To overcome this problem, Cigizoglu (2003) proposed to use ARMA models to generate synthetic flows, and then incorporate these series into the training data sets of ANNs. Imrie et al. (2000) proposed the use of a guidance system to develop networks towards more general solutions.

1.1.2.4 *Other data-driven methods*

(1) Fuzzy logic

Since Zadeh (1965) published the fuzzy set theory as an extension of classic set theory, fuzzy logic has been applied successfully in many fields of application where the relationship between cause and effect (variable and results) is vague. With fuzzy logic techniques, fuzzy variables are used to organize knowledge that is expressed 'linguistically' into a formal analysis. For example, 'high flow, 'average flow and 'low flow became variables. Fuzzy logic techniques have been applied to both real-time flood forecasting (e.g., Han et al., 2002; Nayak et al., 2005) and medium- to long-range streamflow forecasting (e.g., Zhu et al., 1994; Mahabir et al., 2003). By applying fuzzy logic, Mahabir et al. (2003) developed fuzzy expert models to make spring runoff forecasts. The results show that spring runoff forecasts from the fuzzy expert systems are considerably more reliable than the regression models in forecasting the appropriate runoff zone, especially in terms of identifying low or average runoff years.

Fuzzy logic technique can be combined with conceptual models or other data-driven models (e.g., ANN). Mizumura (1995) showed that the combined approach using the conceptual Tank model and a fuzzy logic model yields satisfactory snowmelt-runoff prediction results. When fuzzy logic technique is combined with the ANN model, one can construct a fuzzy-rule based hybrid ANN model (e.g., See and Abrahart, 2001), or a more closely integrated neuro fuzzy system (ANFIS) model (e.g., Chang and Chen, 2001; Nayak et al., 2004).

(2) Nearest neighbor method

The nearest neighbor method (NNM) is a local approximation approach, which deals with a complex problem by dividing it into many subsets, each of which are composed of multi-dimensional points that have similar state patterns, and then locally approximates these points (i.e., nearest neighbors) with nonparametric or parametric models. The NNM stems from pattern recognition work (e.g., Cover and Hart, 1967). Because of its good approximation ability for nonlinear dynamics, with the development of the theory on nonlinear dynamics, NNM is adopted by many researchers in the field of nonlinear dynamics as a standard method for predicting chaotic time series (e.g., Farmer and Sidorowich, 1987; Sugihara and May, 1990). It is first applied to rainfall-runoff forecasting by Karlsson (1985), and then Yakowitz and Karlsson (1987). Following their research, many researchers apply NNM to streamflow forecasting in univariate cases (e.g., Kember et al., 1993; Liu et al., 1998; Bordignon and Lisi, 2000; Sivakumar et al., 2001) as well as multivariate cases (e.g., Porporato and Ridolfi, 2001). Yakowitz and Karlsson (1987) compared the performance of NNM with that of ARMAX model and Sacramento model for making one-day ahead streamflow forecast. They found that while both the NNM and ARMAX performances are considerably better than the performance

of the Sacramento model, the performances of NNM and ARMAX model are comparable. However, some recent studies showed that NNM outperform ARMA-type models (e.g., Bordignon and Lisi, 2000; Porporato and Ridolfi, 2001) or linear regression and linear perturbation models for river flow forecasting (Shamseldin and O'Connor, 1996).

A new method similar to NNM, nonlinear set membership (NSM) prediction method, proposed by Novara and Milanese (2001), is applied to univariate daily streamflow forecasting by Milanese and Novara (2004). Their results showed that NSM appeared to perform better than neural network model.

(3) Canonical correlation analysis

Canonical correlation analysis (CCA) is a way of measuring the linear relationship between two multidimensional variables. It can be defined as the problem of finding two sets of basis vectors, one for x and the other for y, such that the correlations between the projections of the variables onto these basis vectors are mutually maximized. CCA is a favored technique in the field of climatology and meteorology for making statistical forecasts of rainfall and temperature, etc. (e.g., Barnett and Preisendorfer, 1987; Barnston and Smith, 1996). Uvo and Graham (1998) developed canonical correlation analysis models to forecast seasonal discharge, one season in advance, at 12 sites in north-eastern South America from Pacific and Atlantic Ocean SST.

1.2 Some Important Issues in Streamflow Forecasting

In this section, several issues need further investigation in streamflow forecasting will be addresses, especially for the problem of medium- and long-range streamflow forecasting.

1.2.1 Predictability of Streamflow Processes

Predictability is an important aspect of the dynamics of hydrological processes. But the predictability of hydrological processes has not attracted much attention by the hydrology community until recent several years. Some examples of the studies on predictability include those based on the multiple explanatory variables (e.g., Maurer et al., 2003; Maurer et al., 2004) and those based on univariate hydrological time series (e.g., Wang et al., 2004). Maurer et al. (2003) assessed seasonal runoff in the Mississippi River Basin, and found that: at a lead time of 1.5 month, soil moisture is dominant for predictive capability of runoff, but snow dominates in the summer in the western part of the basin; modest winter runoff predictability exists at a lead time of 3 seasons due to both climate and soil moisture; local summer runoff predictability is limited to the western mountainous areas (generating high runoff) through a lead of 2 seasons. Maurer et al. (2004) analyzed the utility of the climate signals, soil moisture, and snow as predictors of runoff variability in North America for lead times up to a year, and concluded that for predicting runoff variability, knowledge of the land surface state, especially in its dry state, can provide valuable predictability as a complement to climate information for lead times of one to two seasons (i.e., up to 4.5 months lead time). Wang et al. (2004) estimated the predictability of two daily streamflow series of the upper and lower Yellow River in China based on the performances of the univariate ARMA models.

1.2.1.1 *Methods of estimating predictability*

Because the research on predictability is still on its early stage, there is a lack of well-established methods on how to estimate the predictability of hydrological processes. According to existent results on this issue in literature, two approaches may be distinguished:

univariate approach and multivariate approach. The former one measures the predictability based on univariate time series analysis techniques (e.g., Wang et al., 2004), whereas the latter one estimates the predictability based on the knowledge of the rainfall-runoff generation mechanism and/or the tele-connections between global environment and streamflow processes of interest (e.g., Maurer et al., 2003, 2004). A multivariate approach may give us an insightful understanding of the watershed system. However, to estimate predictability with a multivariate approach need good knowledge of physical mechanisms underlying the streamflow process of interest and we need a suitable model which is assumed to be able to capture the mechanism well; on the other hand, the knowledge of different streamflow processes varies, and the skills of modeling are highly user-depend. Therefore, the predictabilities estimated with such an approach for different streamflow processes are normally not comparable. Furthermore, dominant factors of a hydrological system may change with the change of temporal and spatial scales, and the explanatory variables used in estimating the predictability of streamflow processes may have a problem of predictability themselves, which would make the problem of estimating the predictability of streamflow processes more complicated. For example, weather processes, which are fundamentally the driven forces of streamflow processes, are only predictable for at most a few weeks because of its inherently nonlinear and chaotic nature. In contrast, the univariate approach can be easily applied to different hydrological time series and the results are comparable among different processes.

1.2.1.2 *Roles of weather forecasts, and the impacts of multi-collinearity*

One issue closely related to the estimation of model predictability is the effectiveness of using forecasted meteorological data as inputs to improve the accuracy of medium-range as well as long-range streamflow forecasts. In practice, for medium-range forecasting, there are two ways of using meteorological data, namely, using historical records (e.g., Day, 1985; Druce, 2001) or using forecasted data (e.g., Wood et al., 2002; De Roo et al., 2003; Tucci et al., 2003). Coulibaly (2003) investigated the effect of meteorological predictions (up to 7 days) on the accuracy of spring flow forecasts using one conceptual model (PREVIS) and one ANN model for a catchment in north-eastern Canada. The result showed that using 4-day-ahead weather forecasts does not provide significant improvement of daily spring flow prediction, which indicated a maximum limit (up to 3 days ahead) for an adequate practical use of meteorological forecasts. Hu (2003) experimented to use forecasted large scale precipitation data from ECMWF to improve one to three days ahead streamflow forecasts with ANN models, but the result indicated that the improvement was marginal. Clark and Hay (2004) examined an archive containing over 40 years of 8-day atmospheric forecasts from the NCEP reanalysis project to assess the possibilities of using medium-range numerical weather prediction model outputs for predictions of streamflow. This analysis showed that the biases in the NCEP forecasts were quite extreme. Consequently, they suggested a need for additional processing of the NCEP medium-range forecast model output before it is used for hydrological predictions. Gouweleeuw et al. (2005) combined numerical weather prediction models with rainfall-runoff models to make medium-range forecasts for two historic flood events of the Muse River and the Odra River. Their results showed that, while the forecasts for the Meuse River were encouraging, the forecasts for the Odra River were poor. Bartholmes and Todini (2005) coupled numerical weather prediction systems and a rainfall-runoff model for a case study on the Po River in northern Italy, and their results demonstrated the poor reliability of the quantitative precipitation forecasts produced by meteorological models presently available. Because of limited accuracy of weather forecasts, the current operational practice at the Colorado Basin River Forecast Center is a blending approach, namely, using "deterministic" meteorological forecasts of temperature for 1-10 days and

precipitation for 1-3 days following the initialization time of the forecast period (Werner et al., 2005). After the meteorological forecast period, ensembles of historical temperature and precipitation data sequences are added to the end of the deterministic meteorological forecasts by blending to form ensemble model inputs extending out to several months.

Another issue that should be considered when searching for potential predictors is the problem of multi-collinearity, which is the condition where at least one predictor closely related to one or more other predictors. Multi-collinearity may lead to the ineffectiveness in the utilization of predictors. For example, Uvo et al. (2000) showed that for ANNs with SST inputs, the inclusion of precipitation as input to train the ANN did not always increase the skill of the ANN in forecasting seasonal discharge for some regions, because the dominant climate there responds strongly to SST variability.

1.2.1.3 *Future directions in the research on predictability*

In 2001 the U.S. Global Change Research Program (USGCRP) Water Cycle Study Group (WCSG) posed several specific goals for predictability research in hydrological systems (WCSG, 2001). More recently, the Committee on Hydrological Science of NRC (NRC, 2002) defined five research challenges in predictability science and limits-to-prediction for hydrological systems. Summarizing these specific goals and challenges, the following directions for the research on the predictability of streamflow processes may be identified:

- Definition of robust measures of limits-to-prediction that account for spatial and temporal scales;
- Systematic investigation and quantification of the spatial and temporal features of how the ocean, the atmosphere, and the watershed system are coupled together;
- Identify the hydrological data requirements for improving the predictability, and improve monitoring systems and enhance the observations according to these requirements.

1.2.2 Quantitative Description of Forecast Uncertainty

Whatever accurate a model may be, there is more or less uncertainty of the forecast from it. The forecast uncertainty is intrinsically related to the model predictability of a streamflow process. When the forecast uncertainty increases to a certain extent with increase of forecasting lead times, the model predictability disappears. To make the forecast user clear about uncertainty of the forecast, and take the risk explicitly into account when making decisions, forecasts should be stated in probabilistic rather than deterministic way. A probabilistic forecast may be described by giving prediction interval (PI) around the mean value, or, equivalently but more informatively, giving predictive distribution. Probabilistic forecasts are scientifically more honest (Krzysztofowicz, 2001). In the risk-based water resources management, which has been attracting more and more attention from water resources authorities, the probabilistic forecast will be an essential ingredient.

1.2.2.1 *Major approaches available for evaluating forecast uncertainty*

In the hydrological community, many efforts have been made on evaluating the uncertainty for hydrological forecasts in the last two decades or so. In the early work of Hirsh (1981), three ways for probabilistic long-range streamflow forecast were suggested: (i) using historical streamflow data; (ii) using stochastic model and (iii) using deterministic hydrological model with simulated antecedent hydrological conditions. Presently, the

following five major types of approaches have been proposed and used in hydrological forecasting.

(1) Develop a stochastic model (e.g., the AR model) of streamflow based on the historic records. Then, initialize the model's state variables to present conditions and use a random number generator to produce multiple (equally likely) traces from this initial condition so as to estimate the predictive distribution. Hirsh (1981) developed this method, where a periodic ARMA model is used to describe monthly streamflow processes. This method relies on assumptions of normality and independence of the error terms of the PARMA model. Unfortunately, the normality assumption is often violated in the cases of streamflow processes.

(2) Produce multiple estimates of the future streamflow with a deterministic hydrological model based on current basin conditions and past (or forecasted) meteorological observations (rain, snow, temperature, humidity, wind). This method is called Extended Streamflow Prediction (Day, 1985), latter on renamed as Ensemble Streamflow Prediction (ESP). The ESP procedure assumes that meteorological events that occurred in the past are representative of events that may occur in the future. Each year of historical meterological data is assumed to be a possible representation of the future and is used to simulate a streamflow trace, so that ESP produces a probabilistic for each streamflow variable and period of interest. Smith et al. (1992) developed a nonparametric framework, which can account for climate information through weighting the historical years and the effects of hydrological model error. The ESP forecast system has been applied for many years as part of the National Weather Service (NWS) River Forecast System (NWSRFS) in the United States. Recently, NWS has extended the original idea to facilitate incorporation of climate outlooks into the ESP by inputting historical meteorological events adjusted with meteorological and climatological forecasts (Perica, 1998; Wood et al, 2002; Werner et al., 2005). The skill of the ESP approach depends on: (1) the accuracy of hydrological models; (2) methods for updating basin initial conditions at the start of the forecast period; and (3) the local-scale weather and climate forecasts. Ensemble forecast techniques are beginning to be used for hydrological prediction by operational hydrological services throughout the world. The ESP approach is also adopted by European Flood Forecasting System (EFFS) for making up to 10 days ahead probabilistic flood forecast (De Roo et al., 2003), but only medium-range ensemble weather forecasts, instead of historical weather data, are used to generate ensemble traces.

(3) Estimate the predictive uncertainty associated with hydrological models based on Monte Carlo simulation. This method is called GLUE (Generalised Likelihood Uncertainty Estimation) proposed by Beven and Binley (1992). By specifying the sampling ranges for each parameter to be considered, as well as a formal definition of the likelihood measure to be used and the criteria for acceptance or rejection of the models, a sample of parameter sets are selected by Monte Carlo simulation, using uniform random sampling across the specified parameter range. The predictions of the Monte Carlo realisations are then weighted by the likelihood measures to formulate a cumulative distribution of predictions from which prediction quantiles can be estimated. Therefore, prediction with the GLUE method is essentially a process of ensemble forecasting, similar to ESP. But instead of using different meteorological inputs (either historical or forecasted) to generate the ensemble of streamflow forecasts as with ESP, GLUE uses a sample of acceptable parameter sets. GLUE method can be used to investigate the effect of parametric and input error using several realizations from single models (Beven and Freer, 2001). A major limitation of the GLUE methodology is the dependence on Monte Carlo simulation, which

requires considerable computing resources. For complex models requiring a great deal of computer time for a single run, it will not be possible to fully explore high order parameter response surfaces. Khu and Werner (2003) proposed a hybrid method, with which genetic algorithm and ANN are used jointly to reduce the runs of Monte Carlo simulation required to establish a reliable of model uncertainty.

(4) Bayesian-theory-based probabilistic forecast method is recently proposed by Krzysztofowicz (1999). The basic idea of Bayesian prediction is to blend together prior and posterior information using Bayes theorem. In the Bayesian forecasting system (BFS), the total uncertainty is decomposed into two sources: (1) input uncertainty associated with random inputs (mainly, precipitation) and (2) hydrological uncertainty arising from all sources beyond those classified as random inputs, including model, parameter, estimation, and measurement errors. BFS offers a theoretically derived structure to quantify the input uncertainty and hydrological uncertainty and then integrate all those uncertainties into a predictive (Bayes) distribution. The main advantage of BFS is that it moves from evaluating the predictive probability based on lumped additive error terms to a framework that considers error source terms individually.

(5) Estimate forecast uncertainty by analyzing the statistical properties of the model error series (i.e., residuals) that occur in reproducing observed historical streamflow data. This approach has been followed by many researchers in statistics (see Chatfield, 2001). Whatever sources of forecasting uncertainty may be, they will be reflected in residuals and therefore we can construct the prediction interval for a specific forecasting model (or method) according to the empirical distribution function of the residuals, supposing that the model (or method) is un-biased, and the hydrological process is stationary and long enough. Montanari and Brath (2004) proposed a meta-Gaussian approach in order to estimate the probability distribution of the model error conditioned by the simulated streamflow. However, the meta-Gaussian approach rests on the assumption that the model error is ergodic which is rarely the case. Wang et al. (2005a) applied the residual based empirical approach and bootstrap approach (proposed by Pascual et al., 2004), and considered the seasonal variations in the model error into account, to construct prediction interval for monthly streamflow forecasts from AR models. The results show that both the empirical approach and the bootstrap method work reasonably well according to their robustness and unbiasedness, and the empirical approach gives results comparable to or even better than bootstrap method.

1.2.2.2 *Some needs in the study on forecast uncertainty*

Albeit a large availability of many methods for evaluating forecast uncertainty, there is a lack of comparisons of these methods. The National Weather Service (NWS) presently carried out a hydrometeorological forecast project (Schaake et al., 2001), within which the Bayesian forecasting system (BFS) described by Krzysztofowicz (1999) and an enhancement of the NWS ensemble streamflow prediction (ESP) system were tested. Krzysztofowicz (1999) stated that, unless hydrological uncertainty is insignificant and can be ignored, neither Monte Carlo simulation nor ensemble forecasting are synonymous with probabilistic forecasting. This statement seems to be substantiated by a visual inspection of the results shown by Freer et al. (1996), in which the 90% uncertainty bound with the GLUE method actually cannot cover 90% of observed values. The sources of uncertainty in hydrological modeling include (e.g. Refsgaard and Storm, 1996): (a) uncertainties in input data (e.g. precipitation and temperature); (b) uncertainties in data used for calibration (e.g. streamflow); (c) uncertainties in model parameters; and (d) imperfect model structure. As a matter of fact, the uncertainties in the model parameters which are considered in the GLUE method, and the uncertainties in

the model inputs which are dealt with in the ESP method, cannot account for all the uncertainties in the simulations. The advantage of the Bayesian forecasting system (BFS) which seems to have included different sources of uncertainty is yet to be confirmed. On the other hand, because no theoretical formula is available for estimating prediction interval for not only some data-driven models (e.g., ANN models) but also all the conceptual rainfall-runoff models, the residual-based empirical approach is likely to be very useful.

While it is believed that climate predictions several seasons in advance can help narrow the uncertainty in hydrological predictions, it is still not clear how the uncertainty from climate prediction will propagate to hydrological predictions and how to minimize the propagation. Furthermore, how the uncertainty propagates in making multi-step forecasts is an issue requiring more investigation. There are some efforts on this respect ongoing. For instance, Kyriakidis et al. (2001) adopted a Monte Carlo framework for propagating uncertainty in dynamically downscaled seasonal forecasts of area-averaged daily precipitation to associated streamflow response calculations.

One issue related to the forecast uncertainty is the education and training of forecast users for the interpretation and use of probabilistic forecasts. Because rainfall-runoff models are almost always believed to provide deterministic forecasts, albeit never exactly, it is not an easy task to make the concept of forecast uncertainty accepted by forecast users.

1.2.3 Model Comparison and Selection

With more and more streamflow forecasting models available, a lot of model intercomparisons have been made, comparing the structure, data requirements, computational requirements and accuracy of forecasts of the models of interest, among which the accuracy is the major concern.

1.2.3.1 *Comparing hydrological models for real-time or short-term streamflow forecast*

The amount of model comparisons for real-time flood forecasting or short-term streamflow forecasting is huge in the literature. World Meteorological Organization (WMO) (1975, 1986, 1992) organized several times of intercomparison of conceptual hydrological models mainly for real-time streamflow forecasting. Many other comparative experiments have been carried out since the work of WMO, comparing not only process-driven models but also data-driven models. For example: Refsgaard and Knudsen (1996) compared three models, i.e., a lumped conceptual modeling system (NAM), a distributed physically based system (MIKE SHE), and an intermediate approach (WATBAL), and concluded that all models performed equally well when at least 1 year's data were available for calibration, while the distributed models performed marginally better for cases where no calibration was allowed; Ribeiro et al. (1998) compared the performances of ARMAX models and the deterministic model for real-time forecasting of daily inflows and found that the ARMAX models with Kalman filters are superior to the deterministic model in real time within a horizon of 2 days; for a 3-day horizon, the models are equivalent; for a horizon of 4 days or more, the deterministic model is superior to the ARMAX models with Kalman filters; Tingsanchali and Gautam (2000) showed that the ANN model was robust and performed better than two conceptual rainfall-runoff model (Tank and NAM models) in 1 and 2 days ahead discharge forecasting. More recently, the Hydrology Laboratory of the National Weather Service (NWS) of United States proposed the Distributed Model Intercomparison Project (DMIP) for comparing the performance of distributed rainfall-runoff models (Reed et al., 2004; Smith et al., 2004), of which the focus is the sensitivity of runoff hydrographs to spatial and temporal variations in precipitation. The results of DMIP show that the lumped model outperformed distributed models in more cases than distributed

models outperformed the lumped model (Reed et al., 2004). So far DMIP project concentrated on simulation intercomparisons, future DMIP phases may contain a test of distributed models in a pseudo-forecast environment.

1.2.3.2 *Comparing hydrological models for medium- to long-range streamflow forecast*

Considerable amount of studies have also been conducted on medium- to long-range streamflow forecasting. For making forecasts at timescales larger than a day, there are more data-driven models than process-driven models. Thompstone et al. (1985) compared ARMA models, PAR models, TFN models and a conceptual model for quarter-monthly streamflow forecasting, finding that the TFN models with one or two exogenous variables (i.e., precipitation and temperature) outperforms competitive models. Noakes et al. (1985) made an extensive forecasting study comparing deseasonalized ARMA models, SARIMA models and PAR models for forecasting monthly streamflows, finding that the PAR model generally outperformed the others. Markus (1997) compared MLP ANN models with linear regression models, PAR(1) models, TFN models, conical correlation analysis models and a conceptual model for forecasting monthly flows, finding that the performance of each model differs for different rivers, and in general, the ANN model performs better than or comparable to other models. Li (1998) compared month-scale Xinanjiang model and multiple regression model for forecasting monthly runoff for several catchments in northern China, and found that the multiple regression model with antecedent precipitation, temperature and discharge as predictors has better forecast accuracy than the conceptual model. Jain et al. (1999) compared an ARMA model and an ANN model fitted to the monthly inflow data series, found that the ANN modelled the high flows better, whereas low flows were better predicted through the ARMA model. Zealand et al. (1999) showed that ANNs consistently outperformed a stochastic-deterministic watershed model for making one to four weeks ahead streamflow forecasting. Some studies compared the process-driven models and data-driven models for making medium-range forecasting at daily timescales. Birikundavyi et al. (2002) showed that ANN models outperform a deterministic conceptual model for up to 5-day-ahead forecasts, and are also superior to the ARMAX model coupled with a Kalman filter for forecasting 7-day-ahead daily streamflows.

1.2.3.3 *Shortcomings of the studies on model intercomparison in literature*

Despite of so many efforts in comparing hydrological forecasting models, no conclusive comparison result is available yet, due to the following shortcomings:

(1) In some comparative studies, such as the comparison between a TFN model with an ARMA model (Thompstone et a., 1985), different input data were used for different models, which makes the comparison unfair. In some other comparisons, it seems that some competitive models are not elaborately built. For example, Hsu et al. (1995) compared ANN model and ARMAX model for daily streamflow forecasting, and Kisi (2004) made a comparison between ANN model and AR model for monthly flow forecasting. In both comparisons, there is no indication that data are appropriately pre-processed before bulding the ARMAX model and the AR model, whereas pre-processing procedures, such as log-transformation and deseasonalization, are necessary for fitting ARMA-type time series model to streamflow processes (Hipel and Mcleaod, 1994). Such kind of comparison is obviously biased.

(2) There is a lack of linkage of model performances and the characteristics of the watershed systems of interest in most comparison studies. A clear linkage of the model performance

and the characteristics of the watershed systems would make the results of model comparison more informative to hydrology practitioners. Such kind of knowledge may lead to the solution to the question "Why does one particular model perform relatively well in one basin but not as well in another basin?" (Reed et al., 2004). Unfortunately, even in some studies where detailed descriptions of the watershed of interest are given (e.g., Reed et al., 2004; Smith et al., 2004), no clear linkage of model performances and the basin characteristics is established.

Although many models have specified their application range (e.g., catchment size and climate zone), such kind of specifications are usually too general to serve as an operational guidance for selecting a model for a given watershed. Due to the lack of conclusive comparisons, model selection is still very subjective in practice, usually user-dependent, and hydrological practitioners are not confident enough to apply new techniques in practical applications, although lots of new methods have been proposed in the last two decades.

1.2.4 Forecast Combination

Forecast combination may avoid unexpected high variability in the final prediction resulted from model uncertainty (Chatfield, 1996), and takes advantage of the availability of both multiple information and computing resources for data-intensive forecasting (Bunn, 1989). While combining forecasts has become a well-established procedure for improving forecasting accuracy and the combined forecasting has a long history in the econometrics community, it has not received much attention in the field of hydrological forecasting until recently.

Forecast combination methods may be roughly broken into two categories. The first one is the *ensemble approach*, by which a set of forecasts are produced on the same task with different models (or one model with different inputs), and then the forecasts are combined. The second one is the *modular approach*, under which a task or problem is divided into a number of subtasks (regimes), and the complete task solution requires the contribution of all of the individual regimes.

1.2.4.1 *Ensemble approach to forecast combination*

While the ensemble prediction technique is normally used to provide probabilistic predictions as in ESP, it may also be extended to be used as a forecast combination technique. There are two main issues about the ensemble combination: First, how to select a set of models and generate an ensemble of forecasts to be combined; and second, how to estimate the combining weights so as to minimize the out-of-sample forecast errors. Ideally, the selection of the ensemble models should provide the information of a specific process from different perspectives. As for the methods of estimating the combining weights, the methods now are available range from the robust equal weighting (i.e., simple average method SAM) to the far more theoretically complex, such as neural network method (Donaldson and Kamstra, 1996). In the early work of McLeod et al. (1987), they showed that significant improvements in forecast performance can be achieved by using weighted average method to combine forecasts produced by different types of models applied to quarter-monthly river flows. Shamseldin et al. (1997) examined three different combination methods in the context of flood forecasting, namely, the simple average method, the weighted-average method and the neural network method, and confirmed that better discharge estimates can be obtained by combining the outputs of different models. Coulibaly et al. (2005) showed that, using weighted average method to combine three dynamically different models can significantly

improve the accuracy of the daily reservoir inflow forecast for up to 4 days ahead. Meanwhile, the robustness of the SAM is confirmed in some other studies. For example, Butts et al. (2004) generated multi-model ensembles using ten distinct model structures, and showed that a simple average of the 10 model structures performed better than any single model, and weighted ensemble was similar to the simple ensemble average but uses a smaller ensemble.

1.2.4.2 *Modular approach to forecast combination*

The *modular approach* is based on the principle of divide-and-conquer (DAC), which deals with a complex problem by breaking it into simple sub-problems whose solutions can be combined to yield a solution to the complex problem (Jordan and Jacobs, 1994). Because streamflow generation processes, especially daily streamflow processes, usually have pronounced seasonal means, variances, and at the same time, dependence structures and the under-lying mechanisms of streamflow generation are likely to be quite different during low, medium, and high flow periods, hence, several approaches have taken to divide a streamflow process: use threshold values to divide the streamflow regimes; cluster the streamflow process into several domains (e.g., low flow, medium flow and flood); or, partition the streamflow process according to the seasonal difference. Hu et al. (2001) developed a threshold-based ANN model to make streamflow forecasts for the Yangtze River. In the studies of Zhang and Govindaraju (2000), See and Openshaw (2000) and Xiong et al. (2001), the model combinations are fundamentally based on dividing the hydrological process into several domains according to the conditions of the hydrological process. Zhang and Govindaraju (2000) used a modular ANN model to forecast monthly discharges, in which different expert neural networks were trained to low-, medium- and high-flow events, and a gating network is trained to combine the expert networks. See and Openshaw (2000) used four different approaches (i.e., an average, a Bayesian approach, and two fuzzy logic models) to combine the river level forecasts of three models (i.e., a hybrid neural network, an autoregressive moving average model, and a simple fuzzy rule-based model), and found that the addition of fuzzy logic to the crisp Bayesian approach yielded overall results that were superior to the other individual and integrated approaches. Xiong et al. (2001) showed that the first-order Takagi-Sugeno fuzzy system (Fiordaliso, 1998), in which the discharge series are divided into two or three regimes, works almost the same as the weighted average method and neural network method in combining five rainfall-runoff models.

1.2.4.3 *Two special issues in the study of forecast combination*

After nearly four decades' study, many results are available about how to combine forecasts in the literature. The scientific outcome of further study on this issue generally seems to be limited. However, there are two issues worth some attention.

First, the roles of judgemental forecasts and judgemental combination in combined forecasting. Despite of a large literature on combining forecasts, the choice of which method to use is not obvious (de Menezes et al., 2000). In the practice of hydrological forecasting, the final forecasts are rarely made solely depend on models. Normally, the meeting discussion is always an imperative procedure before publishing any forecasts, especially in China. The essence of the meeting discussion is the judgemental forecasts and the judgemental combination of model forecasts. However, some evidences from the management science and econometrics show that, the judgemental combination is less accurate than the simple average combination (e.g., Lawrence et al., 1986) or the econometric model alone (Weinberg, 1986). Therefore, on the one hand, how to combine judgemental forecasts, namely, expert experience, with model forecasts is an issue need investigation; on the other hand, it would be interesting

to investigate how many gains are achieved with the meeting discussion procedure compared with the mechanical combination of a number of forecasting models (e.g., more than five models, as suggested by Armstrong (2001)).

Second, how to construction prediction intervals for combined forecasts. Despite of a considerable literature on the combination of forecasts, there is little guidance regarding the assessment of their uncertainty. Some studies have been conducted on this issue in the field of statistics and machine learning (e.g., Taylor and Bunn, 1999; Carney and Cunningham, 1999), but there is no application to the combination of hydrological forecasts so far.

1.2.5 Evaluation of the Value of Streamflow Forecasts

The value of forecasting is the economic, social or other benefit that results from a forecast, that is, the net benefits that accrue if a forecast is available (Lettenmaier and Wood, 1993). Many studies confirm the economic value of streamflow forecasting for flood defence and water management through improved reservoir operation. For example, Yeh et al. (1982) estimated the potential benefits of long-range (one month to one year) streamflow forecasts for the operation of the Oroville-Thermalito reservoir system of the California State Water Project. Benefits included hydropower generation, water conservation for irrigation and other beneficial uses, and decreased seepage damage to crops. The simulation results of Kim and Palmer (1997) demonstrated that including the seasonal forecasts is beneficial to hydropower operation. In the 1998 water year, the Salt River Project (SRP), which is the largest provider of water and electricity in Arizona in the United State, for the first time successfully used the seasonal forecast of a wet winter to influence its reservoir operations, saving about $1.4 million from the reduction in the costs of groundwater pumping. Owing to the success in 1998, SRP plans to continue to use seasonal forecasts in decision-making (NRC, 1999, pp.9). Hamlet et al. (2002) showed that long-lead forecasts derived from improved forecasting of the El Nino Southern Oscillation (ENSO) and the Pacific Decadal Oscillation (PDO) could result in considerable improvements in reservoir operating system performance for hydropower generation. In Europe, there are many flood management actions, such as emptying reservoirs, lowering water levels in rivers, stockpiling emergency supplies and placing key personal on alert, where forecast lead times of greater than the 3–5 days achievable in the largest European basins would be extremely useful (De Roo et al., 2003). Besides the improvement of reservoir operation, many potential values of streamflow forecasting deserve further investigation, including improving water quality and navigation conditions, reducing the expenses on flood defence and the loss of human lives and properties, and so on. Such kind of assessment could help to determine whether it is worthwhile to put additional investment on improving existing forecasting systems. Unfortunately, the measurement and monetary quantification of the diverse range benefits is non-existent or rudimentary, and even in those areas of benefit where research has been focused there remain some obvious research needs (Parker et al., 2005).

Be aware that, the value of forecast may be not always positive. For example, Chiew et al. (2003) investigated the use of seasonal streamflow forecasts to help manage three water resources systems in southeast Australia. The results suggested that although the ENSO–streamflow relationship and the serial correlation in streamflow are statistically significant, the correlations are not sufficiently high to considerably benefit the management of conservative low-risk water resources systems. Furthermore, due to the forecast uncertainty, bad forecasts may even cause economic losses. Therefore, a objective evaluation index system is needed to assess the value of streamflow forecasting in the future work. This would be valuable for risk-based water resources management.

1.3 Overview of the Thesis

1.3.1 Objectives of the Research

Hydrological time series are not simply a collection of numbers, but to analyze the hydrological time series with some mathematical tools from the point of view of the numbers is a beneficial complement to the physically-based research approaches which deal with hydrological processes from the point of view of physical mechanisms. The objectives of this research are:

(1) Study the characteristics of several streamflow processes with mathematically-based methods, so as to understand the streamflow processes from the perspective of the univariate streamflow time series.
(2) Model the streamflow process of the upper Yellow River with different data-driven models, focusing on giving 1- to 10-day ahead forecasts.

1.3.2 Outline of the Thesis

Apart from the introduction in Chapter 1, the thesis is composed of the following two parts.

Chapter 2 and Chapter 3 focus on the stochasticity of streamflow processes and the forecasting of the streamflow of the upper Yellow River with ARMA-type models. In Chapter 2, three important aspects of stochasticity of streamflow processes are discussed, namely, stationarity, seasonality, and long-memory. In Chapter 3, ARMA models and ARFIMA models will be fitted to the daily streamflow, and an AR model will be fitted to monthly flows of the Yellow River at Tangnaihai (TNH). To capture the seasonal variation in autocorrelation structures of streamflow processes, a strategy of building periodic autoregressive (PAR) model based on cluster analysis is proposed for modeling daily flows.

Chapter 4 to Chapter 7 focus on the nonlinearity of streamflow processes and the forecasting of the daily streamflow of the upper Yellow River with nonlinear models. In Chapter 4, the nonlinearity of the streamflow process of four rivers at different timescales will be analyzed with the BDS test. Thereafter, a special type of nonlinearity, chaos, will be further investigated. In Chapter 5, the existence of a special type of stochastic nonlinearity, i.e., autoregressive conditional heteroskedasticity, will be tested for daily and monthly streamflow series of the upper Yellow River at TNH. Following the test, an ARMA-GARCH error model is proposed for the daily flow process at TNH. In Chapter 6 the ANN model will be applied to make 1- to 10-day ahead forecasts for the daily flow process of the Yellow River at TNH. To make the ANN model more flexible in capturing the seasonality as well as nonlinearity, the periodic ANN model is proposed. Forecasts based on linear and nonlinear models will be compared, and the issue of combining the forecasts from different models is discussed. In Chapter 7, some issues about the generalization of ANN models will be discussed, and two widely used techniques (the cross-validated early stopping and the Bayesian regularization) to achieve the generalization will be compared for making predictions for univariate time series (including several streamflow time series). Then, the Bayesian regularization technique will be applied to the streamflow forecasting for the case where exogenous variables (upstream discharges, precipitation and temperature) are included in the ANN model.

Finally, in Chapter 8, conclusions and some suggestions for future research will be presented.

Chapter 2 Stochasticity of Streamflow Processes

Streamflow processes of five rivers, i.e., the Yellow River in China, the Rhine River and the Danube River in Europe, the Umpqua River and the Ocmulgee River in United States, will be studied in this thesis. In this chapter, three important aspects of stochasticity of the streamflow processes of the five rivers at 6 sites are discussed, namely, stationarity, autocorrelation, and long-memory.

2.1 Study Areas and Data Used

2.1.1 Study areas

The Yellow River is the second longest river in China. The headwaters of the Yellow River originate at an elevation of 4,500 m in the Yueguzonglie Basin located on the northern slope of the Bayankara Mountains in the north-eastern part of the Tibet Plateau. In this area, the discharge gauging station Tangnaihai (TNH) has a 133,650 km^2 contributing watershed, including a permanently snow-covered area of 192 km^2. The main channel of this watershed is over 1500 km of length. Most parts of the watershed are 3000 ~ 4000 meters above sea level. Snowmelt water composes about 5% of total runoff. Most rain falls in summer. Because the watershed is partly permanently snow-covered, sparsely populated, and lacks of any major hydraulic works, it is fairly pristine. The average annual runoff volume (1956-2000) at TNH is 20.4 billion cubic meters, about 35% of the whole Yellow River Basin. Therefore, this watershed is the main runoff generation area of the Yellow River basin.

The middle reaches of the Yellow River differ from the headwaters significantly. In this region, the precipitation is concentrated in summer, and occurs usually in the form of storm-rain. Because of the special loess geomorphology and the poor vegetation-cover, the runoff responds usually fast to rainfall. Because the middle reach area is densely populated, water withdrawal for industry and agriculture has substantial influence on the river flow process. Furthermore, there are many hydraulic works in the main channel and main tributaries. Therefore, the streamflow process is severely influenced by human activities. Although the drainage area at Tongguan (TG) (Figure 1), about 682,141 km^2, is more than 5 times as large as that at TNH, the average annual runoff is less than 2 times as much as that at TNH.

The Rhine is the largest hydrological system in Western Europe. Rising in eastern Switzerland, the Rhine flows 1,320 km via Germany and the Netherlands in a generally north-western direction to its mouth on the North Sea. The Rhine basin (185,000 km^2) consists of 4 main sub-basins: the alpine region in Switzerland, and downstream the lower mountain regions of the tributaries the Neckar, the Main (Germany), and the Moselle (Germany and France). It incorporates high Alphine snow and glacier regions, low mountain ranges and the lowland areas. The runoff regime of the Rhine in Germany is characterised mainly by two

effects (Grabs, 1997): during the low water season in summer a major amount of the runoff is generated in the Alps due to melting of snow and ice. The tributaries of the French and German parts of the Rhine do not contribute much to this base flow. Most of the precipitation within non-Alphine area is taken up by evaporation. The situation changes in winter times: much of the precipitation in the Alphine region is stored as snow and ice whereas in the remaining catchments it becomes runoff. Floods are likely to occur from December to March. The Rhine basin is very densely populated (in total around 50 million). Dikes protect the lower parts of the basin from flooding. The gauging station Lobith is located at the lower reaches of the Rhine in the Netherlands, near German-Dutch border. It has a drainage area of 160,800 km^2.

The Danube is the second longest river in Europe, and the most important river of central and south-eastern Europe. It rises in the Black Forest Mountains of western Germany and flows in a generally easterly direction for a distance of about 2,850 km reaching the Black Sea. Due to its geologic and geographic conditions the Danube River Basin can be divided into three main parts: the Upper Danube Basin, the Middle Danube Basin, and the Lower Danube Basin. The gauging station Achleiten is located at the upper reaches of the Danube in the Austria, with a drainage area of 76,653 km^2. The upper regions of the Danube River Basin show strong influence from the Atlantic climate with high precipitation. In the upper part of the Danube the Inn contributes the main water volume, adding more water to the Danube than it has itself at the point of confluence of the two.

The Umpqua River is one of the longest coastal basins in Oregon in north-western United States, approximately 340 km in length, with a drainage area of over 12,200 km^2, rising in two branches and flowing north and west to the Pacific Ocean. The Umpqua River Basin is characterized by a temperate, maritime climate with wet, mild winters and moderately dry, warm summers. Because the river begins at high elevations in the Cascade Mountains, it receives more precipitation and a heavier snowpack than lower headwater elevation coastal rivers (Johnson et al., 1994). Most precipitation falls in the winter. There is little rain June to October in southern-central Oregon, and most regional streams barely flow during those months; but because of the snowmelt stored in deep pumice and other volcanic soils, water flow in the upper North Umpqua River remains relatively constant, with deep, swift, and cold water. Downstream in the North Umpqua River, and even more so in the South Umpqua River, the flow is erratic, with shallower, slower moving, and warmer water. The gauging station near Elkton, Oregon, has a drainage area of 9,535 km^2. Regulation by powerplants on North Umpqua River ordinarily does not affect discharge at this station. There are diversions for irrigation upstream from the station.

The Ocmulgee River basin is located in the middle of the state of Georgia in south-eastern United States. A humid subtropical climate with mild winters and hot moist summers is characteristic of most of Georgia. The average annual precipitation varies from about 1000 mm in central Georgia to more than 1900 mm in northeast Georgia. One of two annual precipitation maxima occurs in February or March, when between 100 and 150 mm of rain may be expected because of the high seasonal incidence of cyclonic activity over the south-eastern United States. The second maximum occurs in June and July, when precipitation from thunderstorm activity may bring 100 to 180 mm of rain monthly. The Ocmulgee River is one

of 14 major river basins defined in the state of Georgia. It begins southeast of the highly urbanized Atlanta metropolitan area in central Georgia, and downstream its watershed is dominated by agriculture and forested areas. The main stem extends approximately 410 km and flows southeast past the city of Macon to join the Oconee River. Together, the Oconee River and the Ocmulgee River form the Altamaha River near Lumber City. The confluence of these two rivers into the Altamaha comprises the largest river system entirely contained within the state of Georgia. The station Macon, Georgia, has a drainage area of 5,799 km^2.

The data of the Rhine and the Danube are provided by the Global Runoff Data Centre (GRDC) in Germany (http://grdc.bafg.de/). The daily discharge data of the Umpqua and the Ocmulgee are available from the website of the United States Geological Survey (USGS) http://water.usgs.gov/waterwatch/.

The locations of the gauging stations are shown in Figure 2.1, and the description of the gauging stations used in this study is listed in Table 2.1.

Figure 2.1 Locations of gauging stations used in the study

Table 2.1 Description of gauging stations used in the study

River	Station	Country	Drainage area (km^2)	Latitude	Longitude	Elevation (m)
Yellow	Tangnaihai	China	133,650	35.5	100.15 E	-
Yellow	Tonguan	China	682,141	34.617	110.3 E	-
Danube	Achleiten	Austria	76,653	48.582	13.504 E	288
Rhine	Lobith	Netherlands	160,800	51.84	6.11 E	10
Ocmulgee	Macon	USA	5,799	32.839	83.621 W	82
Umpqua	Elkton	USA	9,535	43.586	123.554W	27

2.1.2 Statistics of the Data Used

In this study, streamflow processes at four characteristic timescales will be analyzed, namely, one day, 1/3-month, one month and one year. Original data are daily average discharges. Daily data are aggregated to monthly series and annual series by taking the average of every month or every year. For the 1/3-monthly series, the 1st and 2nd 1/3-month streamflows are the averages of the first and the second 10-days' daily streamflows, and the 3rd 1/3-month could be the average of the last 8~11 days' daily streamflow of a month depending on the length of the month. The statistical characteristics, including the means, coefficient of variations, skewness, kurtosis and autocorrelation function at lag 1 (i.e., ACF(1)), of the streamflow series at different timescales are summarized in Table 2.2. Daily streamflow series are plotted in Figure 2.2 to 2.7.

Table 2.2 Statistical characteristics of streamflow series

River (Station)	Period of records	Timescale	Mean (m^3/s)	Coefficient of variation	Skewness	Kurtosis	ACF(1)
Yellow (TNH)	1956-2000	Daily	646	0.8653	1.864	5.034	0.994
		1/3-monthly	643	0.8538	1.770	4.472	0.884
		Monthly	643	0.8103	1.516	2.789	0.703
		Annual	646	0.2570	0.882	-0.076	0.301
Yellow (TG)	1962-2000	Daily	1106	0.8535	2.401	7.512	0.945
		1/3-monthly	1103	0.7951	2.195	5.711	0.856
		Monthly	1103	0.7416	2.141	5.231	0.682
		Annual	1106	0.3472	0.902	0.783	0.440
Danube (Achleiten)	1901-1990	Daily	1421	0.4631	1.464	4.500	0.962
		1/3-monthly	1422	0.4276	1.090	1.969	0.772
		Monthly	1420	0.3873	0.867	1.129	0.639
		Annual	1424	0.1594	0.0774	-0.830	0.094
Rhine (Lobith)	1901-1996	Daily	2217	0.5174	2.121	7.162	0.985
		1/3-monthly	2219	0.4831	1.755	4.602	0.713
		Monthly	2219	0.4182	1.230	2.143	0.544
		Annual	2217	0.2124	-0.135	-0.567	0.140
Ocmulgee (Macon)	1929-2001	Daily	76.1	1.3955	6.711	76.506	0.857
		1/3-monthly	76.4	1.0589	3.508	19.331	0.500
		Monthly	76.3	0.8362	1.903	4.653	0.537
		Annual	76.1	0.3403	0.556	0.441	0.254
Umpqua (Elkton)	1906-2001	Daily	210	1.4571	5.193	49.344	0.864
		1/3-monthly	211	1.1706	2.614	11.126	0.627
		Monthly	211	0.9905	1.625	3.325	0.621
		Annual	210	0.2752	0.401	-0.193	0.233

Note: The skewness and kurtosis are calculated with the moment method.

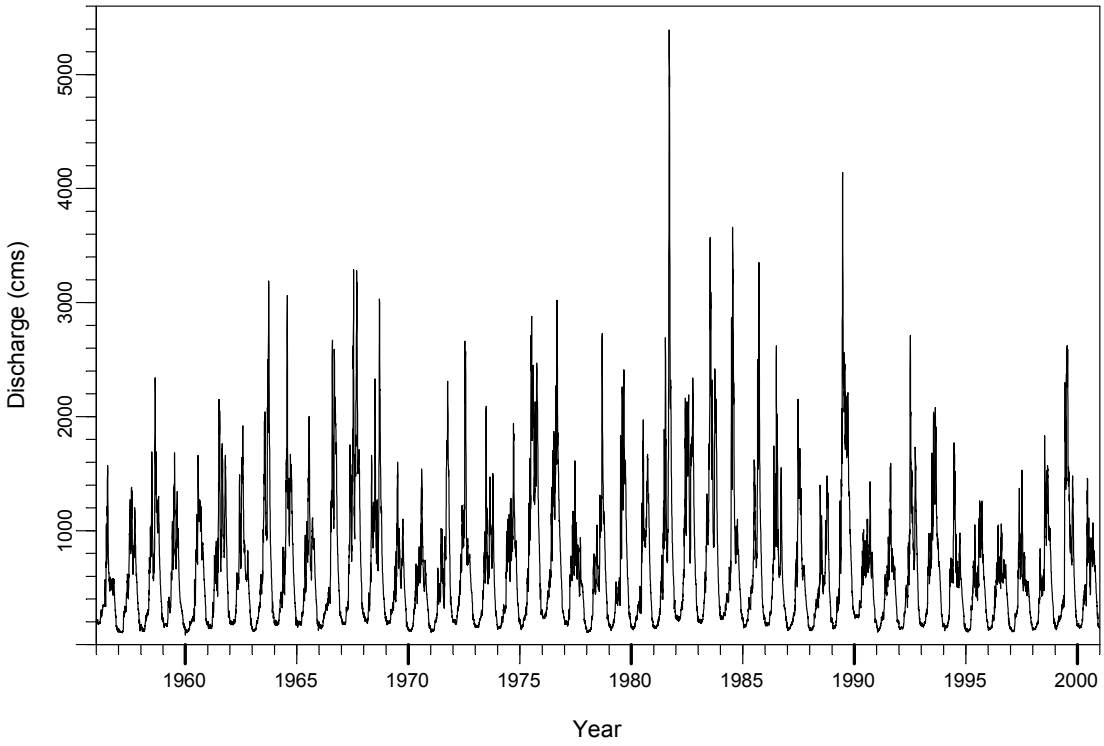


Figure 2.2 Average daily discharges of the upper Yellow River at TNH (1956-2000)

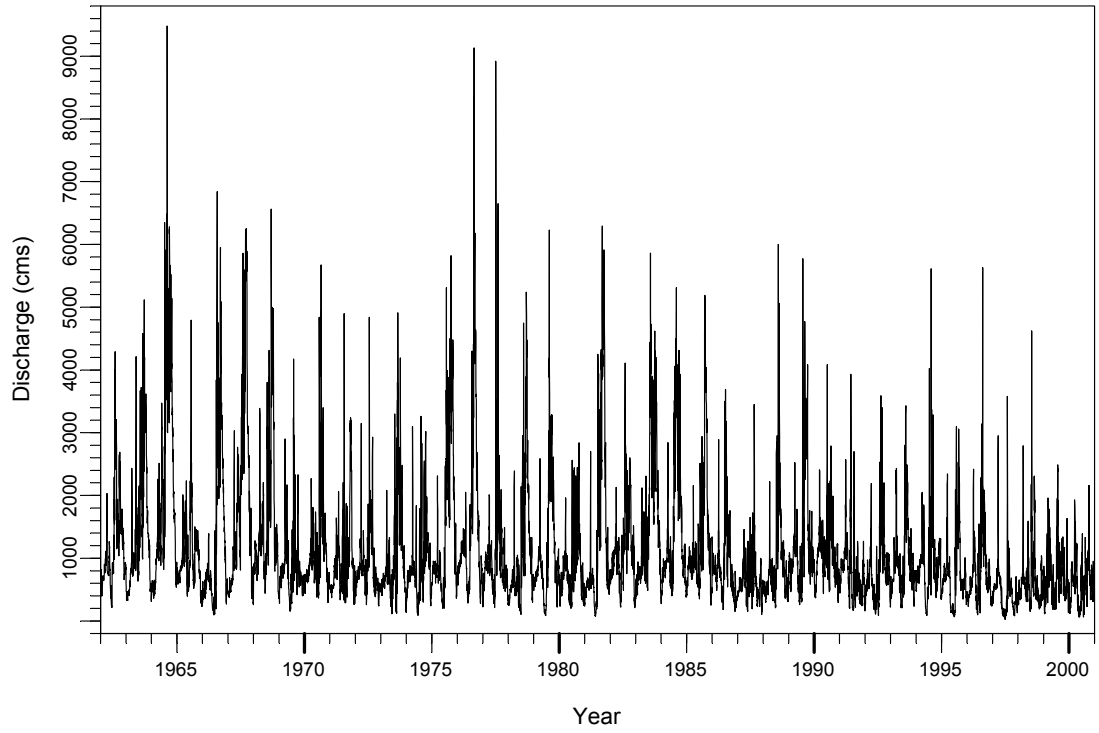


Figure 2.3 Average daily discharges of the middle Yellow River at TG (1962-2000)

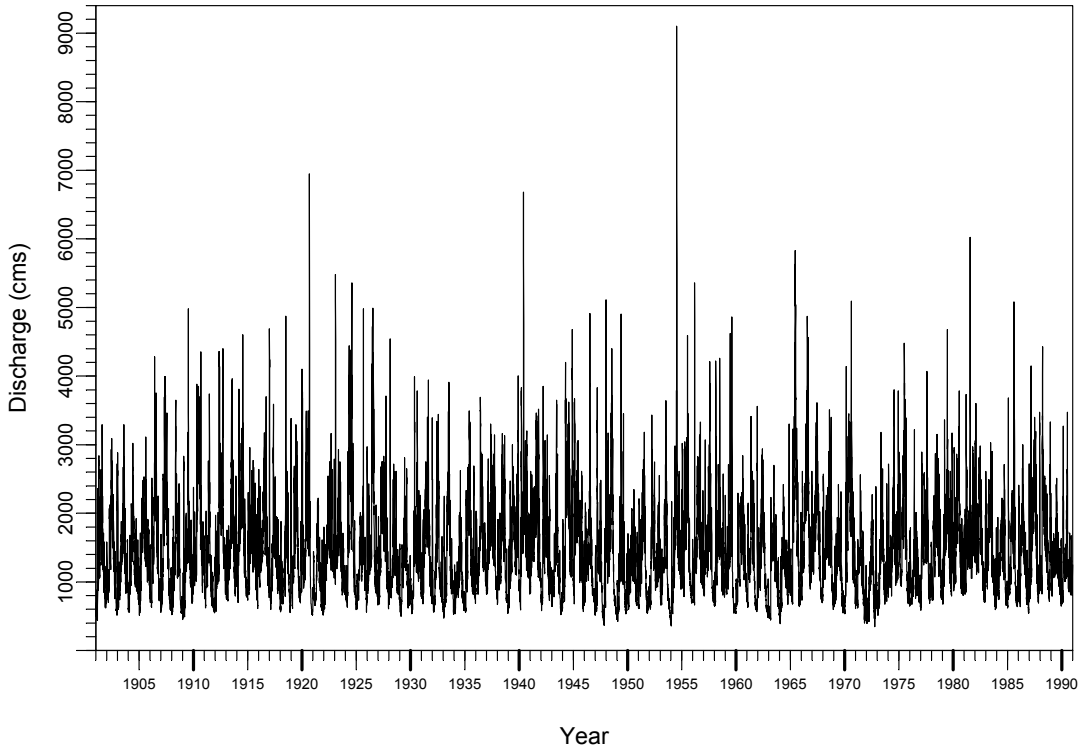


Figure 2.4 Average daily discharges of the Danube River at Achleiten (1901-1990)

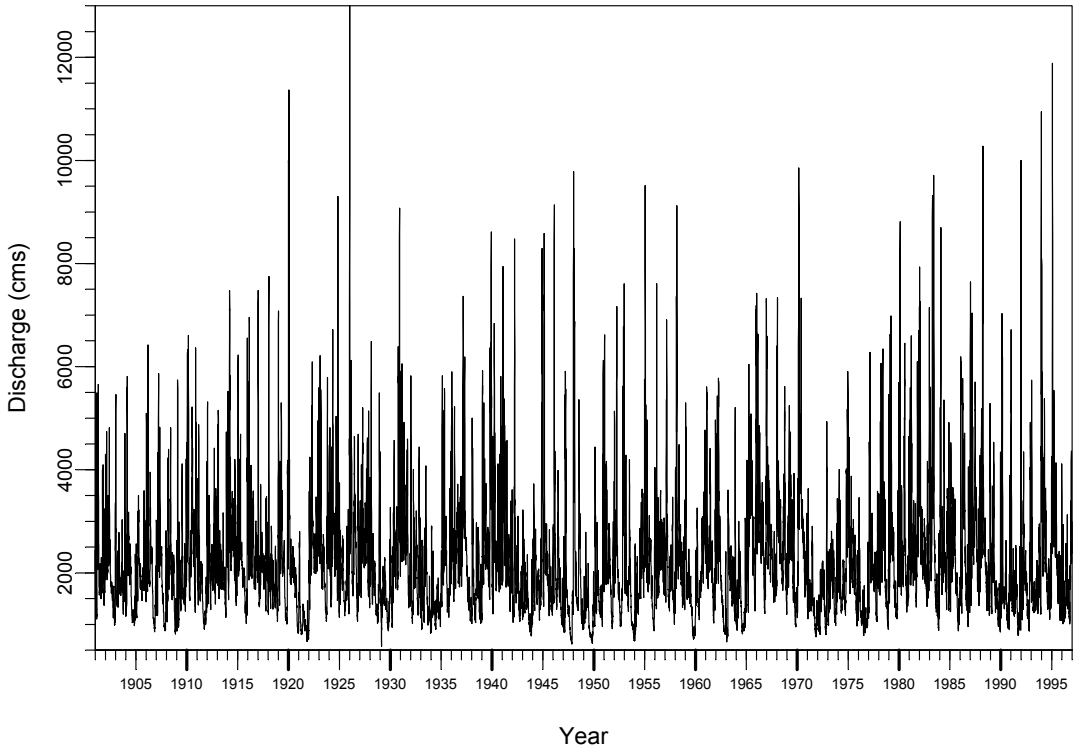


Figure 2.5 Average daily discharges of the Rhine River at Lobith (1901-1996)

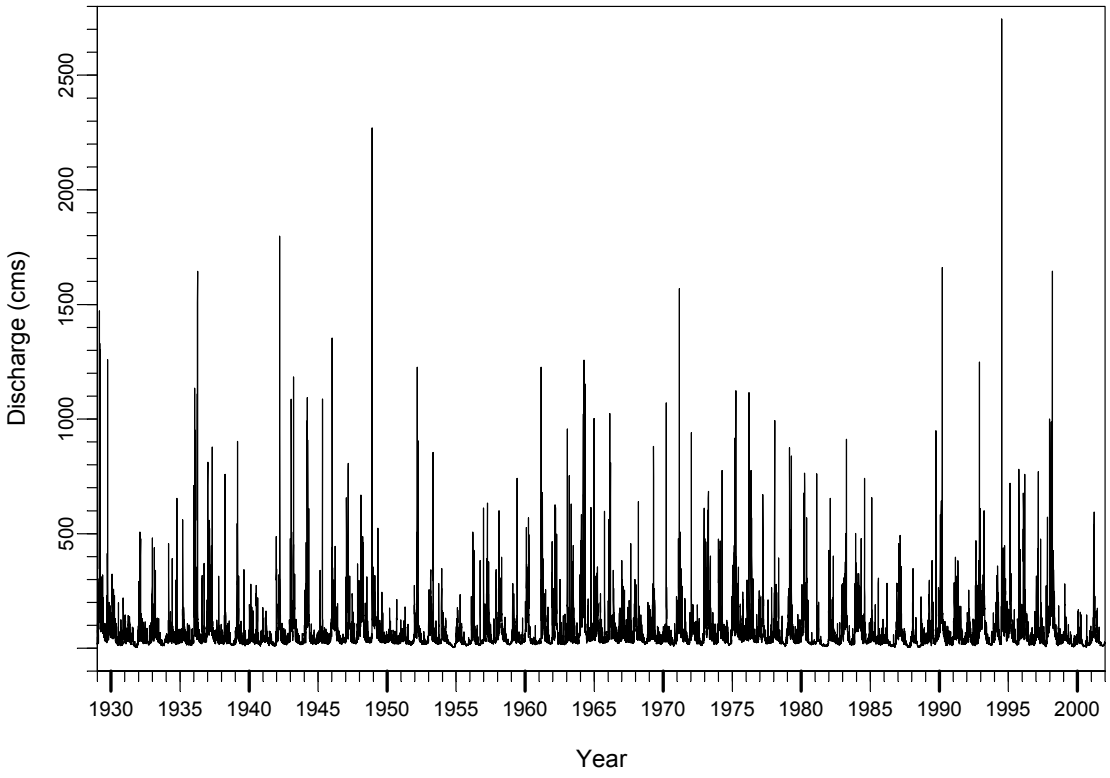


Figure 2.6 Average daily discharges of Ocmulgee River at Macon (1929-2001)

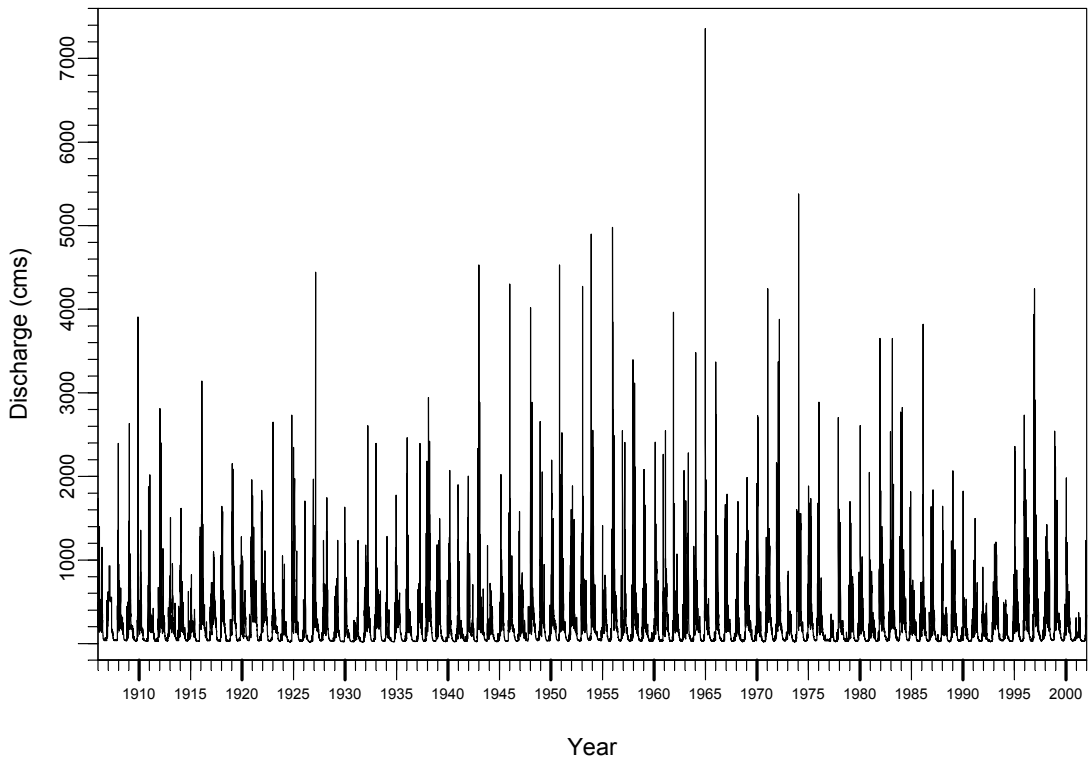


Figure 2.7 Average daily discharges of Umpqua River near Elkton (1906-2001)

2.2 Trend Analysis

Many hydrological time series exhibit trending behavior or nonstationarity. In fact, the trending behavior is a type of nonstationarity. But in this present study, they are treated separately. The purpose of a trend test is to determine if the values of a series have a general increase or decrease with the time increase, whereas the purpose of stationarity test is to determine if the distribution of a series is dependent on the time.

An important task in hydrological modeling is to determine if there is the existence of any trend in the data and how to achieve stationarity when the data is nonstationary. On the other hand, the possible effects of global warming on water resources have been the topic of many recent studies (e.g., Lettenmaier et al., 1999; Jain and Lall, 2001; Kundzewicz et al., 2004). Thus, detecting the trend and stationarity in a hydrological time series may help us to understand the possible links between hydrological processes and changes in the global environment. The focus of the trend analysis and stationarity test in this study is not to detect the changes of regional or world-wide streamflow processes. As a matter of fact, the presence of trends and nonstationarity is undesirable in further analysis. Therefore, we should make sure whether there is the presence of trend and nonstationarity or not, and if the presence of trend and nonstationarity is detected, the appropriate pre-processing procedure should be applied. In this section the issue of trend analysis is studied, and the nonstationarity problem will be addressed in the following section.

Non-parametric trend detection methods are less sensitive to outliers (extremes) than are parametric statistics such as Pearson's correlation coefficient. In addition, nonparametric test can test for a trend in a time series without specifying whether the trend is linear or nonlinear. Therefore, A rank-based nonparametric method, the Mann-Kendall's test (Kendall, 1938; Mann, 1945), is applied in this study to annual and monthly series.

2.2.1 Trend Test for Annual Streamflow Series

First of all, we test for the trend in annual series so as to get an overall view of the possible changes in streamflow processes.

2.2.1.1 *Mann-Kendall test*

Kendall (1938) proposed a measure *tau* to measure the strength of the monotonic relationship between x and y. Mann (1945) suggested using the test for the significance of Kendall's *tau* where one of the variables is time as a test for trend. The test is well known as the Mann-Kendall's test (referred to as MK test hereafter), which is powerful for uncovering deterministic trends. Under the null hypothesis H_0, that a series $\{x_1, \ldots, x_N\}$ come from a population where the random variables are independent and identically distributed, the MK test statistic is

$$S = \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \operatorname{sgn}(x_j - x_i), \text{ where } \operatorname{sgn}(x) = \begin{cases} +1, & x > 0 \\ 0, & x = 0 \\ -1, & x < 0 \end{cases} \tag{2.1}$$

And *tau* is estimated as:

$$\tau = \frac{2S}{N(N-1)}. \tag{2.2}$$

Kendall (1975) showed that the variance of S, $Var(S)$, for the situation where there may be ties (i.e., equal values) in the x values, is given by

$$\sigma_s^2 = \frac{1}{18}\left[N(N-1)(2N+5) - \sum_{i=1}^{m} t_i(t_i-1)(2t_i+5) \right], \tag{2.3}$$

where, m is the number of tied groups in the data set and t_i is the number of data points in the ith tied group.

Under the null hypothesis, the quantity z defined in the following equation is approximately standard normally distributed even for the sample size $N = 10$:

$$z = \begin{cases} (S-1)/\sigma_s & \text{if } S > 0 \\ 0 & \text{if } S = 0 \\ (S+1)/\sigma_s & \text{if } S < 0 \end{cases} \tag{2.4}$$

It has been found that the positive serial correlation inflates the variance of the MK statistic S and hence increases the possibility of rejecting the null hypothesis of no trend (von Storch, 1995). In order to reduce the impact of serial correlations, it is common to prewhiten the time series by removing the serial correlation from the series through $y_t = x_t - \phi x_{t-1}$, where y_t is the prewhitened series value, x_t is the original time series value, and ϕ is the estimated lag 1 serial correlation coefficient. The pre-whitening approach has been adopted in many trend-detection studies (e.g., Douglas et al., 2000; Zhang et al., 2001; Burn and Hag Elnur, 2002).

2.2.1.2 *MK test results*

The first step in time series analysis is visually inspecting the data. Significant changes in level or slope usually are obvious. The annual average streamflow series of the Yellow River at TNH and TG, the Rhine River at Lobith, the Umpqua River near Elkton and the Ocmulgee River at Macon are shown in Figure 2.8.

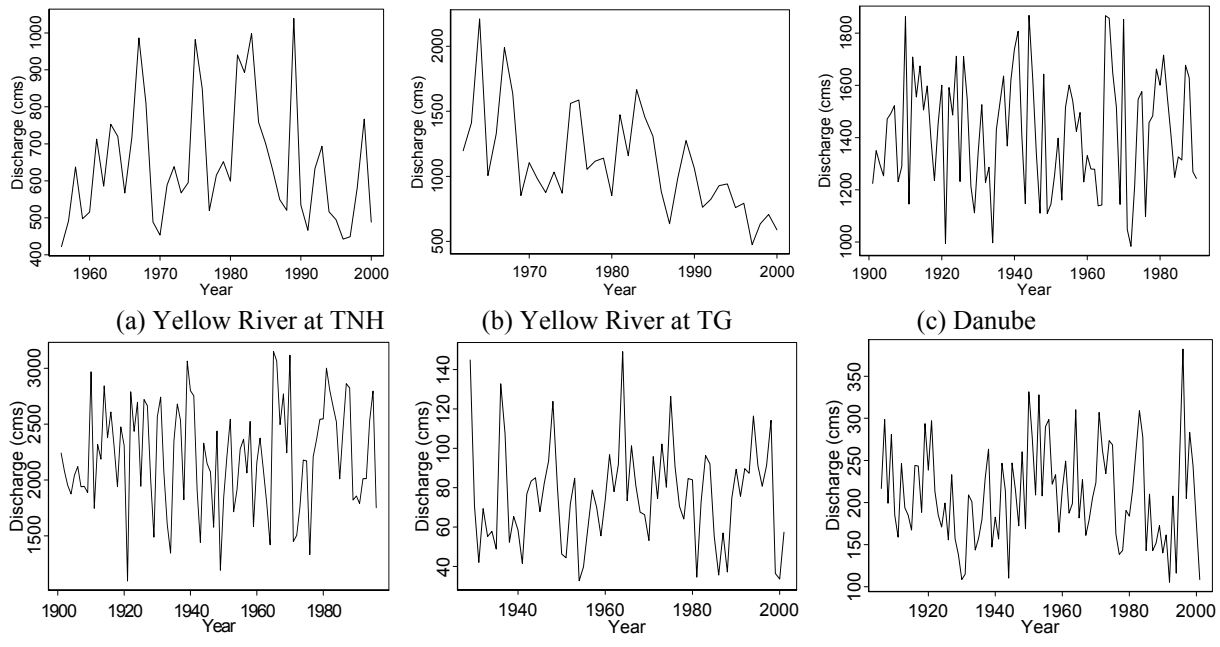


Figure 2.8 Annual average discharge series of the five rivers at six sites

From the visual inspection, it seems that except for the annual flow series of the Yellow River at TG which exhibits obvious downward trend, other annual series have no obvious trend. The MK test results are displayed in Table 2.3. The results are in agreement with the heuristic result by the visual examination.

Table 2.3 Mann-Kendall tests on Annual average discharge series

Streamflow	*tau*	z statistic	p-value
TNH	-0.1015	-0.9609	0.3366
TG	-0.3144	-2.7658	0.0057
Danube	0	0	1
Rhine	0.0467	0.6710	0.5022
Ocmulgee	0.1025	1.2688	0.2045
Umpqua	-0.0258	-0.3665	0.7140

Null hypothesis: *tau* = 0

2.2.2 Trend Test for Monthly Streamflow Series

The trend test for annual series gives us an overall view of the change in streamflow processes. To examine the possible changes occur in smaller timescale, we need to investigate the monthly flow series. Monthly streamflows usually exhibit strong seasonality. Trend test techniques for dealing with seasonality of univariate time series fall into three major categories (Helsel and Hirsh, 1992, pp 337-343): (1) fully nonparametric method, i.e., seasonal Kendall test; (2) mixed procedure, i.e., regression of deseasonalized series on time; (3) parametric method, i.e., regression of original series on time and seasonal terms. The first approach, namely, seasonal Kendall test will be used here.

2.2.2.1 *Seasonal Kendall test*

Hirsch et al. (1982) introduced a modification of the MK test, referred to as the seasonal Kendall test that allows for seasonality in observations collected over time by computing the Mann-kendall test on each of p seasons separately, and then combining the results. Compute the following overall statistic S':

$$S' = \sum_{j=1}^{p} S_j$$

where S_j is simply the S-statistic in the MK test for season j (j = 1, 2, ..., p) (see Equation (2.1)). When no serial dependence exhibit in the time series, the variance of S' is defined as $\sigma_{S'}^2 = \sum_{j=1}^{p} Var(S_j)$. When serial correlation is present, as in the case of monthly streamflow processes, the variance of S' is defined as (Hirsch and Slack, 1984)

$$\sigma_{S'}^2 = \sum_{j=1}^{p} Var(S_j) + \sum_{g=1}^{p-1} \sum_{h=g+1}^{p} \sigma_{gh} \qquad (2.5)$$

where σ_{gh} denotes the covariance between the MK statistic for season g and the MK statistic for season h. The covariance is estimated with the following procedures.

Let the matrix

$$X = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{pmatrix}$$

denote a sequence of observations taken over p seasons for n years. Let the matrix

$$R = \begin{bmatrix} R_{11} & R_{12} & \cdots & R_{1p} \\ R_{21} & R_{22} & \cdots & R_{2p} \\ \vdots & \vdots & & \vdots \\ R_{n1} & R_{n2} & \cdots & R_{np} \end{bmatrix}$$

denote the ranks corresponding to the observations in X, where the n observations for each season are ranked among themselves, that is, $R_{ij} = \frac{1}{2}\left[n+1+\sum_{k=1}^{n} \operatorname{sgn}(x_{ij} - x_{kj}) \right]$. Hirsch and Slack (1984) suggest using the following formula, given by Dietz and Killeen (1981), to estimate σ_{gh} in the case where there are no missing values:

$$\hat{\sigma}_{gh} = \frac{1}{3}\left[K_{gh} + 4\sum_{i=1}^{n} R_{ig} R_{ih} - n(n+1)^2 \right]$$

where $K_{gh} = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \operatorname{sgn}\left[(X_{jg} - X_{ig})(X_{jh} - X_{ih}) \right]$.

If there are missing values, $R_{ij} = \frac{1}{2}\left[n_j + 1 + \sum_{k=1}^{n_j} \operatorname{sgn}(X_{ij} - X_{kj}) \right]$, where n_j denotes the number of observations without missing values for season j. And the covariance between the MK statistic for season g and season h is estimated as

$$\hat{\sigma}_{gh} = \frac{1}{3}\left[K_{gh} + 4\sum_{i=1}^{n} R_{ig} R_{ih} - n(n_g+1)(n_h+1) \right].$$

Then the quantity z' defined in the following equation is approximately standard normally distributed:

$$z' = \begin{cases} (S'-1)/\sigma_{S'} & \text{if } S' > 0 \\ 0 & \text{if } S' = 0 \\ (S'+1)/\sigma_{S'} & \text{if } S' < 0 \end{cases} \tag{2.6}$$

The overall *tau* is the weighted average of the p seasonal τ's, defined as $\tau = \sum_{j=1}^{p} n_j \tau_j \Big/ \sum_{j=1}^{p} n_j$, where τ_j is the *tau* for season j, estimated with Equation (2.2).

Seasonal Kendall test is appropriate for testing for trend in each season when the trend is always in the same direction across all seasons. However, the trend may have different

directions in different seasons. Van Belle and Hughes (1984) suggested using the following statistic to test for heterogeneity in trend

$$\chi^2_{het} = \sum_{j=1}^{p} z_j^2 - p\bar{z}^2 \tag{2.7}$$

where z_j denotes the z-statistic for the jth season computed as $z_j = \frac{S_j}{(Var(S_j))^{1/2}}$, and $\bar{z} = \frac{1}{p}\sum_{j=1}^{p} z_j$. Under the null hypothesis of no trend in any season, the statistic defined in Equation (2.7) is approximately distributed as a chi-square random variable with p -1 degrees of freedom.

2.2.2.2 *Seasonal Kendall test results*

The six monthly streamflow processes are tested for the trend with the seasonal Kendall test which allows fort the serial dependence. And the heterogeneity in trend is also tested. The results are shown in Table 2.4. The results give the same conclusion as the test for annual series, that is, among 5 series, only the streamflow of the Yellow River at TG exhibits significant downward trend. Meanwhile, it is found that while the streamflow processes at TG present downward trend in general, the trend directions of every month are heterogeneous.

Therefore, the trend of streamflows at TG in each month is further investigated with the MK test. The results are shown in Table 2.5. It is seen that, for the streamflows of the Yellow River at TG, the trends in December to April, and in June, are not significant, whereas in other months, there are obvious downward trends. This indicates that the discharges at TG in the summer and autumn are significantly decreased, but in winter, the change is not significant. One reason for such kind of behaviour is the similar change pattern in the monthly rainfall in the area along middles reaches of the Yellow River (Fu et al., 2004). Another reason may be the runoff regulation of about 10 dams over the main channel and thousands of reservoirs along the tributaries in this basin, which were mainly built over the last 50 years.

Table 2.4 Seasonal Kendall Tests on Monthly Series

Streamflow	*tau*	z statistic	trend p-value	Het p-value
TNH	-0.0178	-0.2732	0.7847	0.3705
TG	-0.2431	-3.5561	0.0057	0.0039
Danube	-0.0084	-0.2010	0.8407	0.2558
Rhine	0.0089	0.2047	0.8378	0.5125
Ocmulgee	-0.0101	-0.2078	0.8354	0.5105
Umpqua	-0.0129	-0.3120	0.7550	0.8185

Null hypothesis of trend test: *tau* = 0
Null hypothesis of trend homogeneity test: *tau* of all seasons are equal to 0.
"Het" denotes the van Belle and Hughes heterogeneity test.

Table 2.5 Mann-Kendall Tests for streamflows at TG in each month

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
tau	-0.150	0.031	0.018	-0.144	-0.511	-0.179	-0.350	-0.296	-0.333	-0.457	-0.479	-0.066
p-value	0.183	0.790	0.885	0.200	0.000	0.110	0.002	0.008	0.003	0.000	0.000	0.561

2.3 Stationarity Test

In most applications of hydrological modelling, we have an assumption of stationarity. It is thus necessary to test for stationarity for the justification of using those models. On the other hand, sometimes the investigation of nonstationarity may give us some insights into the underlying physical mechanism, especially in the context of global changes. Therefore, testing for stationarity is an important topic in the field of hydrology.

2.3.1 Test Methods

There are roughly two groups of methods for testing stationarity. The first group is based on the idea of analyzing the statistical differences of different segments of a time series (e.g., Chen and Rao, 2002). If the observed variations in a certain parameter of different segments are found to be significant, that is, outside the expected statistical fluctuations, the time series is regarded as nonstationary. Another group of stationarity tests is based on statistics for the full sequence. We adopt the second approach here.

The stationarity test is carried out with two methods in this present study. The first one is the augmented Dickey-Fuller (ADF) unit root test that is first proposed by Dickey and Fuller (1979) and then modified by Said and Dickey (1984). It tests for the presence of unit roots in the series (difference stationarity). The other one is the KPSS test proposed by Kwiatkowski et al. (1992), which tests for the stationarity around a deterministic trend (trend stationarity) and the stationarity around a fixed level (level stationarity). KPSS test can also be modified to be used as a unit root test, but it was shown by Shin and Schmidt (1992) that the KPSS statistic, designed for use as a test for stationarity, was not as good a unit root test as other standard test. In particular, its power is noticeably less than the power of the Dickey-Fuller test (or other similar tests) against stationary alternatives.

2.3.1.1 *ADF test*

Dickey–Fuller unit-root tests are conducted through the ordinary least squares (OLS) estimation of regression models incorporating either an intercept or a linear trend. Consider the autoregressive AR (1) model

$$x_t = \rho x_{t\text{-}1} + \varepsilon_t, \;\; t = 1, 2, ..., N \tag{2.8}$$

where $x_0 = 0$; $|\rho| \le 1$ and ε_t is a real valued sequence of independent random variables with mean zero and variance σ^2. If $\rho = 1$, the process $\{x_t\}$ is nonstationary and it is known as a random walk process. In contrast, if $|\rho| < 1$, the process $\{x_t\}$ is stationary. The maximum likelihood estimator of ρ is the OLS estimator

$$\hat{\rho} = \left(\sum_{t=2}^{N} x_{t-1}^2 \right)^{-1} \sum_{t=2}^{N} x_t x_{t-1} \;. \tag{2.9}$$

Under the null hypothesis that $\rho = 1$, Dickey and Fuller (1979) showed that $\hat{\rho}$ is characterized by

$$N(\hat{\rho} - 1) = \frac{N \sum_{t=2}^{N} (x_t x_{t-1} - x_{t-1}^2)}{\sum_{t=2}^{N} x_{t-1}^2} \xrightarrow{D} \frac{\Lambda^2 - 1}{2\Gamma}, \tag{2.10}$$

where $(\Gamma, \Lambda) = (\sum_{i=1}^{\infty} \gamma_i^2 Z_i^2, \sum_{i=1}^{\infty} 2^{1/2} \gamma_i Z_i)$, with $\gamma_i = 2(-1)^{i+1}/[(2i - 1)\pi]$, and the Z_i are i.i.d N(0,1) distributed random variables.

The result with Equation (2.10) allows the point estimate $\hat{\rho}$ to be used by itself to test the null hypothesis of a unit root. Another popular statistic for testing the null hypothesis that $\rho = 1$ is based on the usual OLS *t*-test of this hypothesis,

$$t = \frac{\hat{\rho} - 1}{\hat{\sigma}_{\hat{\rho}}} \tag{2.11}$$

where $\hat{\sigma}_{\hat{\rho}}$ is the usual OLS standard error for the estimated coefficient, $\hat{\sigma}_{\hat{\rho}} = s_e \left(\sum_{t=2}^{N} x_{t-1}^2 \right)^{-1/2}$, and s_e denotes the standard deviation of the OLS estimate of the residuals in the regression model with Equation (2.8), estimated as

$$s_e^2 = \frac{1}{N-2} \sum_{t=2}^{N} (x_t^2 - \hat{\rho} x_{t-1})^2 .$$

Dickey and Fuller (1979) derived the limiting distribution of the statistic t under the null hypothesis that $\rho = 1$ as

$$t \xrightarrow{D} 2\Gamma^{-1/2}(\Lambda^2 - 1). \tag{2.12}$$

A set of tables of the percentiles of the limiting distribution of the statistic t under $\rho = 1$ is available in Fuller (1976, pp 371, 373). The test rejects $\rho = 1$ when t is "too negative".

The unit root test described above is valid if the time series $\{x_t\}$ is well characterized by an AR(1) with white noise errors. Many hydrological time series, however, have a more complicated dynamic structure than is captured by a simple AR(1) model. The basic autoregressive unit root test can be augmented (referred to as ADF test) to accommodate general ARMA(p, q) models with unknown orders (Said and Dickey, 1984; Hamilton, 1994, pp 516-530). The ADF test is based on estimating the test regression

$$x_t = \beta D_t + \phi x_{t-1} + \sum_{j=1}^{p} \psi_j \nabla x_{t-j} + \varepsilon_t, \ t = 1, 2, ..., N \tag{2.13}$$

where D_t is a vector of deterministic terms (constant, trend, etc.). The p lagged difference terms, ∇x_{t-j}, are used to approximate the ARMA structure of the errors, and the value of p is set so that the error ε_t is serially uncorrelated. Said and Dickey (1984) show that the Dickey-Fuller procedure, which was originally developed for autoregressive representations of known order, remains valid asymptotically for a general ARIMA(p, 1, q) process in which p and q are unknown orders.

2.3.1.2 *KPSS test*

Let $\{x_t\}$, $t = 1, 2, \ldots, N$, be the observed series for which we wish to test stationarity. Assume that we can decomposes the series into the sum of a deterministic trend, a random walk, and a stationary error with the following linear regression model

$$x_t = r_t + \beta t + \varepsilon_t , \tag{2.14}$$

where r_t is a random walk, i.e., $r_t = r_{t-1} + u_t$, and u_t is iid $N(0, \sigma_u^2)$; βt is a deterministic trend; ε_t is a stationary error.

To test in this model if x_t is a trend stationary process, namely, the series is stationary around a deterministic trend, the null hypothesis will be $\sigma_u^2 = 0$, which means that the intercept is a fixed element, against the alternative of a positive σ_u^2. In another stationarity case, the level stationarity, namely, the series is stationary around a fixed level, the null hypothesis will be $\beta = 0$. So that, under the null hypothesis, in the case of trend stationary, the residuals e_t ($t = 1, 2, \ldots, N$) are from the regression of x on an intercept and time trend, $e_t = \varepsilon_t$; whereas in the case of level stationarity, the residuals e_t are from a regression of x on intercept only, that is $e_t = x_t - \overline{x}$.

Let the partial sum process of the e_t as $S_t = \sum_{j=1}^{t} e_j$, and σ^2 be the long-run variance of e_t, which is defined as $\sigma^2 = \lim N^{-1} E\left[S_N^2\right]$. The consistent estimator of σ^2 can be constructed from the residuals e_t by (Newey and West, 1987)

$$\hat{\sigma}^2(p) = \frac{1}{N}\sum_{t=1}^{N} e_t^2 + \frac{2}{N}\sum_{j=1}^{p} w_j(p) \sum_{t=j+1}^{N} e_t e_{t-j} , \tag{2.15}$$

where p is the truncation lag, $w_j(p)$ is an optional weighting function that corresponds to the choice of a special window, e.g., Bartlett window (Bartlett, 1950) $w_j(p) = 1 - j/(p+1)$.

Then the KPSS test statistic is given by

$$KPSS = N^{-2}\sum_{t=1}^{N} S_t^2 \big/ \hat{\sigma}^2(p) . \tag{2.16}$$

Under the null hypothesis of level stationary, $KPSS \to \int_0^1 V_1(r)^2 dr$, where $V_1(r)$ is a standard Brownian bridge: $V_1(r) = B(r) - rB(1)$ and $B(r)$ is a Brownian motion process on $r \in [0, 1]$. Under the null hypothesis of trend stationary, $KPSS \to \int_0^1 V_2(r)^2 dr$, where $V_2(r)$ is the second level Brownian bridge, given by $V_2(r) = B(r) + (2r - 3r^2)B(1) + (-6r + 6r^2)\int_0^1 B(s)ds$. The upper tail critical values of the asymptotic distribution of the KPSS statistic are listed in Table 2.6, given by Kwiatkowski et al. (1992).

Table 2.6 Upper tail critical values for the KPSS test statistic asymptotic distribution

Distribution	Upper tail percentiles			
	0.1	0.05	0.025	0.01
$\int_0^1 V_1(r)^2 dr$	0.347	0.463	0.574	0.739
$\int_0^1 V_2(r)^2 dr$	0.119	0.146	0.176	0.216

2.3.2 Results of Stationarity Tests

Because on the one hand both the ADF test and the KPSS test are based on the linear regression, which assumes a normal distribution; on the other hand, the log-transformation can convert an exponential trend possibly present in the data into a linear trend, therefore, it is common to take logs of the data before applying the ADF test and the KPSS test (e.g., Gimeno et al., 1999). In this study, the streamflow data are also log-transformed before being applied to stationarity tests.

An important practical issue for the implementation of the ADF test as well as the KPSS test is the specification of the truncation lag values of p in Equation (2.13) and (2.15). The KPSS test statistics are fairly sensitive to the choice of p, and in fact for every series the value of the test statistic decreases as p increases (Kwiatkowski et al., 1992). If p is too small then the remaining serial correlation in the errors will bias the test toward rejecting the null hypothesis. If p is too large then the power of the test will suffer. The larger the p, the less likely was the null hypothesis to be rejected. Following Schwert (1989), Kwiatkowski et al. (1992) and some others, the number of lag length is subjectively chosen as $p = \text{int}[x(N/100)^{1/4}]$, with $x = 4, 12$ in the present study for streamflow processes at from monthly to daily timescales. For annual series, because the autocorrelation at lag one is very low, so it is generally enough to exclude the serial correlation by choosing $p = 1$. The function *unitroot* and *stationaryTest* implemented in S+FinMetrics version 1.0 (Zivot and Wang, 2002) are used to do the ADF test and KPSS test. The stationarity test results are reported in Table 2.7.

The test results show that, except for the streamflow process of the Yellow River at TG which has significant downward trend at different timescales, all the other streamflow series appear to be stationary, since we cannot accept the unit root hypothesis with ADF test at 1% significance level and cannot reject the level stationarity hypothesis with KPSS test mostly at the 10% level or at least at the 2.5% level. In fact, the level stationarity is a major criterion in selecting streamflow series in the present study, while the use of the streamflow series at TG is for the purpose of comparison. For some series (such as the daily series of Rhine at Lobith, etc.) the hypothesis of trend stationarity is rejected by the KPSS test or just accepted at a low significance level, especially when the lag p is small. But this seems to be unreasonable, because the level stationarity can also be interpreted as the stationarity around a deterministic trend with a slope of zero. Therefore, we still consider these series stationary.

Two issues should be noticed. Firstly, although no significant cycle with a period longer than one year is detected with spectral analysis for any streamflow series in the study (results are not shown here for saving space), as we will see later in Section 2.4, streamflow processes normally exhibit strong seasonality, therefore, have periodic stationarity, rather than the stationarity we talk about normally. According to the results shown in Table 2.7, KPSS test is not powerful enough to distinguish the periodic stationarity from the stationarity in normal sense. Secondly, it is not clear how the presence of seasonality impacts the test of stationarity. Besides testing for nonstationarity in log-transfomed series, we have also tested the stationarity for the deseasonalized streamflow series. The deseasonalization is conducted by firstly taking log-transformation, then subtracting the seasonal (daily, 1/3-monthly or monthly) mean values and dividing by seasonal standard deviations. The results are presented in Table 2.8, which show that all the test results are generally larger for KPSS test and “less negative” for ADF test. In consequence, the p-values decrease for KPSS test, indicating the increase of the probability of rejecting the hypothesis of stationariy, and increase for ADF test, indicating the increase of the probability (though still very small) of accepting the hypothesis of unit root. That is, the removal of seasonality in the mean and variance tends to make the streamflow

series less stationary, or at least from the point of view of the KPSS test. This is an issue open for future investigation.

Table 2.7 Stationarity test results for log-transformed streamflow series

Station	Series	Lag	KPSS level stationary		KPSS trend stationary		ADF unit root	
			results	*p*-value	results	*p*-value	results	*p*-value
Yellow (TNH)	Daily	14	0.366	>0.05	0.366	<0.01	-7.6	4.03E-11
		42	0.138	>0.1	0.138	>0.05	-10.89	2.18E-23
	1/3-montly	8	0.078	>0.1	0.078	>0.1	-15.16	1.88E-40
		24	0.113	>0.1	0.113	>0.1	-8.369	2.49E-13
	Monthly	6	0.084	>0.1	0.084	>0.1	-14.2	1.26E-31
		18	0.115	>0.1	0.115	>0.1	-5.982	2.11E-06
	Annual	1	0.186	>0.1	0.1797	>0.01	-4.689	2.53E-03
Yellow (TG)	Daily	13	8.6673	<0.01	0.6473	<0.01	-13.38	1.12E-34
		41	3.5895	<0.01	0.2744	<0.01	-12.4	4.83E-30
	1/3-montly	7	2.3768	<0.01	0.1861	>0.01	-12.55	5.86E-29
		23	1.7241	<0.01	0.166	>0.025	-7.774	2.11E-11
	Monthly	5	1.8194	<0.01	0.1567	>0.025	-8.661	2.08E-13
		17	1.0985	<0.01	0.1239	>0.05	-4.7	7.69E-04
	Annual	1	1.0367	<0.01	0.1277	>0.05	-4.665	3.17E-03
Danube (Achleiten)	Daily	17	0.173	>0.1	0.1699	>0.025	-16.96	6.18E-53
		51	0.0737	>0.1	0.0724	>0.1	-14.32	1.93E-39
	1/3-montly	9	0.0486	>0.1	0.048	>0.1	-15.71	9.98E-45
		28	0.0539	>0.1	0.0533	>0.1	-9.263	1.20E-16
	Monthly	7	0.0478	>0.1	0.0472	>0.1	-14.44	5.57E-36
		21	0.0445	>0.1	0.0441	>0.1	-7.056	3.01E-09
	Annual	1	0.0347	>0.1	0.0335	>0.1	-8.465	7.02E-10
Rhine (Lobith)	Daily	17	0.413	>0.05	0.394	<0.01	-19.23	6.58E-65
		51	0.186	>0.1	0.178	>0.01	-14.54	1.64E-40
	1/3-montly	9	0.119	>0.1	0.114	>0.1	-13.45	3.49E-34
		29	0.076	>0.1	0.073	>0.1	-8.13	1.01E-12
	Monthly	7	0.088	>0.1	0.081	>0.1	-10.18	1.97E-19
		22	0.064	>0.1	0.059	>0.1	-6.573	5.71E-08
	Annual	1	0.0702	>0.1	0.0496	>0.1	-8.57	3.23E-10
Ocmulgee (Macon)	Daily	16	0.543	>0.025	0.408	<0.01	-16.21	5.26E-49
		48	0.228	>0.1	0.171	>0.025	-11.61	1.39E-26
	1/3-montly	9	0.128	>0.1	0.1	>0.1	-13.5	4.11E-34
		27	0.121	>0.1	0.095	>0.1	-8.515	5.91E-14
	Monthly	6	0.097	>0.1	0.086	>0.1	-13.73	2.19E-32
		20	0.081	>0.1	0.072	>0.1	-5.473	2.33E-05
	Annual	1	0.0773	>0.1	0.0749	>0.1	-6.311	6.27E-06
Umpqua (Elkton)	Daily	17	0.254	>0.1	0.242	<0.01	-13.47	2.85E-35
		51	0.101	>0.1	0.096	>0.1	-15.23	5.27E-44
	1/3-montly	9	0.061	>0.1	0.059	>0.1	-21.25	3.99E-71
		29	0.136	>0.1	0.133	>0.1	-9.894	4.94E-19
	Monthly	7	0.079	>0.1	0.08	>0.1	-20.1	2.66E-58
		22	0.133	>0.1	0.132	>0.05	-5.856	3.19E-06
	Annual	1	0.1334	>0.1	0.1328	>0.05	-7.124	1.11E-07

Table 2.8 Stationarity test results for log-transformed and deseasonalized streamflow series

Station	Series	Lag	KPSS level stationary		KPSS trend stationary		ADF unit root	
			results	*p*-value	results	*p*-value	results	*p*-value
Yellow (TNH)	Daily	14	2.4024	<0.01	2.3961	<0.01	-11.940	5.40E-28
		42	0.9972	<0.01	0.9946	<00.01	-8.869	2.05E-15
	1/3-montly	8	0.5581	>0.025	0.5579	<00.01	-6.842	9.75E-09
		24	0.266	>0.1	0.2659	<00.01	-5.133	1.06E-04
	Monthly	6	0.298	>0.1	0.297	<00.01	-5.005	2.14E-04
		18	0.1742	>0.1	0.1737	>0.025	-5.123	1.29E-04
Yellow (TG)	Daily	13	12.5127	<0.01	1.2348	<00.01	-16.66	4.60E-51
		41	5.5909	<0.01	0.5794	<00.01	-11.220	8.30E-25
	1/3-montly	7	3.3145	<0.01	0.3638	<00.01	-8.294	4.89E-13
		23	1.5704	<0.01	0.1904	>0.01	-5.043	1.61E-04
	Monthly	5	1.7948	<0.01	0.2086	>0.01	-6.385	2.66E-07
		17	0.8977	<0.01	0.1203	>0.05	-4.833	4.53E-04
Danube (Achleiten)	Daily	17	0.2835	>0.1	0.2892	<0.01	-21.79	3.07E-78
		51	0.1366	>0.1	0.1394	>0.05	-15.71	1.82E-46
	1/3-montly	9	0.0934	>0.1	0.0951	>0.1	-12.59	2.94E-30
		28	0.054	>0.1	0.0549	>0.1	-8.038	2.05E-12
	Monthly	7	0.0577	>0.1	0.0589	>0.1	-8.712	2.81E-14
		21	0.0407	>0.1	0.0415	>0.1	-6.307	2.73E-07
Rhine (Lobith)	Daily	17	0.5229	>0.025	0.5085	<0.01	-19.25	4.93E-65
		51	0.2347	>0.1	0.2282	<0.01	-14.01	6.76E-38
	1/3-montly	9	0.15	>0.1	0.1452	>0.05	-11.93	2.21E-27
		29	0.0797	>0.1	0.0772	>0.1	-8.065	1.65E-12
	Monthly	7	0.0966	>0.1	0.0897	>0.1	-8.369	3.38E-13
		22	0.0657	>0.1	0.0611	>0.1	-6.463	1.09E-07
Ocmulgee (Macon)	Daily	16	0.9316	<0.01	0.7318	<0.01	-21.160	7.05E-75
		48	0.4387	>0.05	0.3449	<0.01	-12.460	1.92E-30
	1/3-montly	9	0.2471	>0.1	0.2002	>0.01	-9.924	4.76E-19
		27	0.1347	>0.1	0.1092	>0.1	-7.563	6.93E-11
	Monthly	6	0.1366	>0.1	0.1253	>0.05	-8.025	5.45E-12
		20	0.0836	>0.1	0.0766	>0.1	-4.864	3.65E-04
Umpqua (Elkton)	Daily	17	1.2445	<0.01	1.2436	<0.01	-21.170	4.38E-75
		51	0.5697	>0.025	0.5694	<0.01	-14.500	2.38E-40
	1/3-montly	9	0.3536	>0.05	0.355	<0.01	-12.830	2.24E-31
		29	0.2	>0.1	0.2009	>0.01	-7.936	4.31E-12
	Monthly	7	0.2109	>0.1	0.2151	>0.01	-9.110	1.19E-15
		22	0.1365	>0.1	0.1392	>0.05	-5.166	9.44E-05

2.4 Seasonality Analysis

2.4.1 Seasonality in Mean and Variance

The dynamics of streamflow are often dominated by annual variations. How well the seasonality is captured is a very important criterion for assessing a stochastic model for streamflow. The seasonality of hydrological processes is often described in terms of the mean values, the variances, the extrema, and the probability distribution of the variable in each season (in general, a season may denote a day, a month, etc.). We will use the daily streamflow series to present the approaches we adopt here for analysis the seasonality. The same approaches can be easily adapted to the cases of 1/3-monthly series and monthly series.

To make it convenient to analyze the seasonality of a daily flow series of N years, we rewrite it as the following matrix form:

$$X = \begin{Bmatrix} x_{1,1} & x_{1,2} & \dots & x_{1,365} \\ x_{2,1} & x_{2,2} & \dots & x_{2,365} \\ \vdots & \vdots & x_{j,i} & \vdots \\ x_{N,1} & x_{N,2} & \dots & x_{N,365} \end{Bmatrix}, \tag{2.17}$$

where the rows denote year 1 ~ N, the columns denote day 1 ~ 365. For simplicity, the 366th days of leap years are omitted. This would not introduce major errors when analyzing seasonality of daily flows.

Consequently, the mean value, standard deviation and coefficient of variation of each column of the matrix are the daily mean discharge, standard deviation and coefficient of variation (CV) of daily discharges for each day over the year. They are easily calculated as follows:

$$\textit{Mean value}: \ \overline{x}_i = \frac{1}{N}\sum_{j=i}^{N} x_{j,i} \tag{2.18}$$

$$\textit{Standard deviation}: \ s_i = \left(\frac{1}{N}\sum_{j=1}^{N} (x_{j,i} - \overline{x}_i)^2 \right)^{1/2} \tag{2.19}$$

$$\textit{Coefficient of variation}: \ CV_i = \frac{s_i}{\overline{x}_i}$$

Daily mean values and standard deviations of the six streamflow processes are shown in Figure 2.9(a ~ f), and the daily variations in CVs are shown in Figure 2.10 (a ~c). It is shown that, days with high mean values have also high standard deviations, this is a property which has been well recognized (e.g., Mitosek, 2000). But two exceptional cases here are the streamflow processes of Danube and Ocmulgee. Danube has a clear seasonality in mean values, but no clear seasonality in variances. In consequence, it has a similar seasonal pattern in CVs to the Rhine River, as shown in Figure 2.10(b). Ocmulgee has no clear seasonal variations in CVs although it has clear seasonality in means and variances. In June, thunderstorm activity results in high CV values in the daily streamflow process of Ocmulgee, as shown in Figure 2.10(c).

Two special points should be noted about the variations in streamflow processes of the Yellow River:

(1) Streamflow processes of the Yellow River at both TNH and TG are characterized by a bimodal distribution. Extrema occur in July and September at TNH and in late March to early April and August at TG. However, the causes of bimodality of the two streamflow processes are different. The bimodality of the streamflow process at TNH exits mainly in response to the bimodal distribution of rainfall, whereas the first peaks of the streamflow process at TG is caused by snowmelt water and the break-up of the river-ice jam in spring and the second peak is due to concentrated rainfall.

(2) Although the contributing area of TG is about as 5 times larger as that of TNH, the streamflow process at TNH changes much smoother than that at TG, as indicated by CVs shown in Figure 2.10. This is mainly because of less rainfall variability and much less anthropogenic disturbances in the watershed above TNH.

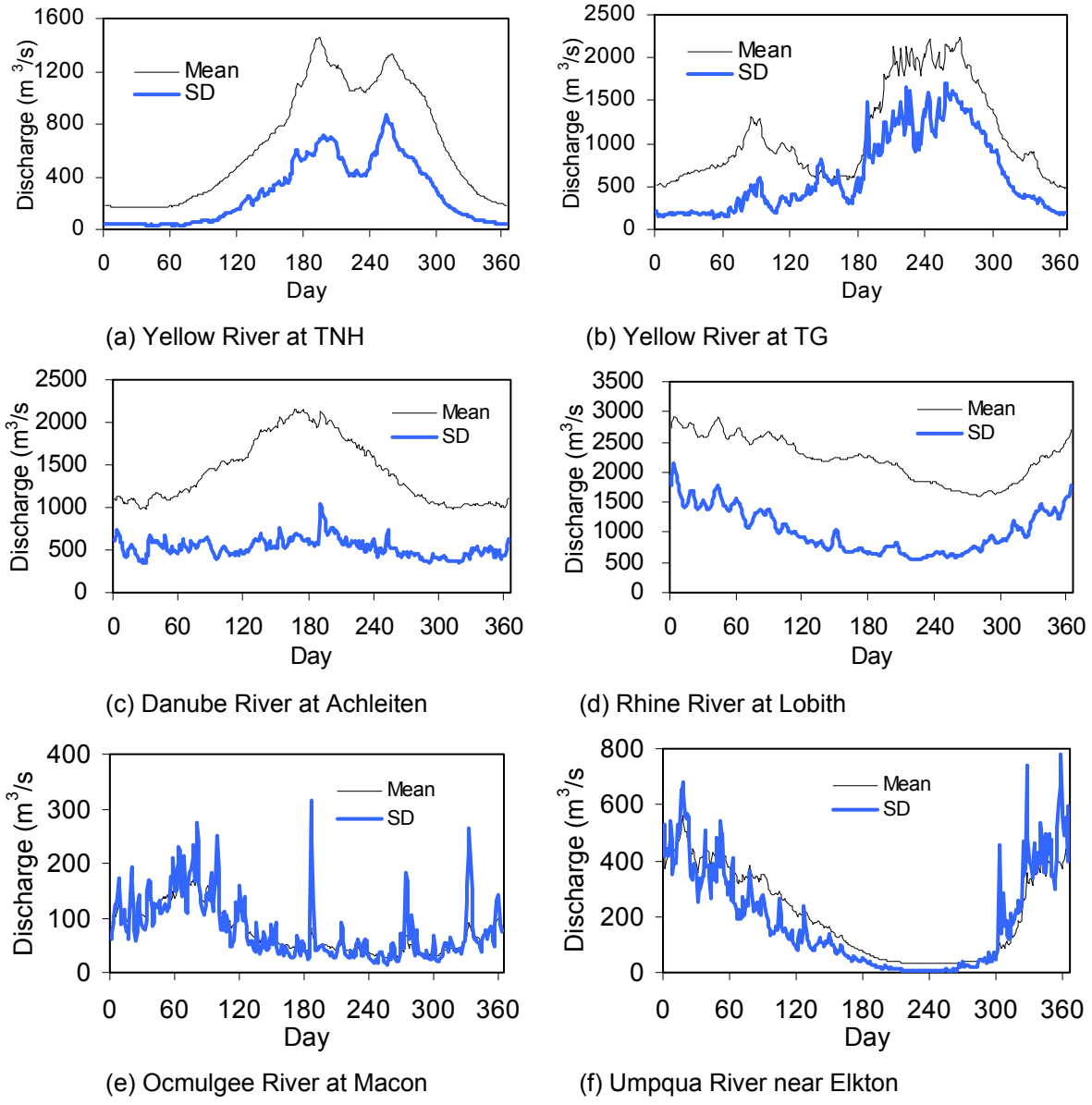
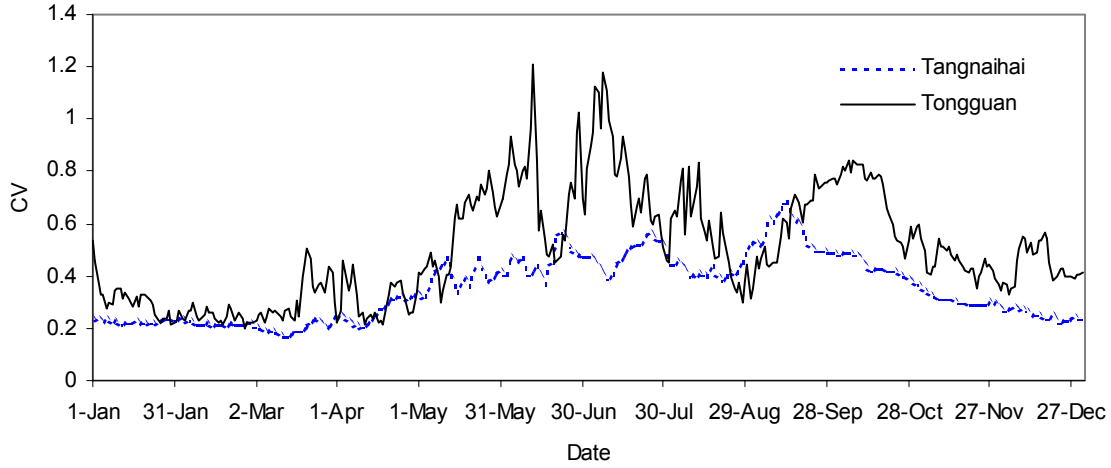
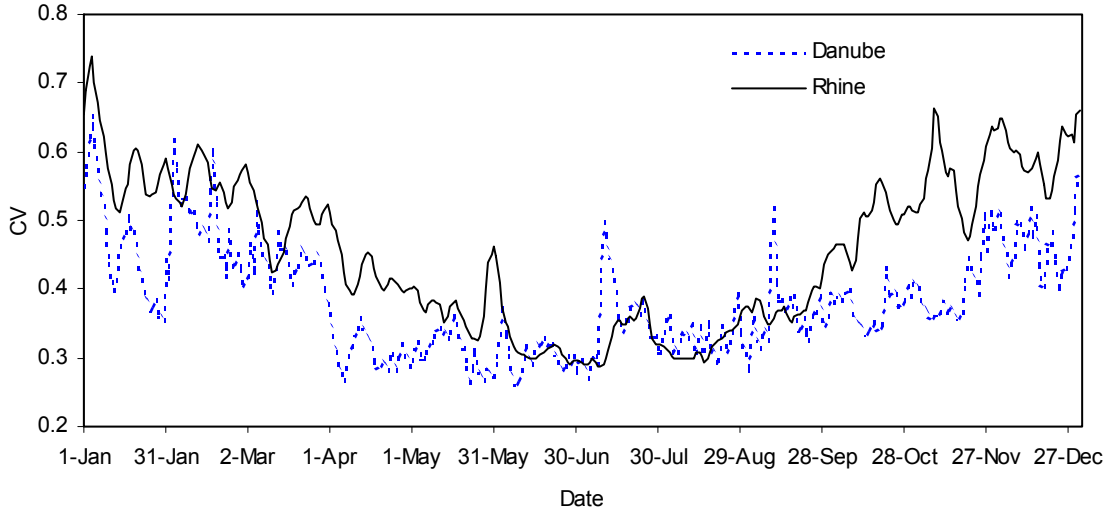


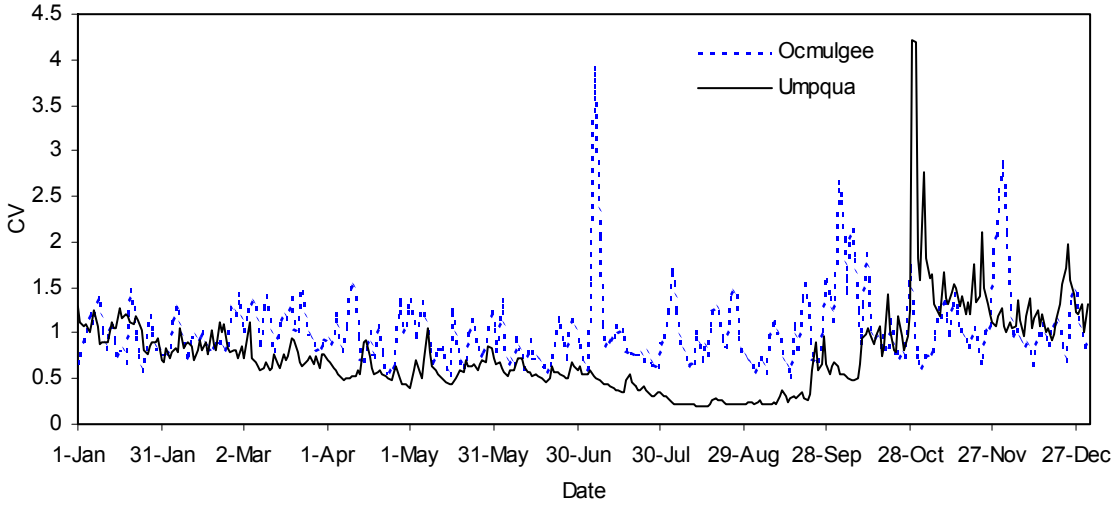
Figure 2.9 Variation in daily mean and standard deviation of streamflow processes



(a) Yellow River at TNH and TG



(b) Danube River at Achleiten and Rhine River at Lobith



(c) Ocmulgee River at Macon and Umpqua River near Elkton

Figure 2.10 Seasonal variation in CVs of streamflow processes

2.4.2 Detrend, Normalization and Deseasonalization

After trend analysis and seasonality analysis, we can remove trend component and seasonal component out of the original river flow series, and get an approximately stationary process, then further analyse autocorrelation properties and long-memory properties.

Because streamflow series are skewed and heavily tailed, whereas many models, such as regression models or autoregressive moving average (ARMA) models, require the time series data to be normally distributed, it is thus necessary to normalize the data to make them at least approximately normally distributed. The most popular approach is the Box-Cox transformation (Box and Cox, 1964):

$$x = \begin{cases} \lambda^{-1}[(x+c)^{\lambda} - 1] & \lambda \neq 0 \\ \ln(x+c) & \lambda = 0 \end{cases}$$

Usually we simply take logarithm to normalize the data. After log-transformation, we can estimate the trend by fitting a regression models if the trend is present, and then subtract it out of the original series.

The deseasonalization can be viewed as the standardization for each season (in the case of daily streamflow series, each season means each day). To do this, we use the daily mean $\overline{x}_i$, standard deviation s_i given by Equation (2.18) and (2.19), then apply to each element $x_{j,i}$ in matrix (2.17) the following standardization transformation:

$$y_{j,i} = \frac{x_{j,i} - \overline{x}_i}{s_i} \tag{2.20}$$

With the above pre-processing procedure, the seasonality in mean values and standard deviations in the streamflow series is removed. With such deseasonalized series, we go further to make autocorrelation analysis.

2.4.3 Seasonality in Autocorrelation Structures

Given a time series $\{x_i\}$, i = 1, 2, …, n, the autocorrelation function (ACF) at lag k for the time series is given by (Box and Jenkins, 1976) $\hat{\rho}(k) = c_k / c_0$, where k = 0, 1, 2, …, and $c_k = \frac{1}{n-k}\sum_{i=1}^{n-k}(x_i - \overline{x})(x_{i+k} - \overline{x})$. The ACF obtained in this way takes the whole time series into consideration, which reflects the overall autocorrelation property for the time series, but to examine the seasonal variation in the autocorrelation structure of a daily streamflow series, we need to calculate values of the autocorrelation coefficient between column vector X_i and X_{i+k} of matrix (2.17), where i = 1, 2, …, 365 and k = 0, 1, 2, …, $k_{\max}$, ($k_{\max} \leq 365$) (Mitosek, 2000), namely,

$$\hat{\rho}_i(k) = \begin{cases} \dfrac{\frac{1}{N}\sum_{j=1}^{N}(x_{j,i} - \overline{x}_i)(x_{j,i+k} - \overline{x}_{i+k})}{s_i s_{i+k}}, & i+k \leq 365 \\ \dfrac{\frac{1}{N-1}\sum_{j=1}^{N-1}(x_{j,i} - \overline{x}_i)(x_{j+1,i+k-365} - \overline{x}_{i+k-365})}{s_i s_{i+k-365}}, & i+k > 365 \end{cases} \tag{2.21}$$

where， $\bar{x}_i$ and s_i are the same as in Equation (2.18) and (2.19), N is the number of years, and

$$\bar{x}_{i+k-365} = \frac{1}{N-1}\sum_{j=i}^{N-1} x_{j+1,i+k-365}$$

$$s_{i+k-365} = \left(\frac{1}{N-1}\sum_{j=1}^{N} x_{j+1,i+k-365} - \bar{x}_{i+k-365})^2 \right)^{1/2}$$

The result obtained by Equation (2.21) is the autocorrelation function on a day-by-day basis, referred to as daily autocorrelation function here. It is calculated after detrending (only for the case of the streamflow of the Yellow River at TG), log-transforming and deseasonalizing the raw series. The daily autocorrelations at different lags for the six daily streamflow processes are displayed in Figure 2.11 to 2.16.

Similarly, we can deseasonalize the 1/3-monthly and monthly streamflow series, and then calculate their autocorrelations at different lags for the six 1/3-monthly and six monthly streamflow processes, as shown in Figure 2.17 and 2.18.

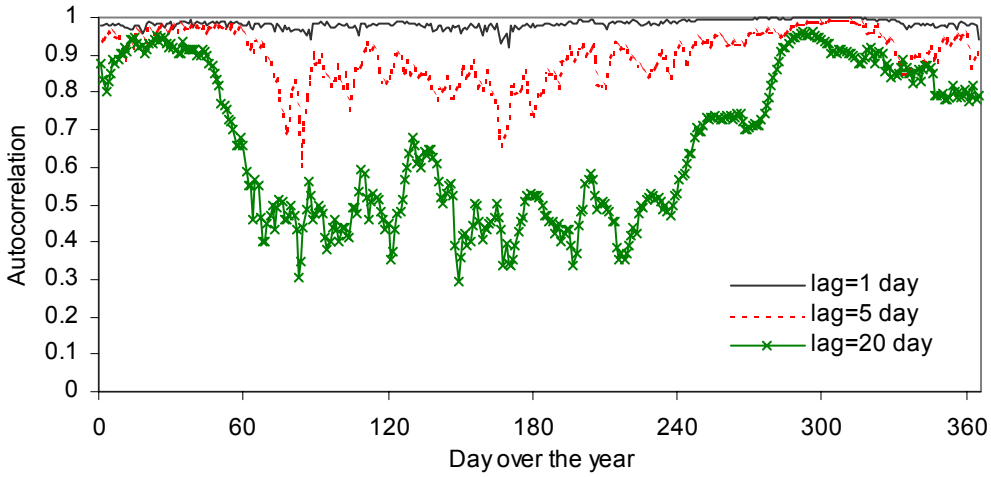


Figure 2.11 Daily autocorrelations at different lag days for daily flow series of Yellow River at TNH

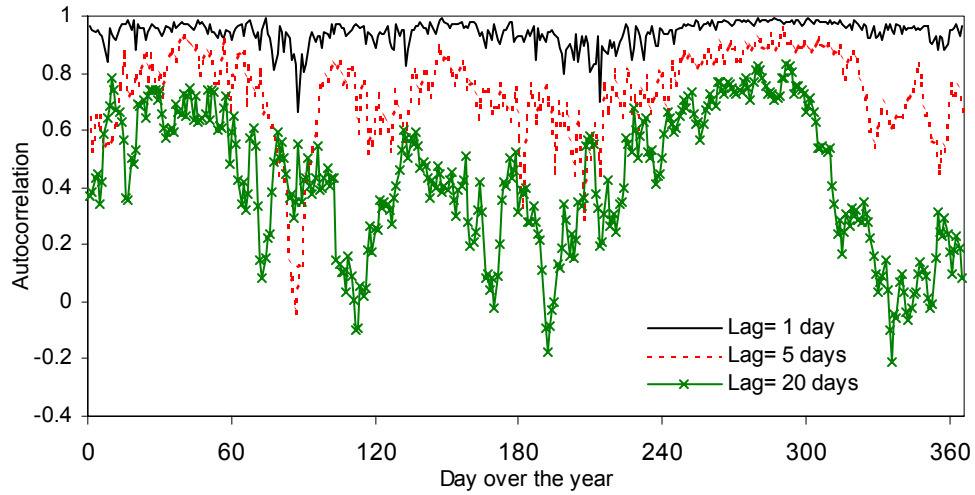


Figure 2.12 Daily autocorrelations at different lag days for daily flow series of Yellow River at TG

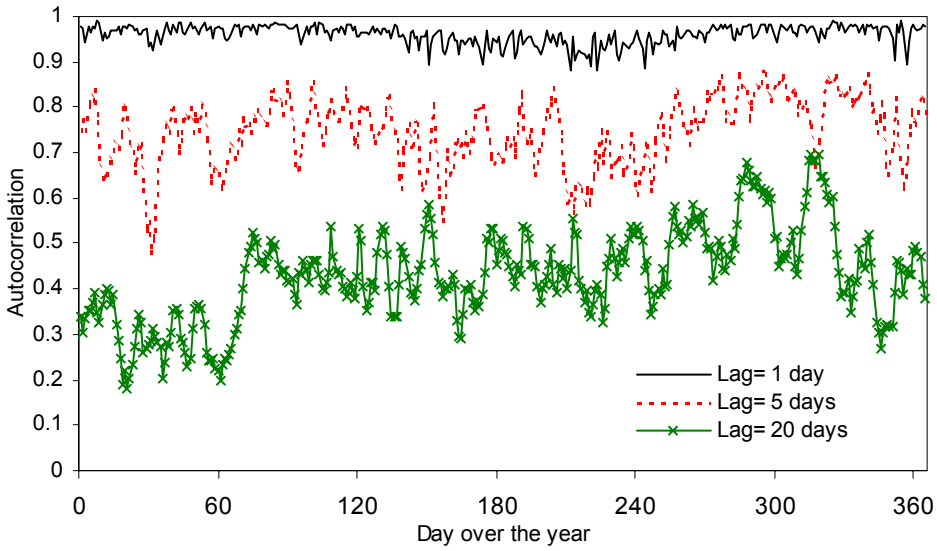


Figure 2.13 Daily autocorrelations at different lag days for daily flow series of the Danube

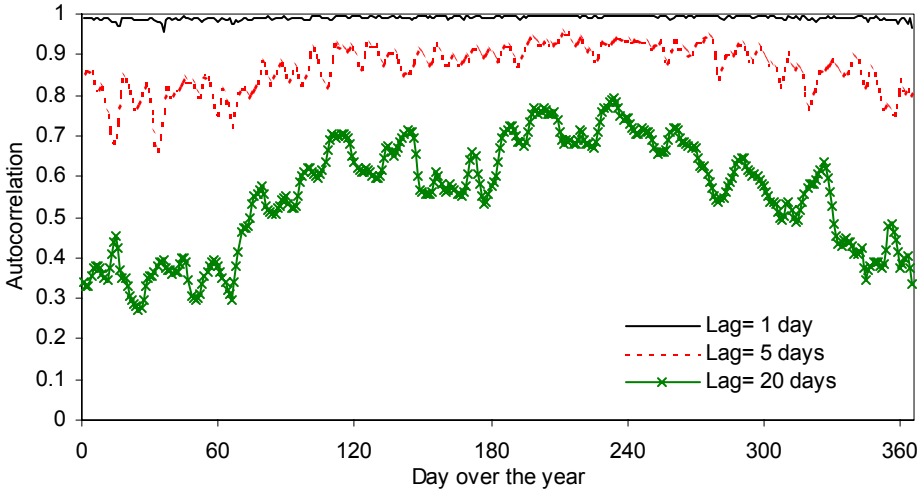


Figure 2.14 Daily autocorrelations at different lag days for daily flow series of the Rhine

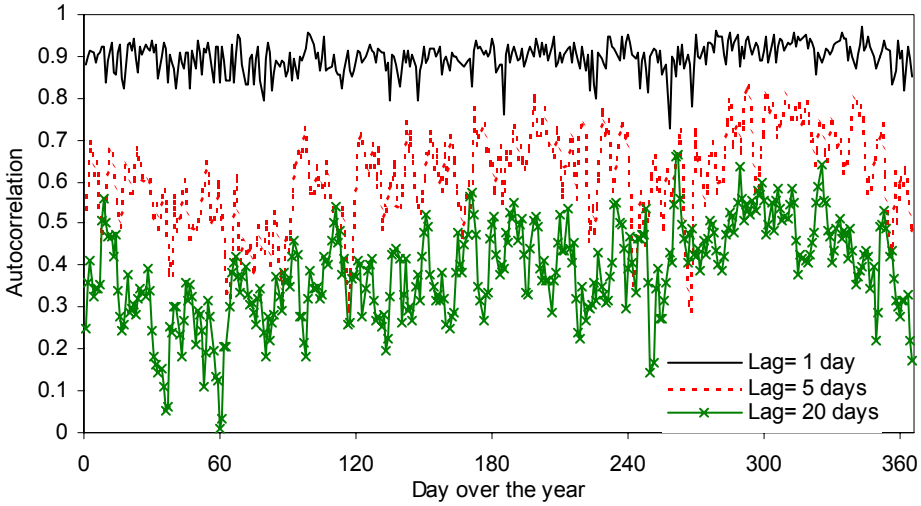


Figure 2.15 Daily autocorrelations at different lag days for daily flow series of the Ocmulgee

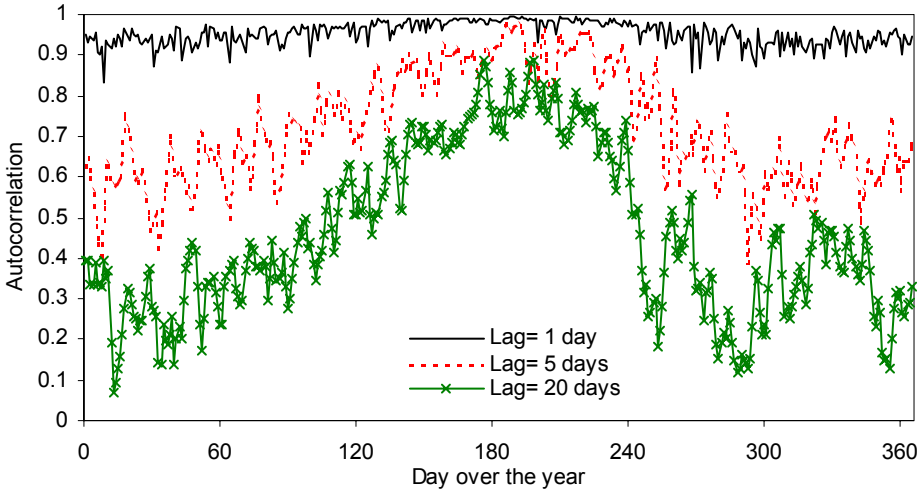


Figure 2.16 Daily autocorrelations at different lag days for daily flow series of the Umpqua

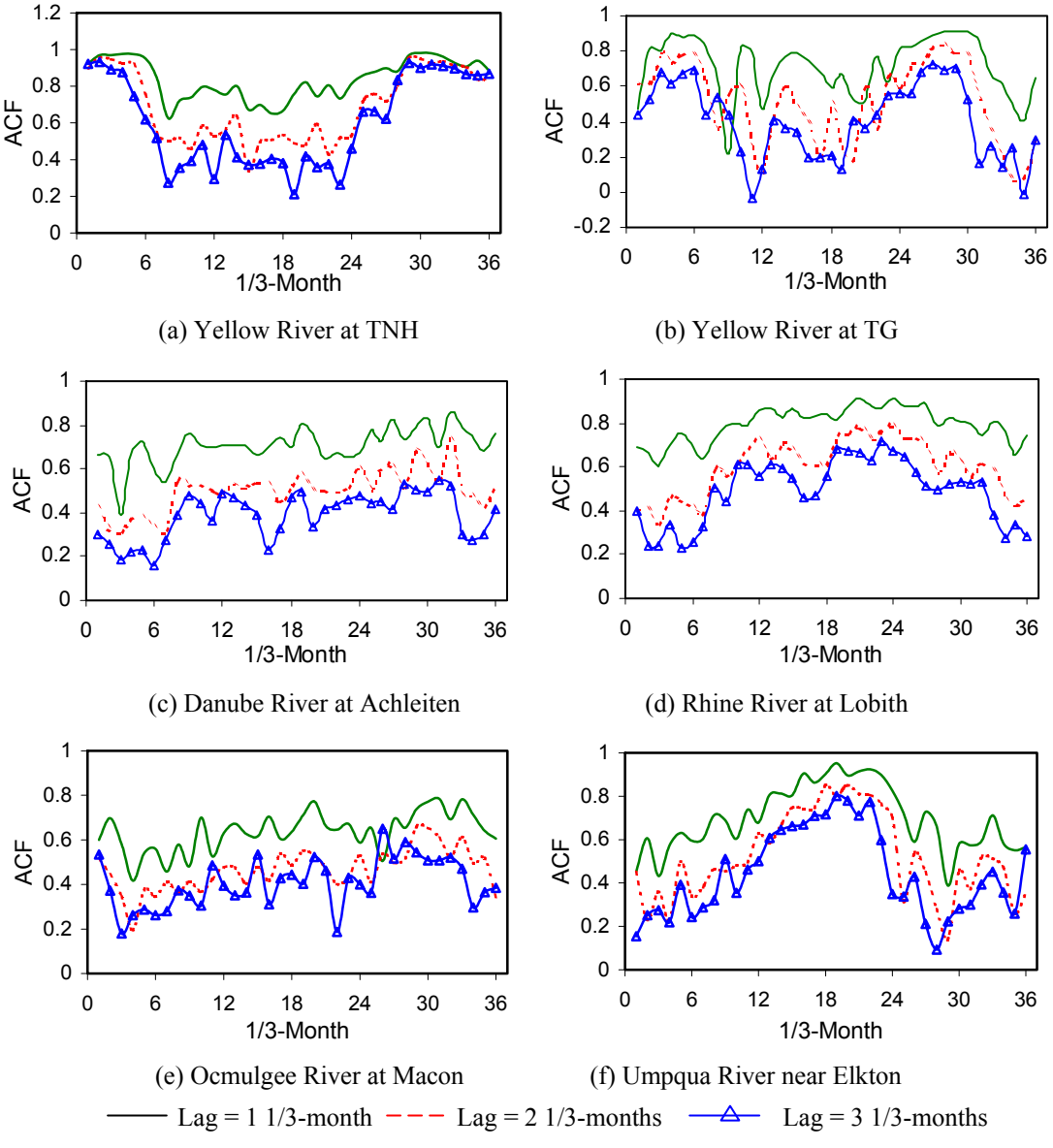


Figure 2.17 1/3-monthly autocorrelations at different lags for 1/3-monthly flow series

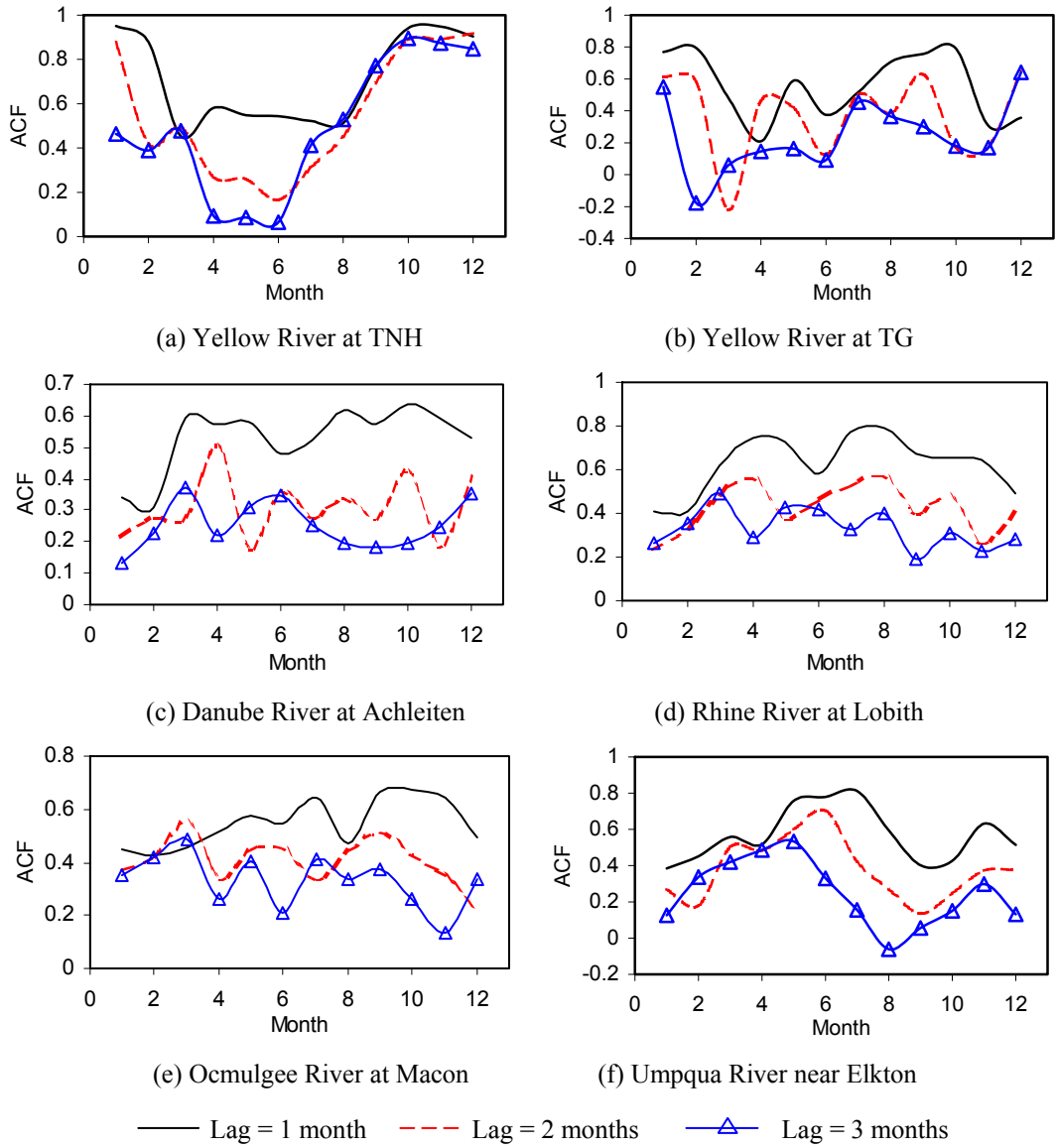


Figure 2.18 Monthly autocorrelations at different lags for monthly flow series

By a visual inspection of Figure 2.11 to 2.18, we see that:

(1) There are more or less seasonal variations in the autocorrelation structures of all the daily, 1/3-monthly and monthly streamflow processes. In general, the autocorrelation is high for low-flow seasons and low for high flow seasons. However, there are some exceptions. For example, the daily flows of the Yellow River at TG have lower autocorrelations in late November and December when discharges are lower than those in August to October. For the Danube, the autocorrelations in January and February are lower than those in June and July, although the flows are lower rather than those in June and July. In fact, the seasonal variation in the autocorrelation functions of streamflows processes has been observed by many researchers (e.g., Vecchia and Ballerini, 1991; Mcleod, 1994). With such kind of season dependence of the autocorrelation structure, the streamflow processes are not second-order stationary. Instead, they are periodic stationary (see the definition of the periodic stationarity in Appendix 2.2).

(2) Daily autocorrelations of the Yellow River at TNH are generally much higher than those at TG. In the period from the end of January to February and in November, the daily

autocorrelations at TNH are especially high, which can still be as high as 0.9 at a lag of 20 days. In March, June and July, daily autocorrelations at TNH are low because of large volume of snowmelt water and heavy rainfall respectively. Daily autocorrelations at TG are generally much lower because the streamflow process changes much more irregularly than that at TNH. The daily autocorrelations at TG are especially low in March because river ice-jam breakup and in July and August because of over-concentrated rainfall. In these two periods, the autocorrelations between contiguous days are very low, for instance, lower than 0.5 in the end of March and the beginning of April, and lower than 0.6 in the end of August.

2.5 Long-Memory Analysis

2.5.1 Introduction to Long-Memory

Long-memory, or long-range dependence, refers to a not negligible dependence between distant observations in a time series. Since the early work of Hurst (1951), it has been well recognized that many time series, in diverse fields of application, such as financial time series (e.g., Lo, 1991; Meade and Maier, 2003), meteorological time series (e.g., Haslett and Raftery, 1989; Bloomfield, 1992; Hussain and Elbergali, 1999) and internet traffic time series (see Karagiannis et al., 2004), etc., may exhibit the phenomenon of long-memory or long-range dependence. In the hydrology community, many studies have been carried out on the test for long-memory in streamflow processes. Montanari et al. (1997) applied fractionally integrated autoregressive moving average (ARFIMA) model to the monthly and daily inflows of Lake Maggiore, Italy. Rao and Bhattacharya (1999) explored some monthly and annual hydrologic time series, including average monthly streamflow, maximum monthly streamflow, average monthly temperature and monthly precipitation, at various stations in the mid-western United States. They stated that there is little evidence of long-term memory in monthly hydrologic series, and for annual series the evidence for lack of long-term memory is inconclusive. Montanari et al. (2000) introduced seasonal ARFIMA model and applied it to the Nile River monthly flows at Aswan to detect whether long-memory is present. The resulting model also indicates that nonseasonal long-memory is not present in the data. At approximately the same time, Ooms and Franses (2001) documented that monthly river flow data displays long-memory, in addition to pronounced seasonality based on simple time series plots and periodic sample autocorrelations.

Long-memory processes can be expressed either in the time domain or in the frequency domain. In the time domain, long-memory is characterized by a hyperbolically decaying autocorrelation function. In fact, it decays so slowly that the autocorrelations are not summable. For a stationary discrete long-memory time series process, its autocorrelation function $\rho(k)$ at lag k satisfies (Hosking, 1981).

$$\rho(k) \sim \frac{\Gamma(1-d)}{\Gamma(d)} k^{2d-1}, \text{ as } k \to \infty,$$

where, d is the long-memory parameter (or fractional differencing parameter), and $0 < |d| < 0.5$.

In frequency domain, long-memory manifests itself as an unbounded spectral density at zero frequency. For a stationary discrete long-memory time series process, its spectral density at zero frequency satisfies

$$f(\lambda) \sim C\lambda^{1-2H}, \text{ as } \lambda \to 0+,$$

for a positive, finite C. H is called the Hurst coefficient (or self-similarity parameter), as originally defined by Hurst (1951), and it represents the classical parameter characterizing long-memory. H is related to the fractional differencing parameter d with a relationship: $d = H - 0.5$.

A number of models have been proposed to describe the long-memory feature of time series. The Fractional Gaussian Noise model is the first model with long-range dependence introduced by Mandelbrot and Wallis (1969). Then Hosking (1981) and Granger and Joyeux (1980) proposed the fractional integrated autoregressive and moving average model, denoted by ARFIMA(p, d, q). When $-0.5 < d < 0.5$, the ARFIMA (p, d, q) process is stationary, and if $0 < d < 0.5$ the process presents long-memory behaviour.

Many methods are available for testing for the existence of long-memory and estimating the Hurst coefficient H or the fractional differencing parameter d. Many of them are well described in the monograph of Beran (1994). These techniques include graphical methods (e.g., classical R/S analysis; aggregated variance method etc.), parametric methods (e.g., Whittle maximum likelihood estimation method) and semiparametric method (e.g., GPH method and local whittle method). Heuristic methods are useful to test if a long-range dependence exists in the data and to find a first estimate of d or H, but they are generally not accurate and not robust. The parametric methods obtain consistent estimators of d or H via maximum likelihood estimation (MLE) of parametric long-memory models. They give more accurate estimate of d or H, but generally require knowledge of the true model which is in fact always unknown. Semiparametric methods, such as the GPH method (Geweke and Porter-Hudak, 1983), seek to estimate d under few prior assumptions concerning the spectral density of a time series and, in particular, without specifying a finite parameter model for the dth difference of the time series. In the present study, two statistic tests: Lo's modified R/S test which is a modified version of classical R/S analysis, and GPH test which is a semiparametric method will be used to test for the null hypothesis of no presence of long-memory. Besides, an approximate maximum likelihood estimation method is used to estimate the fractional differencing parameter d, but without testing for the significance level of the estimate.

In Section 2.5.2, we will use three heuristic methods, i.e., autocorrelation function analysis, classical R/S analysis, and the aggregated variance method to detect the existence of long-memory in the streamflow processes of the upper and middle Yellow River at TNH and TG (To save space, other streamflow processes are not analysed with heuristic methods). Then in the Section 2.5.3, two statistical test methods, i.e., Lo's modified R/S test (Lo, 1991) and the GPH test (Geweke and Porter-Hudak, 1983), to test for the existence of long-memory in the streamflow processes of all the five rivers, and the maximum likelihood estimates of the fractional differencing parameter d will be made as well. To verify the validity of these statistical test and estimation methods, some Monte Carlo simulation results will also be presented in Section 2.5.3.

2.5.2 Detecting Long-memory with Heuristic Methods

2.5.2.1 *Autocorrelation function analysis*

In the presence of long-memory, the autocorrelation function (ACF) of a time series decreases to 0 at a much slower rate than the exponential rate implied by an short-memory ARMA model. So we can compare the sample ACF of the observed time series under investigation with the theoretical ACF (McLeod, 1975) of the ARMA model fitted to the time series. If the sample ACF of the observed series decays much slower than the ACF of the fitted ARMA model, then it probably indicates the existence of long-memory.

First, we select the best fitting AR models for the streamflow series using the Akaike Information Criterion (AIC) (Akaike, 1973), which turns out to be an AR(38), AR(9) and AR(4) model for the daily, 1/3-monthly, and monthly streamflow series at TNH, and an AR(9), AR(5) and AR(15) model for the daily, 1/3-monthly, and monthly streamflow series at TG. The high autoregressive order for monthly series at TG arises from the remaining seasonality that has not been fully removed with the deseasonalization procedure. The sample ACF of the streamflow series and the theoretical ACF of the fitted models from lag 1 to lag 100 are plotted in Figure 2.19 and 2.20.

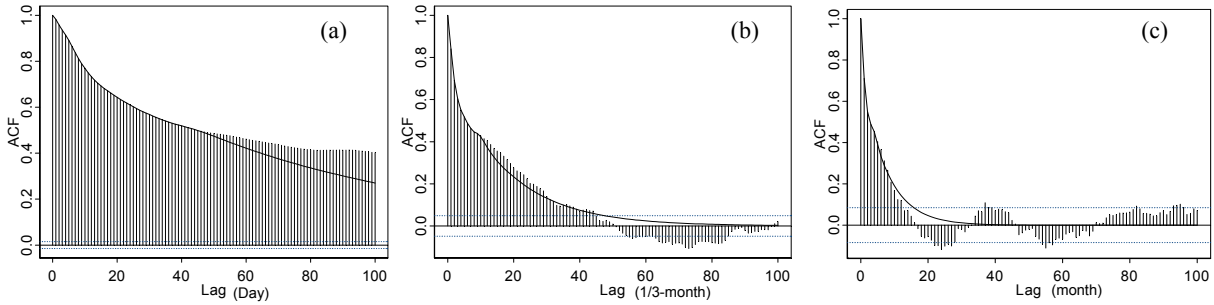


Figure 2.19 Sample ACF (vertical lines) and the theoretical ACF (curve line) of fitted AR models for (a) daily, (b) 1/3-monthly and (c) monthly streamflow at TNH

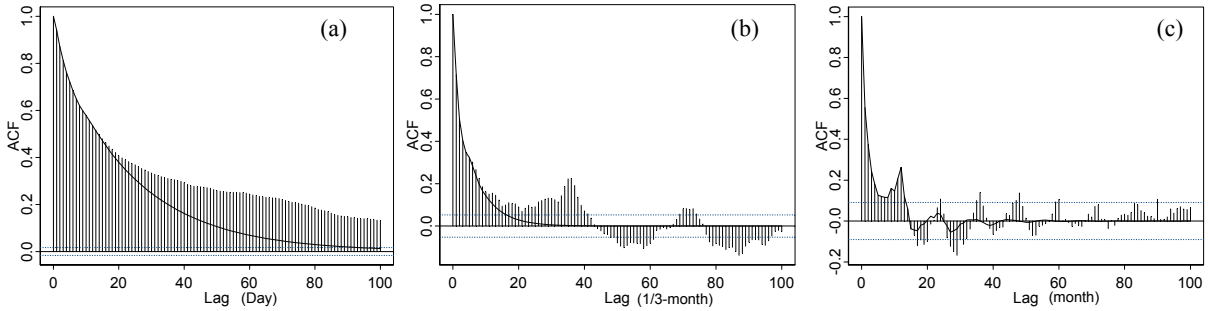


Figure 2.20 Sample ACF (vertical lines) and the theoretical ACF (curve line) of fitted AR models for (a) daily, (b) 1/3-monthly and (c) monthly streamflow at TG

Comparing the theoretical ACF of the fitted AR models with the sample ACF of the observed streamflow series, we can find that:

(1) The daily streamflow process is highly persistent and the autocorrelation remains significant from zero at lag 100. The theoretical autocorrelation closely matches the sample autocorrelation at short lags. However, for large lags, the sample ACF decays much slower than the theoretical ACF.

(2) The 1/3-monthly and monthly streamflow processes are much less persistent. For both 1/3-monthly flow series at TNH and TG, the sample autocorrelations are slightly larger than the theoretical autocorrelations for large lags. But for the monthly flow series, the sample ACF is basically at the same level as the theoretical ACF.

2.5.2.2 *Classical R/S analysis*

The *R*/*S* statistic, or the "rescaled adjusted range" statistic, is the adjusted range of partial sums of deviations of a times series from its mean, rescaled by its standard deviation. It was developed by Hurst (1951) in his studies of river discharges, and suggested by Mandelbrot and Wallis (1969c) using the R/S statistic to detect long-range dependence. Consider a time series $\{x_t\}$, t = 1, 2, …, N, and define the jth partial sum as $Y_j = \sum_{i=1}^{j} x_i$, j = 1, 2, …, N. Suppose to calculate the storage range of a reservoir between time t and $t+k$, and assume that: (1) the storage at time t and $t+k$ is the same; (2) the outflow during time t and $t+k$ is the same; and (3) there is no any loss of storage. Then the rescaled adjusted range, i.e., *R/S* statistic, is defined as (Beran, 1994):

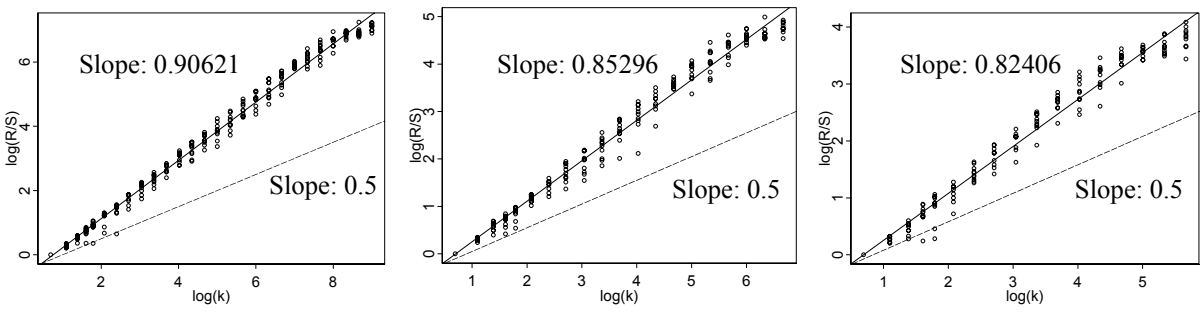
$$R/S_{(t,k)} = \frac{1}{S_{(t,k)}}\left\{\max_{0\le i\le k}\left[Y_{t+i} - Y_t - \frac{i}{k}(Y_{t+k} - Y_t)\right] - \min_{0\le i\le k}\left[Y_{t+i} - Y_t - \frac{i}{k}(Y_{t+k} - Y_t)\right]\right\}, \qquad (2.22)$$

where, $S_{(t,k)} = \sqrt{k^{-1}\sum_{j=t+1}^{t+k}(x_j - \overline{x}_{t,k})^2}$, and $\overline{x}_{t,k} = k^{-1}\sum_{j=t+1}^{t+k} x_j$.

The *R/S* statistic varies with the time span k. Hurst (1951) found that the *R/S* statistic for many geophysical records is well described by the following empirical relation: $E[R/S] \sim c_1 k^H$, as $k \to \infty$, with typical values of H (the Hurst coefficient) in the interval (0.5, 1.0), and c_1 a finite positive constant that does not depend on k.

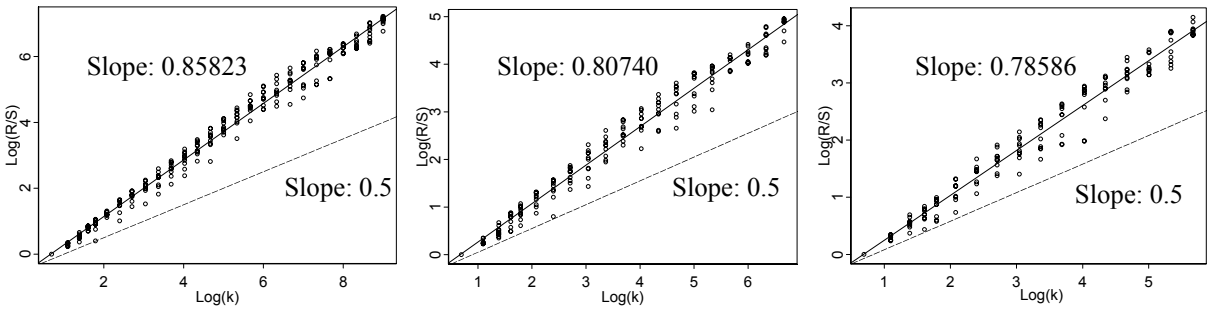
The classical *R*/*S* analysis is based on a heuristic graphical approach. Compute the *R*/*S*-statistic in Equation (2.22) at many different lags k and for a number of different points, and plots the resulting estimates versus the lags on log-log scale. The logarithm of k should scatter along a straight line having a slope equal to H. The value of H can be estimated by a simple least-squares fit. An H value equal to 0.5 means absence of long-memory. The higher the H is, the higher the intensity of long-memory.

The log-log plots of *R/S* versus different lags k for streamflow processes at both TNH and T are displayed in Figure 2.21 and 2.22. The slopes of the fitted lines are the estimates of values of H.



(a) daily (b) 1/3-monthly (c) monthly

Figure 2.21 *R/S* plot of (a) daily, (b) 1/3-monthly and (c) monthly flow series at TNH



(a) daily (b) 1/3-monthly (c) monthly

Figure 2.22 *R/S* plot of (a) daily, (b) 1/3-monthly and (c) monthly flow series at TG

According to the *R*/*S* statistics obtained with the graphical approach, all the streamflow series have values of *H* larger than 0.5, indicating the presence of long-memory in all these streamflow series. The *H* values, which indicate the intensity of long-memory, decrease with the increase of timescales. Furthermore, at each timescale, the intensity of long-memory of the streamflow process at TNH is stronger than that at TG.

To check the effectiveness of the *R*/*S* analysis for detecting long-memory, we generate ten simulations of an AR(1) model, ten simulations of an ARFIMA(0,*d*,0) model, and ten simulations of an ARFIMA(1,*d*,0) model. The AR(1) model is of the form $(1-\phi B)x_t = \varepsilon_t$ with ϕ =0.9, the ARFIMA(0,d,0) of form $(1-B)^d x_t = \varepsilon_t$ with d = 0.3, and the ARFIMA(1,d,0) of form $(1-\phi B)(1-B)^d x_t = \varepsilon_t$ with ϕ =0.9 and d = 0.3, where $\{\varepsilon_t\}$ are i.i.d standard normal, B is the backshift operator, i.e., $Bx_t = x_{t-1}$. Each of them has a size of 3000 points. The AR series and the ARFIMA series are produced by the *arima.sim* and *arima.fracdiff.sim* function built in S-Plus version 6 (Insightful Corporation, 2001). The estimated values of *H* are listed in Table 2.9.

The simulation results show that, for a pure fractionally integrated process ARFIMA (0, *d*, 0), the estimate of *H* is very close to its true value 0.8 (i.e., *d* + 0.5). But when a process is a mixture of short memory and long-memory, as the ARFIMA(1, *d*, 0) process, then the estimates of *H* are biased upwardly. Furthermore, classical *R*/*S* analysis gives estimated *H* values (= *d* + 0.5) higher than 0.5 even for short memory AR (1) processes, which indicates its sensitivity to the presence of explicit short-range dependence.

Table 2.9 Estimated *H* values with classical *R*/*S* analysis for simulated series

Simulation	AR(1)	ARFIMA(0,*d*,0)	ARFIMA(1,*d*,0)
1	0.83789	0.75434	0.91157
2	0.79296	0.76044	0.89271
3	0.78578	0.73048	0.90742
4	0.78821	0.77499	0.87063
5	0.82238	0.75269	0.88660
6	0.82636	0.73367	0.87649
7	0.77678	0.81083	0.89122
8	0.83730	0.77748	0.91854
9	0.77904	0.76316	0.89593
10	0.83119	0.77612	0.90586
Average	0.80779	0.76342	0.89570

2.5.2.3 Aggregated *Variance Method*

For independent random variables x_1, …, x_N, the variance of sample mean is equal to $\mathrm{var}(\overline{x}) = \sigma^2 N^{-1}$. But in the presence of long-memory, Beran (1994) proved that the variance of the sample mean could be expressed by $\mathrm{var}(\overline{x}) \approx cN^{2H-2}$, where c>0 and H is the Hurst coefficient. Correspondingly, Beran (1994) suggested the following method for estimating the Hurst coefficient H.

(1) Take a sufficient number (say m) of subseries of length k ($2 \le k \le N/2$), calculate the sample means $\overline{x}_1(k), \overline{x}_2(k), ..., \overline{x}_m(k)$ and the overall mean $\overline{x}(k) = m^{-1}\sum_{j=1}^{m} \overline{x}_j(k)$;

(2) For each k, calculate the sample variance $s^2(k)$ of the m sample means:

$$s^2(k) = (m-1)^{-1}\sum_{j=1}^{m} (\overline{x}_j(k) - \overline{x}(k))^2 \tag{2.23}$$

(3) Plot log $s^2(k)$ against log$k$. For large values of k, the points in this plot are expected to be scattered around a straight line with negative slope $2H - 2$. The slope is steeper (more negative) for short-memory processes. In the case of independence, the ultimate slope is -1.

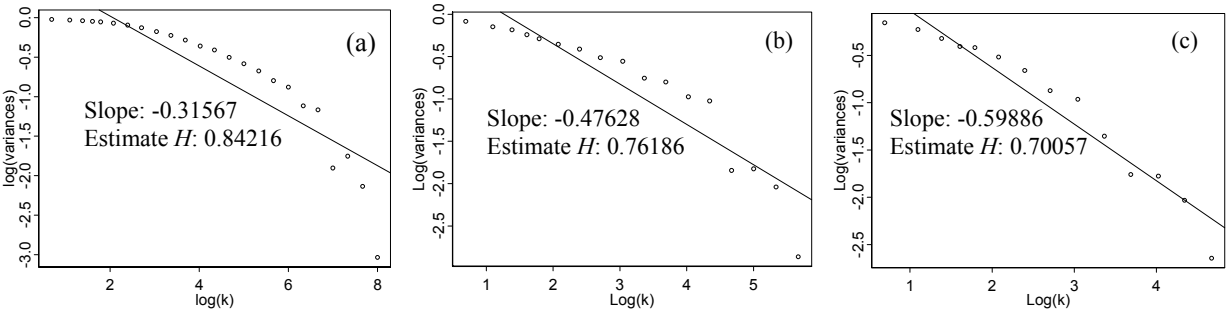


Figure 2.23 Variance plot of (a) daily, (b) 1/3-monthly and (c) monthly flow series at TNH

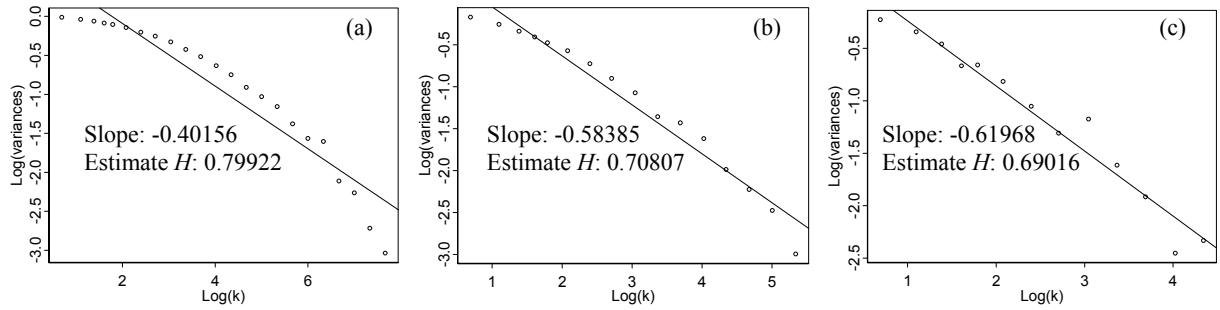


Figure 2.24 Variance plot of (a) daily, (b) 1/3-monthly and (c) monthly flow series at TG

Comparing the variance plot for the streamflow processes at TNH and TG, displayed in Figure 2.23 and 2.24, we can find that the slopes of the fitted lines get more negative as the timescale increases (from day to month) for the streamflow processes at both TNH and TG, which indicates that, from the view of time series themselves, the H values, namely the intensity of long-memory, decreases with the increase of timescales. Furthermore, at each timescale, the intensity of long-memory in streamflow process at TNH is stronger than that at TG.

Similarly to the assessment of the effectiveness of classical *R*/*S* analysis, we assess the effectiveness of variance analysis for detecting the long-memory by estimating the *H* values for the generated simulations of the AR(1) model, ARFIMA(0,d,0) model and ARFIMA(1,d,0) model. The estimated *H* values are listed in Table 2.10. The results show that, variance analysis is also sensitive to the presence of explicit short-range dependence, and generally it gives smaller estimate than the classical *R*/*S* analysis.

Table 2.10 Estimated *H* values with variance analysis for simulated series

Simulation	AR(1)	ARFIMA(0,d,0)	ARFIMA(1,d,0)
1	0.69158	0.78782	0.83284
2	0.64412	0.71964	0.77195
3	0.66903	0.67128	0.84894
4	0.64130	0.80683	0.79878
5	0.65846	0.78597	0.87033
6	0.71512	0.71407	0.87689
7	0.68218	0.80170	0.80999
8	0.69148	0.72700	0.80335
9	0.59842	0.64447	0.82691
10	0.71557	0.72315	0.78931
Average	0.67073	0.73819	0.82293

Because both the *R*/*S* analysis method and variance plot method are sensitive to the presence of explicit short-range dependence, whereas the ACF analysis only gives us a heuristic suggestion without quantitative estimations, we need some formal statistical techniques for detecting long-memory in the streamflow series.

2.5.3 Detecting Long-memory with Statistical Test Method and MLE Method

In this section, we will detect the presence of long-memory in the streamflow processes of streamflow processes with two statistical test techniques, i.e., the Lo's modified *R*/*S* test (Lo, 1991), and the GPH test (Geweke and Porter-Hudak, 1983). In addition we will try to detect the presence of long-memory by estimating the fractional differencing parameter *d*.

2.5.3.1 *Lo's modified R/S analysis*

As having been shown in Section 2.5.2, the classical *R*/*S* analysis is sensitive to the presence of explicit short-range dependence structures, and it lacks of a distribution theory for the underlying statistic. To overcome these shortcomings, Lo (1991) proposed a modified *R*/*S* statistic that is obtained by replacing the denominator $S_{(t,\,k)}$ in Equation (2.22), i.e., the sample standard deviation, by a modified standard deviation S_q which takes into account the autocovariances of the first q lags, so as to discount the influence of the short-range dependence structure that might be present in the data. Instead of considering multiple lags as in Equation (2.22), only focus on lag $k = N$. The S_q is defined as

$$S_q = \left(\frac{1}{N} \sum_{j=1}^{N} (x_j - \overline{x}_N)^2 + \frac{2}{N} \sum_{j=1}^{q} \omega_j(q) \left[\sum_{i=j+1}^{N} (x_i - \overline{x}_N)(x_{i-j} - \overline{x}_N) \right] \right)^{1/2}, \tag{2.24}$$

where, $\overline{x}_N$ denotes the sample mean of the time series, and the weights $\omega_j(q)$ are given by $w_j(q) = 1 - j/(q+1)$, $q < N$. Then the Lo's modified *R*/*S* statistic is defined by

$$Q_{N,q} = \frac{1}{S_q}\left\{\max_{0\le i\le N}\sum_{j=1}^{i}(x_j - \bar{x}_N) - \min_{0\le i\le N}\sum_{j=1}^{i}(x_j - \bar{x}_N)\right\} \tag{2.25}$$

If a series has no long-range dependence, Lo (1991) showed that given the right choice of q, the distribution of $N^{-1/2}Q_{N,q}$ is asymptotic to that of

$$W = \max_{0\le r\le 1} V(r) - \min_{0\le t\le 1} V(r),$$

where, V is a standard Brownian bridge, that is, $V(r) = B(r) - rB(1)$, where B denotes standard Brownian motion. Since the distribution of the random variable W is known as

$$P(W \le x) = 1 + 2\sum_{j=1}^{\infty}(1 - 4x^2 j^2)e^{-2x^2 j^2}, \tag{2.26}$$

Lo gave the critical values of x for hypothesis testing at sixteen significance levels using Equation (2.26), which can be used for testing the null hypothesis H_0 that there is only short-term memory in a time series at a significance level α.

2.5.3.2 *GPH test*

Geweke and Porter-Hudak (1983) proposed a semi-nonparametric approach to testing for long-memory. Given a fractionally integrated process $\{x_t\}$, its spectral density is given by:

$$f(\omega) = \left[2\sin(\omega/2)\right]^{-2d} f_u(\omega),$$

where ω is the Fourier frequency, $f_u(\omega)$ is the spectral density corresponding to u_t, and u_t is a stationary short memory disturbance with zero mean. Consider the set of harmonic frequencies $\omega_j = (2\pi j/n), j = 0, 1, \ldots, n/2$, where n is the sample size. By taking the logarithm of the spectral density $f(\omega)$ we have

$$\ln f(\omega_j) = \ln f_u(\omega_j) - d\ln\left[4\sin^2\left(\omega_j/2\right)\right],$$

which may be written in the alternative form

$$\ln f(\omega_j) = \ln f_u(0) - d\ln\left[4\sin^2(\omega_j/2)\right] + \ln\left[f_u(\omega_j)/f_u(0)\right]. \tag{2.27}$$

The fractional difference parameter d can be estimated by the regression equations constructed from Equation (2.27). Geweke and Porter-Hudak (1983) showed that using a periodogram estimate of $f(\omega_j)$, if the number of frequencies used in the regression Equation (2.27) is a function $g(n)$ (a positive integer) of the sample size n where $g(n) = n^{\alpha}$ with $0 < \alpha < 1$, the least squares estimate $\hat{d}$ using the above regression is asymptotically normally distributed in large samples:

$$\hat{d} \sim N\left(d, \frac{\pi^2}{6\sum_{j=1}^{g(n)}(U_j - \bar{U})^2}\right),$$

where, $U_j = \ln[4\sin^2(\omega_j/2)]$ and $\bar{U}$ is the sample mean of U_j, $j = 1, \cdots, g(n)$. Under the null hypothesis of no long-memory ($d = 0$), the t-statistic

$$t_{d=0} = \hat{d}\cdot\left(\frac{\pi^2}{6\sum_{j=1}^{g(n)}(U_j - \bar{U})^2}\right)^{-1/2}$$

has a limiting standard normal distribution.

2.5.3.3 *Maximum likelihood estimation of fractional differencing parameter d*

Let the observation $X = (x_1, \ldots, x_n)^t$ be the ARFIMA(p,d,q) process defined by

$$\phi(B)(1-B)^d(x_t - \mu) = \theta(B)\varepsilon_t, \tag{2.28}$$

where B is the backshift operator; $\phi(B) = 1 - \phi_1 B - \ldots - \phi_p B^P$ and $\theta(B) = 1 - \theta_1 B - \ldots - \theta_q B^q$ represent the ordinary autoregressive and moving average components; ε_t is a white noise process with zero mean and variance σ^2.

The Gaussian log-likelihood of X for the process (2.28) is given by

$$\log L(\mu, \eta, \sigma^2) = -\frac{n}{2}\log(2\pi) - \frac{1}{2}\log|\Sigma| - \frac{1}{2}X^t\Sigma^{-1}X$$

where $\eta = (\phi_1, \ldots, \phi_p; d; \theta_1, \ldots, \theta_q)$ is the parameter vector; Σ denotes the $n \times n$ covariance matrix of X depending on η and σ^2, $|\Sigma|$ denote the determinant of Σ. The maximum likelihood estimators $\hat{\eta}$ and $\hat{\sigma}^2$ can be found by maximizing $\log L(\eta, \sigma^2)$ with respect to η and σ^2.

In this study, the maximum likelihood estimation method implemented in S-Plus version 6 (referred to as *S-MLE*) is used to estimate the fractional differencing parameter *d*. *S-MLE* is implemented based on the approximate Gaussian maximum likelihood algorithm of Haslett and Raftery (1989). If the estimated *d* is significantly greater than zero, we consider it an evidence of the presence of long-memory.

2.5.3.4 *Monte Carlo simulation results for long-memory detection*

Before applying the Lo's test, GPH test and S-MLE method to the streamflow processes, we perform an extensive Monte Carlo investigation in order to find out how reliable the Lo's test, the GPH test and the *S-MLE* are with AR and ARFIMA processes. We consider five AR(1) and six ARFIMA(1,d,0) processes. All AR(1) models are of the form $(1-\phi B)x_t = \varepsilon_t$, and all ARFIMA(1,d,0) of form $(1-\phi B)(1-B)^d x_t = \varepsilon_t$, where $\{\varepsilon_t\}$ are i.i.d standard normal, B is the backshift operator. For the AR models, large autoregressive coefficients, i.e., $\phi = 0.5, 0.8, 0.9, 0.95, 0.99$, because these are the cases commonly seen in streamflow processes. For the ARFIMA models, $\phi = 0, 0.5, 0.9$ and $d = 0.3, 0.45$. We generate 500 simulated realizations of with size 500, 1000, 3000, 10000 and 20000, respectively, for each model. The AR series and the ARFIMA series are produced by the *arima.sim* and *arima.fracdiff.sim* function built in S-Plus version 6 (Insightful Corporation, 2001).

For Lo's modified *R*/*S* test, the right choice of *q* in Lo's method is essential. It must be chosen with some consideration of the data at hand. Some simulation studies have shown (Lo, 1991; Teverovsky et al., 1999) that, for any of these series, the probability of accepting the null hypothesis varied significantly with *q*. In general, for the larger sample lengths, the larger the *q*, the less likely was the null hypothesis to be rejected. One appealing data-driven formula (Andrew, 1991) for choosing *q* based on the assumption that the true model is an AR(1) model is given by

$$q = \left[\left(\frac{3n}{2} \right)^{1/3} \left(\frac{2\hat{\rho}}{1-\hat{\rho}^2} \right)^{2/3} \right], \tag{2.29}$$

where [•] denotes the greatest integer function, n is the length of the data, $\hat{\rho}$ is the estimated first-order autocorrelation coefficient. However, our simulation for AR processes and ARFIMA processes with different intensity of dependence indicate that this data-driven formula is too conservative in rejecting the null hypothesis of no long-memory, especially for cases where autocorrelations at lag 1 are high. After a trial-and-error procedure, we use the following modified formula to choose the lag q:

$$q = \left[\left(\frac{n}{10} \right)^{1/4} \left(\frac{2\hat{\rho}}{1-\hat{\rho}^2} \right)^{2/3} \right]. \tag{2.30}$$

where $\hat{\rho}$ is the autoregressive function at lag 1, i.e., ACF(1). This modified formula is a trade-off between lowering the probability of wrongly rejecting the null hypothesis of no long-memory for AR processes, and reserving the power of correctly rejecting the null hypothesis for ARFIMA processes. The null hypothesis of no long-memory is rejected at a 5% significance level if $Q_{N,q}$ is not contained in the interval [0.809, 1.862] (Lo, 1991).

Similarly to the case with Lo's test, for the GPH test, there is a choice of the number of frequencies $g(n)$ used in the regression Equation (2.27). This choice entails a bias-variance trade-off. For a given sample size, as $g(n)$ is increased from 1, the variance of the d estimate decreases, but this decrease is typically offset by the increase in bias due to non-constancy of $f_u(\omega)$. Geweke and Porter-Hudak (1983) found that choosing $g(n) = n^{0.5}$ gave good results in simulation. We adopt such a criterion in the Monte Carlo simulation study. The periodogram used for calculating GPH test statistic is smoothed with a modified Daniell smoother of length 5.The null hypothesis of no long-memory ($d = 0$) is rejected at a 5% significance level if t-statistic is not contained in the interval [-1,960, 1.960].

When estimating the parameter d with the *S-MLE* method, we assume that the order p of the AR component for each simulated ARFIMA process is unknown before hand. Instead, we estimate the order p of the AR component by using the AIC criterion (Akaike, 1973).

The results of detecting long-memory in simulated AR and ARFIMA processes of sizes ranging from 500 to 20000 with Lo's test, GPH test and the *S-MLE* estimates of d are reported in Table 2.11. For Lo's test, we list the average values of the lags chosen with the data-driven formula (2.30), their standard deviations (denoted as SD of lag), and the number of acceptance of the null hypothesis for 500 simulations. For GPH test, we list the average of the estimates of d, their standard deviations (denoted as SD of lag), and the number of acceptance of the null hypothesis for 500 simulations. For the S-MLE method, we give the averages and standard deviations (SD) of the estimates of d. According to the results with simulated AR and ARFIMA processes, shown in Table 2.11, we have the following findings:

(1) For AR processes, when the autocorrelation is less than 0.9, both the Lo's *R/S* test and the GPH test work well, and the GPH test has a better performance. But when the autoregressive coefficient is higher than 0.9, the probability of committing Type I error with the GPH test increase very fast, and the GPH test gets useless for the cases when ϕ is as high as 0.95 or above, even for the size of 20000 points. In contrast, the probability of committing Type I error with the Lo's *R/S* test still considerably lower even for AR

processes with a ϕ of as high as 0.99. But it seems that the lag chosen with formula (8) tends to be too small for series of big size, whereas a little bit too large for series of small size for AR processes with large values of ϕ.

Table 2.11 Long-memory detection results for simulated AR and ARFIMA series

Model	Data size	Lo's R/S test			GPH test			S-MLE	
		average lag	SD of lag	accepted	average *d*	SD of *d*	accepted	average *d*	SD of *d*
AR(1) ar = .5	500	2.8	0.5	464	-0.0167	0.1302	495	0.0149	0.0350
	1000	3.2	0.4	454	-0.0123	0.1141	490	0.0189	0.0325
	3000	4.6	0.5	468	-0.0124	0.0772	490	0.0136	0.0220
	10000	6.1	0.2	455	-0.0119	0.0607	490	0.0093	0.0132
	20000	7.8	0.4	469	-0.0078	0.0479	488	0.0057	0.0100
AR(1) ar = .8	500	6.7	0.8	428	0.1220	0.1388	470	0.0269	0.0669
	1000	8.0	0.7	442	0.0637	0.1110	489	0.0209	0.0419
	3000	10.8	0.5	441	0.0163	0.0827	490	0.0199	0.0322
	10000	14.7	0.5	441	-0.0016	0.0605	490	0.0114	0.0207
	20000	17.6	0.5	454	-0.0036	0.0511	483	0.0079	0.0149
AR(1) ar = .9	500	11.3	1.6	431	0.3252	0.1342	268	0.0290	0.0566
	1000	13.5	1.4	408	0.2189	0.1135	326	0.0296	0.0632
	3000	18.1	1.1	414	0.0957	0.0851	436	0.0240	0.0488
	10000	24.6	0.8	441	0.0273	0.0600	483	0.0132	0.0236
	20000	29.4	0.7	457	0.0107	0.0500	489	0.0081	0.0150
AR(1) ar = .95	500	18.7	3.6	451	0.5739	0.1395	24	0.0302	0.0497
	1000	22.4	3.1	429	0.4488	0.1154	34	0.0390	0.0801
	3000	29.6	2.4	426	0.2594	0.0800	91	0.0270	0.0535
	10000	40.3	1.8	416	0.1201	0.0601	300	0.0117	0.0284
	20000	47.9	1.6	416	0.0665	0.0475	409	0.0065	0.0160
AR(1) ar = .99	500	52.9	20.3	494	0.9122	0.1617	0	0.0482	0.0674
	1000	65.3	19.3	484	0.8530	0.1226	0	0.0431	0.0780
	3000	86.8	14.7	399	0.7297	0.0826	0	0.0231	0.0442
	10000	119.7	11.9	389	0.5555	0.0583	0	0.0093	0.0211
	20000	142.4	9.5	380	0.4478	0.0477	0	0.0068	0.0148
ARFIMA d=0.3	500	2.2	0.5	129	0.2587	0.1360	353	0.2144	0.1100
	1000	2.8	0.5	61	0.2749	0.1157	228	0.2571	0.0829
	3000	3.8	0.5	15	0.2821	0.0826	68	0.2786	0.0646
	10000	5.2	0.4	0	0.2884	0.0572	2	0.3043	0.0201
	20000	6.3	0.5	0	0.2900	0.0470	0	0.3072	0.0162
ARFIMA ar=0.5 d=0.3	500	7.1	1.4	255	0.2729	0.1402	333	0.1728	0.1346
	1000	8.6	1.3	139	0.2783	0.1130	233	0.2126	0.1165
	3000	11.4	1.2	63	0.2878	0.0919	83	0.2849	0.0675
	10000	15.6	1.0	8	0.2934	0.0604	4	0.3049	0.0363
	20000	18.6	0.9	5	0.2955	0.0493	0	0.3102	0.0202
ARFIMA ar=0.9 d=0.3	500	41.1	12.2	493	0.6375	0.1513	16	0.1683	0.1451
	1000	49.4	11.6	478	0.5213	0.1123	6	0.2035	0.1333
	3000	65.4	11.2	345	0.3964	0.0881	5	0.2397	0.1243
	10000	89.4	9.2	155	0.3316	0.0627	2	0.3103	0.0678
	20000	106.6	8.3	78	0.3145	0.0512	0	0.3281	0.0501
ARFIMA d=0.45	500	7.0	4.0	130	0.4077	0.1506	157	0.3092	0.1572
	1000	8.5	4.4	56	0.4274	0.1237	53	0.3616	0.1309
	3000	11.2	5.2	11	0.4371	0.0873	0	0.4238	0.0620
	10000	15.4	6.0	0	0.4373	0.0613	0	0.4589	0.0173
	20000	18.6	7.0	0	0.4371	0.0489	0	0.4676	0.0164
ARFIMA ar=0.5 d=0.45	500	19.1	10.1	346	0.4331	0.1515	133	0.2355	0.1628
	1000	22.9	10.6	204	0.4385	0.1164	33	0.3328	0.1311
	3000	31.0	12.2	66	0.4404	0.0893	3	0.4226	0.0668
	10000	42.4	14.6	11	0.4429	0.0635	0	0.4608	0.0228
	20000	50.2	16.2	4	0.4459	0.0507	0	0.4718	0.0170
ARFIMA ar=0.9	500	135.0	78.5	493	0.7956	0.1394	2	0.1306	0.1757
	1000	163.4	90.2	495	0.6733	0.1172	1	0.1712	0.1828

d=0.45	3000	222.9	116.2	472	0.5539	0.0878	0	0.3128	0.1665
	10000	299.5	138.7	273	0.4856	0.0599	0	0.4464	0.0577
	20000	361.8	158.0	140	0.4666	0.0491	0	0.4748	0.0226

Note: The Lo's R/S test and the GPH test are based on 500 replications. The S-MLE estimate of *d* are based on 100 replications.

(2) For ARFIMA processes, the GPH technique yields downwardly biased estimates of *d* when an AR term of low autoregressive coefficient value (e.g., ≤ 0.5) is present, whereas yields upwardly biased estimates of *d* when an AR term of high autoregressive coefficient value (e.g., = 0.9) is present. This seems to be in contradiction with the results of Sowell (1992), who showed that, when the sample length is small, the GPH technique yields upwardly biased estimates of *d* when AR and MA terms are present. On the other hand, the power of GPH test increases with the increase of data size, the intensity of long-memory, and autocorrelations of their AR component. For cases where the data size is over 10000, the percentage of committing Type II error, i.e., false acceptance of the null hypothesis of no long-memory, by GPH test is close to zero. In contrast, the Lo's test only performs slightly better than the GPH test when the intensity of long-memory is not strong and the value of ϕ in the AR component is low, but for the cases of strong intensity of long-memory and with a autoregressive component of strong autocorrelation, the Lo's performs far less powerful than the GPH test.

(3) Although *S-MLE* method does not provide a statistic test for the existence of long-memory, the estimates of *d* seems to give a good indication of whether or not the long-memory is present. It is shown by our simulation study that:

a) For AR(1) processes, *S-MLE* gives basically correct estimates of *d*, i.e., $d = 0$, even when the autoregressive coefficients are very high, although the estimates are slightly positively biased when the data size is small (e.g., 500 points). The estimates get more accurate (according to the averages) and more stable (according to the standard deviations) with the increase of sample size.

b) For ARFIMA processes, *S-MLE* provides significantly downwardly biased estimates when the data size is small (e.g., less than 10^3). The estimates of d given by S-MLE increase with increasing sample size and are basically correct when the data size is close to 10^4. But the estimates of d get upwardly biased when the data size is too big (say, > 10^4). This is in contradiction with the result of Kendziorski (1999), who showed that S-MLE provided unbiased estimates of d for ARFIMA(0,d,0) processes of length 2^{11} (2048) or greater.

(4) Data size has a significant impact on the power of all the three method. The power of Lo's test and GPH test increases with the increase of data size, and the estimates of *d* with GPH test and *S-MLE* converge with the increase of data size. Agiakloglou et al. (1993) found that GPH estimators performed poorly for AR(1) processes with $\phi = 0.9$ for sample size of 100 to 900. The simulation results of Hurvich and Beltrao (1993) also showed the poor performance of the GPH estimator when $\phi = 0.9$ for not only AR(1) processes but also ARFIMA(1,d,0) processes. In our simulation study, it is shown that, on one hand, the power of GPH test does decrease with the increase of the autoregressive coefficient; on the other hand, the power of GPH test increases with the increase of sample size. If we use a sample size of larger than 10^4 points, GPH test still has very good performance for AR(1) processes with $\phi = 0.9$. But the use of GPH test is helpless when ϕ is larger than 0.95, even with a data size of larger than 10^4. One possible solution could be to choose the number of frequencies used in the regression Equation (5) more carefully (Giraitis et al., 1997; Hurvich and Deo, 1999). But the effectiveness of these methods is yet to be

confirmed. For example, as ϕ increases, the estimates of d using the number of frequencies $g(n)$ selected by the plug-in method proposed by Hurvich and Deo (1999) are much more positively biased than simply using $g(n) = n^{1/2}$.

(5) Teverovsky et al. (1999) pointed out that, picking a single value of q with Lo's test to determine whether or not to reject the null hypothesis of no long-range dependence in a given data set is highly problematic. In consequence, they recommend that one always relies on a wide range of different q-values, and does not use Lo's method in isolation, instead, uses it always in conjunction with other graphical and statistical techniques for checking for long-memory, especially when Lo's method results in accepting the null hypothesis of no long-range dependence. While we agree that we should not use Lo's method in isolation, it is doubtful that using a wide range of different q-values may improve the test reliability. With a wide range of q-values, you are still not sure which one gives the right answer.

On the basis of the above findings, to obtain reliable test results on detecting the presence of long-memory, we have two suggestions: First, increase the size of test data, as we see that the power of Lo's test and GPH test increases with the increase of data size, and the estimates of d with the GPH-test and *S-MLE* converge as the sample size increases, but notice that the estimate with *S-MLE* would be biased upwardly when the data size is above 10^4; Second, use the detection results in combination with each other, as have been suggested by Teverovsky et al. (1999). Here we consider the combined use of Lo's test, GPH-test and *S-MLE*. As shown in Table 2.11, the combined use of these three methods produces the following alternatives:

a) Failure to reject by both the Lo's test and the GPH-test, and low values of estimated d (e.g., <0.1) with *S-MLE*, provide evidence in favour of no existence of long-memory
b) Rejection by both Lo's test and GPH test suggests, and high values of estimated d (e.g., >0.2) with *S-MLE*, support that the series is a long-memory process
c) In other cases, the data are not sufficiently informative with respect to the long-memory properties of the series.

We especially recommend the combined use of GPH test and *S-MLE* to detect the existence of long-memory, and the most appropriate date size for estimating d with *S-MLE* is slightly less than 10^4.

2.5.3.5 *Long-memory test for streamflow processes*

According to what we found with the Monte Carlo simulations, we use the Lo's R/S test, GPH test and *S-MLE* method jointly to detect the existence of long-memory in streamflow processes in this study. For Lo's modified R/S test, we adopt the data-driven formula (2.30) to choose the lag q. For GPH test, we choose the number of frequencies $g(n) = n^{-0.5}$ as suggested by Geweke and Porter-Hudak (1983). The null hypothesis of no long-term dependence is rejected if $Q_{N,q}$ in Lo's test is not contained in the interval [0.809, 1.862] (Lo, 1991), or if t-statistic in GPH test is not contained in the interval [-1,960, 1.960]. With the *S-MLE* method, we assume that the processes are ARFIMA(p,d,0) processes, and the order p of AR component is determined by using the AIC criterion (Akaike, 1973).

All the streamflow series are log-transformed and deseasonalized. In addition, the streamflow series of the Yellow River at TG are detrended first. The test results for all streamflow series are listed in Table 2.12. The results show that, the intensity of long-memory decreases with the increase of timescale according to the results of all the three methods. All daily flow series

exhibit strong long-memory, because the presence of long-memory is confirmed by all the three methods for 4 cases, and in another two cases (Danube and Rhine), it is confirmed by the GPH test and S-MLE method. The presence of long-memory in 1/3-monthly series is confirmed in three cases (Yellow at TNH, Yellow at TG, and Umpqua), rejected in two cases (Danube and Rhine), and not conclusive in one case (Ocmulgee). For monthly series, the existence long-memory is rejected by both the GPH test and S-MLE method for four cases because the hypothesis of no long-memory is accepted by the GPH test, and S-MLE give a estimate of *d* less than 0.2. But the monthly series of Yellow River at TG and Umpqua may exhibit long-memory.

Table 2.12 Detecting the existence in streamflow series with Lo's modified *R*/*S* test, GPH test and S-MLE method

River (station)	Timescale	Lo's test		GPH test		*S-MLE*
		Lag	statistic	statistic	*d*	*d*
Yellow (TNH)	Daily	94	2.5111*	7.7853 *	0.4720	0.4922
	1/3-monthly	11	2.2910 *	3.277 0*	0.3854	0.4518
	Monthly	5	1.9644 *	1.4233	0.2357	0.0000
Yellow (TG)	Daily	39	3.06975 *	5.4234 *	0.3422	0.3561
	1/3-monthly	7	2.4679 *	2.9501 *	0.3636	0.3194
	Monthly	3	2.1437 *	1.3756	0.2415	0.3400
Danube (Achleiten)	Daily	63	1.7273	5.4564 *	0.2742	0.3865
	1/3-monthly	8	1.5512	0.8792	0.0848	0.0001
	Monthly	4	1.3044	-0.1307	-0.0176	0.0000
Rhine (Lobith)	Daily	170	1.5288	6.5402 *	0.3229	0.4167
	1/3-monthly	11	1.6853	1.1944	0.1129	0.0000
	Monthly	5	1.4853	0.1528	0.0202	0.0000
Ocmulgee (Macon)	Daily	31	2.7826 *	7.1878 *	0.3821	0.4663
	1/3-monthly	6	2.0852 *	1.8812	0.1916	0.1956
	Monthly	4	1.6260	1.4253	0.2039	0.1368
Umpqua (Elkton)	Daily	58	3.1110 *	5.6400 *	0.2785	0.4445
	1/3-monthly	8	2.6341 *	2.3899 *	0.2259	0.2189
	Monthly	4	2.2376 *	2.5076 *	0.3313	0.1799

Note: An asterisk indicates the rejection of the null hypothesis of no long-memory at the 0.05 significance level.

A special concern here is the value of d for the daily streamflow process of the Yellow River at TNH, because we will model and forecast this streamflow process later. The estimates of d given by the GPH test and S-MLE are 0.472 and 0.4922, respectively. In addition, with S-MLE, we know that the process also has an AR component of high autoregressive coefficients. The size of the series is 16437 points. As we know from the results for simulation ARFIMA series, for a series of this size and strong autocorrelations, both GPH method and S-MLE method give positively biased estimates of *d*. Therefore, taking the results from the heuristic methods into account, we will consider a *d* of less than 0.4 when modeling the daily process at TNH.

2.6 Conclusions

There is no obvious trend in the average annual flow process of the upper Yellow River at TNH over the period 1956 to 2000, whereas the discharges recorded at a downstream station TG exhibit significantly declining trend. Fu et al. (2004) investigated the trend of annual runoffs at another three stations along the mainstream of the Yellow River. Put together the results, we find that the lower the reaches of the Yellow River, the more significant the declining trend. However, generally, there is no significant decline in the precipitation processes in the Yellow River basin (Fu et al., 2004). The phenomenon that the lower the reaches of the Yellow River, the more significant the downward trend is a clear indication of anthropogenic influence, because the lower the reaches, the more human intervention the river would suffer, including the expansion of irrigation areas, the regulation of thousands of reservoirs in both the main channel and tributaries, and the increase of water consumption with the fast growing industry and population. Although the impacts of global warming on water supply are widely concerned, in the case of the Yellow River basin, the impacts of warming on the river flow processes of the Yellow River seem far less significant than anthropogenic influences.

The Augmented Dickey-Fuller (ADF) unit root test (Dickey and Fuller, 1979; Said and Dickey, 1984) and KPSS test (Kwiatkowski et al., 1992) are introduced to test for the nonstationarity in streamflow time series. It is shown that the smaller the timescale of the streamflow process, the more likely it tends to be nonstationary.

Seasonal variations in the autocorrelation structures are present in all the deseasonalized daily, 1/3-monthly and monthly streamflow processes, albeit that such seasonal variation is less obvious for the streamflow of the Danube and the Ocmulgee. This indicates that, even after the deseasonalization procedure, the seasonality still shows itself, not in the mean and variance, but in the autocorrelation structure.

The investigation of the long-memory phenomenon in streamflow processes at different timescales shows that, with the increase of timescale, the intensity of long-memory decreases. According to the Lo's R/S tests (Lo, 1991), GPH test (Geweke and Porter-Hudak, 1983) and the maximum likelihood estimation method implemented in S-Plus version 6 (*S-MLE*), all daily flow series exhibit strong long-memory. Out of six 1/3-monthly series, three series (Yellow River at TNH and TG, and Umpqua) exhibit long-memory, whereas two 1/3-monthly series (Danube and Rhine) seem to be short-memory series. Only one monthly flow series (Umpqua) may exhibit long-memory.

Comparing the stationarity test results and the long-memory test results, we find that these two types of test are more or less linked, not only in that the test results have similar timescale pattern, but also in that there is a general tendency that the stronger the nonstationarity, the more intense the long-memory. In fact, there are some attempts to use KPSS stationarity test to test for the existence of long-memory (e.g., Lee and Schmidt, 1996). It is worthwhile to further the investigation on the issue of the relationship between nonstationsrity and the long-memory in the future.

Appendix 2.1 Hypothesis Testing

Setting up and testing hypotheses is an essential part of statistical inference. In order to carry out a statistical test, it is necessary to define the null and alternative hypotheses; which describe what the test is investigating. In each problem considered, the question of interest is simplified into two competing claims / hypotheses between which we have a choice; the null hypothesis, denoted H_0 (e.g., there is no significant change in the annual maximum flow series), against the alternative hypothesis, denoted H_1 (e.g., the annual maximum flow is changing over time). In carrying out a statistical test one starts by assuming that the null hypothesis is true, and then checks whether the observed data are consistent with this hypothesis. The null hypothesis is rejected if the data are not consistent with H_0.

To compare between the null and the alternative hypotheses a test statistic is selected and then its significance is evaluated, based on the available evidence. A test statistic is a quantity calculated from our sample of data subject to testing. Its value is used to decide whether or not the null hypothesis should be rejected in our hypothesis test. The choice of a test statistic will depend on the assumed probability model and the hypotheses under question.

The significance level of a statistical hypothesis test is a fixed probability of wrongly rejecting the null hypothesis H_0, if it is in fact true. It is the probability of a type I error. Usually, the significance level is chosen to be 0.05.

The probability value (p-value) of a statistical hypothesis test is the probability of getting a value of the test statistic as extreme as or more extreme than that observed by chance alone, if the null hypothesis H_0 is true. It is equal to the significance level of the test for which we would only just reject the null hypothesis. Small p-values suggest that the null hypothesis is unlikely to be true. The smaller it is, the more convincing is the rejection of the null hypothesis.

The diagram below represents four outcomes of the decisions we make, in terms of whether or not the null is true, and whether we reject the null or not.

Decision	Truth of Null	
	True	**Not True**
Reject Null	**TYPE I**	**POWER**
FTR Null	**CORRECT**	**TYPE II**

As you see, FTR (failed to reject) the null when the null is true is a correct decision. However, we're usually interested in trying to find true differences, and therefore look to reject null hypotheses. Rejecting the null when it is really not true is a correct decision as well. More specifically, the probability a test has to do this is referred to as *power*. Power may be defined as the probability of correctly rejecting the null hypothesis. In other words, it is the probability of rejecting the null hypothesis given that the null is incorrect. Some people also refer to power as *precision* or *sensitivity.*

Appendix 2.2 Stationarity and Periodic Stationarity

Let $\{x_t\}$, $t = 1, \ldots, N$, be N consecutive observations of a seasonal time series with seasonal period s. For simplicity, assume that $N/s = n$ is an integer. In other words, there are n full years of data available. The time index parameter t may be written $t = t(r\text{-}m) = (r\text{-}1)s + m$, where $r = 1, \ldots, n$ and $m = 1, \ldots, s$. In the case of monthly data s = 12 and r and m denote the year and month.

If

$$\mu_m = E(z_{t(r,m)})$$

and

$$\gamma_{l,m} = \mathrm{cov}(z_{t(r,m)}, z_{t(r,m)-l})$$

exist and depend only on l and m, z_t is said to be periodically correlated or periodic stationary (Gladysev, 1961). Note that the case where μ_m and $\gamma_{t,m}$ do not depend on m reduces to an ordinary covariance stationary time series.

A series $\{x_t\}$ is called stationary if, loosely speaking, its statistical properties do not change with time. More precisely, $\{x_t\}$ is said to be completely stationary if, for any integer k, the joint probability distribution of $x_t, x_{t+1}, \ldots, x_{t+k-1}$ is independent on the time index t (see e.g., Priestley, 1988, pp. 4-5).

Chapter 3 ARMA-Type Models for Streamflow Processes

In this chapter, the autoregressive moving average (ARMA) model and its two derivatives, i.e., the autoregressive fractionally integrated moving average (ARFIMA) model and the periodic autoregressive (PAR) model will be applied to the daily streamflows, and the AR model will be applied to the monthly streamflows of the upper Yellow River at TNH. In addition, the construction of prediction intervals will be discussed using four monthly series, and the issue of predictability will be discussed with 31 daily streamflow series at different watershed scale.

3.1 Building ARMA Models for Daily and Monthly Flows

3.1.1 Introduction to ARMA Model

One of the most important and highly popularized time series models is the ARMA-type model (including AR, ARMA, ARIMA and seasonal ARIMA model) introduced by Box and Jenkins (1976). It has a long history of being applied to streamflow forecasting problems. For instance, McKerchar and Delleur (1974) used an ARMA process to achieve stochastic modeling of monthly flows. McLeod et al. (1977) applied the ARMA approach to average annual streamflows. Three types of seasonal time series models are commonly used to model hydrological processes which usually have strong seasonality (Hipel and McLeod, 1994): (1) seasonal autoregressive integrated moving average (SARIMA) models; (2) deseasonalized ARMA models; and (3) periodic ARMA models. Deseasonalized modeling approach is adopted in this study.

The general form of ARIMA(p,d,q) model is given by

$$\phi(B)\nabla^d x_t = \theta(B)\varepsilon_t, \tag{3.1}$$

where B is the backshift operator, that is, $Bx_t = x_{t-1}$; $\phi(B) = 1 - \phi_1 B - \ldots - \phi_p B^P$, and $\theta(B) = 1 - \theta_1 B - \ldots - \theta_q B^q$ represent the ordinary autoregressive and moving average components; ε_t is a white noise process with zero mean and variance σ^2. $\nabla = 1 - B$ is the first-order difference operator and $\nabla^d = (1 - B)^d$ is the d-fold differencing. When $d = 0$, ARIMA(p,d,q) model reduces to ARMA(p,q) model

$$\phi(B)x_t = \theta(B)\varepsilon_t \tag{3.2}$$

When $q = 0$, then ARMA(p,q) model further reduces to AR(p) model.

Box and Jenkins (1976) give the following paradigm for fitting ARIMA models.

(1) *Model identification*: Determination of the ARIMA model orders.
(2) *Estimation of model parameters*: The unknown parameters in Equation (3.1) are estimated.
(3) *Diagnostics and model criticism*: The residuals are used to validate the model and suggest potential alternative models, which may be better.

These steps are repeated until a satisfactory model is found. We give some brief descriptions to these steps in this section.

3.1.1.1 *ARMA model identification*

Initial model identification is done using the autocorrelation function (ACF) and partial autocorrelation function (PACF). Chapter 6 of Box and Jenkins (1976) gives a complete discussion on the identification of ARIMA models.

An alternative procedure for selecting the model order is using a penalized log-likelihood measure. One popular measure is the Akaike's Information Criterion (AIC) (Akaike, 1973), which is defined as

$$AIC(k) = -2\log ML + 2k \tag{3.3}$$

where *ML* is maximum likelihood and *k* is the number of independently adjusted parameters within the model. The best model is given by the model with the lowest *AIC* value. For ARMA(p,q) models, $k = p + q$, and the *AIC* can be calculated as (Akaike, 1974)

$$AIC(p,q) = N\log(\hat{\sigma}_\varepsilon^2) + 2(p+q)$$

where $\hat{\sigma}_\varepsilon^2$ is the variance of the innovation process.

3.1.1.2 *Estimation of ARMA model parameters*

The most popular method for estimating the parameters of ARMA models is the maximum likelihood (*ML*) method. The values of the parameters that maximize the likelihood function, or equivalently the log-likelihood function, are called *maximum likelihood estimates*. Finding maximum likelihood estimates conceptually involves two steps. First, the likelihood function must be calculated. Second, values of the parameters must be found that maximise this function with some numerical optimisation method.

The log-likelihood for an ARMA model can be computed using the prediction error decomposition (see Harvey, 1993). Let ε_t be the one-step ahead prediction error, the variance of the prediction error can be written as

$$Var(\varepsilon_t) = \sigma^2 f_t, t = 1,...,N$$

where σ^2 is the variance of the residual process $\{\varepsilon_t\}$; f_t is a scale parameter. Then, the log-likelihood is given by

$$\log L(x_{p+1},...,x_N) = -\frac{N}{2}\log(2\pi\sigma^2) - \frac{1}{2}\sum_{t=1}^{N}\log f_t - \frac{1}{2\sigma^2}\sum_{t=1}^{N}\frac{\varepsilon_t^2}{f_t}$$

The conditional log-likelihood approximation can be obtained by conditioning on the first *p* values of the series, where *p* is the order of the autoregressive operator. This conditional log-likelihood function is given by

$$\log L(x_{p+1},...,x_N \mid x_1,...,x_p) = -\frac{(N-p)}{2}\log(2\pi\sigma^2) - \frac{1}{2}\sum_{t=p+1}^{N}\log f_t - \frac{1}{2\sigma^2}\sum_{t=p+1}^{N}\frac{\varepsilon_t^2}{f_t} \tag{3.4}$$

The main advantage of using the conditional log-likelihood approximation is that the AR parameters can be concentrated out of the likelihood, reducing the computational complexity of the nonlinear optimization. In statistics package S-Plus, the function *arima.mle* fits

ARIMA models to univariate time series data through Gaussian maximum likelihood, and the conditional form of the likelihood is used (Insightful Corporation, 2001).

3.1.1.3 *Diagnostic checking*

The third stage in building ARIMA models consists of validating the model through examination of the one-step prediction residuals $\{\varepsilon_t\}$. A basic diagnostic technique is to examine the autocorrelation function of the residuals. It is well-known that for random and independent series of length n, the lag k autocorrelation coefficient is normally distributed with a mean of zero and a variance of $1/n$, and the 95 percent confidence limits are given by $\pm 1.96/\sqrt{n}$. The presence of large autocorrelations indicates that a model may be inadequate.

In addition to examining the autocorrelation individually, it is useful to base a diagnostic on the autocorrelations as a whole. Define the portmanteau test statistic Q by (Box and Pierce, 1970)

$$Q = N\sum_{k=1}^{L} \hat{r}_k^2(\varepsilon) . \tag{3.5}$$

It is approximately chi-squared with L-(p+q) degrees of freedom. The adequacy of the model is therefore rejected at level α if $Q > \chi^2_{1-\alpha}(L-p-q)$.

A modified version of portmanteau test is the Ljung-Box test (Ljung and Box, 1978), which tests whether the first L autocorrelations $\hat{r}_k^2(\varepsilon)$ (k = 1, ..., L) from a process are collectively small in magnitude. Suppose we have the first L autocorrelations $\hat{r}_k(\varepsilon)$ (k = 1, ..., L) from any ARMA(p, d, q) process. For fixed sufficiently large L, the Ljung-Box Q-statistic is given by

$$Q = N(N+2)\sum_{k=1}^{L} \frac{\hat{r}_k^2(\varepsilon)}{N-k} \tag{3.6}$$

where N is the sample size, L is the number of autocorrelations included in the statistic (typically between 10 and 20), and $\hat{r}_k^2(\varepsilon)$ is the squared sample autocorrelation of residual series $\{\varepsilon_t\}$ at lag k. Under the null hypothesis of model adequacy, the test statistic is asymptotically $\chi^2(L\text{-}p\text{-}q)$ distributed. Thus, we would reject the null hypothesis at level α if the value of Q exceeds the (1-α)-quantile of the $\chi^2(L\text{-}p\text{-}q)$ distribution.

3.1.2 Fit ARMA Models to Daily and Monthly Streamflows of the Yellow River

The procedure of fitting deseasonalized ARMA models to daily and monthly streamflow at TNH includes two steps: deseasonalization and ARMA model construction. We first log-transform both flow series, and deseasonalize them by subtracting the seasonal (e.g., daily or monthly) mean values and dividing by the seasonal standard deviations of the log-transformed series. To alleviate the stochastic fluctuations of the daily means and standard deviations, we smooth them with the first 8 Fourier harmonics before using them for standardization. Figure 3.1 and 3.2 show the ACF and PACF of the deseasonalized daily and monthly series.

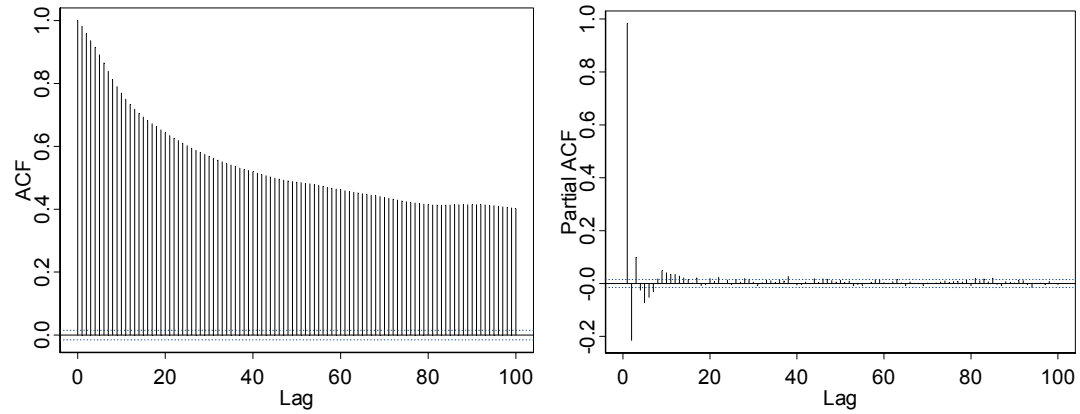


Figure 3.1 ACF and PACF of deseasonalized daily flow series at TNH

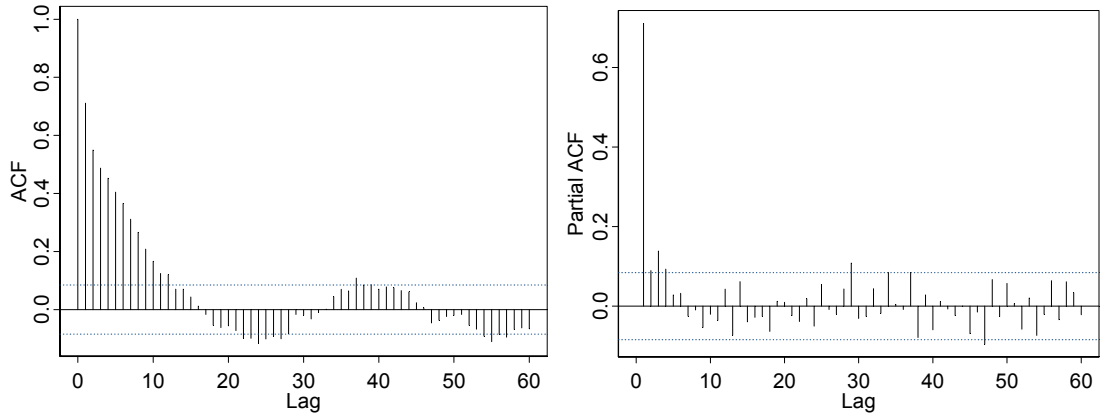


Figure 3.2 ACF and PACF of deseasonalized monthly flow series at TNH

Then, according to the ACF and PACF structures of the two series as well as the model selection criterion AIC, an ARMA (20,1) is fitted to the daily flow series and an AR (4)) fitted to the monthly flow series. The parameters are estimated with the *arima.mle* function in S-Plus version 6, where the Gaussian maximum likelihood estimation method is used.

To examine the goodness of fit of the ARMA(20,1) model for daily flow series and AR(4) for monthly flow series, firstly, we inspect the ACF of the residuals from the models. The ACF plots in Figure 3.3 show that there is no significant autocorrelation left in the residuals from both ARMA-type models for daily and monthly flow.

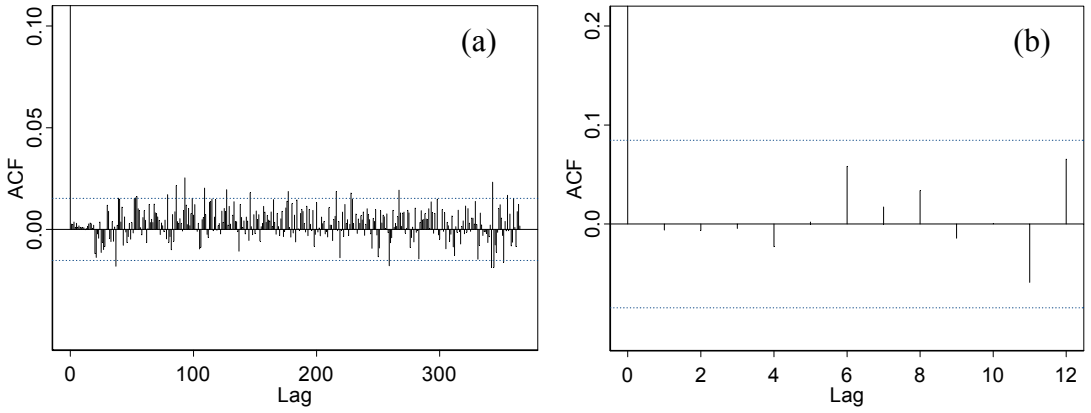


Figure 3.3 ACFs of residuals from (a) ARMA(20,1) model for daily flow and (b) AR(4) model for monthly flow at TNH

Then, more formally, we apply Ljung-Box test to the residual series. The Ljung-Box test results for ARMA(20,1) and AR(4) are shown in Figure 3.4. The p-values' exceedance of 0.05 indicates the acceptance of the null hypothesis of model adequacy at significance level 0.05.

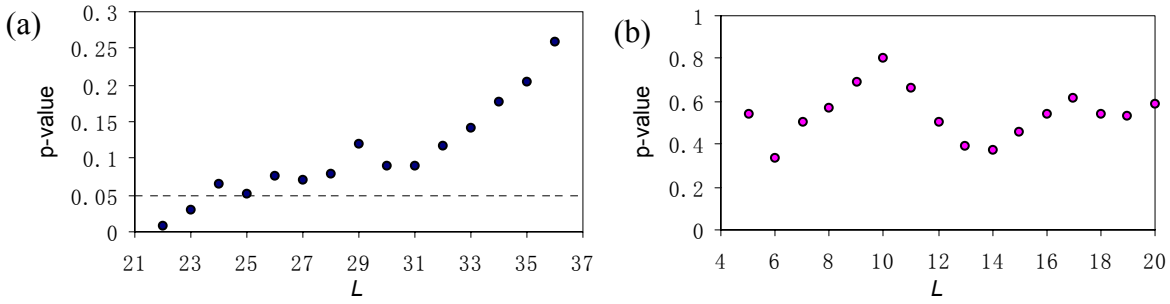


Figure 3.4 Ljung-Box lack-of-fit tests for (a) ARMA(20,1) model for daily flow and (b) AR(4) model for monthly flow at TNH.

3.2 Building ARFIMA Model for Daily Flows

3.2.1 Introduction to ARFIMA Model

The general form of ARFIMA(p,d,q) model is given by

$$\phi(B)\nabla^d x_t = \theta(B)\varepsilon_t,\ |d| < 0.5,$$

where $\phi(B)$, $\theta(B)$ and ∇^d are the same as in Equation (3.1). The difference operator $\nabla^d=(1-B)^d$ by means of the binomial expansion

$$\nabla^d = (1-B)^d = \sum_{j=0}^{\infty} \pi_j B^j,$$

where, $\pi_j = \dfrac{\Gamma(j-d)}{\Gamma(j+1)\Gamma(-d)} = \prod_{0\le k\le j} \dfrac{k-1-d}{k},\ j = 0, 1, 2, \ldots$

The parameters ϕ, θ and d of ARFIMA model may be estimated by maximum likelihood method or log-periodogram based regression method (see e.g., Brockwell and Davis, 1991, pp 527 – 532). The S-Plus function *arima.fracdiff* fits ARFIMA models to univariate time series data through approximate Gaussian maximum likelihood algorithm of Haslett and Raftery (1989).

3.2.2 Fit ARFIMA Model to Daily Streamflow of the Yellow River

Similar to constructing the ARMA (20,1) model to the daily streamflow series, the procedure of fitting deseasonalized ARFIMA models also includes two steps: deseasonalization and ARFIMA model construction. We only consider using the ARFIMA(p,d,0) model for forecasting the daily streamflows of the upper Yelloe River at TNH. Deseasonalization is done in the same way as that for building the ARMA model. The order p of the AR component in the ARFIMA model is determined according to AIC. The chosen model is ARFIMA(7,d,0). The parameters of the AR component and d are estimated using the function *arfima.fracdiff* in the S-Plus. If without any limitation, the function *arfima.fracdiff* gives an

estimate of 0.4922 for d. But as having been analyzed in Section 2.5.3.5, we limit the value of d to less than 0.4 when making the maximum likelihood estimation, thus get an ARFIMA(7,d,0) model with $d \approx 0.4$.

To check the goodness-of-fit of the ARFIMA(7,d,0) model, we inspect the ACF of the residuals from the ARFIMA models. The ACF is plotted in Figure 3.5. It is shown that, no significant autocorrelations is observed for lags less than 6, but there is still significant autocorrelations left at lag 7 to lag 10 in the residuals from the ARFIMA model whereas no significant autocorrelation is observed in the residuals from the ARMA model. For the long lags, the autocorrelations left in the residuals are not significant. Therefore, generally, the ARFIMA model fit the daily flow series well. Comparing the autocorrelation structure of the residuals from the ARFIMA model (Figure 3.5) and that from ARMA model (Figure 3.3a), it is interesting to notice that the number of negative autocorrelations and the number of positive ones are basically equal in the residuals from the ARFIMA model, whereas there are far more positive autocorrelations than negative ones in the residuals from the ARMA model. This may be an indication that the ARFIMA model has a better fit for long lead times.

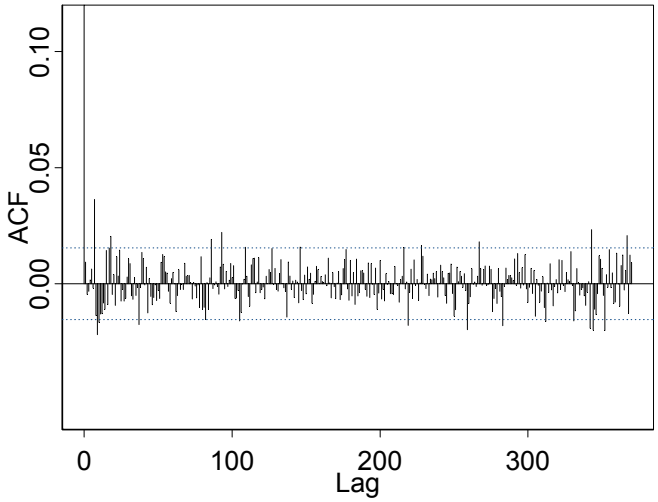


Figure 3.5 ACF of residuals from ARFIMA(7,d,0) for daily flows at TNH

3.3 Building PAR Model for Daily Flows

3.3.1 Introduction to the PAR Model

Streamflow processes usually have seasonal means, variances and serial dependence structures, as having been annalyzed in Chapter 2. A usual procedure in modeling seasonal streamflow series is to deseasonalize the series by first subtracting the seasonal mean and then dividing by the seasonal standard deviation. However such a procedure can only remove the seasonality in the mean and variance, but the seasonality in serial dependence structure remains. To model appropriately such seasonality, periodic models can be employed. Two popular periodic models are the PAR (periodic autoregressive) and PARMA (periodic autoregressive and moving average) models, which are extensions of ARMA models that allow periodic (seasonal) parameters.

The PAR family of models was originally introduced by Thomas and Fiering (1962) for monthly streamflow modeling and simulation. Literature on PAR or PARMA models has abounded since late 1960s' (e.g., Jones and Brelsford, 1967; Salas et al., 1982; Noakes et al., 1986; Salas and Abdelmohsen, 1993; McLeod, 1994). However, PAR or PARMA usually

applied to time series at the time scale of a month or at least a week. Even when applying to such large-time scale time series, there is also a problem of making the model parsimonious, namely, decreasing the number of model parameters required, and fitting a parsimonious PAR model. Salas et al. (1980) proposed a Fourier series approach for reducing the number of parameters in PAR or PARMA models. The same approach is adopted for fitting so-called seasonally varying runoff coefficient (SVRC) model in which more independent exogenous inputs are included in the model (Kachroo and Liang, 1992). Thompstone et al. (1985) proposed alternative approach for developing PPAR model by evaluating the compatibility of the AR equations for neighboring seasons based on residual variance analysis, then combining compatible individual AR models for adjacent seasons. However, when applying to daily flow series, with respect to the former approach, although the number of parameters could be reduced, the order of PAR model would be the same for all seasons, which loses generality and may not be appropriate for catching seasonal dynamics of streamflow. With respect to the latter approach, separate AR models should be fitted to each day of the year, and the computation for calculating the compatibility between neighboring days are overwhelming which makes it infeasible.

Here a method for fitting PAR model to daily flow series is presented based on the cluster analysis. A PAR model is fitted the daily streamflow series of the upper Yellow River at TNH and forecasting experiments are made.

3.3.2 Partition the Days over the Year with the Fuzzy Clustering Technique

When partitioning the days over the year with the clustering analysis method, the raw average daily discharge data and the autocorrelation values at different lag times (1 ~ 10 days) are used. The daily discharge data and the autocorrelation coefficients are organized as a matrix $\boldsymbol{X}$ of the size $(N+10)\times 365$, where N is the number of years and "10" represents the autocorrelation values at 10 lags. To eliminate the influence of big differences among data values on cluster analysis result, the log-transformation is first applied to the daily discharges before making the cluster analysis. Then the 365 days over the year are partitioned with the fuzzy c-means (FCM) clustering method.

FCM clustering is proposed by Bezdek (1981) as an improvement over the hard k-means clustering algorithm. The FCM method partitions a set of n vector x_j, $j = 1, \ldots, n$, into c fuzzy clusters, and each data point belongs to a cluster to a degree specified by a membership grade u_{ij} between 0 and 1. We define a matrix U consisting of the elements u_{ij}, and assume that the summation of degrees of belonging for a data point is equal to 1, i.e., $\sum_{i=1}^{c} u_{ij} = 1, \forall j = 1,...,n$.

The goal of the FCM algorithm is to find c cluster centers such that the cost function of dissimilarity (or distance) measure is minimized. The cost function is defined by

$$J(U, v_1, ..., v_c) = \sum_{i=1}^{c} J_i = \sum_{i=1}^{c} \sum_{j=1}^{n} u_{ij}^m d_{ij}^2 \tag{3.7}$$

where v_i is the cluster center of the fuzzy group i; $d_{ij} = \|v_i - x_j\|$ is the Euclidean distance between the ith cluster center and the jth data point; and m ≥ 1 is a weighting exponent, taken as 2 here. The necessary conditions for Equation (3.7) to reach its minimum are:

$$v_i = \frac{\sum_{j=1}^{n} u_{ij}^m x_j}{\sum_{j=1}^{n} u_{ij}^m} \tag{3.8}$$

and

$$u_{ij} = \left[\sum_{k=1}^{c} \left(\frac{d_{ij}}{d_{kj}} \right)^{2/(m-1)} \right]^{-1} . \tag{3.9}$$

The fuzzy C-means algorithm is an iterative procedure that satisfies the preceding two necessary conditions as follows.

(1) Initialize the membership matrix U with random values between 0 and 1.
(2) Calculate c fuzzy cluster centers v_i, $i = 1, \ldots, c$, using Equation (3.8).
(3) Compute the cost function according to Equation (3.7). Stop if either it is below a tolerance value or its improvement over the previous iteration is below a certain threshold.
(4) Compute a new U using Equation (3.9). Return to step (2).

The cluster result is shown in Figure 3.6. Comparing Figure 2.9(a) and Figure 3.6, we see that if we just follow the clustering result to partition the days over a year into 5 groups, the dynamics of streamflow is not well captured because cluster 2 and 3 in Figure 3.6 mix the streamflow rising limb and falling limb shown in Figure 2.9(a). Therefore, according to the FCM clustering result, and considering the dynamics of the streamflow process, we partition the 365 days over the year into 7 hard segments, as listed in Table 3.1.

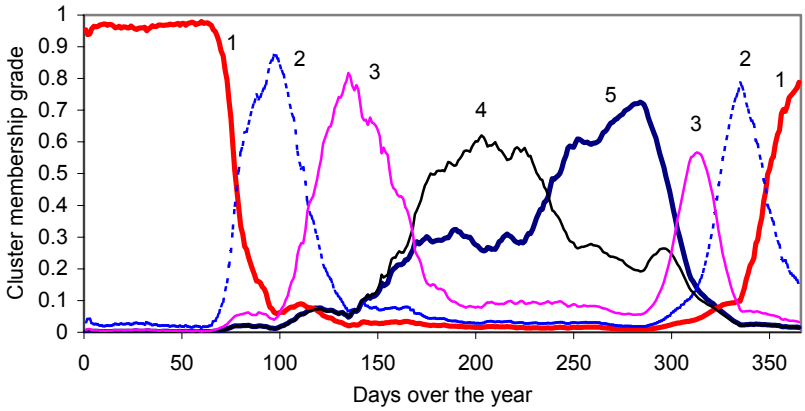


Figure 3.6 Membership grades of the days over the year for the daily streamflow series at TNH with the FCM of 5 clusters

Table 3.1 Partitioning of the days over the year for the daily streamflow at TNH

Partition	1	2	3	4	5	6	7
Day span	1-77, 349-365/366	78-114	115-167	168-237	238-302	303-322	322-348

3.3.3 Building the PAR Model

Based on the partitioning results, one AR model is fitted to one partition. Before fitting the models, the daily streamflow series is deseasonalized. The orders of the AR models are determined according to AIC criterion (Akaike, 1973), and the parameters of these AR models are estimated with the least squares method. The chosen orders for each partition according to minimum AIC are shown in Table 3.2.

Table 3.2 Selected AR orders for the PAR model according to AIC

Partition	1	2	3	4	5	6	7
AIC	22	11	12	12	10	8	14

According to AIC, we fit an AR(22), an AR(11), an AR(12), an AR(12), an AR(10), an AR(8), and an AR(14) to partition 1 to 7 respectively. The parameter estimation is performed by ordinary least squares estimation procedure. Together, these AR models for different seasons compose a PAR model. When forecasting, a specific AR model is applied depending on what season partition the date to be forecasted is in.

3.4 Forecasting with ARMA, ARFIMA and PAR Models

3.4.1 Forecast Evaluation Measures

The following measures are used in the present study to evaluate model performances in streamflow forecasting:

Mean Absolute Error: $$MAE = \frac{1}{n}\sum_{i=1}^{n} | Q_i - \hat{Q}_i |, \tag{3.10}$$

Mean Absolute Percentage Error: $$MAPE = \frac{1}{n}\sum_{i=1}^{n} \frac{| Q_i - \hat{Q}_i |}{Q_i}, \tag{3.11}$$

Root Mean Squared Error: $$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n} (Q_i - \hat{Q}_i)^2}, \tag{3.12}$$

Mean Squared Relative Error: $$MSRE = \frac{1}{n}\sum_{i=1}^{n} \frac{(Q_i - \hat{Q}_i)^2}{Q_i^2}, \tag{3.13}$$

Coefficient of Efficiency: $$CE = 1 - \frac{\sum_{i=1}^{n} (Q_i - \hat{Q}_i)^2}{\sum_{i=1}^{n} (Q_i - \bar{Q})^2}, \tag{3.14}$$

Coefficient of determination: $$r^2 = \left[\frac{\sum_{i=1}^{n} (Q_i - \bar{Q})(\hat{Q}_i - \tilde{Q})}{\sqrt{\sum_{i=1}^{n} (Q_i - \bar{Q})^2 \sum_{i=1}^{n} (\hat{Q}_i - \tilde{Q})^2}} \right]^2, \tag{3.15}$$

where, $\hat{Q}_i$ are the n modelled flows, Q_i are the n observed flows, $\bar{Q}$ is the mean of the observed flows , and $\tilde{Q}$ is the mean of the modelled flows.

There is an extensive literature on model forecasting evaluation indices (e.g. Nash and Sutcliffe, 1970; Garrick et al., 1978; Wilmott et al., 1985, Legates and McCabe, 1999; and Kneale et al., 2001). According to Karunanithi et al. (1994), mean squared errors (MSE), or equivalently RMSE, provide a good measure of the goodness-of-fit at high flows, whilst relative errors (MSRE) provide a more balanced perspective of the goodness-of-fit at moderate flows. *CE* and r^2 provide useful comparisons between studies since they are independent of the scale of data used. The r^2 measures the variability of observed flow that is explained by the model. Despite its crudeness and identified weaknesses (Kachroo and Natale,

1992), the *CE* introduced by Nash and Sutcliffe (1970) is still one of the most widely used criteria for the assessment of model performance. The *CE* reflects the fraction of the total sum of the squares of the observations explained by a model. It provides a measure of the ability of a model to predict flows that are different from the mean. A *CE* of 0.9 and above is very satisfactory, 0.8 to 0.9 represents a fairly good model, and below 0.8 is considered unsatisfactory (e.g., Shamseldin, 1997).

Albeit its popularity, the utilization of *CE* has several problems when it is applied to a model fitted to a seasonally stationary series, such as streamflow processes. Firstly, we know that a value of zero for the coefficient of efficiency indicates that the observed mean is as good a predictor as the model, while a negative value indicates that the observed mean is a better predictor than the model (Wilcox et al., 1990). But for hydrological time series, which usually have strong seasonality, what we concern is whether the model is better than seasonal mean values of the series rather than overall observed mean. This is a question that cannot be answered by *CE* value obtained from Equation (3.14). Secondly, there is an interesting phenomenon that when we assess model performance in terms of *CE*, the *CE* value calculated for the whole year will be higher than the average of *CE* values calculated for separate seasons, which illogically indicates that the model performance for the whole year is better than for most separate seasons. In fact, this problem will also be encountered when applying another commonly used goodness-of-fit criterion - coefficient of determination.

These problems arise from the inadequacy of the definition of *CE* for dealing with seasonal processes. As aforementioned, *CE* is defined under the assumption that the process of interest is stationary (Bhansali, 1992), whereas hydrological time series are usually strongly seasonal. When seasonality exists, especially when the mean value changes with season, for most of the seasons (such as days or months) in a year, the value of overall standard deviation is larger than the values of seasonal standard deviation. Take the case of the daily streamflows at TNH for instance, the overall standard deviation (calculated with the overall mean) is about 559.5 m^3/s, while the average of daily standard deviations (calculated for the average discharges in each day over the year) is about 275.7 m^3/s (see Figure 3.7). Therefore, when only considering overall mean value, the denominator of the right-most item in Equation (3.14), which indicates the overall standard deviation of the observed series, is larger than if seasonal means are taken into account. In consequence, using overall mean gives rise to a larger *CE*, that is, the performance of the forecasting model is exaggerated.

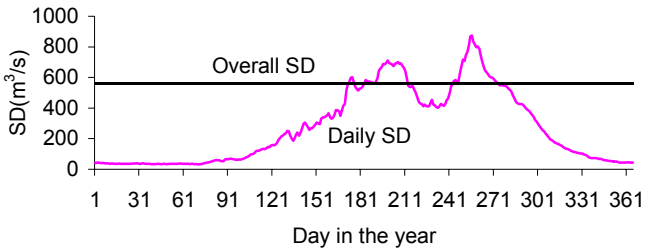


Figure 3.7 The overall standard deviation (SD) compared with the SDs of discharges of each day over the year for daily streamflows at TNH

To overcome these problems, seasonal mean values must be considered for calculating *CE*, by replacing $\bar{Q}$ in Equation (3.14) with seasonal mean values, which vary depending on what season for which the forecast to be made is in. Therefore, we get the seasonally-adjusted *CE* (*SACE*) for as

$$SACE = 1 - \frac{\sum_{i=1}^{n}(Q_i - \hat{Q}_i)^2}{\sum_{i=1}^{n}(Q_i - \bar{Q}_m)^2} \tag{3.16}$$

where $m = i \bmod S$ (mod is the operator calculating the remainder) is the "*season*", ranging from 0 to S-1; and S is the total number of "*season*" (Note that, a "*season*" here does not necessarily denote a real season. It may be a month or a day depending on the timescale of the time series. For daily streamflow series, a season is a day over the year.); $\bar{Q}_m$ is the mean value of season m. For daily series, $\bar{Q}_m$ is given by

$$\bar{Q}_m = \frac{1}{N}\sum_{j=i}^{N} Q_{j,m}$$

where m is a day over the year, and N is the total number of years for which the streamflow forecasts are made. Three points should be noticed regarding the use of seasonal mean values: (i) To know how good the forecasts are compared with long-term seasonal mean values, we may use long-term seasonal mean values instead of the seasonal mean values of the forecasted period; (ii) To avoid the stochastic functuation in the seasonal mean values, especially for the daily means, we may smooth them with first 6 to 8 Fourier harmonics; (iii) when a deterministic trend existing, the trend should be considered by adding a trend term to $\bar{Q}_m$. With this modification, a value of zero for the coefficient of efficiency indicates that the mean value of each season is as good a predictor as the model. Similar modification can be applied to the calculation of coefficient of determination (r^2) for evaluating seasonal time series.

3.4.2 Forecasting Results

3.4.2.1 *Forecsting monthly flows of the Yellow Rvier at TNH*

With a monthly flow series of 45 years, generally, we make forecasts for the last 5 years (1996 ~ 2000) to evaluate the performance of the AR(4) model. The model building is on a rolling-forward basis. That is, we use the data of 1956 ~ 1996 to fit an AR(4) model, and make forecasts for 1997; then use the data of 1956 ~ 1997 to fit another AR(4) model, and make forecasts for 1998; and so on. Data are log-transformed and deseasonalized before being used. The observed discharges versus one-month forecasted discharges are plotted in Figure 3.8. The evaluation results for one-month ahead monthly flow forecasts for each month are reported in Table 3.3.

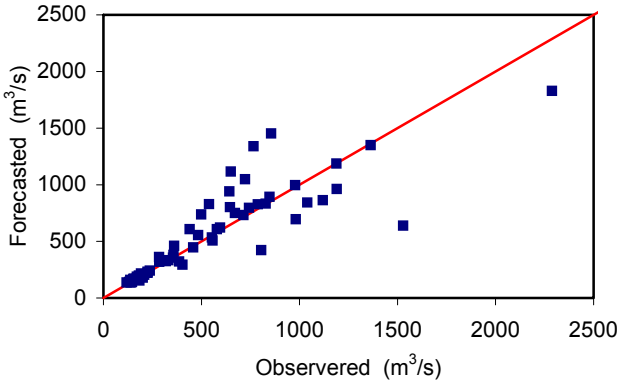


Figure 3.8 1-month ahead forecasts with AR(4) model for monthly flow at TNH

Table 3.3 One-month ahead forecast performance for monthly streamflow at TNH with AR(4) model

Month	MAE	MAPE	RMSE	MSRE	*CE*	r^2
Jan	10.555	0.074	13.34	0.009	0.112	0.680
Feb	13.585	0.089	16.73	0.011	0.389	0.400
Mar	14.424	0.074	16.43	0.007	-0.250	0.292
Apr	57.714	0.168	68.18	0.038	-0.897	0.518
May	145.334	0.254	193.62	0.091	-0.645	0.133
Jun	366.968	0.383	462.33	0.184	-0.853	0.490
Jul	343.083	0.375	399.27	0.231	0.541	0.595
Aug	122.439	0.153	170.31	0.052	0.654	0.668
Sep	199.816	0.285	296.26	0.155	-0.633	0.407
Oct	95.687	0.119	125.56	0.019	0.681	0.756
Nov	12.038	0.029	14.11	0.001	0.973	0.989
Dec	6.936	0.038	9.17	0.003	0.898	0.934
Year	115.715	0.170	213.86	0.067	0.732	0.741

In terms of the values of CE, the performance of the AR(4) model for monthly flows is poor. Satisfying forecasts are available only for November and December with the AR(4) model. However, as shown by the diagnostic checking results in Section 3.1.3, there is no useful information left in the residuals from the AR(4) model. We cannot expect better results with the AR model. Therefore, to improve the skill of monthly streamflow forecasting, exogenous explanatory variables must be included in forecasting models.

Notice that for several months (March to June, and September), the values of *CE* are below zero. That just means that for these months, the forecasted values are not as good as the mean values of the forecasting period. It does not mean that long-term mean values for these months are better than predicted values. Long-term monthly mean values may differ from those of the forecasting period, especially when the sample size is short (e.g., the monthly flows of 5 years). In fact, if we use the long-term monthly mean values as the forecasted values for years 1996 to 2000, *CE* values are negative for all months but December.

3.4.2.2 *Forecasting daily flows of the Yellow River at TNH*

Forecast experiments are made using the ARMA (20,1), ARFIMA(7,d,0) and PAR model for daily flow at TNH during year 1996 to 2000. The model building is also on a rolling-forward basis, as that in the case of monthly flow forecasting. The observed discharges versus forecasted daily discharges for 1-day and 5-day ahead forecasts are plotted in Figure 3.9, 3.10 and 3.11.

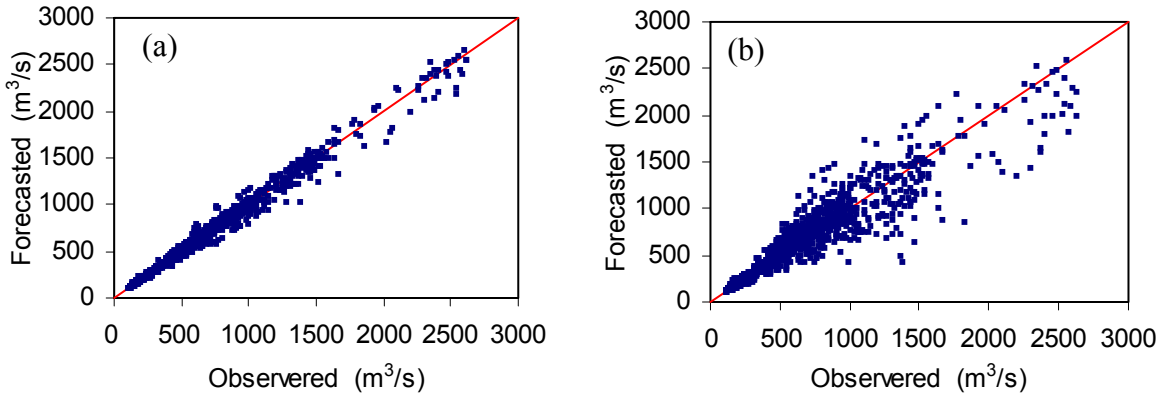


Figure 3.9 (a) 1-day ahead and (b) 5-day ahead forecasts with the ARMA(20,1) model

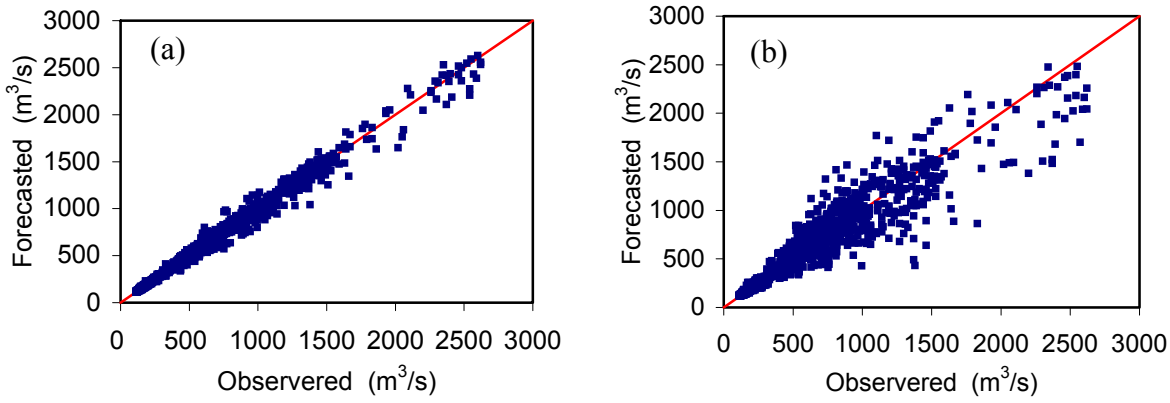


Figure 3.10 (a) 1-day ahead and (b) 5-day ahead forecasts with the ARFIMA(7,d,0) model

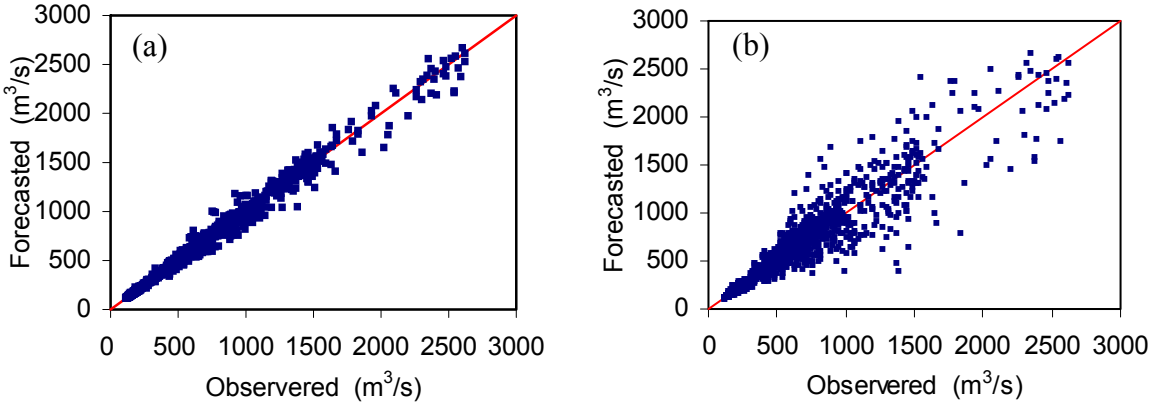


Figure 3.11 (a) 1-day ahead and (b) 5-day ahead forecasts with the PAR model

It is shown from the scatter plots that while there is no visible performance difference for one-day ahead forecasts with the three model, ARMA model and the ARFIMA model seems outperform the PAR model for the forecasts at the long lead time (5 days). However, the forecasts of both the ARMA model and the ARFIMA model slightly biased downwardly for high flows whereas no significant bias is seen with the PAR model.

The forecast results are further evaluated with different measures for the whole year as well as for individual seasons, namely, spring (March to May), summer (June to August), autumn (September to November) and winter (December to February). The evaluation results for 1- to 10-day ahead forecasts with the ARMA(20,1) are listed in Table 3.4-3.8, with the ARFIMA(7,d,0) in Table 3.9-3.13, and with the PAR model in Table 3.14 – 3.18. We have the following remarks with respect to the results of forecast experiments:

(1) In terms of the values of CE, we see that when taking the forecasts of the whole year into account, the measure CE indicates that both the ARMA(20,1) model and the ARFIMA(7,d,0) model may give satisfying forecasts for a lead time of up to 9 days. On the other hand, the lead times for which satisfying forecasts can be achieved for individual seasons are mostly less than 9 days. It means that, the forecast performance is exaggerated by using CE. In contrast, SACE gives a good correction. According to SACE, generally, we can make reliable 5-day ahead forecasts for the daily flow series at TNH.

Table 3.4 Forecast performance of ARMA(20,1) for whole year for daily flows at TNH

Lead time (days)	MAE	MAPE	RMSE	MSRE	*CE*	r^2	*SACE*
1	24.25	0.038	45.93	0.003	***0.989***	0.989	***0.980***
2	42.69	0.066	80.21	0.009	***0.967***	0.967	***0.938***
3	55.44	0.084	104.47	0.014	***0.943***	0.943	***0.895***
4	65.24	0.097	122.87	0.019	***0.922***	0.922	***0.855***
5	73.76	0.110	138.60	0.024	***0.900***	0.900	***0.815***
6	83.25	0.123	153.36	0.030	***0.878***	0.878	0.774
7	92.85	0.135	167.89	0.035	***0.854***	0.854	0.729
8	101.31	0.146	181.53	0.041	***0.829***	0.829	0.683
9	108.37	0.156	193.34	0.047	***0.806***	0.807	0.640
10	114.21	0.164	203.99	0.053	0.784	0.785	0.599

Table 3.5 Forecast performance of ARMA(20,1) for spring for daily flows at TNH

Lead time (days)	MAE	MAPE	RMSE	MSRE	*CE*	r^2
1	17.69	0.041	30.86	0.003	***0.974***	0.974
2	31.56	0.072	54.68	0.01	***0.920***	0.921
3	40.03	0.091	70.73	0.015	***0.867***	0.870
4	46.87	0.105	81.99	0.020	***0.823***	0.829
5	52.53	0.116	91.54	0.024	0.782	0.790
6	59.34	0.129	102.77	0.029	0.732	0.742
7	68.66	0.144	119.80	0.035	0.662	0.672
8	77.76	0.158	135.27	0.042	0.602	0.612
9	85.67	0.169	149.52	0.048	0.551	0.561
10	93.29	0.181	161.39	0.054	0.510	0.521

Table 3.6 Forecast performance of ARMA(20,1) for summer for daily flows at TNH

Lead time (days)	MAE	MAPE	RMSE	MSRE	*CE*	r^2
1	56.57	0.058	81.26	0.006	***0.974***	0.974
2	101.02	0.103	141.58	0.018	***0.921***	0.921
3	132.55	0.136	184.54	0.030	***0.866***	0.866
4	155.78	0.160	216.83	0.042	***0.815***	0.815
5	175.63	0.183	243.82	0.054	0.766	0.766
6	198.59	0.207	268.33	0.066	0.717	0.717
7	220.65	0.231	290.23	0.079	0.670	0.670
8	239.73	0.252	311.07	0.093	0.622	0.622
9	255.07	0.27	328.66	0.107	0.579	0.579
10	267	0.285	344.95	0.121	0.536	0.536

Table 3.7 Forecast performance of ARMA(20,1) for autumn for daily flows at TNH

Lead time (days)	MAE	MAPE	RMSE	MSRE	*CE*	r^2
1	16.95	0.024	27.22	0.001	***0.991***	0.991
2	29.51	0.042	48.09	0.004	***0.971***	0.972
3	38.87	0.055	63.15	0.006	***0.950***	0.951
4	46.97	0.066	76.28	0.008	***0.927***	0.929
5	54.47	0.077	89.41	0.011	***0.899***	0.902
6	61.95	0.089	101.55	0.015	***0.869***	0.875
7	68.07	0.098	112.90	0.018	***0.838***	0.847
8	73.13	0.105	122.99	0.021	***0.808***	0.82
9	77.79	0.111	131.72	0.023	0.779	0.796
10	81.24	0.117	139.21	0.026	0.752	0.774

Table 3.8 Forecast performance of ARMA(20,1) for winter for daily flows at TNH

Lead time (days)	MAE	MAPE	RMSE	MSRE	*CE*	r^2
1	5.05	0.03	7.29	0.002	***0.967***	0.968
2	7.23	0.043	10.35	0.004	***0.927***	0.930
3	8.44	0.051	12.05	0.005	***0.890***	0.897
4	9.01	0.054	13.12	0.006	***0.854***	0.867
5	9.76	0.06	13.82	0.007	***0.821***	0.840
6	10.19	0.063	14.16	0.007	0.794	0.817
7	10.65	0.066	14.17	0.007	0.776	0.801
8	10.93	0.068	14.23	0.007	0.758	0.784
9	11.01	0.069	14.17	0.007	0.748	0.772
10	11.03	0.070	14.39	0.008	0.731	0.753

Table 3.9 Forecast performance of ARFIMA(7,d,0) for whole year for daily flows at TNH

Lead time (days)	MAE	MAPE	RMSE	MSRE	*CE*	r^2	*SACE*
1	24	0.038	45.57	0.003	**0.989**	0.989	**0.980**
2	42.31	0.065	79.59	0.009	**0.967**	0.967	**0.939**
3	54.91	0.082	103.78	0.014	**0.944**	0.944	**0.896**
4	64.23	0.095	122.19	0.019	**0.922**	0.922	**0.856**
5	72.66	0.107	137.86	0.023	**0.901**	0.901	**0.817**
6	81.77	0.119	152.52	0.028	**0.879**	0.879	0.776
7	90.94	0.132	166.95	0.034	**0.855**	0.855	0.732
8	98.96	0.142	180.17	0.039	**0.831**	0.831	0.687
9	105.79	0.151	191.62	0.044	**0.809**	0.809	0.646
10	111.36	0.158	202.18	0.049	0.788	0.788	0.606

Table 3.10 Forecast performance of ARFIMA(7,d,0) for spring for daily flows at TNH

Lead time (days)	MAE	MAPE	RMSE	MSRE	*CE*	r^2
1	17.7	0.041	30.64	0.003	**0.975**	0.975
2	31.51	0.072	53.94	0.01	**0.922**	0.922
3	39.46	0.089	69.5	0.015	**0.871**	0.873
4	45.89	0.103	80.35	0.019	**0.83**	0.832
5	51.37	0.113	89.57	0.023	0.791	0.794
6	57.8	0.125	100.68	0.027	0.743	0.747
7	66.99	0.139	118	0.033	0.673	0.677
8	75.72	0.152	133.67	0.039	0.612	0.617
9	83.15	0.162	147.95	0.045	0.561	0.568
10	90.41	0.173	159.7	0.05	0.52	0.529

Table 3.11 Forecast performance of ARFIMA(7,d,0) for summer for daily flows at TNH

Lead time (days)	MAE	MAPE	RMSE	MSRE	*CE*	r^2
1	55.91	0.057	80.76	0.006	**0.974**	0.975
2	100.46	0.103	141.04	0.018	**0.922**	0.922
3	132.5	0.135	184.36	0.03	**0.866**	0.866
4	154.85	0.159	217.16	0.041	**0.814**	0.815
5	175.22	0.181	244.46	0.053	0.765	0.766
6	197.41	0.205	269.22	0.065	0.716	0.717
7	218.23	0.226	291.26	0.077	0.668	0.670
8	236.06	0.246	311.55	0.09	0.621	0.623
9	251.25	0.263	328.74	0.102	0.578	0.580
10	262.92	0.277	345.2	0.115	0.535	0.537

Table 3.12 Forecast performance of ARFIMA(7,d,0) for autumn for daily flows at TNH

Lead time (days)	MAE	MAPE	RMSE	MSRE	*CE*	r^2
1	16.66	0.024	26.58	0.001	**0.991**	0.991
2	28.72	0.042	46.45	0.004	**0.973**	0.973
3	37.54	0.054	60.5	0.006	**0.954**	0.955
4	45.15	0.064	72.74	0.008	**0.934**	0.935
5	52.02	0.075	85.06	0.011	**0.909**	0.911
6	59.15	0.085	96.15	0.014	**0.883**	0.887
7	64.93	0.094	106.46	0.017	**0.856**	0.862
8	69.81	0.101	115.41	0.019	**0.831**	0.839
9	74.22	0.107	123.13	0.022	**0.807**	0.819
10	77.26	0.112	129.94	0.024	0.784	0.800

Table 3.13 Forecast performance of ARFIMA(7,d,0) for winter for daily flows at TNH

Lead time (days)	MAE	MAPE	RMSE	MSRE	*CE*	r^2
1	5.02	0.03	7.22	0.002	**0.968**	0.969
2	7.13	0.043	10.15	0.003	**0.930**	0.933
3	8.27	0.05	11.72	0.005	**0.896**	0.902
4	8.73	0.053	12.68	0.005	**0.864**	0.875
5	9.42	0.057	13.35	0.006	**0.833**	0.849
6	9.83	0.06	13.7	0.007	**0.807**	0.827
7	10.29	0.064	13.72	0.007	0.790	0.812
8	10.67	0.066	13.88	0.007	0.770	0.792
9	10.77	0.067	13.9	0.007	0.758	0.778
10	10.8	0.067	14.17	0.007	0.740	0.758

Table 3.14 Forecast performance of PAR for whole year for daily flows at TNH

Lead time (days)	MAE	MAPE	RMSE	MSRE	*CE*	r^2	*SACE*
1	23.35	0.037	45.38	0.003	***0.989***	0.989	***0.980***
2	41.99	0.065	80.68	0.010	***0.966***	0.966	***0.937***
3	54.97	0.082	104.40	0.014	***0.943***	0.944	***0.895***
4	65.02	0.095	125.05	0.019	***0.919***	0.921	***0.849***
5	75.10	0.108	143.60	0.025	***0.893***	0.896	***0.801***
6	84.97	0.120	161.28	0.031	***0.865***	0.870	0.750
7	94.25	0.133	178.93	0.038	***0.834***	0.843	0.692
8	102.95	0.144	195.47	0.045	***0.802***	0.815	0.632
9	110.37	0.154	209.81	0.052	0.771	0.789	0.576
10	116.69	0.163	223.80	0.059	0.740	0.764	0.518

Table 3.15 Forecast performance of PAR for spring for daily flows at TNH

Lead time (days)	MAE	MAPE	RMSE	MSRE	*CE*	r^2
1	17.40	0.041	30.00	0.003	***0.976***	0.976
2	31.55	0.073	54.13	0.010	***0.921***	0.922
3	41.03	0.093	71.15	0.016	***0.865***	0.873
4	48.18	0.108	83.94	0.022	***0.814***	0.829
5	54.06	0.119	94.99	0.027	0.765	0.787
6	61.35	0.132	108.59	0.033	0.701	0.733
7	72.21	0.15	128.38	0.041	0.612	0.652
8	83.37	0.167	146.92	0.050	0.531	0.579
9	91.96	0.18	164.15	0.059	0.459	0.517
10	99.44	0.192	178.56	0.067	0.400	0.467

Table 3.16 Forecast performance of PAR for summer for daily flows at TNH

Lead time (days)	MAE	MAPE	RMSE	MSRE	*CE*	r^2
1	55.26	0.056	80.78	0.006	***0.974***	0.974
2	100.84	0.103	143.31	0.018	***0.919***	0.919
3	131.47	0.134	184.69	0.030	***0.865***	0.869
4	155.56	0.158	221.06	0.043	***0.807***	0.814
5	180.91	0.182	253.17	0.056	0.748	0.761
6	205.91	0.208	283.02	0.071	0.686	0.707
7	225.56	0.228	310.59	0.086	0.622	0.656
8	243.65	0.247	336.61	0.102	0.557	0.607
9	259.84	0.266	358.22	0.118	0.499	0.566
10	273.37	0.283	380.41	0.136	0.436	0.524

Table 3.17 Forecast performance of PAR for autumn for daily flows at TNH

Lead time (days)	MAE	MAPE	RMSE	MSRE	*CE*	r^2
1	15.15	0.023	26.01	0.001	***0.992***	0.992
2	27.10	0.04	46.14	0.004	***0.974***	0.974
3	37.58	0.052	61.73	0.005	***0.952***	0.954
4	45.37	0.061	75.88	0.007	***0.928***	0.931
5	53.35	0.072	90.71	0.010	***0.896***	0.902
6	59.61	0.081	103.79	0.013	***0.864***	0.873
7	65.25	0.088	115.82	0.016	***0.830***	0.844
8	70.12	0.094	126.41	0.018	0.797	0.816
9	74.57	0.100	136.39	0.02	0.763	0.788
10	78.28	0.105	145.48	0.023	0.729	0.761

Table 3.18 Forecast performance of PAR for winter for daily flows at TNH

Lead time (days)	MAE	MAPE	RMSE	MSRE	*CE*	r^2
1	4.85	0.029	7.01	0.002	***0.970***	0.970
2	7.06	0.043	12.77	0.007	***0.889***	0.896
3	7.75	0.046	10.83	0.004	***0.911***	0.916
4	8.31	0.050	11.89	0.005	***0.880***	0.891
5	8.99	0.054	12.69	0.005	***0.849***	0.867
6	9.57	0.058	13.33	0.006	***0.817***	0.842
7	10.07	0.061	13.64	0.006	0.793	0.823
8	10.43	0.063	14.05	0.007	0.764	0.800
9	10.66	0.065	14.27	0.007	0.745	0.783
10	10.90	0.067	14.64	0.007	0.722	0.761

(2) For both daily and monthly flow forecasting, we can observe significant seasonal variation in forecast accuracy. For daily flow forecasting, forecast accuracy for autumn is much higher than other seasons, we can give satisfying forecasts for up to 8 days with ARMA model, 9 days with the ARFIMA model, according to the values of CE. For summer, the forecast accuracy is the lowest, we can give only 4 days good forecast. For monthly flow, forecast accuracy in winter half year is generally much higher than that in summer half year. Such a variation in forecast accuracy is closely related to the seasonal variation in autocorrelation structure, as we have discussed in Chapter 2.

(3) The long-memory ARFIMA(7,d,0) model performs slightly better than short memory ARMA(20,1) model. That gives us another evidence of the presence of long-memory in

addition to the evidence given by the statistical tests. The reseason of the improvement in forecasting accuracy may be partly seen from the perspective of how the models capture the sample ACF structure. The theoretical ACF of ARMA(20,1) model, ARFIMA(7,d,0) and the sample ACF of daily flows at TNH are plotted in Figure 3.12. It is shown that, while the theoretical ACF of ARMA(20,1) model decays much faster than the sample ACF of daily flows at TNH, the theoretical ACF of ARFIMA(7,d,0) fits the sample ACF better, although it decays much slower than the sample ACF at long lags.

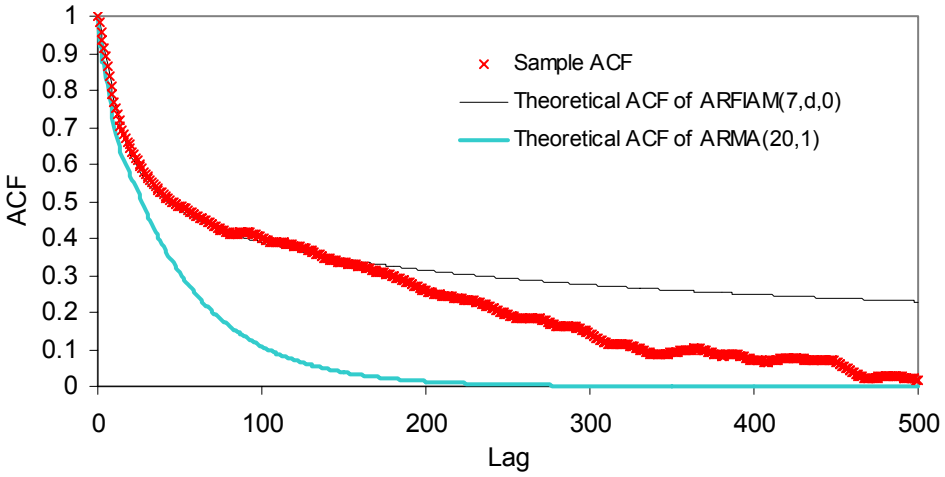


Figure 3.12 Sample ACF of daily flows at TNH and theoretical ACF of fitted ARMA(20,1) and ARFIMA(7,d,0) model

It should be noticed that ARMA models of very high autoregressive orders, e.g., an ARMA(150,0) model, could give ACF structures even closer to the sample ACF then the ARFIMA model, as shown in Figure 3.13. Thus, it is possible that the long-memory of daily streamflow process could be described with a very high order AR process. In fact, there are some attempts to describe the long-memory process with ARMA model (e.g., Tiao and Tsay, 1994; Basak et al., 2001; Man, 2003). Ideally, if the true structure of an ARFIMA(p,d,q) process can be correctly identified, then using the ARFIMA model will produce the best forecasts. However, in practice, as pointed out in Crato and Ray (1996), the low success rate in the selection of the right ARFIMA model, along with the large variance in the estimation of the parameters, make the use of simple ARMA model appealing for forecasting purposes.

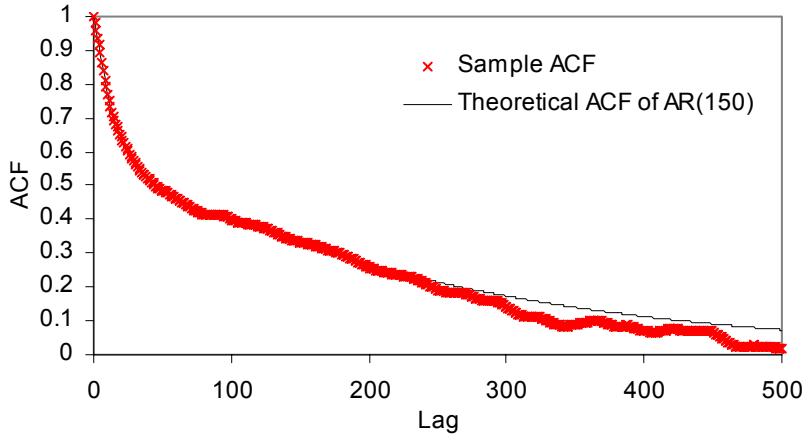


Figure 3.13 Sample ACF of daily flows at TNH and theoretical ACF of fitted AR(150) model

3.5 Constructing Prediction Intervals for Streamflow Forecasts

3.5.1 Introduction of Constructing Prediction Intervals

Forecasts are often expressed as single numbers, called *point forecasts*. The vast majority of research in hydrological forecasting, as well as most operational hydrological forecasting systems, centers around producing and evaluating point forecasts. Point forecasts are of course of first-order importance. However, point forecasts give no information about predictive uncertainty, which is of vital importance in planning and decision-making. *Interval forecasts* are important to supplement point forecasts, especially for medium- and long-range forecasting. *Interval forecasts* usually consists of an upper and a lower limit (prediction interval, PI) between which the future value is expected to lie with a prescribed probability, or a probability distribution function of the predictand (the variable to be forecasted). Hirsch (1981) stated that the presentation of a single "most likely" hydrograph is of no value in long-range hydrological forecasting.

A variety of approaches to computing PIs are available (e.g., Chatfield, 1993). However, no generally accepted method exists for calculating PIs except for forecasts calculated conditional on a fitted probability model, for which the variance of forecast errors can be readily evaluated (Chatfield, 2001). In the hydrological community, many efforts have been made on evaluating the forecast uncertainty, which is essentially equivalent to constructing PIs, in hydrological modeling in the last two decades or so (e.g., Kitanidis and Bras, 1980; Beven and Binley, 1992; Freer et al., 1996; Kuczera and Parent, 1998; Krzysztofowicz, 1999). As mentioned in Section 1.2.2, five major types of approaches have been proposed and used for hydrological forecasting so far. In this study, the approaches, which are based the residuals and independent of any assumption, will be applied to build PIs for forecasts from the univariate ARMA models for monthly streamflow forecasts.

3.5.2 Methodology

Although theoretical formulae are available for computing PIs for ARMA-type time-series model (Box and Jenkins, 1976), it is well-known that streamflow processes often have heavy tails (e.g., Anderson and Meerschaert, 1998), therefore, theoretical formulae which assume normality are not applicable. In contrast with theoretical method, residual based PI estimation methods do not require any distributional assumptions. Two methods will be used in this study to estimate PIs based on residuals from ARMA models, i.e., empirical method and bootstrap-based method. Empirical method constructs PIs relying on the properties of the observed distribution of residuals (rather than on an assumption that the model is true). Bootstrap-based method samples from the empirical distribution of the residuals from fitted models to construct a sequence of possible future values, and evaluates PIs at different horizons by simply finding the interval within which the required percentage of resampled future values lies.

3.5.2.1 *Empirical PI construction*

Let $\{x_t\}$ be a sequence of n observed streamflow series. The empirical PI estimation for k-step ahead prediction proceeds as follows:

Step 1. Log-transform the streamflow series, and then deseasonalize the log-transformed series by subtracting the monthly mean values and dividing by the monthly standard deviations of the log-transformed series.

Step 2. Fit AR(*p*) model to the transformed series, in the form of $\phi(B)x_t = \varepsilon_t$, where $\phi(B)$ represents the ordinary autoregressive components. Compute the *k*-step ahead fitted error (residuals):

$$\varepsilon_{t+k} = x_{t+k} - \sum_{j=1}^{p} \phi_j x_{t+k-j}, t = p+1,...,n-k . \quad (3.17)$$

Notice that, when *k-j* ≥ 1, calculated value (with $x_t = \sum_{j=1}^{p} \phi_j x_{t-j}$), instead of the observed value, will be used for x_{t+k-j} in the Equation (3.17).

Step 3. Define the empirical distribution function $F_{\varepsilon,k}$ of the residuals ε_{t+k}:

$$F_{\varepsilon,k}(x) = \frac{1}{n-p} \sum_{t=p+1}^{n} 1_{\{\varepsilon_{t+k} \le x\}} \quad (3.18)$$

Step 4. Obtain the upper bound of *k*-step ahead future value by expression

$$x_{n+k}^{U} = \sum_{j=1}^{p} \phi_j x_{n+k-j} + \varepsilon_{n+k}^{U}, \quad (3.19)$$

and the lower bound by

$$x_{n+k}^{L} = \sum_{j=1}^{p} \phi_j x_{n+k-j} + \varepsilon_{n+k}^{L} \quad (3.20)$$

where, ε_{n+k}^{U} and ε_{n+k}^{L} are the upper and lower *p*/2-th empirical quantile drawn from $F_{\varepsilon,k}$ according to the nominal coverage level (1 - *p*). Notice that, as in step 2, when *k-j* ≥ 1, calculated value will be used for x_{t+k-j} in the Equation (3.19) as well as in the Equation (3.20).

Step 5. Inversely transform the upper and lower bounds to their original scale.

3.5.2.2 *Bootstrap PI construction*

Bootstrap method is a distribution-free, but computationally intensive approach. There are several bootstrap alternatives in the literature to construct prediction intervals for AR processes (e.g., Stine, 1987; Thombs and Schucany, 1990). In this study, we use the method recently proposed by Pascual et al. (2004), which has the advantage over other bootstrap methods previously proposed for autoregressive integrated processes that variability due to parameter estimation can be incorporated into prediction intervals without requiring the backward representation of the process. Consequently, the procedure is very flexible and can be extended to processes even if their backward representation is not available. Furthermore, its implementation is very simple.

The steps for obtaining bootstrap prediction intervals for monthly streamflow processes are as follows:

Step 1. Transform the series in the same way as Step 1 in Section 3.5.2.1.

Step 2. Fit AR(p) model to the transformed series, in the form of $\phi(B)x_t = \varepsilon_t$. Compute the one-step fitted error (residuals) ε_t as in Equation (3.17), where k=1.

Step 3. Let F_ε be the empirical distribution function of the centered and rescaled residuals by the factor $[(n-p)/(n-2p)]^{0.5}$.

Step 4. From a set of p initial values, generate a bootstrap series from

$$x_t^* = \sum_{j=1}^{p} \phi_j x_{t-j}^* + \varepsilon_t^* \tag{3.21}$$

where ε_t^* are sampled randomly from F_ε.

Step 5. Use the generated bootstrap series to re-estimate the original model, and obtain one bootstrap draw of the autoregressive coefficients $\phi^* = (\phi_1^*, \phi_2^*, \ldots, \phi_p^*)$.

Step 6. Generate a bootstrap future value through the recursion of the AR model with the bootstrap parameters

$$x_{n+k}^* = \sum_{j=1}^{p} \phi_j^* x_{n+k-j}^* + \varepsilon_{n+k}^* \tag{3.22}$$

with ε_t^* a random draw from F_ε; $x_{n+h}^* = x_{n+h}$ for $h \leqslant 0$.

Step 7. Repeat the last three steps B times and then go to step 8.

Step 8. The endpoints of the prediction interval are given by quantiles of G_B^*, the bootstrap distribution function of x_{n+k}^*.

Step 9. Inversely transform the upper and lower bounds to their original scale.

3.5.3 Application of the Two PI Construction Methods

3.5.3.1 *Split the data set and fit AR models*

The monthly streamflow series of four rivers at four sites (the Yellow River at TNH, the Rhine River at Lobith, the Ocmulgee River at Macon and the Umpqua River near Elkton) will be used in this study. It has been analysed in Chapter 2 (Section 2.2 and 2.3), that the four monthly streamflow series are basically stationary without significant trending behaviour. Hence, no pre-processing (e.g., detrending or differencing) is considered. All the streamflow series are transformed with log-transformation and deseasonalization. Then we split each series into two parts, with the first part for fitting ARMA models and getting the residuals and the second part for constructing prediction intervals with the ARMA models fitted to the first part. Chernick (1999, pp 150-151) suggested that the sample size for bootstrap sampling should be larger than 30. To meet the requirement, we keep the size of first part larger than 360 (i.e., 30 years for monthly series), so that when generating bootstrap samples for each individual month, we have a sample size larger than 30. The orders of the AR models fitted to the first parts of each transformed flow series are chosen according to AIC. The details of the data size, the partition of the data set and the order of AR models for each series are listed in Table 3.19.

Table 3.19 Data size, the partition of the data set and the order of AR models

River	Data period	Part 1	Part 2	AR(p)
Yellow	1956-2000	420	10	4
Rhine	1901-1996	600	552	4
Ocmulgee	1929-2001	444	432	3
Umpqua	1906-2001	600	552	5

In this study, only one-step ahead forecast is considered, and the empirical distribution function of residuals is defined according to fitted error from the AR models fitted to the first part. But in the practice, all the fitted residuals up to the forecast point could be used.

3.5.3.2 *Construct PIs according to the overall empirical distribution function of residuals*

The 95% probability is commonly applied to build prediction interval. However, for a 95% probability, PIs may become so embarrassingly wide that they are of little practical use other than to indicate the high degree of future uncertainty. Granger (1996) suggests using 50%, rather than 95%, PIs because this gives intervals that are better calibrated in regard to their robustness to outliers and to departures from model assumptions. Such intervals will be narrower but imply that a future value has only a 50% chance of lying inside the interval. Therefore, Chatfield (2001) suggests using 90% or 80% prediction interval. In this study, we build the 95%, 90%, 80% and 50% PIs for one-step ahead monthly average discharge prediction with the methods described in Section 3.5.2. For the bootstrap procedure, $B = 1000$.

To evaluate the performance of PI construction methods, the following measures are used: the actual PI coverage, the average PI length, the proportions of observations lying out to the left and to the right of the interval. A good PI construction method should have coverage close to the nominal coverage, a small interval length, and balanced proportions of observations below and above the interval. Table 3.20 reports the results for the four monthly streamflow processes, comparing empirical PIs with bootstrap-based PIs. It is shown that both methods give reasonable performance in terms of interval coverage, and there is no significant bias of the interval, namely, the numbers of observed values falling to the left and to the right are mostly similar. In terms of the interval length, the empirical method outperforms the bootstrap method because the empirical method has generally shorter interval length.

Since the streamflow processes exhibit strong seasonality as shown in Figure 2.9 in Chapter 2, to inspect possible impacts of the presence of seasonality on the performance of empirical method and bootstrap method, we check the PIs month by month. Table 3.21 lists PI construction results for these flow series with nominal coverage of 80% month by month (to save space, results for other three nominal coverage levels are not listed here).

From Table 3.21, we find that there is a systematic bias that for low-flow months the coverage of PIs is larger than the nominal coverage, whereas for high-flow months the coverage of PIs is smaller than the nominal coverage, especially for the Yellow River and the Umpqua River. That indicates that for low-flow months the PIs are over-estimated, and for high-flow months the PIs are under-estimated. We examine the standard deviation of the residuals of the months over the year, plotted in Figure 3.14. It is shown that for the Yellow River and Umpqua River, there is obvious seasonal variation in standard deviation of the residuals. Comparing Figure 2.9 and Figure 3.14, we observe that there is a general tendency that the months with high flow also have high residual standard deviation. Therefore, when we use the overall empirical distribution function to construct the PIs, the coverage between the upper and lower $p/2$-th empirical quantile for the nominal coverage $(1-p)$ is too large for low-flow months, and too small for high flow months, which causes the systematic bias of PI coverage shown in Table 3.21.

Table 3.20 Prediction intervals for monthly flow series

Nominal Cov.	Measure	Yellow		Rhine		Ocmulgee		Umpqua	
		Emp.	Boot.	Emp.	Boot.	Emp.	Boot.	Emp.	Boot.
95%	Cov.	0.925	0.925	0.964	0.940	0.938	0.910	0.964	0.944
	Above Cov.	0.033	0.033	0.016	0.034	0.016	0.021	0.016	0.027
	Below Cov.	0.042	0.042	0.020	0.025	0.046	0.069	0.020	0.029
	Average Len.	623	704	2828	2798	171	190	437	430
90%	Cov.	0.875	0.892	0.879	0.888	0.882	0.856	0.908	0.902
	Above Cov.	0.050	0.042	0.067	0.054	0.046	0.042	0.027	0.049
	Below Cov.	0.075	0.067	0.054	0.058	0.072	0.102	0.065	0.049
	Average Len.	473	551	1996	2266	118	134	350	355
80%	Cov.	0.742	0.742	0.784	0.772	0.762	0.736	0.806	0.821
	Above Cov.	0.117	0.092	0.120	0.123	0.095	0.118	0.087	0.092
	Below Cov.	0.142	0.167	0.096	0.105	0.144	0.146	0.107	0.087
	Average Len.	346	438	1500	1582	89	91	247	276
50%	Cov.	0.500	0.500	0.527	0.518	0.491	0.451	0.504	0.489
	Above Cov.	0.200	0.192	0.252	0.248	0.255	0.269	0.268	0.281
	Below Cov.	0.300	0.308	0.221	0.234	0.255	0.280	0.228	0.230
	Average Len.	171	225	851	867	43	47	110	124

Note: Cov. - coverage; Len. - length; Emp. - empirical method; Boot. - bootstrap method.

Table 3.21 Month by month PI coverages for monthly flow with nominal coverage of 80%

(Without considering the seasonal variations in variance of the residuals)

Month	Yellow		Rhine		Ocmulgee		Umpqua	
	Emp.	Boot.	Emp.	Boot.	Emp.	Boot.	Emp.	Boot.
1	0.900	0.700	0.717	0.783	0.833	0.944	0.717	0.739
2	0.900	0.700	0.696	0.804	0.722	0.722	0.652	0.739
3	1.000	0.500	0.696	0.739	0.722	0.722	0.739	0.761
4	0.200	0.400	0.739	0.761	0.667	0.639	0.761	0.891
5	0.800	0.800	0.848	0.804	0.722	0.694	0.783	0.870
6	0.500	0.500	0.826	0.717	0.861	0.889	0.891	0.848
7	0.500	0.500	0.761	0.739	0.722	0.528	0.957	0.783
8	0.500	0.900	0.913	0.848	0.750	0.500	0.957	0.870
9	0.600	1.000	0.891	0.848	0.722	0.750	0.913	0.870
10	1.000	1.000	0.804	0.739	0.806	0.778	0.848	0.891
11	1.000	0.900	0.761	0.717	0.861	0.833	0.696	0.826
12	1.000	1.000	0.761	0.761	0.750	0.833	0.761	0.761

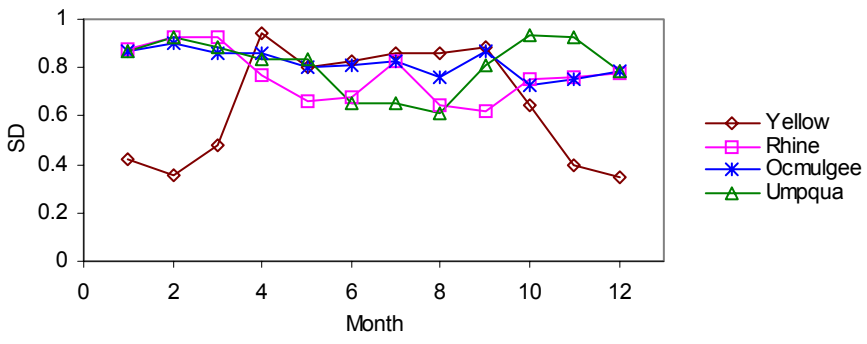


Figure 3.14 Seasonal variations in standard deviation (SD) of the residuals from AR models

3.5.3.3 *Construct PIs according to the seasonal empirical distribution function of residuals*

To take the season-dependant variance of residuals into account, we define the seasonal empirical distribution function $F_{\varepsilon}^{(m)}$ for the residuals of each month m. Then choose the upper and lower p/2-th empirical quantile for the nominal coverage (1-p) from $F_{\varepsilon}^{(m)}$ for empirical PI construction method, and generate bootstrapping samples from $F_{\varepsilon}^{(m)}$ for the bootstrap method, so that we construct the PIs considering the seasonal variation in variance of the residuals. Table 3.22 lists the results for the streamflow processes with the seasonal variation in variance considered.

Table 3.22 PIs considering the seasonal variation in variance of the residuals

Nominal Cov.	Measure	Yellow		Rhine		Ocmulgee		Umpqua	
		Emp.	Boot.	Emp.	Boot.	Emp.	Boot.	Emp.	Boot.
95%	Cov.	0.917	0.933	0.953	0.937	0.905	0.905	0.957	0.944
	Above Cov.	0.042	0.025	0.036	0.033	0.039	0.023	0.027	0.025
	Below Cov.	0.042	0.042	0.011	0.031	0.056	0.072	0.016	0.031
	Average Len.	597	710	2768	2807	147	189	424	432
90%	Cov.	0.875	0.892	0.899	0.886	0.824	0.859	0.911	0.911
	Above Cov.	0.067	0.042	0.058	0.058	0.079	0.042	0.049	0.043
	Below Cov.	0.058	0.067	0.043	0.056	0.097	0.100	0.040	0.045
	Average Len.	513	553	2316	2276	110	132	357	355
80%	Cov.	0.775	0.750	0.768	0.779	0.704	0.745	0.804	0.810
	Above Cov.	0.092	0.092	0.136	0.120	0.139	0.109	0.107	0.096
	Below Cov.	0.133	0.158	0.096	0.101	0.157	0.146	0.089	0.094
	Average Len.	439	439	1507	1584	82	92	269	276
50%	Cov.	0.483	0.483	0.529	0.520	0.458	0.447	0.502	0.507
	Above Cov.	0.217	0.200	0.255	0.246	0.278	0.271	0.272	0.264
	Below Cov.	0.300	0.317	0.216	0.234	0.264	0.282	0.226	0.228
	Average Len.	225	228	883	868	47	47	123	124

Note: Cov. - coverage; Len. - length; Emp. - empirical method; Boot. - bootstrap method.

Comparing Table 3.22 with Table 3.20, we observe that no significant improvement is achieved in terms of PI coverages after considering the seasonal variation in variance. The values of coverage length are even bigger than those without considering seasonal variation in variance. However, when we examine the PIs month by month, as shown in Table 3.23 for nominal coverage 80%, it is clear that the systematic bias shown in Table 3.21 disappears, and maximum errors between the nominal coverage and actual coverage are reduced, especially for the empirical method. For example, for the Umpqua River, the range of the difference between the nominal coverage and actual coverage with the empirical method is –0.148 (= 0.652 – 0.8) to 0.157 (=0.957 – 0.8) before considering seasonal standard deviation in residuals, and shrinks to –0.039 (= 0.761- 0.8) to 0.07 (=0.870 – 0.8) after considering seasonal standard deviation. Therefore, by considering the seasonal variation in variance of the residuals, more accurate PI construction is obtained. Figure 3.15 plots the observed discharges during 1991 to 2000 of the Yellow River and their upper and lower prediction bounds for 80% nominal coverage constructed using the empirical method considering the seasonal empirical distribution function of residuals. At the same time, we should notice that for streamflow series, like the monthly streamflow of the Ocmulgee, which exhibits no significant seasonal variation in variance in the residuals, to construct PI according to the seasonal empirical distribution function of residuals may deteriorate the PI construction

results, because the seasonal empirical distribution function of residuals are defined with a much smaller sample size (for monthly streamflow series, the size is reduced to 1/12), which may cause error for PI construction.

Table 3.23 Month by month PI coverages for monthly flow with nominal coverage of 80%

(Considering the seasonal variations in variance of the residuals)

Month	Yellow		Rhine		Ocmulgee		Umpqua	
	Emp.	Boot.	Emp.	Boot.	Emp.	Boot.	Emp.	Boot.
1	0.700	0.700	0.783	0.783	0.944	0.944	0.739	0.739
2	0.700	0.700	0.739	0.826	0.722	0.750	0.739	0.739
3	0.800	0.600	0.783	0.804	0.694	0.722	0.783	0.783
4	0.500	0.400	0.783	0.761	0.611	0.667	0.848	0.870
5	0.800	0.800	0.804	0.804	0.694	0.694	0.870	0.870
6	0.400	0.500	0.717	0.717	0.833	0.917	0.848	0.848
7	0.600	0.500	0.739	0.761	0.528	0.528	0.783	0.783
8	0.900	0.900	0.848	0.848	0.500	0.500	0.761	0.783
9	1.000	1.000	0.826	0.826	0.694	0.722	0.848	0.848
10	1.000	1.000	0.717	0.717	0.778	0.833	0.870	0.891
11	0.900	0.900	0.717	0.717	0.667	0.833	0.804	0.826
12	1.000	1.000	0.761	0.783	0.778	0.833	0.761	0.739

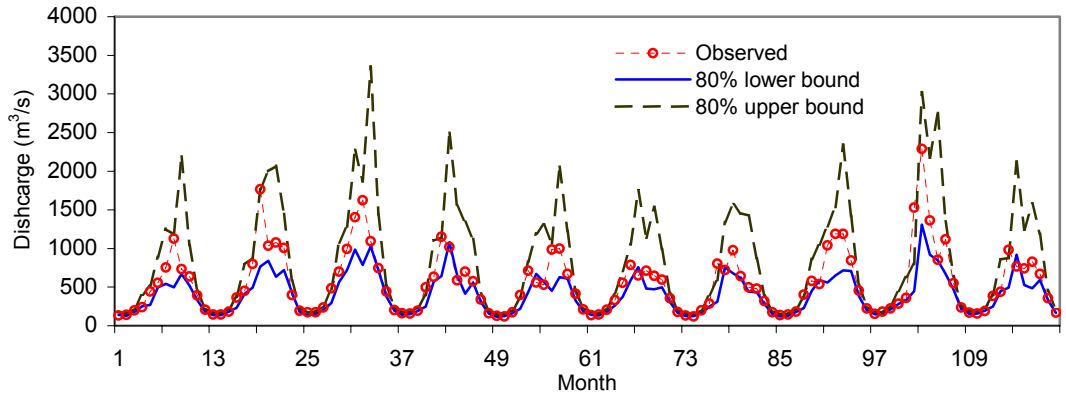


Figure 3.15 Observed monthly discharges (1991-2000) of the Yellow River and their 80% upper and lower prediction bounds.

3.6 Predictability of Streamflow Processes

3.6.1 Predictability and Its Definition

Predictability is an important aspect of the dynamics of hydrological processes. Some processes are inherently easy to forecast, and others are more difficult. Basic understanding of predictability is the basis for building a meaningful and robust prediction system. The predictability of hydrological processes has not attracted much attention by the hydrology community until recent several years. Some examples of the studies on predictability include those based on the multiple explanatory variables (e.g., Maurer et al., 2003; Maurer et al., 2004) and those based on univariate hydrological time series (e.g., Wang et al., 2004).

3.6.1.1 *Sources of predictability*

Common sources of the predictability of a streamflow process include (Wang et al., 2004): (1) the temporal dependence (either short-rang or long-range) on the initial conditions of the watershed system; (2) the relations between the watershed system and its boundary conditions, including local factors (e.g., rainfall, temperature) as well as remote factors (e.g., sea surface temperature); and (3) the accuracy of the forecasting model (or method).

Whatever type of predictability to be estimated, we should use a forecasting model which relates the sources of predictability to the future values of a hydrological time series. The predictability estimated on the basis of a forecasting model could be defined as *model predictability*. The essence of estimating the model predictability is to measure the accuracy of the forecasting model. Depending on what information is involved in the model, we may have a univariate model, which only take the temporal dependence of the observed time series, or a more complex multivariate model, either process-driven (i.e., physically-based) or data-driven. In consequence, two approaches may be distinguished in estimating the predictability of streamflow processes: univariate approach and multivariate approach. The former one measures the predictability based on univariate time series analysis techniques (e.g., Wang et al., 2004), whereas the latter one estimates the predictability based on the knowledge of the rainfall-runoff generation mechanism and/or the tele-connections between global environment and streamflow processes of interest (e.g., Maurer et al., 2003, 2004). A multivariate approach may give us an insightful understanding of the watershed system. However, to estimate predictability with a multivariate approach need good knowledge of physical mechanisms underlying the streamflow process of interest and we need a suitable model which is assumed to be able to capture the mechanism well; on the other hand, the knowledge of different streamflow processes varies, and the skills of modeling are highly user-depend. Therefore, the predictabilities estimated with such an approach for different streamflow processes are normally not comparable. Furthermore, dominant factors of a hydrological system may change with the change of temporal and spatial scales, and the explanatory variables used in estimating the predictability of streamflow processes may have a problem of predictability themselves, which would make the problem of estimating the predictability of streamflow processes more complicated. For example, weather processes, which are fundamentally the driven forces of streamflow processes, are only predictable for at most a few weeks because of its inherently nonlinear and chaotic nature. In contrast, the univariate approach can be easily applied to different hydrological time series and the results are comparable among different processes.

Because of the lack of accurate understanding of the streamflow process, there is always some unknown information that has not been used in any forecasting model. In consequence, we have another type of predictability, the *potential predictability*, which refers to the part of predictability that exists but is beyond of our knowledge so far, and have not been captured by any existing model. With the deepening of our understanding of a process, the potential predictability of the process would turn to be part of model predictability. In the following part of the study, we only consider the model predictability.

3.6.1.2 *Definition of predictability*

To quantify the predictability of a time series, we need some measures. Granger and Newbold (1978) proposed a definition of predictability for covariance stationary series, patterned after the familiar R^2 of linear regression, as the ratio of the variance of the optimal prediction to the variance of the original time series. For calculating the predictability measure, Bhansali (1992)

proposed a procedure for estimating the variance of the mean squared error of prediction. In fact, the predictability measure used by the above-mentioned authors is essentially the same as the coefficient of efficiency (*CE*) which is proposed by Nash and Sutcliffe (1970) and widely used as model performance measure in the hydrology community.

Despite its crudeness and identified weaknesses (Kachroo and Natale, 1992), *CE*, given by formula (3.14), is still one of the most widely used criteria for the assessment of hydrological model performance. While *CE* is a global measure of comparing the predicted value with the overall mean value, it is not efficient enough to evaluate the predictions for those series whose mean values change with seasons, which is almost always the case for hydrological processes. Therefore, a seasonally-adjusted coefficient of efficiency (*SACE*), given by formula (3.16), is also used here for evaluate the model performance for modeling seasonal time series.

Forecasts are always made with some model or method. To measure the predictability of a process, we need a suitable model together with the model performance measure. Unfortunately, the real model underlying a real-world process is rarely known. On the other hand, if different types of models are used to measure the predictability, then the results between different time series will be not comparable. Therefore, we suggest using autoregressive (AR) model to measure the predictability.

With the forecasting model and the model performance measures, we define the predictability as the predictable horizon, i.e., the lead time of prediction, for which the prediction is no better than the mean value for a stationary process or the seasonal mean value for a seasonal process. Theoretically, if the prediction is no better than the mean value for a stationary process or the seasonal mean value for a seasonal process, then $CE \leq 0$ or $SACE \leq 0$. However, our simulation with Gaussian white noise processes shows that, for 500 replications of a Gaussian white noise process of length 4000, the values of *CE* for one-step-ahead forecasts for the last 1000 points using AR models fitted to the first 3000 points may exceed 0.006. Therefore, it is more robust to estimate the predictability at $CE = 0.01$ level instead of at $CE = 0$ level. Besides, for practical purposes, we define the predictability at a given *CE* or *SACE* level as the predictable horizon for which the *CE* or *SACE* of predictions is larger than a given value. In the present study, we also measure the predictability at *CE* (or *SACE*) = 0.1 and 0.5 level.

3.6.2 Predictability of Simulated AR and ARFIMA Processes

Here, we present the results of our simulation study measuring the predictability of AR(1) processes and ARFIMA(1,d,0) processes. We consider seven AR(1) and three ARFIMA(1,d,0) processes. For the AR(1) models, ϕ = 0.2, 0.4, 0.6, 0.8, 0.9, 0.95, and 0.99. For the ARFIMA(1,d,0) models, $\phi = 0$, 0.5, 0.9 and $d = 0.15$, 0.3, 0.4, 0.45, and 0.49. We generate a hundred simulated realizations of size 5,000 for each model. The AR series and the ARFIMA series are produced by the *arima.sim* and *arima.fracdiff.sim* function built in S-Plus version 6. For each realization, we use the first 3000 points to fit an AR model, then make forecasts for the next 1000 points. The forecasting horizons vary from 100 (for the AR model with $\phi = 0.2$) to 1,000 (for the AR model with $\phi = 0.99$) depending on the model.

The estimates of the predictability (reflected by predictable horizon) of these simulated AR(1) processes are presented in Table 3.24. At the same time, the results are plotted in Figure 3.16 with the axis of predictable horizon log-scaled. It is shown that the predictability of AR processes still exhibits an exponential growth with the increase of autoregressive coefficient ϕ

on the log-scaled plot at different CE levels. That means, the predictability of AR processes follows a double-exponential growth with the increase of ϕ.

Table 3.24 Predictability of AR(1) models at *CE* = 0.01, 0.1 and 0.5 levels

Level	ϕ = 0.2	ϕ = 0.4	ϕ = 0.6	ϕ = 0.8	ϕ = 0.9	ϕ = 0.95	ϕ = 0.99
CE =0.01	0.97	1.85	3.71	8.51	17.99	31.74	87.76
	(0.17)	(0.44)	(0.82)	(2.23)	(6.42)	(13.96)	(53.30)
CE =0.1	0.00	0.98	1.83	4.49	9.69	19.26	65.12
	(0.00)	(0.14)	(0.38)	(0.82)	(2.04)	(5.71)	(31.93)
CE =0.5	0.00	0.00	0.00	1.01	2.73	5.91	24.52
	(0.00)	(0.00)	(0.00)	(0.10)	(0.51)	(1.39)	(9.63)

Note: The predictability is reflected by the average predictable horizons of 100 simulations for each AR model. The figures in the brackets are the standard deviations of the estimated predictable horizons.

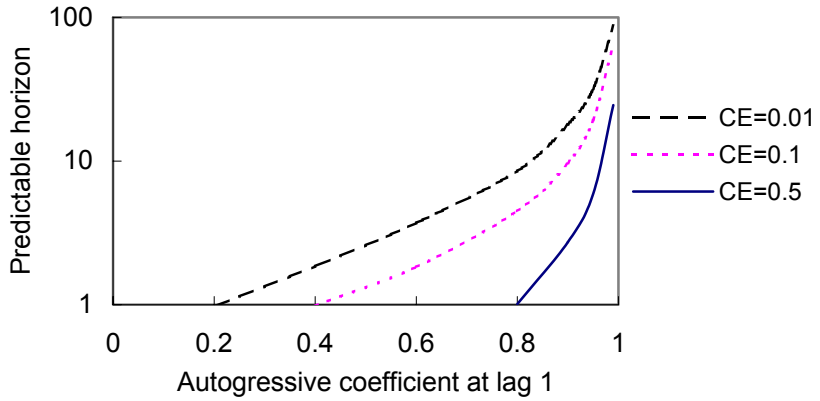


Figure 3.16 Predictability (reflected by predictable horizon) versus ϕ for AR(1) models

When estimating the predictability of ARFIMA(1,d,0) processes, to compare the performance of using AR model versus using ARFIMA model in forecasting ARFIMA processes, the ARFIMA model is also used. Therefore, for each ARFIMA(1,d,0) process, there are two estimates of predictability, one by using the AR model, another by using the ARFIMA model. The estimates of the predictability (reflected by predictable horizon) of the simulated ARFIMA(1,d,0) series at *CE* = 0.5 level and *CE* = 0.01 level given by the AR model are plotted against their theoretical autocorrelations at lag 1, ACF(1), in Figure 3.17 and Figure 3.18, respectively. In the plots, the numbers in the brackets denote the value of autocorrelation coefficient of the AR component, the value of d, and the order of the moving average component in the ARFIMA(1,d,0) model. Comparing the predictability of ARFIMA processes with the predictability of AR(1) processes, we find that:

(1) The growth of the predictability seems to follow an exponential growth at high *CE* levels (e.g., *CE* = 0.5), similar to that for AR(1) processes, but at low *CE* levels (e.g., *CE* = 0.01) the growth of the predictability seems to be not that fast.

(2) At *CE* = 0.5 level, the ARFIMA model is less predictable than their equivalent AR(1) process which has a ϕ equal to the theoretical ACF(1) of the ARFIMA process. That indicates that, long-memory processes do not have longer predictability than their equivalent AR(1) processes at high *CE* levels.

(3) At CE = 0.01 level, when the autoregressive coefficient of the AR component in the ARFIMA model is low (e.g., $\leq$ 0.5), the ARFIMA process is more predictable than the equivalent AR(1) process which has a ϕ equal to the theoretical ACF(1) of the ARFIMA

process, and the ARFIMA process is further more predictable if the predictability is estimated by using an ARFIMA model. That is in agreement with our expectation, because long-range dependence may give long-term predictability which only shows itself at low *CE* levels. But the predictability of ARFIMA(0.9,d,0) processes is less than that of equivalent AR(1) processes. For example, the average predictable horizon of ARFIMA(0.9,0.45,0) at *CE* = 0.01 level is 41.84 steps if the AR model is used, 46.23 steps if the ARFIMA model is used. In contrast, the AR(1) with $\phi = 0.99589$, which is equivalent to the theoretical ACF(1) of the ARFIMA(0.9,0.45) model, has an average predictable horizon of 126.81 steps. Similar results (to save space, not shown here) are found at *CE* = 0.1 level. In summary, at low CE levels, a long-memory ARFIMA process has longer predictability than its equivalent AR(1) process when the autocorrelation coefficient of the AR component in ARFIMA model is not very high (e.g., ≤ 0.5). But, with a high value (e.g., 0.9) of the autocorrelation coefficient of the AR component, the predictability of the long-memory ARFIMA process will be less than that of the equivalent AR(1) process.

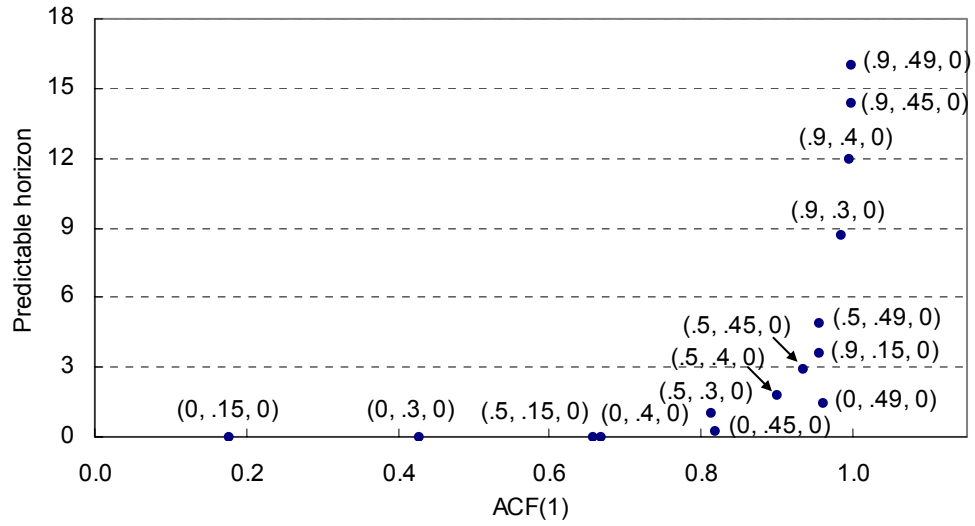


Figure 3.17 Predictability (reflected by predictable horizon) at CE = 0.5 level of the simulated ARFIMA(1,d,0) series versus the theoretical ACF(1) of the ARFIMA(1,d,0) models.

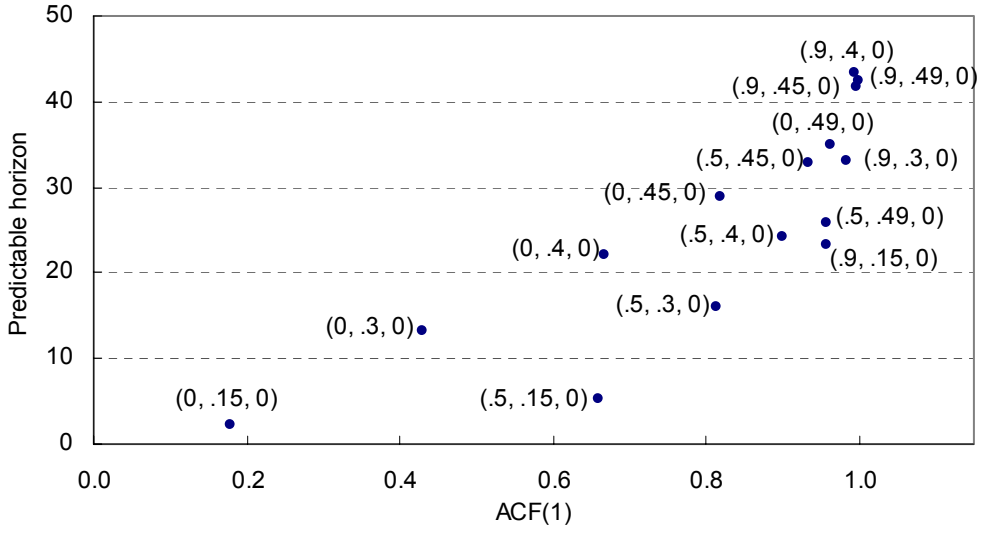


Figure 3.18 Predictability (reflected by predictable horizon) at CE = 0.01 level of the simulated ARFIMA(1,d,0) series versus the theoretical ACF(1) of the ARFIMA(1,d,0) models.

To investigate the relationship between the predictability and the intensity of long-memory, the estimates of the predictability (reflected by predictable horizon) of ARFIMA(1,d,0) models at *CE* = 0.01, 0.1, and 0.5 levels are plotted against the values of *d* in Figure 3.19 (a), (b) and (c). Here, the ARFIMA(0.5,d,0) and ARFIMA(0.9,d,0) stand for the ARFIMA(1,d,0) model with the autoregressive coefficient of 0.5 or 0.9 in their AR components.

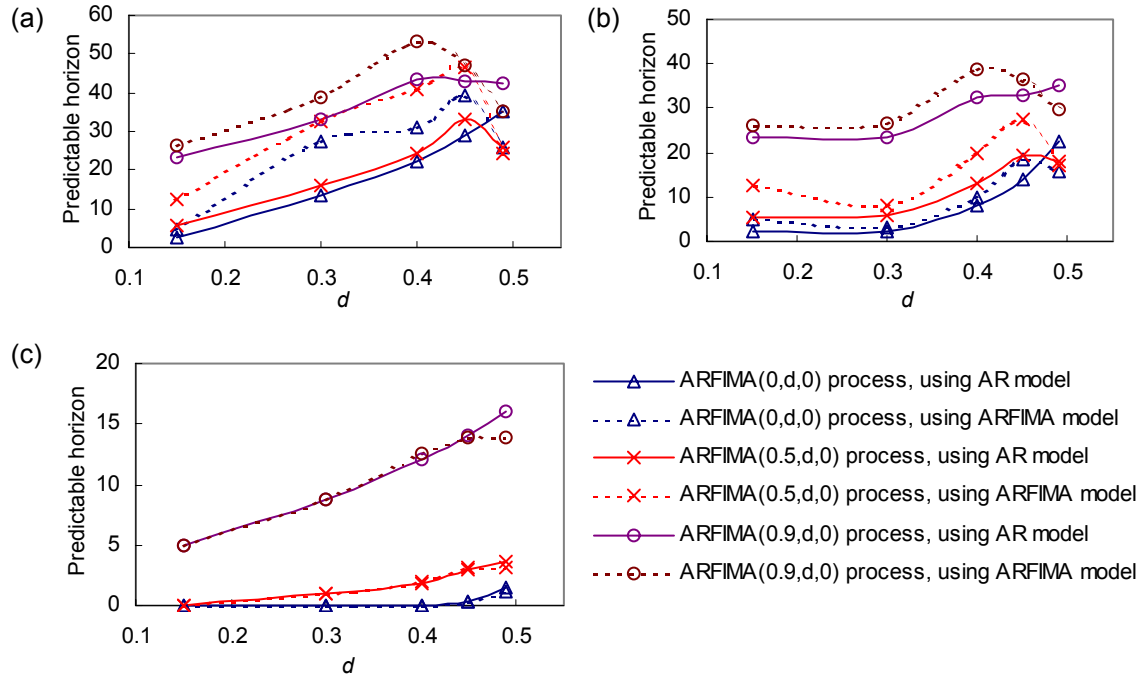


Figure 3.19 Predictability (reflected by predictable horizon) of ARFIMA(1,d,0) models at (a) *CE* = 0.01; (b) CE = 0.1; and (c) CE = 0.5 levels.

By a visual inspection at Figure 3.19, it seems that the relationship between the predictability of the ARFIMA(1,d,0) process and the fractional differencing parameter *d* is far more complicated than that between the predictability of AR(1) models and ϕ. In this experiment, for each series of 5,000 points, the first 3,000 points are used to fit the model, and out-of-sample predictions are made for the next 1,000 points. It is shown that, with data samples of such a finite size:

(1) When *d* is less than 0.45, in most cases (except for the case of ARFIMA(0.9, 0.45, 0) process), the predictability increases with the increase of *d*. And, the increase seems to be of a polynomial growth. When *d* is greater than 0.45, the predictability unexpectedly drops.
(2) The estimates of the predictability of ARFIMA processes using ARFIMA models are better than those using AR models. That indicates that the ARFIMA model gives better predictions than the AR model when the process of interest is indeed a long-memory ARFIMA process.
(3) With the increase of *CE* level, the difference between the estimates of the predictability estimated by using AR model and those estimated by using the ARFIMA model decreases.

3.6.3 Predictability Of Daily Streamflow Processes

3.6.3.1 *Daily streamflow data used*

Daily average discharge series recorded at 31 gauging stations in eight basins all over the world are analyzed in the present study. We generally have the following three rules to select stations in each basin:

(1) The selection of basins covers different geographical and climatic regions;
(2) The drainage area of each station is basically within 5 different watershed scales, namely, $> 10^6$ km^2; $10^6 \sim 10^5$ km^2; $10^5 \sim 10^4$ km^2; $10^4 \sim 10^3$ km^2; $< 10^3$ km^2;
(3) The stations are located in the main river channel of the river if possible. When stations at the main channel are not available, stations at major tributaries are used.

For each station, we select a segment of historical daily streamflow records of mostly 30 years long. However, because of data limitation, the shortest series covers a period of only 14 years. The segments are chosen with following criteria:

(1) The series should be approximately stationary, as least by visual inspection. We have stationarity as our primary data requirement because on one hand, when certain types of nonstationarity are present, many long-memory parameter estimators may fail (Klemes, 1974); on the other hand, the building of AR model used to estimate the predictability has a stationarity assumption.
(2) The data should be recorded as early as possible, so as to limit the influence of human intervention to the minimum.
(3) The temporal spans of different streamflow series at different watershed scales in one basin should be as close as possible, so as to avoid possible impacts of regional low-frequency climatic variations.

The description of selected stations and their corresponding daily streamflow series is listed in Table 3.25.

Table 3.25 Description of selected daily streamflow time series

No.	Basin	Location of gauging stations	Area (km^2)	Latitude	Longitude	Elevation (m)	Period	Average discharge (m^3/s)
Color-1	Colorado	Colorado River At Lees Ferry	289,400	36.865	-111.588	946.8	1922-1951	489.1
Color-2		Colorado River Near Cisco	62,390	38.811	-109.293	1246.6	1923-1952	222.3
Color-3		Colorado River Near Kremmling	6,167	40.037	-106.439	2231.1	1904-1918	52.3
Color-4		Williams Fork Near Parshall	476	40.000	-106.179	2380.2	1904-1924	4.9
Colum-1	Columbia	Columbia River At The Dalles	613,565	45.108	-121.006	0.0	1880-1909	6065.7
Colum-2		Columbia River at Trail	88,100	49.094	-117.698	-	1914-1936	2029.4
Colum-3		Columbia River at Nicholson	6,660	51.244	-116.912	-	1933-1962	107.5
Colum-4		Columbia River Near Fairmont Hot Springs	891	50.324	-115.863	-	1946-1975	11.1
Danu-1	Danube	Danube river at Orsova	576232.	44.700	22.420	44	1901-1930	5711.9
Danu-2		Danube river at Achleiten	76653.	48.582	13.504	288	1901-1930	1427.0
Danu-3		Inn river at Martinsbruck	1945.	46.890	10.470	-	1904-1933	57.8
Fras-1	Fraser	Fraser River at Hope	217,000	49.381	-121.451	-	1913-1942	2648.8
Fras-2		Fraser River at Shelley	32,400	54.011	-122.617	-	1950-1979	825.3
Fras-3		Fraser River at Mcbride	6,890	53.286	-120.113	-	1959-1988	197.3
Fras-4		Canoe River below Kimmel Creek	298	52.728	-119.408	-	1972-1994	14.5
Missi-1	Mississippi	Mississippi River At Vicksburg	2,962,974	32.315	-90.906	14.1	1932-1961	16003.1
Missi-2		Mississippi River at Clinton	221,608	41.781	-90.252	171.5	1874-1903	1477.3
Missi-3		Minnesota River At Mankato	38,574	44.169	-94.000	228.0	1943-1972	94.9
Missi-4		Minnesota River At Ortonville	3,003	45.296	-96.444	291.5	1943-1972	3.4
Misso-1	Missouri	Missouri River at Hermann	1,353,000	38.710	-91.439	146.8	1929-1958	2162.0
Misso-2		Missouri River at Bismarck,	482,776	46.814	-100.821	493.0	1929-1953	604.6
Misso-3		Missouri River at Fort Benton	64,070	47.818	-110.666	796.8	1891-1920	219.7
Misso-4		Madison River near McAllister	5,659	45.490	-111.633	1429.2	1943-1972	50.5
Ohio-1	Ohio	Ohio River At Metropolis	525,500	37.148	-88.741	84.2	1943-1972	7567.5
Ohio-2		Ohio River at Sewickley	50,480	40.549	-80.206	207.3	1943-1972	922.4
Ohio-3		Tygart Valley River At Colfax	3,529	39.435	-80.133	261.0	1940-1969	72.4
Ohio-4		Tygart Valley River Near Dailey	479	38.809	-79.882	591.3	1940-1969	9.2
Rhine-1	Rhine	Rhine at Lobith	160,800	51.840	6.110	8.5	1911-1940	2217.8
Rhine-2		Rhine at Rheinfelden	34,550	47.561	7.799	259.6	1931-1960	1017.3
Rhine-3		Rhine at Domat/Ems	3,229	46.840	9.460	562.0	1911-1940	126.9
Rhine-4		Emme River at Emmenmatt	443	46.960	7.740	-	1915-1944	12.0

3.6.3.1 *Estimation results of the predictability for Daily streamflow series*

Before estimating the predictability of the streamflow processes, we estimate the intensity of the long-memory with Lo's modified *R*/*S* test and GPH test, and estimate the fractional differencing parameter *d* with GPH method and maximum likelihood estimation method implemented in S-Plus (*S-MLE*). To alleviate the impact of seasonality, all the series are deseasonalized by subtracting the daily means and dividing by the daily standard deviations. In Lo's *R*/*S* test, both a fixed lag (i.e., 50) and a lag determined by the data-driven formula (Equation (2.30)) are used. In GPH test, we choose $g(n) = n^{0.5}$ as suggested by Geweke and Porter-Hudak (1983). When using S-MLE to estimate the fractional differencing parameter *d*, the order *p* of AR component in ARFIMA(*p*,*d*,*q*) model is determined by the AIC criteria.

The predictability of each streamflow series is estimated based on in-sample forecasts, and measured according to different *SACE* levels. Namely, for each series, we fit an AR model for the entire series, then use the fitted AR model to make forecasts for the last 11th to the 2nd year (in total, 10 years). The results of long-memory detection and predictability estimation are reported in Table 3.26, which reveal the following:

Table 3.26 Results of long-memory detection and predictability estimation for daily streamflow series

No.	data size	ACF(1)	Lo's R/S test				GPH test		*S-MLE*	Predictable horizon (SACE level)		
			Lag-1	Stat-1	Lag-2	Stat-2	*d*	Stat	*d*	0.01	0.1	0.5
Color-1	10957	0.9738	64	2.9566	50	3.2475	0.5125	7.5412	0.4478	227	98	8
Color-2	10958	0.9627	50	3.4320	50	3.4320	0.4906	7.2192	0.4506	95	18	5
Color-3	5113	0.9431	31	2.1437	50	1.8067	0.4766	5.6613	0.4863	25	17	5
Color-4	7305	0.9549	40	1.1811	50	1.0826	0.4043	5.3169	0.0000	38	34	11
Colum-1	10957	0.9910	132	1.5357	50	2.1519	0.5071	7.4617	0.4615	180	45	15
Colum-2	8401	0.9966	238	1.1342	50	1.8357	0.4673	6.3838	0.4187	204	26	13
Colum-3	10957	0.9778	72	3.1202	50	3.5159	0.3466	5.101	0.4392	26	21	9
Colum-4	10957	0.9676	55	1.8590	50	1.9213	0.3642	5.36	0.4213	24	20	9
Danu-1	10957	0.9931	158	1.5328	50	2.0899	0.3441	5.0639	0.2634	136	55	17
Danu-2	10957	0.9577	46	1.9412	50	1.8957	0.3017	4.4398	0.3598	69	26	4
Danu-3	10958	0.9326	33	3.1827	50	2.7771	0.3782	5.5651	0.4059	15	9	2
Fras-1	10957	0.9772	70	1.5279	50	1.6994	0.3879	5.7077	0.3878	45	32	8
Fras-2	10958	0.9734	63	2.9821	50	3.1849	0.2511	3.6952	0.3529	18	12	3
Fras-3	10958	0.9582	47	2.3767	50	2.3411	0.2272	3.343	0.1886	14	9	3
Fras-4	8401	0.9294	30	2.2163	50	1.9096	0.2769	3.7833	0.3100	9	5	1
Missi-1	10958	0.9961	232	1.8789	50	3.0163	0.4133	6.0813	0.3909	193	73	15
Missi-2	10956	0.9921	144	2.6780	50	3.7589	0.3846	5.6601	0.4001	44	33	13
Missi-3	10958	0.9917	139	1.8277	50	2.6476	0.5098	7.5018	0.4847	25	17	7
Missi-4	10958	0.9563	45	2.7527	50	2.6345	0.5358	7.8847	0.0000	69	49	10
Misso-1	10958	0.9711	60	3.6930	50	3.9396	0.4484	6.5985	0.4238	195	28	4
Misso-2	9131	0.9805	75	3.6145	50	4.1707	0.4639	6.4915	0.4124	111	15	2
Misso-3	10958	0.9165	29	5.1261	50	4.1325	0.4179	6.1498	0.0000	55	38	7
Misso-4	10958	0.9522	42	3.2612	50	3.0869	0.2450	3.605	0.0000	104	45	7
Ohio-1	10958	0.9723	62	1.7652	50	1.8735	0.2910	4.2822	0.2983	20	15	7
Ohio-2	10958	0.9547	44	2.1173	50	2.0477	0.2569	3.781	0.2581	12	8	2
Ohio-3	10958	0.9291	32	1.7894	50	1.6164	0.3289	4.8401	0.2263	22	7	2
Ohio-4	10958	0.8985	25	1.9601	50	1.5937	0.3659	5.3839	0.3324	7	1	0
Rhine-1	10957	0.9897	120	1.2813	50	1.6822	0.3787	5.5729	0.4254	153	46	8
Rhine-2	10958	0.9715	61	2.0457	50	2.1880	0.3513	5.1699	0.0000	87	43	9
Rhine-3	10958	0.9048	26	2.1554	50	1.7478	0.3792	5.5799	0.4176	16	7	1
Rhine-4	10958	0.8739	21	2.2409	50	1.7306	0.2489	3.6627	0.3447	35	7	1

Note: In the Lo's R/S test, lag-1 is determined by the data-driven formula, lag-2 is the fixed lag, and stat-1 and stat-2 are their corresponding test statistics.

(1) The Lo's test indicates that about 1/3 (11 according to the data-driven lag, and 9 according to the fixed lag) of all the 31 streamflow series do not exhibit long-memory property, whereas the estimates of *S-MLE* also show that 4 out of all the series have *d*'s of zero value. But the results of Lo's test and *S-MLE* are not in agreement, namely, those series with zero *d* indicted by *S-MLE* seem to exhibit long-memory according to Lo's test. On the other hand, GPH test tells us that all the series exhibit long-memory. Therefore, for each series, at least two methods applied here give evidences of the existence of long-memory in all the daily streamflow processes.

(2) The GPH estimates and the *S-MLE* estimates are in good agreement, as shown in Figure 3.20, except for four series whose estimates of *d* with *S-MLE* are zero. According to such a consistency, we may consider that the estimates of zero given by *S-MLE* probably are resulted from its erroneousness.

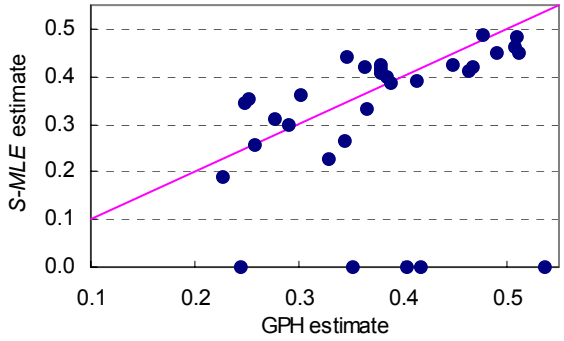


Figure 3.20 The GPH estimate versus the *S-MLE* estimate of *d*

(3) The predictability is closely related to autocorrelations at lag 1, ACF(1). An exponential relationship between ACF(1) and the predictability at different *CE* levels can be seen by visual inspection at Figure 3.21, which is basically the same as the relationship we found for the simulated AR and ARFIMA processes.

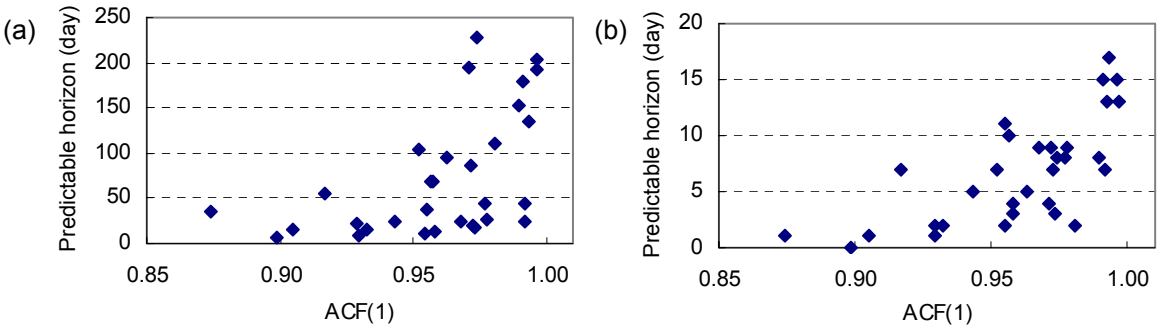


Figure 3.21 Predictability versus ACF(1) for streamflow processes (a) at SACE=0.01 level and (b) SACE = 0.5 level

(4) The fractional differencing parameter *d*, estimated with the *S-MLE* method, is linked to the predictability at both low level (SACE = 0.01) and high level, as shown in Figure 3.22 (Note that the erroneous zero estimates of *d* are removed in the plot). This is in accordance with the relationship between the predictability of the simulated ARFIMA processes and their fractional differencing parameters.

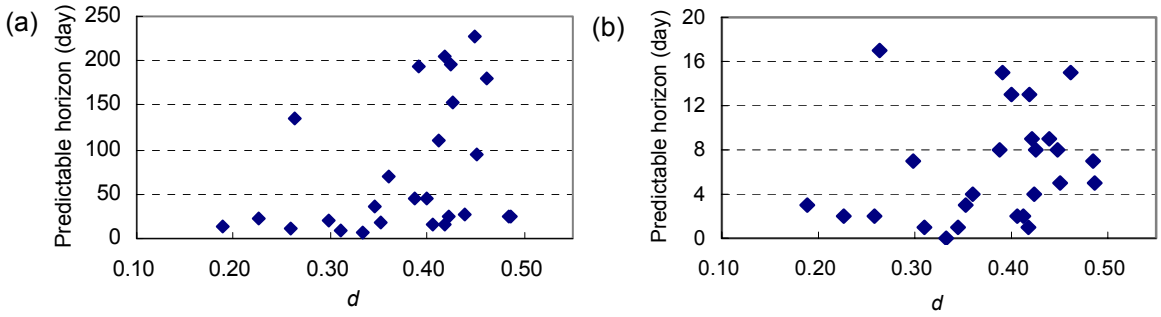


Figure 3.22 Predictability versus ACF(1) for streamflow processes (a) at SACE=0.01 level and (b) SACE = 0.5 level

(5) The intensity of long-memory, denoted by the value of *S-MLE* estimates of *d* (the zeros are removed) has little relationship with the watershed scale, as shown in Figure 3.23.

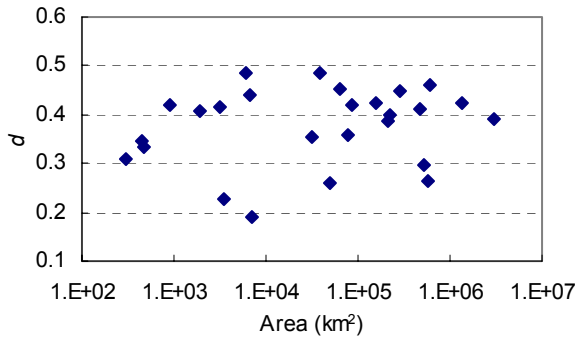


Figure 3.23 *d* versus watershed scale for streamflow processes

(6) There is a strong relationship between the predictability and the watershed scale. It is shown in Figure 3.24 that, the larger the watershed scale, the better the predictability of the streamflow processes. And the relationship is especially clear at low *SACE* level, as shown in Figure 3.24(a).

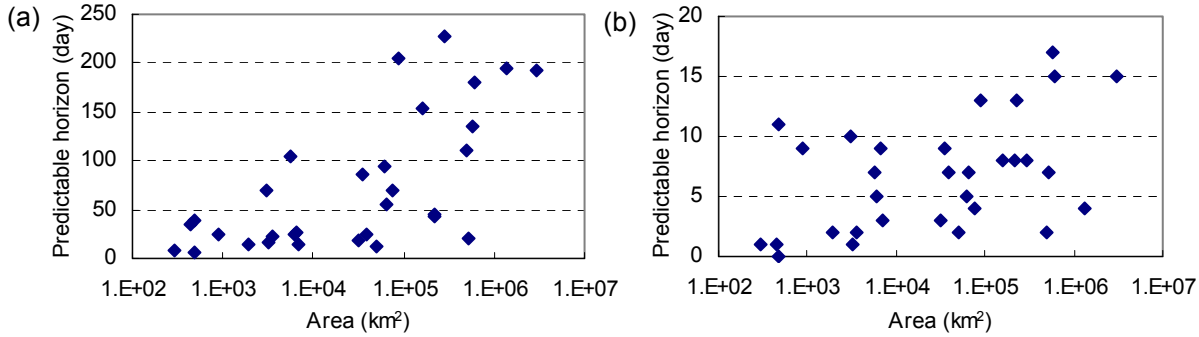


Figure 3.24 Predictability versus watershed scale for streamflow processes (a) at SACE = 0.01 level and (b) SACE = 0.5 level

3.6.4 Discussions on Long-memory and Predictability

3.6.4.1 *Sources of the long-memory in streamflow processes*

While the long-memory property has been detected in diverse fields of application, such as financial time series, meteorological time series and internet traffic time series, etc., the

physical mechanism of such type of phenomenon is considered a puzzle (Klemes, 1974). Compared with the long-memory phenomenon in other field (such as financial time series), While the mechanisms of long-memory underlying the streamflow process are not well-known, it may be comparatively easier to give some explanations about the sources of long-memory in streamflow time series than those in other time series.

The major source of long-memory may come from the storage of groundwater. Therefore the streamflow processes in the arid area may exhibit longer memory than the processes in humid areas because the streamflow are dominated by groundwater. This is why the streamflow processes in the Colorado Basin, located in the arid area, has the highest average value of fractional differencing parameters (excluding the estimate of zero *d*, which is due to the erroneousness of *S-MLE*). Another source of long-memory is the activity of the weather system. The less active the weather system, the less active the watershed system. In turn, the streamflow process has a longer memory. This is a postulation yet to be investigated. One more postulation yet to be examined is that the more complicated the groundwater system, the longer the memory.

The sources of long-memory many change over timescale. For example, while the long-memory in daily streamflow may come from the storage of groundwater system and the variation in precipitations process, the long-memory in annual hydrological time series, if it does exist, must be due to the low-frequency climatic variation. But since it is the behaviour of autocorrelations at long lags that matters, in small samples, such as an annual streamflow series of less than 100 points, it is hard to detect long-memory correctly.

3.6.4.2 *Limitations of the estimation of predictability*

Measurement of predictability is model-dependent, therefore there is always some degree of subjectivity and uncertainty. We suggest to use AR model to estimate predictability for the purpose of comparison, because of its easiness for using and simplicity for building.

Another source of uncertainty in estimating predictability is the mean values used for calculating *CE* or SA*CE*. Due to the possible long-term variation of climate system, watershed system may also exhibit long-term variation. In turn, the mean values of streamflow processes may change over time at certain timescale, and the long-term mean values may differ from short-term mean values, thus resulting in the exaggeration of predictability if long-term mean values are used to measure the predictability. For example, when estimating the predictability of the streamflow processes, we use the daily mean values of the entire series (mostly, data of 30 years) to calculate values of *SACE* for the forecasts of 10 years, this could be one of the reasons why many streamflow series exhibit long predictable horizons at low *SACE* level, longer than those of both the AR(1) and ARFIMA process which have equivalent values of ACF(1).

Data quality may affect the estimation of the predictability. For example, for some gauging stations, the gauged discharges are often the same for two weeks or even over a month continuously, especially when the discharge is very low. For instance, Minnesota River at Ortonville, MN, the discharge kept to be 58 cubic feet per second (cfs) for 17 days (1945.1.12 ~ 1945.1.28); 0.7 cfs for (December 28, 1964 ~ February 5, 1965); 3 cfs for 49 days (1968.1.6 ~ 1968.2.24), 11 cfs for 44 days (1971.1.11 ~ 1971.2.16). While in some cases, this may be true, but mostly probably, this is due to the limited measurement accuracy or even error. This leads to an exaggeration of the predictability.

While the predictability of is generally a stable physical feature of a streamflow process, it is not a constant. It may change due to the change of the basin characteristics, climate changes and human activities.

3.7 Conclusions

Coefficient of efficiency (*CE*) is a good criterion for evaluating model performance or model predictability for stationary processes. However, this criterion could be misleading about the model performance when being applied to seasonal processes. Therefore, a seasonally-adjusted *CE* (*SACE*) is proposed to make it more suitable for evaluating seasonal processes. For non-seasonal stationary processes, *SACE* is equal to *CE*.

According to *SACE*, generally, we can make reliable 5-day ahead forecasts for the daily flow series at TNH. **T**he long-memory ARFIMA(7,d,0) model performs slightly better than short memory ARMA(20,1) model. That gives us another evidence of the presence of long-memory in addition to the evidence given by the statistical tests.

Interval forecasts are important to supplement point forecasts so as to define the predictive uncertainty. In this study, the residual based empirical approach and bootstrap approach are applied to construct prediction interval (PI) for monthly streamflow forecasts. The results show that both empirical approach and bootstrap method work reasonably well, and the empirical approach gives results comparable to or even better than bootstrap method. Because of the simplicity and calculation-effectiveness, empirical method is preferable to the bootstrap method. When there is significant seasonal variation in the variance of the residuals, as shown in the residuals of the Yellow River and the Umpqua River, to improve the PI construction, it is necessary to use seasonal empirical distribution functions which are defined by seasonal residuals rather than use overall empirical distribution functions which are defined by entire residual. The result of this study may suggest that for certain types of model, especially when the non-linearity is involved (such as neural network models and the nearest neighbor method), for which theoretical formulae are not available for computing PIs, the empirical method could be a good practical choice to construct prediction interval in comparison with those more data-demanding and more complicated methods, such as ESP (Day, 1985), GLUE (Beven and Binley, 1992) and Bayesian method (Krzysztofowicz, 1999).

A univariate time series based approach is proposed to measure the predictability is proposed in this study. We define the predictability as the predictable horizon for which the prediction is no better than the mean value for a stationary process or the seasonal mean value for a seasonal process. At the same time, for practical purposes, we define the predictability at a given *CE* or *SACE* level as the predictable horizon for which the *CE* or *SACE* of predictions is larger than a given value. With such a definition, the predictability is easily comparable among different hydrological processes. Investigation of the predictabilities of a number of streamflow series at different basin scales shows that, in general, the larger the basin scale, the better the predictability.

Chapter 4 Testing and Modelling Autoregressive Conditional Heteroskedasticity of Streamflow Processes

4.1 Introduction to Autoregressive Conditional Heteroskedasticity

A univariate stochastic process $\{x_t\}$ is said to be homoskedastic if variances of x_t are constant for all times t. Otherwise, it is said to be heteroskedastic. When modeling hydrological time series, we usually focus on modeling and predicting the mean behaviour, and are rarely concerned with the time dependency of variances, although season-dependent variances are usually considered. The increased importance played by risk and uncertainty considerations in water resources management and flood defence practice as well as in modern hydrology theory, however, has necessitated the development of new time series techniques that allow for the modeling of time varying variances.

ARCH-type (autoregressive conditional heteroskedasticity) models that originate from econometrics give us an appropriate framework for studying this problem. Volatility (i.e. time-varying variance) clustering, in which large changes tend to follow large changes, and small changes tend to follow small changes, has been well recognized in financial time series. This phenomenon is called conditional heteroskedasticity, and can be modelled by ARCH-type models, including the ARCH model proposed by Engle (1982) and the later extension GARCH (generalized ARCH) model proposed by Bollerslev (1986), etc. Accordingly, when a time series exhibit autoregressive conditional heteroskedasticity, we say it has the ARCH effect or GARCH effect. ARCH-type models have been widely used to model the ARCH effect for economic and financial time series.

The ARCH-type model is a nonlinear mechanism that includes past variances in the explanation of future variances. ARCH-type models can generate accurate forecasts of future volatility, especially over short horizons for those series exhibiting ARCH effects; therefore provide a better estimate of the forecast uncertainty, which is valuable for water resource management and flood defence. And they take into account excess kurtosis (i.e. fat tail behaviour), which is common in financial and well as hydrological processes. Therefore ARCH-type models could be very useful for hydrological time series modelling. Some authors propose new models to reproduce the asymmetric periodic behaviour with large fluctuations around large streamflow and small fluctuations around small streamflow (e.g., Livina et al., 2003), which basically can be handled with those conventional time series models that have taken season-dependent variance into account, such as periodic autoregressive moving average (PARMA) models and deseasonalized ARMA models. However, little attention has been paid so far by the hydrological community to test and model the possible presence of the ARCH effect with which large fluctuations tend to follow large fluctuations, and small fluctuations tend to follow small fluctuations in streamflow series.

In this chapter, we will take the daily and monthly streamflow series of the upper Yellow River at Tangnaihai (TNH) in China as case study hydrological time series to test for the existence of the ARCH effect, and propose an ARMA-GARCH error model for daily flow

series. This chapter is organized as follows. First, the method of testing conditional heteroskedasticity of streamflow process is described; then, the causes of the ARCH effect and the inadequacy of commonly used seasonal time series models for modeling streamflow are discussed; finally, an ARMA-GARCH error model is proposed for capturing the ARCH effect existing in daily streamflow series.

4.2 Testing for the ARCH Effect in Streamflow Processes

The detection for the ARCH effect of a streamflow series is actually a test of serial independence applied to the serially uncorrelated fitting error of some model, usually a linear autoregressive (AR) model. We assume that linear serial dependence inside the original series is removed with a well-fitted pre-whitening model; any remaining serial dependence must be due to some nonlinear generating mechanism, which is not captured by the model. Here, the nonlinear mechanism we concern is the conditional heteroskedasticity. We will show that the nonlinear mechanism remaining in the pre-whitened streamflow series, namely the residual series, can be well interpreted as autoregressive conditional heteroskedasticity.

4.2.1 ARMA Models Fitted to Daily and Monthly Streamflows: Revisit

The construction of deseasonalized ARMA models to daily and monthly streamflow at TNH has been described in Chapter 3. Two linear ARMA-type models (one ARMA(20,1) and one AR(4)) are fitted to the log-transformed and deseasonalized daily and monthly flow series respectively. Figure 4.1 shows parts of the two residual series obtained from the two models.

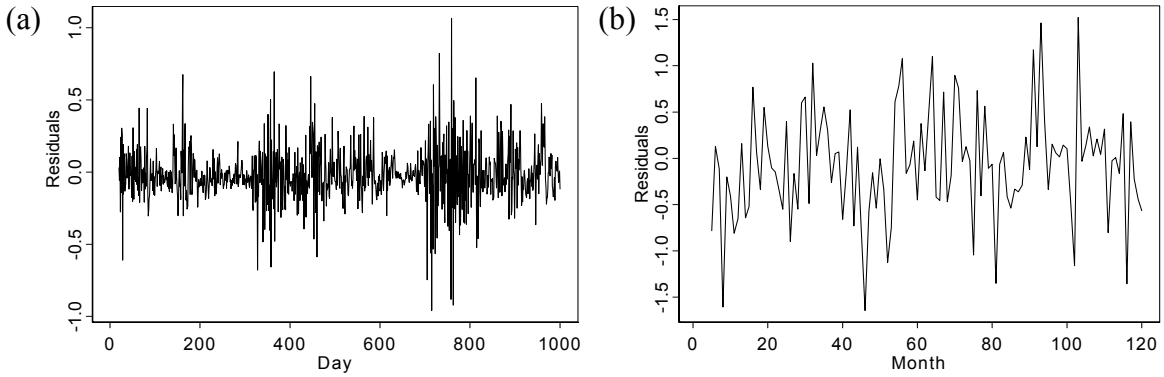


Figure 4.1 Segments of the residual series from (a) ARMA(20,1) for daily flows and (b) AR(4) for monthly flows of the Yellow River at TNH

Before applying ARCH tests to the residual series, to ensure that the null hypothesis of no ARCH effect is not rejected due to the failure of the pre-whitening linear models, we must check the goodness-of-fit of the linear models. The diagnosis has been done in Chapter 3, Section 3.1.2. It was shown that the ARMA models fit the daily and monthly flow series well. No serial correlation left in the residuals. However, albeit the residuals seem statistically uncorrelated according to their ACF as shown in Figure 3.3 in Chapter 3, they are not identically distributed from visual inspection of the Figure 4.1, that is, the residuals are not independent identically distributed (i.i.d.) through time. There is a tendency, especially for daily flow, that large (small) absolute values of the residual process are followed by other large (small) values of unpredictable sign, which is a common behaviour of GARCH processes.

Granger and Andersen (1978) found that some of the series modelled by Box and Jenkins (1976) exhibit autocorrelated squared residuals even though the residuals themselves do no

seem to be correlated over time, and therefore suggested that the ACF of the squared time series could be useful in identifying nonlinear time series. Bollerslev (1986) stated that the ACF and PACF of squared process are useful in identifying and checking GARCH behaviour. Figure 4.2 shows the ACFs of the squared residual series from ARMA(20,1) model for daily flow, AR(9) model for 1/3-monthly flow and AR(4) model for monthly flow at TNH. It is shown that although the residuals are almost uncorrelated as shown in Figure 3.3 in Chapter 3, the squared residual series are clearly autocorrelated, and the ACF structures of both squared residual series exhibit strong seasonality. That indicates that the variance of residual series is conditional on its past history, namely, the residual series may exhibit ARCH effect.

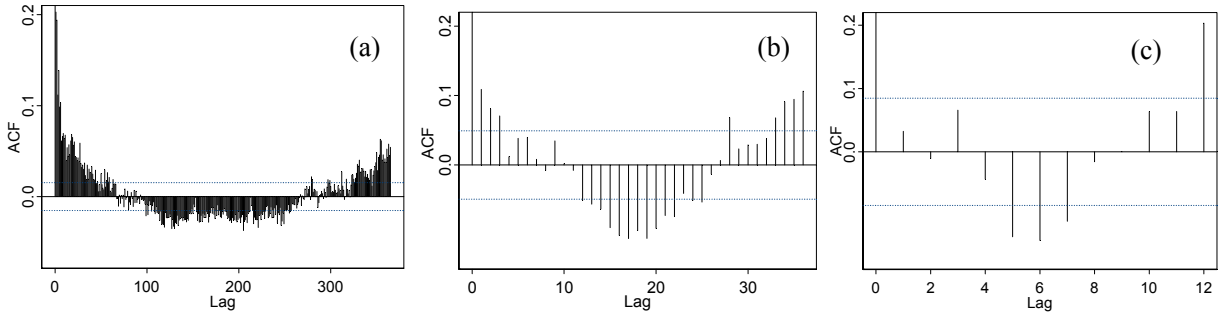


Figure 4.2 ACFs of the squared residuals from (a) ARMA(20,1) model for daily flows, (b) AR(9) for 1/3-monthly flows and (c) AR(4) model for monthly flows of the Yellow River at TNH

We also fitted AR models to streamflow processes of Danube, Rhine, Ocmulgee and Umpqua at different timescales, and obtain the residuals from these AR models. ACFs of squared residuals are displayed in Figure 4.3 to 4.6.

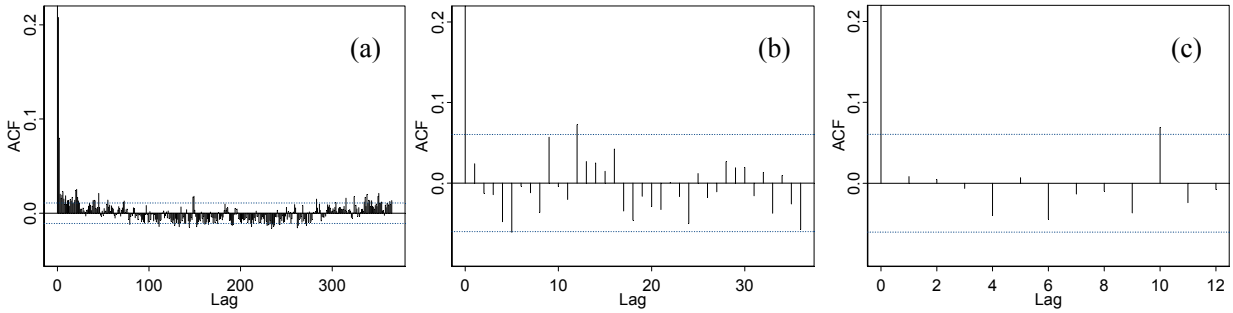


Figure 4.3 ACFs of the squared residuals from (a) AR(44) model for daily flows; (b) AR(12) for 1/3-monthly flows and (c) AR(6) model for monthly flows of Danube at Achleiten

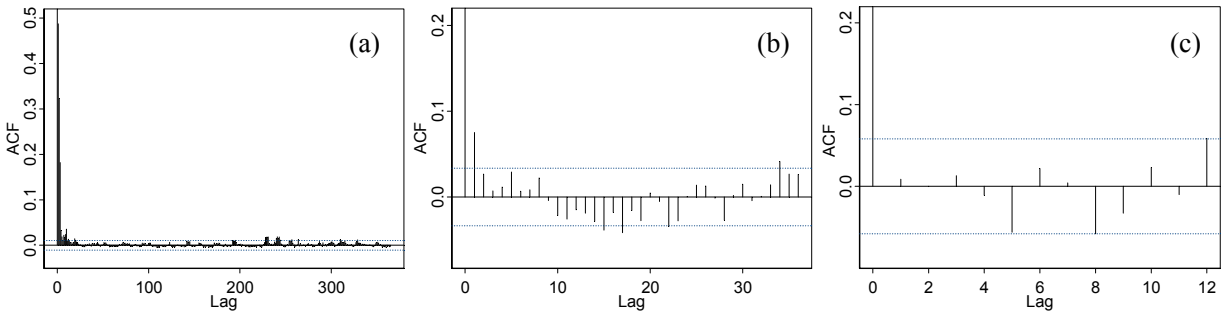


Figure 4.4 ACFs of the squared residuals from (a) AR(40) model for daily flows; (b) AR(16) for 1/3-monthly flows and (c) AR(4) model for monthly flows of Rhine at Lobith

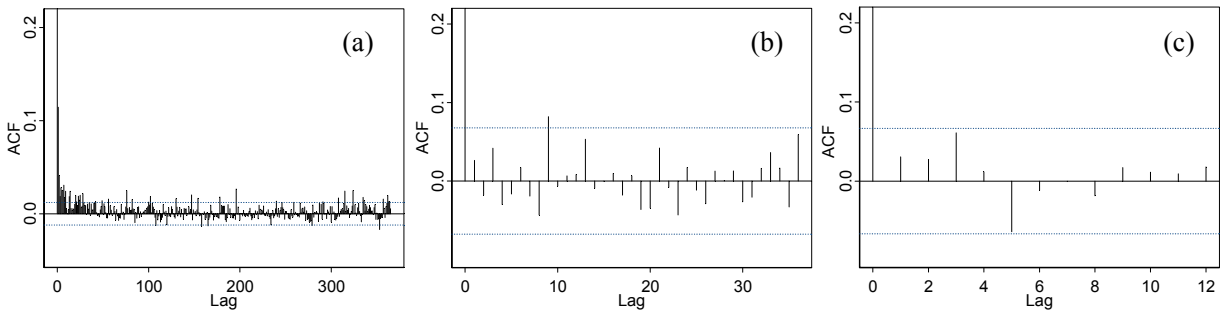


Figure 4.5 ACFs of the squared residuals from (a) AR(44) model for daily flows; (b) AR(8) for 1/3-monthly flows and (c) AR(3) model for monthly flows of Ocmulgee at Macon

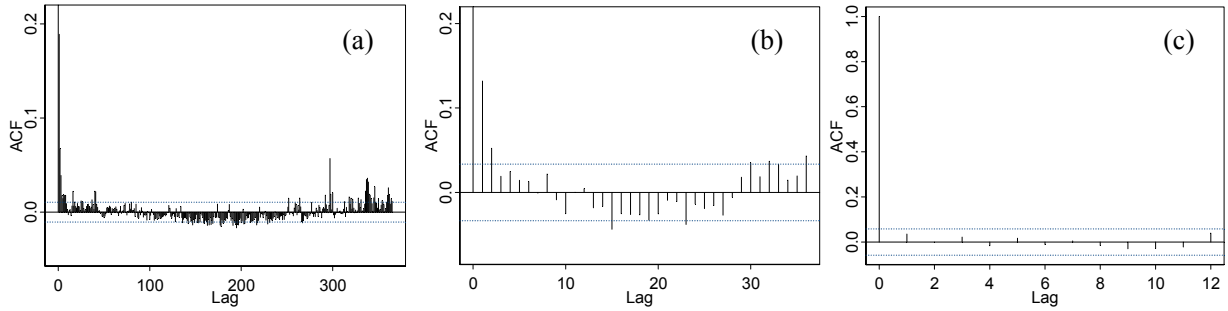


Figure 4.6 ACFs of the squared residuals from (a) AR(36) model for daily flows; (b) AR(7) for 1/3-monthly flows and (c) AR(5) model for monthly flows of Umpqua near Elkton

There are some formal methods to test for the ARCH effect of a process, such as McLeod-Li test (McLeod and Li, 1983), Engle's Lagrange Multiplier test (Engle, 1982), BDS test (Brock et. al., 1996), etc. McLeod-Li test and Engle's Lagrange Multiplier test are used here to check the existence of ARCH effect in the streamflow series.

4.2.2 Mcleod-Li Test for the ARCH Effect

McLeod and Li (1983) proposed a formal test for ARCH effect based on Ljung-Box test. It looks at the autocorrelation function of the squares of the pre-whitened data, and tests whether the first L autocorrelations for the squared residuals are collectively small in magnitude. Similar to Equation (3.6) in Chapter 3, for fixed sufficiently large L, the Ljung-Box Q-statistic of Mcleod-Li test is given by

$$Q = N(N+2)\sum_{k=1}^{L}\frac{\hat{r}_k^2(\varepsilon^2)}{N-k} \tag{4.1}$$

where, N is the sample size, and $\hat{r}_k^2(\varepsilon^2)$ is the squared sample autocorrelation of squared residual series at lag k. Under the null hypothesis of a linear generating mechanism for the data, namely, no ARCH effect in the data, the test statistic is asymptotically $\chi^2(L)$ distributed. Figure 4.7 shows the results of McLeod-Li test for daily and monthly flow. It illustrates that the null hypothesis of no ARCH effect is rejected for both daily and monthly flow series.

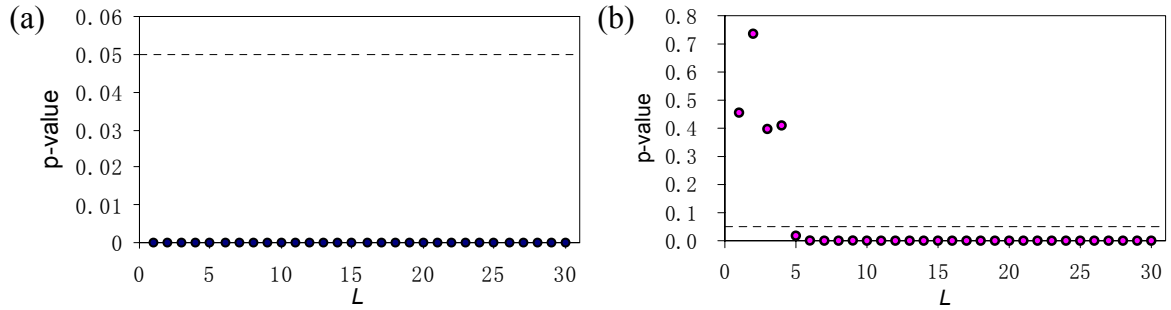


Figure 4.7 McLeod-Li test for the residuals from (a) ARMA(20,1) model for daily flow and (b) AR(4) model for monthly flow of Yellow River at TNH

4.2.3 Engle's Lagrange Multiplier Test for the ARCH Effect

Since the ARCH model has a form of an autoregressive model, Engle (1982) proposed the Lagrange Multiplier (LM) test to test for the existence of ARCH behaviour based on the regression. The test statistic is given by TR^2, where R is the sample multiple correlation coefficient computed from the regression of ε_t^2 on a constant and $\varepsilon_{t-1}^2,\ldots,\varepsilon_{t-q}^2$, and T is the sample size. Under the null hypothesis that there is no ARCH effect, the test statistic is asymptotically distributed as chi-square distribution with q degrees of freedom. As Bollerslev (1986) suggested, it should also have power against GARCH alternatives.

Figure 4.8 shows Engle's LM test results for the residuals from ARMA(20,1) model for daily flow and from AR(4) model for monthly flow. The results also firmly indicate the existence of ARCH effect in both residual series.

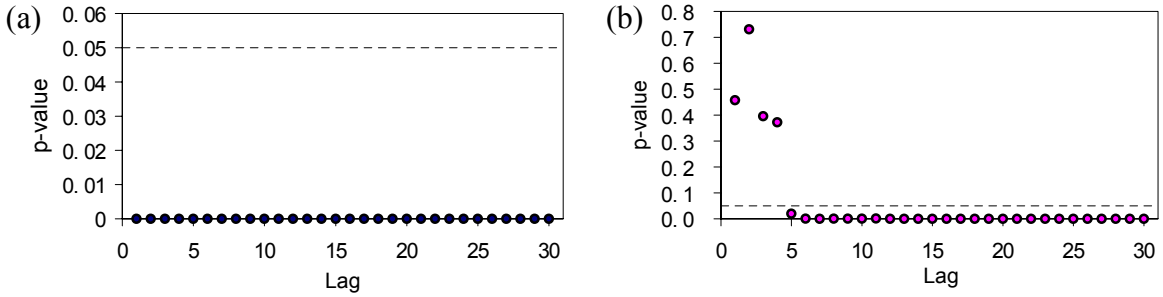


Figure 4.8 Engle's LM test for residuals from (a) ARMA(20,1) model for daily flow and (b) AR(4) model for monthly flow of Yellow River at TNH

One point should be noticed is that, although the ACF in Figure 4.2(b), McLeod-Li test result in Figure 4.7(b) and Engle's LM test in Figure 4.8(b) show that for monthly flow, autocorrelations at lags less than 4 are removed by AR(4) model, when we take autocorrelations at longer lags into consideration, significant autocorrelations remain and the null hypothesis of no ARCH effect is rejected. Because it is required for McLeod-Li test to use sufficiently large L, namely, sufficient number of autocorrelations to calculate the Ljung-Box statistic (typically around 20), therefore, we still consider that the monthly flow has the ARCH effect.

On the whole, evidences are clear with the McLeod-Li test and Engle's LM test about the existence of conditional heteroskedasticity in the residual series from linear models fitted to the log-transformed and deseasonalized daily and monthly streamflow processes of the upper Yellow River at TNH.

4.3 Discussion on the Causes of ARCH Effects and Inadequacy of Commonly Used Seasonal Time Series Models

4.3.1 Causes of ARCH Effects in Daily and Monthly Flows

From the above analyses, it is clear that although the residuals are serially uncorrelated, they are not independent through time. At the mean time, we notice that seasonality dominates autocorrelation structures of squared residual series for both daily and monthly flow process (as shown in Figure 4.2). This suggests that there are seasonal variations in the variance of the residual series, and we should standardize the residual series from linear models with seasonal standard deviations of the residuals first, and then look at the standardized series to check whether seasonal variances can explain ARCH effects.

Seasonal standard deviations of the residual series from ARMA (20,1) model for daily flow and AR(4) mode for monthly flow are calculated and shown in Figure 4.9 (a) and (b). They are used to standardize the residual series from ARMA (20,1) model and AR(4) model. Figure 4.10 shows the ACFs of the squared standardized residual series of daily and monthly flow. It is illustrated that, after seasonal standardization autocorrelation as well as the seasonality in the squared standardized residual series for monthly flow is basically removed (Figure 10b), whereas the significant autocorrelation still exists in the squared standardized residual series for daily flow (Figure 10a) despite the fact that the autocorrelations are significantly reduced compared with Figure 8a and the seasonality in the ACF structure is removed. That means that the seasonality as well as the autocorrelation in the squared residuals from the AR model of monthly flow series is basically caused by seasonal variances. But seasonal variances only explain partly the autocorrelation in the squared residuals of daily flow series.

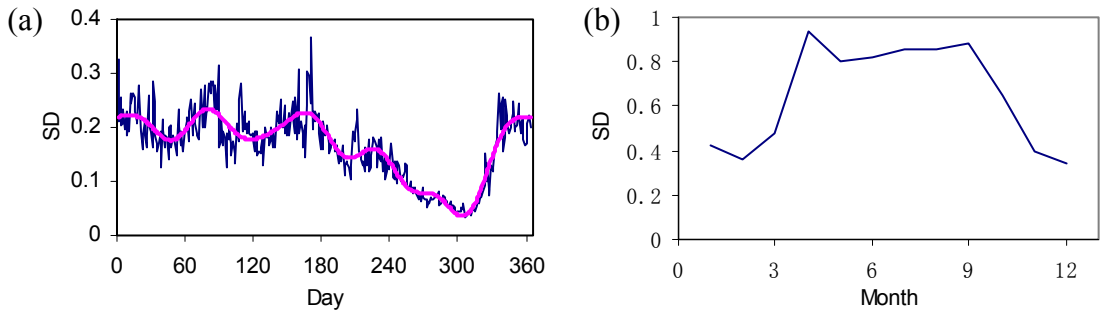


Figure 4.9 Seasonal standard deviations (SD) of the residuals form (a) ARMA(20,1) for daily flow and (b) AR(4) for monthly flow (Note: The smoothed line in 4.9(a) is given by the first 8 Fourier harmonics)

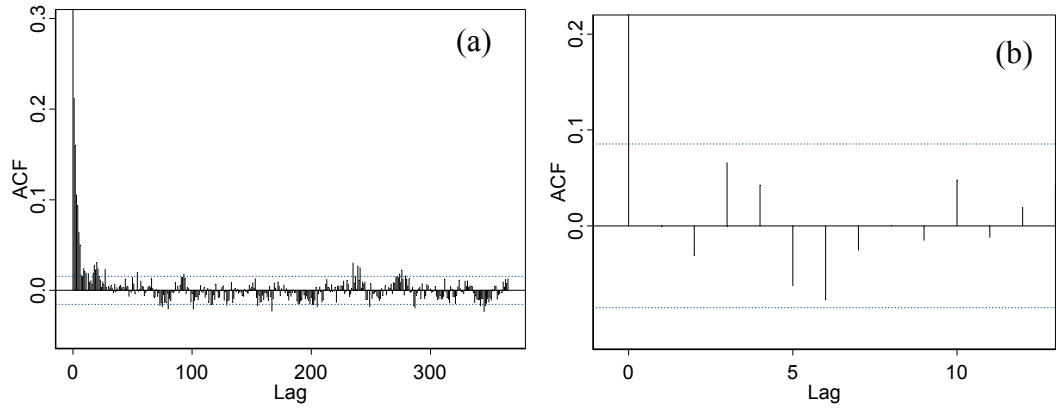


Figure 4.10 ACFs of squared seasonally standardized residuals from (a) ARMA(20,1) model for daily flow and (b) AR(4) model for monthly flow at TNH

The residual series of daily flow and monthly flow standardized by seasonal standard deviation are also tested for ARCH effects with the McLeod-Li test and Engle's LM test. Figure 4.11 shows that the seasonally standardized residual series of daily flow still cannot pass the LM test (Figure 13a), whereas the seasonally standardized residual series of monthly flow pass the LM test with high p-values (Figure 4.11b). The McLeod-Li test gives similar results.

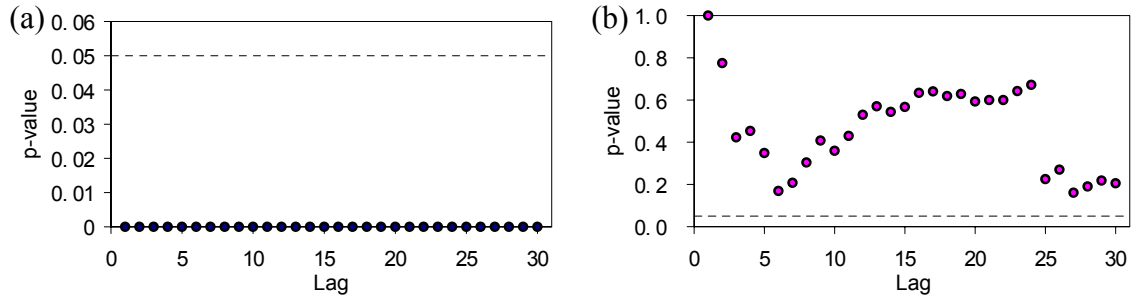


Figure 4.11 Engle's LM test for seasonally standardized residuals from (a) ARMA(20,1) model for daily flow and (b) from AR(4) model for monthly flow

From the above analyses, it is clear that the ARCH effect is fully caused by seasonal variances for monthly flow, but only partly for daily flow. Other causes, besides the seasonal variation in variance, of the ARCH effect in daily flow may include the perturbations of the temperature fluctuations which is an influential factor for snowmelt as well as evapotranspiration, and the precipitation variation which is the dominant factor for streamflow processes. As reported by Miller (1979), when modeling a daily average streamflow series, the residuals from a fitted AR(3) model signalled white-noise errors, but the squared residuals signalled bilinearity. When precipitation covariates were included in the model, Miller found that neither the residuals nor the squared residuals signalled any problems. While we agree that the autocorrelation existing in the squared residuals is basically caused by precipitation process, we want to show that the autocorrelation in the squared residuals can be well described by an ARCH model, which is very close to the bilinear model (Engle, 1982).

4.3.2 Inadequacy of Commonly Used Seasonal Time Series Models for Modelling Streamflow Processes

As mentioned in Chapter 3 Section 3.1, SARIMA models, deseasonalized ARMA models and periodic models are commonly used to model hydrological processes (Hipel and McLeod, 1994). The general form of SARIMA model, denoted by SARIMA(p,d,q)×(P,D,Q)$_S$, is

$$\phi(B)\Phi(B^s)\nabla^d\nabla_s^D x_t = \theta(B)\Theta(B^s)\varepsilon_t \tag{4.2}$$

where $\phi(B)$ and $\theta(B)$ of orders p and q represent the ordinary autoregressive and moving average components; $\Phi(B^s)$ and $\Theta(B^s)$ of orders P and Q represent the seasonal autoregressive and moving average components; $\nabla^d = (1\text{-}B)^d$ and $\nabla_S^D = (1\text{-}B^s)^D$ are the ordinary and seasonal difference components.

The general form of ARMA(p,q) model fitted to deseasonalized series is

$$\phi(B)x_t = \theta(B)\varepsilon_t \tag{4.3}$$

where $\phi(B)$ and $\theta(B)$ of orders p and q represent the ordinary autoregressive and moving average components.

From the model equations we know that, although the seasonal variation in the variance present in the original time series is basically dealt with well by the deseasonalized approach, the seasonal variation in variance in the residual series is not considered by none the two models, because in both cases the innovation series ε_t is assumed to be i.i.d. $N(0, \sigma^2)$. Therefore, both SARIMA models and deseasonalized models cannot capture the ARCH effect that we observed in the residual series.

In contrast, the periodic model, which is basically a group of ARMA models fitted to separate seasons, allows for seasonal variances in not only the original series but also the residual series. Taking the special case PAR(p) model (periodic autoregressive model of order p) as an example of PARMA model, given a hydrological time series $x_{n,s}$, in which n defines the year and s defines the "season" (could represent a day, week, month or a real season), we have the following PAR(p) model (Salas, 1993):

$$x_{n,s} = \mu_s + \sum_{j=1}^{p} \phi_{j,s}(x_{v,s-j} - \mu_{s-j}) + \varepsilon_{n,s} \tag{4.4}$$

where $\varepsilon_{n,s}$ is an uncorrelated normal variable with mean zero and variance σ_s^2. For daily streamflow series, to make the model parsimonious, we can cluster the days in the year into several groups and fit separate AR models to separate groups, as shown in Section 3.3. Periodic models would perform better than SARIMA model and deseasonalized ARMA model for capturing the ARCH effect, because it takes season-varying variances into account. However, as analyzed in Section 4.1, while considering seasonal variances could be sufficient for describing the ARCH effect in monthly flow series because the ARCH effect in monthly flow series is fully caused by seasonal variances, it is still insufficient to fully capture the ARCH effect in daily flow series.

In summary, while PARMA model is adequate for modeling the variance behaviour for monthly flow, none of the commonly used seasonal models is efficient enough to describe the ARCH effect for daily flow, although PARMA can partly make it by considering seasonal variances. It is necessary to apply GARCH model to achieve the purpose.

4.4 Modelling Daily Streamflow Process with ARMA-GARCH Error Model

4.4.1 Model Building

Weiss (1984) proposed ARMA models with ARCH errors. This approach is adopted and extended by many researchers for modeling economic time series (e.g., Hauser and Kunst; 1998; Karanasos, 2001). In the field of geo-sciences, Tol (1996) fitted a GARCH model for the conditional variance and the conditional standard deviation, in conjunction with an AR(2) model for the mean, to model daily mean temperature. In this study, we propose to use ARMA-GARCH error (or for notation convenience, call ARMA-GARCH) model for modeling daily streamflow processes.

The ARMA-GARCH model may be interpreted as a combination of an ARMA model which is used to model the mean behaviour and an ARCH model to model the ARCH effect in the residual series form the ARMA model. The ARMA model has the form as in Equation (4.3). The GARCH(p, q) model has the form (Bollerslev, 1986)

$$\begin{cases} \varepsilon_t \,|\, \psi_{t-1} \sim N(0, h_t) \\ h_t = \alpha_0 + \sum_{i=1}^{q} \alpha_i \varepsilon_{t-i}^2 + \sum_{i=1}^{p} \beta_i h_{t-i} \end{cases} \tag{4.5}$$

where, ε_t denotes a real-valued discrete-time stochastic process, and ψ_t the available information set, $p \geq 0$, $q > 0$, $\alpha_0 > 0$, $\alpha_i \geq 0$, $\beta_i \geq 0$. When $p=0$, the GARCH(p,q) model reduces to the ARCH(q) model. Under the GARCH(p, q) model, the conditional variance of ε_t, h_t, depends on the squared residuals in the previous q time steps, and the conditional variance in the previous p time steps. Since GARCH models can be treated as ARMA models for squared residuals, the order of GARCH can be determined with the method for selecting the order of ARMA models, and traditional model selection criteria such as Akaike information criterion (AIC) and Bayesian information criterion (BIC) can also be used for selecting models. The unknown model parameters α_i (i = 0, …, q) and β_j (j = 1, …, p) can be estimated using (conditional) maximum likelihood estimation (MLE). Estimates of the conditional standard deviation $h_t^{1/2}$ are also obtained as a side product with the MLE method.

When there is obvious seasonality present in the residuals (as in the case of daily streamflow at TNH), to preserve the seasonal variances in the residuals, instead of fitting the ARCH model to the residual series directly, we fit the ARCH model to the seasonally standardized residual series, which is obtained by dividing the residual series by seasonal standard deviations (i.e. daily standard deviations for daily flow). Threrefore, the general ARMA-GARCH model with seasonal standard deviations we propose here has the following form

$$\begin{cases} \phi(B) x_t = \theta(B) \varepsilon_t \\ \varepsilon_t = \sigma_s z_t, \;\; z_t \sim N(0, h_t) \\ h_t = \alpha_0 + \sum_{i=1}^{q} \alpha_i z_{t-i}^2 + \sum_{i=1}^{p} \beta_i h_{t-i} \end{cases} \tag{4.6}$$

where, σ_s is the seasonal standard deviation of ε_t, s is the season number depending on which season the time t belongs to. For daily series, s ranges from 1 to 366. Other notations are the same as in Equation (4.3) and (4.5).

The model building procedure proceeds in the following steps:

(1) Log-transform and deseasonalize the original flow series;
(2) Fit an ARMA model to the log-transformed and deseasonalized flow series;
(3) Calculate seasonal standard deviations of the residuals obtained from ARMA model, and seasonally standardize the residuals with the first 8 Fourier harmonics of the seasonal standard deviations;
(4) Fit a GARCH model to the seasonally standardized residual series.

For forecasting and simulation, inverse transformation (including log-transformation and deseasonalization) is needed. When forecasting, the ARMA part of the ARMA-GARCH model forecasts future mean values of the underlying time series following the traditional approach for ARMA prediction, whereas the GARCH part gives forecasts of future volatility, especially over short horizons.

Following the above-mentioned steps, a preliminary ARMA-GARCH model is fitted to the daily streamflow series at TNH. The ACF and PACF structure of the squared seasonally standardized residuals are shown in Figure 4.10a and Figure 4.12, respectively. According to the AIC as well as the ACF and PACF structure, a GARCH(0,21) model, i.e., ARCH(21)

model, which has the smallest AIC value is selected. Therefore, the preliminary ARMA-GARCH model fitted to the daily streamflow series at TNH is composed of an ARMA(20,1) model and an ARCH(21) model. The model is constructed with statistics software S+Finmetrics (Zivot and Wang, 2003).

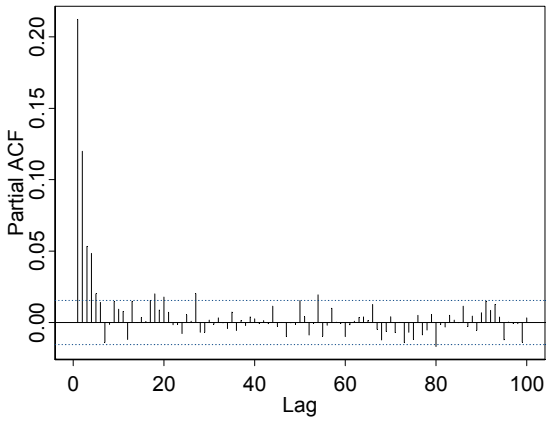


Figure 4.12 PACF of the squared seasonally standardized residual series from ARMA(20,1) for daily flow series of the Yellow River at TNH

4.4.2 Model Diagnostic and Modification

If the ARMA-GARCH model is successful in modeling the serial correlation structure in the conditional mean and conditional variance, then there should be no autocorrelation left in both the residuals and the squared residuals standardized by the estimated conditional standard deviation.

A segment of the seasonally standardized residual series from ARMA(20,1) model and its corresponding conditional standard deviation sequence estimated with ARCH(21) model are shown in Figure 4.13 (a) and (b). We standardize the seasonally standardized residual series from ARMA(20,1) model by dividing it by the estimated conditional standard deviation sequence. The autocorrelations of the standardized residuals and squared standardized residuals are plotted in Figure 4.14. It is shown that although there is no autocorrelation left in the squared standardized residuals, which means that the ARCH effect has been removed (Figure 4.14b), however, in the non-squared standardized residuals of daily flow significant autocorrelation remains (Figure 4.14a).

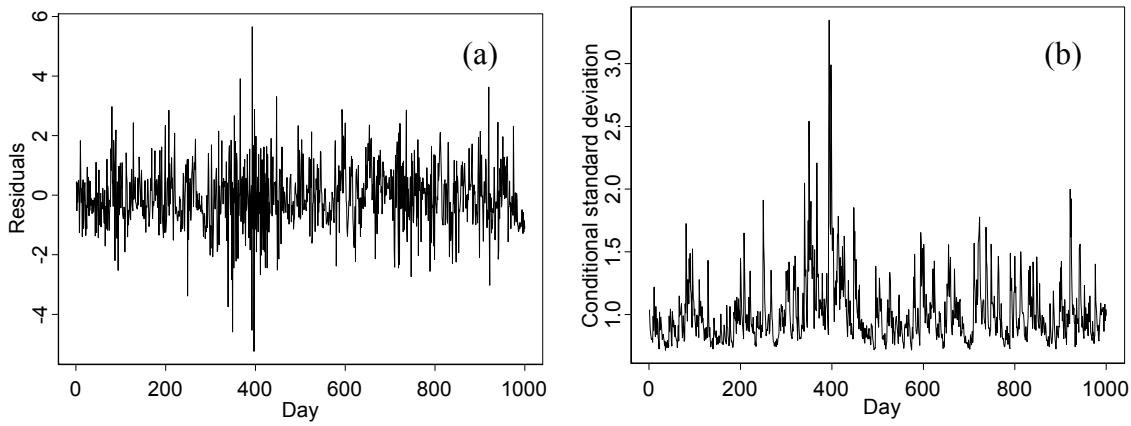


Figure 4.13 A segment of (a) the seasonally standardized residuals from ARMA(20,1) and (b) its corresponding conditional standard deviation sequence estimated with ARCH(21) model

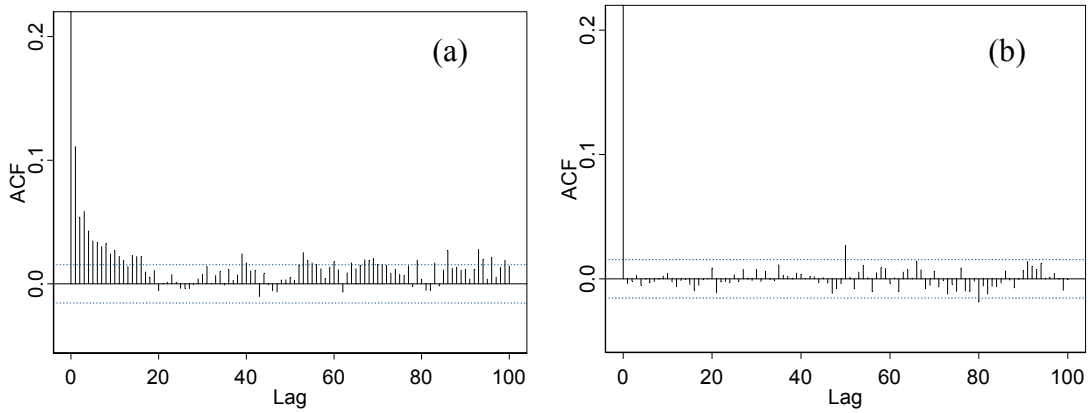


Figure 4.14 ACFs of (a) the standardized residuals and (b) squared standardized residuals from ARMA(20,1)-ARCH(21) model. The standardization is accomplished by dividing the seasonally standardized residuals from ARMA(20,1) by the conditional standard deviation estimated with ARCH(21).

Because the GARCH model is designed to deal with the conditional variance behavior, rather than mean behavior, the autocorrelation in the non-squared residual series must arise from the seasonally standardized residuals obtained in step (3) of ARMA-GARCH model building procedure. Therefore we revisit the seasonally standardized residuals. It is found that, although the residuals from the ARMA(20,1) model present no obvious autocorrelation as shown in Figure 3.3 in Chapter 3, weak but significant autocorrelations in the residuals are revealed after the residuals are seasonally standardized, as shown by the ACF and PACF in Figure 4.15. We refer to this weak autocorrelation as the hidden weak autocorrelation.

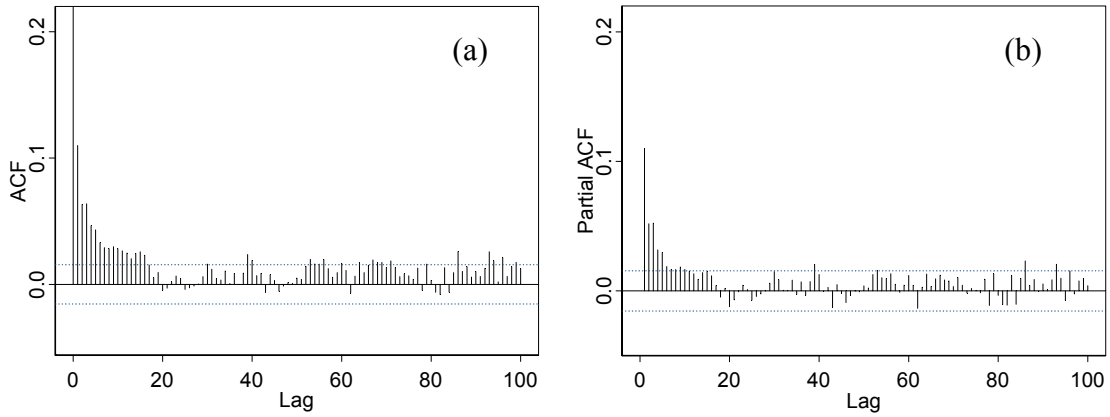


Figure 4.15 ACF and PACF of seasonally standardized residuals from ARMA(20,1) model

The mechanism underlying such weak autocorrelation is not clear yet. Similar phenomena are also found for some other daily streamflow processes (such as the daily streamflow of the Umpqua River near Elkton and Wisconsin River near Wisconsin Dells in the United States, available on the USGS website http://water.usgs.gov/waterwatch), which have strong seasonality in the ACF structures of their original series as well as their residual series. To handle the problem of the weak correlations, an additional ARMA model is needed to model the mean behaviour in the seasonally standardized residuals, and a GARCH is then fitted to the residuals from this additional ARMA model. Therefore, we get an extended version of the model in Equation (4.6) as

$$\begin{cases} \phi(B)x_t = \theta(B)\varepsilon_t \\ \varepsilon_t = \sigma_s y_t \\ \phi'(B)y_t = \theta'(B)z_t,\ z_t \sim N(0,h_t) \\ h_t = \alpha_0 + \sum_{i=1}^{q}\alpha_i z_{t-i}^2 + \sum_{i=1}^{p}\beta_i h_{t-i} \end{cases} \tag{4.7}$$

where, y_t is the seasonally standardized residuals from the first ARMA model, z_t is the residuals (for notation convenience, we call it second-residuals) from the second ARMA model fitted to y_t.

An AR (16) model, whose autoregressive order is chosen according to AIC, is fitted to the seasonally standardized residuals from the ARMA(20,1) model of the daily flow series at TNH, and we get a second-residual series from this AR(16) model. The autocorrelations of the second-residual series and the squared second-residual series from the ARMA(20,1)-AR(16) combined model are shown in Figure 4.16. From visual inspection, we know that no autocorrelation is left in the second-residual series, but there is strong autocorrelation in the squared second-residual series which indicates the existence of ARCH effect.

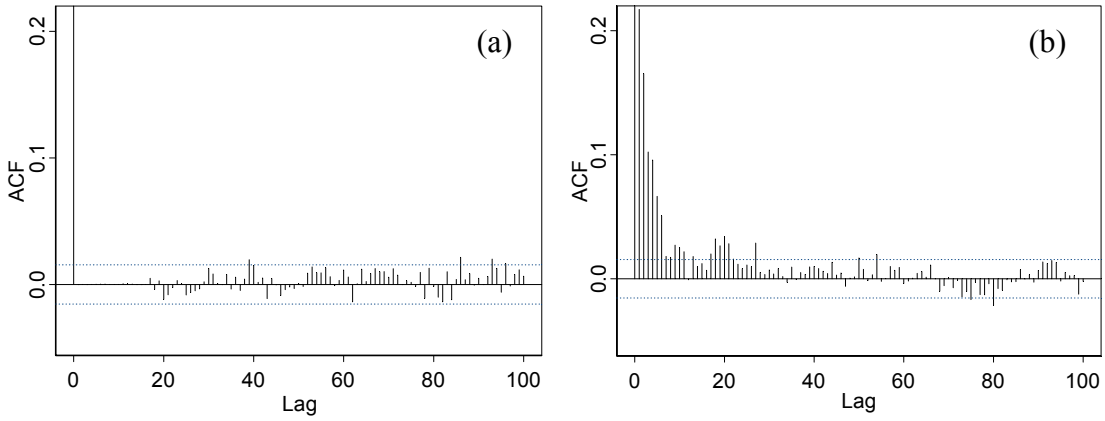


Figure 4.16 ACFs of (a) the second-residuals and (b) the squared second-residuals from the ARMA(20,1)-AR(16) model

Because the squared second-residual series has similar ACF and PACF structure to the seasonally standardized residuals from ARMA(21,0) model, the same structure of GARCH model, i.e., an ARCH(21) model, is fitted to the second-residual series. Therefore, the ultimate ARMA-GARCH model fitted to the daily streamflow at TNH is ARMA(20,1)-AR(16)-ARCH(21), composed of an ARMA(20,1) model fitted to log-transformed and deseasonalized series, an AR(16) model fitted to the seasonally standardized residuals from the ARMA(20,1) model, and an ARCH(21) model fitted to the second-residuals from the AR(16) model.

We standardize the second-residual series with the conditional standard deviation sequence obtained with the ARCH(21) model. The autocorrelations of the standardized second-residuals and the squared standardized second-residuals are shown in Figure 4.17. Compared with Figure 4.14, the autocorrelations are basically removed for both the squared and non-squared series although the autocorrelation at lag 1 of the standardized second-residuals slightly exceeds the 5% significance level. The McLeod-Li test and the LM-test (shown in Figure 4.18) for standardized second-residuals also confirm that the ARCH(21) model fits the

second-residual series well. The small lag-1 autocorrelation in the standardized second-residual series (shown in Figure 4.17) is hidden autocorrelation covered by conditional heteroskedasticity. This autocorrelation can be further modelled with another AR model, but because the autocorrelation is very small, it could be neglected.

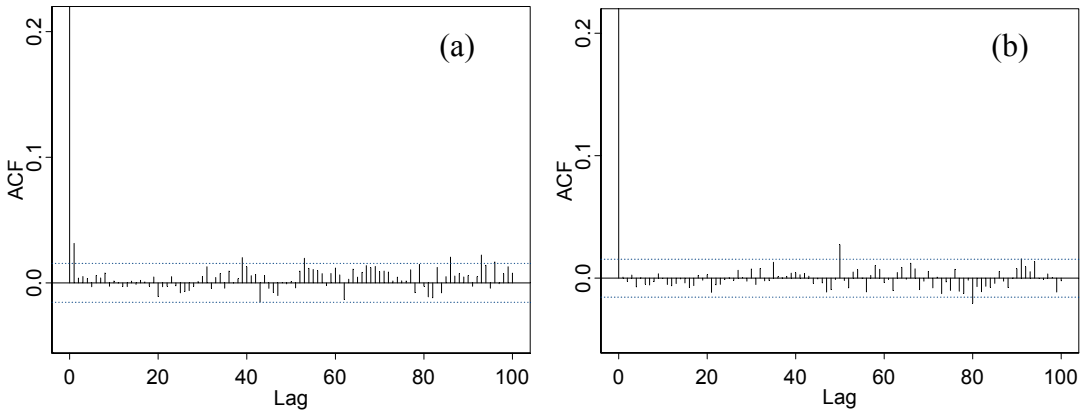


Figure 4.17 ACFs of the (a) standardized second-residuals and (b) squared standardized second-residuals from ARMA(20,1)-AR(16)-ARCH(21) model. The second-residuals are obtained from AR(16) fitted to the seasonally standardized residuals form ARMA(20,1).

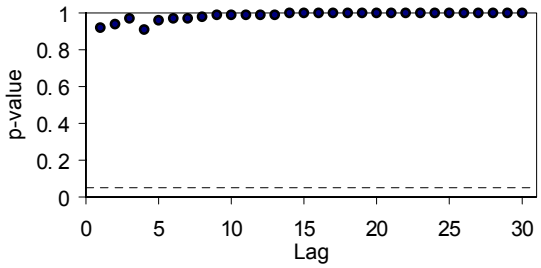


Figure 4.18 Engle's LM test for the standardized second-residuals from the ARMA(20,1)-AR(16)-ARCH(21) model

4.5 Conclusions

The nonlinear mechanism conditional heteroskedasticity in hydrological processes has not received much attention in literature so far. Modelling data with time varying conditional variance could be attempted various ways, including nonparametric and semi-parametric approaches (see Lall, 1995 and Sankarasubramanian and Lall, 2003). A parametric approach with ARCH model is proposed in this study to describe the conditional variance behavior in the streamflow process.

The existence of ARCH effect is verified in the residual series from linear models fitted to the daily and monthly streamflow processes of the upper Yellow River with McLeod-Li test and Engle's Lagrange Multiplier test. It is shown that the ARCH effect is fully caused by seasonal variation in the variance for monthly flows, but seasonal variation in variance only partly explains the ARCH effect for daily streamflow. Among three types of conventional seasonal time series model (i.e., SARIMA, deseasonalized ARMA and PARMA), no one is efficient enough to describe the ARCH effect for daily flow, although PARMA model is enough for monthly flow by considering season-dependent variances. Therefore, to fully capture the ARCH effect as well as the seasonal variances inspected in the residuals from linear ARMA

models fitted to the daily flow series; the ARMA-GARCH error model with seasonal standard deviations is proposed.

The ARMA-GARCH model is basically a combination of an ARMA model, which is used to model the mean behaviour, and a GARCH model to model the ARCH effect in the residuals from the ARMA model. To preserve the seasonal variation in variance in the residuals, the ARCH model is not fitted to the residual series directly, but to the seasonally standardized residuals. Therefore, an important feature of the ARMA-GARCH model is that the unconditional seasonal variance of the process is seasonally constant but the conditional variance is not. To resolve the problem of the weak hidden autocorrelation revealed after the residuals are seasonally standardized, the ARMA-GARCH model is extended by applying an additional ARMA model to model the mean behaviour in the seasonally standardized residual series. With such a modified ARMA-GARCH model, the daily streamflow series is well-fitted.

Because the ARCH effect in daily streamflow mainly arises from daily variations in temperature and precipitation, therefore, given that we have reasonably good skill in predicting weather in two to three days (for example, see http://weather.gov/rivers_tab.php), the use in developing an ARMA-GARCH model would be limited. However, because on the one hand, the relationship between runoff and rainfall and temperature is hard to captured precisely by any model so far; on the other hand, usually there are not enough rainfall data available to fully capture the rainfall spatial pattern, especially for remote areas, such as Tibet Plateau, and the accuracy of the weather forecasts for these areas are very limited, the ARCH effect cannot be fully removed even after limited rainfall data and temperature data are included in the model. Therefore, the ARMA-GARCH model would be a very useful addition in terms of statistical modeling of daily streamflow processes for the hydrological community.

Chapter 5 Testing for Nonlinearity of Streamflow Processes

5.1 Introduction

A major concern in many scientific disciplines is whether a given process should be modelled as linear or as nonlinear. In algebra, we define linearity in terms of functions that have the property $f(x+y) = f(x)+ f(y)$ and $f(a\times x) = a \times f(x)$. Nonlinear is defined as the negation of linear. This means that the result f may be out of proportion to the input x or y. It is currently well accepted that many natural systems are nonlinear with feedbacks over many space and timescales. However, certain aspects of these systems may be less nonlinear than others and the nature of nonlinearity may not be always clear (Tsonis, 2001). As an example of natural systems, streamflow processes are also commonly perceived as nonlinear. They could be governed by various nonlinear mechanisms acting on different temporal and spatial scales. Investigations on nonlinearity and applications of nonlinear models to hydrological processes have received much attention since several decades ago (e.g., Amorocho, 1963; Amorocho and Brandstetter, 1971). Rogers (1980, 1982) and Rogers and Zia (1982) developed a heuristic method to quantify the degree of nonlinearity of drainage basins by using rainfall-runoff data. Rao and Yu (1990) used Hinich bispectrum test (1982) to investigate the linearity and nongaussian characteristics of annual streamflow and daily rainfall and temperature series. They detected nonlinearity in daily meteorological series, but not in annual streamflow series. Chen and Rao (2003) investigated nonlinearity in monthly hydrological time series with the Hinich test. The results indicate that all of the stationary segments of standardized monthly temperature and precipitation series are either Gaussian or linear, and some of the standardized monthly streamflow are nonlinear.

As a special case of nonlinearity, chaos is widely concerned and has gained much attention in the hydrology community in the last two decades (e.g., Wilcox et al., 1991; Jayawardena and Lai, 1994; Porporato and Ridolfi, 1996; Sivakumar, 1999; Elshorbagy et al., 2002). Most of the research in literature confirms the presence of chaos in the hydrological time series. Consequently, some researchers (e.g., Sivakumar, 2000) believed that the dynamic structures of the seemingly complex hydrological processes, such as rainfall and runoff, might be better understood using nonlinear deterministic chaotic models than the stochastic ones. Meanwhile, some studies denied the existence of chaos in hydrological processes (e.g., Wilcox et al., 1991; Koutsoyiannis and Pachakis, 1996; Pasternack, 1999; Khan et al., 2005), and there are many disputes about the existence of low-dimensional chaos in hydrological processes (e.g., Ghilardi and Rosso, 1990; Schertzer et al., 2002).

Albeit the advances in the research on the nonlinear features of streamflow processes, further investigation is still desirable, because on the one hand, there is no common knowledge about

what type of nonlinearity exists in the streamflow process, and on the other hand, it is not clear how the character and intensity of nonlinearity of streamflow processes may change as the timescale changes. More insights into the nature of nonlinearity would allow one to decide whether a given process should be modelled with a linear or a nonlinear model.

In this chapter, several issues are addressed. Firstly, streamflow series of different timescales, namely, one year, one month, 1/3-month and one day, of four streamflow processes in different climate regions are studied to investigate the existence and intensity of general nonlinearity with the BDS test. Secondly, the correlation exponent method will be applied to test for the presence of chaos in the streamflow series. Thirdly, we will discuss whether it is possible that the streamflow process is a deterministic chaotic process. Fourthly, we will discuss the sources of nonlinearity in streamflow processes.

5.2 Testing for Nonlinearity with BDS Test

It is hard to explore different types of nonlinearity one by one which may possibly act underlying streamflow processes. We here want to investigate the existence of general nonlinearity in the streamflow process from a univariate time series data based quantitative point of view. There are a wide variety of methods available presently to test for linearity or nonlinearity, which may be divided into two categories: portmanteau tests, which test for departure from linear models without specifying alternative models, and the tests designed for some specific alternatives. Patterson and Ashely (2000) applied 6 portmanteau test methods to 8 artificially generated nonlinear series of different types, and found that the BDS test is the best and clearly stands out in terms of overall power against a variety of alternatives. The power of BDS test and some nonparametric tests have also recently been compared and applied to residual analysis of fitted models for monthly rainfalls by Kim et al. (2003), and the results also indicate the effectiveness of BDS test.

5.2.1 BDS Test

The BDS test (Brock et al., 1996) is a nonparametric method for testing for serial independence and nonlinear structure in a time series based on the correlation integral of the series. As stated by the authors, the BDS statistic has its origins in the work on deterministic nonlinear dynamics and chaos theory, it is not only useful in detecting deterministic chaos, but also serves as a residual diagnostic tool that can be used to test the goodness-of-fit of an estimated model. The null hypothesis is that the time series sample comes from an independent identically distributed (i.i.d.) process. The alternative hypothesis is not specified. In this section, the theoretical aspects of BDS test are presented.

Embed a scalar time series $\{x_t\}$ of length N into a m-dimensional space, and generate a new series $\{X_t\}$, $X_t = (x_t, x_{t-\tau}, ..., x_{t-(m-1)\tau})$, $X_t \in R^m$. Then, calculate the correlation integral $C_{m,M}(r)$ given by (Grassberger and Procacccia, 1983a):

$$C_{m,M}(r) = \binom{M}{2}^{-1} \sum_{1 \le i < j \le M} H\left(r - \left\| X_i - X_j \right\|\right) \tag{5.1}$$

where, $M = N - (m - 1)\tau$ is the number of embedded points in m-dimensional space; r the radius of a sphere centered on X_i; $H(u)$ is the Heaviside function, with $H(u) = 1$ for $u > 0$, and $H(u) = 0$ for $u \leqslant 0$; $\|\bullet\|$ denotes the sup-norm.

$C_{m,M}(r)$ counts up the number of points in the m-dimensional space that lie within a hypercube of radius r. Brock et al. (1996) exploit the asymptotic normality of $C_{m,M}(r)$ under the null hypothesis that $\{x_t\}$ is an i.i.d. process to obtain a test statistic which asymptotically converges to a unit normal.

If the series is generated by a strictly stationary stochastic process that is absolutely regular, then the limit $C_m(r) = \lim_{N\to\infty} C_{m,M}(r)$ exists. In this case the limit is

$$C_m(r) = \iint H(r - \| X - Y \|) dF_m(X) dF_m(Y) \tag{5.2}$$

where X and Y are two arbitrary points in the embedded m-dimensional space; F_m denotes the distribution function of the embedded time series $\{X_t\}$.

When the process is independent, and since $H(r - \| X_i - Y_j \|) = \prod_{k=1}^{m} H(r - | X_{i,k} - Y_{j,k} |)$, Equation (5.2) implies that $C_m(r) = C_1^m(r)$. Also $C_m(r) - C_1^m(r)$ has asymptotic normal distribution, with zero mean and variance given by

$$\frac{1}{4}\sigma_{m,M}^2(r) = m(m-2)C^{2m-2}(K - C^2) + K^m - C^{2m} + 2\sum_{j=1}^{m-1}\left[C^{2j}(K^{m-j} - C^{2m-2j}) - mC^{2m-2}(K - C^2)\right] \tag{5.3}$$

The constants C and K in Equation (5.3) can be estimated by

$$C_M(r) = \frac{1}{M^2}\sum_{i=1}^{M}\sum_{j=1}^{M} H(r - \| X_i - X_j \|),$$

and

$$K_M(r) = \frac{1}{M^3}\sum_{i=1}^{M}\sum_{j=1}^{M}\sum_{k=1}^{M} H(r - \| X_i - X_j \|) H(r - \| X_j - X_k \|).$$

Under the null hypothesis that $\{x_t\}$ is an i.i.d. process, the BDS statistic for $m > 1$ is defined as

$$BDS_{m,M}(r) = \sqrt{M}\,\frac{C_m(r) - C_1^m(r)}{\sigma_{m,M}(r)} \tag{5.4}$$

It asymptotically converges to a unit normal as $M \to \infty$. This convergence requires large samples for values of embedding dimension m much larger than 2, so m is usually restricted to the range from 2 to 5. Brock et al. (1991) recommended that r is set to between half and three halves the standard deviation σ of the data. We find that if r is set as half σ, there would be too few or even no nearest neighbors for many points in the embedded m-dimensional

space when m is large (e.g., m = 5), especially for series of short size (e.g., less than 100); on the other hand, when r is set as three halves σ, there would be too many nearest neighbors for many points in the embedded m-dimensional space when m is small (e.g., m = 2). Such kind of "shortage" of neighbors or "excess" of neighbors will probably bias the calculation of $C_{m,M}$ (r). Therefore, we only consider r equal to the standard deviation of the data in this study.

5.2.2 BDS Test Results for Streamflow Processes

Because usually linearity/nonlinearity tests (e.g., BDS test) assume the series of interest is stationary, it is necessary to test the stationarity before taking nonlinearity test. The stationarity test is carried out with two methods; one is augmented Dickey-Fuller (ADF) unit root test (Dickey and Fuller, 1979; Said and Dickey, 1984) and KPSS test (Kwiatkowski et al., 1992) in Chapter 2. The results show that among six groups of streamflow processes, except for the streamflow processes of the Yellow River at TG, all other series are basically stationary, albeit the daily series of the Yellow River at TNH is slightly nonstationary. Therefore, we will use five data sets (excluding the data set at TG) to make nonlinearity test.

BDS test needs the extraction of linear structure from the original series by the use of an estimated linear filter. Therefore, the first step for the test is fitting linear models to the streamflow series.

Because streamflow processes (except annual series) usually exhibit strong seasonality, to analysis the role of the seasonality played in nonlinearity test, the streamflow series are pre-processed in two ways, log-transformation and deseasonalization. Correspondingly, the pre-processed series are referred to as Log series and Log-DS series respectively. The Log-DS series is obtained with two steps. Firstly, log-transform the flow series. Then deseasonalize them by subtracting the seasonal (e.g., daily or monthly) mean values and dividing by the seasonal standard deviations of the log-transformed series. To alleviate the stochastic fluctuations of the daily and 1/3-monthly means and standard deviations, we smooth them with first 8 Fourier harmonics before using them for standardization. Annual series is analyzed with only log transformation. All series are pre-whitened with AR models. The parameters of AR models are estimated with Burg's algorithm (see e.g., Brockwell and Davis, 1991). The orders of the AR models are selected according to AIC as well as the PACF structure, shown in Table 5.1. Residuals are obtained from these models, and then the BDS test is applied to the residual series.

Table 5.1 Order of AR models used to pre-whiten streamflow series

Timescale	Yellow		Danube		Rhine		Umpqua		Ocmulgee	
	Log	Log-DS	Log	Log-DS	Log	Log-DS	Log	Log-DS	Log	Log-DS
daily	41	38	44	44	44	40	45	36	43	44
1/3-monthly	32	9	34	12	34	16	35	7	33	8
monthly	23	4	29	6	26	4	27	5	29	3
annual	3	-	6	-	6	-	8	-	9	-

Test results are shown in Table 5.2. It is noted that, according to the values of test statistics, with the increase of the timescale, the nonlinearity decreases. Among the flow series at four characteristic time scales, the strongest nonlinearity exhibits in daily series and no

nonlinearity exists in annual series. The result that all annual flow series are linear is in agreement with that of Rao and Yue (1990). Among the monthly series, Log series of Danube and Ocmulgee, and Log-DS series of Danube and Rhine pass the BDS test, while Log-DS series of Ocmulgee narrowly pass the test at significance level 0.05. But all the other series cannot pass BDS test at 0.01 significance level. Except for the daily and monthly streamflow series of Ocmulgee, and daily flow of Umpqua, there is a general feature that the test statistics of Log-DS series are smaller than those of the Log series, which implies that deseasonalization may more or less alleviate the nonlinearity.

Table 5.2 BDS test results for pre-whitened streamflow series

River (station)	Transform	Timescale	m = 2		m = 3		m = 4		m = 5	
			statistic	p-value	statistic	p-value	statistic	p-value	statistic	p-value
Yellow (TNH)	Log	Daily	47.0533	0	61.9571	0	74.1301	0	85.8811	0
		1/3-montly	10.5202	0	14.7704	0	19.0192	0	22.725	0
		Monthly	7.1791	0	7.9968	0	8.19	0	7.1412	0
		Annual	***-0.1481***	***0.6752***	***-0.5146***	***0.4806***	***-0.7211***	***0.3109***	***-1.1057***	***0.2688***
	Log-DS	Daily	43.4626	0	56.7527	0	68.4717	0	81.7653	0
		1/3-montly	6.044	0	8.1641	0	9.837	0	10.8218	0
		Monthly	2.8223	0.0048	3.0398	0.0024	3.143	0.0017	2.6285	0.0086
Danube (Achleiten)	Log	Daily	64.5929	0	71.4827	0	74.7811	0	78.3698	0
		1/3-montly	6.3608	0	7.1313	0	7.0309	0	6.7469	0
		Monthly	***0.4876***	***0.6258***	***0.524***	***0.6003***	***0.1309***	***0.8959***	***-0.3344***	***0.7381***
		Annual	***1.8646***	***0.0622***	***1.4325***	***0.152***	***0.7672***	***0.443***	***0.6296***	***0.5289***
	Log-DS	Daily	66.0556	0	74.0606	0	78.6511	0	83.3938	0
		1/3-montly	4.8753	0	5.403	0	4.7642	0	4.0397	0
		Monthly	***-0.433***	***0.665***	***-0.4672***	***0.6404***	***-0.6501***	***0.5156***	***-1.0611***	***0.2887***
Rhine (Lobith)	Log	Daily	81.3407	0	92.8541	0	99.0233	0	104.134	0
		1/3-montly	13.0673	0	15.9534	0	18.2171	0	20.4593	0
		Monthly	3.2478	0.0012	3.427	0.0006	3.2987	0.001	2.9891	0.0028
		Annual	***0.9354***	***0.3496***	***-0.3757***	***0.7072***	***-1.258***	***0.2084***	***-1.7697***	***0.0768***
	Log-DS	Daily	76.337	0	87.674	0	93.9964	0	99.3641	0
		1/3-montly	9.1439	0	9.7837	0	9.8396	0	9.9765	0
		Monthly	***1.0252***	***0.3053***	***0.6442***	***0.5195***	***0.3424***	***0.7321***	***0.0815***	***0.935***
Ocmulgee (Macon)	Log	Daily	39.7005	0	46.4219	0	50.3533	0	54.1384	0
		1/3-montly	8.6812	0	10.3209	0	11.6106	0	13.4153	0
		Monthly	***1.2264***	***0.22***	***0.6615***	***0.5083***	***0.7075***	***0.4793***	***0.8972***	***0.3696***
		Annual	***1.9058***	***0.0567***	***0.2874***	***0.7738***	***-0.0001***	***0.9999***	***-0.0924***	***0.9264***
	Log-DS	Daily	39.7893	0	46.5039	0	50.418	0	54.2164	0
		1/3-montly	6.0087	0	6.7851	0	7.0996	0	7.7285	0
		Monthly	2.1146	0.0345	***1.8856***	***0.0594***	***1.9118***	***0.0559***	***1.9514***	***0.051***
Umpqua (Elkton)	Log	Daily	82.7014	0	90.6942	0	94.1177	0	97.4194	0
		1/3-montly	20.3057	0	26.6761	0	32.5019	0	38.4548	0
		Monthly	6.4086	0	7.027	0	6.211	0	4.961	0
		Annual	1.3294	***0.1837***	***1.2578***	***0.2085***	***0.6557***	***0.512***	***-0.104***	***0.9172***
	Log-DS	Daily	82.7829	0	90.8138	0	94.2531	0	97.5695	0
		1/3-montly	13.9061	0	17.8505	0	20.5563	0	23.3343	0
		Monthly	3.0916	2.00E-03	3.6568	3.00E-04	3.8706	1.00E-04	3.6901	2.00E-04

With a close inspection of the pre-whitened streamflow series, i.e., the residual series, we find that although the residuals are serially uncorrelated, there is seasonality in the variance of the residual series (see Figure 4.9 in Chapter 4). Therefore, it's worthwhile to have a look at the

residuals after removing such kind of season-dependent variance. Table 5.3 shows the BDS test results for the residual series after being standardized by dividing by the seasonal standard deviations of the residuals.

Table 5.3 BDS test results for standardized pre-whitened streamflow series

River (station)	Transform	Time scale	m = 2		m = 3		m = 4		m = 5	
			statistic	p-value	statistic	p-value	statistic	p-value	statistic	p-value
Yellow (TNH)	Log-std	Daily	36.5463	0	46.8049	0	55.9885	0	63.2352	0
		1/3-montly	3.2901	1.00E-03	3.9963	1.00E-04	4.8369	0	5.0091	0
		Monthly	3.3106	9.00E-04	3.6546	3.00E-04	3.8527	1.00E-04	3.6088	3.00E-04
	Log-DS-std	Daily	39.4409	0	46.3156	0	50.7325	0	54.8716	0
		1/3-montly	***1.572***	***0.116***	***1.9548***	***0.0506***	***1.8772***	***0.0605***	***1.3839***	***0.1664***
		Monthly	***0.2841***	***0.7763***	***0.0009***	***0.9993***	***0.2121***	***0.8321***	***0.33***	***0.7414***
Danube (Achleiten)	Log-std	Daily	64.1614	0	70.781	0	73.5179	0	76.5167	0
		1/3-montly	5.013	0	5.5489	0	5.1878	0	4.6661	0
		Monthly	***-0.4842***	***0.6283***	***-0.5164***	***0.6056***	***-0.8201***	***0.4122***	***-1.134***	***0.2568***
	Log-DS-std	Daily	64.314	0	70.8533	0	73.5682	0	76.3846	0
		1/3-montly	4.0241	0	4.5249	0	3.9327	0	3.2051	0
		Monthly	***-0.5095***	***0.6104***	***-0.5129***	***0.608***	***-0.6604***	***0.509***	***-0.9927***	***0.3209***
Rhine (Lobith)	Log-std	Daily	76.0294	0	88.0105	0	94.9959	0	101.1317	0
		1/3-montly	6.0181	0	5.5306	0	4.8313	0	4.2881	0
		Monthly	***-0.1027***	***0.9182***	***-0.4843***	***0.6282***	***-0.4602***	***0.6454***	***-0.3668***	***0.7138***
	Log-DS-std	Daily	76.0294	0	88.0105	0	94.9959	0	101.1317	0
		1/3-montly	6.9796	0	6.5555	0	5.7393	0	5.0276	0
		Monthly	***0.3267***	***0.7439***	***-0.2032***	***0.839***	***-0.4073***	***0.6838***	***-0.6029***	***0.5466***
Ocmulgee (Macon)	Log-std	Daily	39.2541	0	45.9753	0	49.8505	0	53.5797	0
		1/3-montly	8.1499	0	9.351	0	10.4597	0	12.1685	0
		Monthly	***1.1009***	***0.271***	***0.4957***	***0.6201***	***0.5591***	***0.5761***	***0.8345***	***0.404***
	Log-DS-std	Daily	39.339	0	46.0532	0	49.9094	0	53.6486	0
		1/3-montly	5.495	0	5.8351	0	5.99	0	6.6117	0
		Monthly	***1.839***	***0.0659***	***1.6546***	***0.098***	***1.6696***	***0.095***	***1.6642***	***0.0961***
Umpqua (Elkton)	Log-std	Daily	79.2755	0	87.0849	0	90.5519	0	93.9577	0
		1/3-montly	11.4493	0	12.9663	0	13.1782	0	13.3861	0
		Monthly	3.3133	9.00E-04	3.8847	1.00E-04	4.0829	0.00E+00	3.9246	1.00E-04
	Log-DS-std	Daily	79.4946	0	87.334	0	90.8304	0	94.294	0
		1/3-montly	10.8964	0	12.6817	0	13.1041	0	13.604	0
		Monthly	2.2041	0.0275	2.7437	0.0061	3.0048	0.0027	3.0525	0.0023

Note: "-std" in clolumn "transform" refers to the standardization of pre-whitened streamflow series.

Comparing Table 5.2 and Table 5.3, we can find that, the BDS test statistics of all the series are generally smaller than those of the series before standardization. Especially, 1/3-montly and monthly Log-DS series of the Yellow River, and the monthly Log series of the Rhine River, which are nonlinear before standardization, pass the BDS test at 0.05 significance level after standardization. Therefore, the seasonal variation in variance in the residuals is probably a dominant source of nonlinearity in the 1/3-montly and monthly Log-DS series of the Yellow River, and the monthly Log series of the Rhine River. But all the daily series, most 1/3-monthly series and some monthly series still exhibit nonlinearity even after standardization. That indicates that the seasonal variance composes only a small, even negligible, fraction of the nonlinearity underlying these processes, especially daily streamflow processes.

The BDS test results are summarized in Table 5.4. The summary gives us a clear indication of how the degree of nonlinearity changes with the change of timescale and the change of how we process the flow processes. In general, the following two points can be revealed. In general two points are revealed by the BDS test results:

(1) There are stronger and more complicated nonlinearity mechanisms acting at small timescales than at large timescales. As the timescale increases, the nonlinearity weakens.

(2) Asymmetric seasonality in the mean and variance of raw (or log-transformed) streamflow processes and the seasonality in the variance of the pre-whitened streamflow processes (i.e., residual series) play a role in the exhibition of nonlinearity. The effects of seasonal variation in variances dominate the nonlinearity of some 1/3-monthly and monthly streamflow series.

Table 5.4 Summary of BDS test results

Timescale	Transform	Linear	Non-linear	Average test statistic
Daily	Log	0	5	74.557
	Log-DS	0	5	73.317
	Log-std	0	5	69.362
	Log-DS-std	0	5	68.878
1/3-monthly	Log	0	5	16.198
	Log-DS	0	5	9.798
	Log-std	0	5	7.466
	Log-DS-std	1	4	6.050
Monthly	Log	2	3	3.652
	Log-DS	3	2	1.926
	Log-std	3	2	1.850
	Log-DS-std	4	1	1.144
Annual	Log	5	0	0.858

5.2.3 Analysis of the Power of BDS Test

Although most monthly flow series and some 1/3-monthly series are diagnosed as linear with BDS test after, or even before, being standardized by seasonal variance, this does not exclude the possibility that there exists some weak nonlinearity in these series. For example, some studies indicate that monthly streamflow could be modelled by TAR model or PAR model (e.g., Thompstone et al., 1985). TAR model is a well-acknowledged nonlinear model. PAR model is also a nonlinear model, which differs from TAR model in that TAR model uses observed values as threshold whereas PAR model uses season as threshold. Passing BDS test does not mean that there is no nonlinearity such as TAR or PAR mechanism present in the time series. It is possible that BDS test is not powerful enough to detect weak nonlinearity. We will make an analysis on the power of BDS test in this section with some simulated series.

Consider one AR model, two TAR models, two bilinear models and Henon map series of the following form:

(1) AR: $x_t = 0.7x_{t-1} + \varepsilon_t$;

(2) TAR-1: $\begin{cases} x_t = 0.9x_{t-1} + \varepsilon_t & for\ x_{t-1} < 1 \\ x_t = 0.3x_{t-1} + \varepsilon_t & for\ x_{t-1} \geq 1 \end{cases}$;

(3) TAR-2: $\begin{cases} x_t = 0.9x_{t-1} + \varepsilon_t & for\ x_{t-1} < 1 \\ x_t = -0.3x_{t-1} + \varepsilon_t & for\ x_{t-1} \geq 1 \end{cases}$;

(4) Bilinear-1: $x_t = 0.9\ x_{t-1} + 0.1\ x_{t-1} * \varepsilon_{t-1} + \varepsilon_t$;

(5) Bilinear-2: $x_t = 0.4x_{t-1} + 0.8\ x_{t-1} * \varepsilon_{t-1} + \varepsilon_t$;

(6) Henon map series: $\begin{cases} x_{t+1} = 1 - ax_t^2 + by_t \\ y_{t+1} = x_t \end{cases}$, a=1.4, b=0.3.

In all the above models, $\{x_t\}$ (or $\{y_t\}$) is time series, and $\{\varepsilon_t\}$ is independent standard normal error. Obviously, among the above models, model TAR-1 and Bilinear-1 have weak nonlinearity while model TAR-2 and Bilinear-2 have stronger nonlinearity, because TAR-2 has a larger parameter difference and Bilinear-2 has a more significant bilinear item. Henon map series is a typical chaotic series (Henon, 1976). For model (1) to (5), 1000 simulations are generated, and each simulation has 500 points. For Henon series, one simulation with 500000 points is generated (referred to as clean-Henon in Table 5.5). Then the Henon series is divided into 1000 segments, and each segment has 500 points. To evaluate the influence of noise on BDS test, noise is added to the simulated Henon series (referred to as noise-Henon in Table 5.5). The noise is normally distributed with zero mean, and its standard deviation is 5% of the standard deviation of the Henon series.

Table 5.5 Rates of accepting linearity with BDS test based on 1000 replications at 0.05 level

Series	m = 2		m = 3		m = 4		m = 5	
	p-value	accepted	p-value	accepted	p-value	accepted	p-value	accepted
AR(1)	0.483	926	0.484	925	0.484	928	0.462	933
TAR-1	0.361	831	0.371	838	0.380	840	0.403	843
TAR-2	0.002	212	0.004	272	0.009	326	0.014	373
Bilinear-1	0.178	703	0.196	714	0.238	735	0.264	763
Bilinear-2	6.73E-40	0	5.98E-46	0	3.62E-47	0	3.62E-47	0
clean-Henon	7.24E-50	0	3.27E-84	0	3.50E-115	0	2.25E-142	0
noise-Henon	3.61E-49	0	5.32E-82	0	5.83E-112	0	5.92E-138	0

Note: p-value in the table is the median value for each group of 1000 replications.

Then we use BDS test to detect the presence of nonlinearity in the simulated series. All the series are pre-whitened with AR models. The test results are shown in Table 6. It is shown that the hypothesis of linearity for Henon series (pure or with noise) are firmly rejected, which indicates that BDS test is very powerful for detecting such kind of strong nonlinearity. In most cases, BDS test correctly rejects the hypothesis that TAR-2 and Bilinear-2 processes are linear, but wrongly accepts TAR-1 and Blinear-1 processes as linear. That means that although BDS test is considered very powerful for testing nonlinearity, but not powerful enough for detecting weak nonlinearity in TAR-1 and Bilinear-1, whereas such kinds of weak nonlinearity probably present in the streamflow series, because it is impossible that

streamflow processes are driven by the mechanism like TAR-2, which switches between dramatically different regimes.

Therefore, BDS test results tell us that there is strong nonlinearity present in daily streamflow series as well as most 1/3-monthly series, even after taking away the effects of seasonal variance, but there is no strong nonlinearity presents in most monthly streamflow series and some 1/3-monthly series after removing the effects of seasonal variance. However, we cannot say there is no nonlinearity present in those 1/3-monthly and monthly streamflow series even if they pass BDS test, because BDS test is not powerful enough for detecting weak nonlinearity. In addition, comparing the BDS test results for chaotic Henon series with those for streamflow series, while it is not clear whether most 1/3-monthly series and all the daily series have chaotic properties, it seems that all monthly series may not be chaotic because the BDS test p-values for monthly flow series are far much higher than those for chaotic Henon series. We would further detect the existence of chaos with the correlation exponent method in the next section.

5.3 Testing for Chaos in Streamflow Processes

Many methods are available for detecting the existence of chaos, among which the correlation exponent method (e.g. Grassberger and Procaccia, 1983a), the Lyapunov exponent method (e.g. Wolf et al., 1985; Kantz,, 1994), the Kolmogorov entropy method (e.g. Grassberger and Procaccia, 1983b), the nonlinear prediction method (e.g. Sugihara and May, 1990), and the surrogate data method (e.g., Theiler et al., 1992; Schreiber and Schmitz, 1996) are commonly used. However, many researchers (e.g, Sivakumar, 2000) have noticed that there is no reliable method to clearly distinct a chaotic system and a stochastic system. A finite correlation dimension may be observed not only for chaotic processes but also for a stochastic process (e.g. Osborne and Provenzale, 1989), and when examined by the Grassberger-Procaccia algorithm alone, filtered noise can mimic low-dimensional chaotic attractors (Rapp et al, 1993. The occurrence of a positive time-average Lyapunov exponent in a nonlinear system subject to noise, be additive or multiplicative, does not necessarily imply deterministic chaos (Van den Broeck and Nicolis, 1993), and a positive Lyapunov exponent may be observed also for stochastic processes, such as ARMA processes (e.g. Jayawardena and Lai, 1994). Random noises with power law spectra may provide convergence of the Kolmogorov entropy, which implies that the observation of a finite or null value of the K_2 entropy in the analysis of data is not enough to infer that the system is dominated by a deterministic process such as low-dimensional chaos. (e.g. Provenzale et al., 1991). A stochastic process, be it linear or nonlinear, can also produce accurate short-term prediction but not long-term predictability, which is a typical characteristic of a chaotic process. Surrogate data method, which has a typical null hypothesis of Gaussian linear stochastic process, is not specifically designed for detecting chaos. In addition, phase-randomized surrogates can produce spurious identifications of non-random structure (e.g. Rapp et al., 1994). Consequently, none of the above methods can provide a conclusive resolution of whether a given data set is chaotic. What makes it worse for testing for chaos is the inevitable presence of noise in the observed data sets (e.g., Schertzer, 1997), such as hydrological time series.

While we cannot prove the existence of chaos with the above methods, we can conclude the absence of chaos if there is no finite correlation dimension for the data set of interest, because it is well recognized that chaotic processes should have finite correlation dimension, provided that the data set is of sufficient size. Therefore, it is crucial to estimate correlation dimension for detecting the presence or absence of chaos. For this reason, correlation exponent method is used by almost all the researchers for detecting chaos in hydrological processes (e.g., Jayawardena and Lai, 1994; Porporato and Ridolfi, 1997; Pasternack, 1999; Bordignon and Lisi, 2000; Elshorbagy et al., 2002, Khan et al., 2005). In this study, the analysis based on the correlation exponent method is done with TISEAN package (Hegger et al., 1999).

When testing for general nonlinearity, it is common to filter the data to remove linear correlations (prewhitening) (e.g., Brock et al., 1996), because linear autocorrelation can give rise to spurious results in algorithms for estimating nonlinear invariants, such as correlation dimension and Lyapunov exponents. But it has been observed that in numerical practice prewhitening may severely impairs the underlying deterministic nonlinear structure of low-dimensional chaotic time series (e.g., Theiler and Eubank, 1993; Sauer and Yorke, 1993). Therefore, mostly chaos analyses are based on original series, and the same in our analysis.

Correlation exponent method is most frequently employed to investigate the existence of chaos. The basis of this method is multi-dimension state space reconstruction. The most commonly used method for reconstructing the state space is the time-delay coordinate method proposed by Packard et al. (1980) and Takens (1981). In the time delay coordinate method, a scalar time series $\{x_1, x_2, \ldots, x_N\}$ is converted to state vectors $X_t = [x_t, x_{t-\tau}, \ldots, x_{t-(m-1)\tau}]$ after determining two state space parameters: the embedding dimension m and delay time τ . To check whether chaos exists, the correlation exponent values are calculated against the corresponding embedding dimension values. If the correlation exponent leads to a finite value as embedding dimension increasing, then the process under investigation is thought of as being dominated by chaotic dynamics. Otherwise, the process is considered as non-chaotic.

To calculate the correlation exponent, the delay time τ should be determined first. Therefore, the selection of delay time is discussed first in the following section, followed by the estimation of correlation dimension.

5.3.1 Selection of Delay Time

The delay time τ is commonly selected by using the autocorrelation function (ACF) method where ACF first attains zeros or below a small value (e.g., 0.2 or 0.1), or the mutual information (MI) method (Fraser and Swinney, 1986) where the MI first attains a minimum. We first take the streamflow of the Yellow River at TNH as an example to analyze the choice of τ .

We calculate ACF and MI of daily, 1/3-monthly and monthly flow series of the Yellow River, shown in Figure 5.1. Because of strong seasonality, ACF first attains zeros at the lag time of about 1/4 period, namely, 91, 9 and 3 for daily, 1/3-monthly and monthly series respectively. The MI method gives similar estimates for τ to the ACF method, about approximately 1/4 annual period.

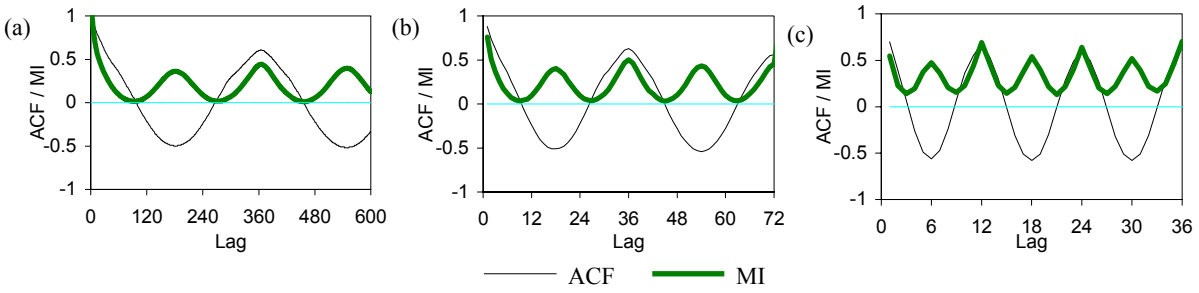


Figure 5.1 ACF and MI of (a) daily, (b) 1/3-monthly and (c) monthly river flow of the Yellow River

In practice, the estimate of τ is usually application and author dependent nonetheless in practice. For instance, for daily flow series, some authors take the delay time as 1 day (Porporato and Ridolfi 1997), 2 days (Jayawardena and Lai, 1994), 7 days (Islam and Sivakumar, 2002), 10 days (Elshorbagy et al., 2002), 20 days (Wilcox et al., 1991) and 146 days (Pasternack, 1999). These differences may arise from different ACF structure. To compare the influence of different τ on the reconstruction of state space, we can plot $x_t \sim x_{t+\tau}$ state-space maps for the streamflow series with different τ. The best τ value should make the state space best unfolded. For the streamflow series of the Yellow River, the $x_t \sim x_{t+\tau}$ state-space maps with small τ values (i.e., 1, 7, 10, and 20) are displayed in Figure 5.2, and the 2- and 3-dimensional $x_t \sim x_{t+\tau}$ state-space maps with τ taken as 1/4 of the annual period are displayed in Figure 5.3. Obviously, especially clearly in the 3-D maps, state spaces for daily, 1/3-monthly and monthly streamflow series are best unfolded when delay time τ = 91, 9, 3 respectively.

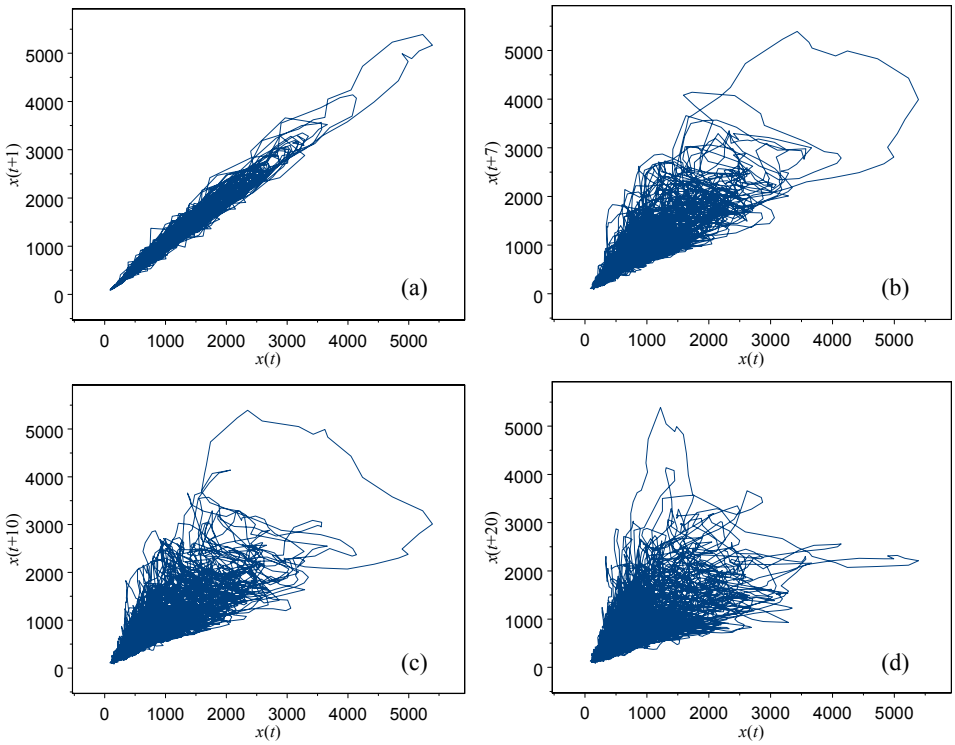


Figure 5.2 $x_t \sim x_{t+\tau}$ state-space maps of daily streamflow series of the Yellow River at TNH with (a) τ =1; (b) τ =7; (c) τ =10; (d) τ =20

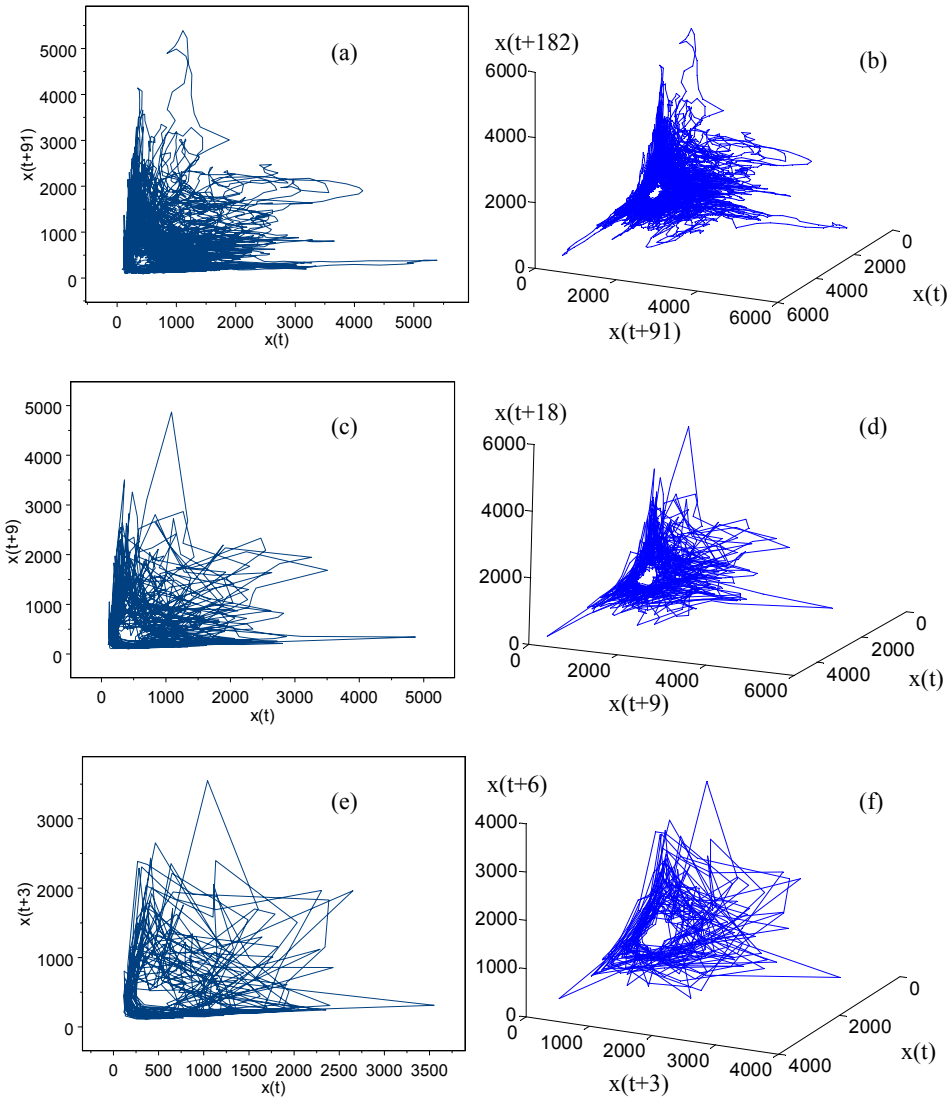


Figure 5.3 2-D and 3-D state space maps of (a), (b) daily; (c), (d) 1/3-monthly; and (e), (f) monthly streamflow of the Yellow River at TNH with delay time τ = 91, 9 and 3.

We therefore select τ = 91, 9, 3 for estimating correlation dimension for the streamflow series of the Yellow River. Similar results are obtained for the streamflow processes of the Umpqua River and the Ocmulgee River (to save space, the plots are not displayed here). But for the Rhine River, the seasonality is not that obvious. The ACF and MI of daily, 1/3-monthly and monthly flow series of the Rhine River are shown in Figure 5.4. If we determine the delay time according to the lags where ACF attains 0 or MI attains its minimum for the Rhine River, the lags would be about 200 days which seems to be too large, which would possibly make the successive elements of the state vectors in the embedded multi-dimensional state space almost independent. Therefore we select the delay time equal to the lags before ACF attains 0.1, namely, τ = 92, 9, 3 for daily, 1/3-monthly and monthly streamflow series, respectively.

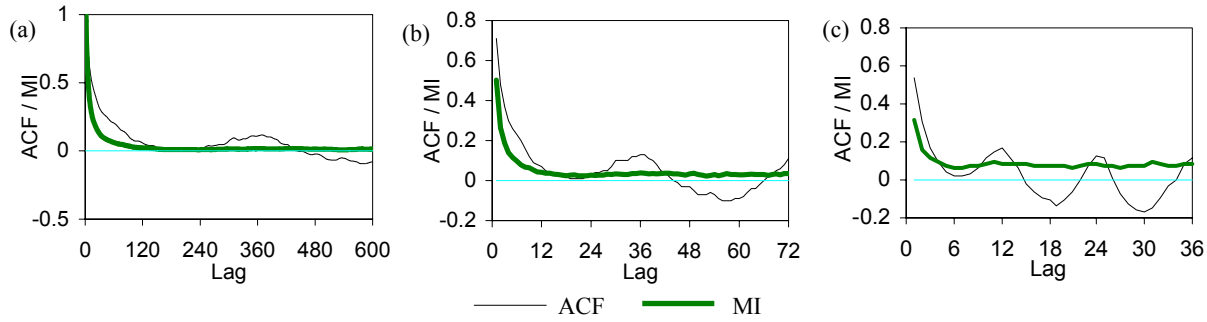


Figure 5.4 ACF and MI of (a) daily, (b) 1/3-monthly and (c) monthly river flow of the Rhine River

5.3.2 Estimation of the Correlation Dimension

The most commonly used algorithm for computing the correlation dimension is the Grassberger - Procaccia algorithm (Grassberger and Procaccia, 1983a), modified by Theiler (1986). For a m-dimension phase-space, the modified correlation integral $C(r)$ is defined by (Theiler, 1986)

$$C(r) = \frac{2}{(M+1-w)(M-w)} \sum_{i=1}^{M} \sum_{j=i+w+1}^{M-i} H(r - \|X_i - X_j\|) \tag{5.5}$$

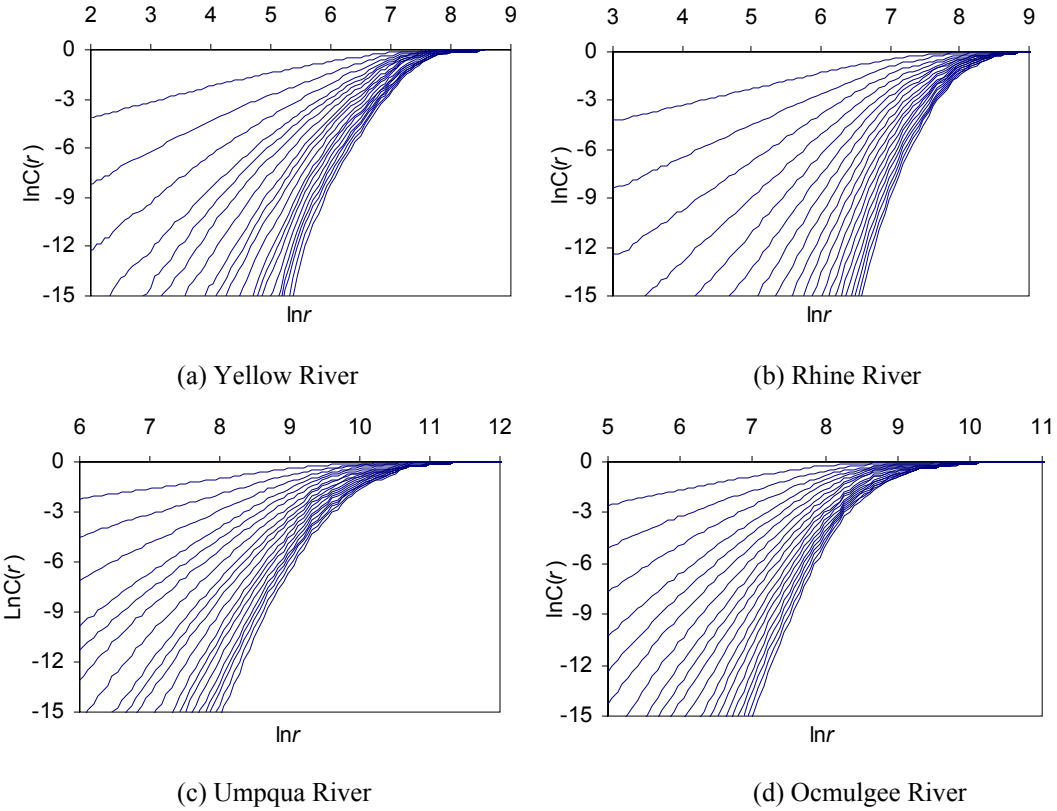
where M, r, H have the same meaning as in Equation (5.1), w (≥ 1) is the Theiler window to exclude those points which are temporally correlated. In this study, w is set as about half a year, namely 182, 18, and 6 for daily, 1/3-monthly and monthly series respectively.

For a finite dataset, there is a radius r below which there are no pairs of points, whereas at the other extreme, when the radius approaches the diameter of the cloud of points, the number of pairs will increase no further as the radius increases (saturation). The scaling region would be found somewhere between depopulation and saturation. When $\ln C(r)$ versus $\ln r$ is plotted for a given embedding dimension m, the range of $\ln r$ where the slope of the curve is approximately constant is the scaling region where fractal geometry is indicated. In this region $C(r)$ increase as a power of r, with the scaling exponent being the correlation dimension D. If the scaling region vanishes as m increases, then finite value of correlation dimension cannot be obtained, and the system under investigation is considered as non-chaotic. Local slopes of $\ln C(r)$ versus $\ln r$ plot can show scaling region clearly when it exists. Because the local slopes of $\ln C(r)$ versus $\ln r$ plot often fluctuate dramatically, to identify the scaling region more clearly, we can use the Takens-Theiler estimator or smooth Gaussian kernel estimator to estimate correlation dimension (Hegger et al., 1999).

The $\ln C(r)$ versus $\ln r$ plots of daily, 1/3-monthly and monthly streamflow series of the four rivers are displayed in Figure 5.5 – 5.7 respectively, and the Takens-Theiler estimates (D_{TT}) of correlation dimension are displayed in Figure 5.8 – 5.10.

We cannot find any obvious scaling region from the Figure 5.8 – 5.10. Take the Yellow River for instance, an ambiguous $\ln r$ region could be identified as scaling region is around $\ln r$ = 7~7.5 for the three flow series of different timescales. But in this region, as shown in Figure

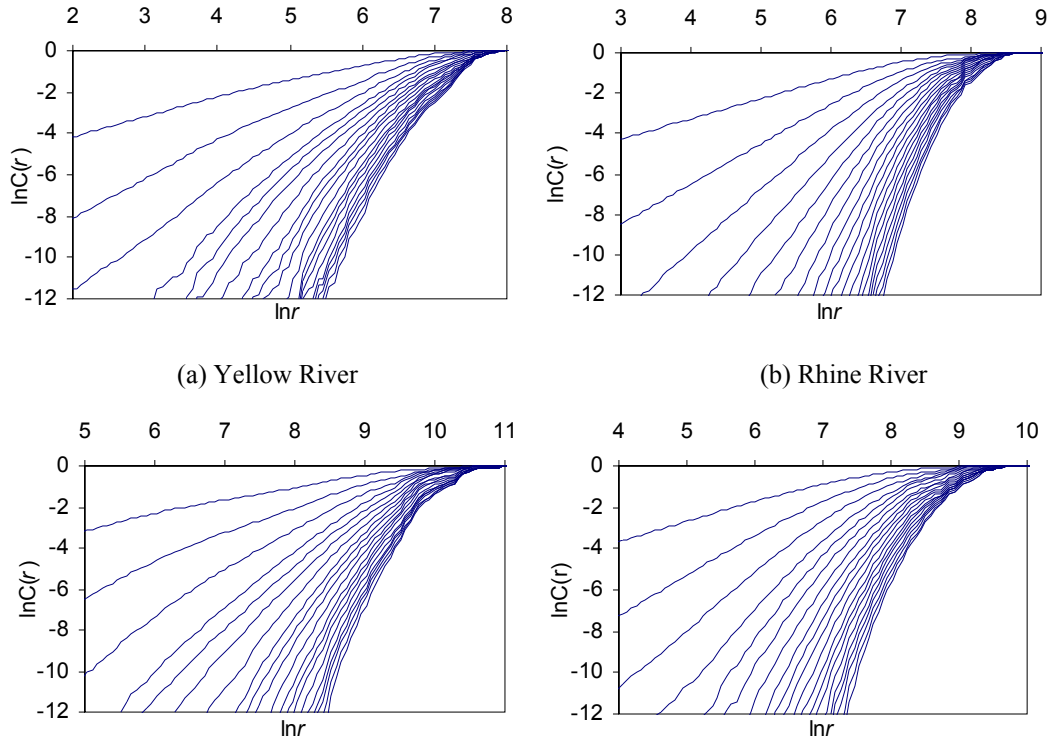
5.11, the D_{TT} increases with the increment of the embedding dimension, which indicates that the system under investigation is non-chaotic.



(a) Yellow River (b) Rhine River

(c) Umpqua River (d) Ocmulgee River

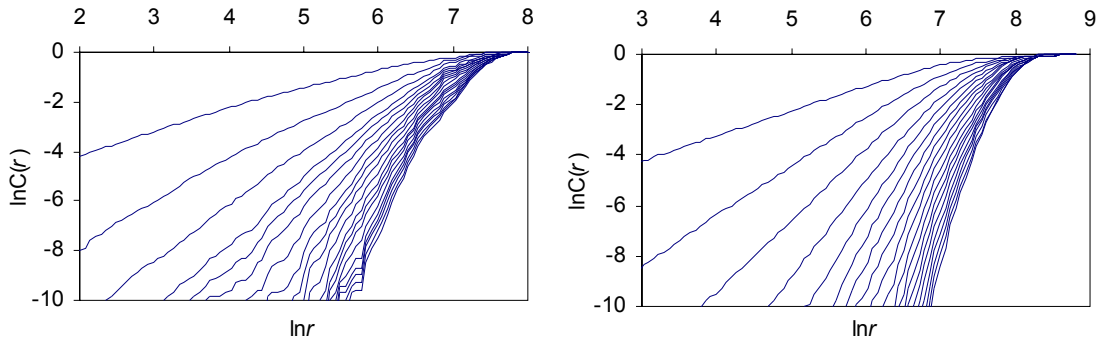
Figure 5.5 ln$C(r)$ versus ln$r$ plot for daily streamflow processes



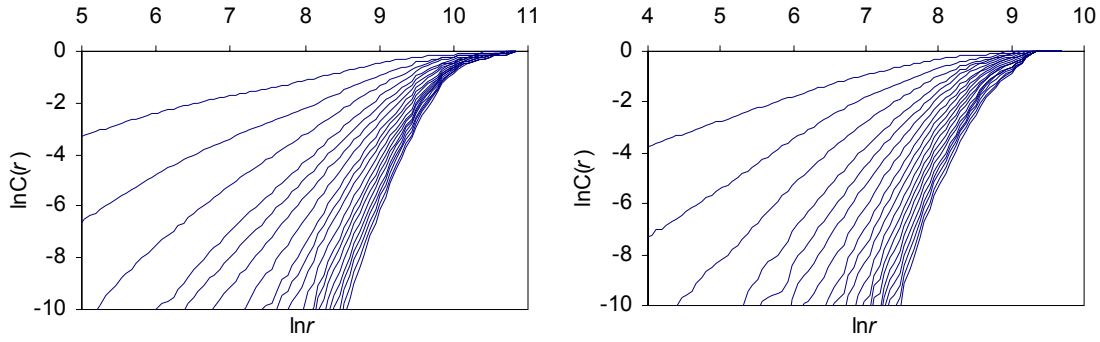
(a) Yellow River (b) Rhine River

(c) Umpqua River (d) Ocmulgee River

Figure 5.6 ln$C(r)$ versus ln$r$ plots for 1/3-monthly streamflow processes

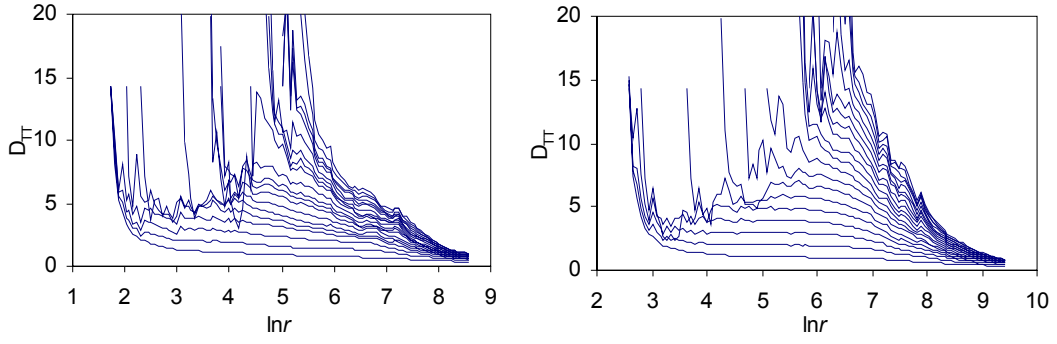


(a) Yellow River (b) Rhine River

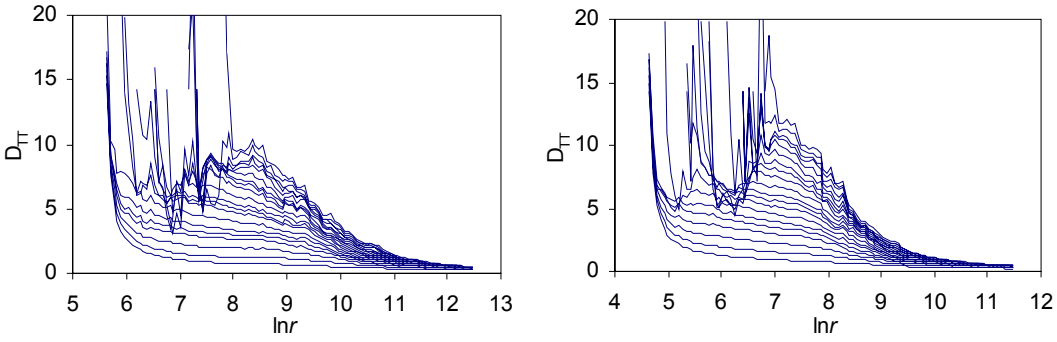


(c) Umpqua River (d) Ocmulgee River

Figure 5.7 lnC(*r*) versus ln*r* plots for monthly streamflow processes

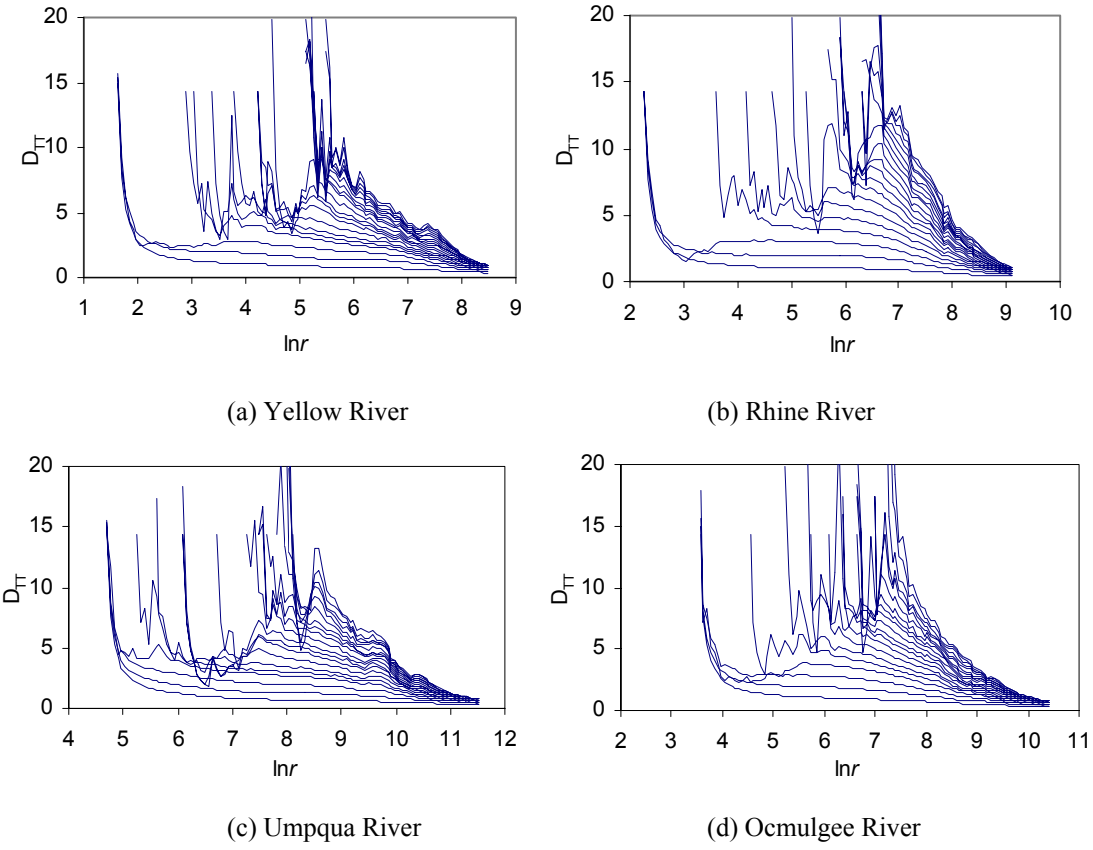


(a) Yellow River (b) Rhine River



(c) Umpqua River (d) Ocmulgee River

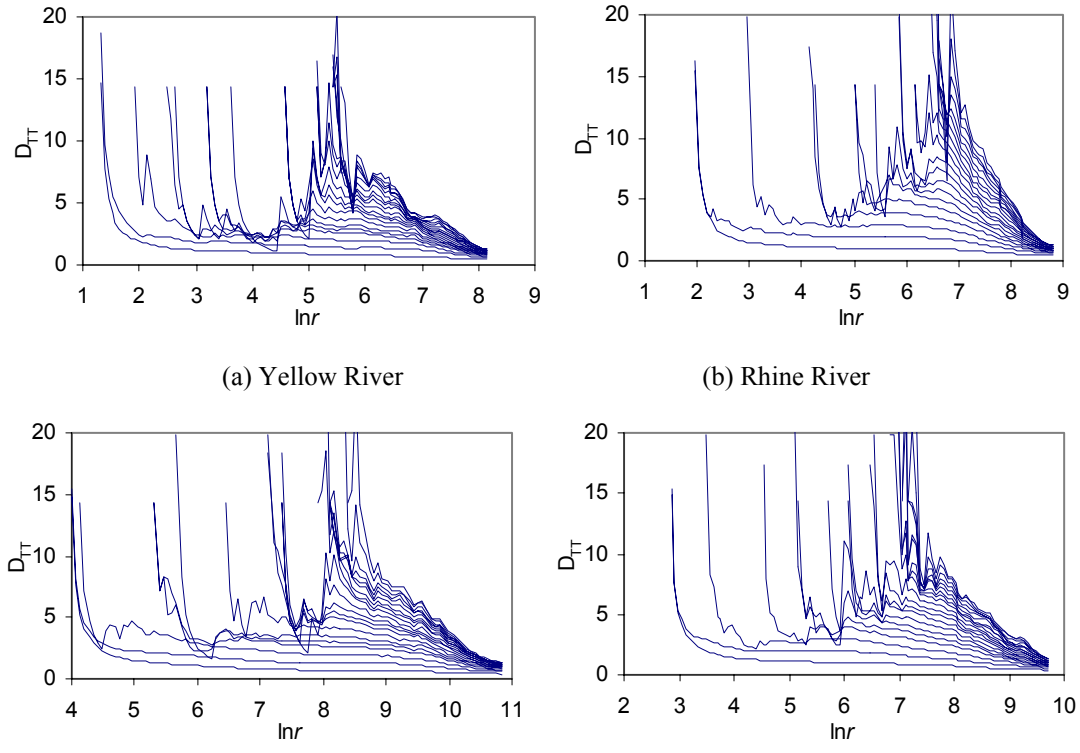
Figure 5.8 Takens-Theiler estimates of correlation dimension for daily streamflow processes



(a) Yellow River (b) Rhine River

(c) Umpqua River (d) Ocmulgee River

Figure 5.9 Takens-Theiler estimates of correlation dimension for 1/3-monthly streamflow processes



(a) Yellow River (b) Rhine River

(c) Umpqua River (d) Ocmulgee River

Figure 5.10 Takens-Theiler estimates of correlation dimension for monthly streamflow processes

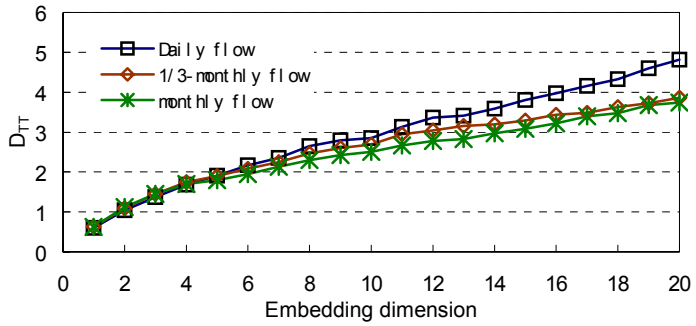


Figure 5.11 Relationship between D_{TT} and the embedding dimension for the streamflow of the Yellow River at TNH

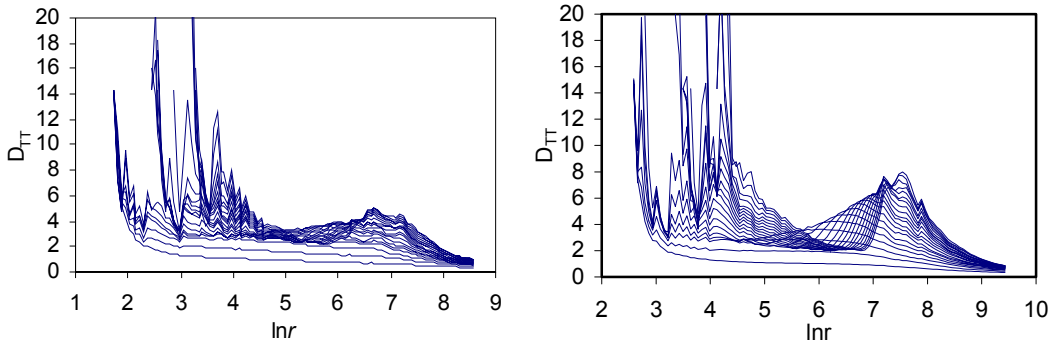
5.3.3 Discussion on the Estimation of Correlation Dimension

Three issues regarding the estimation of correlation dimension should be noticed.

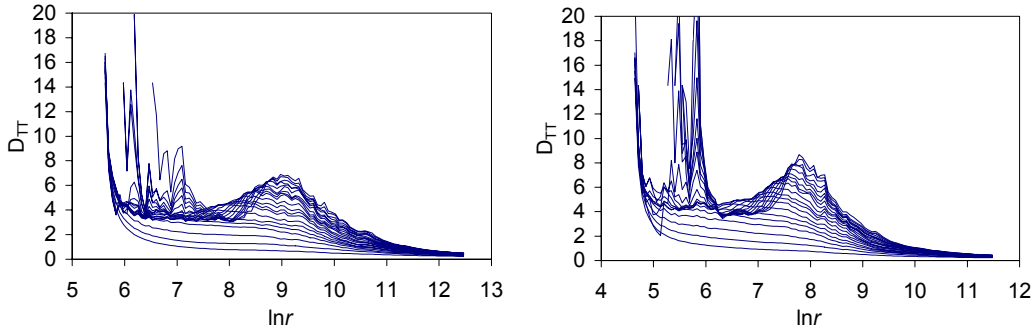
First, about the minimum data size for estimating correlation dimension. Some authors claim that the size of 10^A (Procaccia, 1988) or $10^{(2+0.4m)}$ (Nerenberg and Essex, 1990; Tsonis et al., 1993), where A is the greatest integer smaller than correlation dimension and m is the embedding dimension, is needed for estimating correlation dimension with an error less than 5%. On the other hand, some researchers found that smaller data size is required. For instance, the minimum data points for reliable correlation dimension D is $10^{D/2}$ (Eckmann and Ruelle, 1992), or $\sqrt{2}\bullet\sqrt{27.5}^{D}$ (Hong and Hong, 1994), or $10^{(D+2)/2}$ (Kantz and Schreiber, 2003, pp 167-168), or 5^m to keep the edge effect error in the correlation dimension estimation below 5% (Theiler, 1990), and empirical results of dimension calculations are not substantially altered by going from 3 000 or 6 000 points to subsets of 500 points (Abraham et al., 1986). In our study, data length is long enough for estimating correlation dimension for daily flow, but the data size used for monthly streamflow analysis seems short, especially the size of 540 points of monthly flow series of the Yellow River. However, as shown in Figure 5.5 – 5.10, there is no significant difference among the behavior of correlation integrals of the flow series with different sampling frequency. The agreement among the behavior of correlation integrals for daily, 1/3-monthly and monthly flow series indicates that it is possible to make basically reliable correlation dimension calculation with a series of size as short as 540. This is consistent with the empirical result of Abraham et al. (1986) and approximately satisfying the theoretical minimum size of Hong and Hong (1994) or Kantz and Schreiber (2003) if there is indeed a finite correlation dimension less than 3.5.

Second, about scaling region. Some authors do not provide scaling plot when investigating the existence of chaos (e.g., Jayawardena and Lai, 1994; Sivakumar, 1999; Elshorbagy et al., 2002), whereas some other authors provide scaling plot, but give no obvious scaling region (e.g., Porporato and Ridolfi 1997). However, a clearly discernible scaling region is crucial to make a convincing and reliable estimate of correlation dimension (Kantz and Schreiber, 2003, pp 82-87).

Third, about temporally related points for computing $C(r)$. To exclude temporally related points from the computation of $C(r)$, the Theiler window as in Equation (5.5) is indispensable. Grassberger (1990) remarked that when estimating the dimension of an attractor from a time sequence, one has to make sure that there exist no dynamical correlations between data points, so that all correlations are due to the geometry of the attractor rather than due to short-time correlations. He urged the reader to be very generous with the Theiler window parameter. Because streamflow series is highly temporally related, especially for daily flow, therefore, without setting Theiler window w (i.e., $w = 1$), we would find a spurious scaling region in the plot of D_{TT} versus ln$r$ which gives an incorrect estimate of correlation dimension. This problem has been pointed out by Wilcox et al. (1991) a decade ago, however, some authors ignored this (e.g., Elshorbagy et al., 2002), and some others take a very small Theiler window, which is maybe not large enough to exclude temporal correlations between the points (for example, Porporato and Ridolfi (1996) take $w = 5$ for daily flow series). Figure 5.12 shows the Takens-Theiler's estimate for daily streamflow series of the four rivers with w set to be 1. It is clear that with $w = 1$, we would find spurious scaling regions in all these plots. Furthermore, comparing the plots for the daily streamflow of the Rhine river with different values of w, namely, Figure 5.12(b), Figure 5.13 (a) and (b), we can further find that the smaller the value of w, the lower the estimated correlation dimension. According to these plots, when $w = 1$, the correlation dimension D is less than 4; when $w = 5$, D is less than 8, and when $w = 15$, D is less than 10. Therefore, the dimension estimate could be seriously too low if temporal coherence in the time series is mistaken for geometrical structure (Kantz and Schreiber, 2003, pp 87-91).

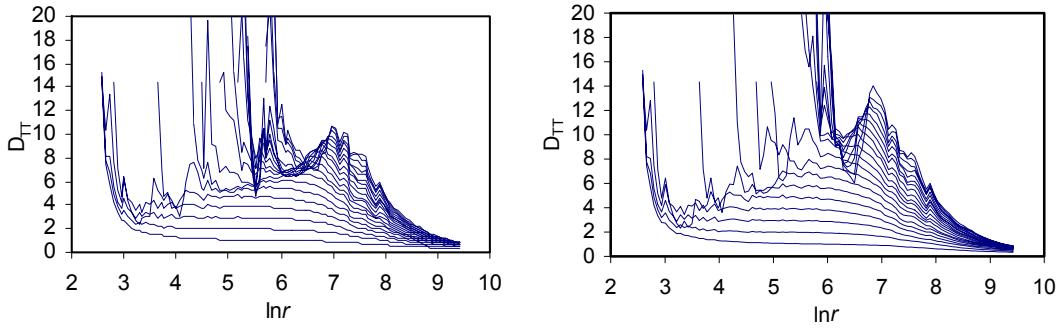


(a) Yellow River (b) Rhine River



(c) Umpqua River (d) Ocmulgee River

Figure 5.12 Takens-Theiler estimates without considering Theiler window for daily streamflow



(a) w = 5 (b) w = 15

Figure 5.13 Takens-Theiler estimates with small Theiler window for daily streamflow of the Rhine River

5.4 Effects of Dynamical Noises on the Identification of Chaotic Systems

When analyzing the chaos properties in observational time series, we cannot avoid the problem of noise. There are two distinct types of noise: (1) measurement noise, which refers to the corruption of observations by errors that does not influence the evolution of the dynamical system; and (2) dynamical noise, which perturbs the system more or less at each time step. In the presence of dynamical noises, the time series is not a simple superposition of signal plus noise, but rather a signal modulated by the noise. With regard to an observed hydrological series, its dynamics is inevitably contaminated not only by measurement noise, but also more significantly by dynamical noise, such as the disturbance of storm rain. What are the effects of dynamical noise on the estimation of characteristic quantities of chaotic systems? We will discuss this issue through experiments with three well-known chaotic systems: (1) Henon map (Henon, 1976), which has one attractor with an attraction basin nearly touching the attractor in several places; (2) Ikeda map (Ikeda, 1979), which has one chaotic attractor with a small atttaction basin and a non-chaotic attractor with much larger attraction basin; (3) Mackey –Glass flow (Mackey and Glass, 1977), which has one attractor with unbounded attraction basin.

To analyze the effects of dynamical noises, we add different levels of dynamical noises to the known chaotic systems by adding a noise item to the equations, and then check the phase state portraits and correlation dimensional estimates. The noise we add to the chaotic series are independently identically distributed Gaussian noise, with zero mean and variance of 2%, 5%, 10% and 100% of the original pure chaotic series. Besides the Gaussian noise, autoregressive (AR) dynamical noises of AR(1) structure have also been tested. Results show that, due to the impacts of the serial dependence of AR noises, the impacts of the AR noise on the dynamics of chaotic systems are slightly stronger than, but similar to, those of i.i.d. noises. To save space, the results about AR noise are not displayed here.

5.4.1 Some Experiments with Classical Chaotic Time Series

5.4.1.1 *Impact of dynamical noises on Henon map*

First, we consider the Henon map (Henon, 1976) of the form

$$\begin{cases} x_{n+1} = 1 - ax_n^2 + by_n \\ y_{n+1} = x_n \end{cases}$$

With a = 1.4 and b= 0.3, for initials such as $x = 0$, $y = 0.9$, Henon map has one strange chaotic attractor (Figure 5.14a).

Experiments show that Henon map is sensitive to the disturbance of dynamical noises. Even 1% dynamical noise can lead the Henon map to infinity. Noises above 2% could easily push the orbit outside the basin of attraction, and the series goes to minus infinity exponentially. Figure 5.14b shows the portrait of a comparatively short Henon map series (1000 points) with 2% noise. It resembles the pure Henon map series in the appearance. However, with the evolution of the system, namely, the increase of the iterations of Henon map (e.g., >2000), it will surely go to infinity. The reason that such a low level as 2% of noise may lead the Henon map to infinity is that the boundary of the attraction basin of the Henon map nearly touches the attractor in several places, at these places, very small disturbances will push the trajectory outside the basin. Therefore, in the presence of dynamical noise, only for short series, the series can stay in the attractor. For example, with 2% noise, the Henon map rarely remains in the attractor after 1000 iterations. But as the noisy Henon map preserves a chotic attractor, we can identify a clear scaling region on the Takens-Theiler estimate D_{TT} versus ln$r$ plot (Figure 5.15), and give a finite correlation dimension estimate about 1.25 (for noise-free series, about 1.22). With 10% dynamical noise, the Henon map series usually start to grow exponentially to infinity within 20 iterations (as shown in Figure 5.14c). With 100% dynamical noise, the exponential growth starts within 10 steps. With such short series, low-dimension chaos cannot be identified.

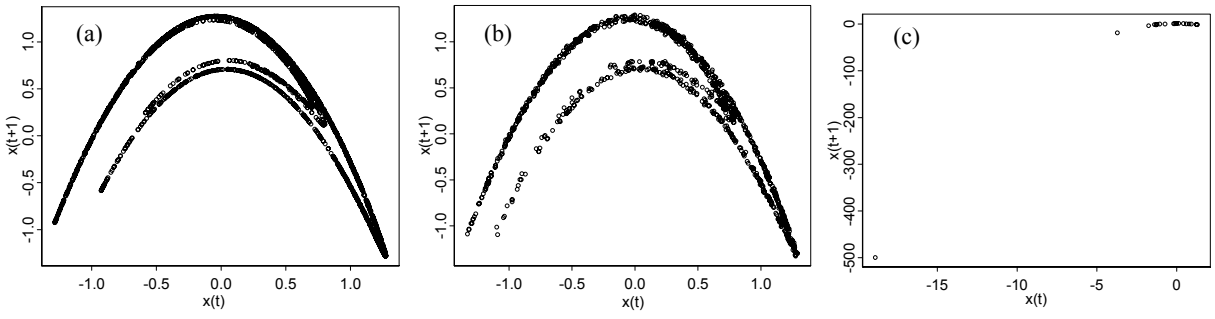


Figure 5.14 Henon map with (a) 0%; (b) 2% and (c) 10% Gaussian dynamical noise

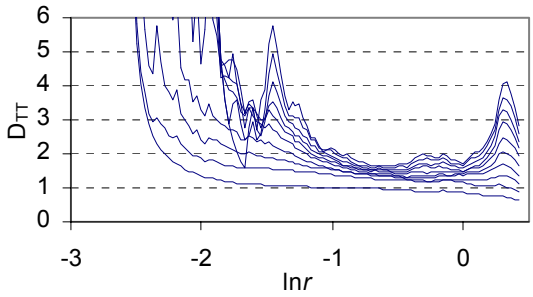


Figure 5.15 Correlation dimension estimate of Henon map series with 2% dynamical noise

5.4.1.2 *Impact of dynamical noises on Ikeda map*

Secondly, we consider Ikeda map (Ikeda, 1979) of the form

$$z_{n+1} = \gamma + \mu \exp\left(i\beta - \frac{i\alpha}{1+|z_n|^2} \right) z_n$$

where the z_n are complex variables. This map can be written as a two-dimensional system in the following form

$$\begin{cases} x_{n+1} = \gamma + \mu(x_n \cos\phi - y_n \sin\phi) \\ y_{n+1} = \mu(x_n \sin\phi + y_n \cos\phi) \end{cases}$$

where $\phi = \beta - \alpha / (1 + x_n^2 + y_n^2)$. With $\alpha = 6$, $\beta = 0.4$, $\gamma = 1$ and $\mu = 0.9$, for an initial such as x = 0 and y = 0, the map gives chaos (Figure 5.16a). It is known that Ikeda map has a chaotic attractor with a complex attraction basin and one stable point attractor centered at (2.9721316, 4.145946).

With 2% dynamical noise, the system will usually stay in the chaotic attractor first (Figure 5.16b), then move to point attractor as the system evolves. With the increase of noise level, the chance of staying in the chaotic attractors decrease, meanwhile the chance of moving to the point attractor increase. When noise level reaches 10%, the system only stays in chaotic attractor for short time (usually less than 1000 iterations), then move to the point attractor (Figure 5.16c). When the noise level is as high as 100%, the system usually escapes the chaotic attractor within 10 steps. Even for small noise level (e.g., 2% noise), with the evolution of the system (e.g., iterate the map more than 20000 times), the system will finally be trapped into the point attractor.

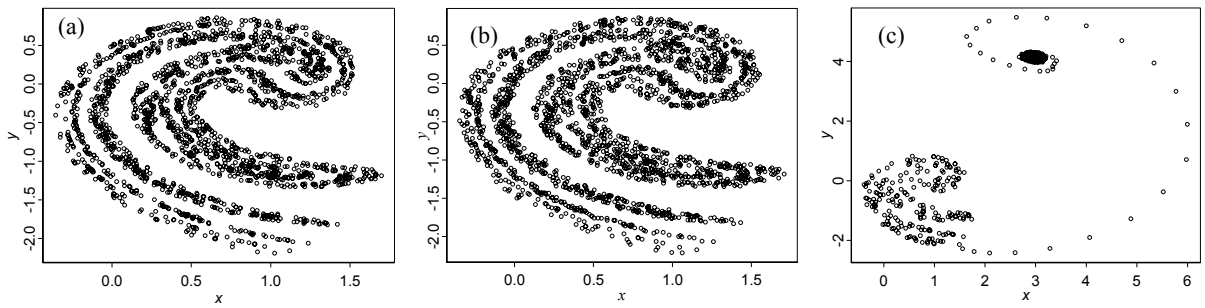


Figure 5.16 Ikeda map with (a) 0%; (b) 2% and (c) 10% Gaussian dynamical noise

We also calculate the Takens-Theiler estimate D_{TT} of correlation dimension for Ikeda map series with 2% and 10% dynamical noise when the series stays in the chaotic attraction basin, and plot D_{TT} versus ln$r$ in Figure 5.17. We can clearly identify the scaling region in Figure 5.17a, and estimate the correlation dimension about 1.85 (for pure series, about 1.69). But the scaling region in Figure 5.17b is not clearly discernable. From the vague region around ln$r$ = -0.5, the correlation dimension estimate for Ikeda series with 10% noise is about 2.12.

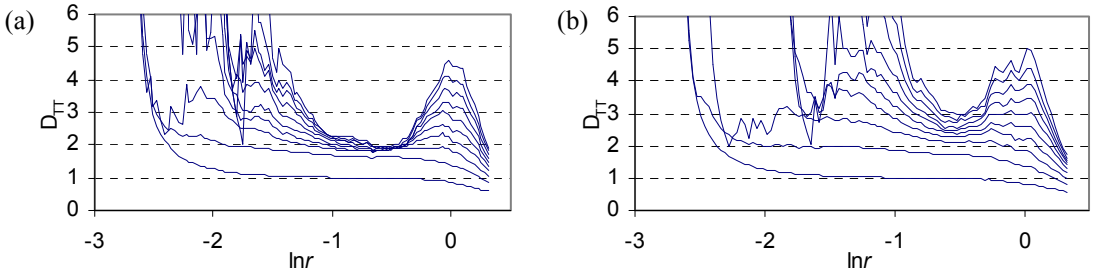


Figure 5.17 Correlation dimension estimate of Ikeda map with (a) 2% and (b) 10% dynamical noises

5.4.1.3 *Impact of dynamical noises on discretized Mackey-Glass flow*

Finally, we consider Mackey-Glass delay differential equation (Mackey and Glass, 1977) of the form

$$\frac{dx}{dt} = \frac{ax_{t-\tau}}{1+x_{t-\tau}^{c}} - bx_t$$

where $x_{t-\tau}$ is the value of x at time t-τ. It can be written as an approximate m+1 dimensional map in delay coordinates:

$$x_{n+1} = \frac{2m-b\tau}{2m+b\tau}x_n + \frac{a\tau}{2m+b\tau}\left(\frac{x_{n-m}}{1+x_{n-m}^{c}} + \frac{x_{n-m+1}}{1+x_{n-m+1}^{c}}\right)$$

With a = 0.2, b = 0.1 and c = 10, this map can generate time series with chaotic attractors of different dimension for τ > 16.8. We choose m = 30, τ = 30. The noise-free series and the series with 2% and 10% dynamical noise are plotted in Figure 5.18a, b and c.

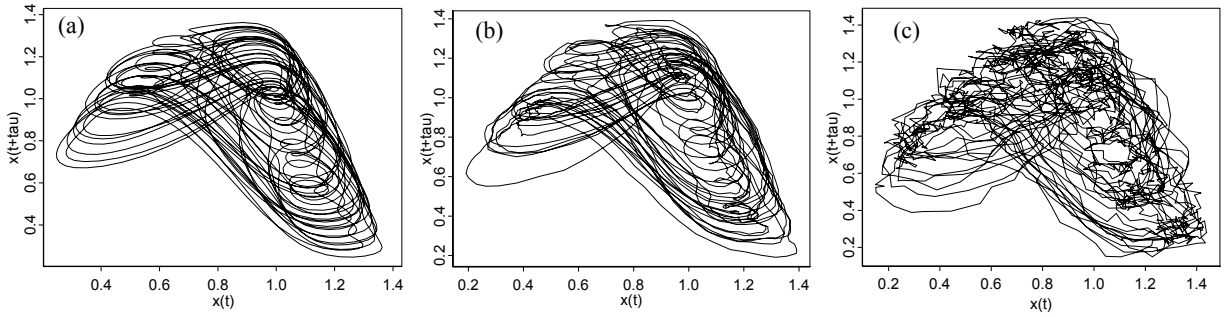


Figure 5.18 Mackey-Glass flow with (a) 0%; (b) 2% and (c) 10% Gaussian dynamical noise

From Figure 5.18b, we see that the chaotic attractor is very clear with 2% Gaussian noise. Because the attraction basin of Mackey-Glass flow is unbounded, even for high level (e.g.,

10%) of noises, the attractor still can be vaguely discerned (Figure 5.18c). But the attractor is not discernable anymore with the noise of 100% level.

We calculate the D_{TT} for discretized Mackey-Glass flow contaminated with 2% and 10% dynamical noise, plotted in Figure 5.19. With 2% noise, we can get an estimate of correlation dimension about 2.5 (correct correlation dimension is about 2.45). Although we can discern the attractor vaguely with noises as high as 10% as shown in Figure 5.18c, it is hard to define the scaling region in Figure 5.19b and hard to estimate the correlation dimension correctly. With the noise level as high as 100%, the scaling region is totally lost.

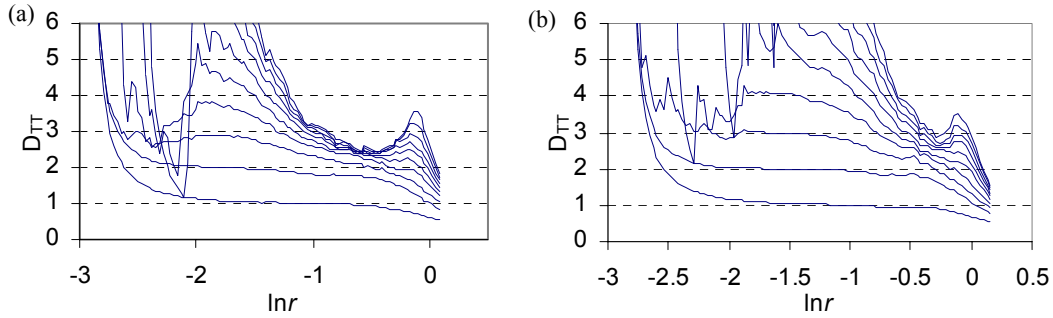


Figure 5.19 Estimate of correlation dimension for Mackey-Glass flow series with (a) 2% and (b) 10% dynamical noise

5.4.2 Analysis of the Results for the Cases of Classic Chaotic Processes

According to the above analyses, we have some remarks on the following two aspects:

(1) About the identification of chaotic system: from possible to impossible

Although the presence of noise limits the performance of many techniques of identification and prediction of chaotic systems [e.g., Schreiber and Kantz, 1996], with low level (e.g., 2%) Gaussian noise, the chaotic attractor can still be well preserved and basically correct estimate of correlation dimension can be made. However, in the presence of dynamical noises, the estimate is biased to a higher value, and the higher the noise level, the larger the bias. When the level of dynamic noise is very high (e.g., 10%), it is hard to identify the systems analyzed above correctly, let alone in the presence of noises of 100% level.

(2) About the property of chaotic system: from deterministic to stochastic

Although chaotic systems are widely considered as deterministic, in the presence of dynamical noise, the system may still present chaotic behavior. Because a chaotic system with dynamic noise has a stochastic component and the system turns out to be stochastic instead of being deterministic, that means, a stochastic may present chaotic behavior. In the presence of dynamical noise, whether or not the chaotic system remains in the chaotic attractor depends on the intensity of stochastic disturbances. If the disturbance is so strong as to push the orbit outside the chaotic attraction basin, then the system may go to infinity, or fall into neighboring non-chaotic attractors, or just lost the geometry of the chaotic attractor, and the system becomes non-chaotic.

Two factors affect whether the trajectories would escape the chaotic attractor of a chaotic system with stochastic noise. (1) The distance *DB* between the boundary of attraction basin and that of the attractor. The larger the distance is, the harder to push the trajectory away from the attractor. If the attraction basin is unbounded, which means infinity *DB*, then the system will never escape. Systems with chaotic attractors nearly touch the boundary of their attraction basins in some places (namely, have very small DB in these places, such as the Henon map), may easily been pushed outside the chaotic attractors. (2) The distribution of the stochastic noise. For uniform distributed noise, if the maximum value of disturbance is less than minimum *DB*, then the stochastic chaotic system will never escape the chaotic attractor. For un-uniformly distributed noise, when the attraction basin is bounded, the dynamical systems would probably ultimately escape the attractor basin and go to infinity or fall into other neighboring attractors as the evolution of the system goes to infinity.

5.4.3 The Case of Streamflow Processes

With regard to an observed hydrological series, its dynamics is inevitably contaminated by not only measurement noise, but also dynamical noise. For instance, a streamflow process, which is the major output of a watershed system, may be influenced by many factors, including external inputs (e.g., precipitation, temperature, solar radiation), other outputs (e.g., evaporation, transpiration), and various human interventions. These factors are generally composed of both deterministic components and stochastic components. Among all the factors, the precipitation is the dominant one which may disturb the streamflow process most significantly. In a flood event, it is very common that the flow generated by storm rainfall makes up over 50% of the total discharge. Assuming that base flow of a streamflow process is the noise-free time series, that 50% streamflow is generated by rainfall means that the dynamical noise is 100% of the original series. Even if the streamflow process is chaotic, according to the experiments we made with the known chaotic systems, it is impossible to detect the chaotic characters with such intense disturbances.

One may argue that dynamical noise may be a higher dimensional part of the system dynamics. If viewing it in this way, then we may say, the streamflow process may be chaotic, but definitely not a low-dimensional chaotic process, because as a major output of watershed system, it is influenced by not only many dynamical inputs and other outputs of the watershed system, but also dynamical variations of many internal factors, such as the spatial-temporal variability of soil infiltration capacity. That means, if the streamflow process is chaotic, it would be a very high dimensional chaotic process, rather than a low-dimensional process as claimed by some researchers. Although it is possible that collective behavior of a huge number of external and internal degrees of freedom may lead to low-dimensional dynamics, it seems not the case for streamflow processes because we cannot observe finite correlation dimension in the streamflow processes we studied.

The objective of detecting chaos should be giving a better understanding of the hydrological time series, rather than just trying to give evidences of the presence of chaos. After the tide of detecting the existence of chaos in hydrological time series, some researchers have tried to make some physical explanation to the chaos they claimed. For example, Porporato and Ridolfi (2003) state that the climate dynamics that produces the input of the rainfall–runoff

transformation is the first source of possible determinism, and the strong low-pass filtering action of the basin, while smoothing out some of the space–time complexity of rainfall, could make more evident the low-dimensional deterministic components originating from both climate and rainfall–runoff transformation. This is a doubtful statement. On the one hand, whether climate dynamics is deterministic is questionable, and consequently, whether the inputs of watershed system are deterministic is questionable. On the other hand, watershed system should be a damping system, which damps down the disturbance of the space–time complexity of rainfall as well as other dynamical inputs, rather than smoothes out the high-frequency space–time complexity with so-called strong low-pass filtering action. While the disturbance is being damped down, it will surely cause space–time variations of internal variables. In fact, the focus of the enormous efforts made by the hydrology community of developing physically-based distributed models (e.g., MIKE-SHE model) is to describe the rainfall-runoff via considering the spatial-temporal heterogeneity rather than searching for a mechanism which can smooth out the space–time complexity of rainfall or other variables.

All in all, on the one hand, due to high level of dynamic disturbances, it is not possible to accurately identify the chaos in streamflow processes even if it exists; on the other hand, the existence of chaotic characteristics does not necessarily mean determinism, consequently, even if we chaos exhibits in a streamflow process, we cannot conclude that the streamflow process is deterministic. As pointed out by Schertzer et al. (2002), it is a questionable attempt to reduce complex systems to their low-dimensional caricatures, and there is no obvious reason that processes should be run by deterministic equations rather than by stochastic equations, since the former are merely particular cases of the latter.

5.5 Discussions on the Sources of Nonlinearity

Streamflow processes are fundamentally driven by meteorological processes. It is currently well accepted that climatic and atmospheric dynamics are strongly nonlinear (e.g. Saltzman, 1983; Lorenz, 1991), and the daily streamflow process is usually perceived as nonlinearly dependent on the magnitude of the rainfall (e.g., Minshall, 1960; Wang et al., 1981), therefore, the first source of nonlinearity in daily streamflow processes probably stems from the nonlinearity in daily atmospheric processes. In fact, the seasonality which plays a role in nonlinearity of streamflow processes, as show in Section 5.2.2, also essentially arise from the seasonality in meteorological processes. What makes the nonlinearity of streamflow processes more complicated is the nonlinear rainfall-runoff response. In the rainfall–runoff transformation, many more sources of nonlinearity are involved and they may combine differently in different basins. Examples include: the rainfall interception due to vegetation (a typical season-dependent, threshold-like mechanism), the unsaturated groundwater flow (where the hydraulic conductivity depends in a strongly nonlinear way on the degree of soil saturation), and open channel hydraulics.

As the timescale increases, the nonlinearity in the meteorological series weakens. For example, Chen and Rao (2003) found that all of the stationary segments of standardized monthly temperature and precipitation series they studied are either Gaussian or linear. One

the other hand, nonlinear fluctuations present in the daily streamflow processes, such as the ARCH (autoregressive conditional heteroskedasticity) effect in daily streamflow processes, are generalized. Therefore, the degree nonlinearity in streamflow processes decreases with the increase of timescale.

However, because of differences in the geographical and climatological environment, the character and intensity of nonlinearity of different streamflow systems are consequently different. For example, temperature may be a dominant variable for the whole dynamics of the streamflow process of the Yellow River at TNH. Its effect on both rainfall and snow cover has a very strong influence in determining the inertia of the whole hydrological system. That probably makes the streamflow system of the Yellow River at TNH appears to be less reactive than other streamflow systems. Even though the system exhibits a nonlinear character at short timescale, it may decay faster than other more active streamflow systems as the timescale increase. That may be the reason why only the 1/3-monthly flow series at TNH can exhibit linearity after being standardized by seasonal variance.

Another aspect should be noticed is the nonlinearity of streamflow response with respect to the catchment characteristics (e.g., area, topography, channel network geometry and groundwater system). A number of studies indicate that the nonlinearity decreases and catchments become more linear with increasing catchment area (Minshall, 1960; Wang et al., 1981). However, nonlinearity does not disappear as the catchment scale increase because channel network hydrodynamics would be an important source of nonlinearity at large scales (Robinson, 1995). Among the streamflow processes of 4 rivers in our study, streamflow processes of the Yellow River and the Rhine River that have much larger catchment areas seem possess no less nonlinearity in terms of BDS statistics than streamflow processes of the other two rivers that have smaller catchment areas, namely, no clear relationship is found between the catchment area and the intensity of nonlinearity.

Comparing the results of stationarity test, long-memory test, conditional heteroskedasticity test and nonlinearity test, we find that these results have similar timescale pattern, that is, the intensity of nonstationarity, long-memory, conditional heteroskedasticity and nonlinearity decays with the increase of timescale. As conditional heteroskedasticity is a stochastic nonlinearity mechanism, whereas the mechanism of long-memory is not clear, therefore, as viewed from the streamflow time series itself, the conditional heteroskedasticity is also an important source of the nonlinearity of streamflow processes, while the intensity of long-memory may impact the test of nonlinearity. As for the nonstationairty, it is not distinguishable from nonlinearity by the test methods we applied here. The intensity of nonlinearity may be closely related to the intensity of nonstationarity. Although the streamflow series in this study are mostly stationariy at certain significance levels, low level stationarity may give rise to positive results (namely, the existence of nonlinearity) in nonlinearity test. However, the relationship of the stationary level and the nonlinear intensity is a question worth further investigation in the future.

5.6 Conclusions

Streamflow processes are commonly considered to be nonlinear. However, it is not clear what kind of nonlinearity is acting underlying the streamflow processes and how strong the nonlinearity is within the streamflow processes at different timescales.

Streamflow processes of five rivers in different geographical and climatic regions are tested for nonlinearity with BDS test (Brock et al., 1996). The tests focus on four characteristic time scales (i.e., one year, one month, 1/3 month and one day). It is found that the shorter the timescale, the stronger the nonlinearity. All annual series are linear, whereas all daily series are nonlinear. Deseasonalized series generally have less nonlinearity in terms of BDS test statistics, implying that seasonality in the original flow series plays a role in nonlinearity. To exclude the impact of seasonal variation in variance of the pre-whitened series, the pre-whitened series are further deseasonalized by dividing their seasonal standard deviation. It is found that the intensity of all series generally weakens after such further deseasonalization, which indicates that the seasonal variation in variance of the pre-whitened series also play a role in making the streamlfow processes nonlinear. Although some 1/3-monthly series (after removing seasonal effects) and monthly streamflow series are linear according to BDS test results, we cannot conclude that there is no weak nonlinearity present in those 1/3-monthly and monthly streamflow series, because the power analysis of BDS test shows that BDS test is not powerful enough for detecting weak nonlinearity.

There is no evidence found of the existence of low-dimensional chaos in the streamflow series of all the fiver rivers with correlation exponent method. When testing for chaos in streamflow processes, some authors tend to accept the existence of chaos in streamflow processes even if test results do not give really clear evidences. For instance, many published results claim clear evidences of the existence of low-dimensional chaos in streamflow series without providing scaling plots or without providing convincing scaling plots that have clearly discernible scaling regions, whereas clearly discernible scaling region is imperative for identifying the finite correlation dimension. Furthermore, cares must be taken when computing correlation dimension for serially dependent hydrological series, because temporal coherence could be mistaken for geometrical structure if temporally correlated points are not excluded for calculating correlation integrals.

The dynamics of observed hydrological series is inevitably contaminated by not only measurement noise, but also dynamical noise. Experiments with three well-known chaotic systems (i.e., Henon map, Ikeda map, discretized Mackey-Glass flow) show that, with low-level (e.g., 2% or less) Gaussian noise, especially low-level uniformly distributed noise, the chaotic attractors may still be well preserved and we can give basically correct estimate of correlation dimension. That indicates that even if we found clear evidences of the existence of chaos in a time series, it does not necessarily mean determinism. A chaotic system with stochastic components, which turns out to be a stochastic system, could present chaotic behavior. It behaves similar to a noise-free system when the stochastic disturbances are not strong. Therefore, even if we find undoubtable evidences that streamflow process is chaotic, it does not necessarily mean determinism. On the other hand, when the noise level is high (e.g.,

10%), it is hard to correctly identify the chaotic system, let alone 100% level noise. Consequently, because the streamflow process usually suffers from strong natural and anthropogenic disturbances (could be of a level of up to 100% or higher) that are by themselves composed of both stochastic and deterministic components, it is not likely to correctly identify the chaotic dynamics even if the streamflow process is indeed low-dimension chaotic process under ideal circumstances (i.e., without any or only with small enough stochastic disturbances). On the other hand, because the existence of chaotic characteristics does not necessarily mean determinism, consequently, even if we chaos exhibits in a streamflow process, we cannot conclude that the streamflow process is deterministic.

The intensities of nonstationarity, long-memory, conditional heteroskedasticity and nonlinearity have similar timescale pattern, that is, they decays with the increase of timescale. As viewed from the streamflow time series itself, the conditional heteroskedasticity is an important source the nonlinearity of streamflow processes, whereas long-memory may impact the test of nonlinearity. The intensity of nonlinearity may be closely related to the intensity of nonstationarity. Although the streamflow series in this study are mostly stationariy at certain significance levels, low level stationarity may give rise to positive results (namely, the existence of nonlinearity) in nonlinearity test.

Chapter 6 Forecasting Daily Streamflow Using ANN Models

6.1 Introduction

Since the daily streamflow processes exhibit a high degree of nonlinearity, as shown in the previous chapters, in this chapter we discuss how to make daily streamflow forecasts with the artificial neural network (ANN) model, which is generally a nonlinear model that may be mathematically treated as a universal approximator. ANNs mimic the functioning of a human brain by acquiring knowledge through a learning process that involves finding an optimal set of weights for the connections and threshold values for the nodes. ANNs are able to extract the underlying relation between the inputs and outputs of a process, without the physics being explicitly provided to them. The ability to learn and generalize ‘‘knowledge’’ from sufficient data pairs makes it possible for ANNs to solve large-scale complex problems such as pattern recognition, nonlinear modeling, classification, association, control, and others—all of which find applications in hydrology today.

ANNs have gained more and more popularity for hydrological forecasting in the last decade (e.g., Maier and Dandy, 2000; Dawson and Wilby, 2001). In one of the early applications involving streamflows, Kang et al. (1993) used ANNs and autoregressive moving average models to predict daily and hourly streamflows in the Pyung Chang River basin in Korea. This preliminary study concluded that ANNs are useful tools for forecasting streamflows. Many studies have confirmed the superiority of ANN models over or equality to the traditional statistical and/or conceptual techniques in modeling the hydrological process (e.g., Hsu, et al., 995; Raman and Sunilkumar 1995; Dibike and Solomatine, 2001; Tokar and Markus 2000; Birikundavyi et al., 2002).

We will use ANN model to make 1 to 10 day ahead daily discharge forecasts for the streamflow process of the upper Yellow River at TNH. The organization of this chapter is as follows. In Section 6.2 normal ANNs are applied to forecasting 1- to 10-day ahead daily streamflow. In Section 6.3, different ANN hybridization approaches are presented, and forecasts are made with these hybrid ANNs. Some discussions and conclusions of the study are given in Sections 6.4 and 6.5, respectively.

6.2 Fitting Normal ANN Models to Daily Streamflow Series

When building an ANN model, a number of decisions must be made, including the neural network type, network structure, methods of pre- and post-processing of input/output data,

training algorithm and training stop criteria. The feed-forward multi-layer perceptron (MLP) ANN is the most widely used type of ANN in hydrological modeling, and is also adopted in this study. We discuss issues of fitting the normal MLP-ANN network model to daily flows at TNH in this section.

6.2.1 Determine the Architecture of ANN

Network architecture mainly denotes the number of input/output variables, the number of hidden layers and the number of neurons in each hidden layer. It determines the number of connection weights and the way information flows through the network. Among these structure parameters, usually only one parameter is clear, namely, the number of ANN output variables. The other structure parameters should be determined in some way.

6.2.1.1 *Determine the inputs for ANN*

In a large number of applications of ANN, we face an implementation dilemma: a great number of input features is often available to solve the problem, but the limited size of the training set makes it seemingly impossible to use them all without running the risk of severely overfitting the data. There are two approaches to the resolution of this problem: (1) applying some variable selection procedures, such as stepwise selection; (2) using generalization techniques, such as cross-validated early stopping and Bayesian regularization. The second approach is related to the training process, which will be addressed in Section 6.2.3. We take the first approach with two techniques, namely, the method of partial autocorrelation function and the method of phase-space reconstruction, to determine the inputs for ANN.

(1) With the method of partial autocorrelation function

In determining the number of inputs of networks, the autocorrelation analysis, cross-correlation analysis and some physical considerations can be helpful (Lachtermacher and Fuller, 1995). In Section 3.1.2, we have analysed PACF of the deseasonalized daily streamflow series. According to the structure of PACF, it seems that daily discharges about 40-days apart still have weak correlation. Furthermore, it has been shown in Section 5.2.2 that, according to AIC, if we fit AR models to the daily flow series, the AR order should be 41 for log-transformed daily flow series and 38 for log-transformed and deseasonalized daily flow series. Consequently, we may use 41 or 38 as the number of inputs in the ANN model.

(2) With the method of reconstructing state-space

To describe the time evolution of a dynamical system in some multi-dimensional state space with a scalar time series, one needs to employ some technique to unfold the multi-dimensional structure using the available data. Packard et al. (1980) and Takens (1981) proposed the time delay coordinate method to reconstruct the state space from a scalar time series. According to the method, the state vector X_i in a new space, the embedding space, is formed from time delayed values of the scalar measurements $\{x_i\}$ as $\boldsymbol{X}_i = [x_i, x_{i-\tau}, \ldots, x_{i-(m-1)\tau}]$, where x_i is the observed value of the time series at time $t_i = i(\Delta t)$, Δt is the sampling interval which is normally taken as 1, m is the embedding dimension, and τ is the delay time.

As shown in Section 5.3.1, due to the strong annual seasonality, the τ value would be about ¼ of the annual period of the daily streamflow process, namely about 91. However, for the purpose of forecasting, although the reconstructed phase-space is redundant with a value of τ = 1, we tolerate having some information redundancy in preference to losing any useful information.

A method to determine the minimal sufficient embedding dimension *m* was proposed by Kennel et al. (1992), called the '*false nearest neighbor*' method. The minimal embedding dimension *m* for a given time series means that *m* is the minimal value that is sufficient to insure that in a *m*-dimensional embedded space the reconstructed attractor is a one-to-one image of the attractor in the original state space. If the series is embedded in an *m'*-dimensional space with $m' < m$, then some points that are actually far from each other will appear as neighbors because the geometric structure of the attractor has been projected down to a smaller space. These points are called false neighbors. For false neighbors, their trajectories will move far away as the embedding dimension increases. Suppose the point $\boldsymbol{X}_i = [x_{i-p+1}, \ldots, x_i]$ has a neighbor $X_j = [x_{j-p+1}, \ldots, x_j]$ in a *p*-dimensional space. Calculate the distance $\| X_i - X_j \|$ and compute

$$R_i = \frac{| x_{i+1} - x_{j+1} |}{\| X_i - X_j \|}. \tag{6.1}$$

If R_i exceeds a given threshold R_T (say, 10 or 15), the point X_i is marked as having a false nearest neighbor. We say the embedding dimension *p* is high enough if the fraction of points that have false nearest neighbors is actually zero, or sufficiently small, say, smaller than a criterion R_f.

Setting the false neighbor threshold $R_T = 10$, we calculate the fraction of false nearest neighbors as a function of the embedding dimension for daily streamflow series at TNH, show in Figure 6.1. If we set the fraction criterion $R_f = 0.01$, then the embedding dimension is 5, which means that the state of streamflow process is determined by 5 lagged observed values. Correspondingly, when fitting an ANN model to the series, we use 5 lagged values as input to forecast the one-step-ahead value.

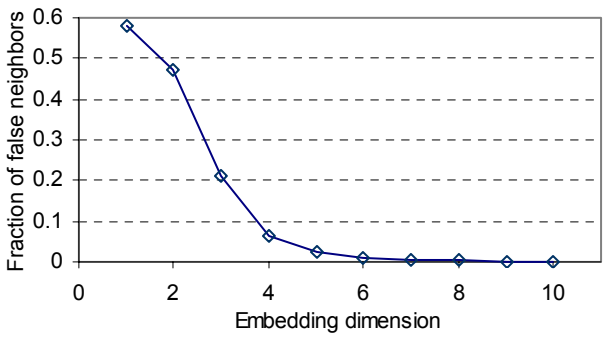


Figure 6.1 The fraction of false nearest neighbors as a function of the embedding dimension for daily streamflow series at TNH

Therefore, according to above two techniques, we will try ANN models with 5 inputs and 41 inputs for for forecasting daily streamflows of the Yellow River at TNH.

6.2.1.2 *Determine the number of hidden layers and the number of neurons in each layer*

The major concern of the designer of an ANN structure is to determine the appropriate number of hidden layers and the number of neurons in each layer. There is no systematic way to establish a suitable architecture, and the selection of the appropriate number of neurons is basically problem specific.

Hornik et al. (1989) proved that a single hidden layer network containing a sufficiently large number of neurons can be used to approximate any measurable functional relationship between the input data and the output variable to any desired accuracy. De Villars and Barnard (1993) showed that an ANN comprised of two hidden layers tends to be less robust and converges with less accuracy than its single hidden layer counterpart. Furthermore, some studies indicate that the benefits of using a second hidden layer are marginal to the rainfall-runoff modeling problem (e.g., Minns and Hall, 1996; Abrahart and See, 2000). Taking cognizance of the above studies, a single hidden layer is used in this study.

There are some algorithms, including pruning and constructive algorithms, to determine an 'optimum' number of neurons in the hidden layer(s) during training. However, a trial and error procedure using different number of neurons is still the preferred choice of most users (e.g., Shamseldin, 1997; Zealand et al., 1999; Abrahart and See, 2000) and is the method used also in this research. With the trial and error procedure, Ebberhart and Dobbins (1990) suggested starting with hidden nodes equal to half of the input nodes; Tingsanchali and Gautam (2000) found that starting with hidden nodes equal to or slightly greater than the input nodes is adequate.

6.2.2 Data Pre-Processing

Before fitting an ANN model, the data should be pre-processed. There are basically two reasons for pre-processing. Firstly, pre-processing can ensure that all variables receive equal attention during the training process. Otherwise, input variables measured on different scales will dominate training to a greater or lesser extent because initial weights within a network are randomized to the same finite range (Dawson and Wilby, 2001). Secondly, pre-processing is important for the efficiency of training algorithms. For example, the gradient descent algorithm (error backpropagation) used to train the MLP is particularly sensitive to the scale of data used. Due to the nature of this algorithm, large values slow training because the gradient of the sigmoid function at extreme values approximates zero (Dawson and Wilby, 2001). In general, there are fundamentally two types of pre-processing methods. The first is to rescale the data to a small interval (referred to as *rescaling*), such as [–1, 1] or [0, 1], depending on the transfer (activation) function used in the neurons, because some transfer functions are bounded (e.g. logistic and hyperbolic tangent function). Another is to standardize the data by subtracting the mean of the series and dividing by the standard deviation to make the data have a mean of 0 and variance 1 (referred to as *standardization*).

In this study, to compare the influence of different pre-processing procedures on model performance, 6 different pre-processing procedures are applied:

- Standardizing the raw data series;
- Rescaling the raw data series;
- Standardizing the log-transformed data series;
- Rescaling the log-transformed data series;
- Deseasonalizing the log-transformed data series;
- Deseasonalizing the log-transformed data series and then rescaling the deseasonalized series.

The deseasonalization is a special type of standardization, which is commonly used when fitting time series model to streamflow series (e.g., Hipel and Mcleod, 1994). But instead of using the overall mean value and the overall deviation to make the standardization, deseasonalization is accomplished by subtracting the seasonal (e.g., daily, monthly) means and dividing by seasonal standard deviations.

Because the transfer (or activation) function in the hidden neurons used in this study is the tan-sigmoid function in the form of $f(x) = 2/(1+e^{-2x}) - 1$, and the output value of the tan-sigmoid function is bounded between –1 and 1, so when rescaling the data, we rescale both the input and output data to [-1, 1] with

$$x = 2(x - x_{\min})/(x_{\max} - x_{\min}) - 1$$

where x_{min} and x_{max} are the minimum and maximum values in the data set, respectively.

6.2.3 ANN Training

The ANN training is fundamentally a problem of nonlinear optimization, which minimizes the error between the network output and the target output by repeatedly changing the values of ANN's connection weights according to a predetermined algorithm. Error back-propagation (BP) (e.g., Rumelhart et al., 1986) is by far the most popular algorithm for optimizing feedforward ANNs. It is essentially a gradient descent technique that minimizes the network error function. Each input pattern of the training data set is passed through the network from the input layer to the output layer. For an ANN with one output, the network output $\{y_t\}$ is compared with the desired target output $\{x_t\}$, and an error is computed based on error function

$$E = \sum_{i=1}^{N} (x_i - y_i)^2 \tag{6.2}$$

This error is propagated backward through the network to each node, and correspondingly the connection weights are adjusted based on equation

$$\Delta w_{ij}(n) = -\eta * \frac{\partial E}{\partial w_{ij}} + m * \Delta w_{ij}(n-1) \tag{6.3}$$

where Δw_{ij} (n) and Δw_{ij} (n-1) are weight increments between node i and j during the nth and (n-1)th pass, or epoch; η and m are called learning rate and momentum, respectively. The momentum factor can speed up training in very flat regions of the error surface and help prevent oscillations in the weights. A learning rate is used to decrease the chance of the training process being trapped in local minima instead of global minima. The BP algorithm involves two steps. The first step is a forward pass, in which the effect of the input is passed forward through the network to reach the output layer. After the error is computed, a second step starts backward through the network. The errors at the output layer are propagated back toward the input layer with the weights being modified according to (6.3).

An inherent problem of training ANNs with optimizers, such the Levenberg–Marquardt optimizer (Marquardt, 1963), is that Networks trained with the backpropagation algorithm are sensitive to initial conditions and may get stuck in local minima of error surface. One way to alleviate this problem and to increase the likelihood of obtaining near-optimum local minima is to train some ANNs with a random set of initial weights, and choose the one with the lowest error. For example, Park et al. (1996) select the best neural network for predicting yearly sunspots among ten networks of the same architecture, each of which was initiated with a random set of weights. This is the way of choosing the "best performing" networks by many authors (e.g., Rajurkar et al., 2004). However, in operational application, we actually don't know which network performs best for future streamflow process. There may be many parameter sets within a model structure that are equally acceptable as simulators of a dynamical process of interest. Consequently, instead of attempting to find a best single ANN model, we may make predictions based on an ensemble of neural networks trained for the same task (see e.g., Sharkey, 1996). In this present study, the idea of ensemble prediction is adopted, and the simple average ensemble method is used. For each ANN model, we train it ten times so as to get 10 networks, and choose 5 best ones according to their training performances. With the 5 selected networks, we get 5 outputs. Then we take a simple average of the 5 outputs to be the final output.

An important issue in training an ANN model is when to stop the training process so as to avoid the problem of overfitting. The simplest way of avoiding overfitting is using enough training data. Amari et al. (1997) showed that, when the ratio of the sample size to the number of weights is larger than 30, no overfitting is observed. In fact, in the experiments in the next chapter with simulated time series we will show that, to avoid overfitting, the ratio may be as high as 50. In our study in the case of univariate streamflow time series forecasting, the ratio of the number of input data sets and the number of network weights is far larger than 50, therefore, the use of cross-validation data is not necessary. The training is stopped after 3000 epochs.

The training of ANNs In this study was implemented using the *traingdm* function in Matlab (release 13) Neural Network Toolbox which uses error backpropagation algorithm by updating the weight and bias values according to gradient descent with momentum.

6.2.4 Forecasting with Normal ANN Models

The normal MLP-ANN models are fitted using the daily discharge data at TNH from 1956 to 1995, and 1 to 10-day ahead forecasts are made for year 1996 to 2000. To predict daily discharges up to 10 days of lead time, a simple recursive algorithm will be used to obtain forecasts for successive lead times. An ANN model will first predict Q_{t+1}, and the predicted Q_{t+1} will then be used to update the input variables into the ANN so as to predict Q_{t+2}. This procedure is thus repeated until the forecasted values ranging from Q_{t+1} to Q_{t+10} are made.

Matlab Neural Network Toolbox is used to construct different one-hidden-layer MLP networks with a range of hidden neurons in the hidden layer. The model forecating results with different architectures indicate that for ANNs with 5 inputs, 3-node networks generally perform best. Therefore, the chosen configuration for MLP ANNs is 5-3-1, namely, 5 inputs, one hidden layer with 3 hidden neurons and one output. This MLP structure is adopted throughout all the ANN models in this study. For ANNs with 41 inputs, we test ANNs with 10, 15, 20, 25, 30, 35 and 41 hidden nodes, it is found that there is no significant difference among these ANNs, consequently, ANNs with 10 hidden nodes are applied for alleviate computational burden.

We first test the performance of networks with a 5-3-1 structure. Because 6 different pre-processing procedures are applied in this study, as mentioned in Section 6.2.2, hence 6 MLP models in total are fitted to the daily streamflow data, each of which has a different pre-processing procedure. When making forecasts, post-processing is needed to inversely transform the outputs to their original scale. The forecasts for log-transformed-and-deseasonalized (referred to as Log-DS hereafter) series are the average of an ensemble of 5 networks, which give best training performance. For other series, we simply choose the forecasts of the networks that give the best test performance. Table 6.1 lists model performance evaluation results of the 1- to 10-day ahead forecasts with the MLP models. According to the forecasting evaluation results, we see that: (i) MLP models fitted to standardized data perform better than those fitted to rescaled data (note that deseasonalization without rescaling is a special type of standardization); (ii) in the case of the longer lead time forecasts, it is better to do a log-transformation before standardizing the data; and (iii) Overall, the MLP model fitted to the deseasonalized data, without rescaling, performs best.

According to evaluations results for networks with the 5-3-1 structure, we fit networks with the 41-10-1 structure only to Log-DS series. The performances of the 5-member ensemble forecast with 41-10-1 networks are listed in Table 6.2. The results show that the performances of more complicated networks with the 41-10-1 structure perform almost as good as the simpler networks with 5-3-1 structure. Therefore, we prefer to use networks with the simpler 5-3-1 structure in the following analysis.

Table 6.1 Forecasting performances with MLP networks with 5-3-1 structure for daily flows at TNH

Lead time (days)		1	2	3	4	5	6	7	8	9	10
CE	raw_std	0.989	0.966	0.938	0.910	0.881	0.849	0.814	0.776	0.739	0.699
	raw_rescale	0.981	0.955	0.918	0.887	0.851	0.815	0.774	0.735	0.694	0.655
	Log-std	0.988	0.965	0.942	0.919	0.895	0.870	0.843	0.815	0.788	0.760
	Log-rescale	0.963	0.942	0.919	0.894	0.864	0.834	0.806	0.778	0.750	0.723
	Log-DS	0.989	0.967	0.943	0.92	0.897	0.873	0.848	0.823	0.799	0.775
	Log-DS_rescale	0.985	0.964	0.939	0.916	0.891	0.866	0.838	0.811	0.785	0.758
SACE	raw_std	0.980	0.936	0.885	0.834	0.779	0.721	0.654	0.585	0.516	0.442
	raw_rescale	0.965	0.916	0.848	0.791	0.724	0.657	0.582	0.508	0.434	0.360
	Log-std	0.978	0.935	0.892	0.849	0.806	0.759	0.708	0.658	0.607	0.555
	Log-rescale	0.932	0.893	0.849	0.803	0.748	0.692	0.641	0.588	0.536	0.487
	Log-DS	0.980	0.938	0.894	0.852	0.810	0.765	0.719	0.672	0.628	0.582
	Log-DS_rescale	0.972	0.933	0.887	0.845	0.799	0.751	0.700	0.650	0.601	0.551

Note: "raw" denotes the raw data; "Log" denotes log-transformation; "std" denotes the standardization by subtracting the mean and dividing by the standard deviation; "DS" denotes the deseasonalization by subtracting seasonal means and dividing by seasonal standard deviations; "rescale" denotes rescaling to –1 ~ 1.

Table 6.2 Forecasting performances with MLP networks with 41-10-1 structure for daily flows at TNH

Lead time (days)	1	2	3	4	5	6	7	8	9	10
CE	0.988	0.965	0.94	0.917	0.895	0.872	0.847	0.822	0.799	0.777
SACE	0.979	0.935	0.89	0.847	0.805	0.762	0.716	0.67	0.627	0.587

6.3 Building Hybrid ANN Models for Daily Flow Forecasting

6.3.1 Introduction to Hybrid ANN Models

It is generally accepted that streamflow generation processes, especially daily streamflow processes, are seasonal and nonlinear, since the processes usually have pronounced seasonal means, variances, and dependence structures, and the under-lying mechanisms of streamflow generation are likely to be quite different during low, medium, and high flow periods, especially when extreme events occur. For instance, low-flow events are mainly sustained by base flow, whereas high-flow events are typically generated by intense storm rainfall.

Many models can be found in the literature for modeling the complex nonlinearity of streamflow processes, among which are those based on the principle of divide-and-conquer (DAC). DAC algorithms deal with a complex problem by dividing it into simple problems whose solutions can be combined to yield a solution to the complex problem (Jordan and Jacobs, 1994). Depending on the feature of nonlinearity, usually a process could be divided, for example through using thresholds, into a number of regimes and fit a linear or nonlinear model for each regime. Correspondingly, the DAC algorithms could be roughly categorized into two types, i.e., local-linear models and local-nonlinear models.

Local-linear DAC models approximate a complex problem locally with linear models. They are fundamentally derivatives of the threshold regression models, such as the constrained

linear system with thresholds (CLS-Ts) of Todini and Wallis (1977) and the multilinear approach of Becher and Kundzewicz (1987), or the threshold autoregressive (TAR) model (Tong & Lim, 1980), which treats a nonlinear process piecewise with linear regression models according to the value of some explanatory variable or the preceding value of the process itself. Local-linear DAC models are still used for modeling rainfall-runoff process (e.g., Solomatine and Dulal, 2003). The periodic autoregressive moving average (PARMA) model, as well as its abbreviated version periodic autoregressive (PAR) model, is widely used to model hydrological time series (e.g., Mcleod and Hipel, 1994). It is the extension of the ARMA model that allows periodic (seasonal) parameters, which can be perceived as a modification of the threshold regression model. Instead of using the value of some explanatory variable or the preceding value of the process itself as the threshold, the PARMA model treats a seasonal process piecewise with linear ARMA models according to which season the process is operating in. When fitting a PARMA model to a seasonal series, a separate ARMA model is fitted for each season of the year. Literature on PARMA (including PAR) models has abounded since the late 1960s' (e.g., Jones and Brelsford, 1967; Pagano, 1978; Salas et al., 1982; Vecchia, 1985; Bartolini et. al., 1988; Salas et al., 1993).

Local-nonlinear DAC models approximate a complex problem locally with nonlinear models, such as nonlinear regression models or ANN models. Local-nonlinear DAC models are increasingly popular for dealing with complex nonlinear processes, mainly owing to the rapid development of ANN techniques. ANNs are known as having the ability of modeling nonlinear mechanisms. They have been increasingly applied to various hydrological problems in the past decade (Maier & Dandy, 2000; Dawson and Wilby, 2001). Nonetheless, some studies have suggested that a single ANN cannot predict the high- and low-runoff events satisfactorily (e.g., Minns and Hall, 1996) since, as aforementioned, the under-lying mechanisms of streamflow generation can be quite different during low, medium, and high flow periods. The mapping ability of a single ANN is limited when faced with complex problems like rainfall-runoff processes. To resolve such complex processes, tree-structured neural networks, such as the modular neural network (MNN) (Jacobs et al., 1991; Jacobs and Jordan, 1993; Jordan and Jacobs, 1994), could be employed to model the nonlinear systems by dividing the input space into a set of regions, each of which is approximated with a single ANN model. In general, a MNN is constructed from two types of network, namely expert networks and a gating network. Expert networks may be of a variety of different types of neural networks. Each network is designed for a particular task. A gating network receives the input vector and produces as many outputs as there are expert networks. These outputs must be nonnegative and sum to unity, representing the weights of the output of each expert network. A weighted sum of the outputs of the experts forms the MNN output. During training, the weights of the expert and gating networks are adjusted simultaneously using the backpropagation algorithm.

There are many local-nonlinear type DAC models similar to MNN but with different names, such as hybrid ANNs, integrated ANNs, threshold (or domain-dependent, range-dependent) ANNs, committee machine and so on. Some of them could be considered as special cases of MNN (e.g., threshold ANN), and some others have different ways of combining the separate expert neural networks. For example, instead of using a gating network to mediate the

competition of expert networks, some hybrid neural networks use a fuzzy logic model to link individual expert networks into an integrated modeling system, or use the cluster analysis technique in which the input space is first divided into several clusters, and then a separate expert network is fitted to each cluster. There are already some examples of applications of modular or hybrid ANN models in the field of hydrological modeling. Zhang and Govindaraju (2000) examined the performance of modular networks in predicting monthly discharges over three medium-sized watersheds based on the Bayesian concept. See and Openshaw (1999) developed a fuzzy-logic based hybrid model to forecast river level, in which the forecasting data set is split into subsets before training with a series of neural networks. Abrahart and See (2000) presented a hybrid network solution based the clustering of the hydrological records with a self-organizing map (SOM) neural network. Hu et al. (2001) developed range-dependent hybrid neural networks (RDNN), which are virtually threshold ANNs, to forecast annual and daily streamflows. Pal et al. (2003) proposed a hybrid ANN model that combines the self-organizing feature map (SOFM) and the MLP network for temperature prediction, where the SOFM serves to partition the training data.

In addition, perhaps the most thorough DAC algorithm is the nearest neighbor method (NNM), in which a time series is reconstructed in a multi-dimensional state-space and then local approximation models (parametric or non-parametric) are fitted to the nearest neighbors in the space. Parametric NNM could be linear or nonlinear depending on what type of local parametric models is used. NNM has been applied to streamflow forecasting by many researchers (e.g., Yakowitz and Karlsson, 1987; Bordignon and Lisi, 2000; Sivakumar et al., 2001).

In this study, three types of hybrid ANN models, namely, the threshold ANN (TANN), the cluster-based ANN (CANN), and the periodic ANN (PANN), are used to forecast daily streamflows, and the model performance is compared with normal MLP-ANN models.

6.3.2 Threshold ANN (TANN)

Threshold ANN (TANN), analogous to the threshold regression model, divides the streamflow series into several regimes according to some threshold values, and then builds one ANN model for each regime.

To fit a threshold model, whether it is a threshold linear model or a threshold nonlinear model, the main concern is to determine the threshold value. Tong (1983) suggested that the location of modes and antimodes of the univariate histogram for x_t and the bivariate histogram for (x_t, x_{t-i}) (i = 1, 2, …, p, say) may assist in the identification of the threshold parameters. Histogram is the simplest estimator of the probability density function (pdf). An improved histogram estimate is the kernel density estimate, which can overcome the histogram's sensitivity to choice of data origin and bin (i.e., class interval) width (Silverman, 1986). The kernel pdf is estimated as

$$f(x) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{x - x_i}{h} \right) \tag{6.4}$$

where n is the sample size, $x_i = 1, \ldots, n$ are the data, h is the bandwidth, and $K(u)$ is the kernel function, inthis case the Gaussian kernel function.

The kernel density estimate of pdf is calculated for the daily streamflow of the Yellow River at TNH. Figure 6.2 shows that the streamflow series has a bimodal pdf, whose two modes may indicate two regimes of the flow process dynamics, and the antimode may correspond to a point of separation of the two regimes. The low-flow regime is around 195 m^3/s (≈ exp(5.273)), and the high-flow regime around 671 m^3/s (≈ exp(6.508)), the two regimes being separated at the antimode, near 381 m^3/s (≈ exp(5.942)).

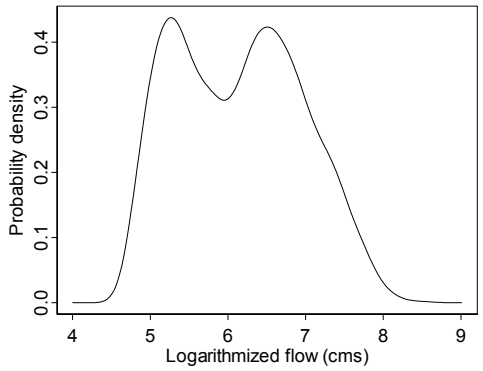


Figure 6.2 Probability density estimate of the log-transformed daily average discharges of the Yellow River at TNH

Therefore, we divide the streamflow states in the reconstructed state-space (described in Section 2.2) into two regimes, namely, one regime composed of the streamflow states whose average discharges are larger than 381 m^3/s, and another one composed of the streamflow states less than 381 m^3/s. Then one MLP network is fitted to the streamflows in each regime.

6.3.3 Cluster-Based ANN (CANN)

For the cluster-based ANN (*CANN*) model, we divide the streamflow state vectors in the reconstructed state space into several clusters based on cluster analysis techniques, and then build one ANN model for each cluster. The fuzzy C-means (FCM) clustering technique (see Section 3.3.2) is applied in the present study to do the clustering, so that we can cluster the streamflow state vectors both softly and crisply.

The streamflow states in the reconstructed state space of the Yellow River at TNH are grouped into 3 clusters with FCM clustering method. The clustering result of the streamflow states in two typical years is shown in Figure 6.3. Compared with Figure 2.9(a) in Chapter 2, it is shown that, when grouping the streamflow states into 3 clusters, the three groups basically represent three different daily flow regimes, i.e., low flow, medium flow and peak flow. For some years, the three groups may be much more fragmented.

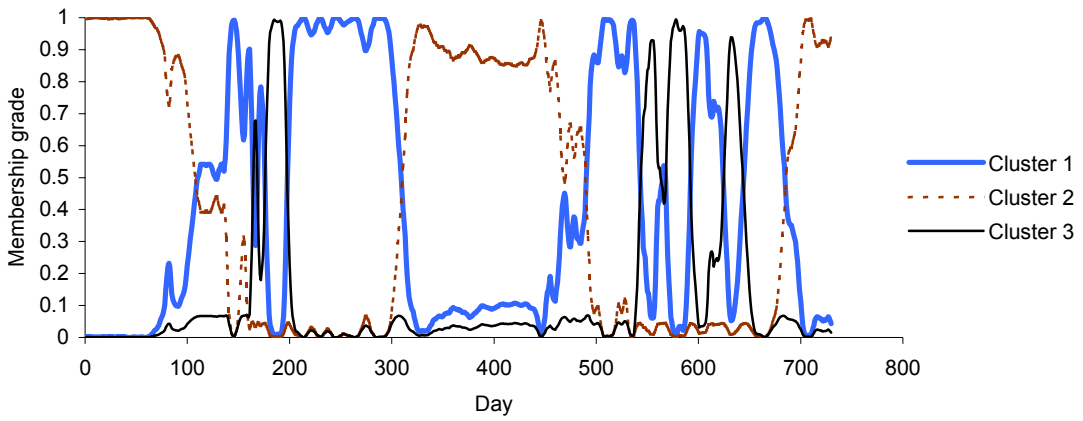


Figure 6.3 Membership grades of the daily streamflow states in two year with the FCM of 3 clusters

After the streamflow state vectors in the reconstructed state space are grouped into 3 clusters, we then fit one ANN to the streamflow states in each cluster, three ANN models in total. When making a forecast from a current streamflow state, the final output of the overall CANN model is a weighted average of the outputs of the three ANN models. The weights are determined according to the distance between the current streamflow state and each cluster center.

6.3.4 Periodic ANN (PANN) Model

The periodic ANN (PANN) model essentially is a group of ANN models, each of which is fitted to the streamflows that occurred in a separate "*season*" (notice that, *season* here does not mean a real season. It may be a group of neighbouring days or months over the year). The idea of fitting the PANN model to daily streamflows is the same as that of fitting periodic autoregressive (PAR) model (in Section 3.3), whereby a periodic AR model is fitted to daily streamflow based on partitioning of days over the year with clustering techniques. We will fit an ANN model to each seasonal partition, and all the ANN models together comprise the periodic ANN model.

The PANN differs from CANN in that, in CANN we clusters the discharges in the reconstructed phase-space, whereas in PANN we cluster the days over the year according to the characters of the discharges of each day. When partitioning the days over the year with the clustering analysis method, the raw average daily discharge data and the autocorrelation values at different lag times (1 ~ 10 days) are used. The daily discharge data and the autocorrelation coefficients are organized as a matrix of the size (N+10)×365, where N is the number of years and "10" represents the autocorrelation values at 10 lags. To eliminate the influence of big differences among data values on cluster analysis result, the log-transformation is first applied to the daily discharges before making the cluster analysis.

Then the 365 days over the year are partitioned with the fuzzy c-means clustering described in Section 3.3.2. The cluster result is shown in Figure 6.4. Comparing Figure 2.9(a) and Figure 6.4, we see that if we just follow the clustering result to partition the days over a year into 5

groups, the dynamics of streamflow is not well captured because cluster 2 and 3 in Figure 6.4 mix the streamflow rising limb and falling limb shown in Figure 2.9(a). Therefore, according to the FCM clustering result, and considering the dynamics of the streamflow process, we partition the 365 days over the year into 7 hard segments, as listed in Table 6.3.

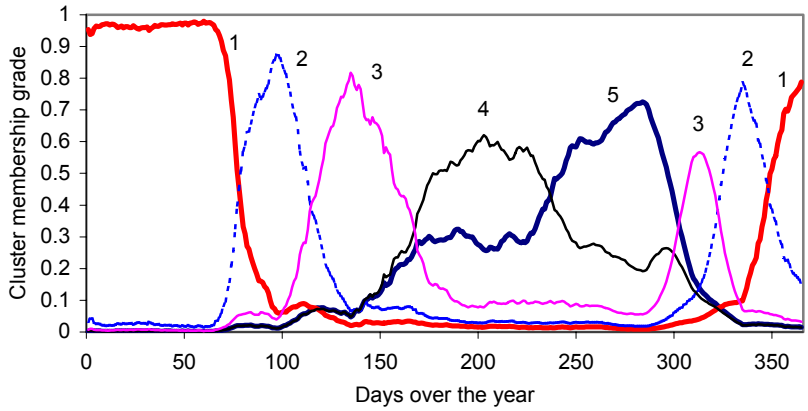


Figure 6.4 The membership grades of the days over the year for the daily streamflow at TNH with the FCM of 5 clusters

Table 6.3 Hard partitioning of the days over the year for the daily streamflow at TNH

Pratition	1	2	3	4	5	6	7
Day span	1-78, 349-365/366	79-114	115-167	168-237	238-302	303-322	323-348

The days over the year can also be softly partitioned, such that each day could belong to several partitions. The essence of soft partitioning is the determination of the membership grade (or membership function) of each partition, which is one of the most crucial issues in the foundation of fuzzy reasoning. In this study, following the pattern of the FCM clustering result, the membership grade is formed intuitively for the days over the year, as shown in Figure 6.5.

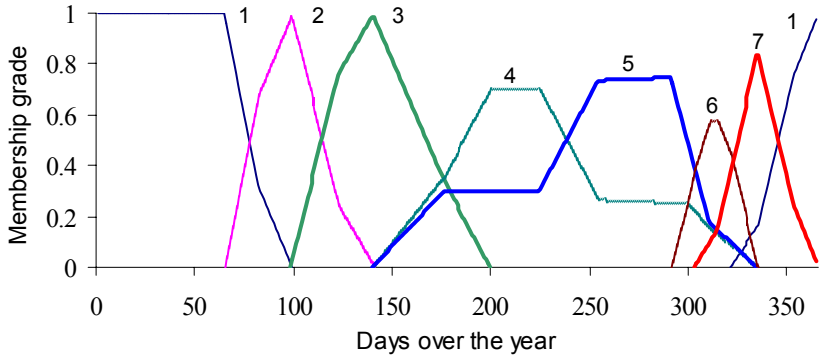


Figure 6.5 Soft partitioning of the days over the year for the daily streamflow at TNH according to the FCM with 5 clusters

Based on the partitioning results, the PANN model is constructed, whereby one MLP model is fitted to each (hard) seasonal partition. When forecasting, different MLP models are used depending on what season partition the date for which the forecasted is made lies in. The fitted PANN model can be applied to forecasting in two ways: based on hard partitioning (referred to as *hard PANN*) and based on soft partitioning (referred to as *soft PANN*). When soft partitioning is applied, one day may belong to several season partitions. Correspondingly, the final output would be a weighted average of the outputs of the several ANN models fitted for these seasonal partitions. The weight is equal to the membership grade obtained with the FCM clustering results.

6.3.5 Model Performance of Hybrid ANN Models

The components of all the hybrid ANN models have the same 5-3-1 MLP structure, as identified for normal MLP-ANN ANN models considered in Section 6.2.4. According to the comparison among different data pre-processing methods described in Section 6.2.4, deseasonalization is the best choice of pre-processing method. However, because TANN and CANN divide up the streamflow states into different regimes according to their values, it is more convenient to standardize the data than to deseasonalize the data for building TANN and CANN models. Therefore, the input/output data are log-transformed and then standardized before fitting the TANN model and the CANN model, and log-transformed and then deseasonalized before fitting the PANN models.

Table 6.4 lists performance evaluation results of the 1- to 10-day ahead forecasts with these 4 models (including hard PANN and soft PANN). It is shown that the PANN model performs best among the three variations of hybrid ANN models under study, and the soft PANN performs better than the hard PANN. The scatter plots of one-day, five-day and 10-day ahead forecasts versus observed daily discharges for year 1999 and 2000 with the soft- partition based PANN are shown in Figure 6.6. The hydrograhs of one-day ahead, five-day ahead, and ten-day ahead forecasted daily discharge versus the observed hydrograph are shown in Figure 6.7, 6.8, and 6.9.

Because the model performance difference measured with *CE* and *SACE* is not very informative (comparing Table 6.2 and Table 6.4), in order to give a more clear comparison between the PANN models and the normal MLP-ANN model that is fitted to the deseansonalized data without rescaling or any grouping procedure, the RMSE measure is used to evaluate the model performance. The results are listed in Table 6.5, where it is seen that for the shorter lead times, the soft PANN performs better than the normal MLP model, but the advantage vanishes as the lead time increases ($\geq$ 4 days).

To make a comparison of forecasts for each season, we list the performance evaluation results of normal MLP-ANN and soft PANN in Table 6.6 to 6.13. The results show that except for daily flows in summer, soft PANN improves the forecasts at short lead time as well as long lead time comparing with the normal MLP-ANN model.

Table 6.4 Forecsast performances with hybrid ANN models for daily flows at TNH

Lead time (days)		1	2	3	4	5	6	7	8	9	10
CE	CANN	0.984	0.942	0.898	0.856	0.815	0.774	0.735	0.698	0.665	0.633
	TANN	0.989	0.967	0.941	0.916	0.892	0.867	0.839	0.811	0.784	0.755
	hard PANN	0.989	0.966	0.942	0.919	0.895	0.870	0.844	0.818	0.793	0.769
	soft PANN	0.990	0.967	0.943	0.920	0.897	0.873	0.848	0.823	0.799	0.776
SACE	CANN	0.969	0.893	0.810	0.733	0.657	0.581	0.508	0.441	0.378	0.320
	TANN	0.980	0.938	0.891	0.845	0.799	0.753	0.702	0.650	0.599	0.546
	hard PANN	0.980	0.937	0.892	0.849	0.805	0.759	0.711	0.663	0.617	0.571
	soft PANN	0.981	0.94	0.895	0.852	0.808	0.765	0.718	0.671	0.628	0.584

Table 6.5 RMSE measures with normal MLP-ANN and PANN models to daily flows at TNH

Lead time (days)	1	2	3	4	5	6	7	8	9	10
MLP	45.9	80.3	105.0	123.9	140.6	156.2	171.0	184.5	196.6	208.3
hard PANN	45.5	80.6	105.7	125.3	142.4	158.1	173.3	187.2	199.5	211.0
soft PANN	44.8	79.2	104.4	124.1	141.1	156.3	171.1	184.8	196.7	207.9

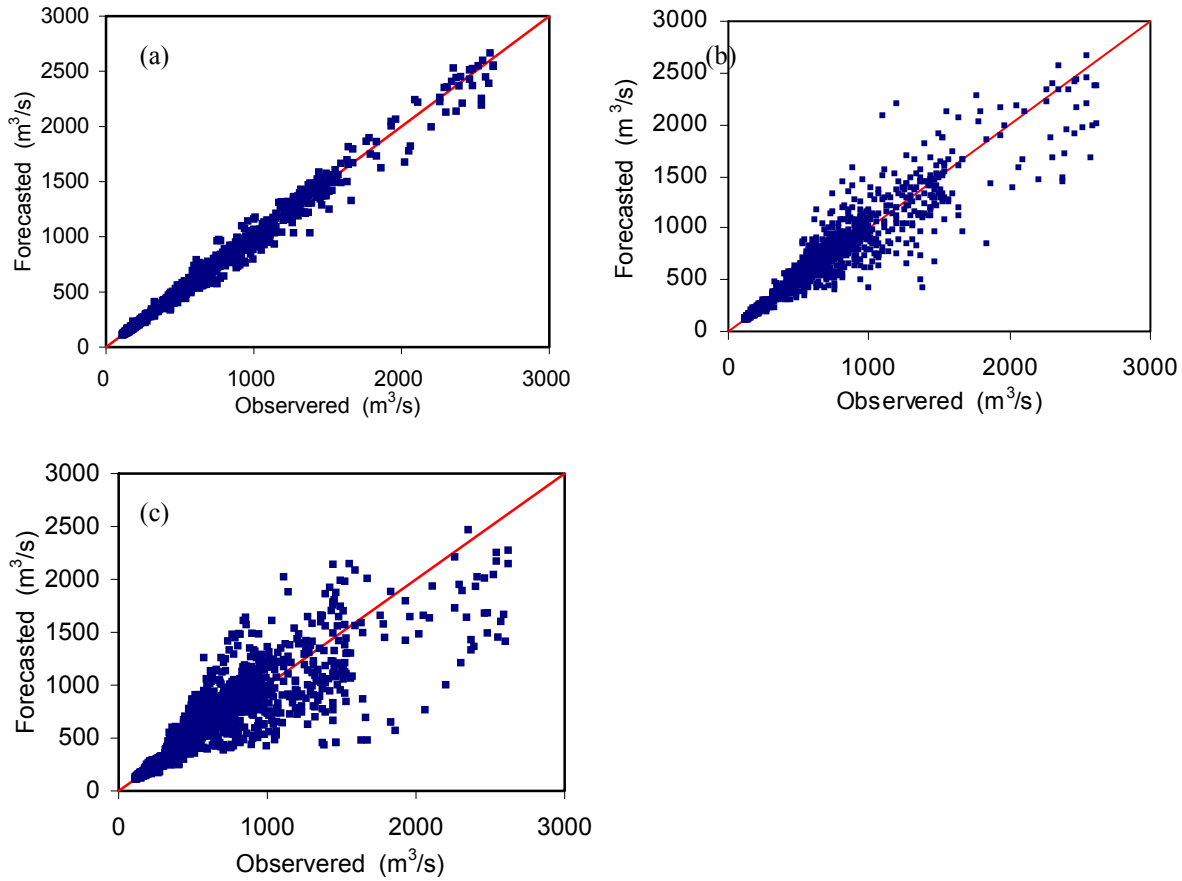


Figure 6.6 Scatter plots of (a) one-day; (b) five-day and (c) 10-day ahead discharge forecasts for year 1996-2000 with the PANN based on soft seasonal partitions

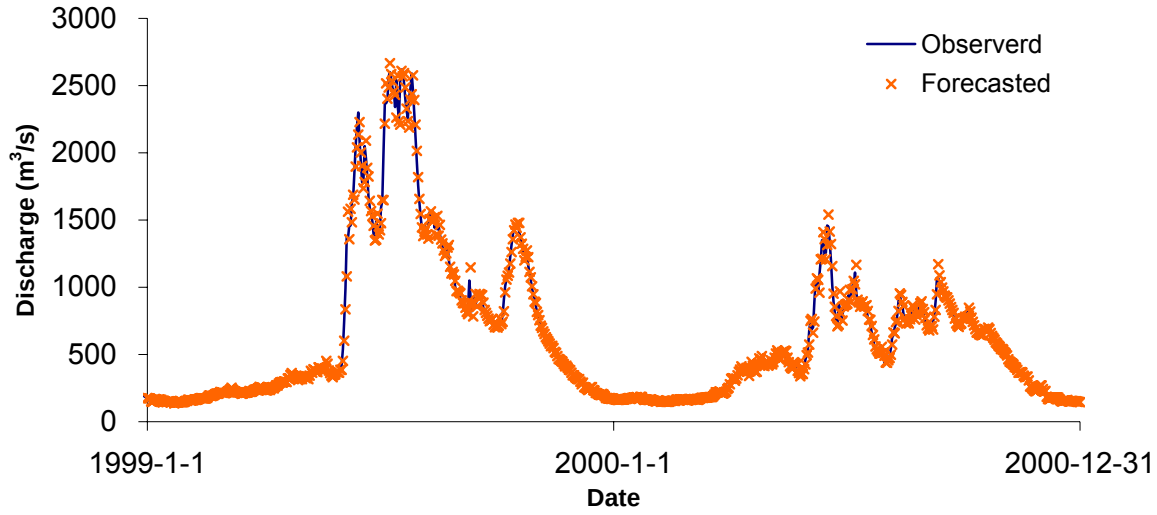


Figure 6.7 One-day ahead forecasts with the PANN based on soft seasonal partitions

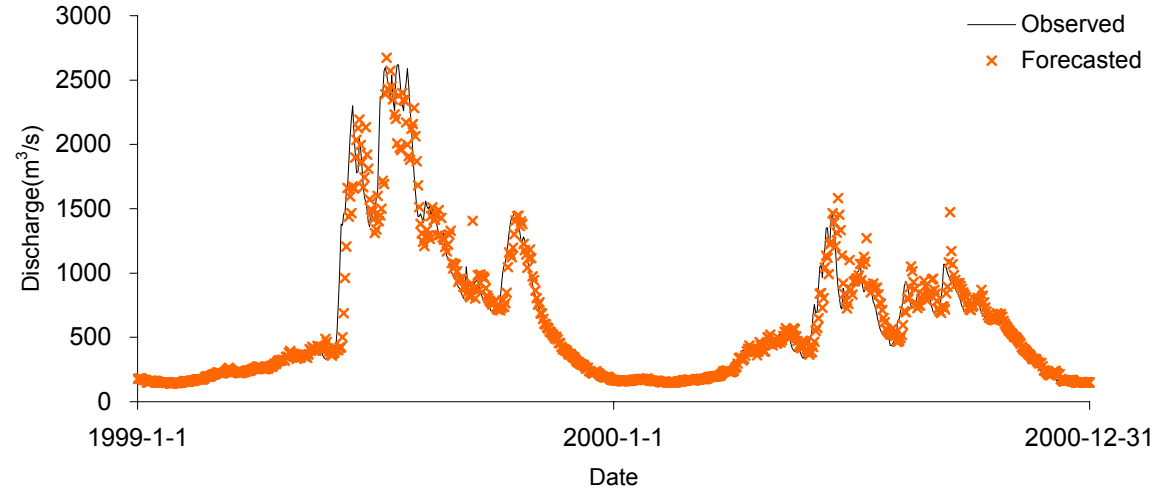


Figure 6.8 Five-day ahead forecasts with the PANN based on soft seasonal partitions

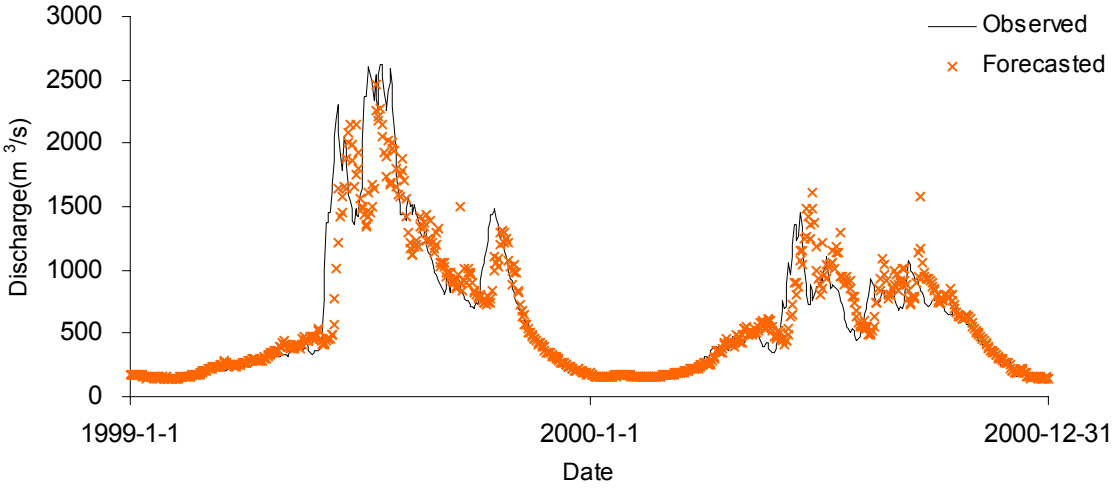


Figure 6.9 Ten-day ahead forecasts with the PANN based on soft seasonal partitions

Table 6.6 Forecast performance with the normal MLP-ANN model for daily flow in spring at TNH

Lead time (day)	MAE	MAPE	RMSE	MSRE	*CE*	r^2
1	18.12	0.042	31.2	0.003	0.974	0.974
2	32.33	0.074	55.2	0.01	0.918	0.921
3	40.51	0.092	71.7	0.016	0.863	0.87
4	47.33	0.106	83.5	0.021	0.816	0.828
5	52.98	0.117	93.7	0.025	0.771	0.787
6	60.27	0.13	106.1	0.031	0.714	0.735
7	70.5	0.147	124.6	0.038	0.635	0.658
8	80.14	0.162	141.5	0.046	0.565	0.591
9	87.94	0.173	156.6	0.053	0.508	0.535
10	95.11	0.184	169.0	0.059	0.463	0.491

Table 6.7 Forecast performance with the normal MLP-ANN model for daily flow in summer at TNH

Lead time (day)	MAE	MAPE	RMSE	MSRE	*CE*	r^2
1	56.88	0.058	81.2	0.006	0.974	0.974
2	102.91	0.105	142.2	0.018	0.92	0.921
3	133.21	0.136	186.1	0.031	0.863	0.864
4	155.17	0.159	219.6	0.043	0.81	0.811
5	177.3	0.183	248.7	0.055	0.757	0.759
6	200.62	0.207	274.9	0.069	0.703	0.706
7	219.54	0.226	297.1	0.082	0.654	0.658
8	236.04	0.244	317.5	0.095	0.606	0.611
9	250.32	0.262	335.4	0.108	0.561	0.568
10	263.89	0.277	353.8	0.122	0.512	0.521

Table 6.8 Forecast performance with the normal MLP-ANN model for daily flow in autumn at TNH

Lead time (day)	MAE	MAPE	RMSE	MSRE	*CE*	r^2
1	16.47	0.024	26.5	0.001	0.991	0.991
2	28.86	0.041	46.4	0.003	0.973	0.974
3	37.95	0.054	60.8	0.005	0.954	0.955
4	45.54	0.064	73.4	0.008	0.932	0.934
5	52.28	0.073	86.2	0.01	0.906	0.909
6	58.46	0.082	97.5	0.013	0.879	0.884
7	64.12	0.091	107.8	0.016	0.853	0.859
8	68.32	0.096	117.0	0.018	0.826	0.835
9	72.23	0.101	125.3	0.02	0.8	0.812
10	75.44	0.106	132.8	0.022	0.774	0.79

Table 6.9 Forecast performance with the normal MLP-ANN model for daily flow in winter at TNH

Lead time (day)	MAE	MAPE	RMSE	MSRE	*CE*	r^2
1	5.02	0.03	7.2	0.002	0.968	0.969
2	7.15	0.043	10.1	0.003	0.93	0.934
3	8.22	0.049	11.6	0.005	0.897	0.905
4	8.74	0.053	12.6	0.005	0.866	0.88
5	9.44	0.057	13.3	0.006	0.834	0.854
6	9.91	0.061	13.7	0.007	0.806	0.831
7	10.42	0.064	13.9	0.007	0.784	0.812
8	10.72	0.066	14.2	0.007	0.759	0.791
9	10.92	0.068	14.3	0.007	0.744	0.774
10	11.09	0.069	14.6	0.008	0.723	0.752

Table 6.10 Forecast performance with soft PANN model for daily flow in spring at TNH

Lead time (day)	MAE	MAPE	RMSE	MSRE	*CE*	r^2
1	17.48	0.041	29.9	0.003	0.976	0.976
2	31.46	0.073	53.5	0.01	0.923	0.924
3	39.84	0.091	69.3	0.015	0.872	0.874
4	46.45	0.105	80.5	0.02	0.829	0.833
5	52.22	0.117	89.9	0.024	0.79	0.795
6	58.93	0.129	101.1	0.029	0.741	0.748
7	68.52	0.145	118.5	0.035	0.67	0.677
8	77.63	0.159	134.5	0.042	0.607	0.615
9	85.28	0.17	148.9	0.048	0.555	0.563
10	92.39	0.181	160.6	0.054	0.515	0.523

Table 6.11 Forecast performance with soft PANN model for daily flow in summer at TNH

Lead time (day)	MAE	MAPE	RMSE	MSRE	*CE*	r^2
1	54.55	0.055	79.4	0.006	0.975	0.975
2	99.95	0.102	140.7	0.018	0.922	0.922
3	131.9	0.135	186.4	0.031	0.863	0.864
4	156.29	0.159	222.3	0.044	0.805	0.807
5	177.3	0.182	252.3	0.057	0.749	0.752
6	201.43	0.207	278.7	0.071	0.695	0.698
7	222.37	0.23	301.8	0.084	0.643	0.646
8	241.76	0.25	323.0	0.098	0.592	0.596
9	258.11	0.27	341.0	0.112	0.547	0.551
10	272.7	0.287	358.6	0.126	0.499	0.504

Table 6.12 Forecast performance with soft PANN model for daily flow in autumn at TNH

Lead time (day)	MAE	MAPE	RMSE	MSRE	*CE*	r^2
1	15.47	0.023	25.9	0.001	0.992	0.992
2	26.44	0.039	45.2	0.003	0.975	0.976
3	34.68	0.051	58.3	0.005	0.958	0.96
4	42.16	0.06	69.9	0.007	0.939	0.942
5	49.93	0.072	82.4	0.01	0.914	0.92
6	56.87	0.082	92.8	0.013	0.891	0.9
7	62.39	0.09	102.6	0.016	0.867	0.878
8	67.3	0.097	111.6	0.018	0.842	0.856
9	71.59	0.103	120.2	0.02	0.816	0.833
10	75.18	0.109	128.0	0.023	0.79	0.812

Table 6.13 Forecast performance with soft PANN model for daily flow in winter at TNH

Lead time (day)	MAE	MAPE	RMSE	MSRE	*CE*	r^2
1	4.9	0.029	7.0	0.002	0.97	0.97
2	6.81	0.041	9.6	0.003	0.937	0.939
3	7.86	0.047	11.0	0.004	0.908	0.913
4	8.54	0.052	12.0	0.005	0.879	0.888
5	9.24	0.056	12.7	0.005	0.849	0.864
6	9.71	0.06	13.2	0.006	0.822	0.841
7	10.11	0.062	13.4	0.006	0.8	0.823
8	10.3	0.064	13.7	0.007	0.776	0.801
9	10.5	0.065	13.9	0.007	0.757	0.782
10	10.83	0.068	14.3	0.007	0.735	0.758

6.4 Some Discussions about Building ANN Models

6.4.1 About the Pre-processing of Input/Output Data

It is recognized that data pre-processing can have a significant effect on model performance (e.g., Maier and Dandy, 2000). It is commonly considered that, because the outputs of some transfer functions are bounded, the outputs of an MLP-ANN model must be in the interval [0,1] or [-1,1] depending on the transfer function used in the neurons. Some authors suggest using even smaller intervals for streamflow modelling, such as [0.1, 0.85] (Shamseldin, 1997), [0.1, 0.9] (e.g., Hsu et al., 1995; Abrahart and See, 1999) and [-0.9, 0.9] (e.g., Braddock et al., 1998), so that extreme (high and low) flow events occurring outside the range of the calibration data may be accommodated. For model building convenience, it is common that both input and output data are rescaled before fitting ANN models (e.g., Hsu *et al.*, 1995; Pajurkar et al., 2004). However, the advantage of rescaling the data into a small interval is not supported by this study. In the case of forecasting for the daily flow at TNH, not only is the general performance of the MLP-ANN with standardization pre-processing better than the MLP-ANN with rescaling pre-processing, but the performance for low flow and high flow periods is better also.

There are two explanations for this result. On the one hand, suppose we only consider the relationship between one input and one neuron with a hyperbolic tangent transfer function *tansig*(x). To rescale the input data to [-1, 1] would limit the output range of the *tansig*(x) function approximately to [-0.7616, 0.7616]. To rescale the input range to [-0.9, 0.9] would further shrink the output range approximately to [-0.7163, 0.7163]. Both 0.7616 and 0.7163 are still far away from the extreme limits of the *tansig*(x) function, whereas such a small output data range will make the output less sensitive to the change of the weights between the hidden layer and output layer, and will therefore possibly make the training process more difficult. On the other hand, because the neurons in an ANN are combined linearly with many weights (as in a MLP model), any rescaling of the input vector can be effectively offset by changing the corresponding weights and biases. Therefore, to standardize the input/output data may be a better choice than to rescale them into a small interval (e.g., [–1, 1]), especially when the data size is large enough to include possible data extremes.

A related subject is the choice of transfer function. The most commonly used transfer functions are the logistic sigmoid function and hyperbolic tangent function, the logistic sigmoid function (logsig(x) = $1/(1+e^{-x})$) being much more frequently used than the hyperbolic tangent function (e.g., Hsu et al., 1995; Minns and Hall, 1996; Zealand et al., 1999; Abrahart and See, 2000). However, Kalman and Kwasny (1992) argue that the hyperbolic tangent transfer function should be used, and the empirical results obtained by Maier and Dandy (1998) also indicate that not only is training with the hyperbolic tangent function faster than

training with the sigmoidal transfer function, but the predictions obtained using networks with the hyperbolic tangent are slightly better than those with the sigmoid transfer functions. However, no comparison of the logistic sigmoid function and hyperbolic tangent function is carried out in this present study. Kalman and Kwasny (1992) show mathematically that the hyperbolic tangent function possesses particular properties that make it appealing for use while training. We only make a heuristic analysis about this problem here. Suppose we only consider the relationship between one input and one neuron with a logistic sigmoid function. When the input is in the interval [-1,1], then the output is in the range 0.26894 ~ 0.73106 (logsig(-1) ≈ 0.26894, logsig(1) ≈ 0.73106), less than 1/3 of that of the hyperbolic tangent function. It is possible that, because the output of the logistic sigmoid function is constrained into a much smaller range than that of the hyperbolic tangent function, this results in the output of the ANN with logistic sigmoid functions being less sensitive to the change of connection weights, consequently making the training of the ANN with the logistic sigmoid function more difficult than that of ANNs with the hyperbolic tangent function.

6.4.2 About the Selection of the Optimal ANN Model

It is well known that an ANN is sensitive to the initial weights. When building an ANN model, every time we train it, we may get a different set of parameters. One way to increase the likelihood of obtaining near-optimum set of parameters is to train the ANN with different independent initial weights, and then the model builders choose the best ANN model according to the training or validation performance among many competitive ANN models (e.g., Pajurkar et al., 2004). However, in practical applications, real validation data are future values which virtually do not exist when we fit an ANN model, therefore we cannot choose the best model according to the validation performance. On the other hand, if we choose the best ANN model according to the training performances, best training performance actually does not guarantee best validation performance, especially for multi-step forecasts, although best training performance mostly indicate best validation performance. That means, suppose we split all the available data into 3 parts, namely one training data set, and two validation data sets (A and B), a model which is the best for validation set A does not necessarily perform the best for validation set B. This is because the generality of an ANN model may be limited even though the size of training data may be sufficiently large, especially in the case of a natural watershed system which may have more or less underlying changes due to climate changes and human activities. On the other hand, there may be many parameter sets within a model structure that are equally acceptable as simulators of a dynamical process of interest. Therefore, the attempt to choose a best ANN model is not sound. Instead, it is much better to make the ensemble forecast (e.g. Sharkey, 1996). One robust way of making ensemble forecast is simply taking the average of the forecasts of an ensemble of ANN models. To minimize the possibility that some of the ensemble members are poorly trained due to the effects of local minima in the error surface, we may train a number of networks (say, 10) first, then choose several best ones (say, 5) as ensemble members according to their training performance. This is the approach taken in this study.

6.5 Combining the Forecasts of ARMA, PAR, MLP and PANN

6.5.1 Introduction to Forecast Combination

There are various models available for streamflow forecasting nowadays. However, when building a forecasting model, it is not an easy task to choose a suitable model, because on one hand, no model is powerful and general enough to outperform the others for all types of catchments and under all circumstances; on the other hand, every model has some degree of uncertainty, including structure uncertainty and parameter uncertainty. Instead of using a single model, we may alternatively handle the model selection problem by combining the forecasts from several models so as to obtain a more reliable and accurate output than would be obtained by selecting a single model. After the seminal paper of Bates and Granger (1969), many combining methods have been proposed such as simple average method, weighted average method, Bayesian methods (Bunn, 1975; Winkler, 1981), the minimum-variance method (Dickinson, 1975) and regression based methods (Granger and Ramanathan, 1984). More recently, Deutsch et al. (1994) proposed to combine forecasts using changing weights derived from switching regression models or from smooth transition regression models; Donaldson and Kamstra (1996) developed a neural network based approach to the nonlinear combination of forecasts; Fiordaliso (1998) proposed a nonlinear forecast combining method, in which a first order Takagi-Sugeno fuzzy system is used to combine a set of individual forecasts; He and Xu (2005) proposed to use the self-organizing data mining algorithms to combine forecasts. Many studies and empirical tests have shown the advantage of combining forecasts in practice (see Clemen, 1989).

While the combined forecasting has a long history in the econometrics community, it has not received much attention in the field of hydrological forecasting until recently. In the pioneering work of McLeod et al. (1987), they showed that significant improvements in forecast performance can be achieved by combining forecasts produced by different types of models applied to quarter-monthly river flows. Shamseldin et al. (1997) examined three different combination methods in the context of flood forecasting, namely, the simple average method, the weighted-average method and the neural network method, and confirmed that better discharge estimates can be obtained by combining the outputs of different models. See and Openshaw (2000) used four different approaches (i.e., an average, a Bayesian approach, and two fuzzy logic models) to combine the river level forecasts of three models (i.e., a hybrid neural network, an autoregressive moving average model, and a simple fuzzy rule-based model), and found that the addition of fuzzy logic to the crisp Bayesian approach yielded overall results that were superior to the other individual and integrated approaches. Xiong et al. (2001) showed that the first-order Takagi-Sugeno fuzzy system works almost the same as the weighted average method and neural network method in combining five rainfall-runoff models. Coulibaly et al. (2005) showed that, using weighted average method to combine three dynamically different models can significantly improve the accuracy of the daily reservoir inflow forecast for up to 4 days ahead.

While many studies confirm the effectiveness of the weighted average method (WAM) for hydrological applications (e.g., McLeod et al., 1987; Shamseldin et al., 1997; Xiong et al., 2001), none of the previous studies has considered the effect of how the weights are estimated on the effectiveness of WAM. In this section, we will investigate the possibility of improving the daily streamflow

forecasts by combining four dynamically different models that have been built in previous chapters/sections for the daily streamflow process of the upper Yellow River at TNH. The four models used here include: (i) the ARMA model, which captures the overall linear autocorrelation structure of the streamflow process; (ii) the PAR model, which captures the seasonal difference of linear autocorrelation structure; (iii) the MLP-ANN model, which captures the overall nonlinear autocorrelation structure; and (iv) the PANN model, which captures the seasonal difference of nonlinear autocorrelation structure. The dynamical difference of the four models makes it possible to improve the forecast accuracy by combining them together.

6.5.2 Methods of Combining Forecasts

Forecast combination methods may be roughly broken into two categories. The first one is the ensemble approach, by which a set of forecasts are produced on the same task with different models (or one model with different inputs), and then the forecasts are combined. The second one is the modular approach, under which a task or problem is divided into a number of subtasks (regimes), and the complete task solution requires the contribution of all of the individual regimes.

6.5.2.1 *Ensemble approach*

Essentially, the ensemble combination is a weighted average of the outputs of ensemble members. While the ensemble prediction technique is normally used to provide probabilistic predictions as in ESP, it may also be extended to be used as a forecast combination technique. What differs between using ensemble prediction technique in ESP and using ensemble prediction technique for combining forecasts is that, in ESP the ensemble members are composed of forecasts from a single model with different inputs, whereas in forecast combination different models (either different parameters, or different structures, or even difference types) with the same (or basically the same) inputs are used.

There are two main issues about ensemble combination: First, how to select a set of models and generate an ensemble of forecasts to be combined; and second, how to estimate the combining weights so as to minimize the out-of-sample forecast errors. The selection of the ensemble models should provide the information of a specific process from different perspectives. In this study, we choose ARMA-type model and ANN-type model aiming to combine the strength of linear and nonlinear approximation ability of ARMA and ANN respectively. Furthermore, to capture the seasonality of streamflow processes, periodic models (PAR and PANN) are applied. As for the estimation of combining weights, some studies show that equally weighted combination, namely, the simple average method (SAM), can produce forecasts that are better than those of the individual models (Makridakis et al., 1982), and its accuracy depends mainly on the number of the models involved and on the actual forecasting ability of the specific models included in the simple average (Makridakis and Winkler, 1983). Because of its robustness, the SAM method has been consistently the choice of many researchers (see Clemen, 1989). However, when some of the individual models selected for combination appear to be consistently more accurate than others, in which case the use of the SAM for combination can be quite inefficient (Armstrong, 1989), the use

of weighted average method (WAM) would be considered. One of the most common procedures used to estimate the combining weights is to perform the ordinary least squares regression (see e.g., Crane and Crotty, 1967; Winkler and Makridakis, 1983; Granger and Ramanathan, 1984):

$$y_{t+1} = a_0 + \sum_{j=1}^{k} a_j f_{t,j} + \varepsilon_{t+1} \tag{6.5}$$

where $f_{t,j}$ is the one step ahead forecast made at time t of y_{t+1} with model i; a_0 is a constant term; and a_j is the regression coefficient.

Another common method to estimate the combining weights is the optimal method (Bates and Granger, 1969), in which the linear weights are calculated to minimize the error variance of the combination (assuming the unbiasedness for individual forecast). Granger and Ramanathan (1984) showed that the optimal method is equivalent to a least squares regression (referred to as Equality Restricted Least Squares, ERLS) in which the constant is suppressed and the weights are constrained to sum to one. One more option for estimating the combining weights is the Nonnegativity Restricted Least Squares (NRLS) regression (see Gunter, 1992), in which the weights are constrained to be nonnegative. Aksu and Gunter (1992) examined the relative accuracy of OLS, ERLS and NRLS and SAM combined forecasts using 40 economic series, and the empirical results revealed that NRLS and SA combinations almost always outperform OLS and ERLS combinations, while NRLS combinations are at least as robust and accurate as SA combinations.

6.5.2.2 *Modular approach*

The *modular approach is* based on the principle of divide-and-conquer (DAC), which deals with a complex problem by breaking it into simple problems whose solutions can be combined to yield a solution to the complex problem (Jordan and Jacobs, 1994). In a narrow sense, the ensemble approach and the modular approach are distinct, in that the modular approach assumes that each data point is assigned to only one model whereas with ensemble combination, each data point is likely to be treated by all the component models in an ensemble. However, the two approaches may be mixed up in a broad sense, in that on the one hand, the component model in the ensemble approach may be a modular model (e.g., a PAR model may be viewed as a modular AR model), on the other hand, each component in a modular combination can take the form of an ensemble of models. In fact, the modular *approach* may be viewed as a modeling strategaty as well as a forecast combination approach. When each component in a modular combination is made of a single model rather than an ensemble of several models, the *modular approach* is simply reduced to a hybrid modeling approach.

The modular combination may be expressed in a fashion of switching regime model (see e.g., Goldfeld and Quandt, chapter 9, 1972) where the model parameters change over time:

$$y_t = \sum_{j=1}^{k} g(t \in I_j) f_{t,j} \tag{6.6}$$

where I_j is the regime; $g(t \in I_j) = 1$ if $t \in I_j$ and $g(t \in I_j) = 0$ if $t \notin I_j$; $f_{t,j}$ denotes the forecast for regime I_j at time t. The forecast $f_{t,j}$ may come either from a single model fitted to regime I_j, or more generally, from an ensemble combination of the forecasts from several models. Consequently, there are three major issues about the modular combination method: first, the division of the problem under concern; second, the selection of models for each regime; and third, the method of combination if more than one model is chosen for each regime. A sensible division relies on a clear understanding of the problem. Because streamflow generation processes, especially daily streamflow processes, usually have pronounced seasonal means, variances, and at the same time, dependence structures and the under-lying mechanisms of streamflow generation are likely to be quite different during low, medium, and high flow periods, hence, three approaches may be taken to divide a streamflow process: use threshold values to divide the streamflow regimes; cluster the streamflow process into several domains (e.g., low flow, medium flow and flood); or, partition the streamflow process according to the seasonal difference. Hu et al. (2001) developed a threshold-based ANN model to make streamflow forecsts for the Yangtze River. In the studies of Zhang and Govindaraju (2000), See and Openshaw (2000) and Xiong et al. (2001), the model combinations are fundamentally based on dividing the hydrological process into several domains according to conditions of the hydrological process. However, the comparison of several hybrid ANN models in Section 6.3 shows that, among three hybrid ANN models (i.e., threshold-based hybrid ANN, cluster-based ANN and season-based periodic ANN), season-based periodic ANN performs best, indicating a possibly better generality of dividing the daily streamflow according to seasonal difference. After we divide a time series, we may either chose one optimal model or choose a set of models for each partition of the series. In the cases that we choose a set of models, the same methods for estimating the combining weights for the ensemble combination may apply here too.

6.5.3 Results of Combined Forecasts

In this study, both the ensemble approach and the modular approach are applied. Be aware that, we cannot use the forecasts for the validation period to estimate the weights in either approaches, therefore, one important practical issue for estimating the combining weights is which data are used for the estimation. Some studies use fixed weights that are estimated with the calibration data (e.g., Shamseldin et al., 1997; Xiong et al., 2001), whereas some others considered the combination using changing weights that are estimated with a number of previous forecasts (e.g., McLeod et al., 1987). Taking the selection of the techniques as well as the data used for estimating weights into account, the following four combination methods are compared.

(1) Simple average (SA). A simple average of the forecasts from four competitive models, implying equal weights. In fact, the SAM method has already been used in the construction of the MLP-ANN and PANN model in this study, where the ensemble members are composed of purely neural networks of the same structure.

(2) Rollingly-updated Weighted Average (RWA). Weights are updated on the basis of a rolling-forward window. That is, we estimate the weights for each day according to the

forecasts of previous L days, where L is the length of the rolling window. By trying different values of L, we find that the greater the length of rolling window employed to calculate the weights, the smaller the resulting combined *RMSE*. But no significant improvement is observed after L is larger than *365*. So we choose L to be *365*. Weights are estimated with the nonnegative least-squares regression method (see Lawson and Hanson, 1974), in which the constant is suppressed.

(3) Semi-Fixed Weighted Average (SFWA). Weights are estimated on the forecasts for the previous two years and these weights are unchanged for making forecasts for the current year. When making forecasts for the next year, we update the weights again.

(4) Modular Semi-Fixed Weighted Average (MSFWA). Weights are estimated with a modular approach, where the modules are defined on the basis of seasonal partitions. The weights for each seasonal partition are estimated with the nonnegative regression method based on the previous two years' forecasts. The weights are updated every year when we make forecasts for a new year.

With the above-mentioned methods, we combine the one- to ten-day forecasts of the four models for year 1996 to 2000. We compare the overall performance of five competitive models (including ARFIMA model) and four combination methods by taking the entire validation period (1996 – 2000) into account. The results are listed in Table 6.14. Notice that, the values in bold style indicate the corresponding model behaves best among all models/methods for the specific leading time. An examination of the two tables reveals the following:

Table 6.14 Overall performances of five models and four combination methods

Lead time (days)		1	2	3	4	5	6	7	8	9	10
RMSE	ARMA	45.93	80.21	104.47	122.87	138.60	153.36	167.89	181.53	193.34	203.99
	ARFIMA	45.57	79.59	103.78	122.19	**137.86**	**152.52**	**166.95**	**180.17**	**191.62**	**202.18**
	PAR	45.38	80.68	104.40	125.05	143.60	161.28	178.93	195.47	209.81	223.80
	MLP	45.86	80.30	104.99	123.93	140.65	156.21	170.98	184.53	196.61	208.26
	PANN	**44.75**	79.16	104.36	124.10	141.06	156.32	171.12	184.77	196.68	207.86
	SA	44.87	**78.95**	103.11	**122.16**	138.70	153.99	168.90	182.72	194.77	206.18
	RWA	45.03	79.76	104.71	124.33	141.42	157.01	172.20	186.40	198.90	210.65
	SFWA	44.99	79.04	**102.95**	122.38	139.27	154.77	169.77	183.66	195.75	207.18
	MSFWA	45.11	79.05	103.48	123.16	140.30	156.20	171.67	185.94	198.31	209.96
SACE	ARMA	0.980	0.938	0.895	0.855	0.815	0.774	0.729	0.683	0.640	0.599
	ARFIMA	0.980	0.939	0.896	0.856	**0.817**	**0.776**	**0.732**	**0.687**	**0.646**	**0.606**
	PAR	0.980	0.937	0.895	0.849	0.801	0.750	0.692	0.632	0.576	0.518
	MLP	0.980	0.938	0.894	0.852	0.810	0.765	0.719	0.672	0.628	0.582
	PANN	**0.981**	0.940	0.895	0.852	0.808	0.765	0.718	0.671	0.628	0.584
	SA	0.981	**0.940**	0.898	**0.856**	0.815	0.772	0.725	0.679	0.635	0.591
	RWA	0.980	0.939	0.894	0.851	0.807	0.763	0.715	0.665	0.619	0.573
	SFWA	0.981	0.940	**0.898**	0.856	0.813	0.769	0.723	0.675	0.631	0.587
	MSFWA	0.980	0.940	0.897	0.854	0.810	0.765	0.716	0.667	0.621	0.576

(1) The overall performance of PANN is the best for one-day ahead forecasts according to the fact that it has a minimum *RMSE* value, whereas the ARFIMA model outperforms all the other models/methods for long lead-time forecasts.

(2) SAM generally performs best among the four competitive combination methods, and it outperforms all the 5 individual models for up to 4-day ahead forecasts (except for the PANN model for one-day ahead forecasts), and outperforms three models (except for the ARMA model and the ARFIMA model) for up to 10-day ahead forecasts. The result confirms the robustness of SAM method for improving the forecast accuracy.

(3) Among the other three combination methods (i.e., RWA, SFWA and MSFWA), SFWA performances slightly better than the two others. In addition, RWA has a disadvantage of being much more computation-intensive than the other methods. There is no significant improvement with the modular combination approach for this particular case of the Yellow River.

The reason that SAM perform the best in the four competitive methods probably results from the fact that no individual models selected for combination appears to be consistently more accurate than others, and no individual models appears to be consistently poorer than others. Take a close look at the semi-fixed weights in either SFWA or MSFWA, we find that weights may change significantly from one year to the next, which means that the performance of models are not consistently good or poor throughout the validation period. Evidence from economics also supports the use of equal weights. In an analysis of five econometric models' forecasts, Pencavel (1971) found no tendency for models that produced the most accurate in one year to do so in the next. Similarly, Batchelor (1990) concluded that "all forecasters are equal" in economics. In a comprehensive review, Cleman (1989) found equal weighting to be accurate for many types of forecasting. Armstrong (2001) suggested using equal weights unless one has strong evidence to support unequal weighting of forecasts.

6.6 Conclusions

ANN models are gaining more and more popularity the hydrology community. ANN models have a good ability of nonlinear approximation. However, for modeling a complicated streamflow process efficiently, an overall ANN may be not enough. Based on the principle of divide-and-conquer (DAC), which deals with a complex problem by dividing it into simple problems whose solutions can be combined to yield a solution to the complex problem, three types of hybrid ANN models, namely, the periodic ANN (PANN) model, the threshold ANN (TANN) and the cluster-based ANN (CANN), are applied to forecasting 1- to 10-day ahead daily discharges of the upper Yellow River at TNH. The model evaluation results indicate that the PANN based on soft seasonal partitions (referred to as soft PANN) performs slightly better than the PANN based on hard seasonal partitions. Comparing with the normal feed-forward multi-layer perceptron (MLP) ANN, soft PANN outperform normal ANN at short

lead times, but slightly worse in long lead times taking the whole year into account. However, comparing the performance of normal MLP-ANN and soft PANN, we find that, except for the summer, soft PANN outperforms normal MLP-ANN in all other three seasons.

In addition, the influence of different data pre-processing procedures, namely, standardization, rescaling and deseasonalization, on the ANN model performance is analysed in the case univariate time series forecasting. It is shown that, for MLP networks with a tan-sigmoid transfer function, when data length is long enough, standardizing the data by subtracting the mean value and dividing by the standard deviation is better than with rescaling the data to a small interval of [-1, 1]. For seasonal data such as streamflow series, deseasonalization accomplished by subtracting the seasonal (e.g., daily or monthly) means and dividing by seasonal standard deviations, is better than normal standardization.

The comparison of the ARMA model, ARFIMA model, PAR model, ARFIMA model, normal MLP-ANN model and soft PANN model shows that the overall performance of PANN is the best for one-day ahead forecasts, whereas the ARFIMA model outperforms all the other models/methods for long lead-time forecasts. Despite of the existence of nonlinearity in daily streamflow series and despite of the capacity of ANN-type models to model nonlinearity, ANNs are not always superior to ARMA-type model (either ARMA model or ARFIMA model). This is in agreement with some other studies. For instance, Goswami et al. (2002) found that simpler models for continuous river-flow simulation can surpass their complex counterparts in performance. For some large catchments with strong seasonality, the Linear Perturbation Model outperforms both the SMAR (Soil Moisture Accounting and Routing) model and the ANN model. In comparing the forecasting results of the naïve AR model with the MLP ANN model, O'Connor *et al.* (2004) found that the performances were virtually the same in the calibration period for both model forms, for lead times up to 6 days. In another comparison of linear and nonlinear techniques in river flow forecasting, Yawson et al. (2005) found that, where limited data are available, the use of more complex models in river flow forecasting might not be very advantageous to the modeller.

To combine the strength of four dynamically different models (the ARMA model, PAR model, ARFIMA model, normal MLP-ANN model and soft PANN model), four forecast combination methods are applied in this study: simple average method (SAM), rollingly-updated weighted average method, semi-fixed weighted average method, and modular semi-fixed weighted average method. The results show that SAM can improve the accuracy of up to 4- to 5-day ahead forecasts compared with any individual models, and it generally performs the best among the four competitive combination methods, which confirms the robustness of the SAM method. Because of its simplicity and robustness, SAM is recommended for improving streamflow forecast accuracy when no individual model performs consistently more accurately or more poorer than the others among the members to be combined.

Chapter 7 Generalization of ANN Models for Streamflow Forecasting

7.1 Introduction to the Generalization of ANN model

Artificial neural networks (ANNs) are prone to either underfitting or overfitting (Sarle, 2002). A network that is not sufficiently complex can fail to detect fully the signal in a complicated data set, leading to underfitting. A network that is too complex may fit the noise, not just the signal, leading to overfitting, which may result in predictions far beyond the range of the training data. Therefore, one crucial issue in constructing a neural network is generalization, namely, the capacity of an ANN to make predictions for cases that are unseen in the training set.

The simplest way of achieving generalization is of course using enough training data. Amari et al. (1997) showed that, when the ratio (R) of the sample size to the number of weights in the network is larger than 30, no overfitting is observed. However, Wang et al. (2005b) showed that slight overfitting might still be observed even when R is as high as 50, and the shortage of data is common in the real world. There are basically two ways to deal with overfitting when training sample size is not large enough: reducing the size (i.e., number of neurons) of the network or reducing the size of each weight connecting neurons. Methods for reducing the size of networks include constructive learning algorithm (see e.g., Kwok and Yeung, 1997) which starts with a small network and then adds additional hidden units and weights until a satisfactory solution is found, pruning (e.g., Reed, 1993; Prechelt, 1997) which searches for a good network in the other direction, and weight sharing (e.g., Nowlan and Hinton, 1992) in which two or more units are forced to have the same weight values so that the true number of parameters in the model are reduced. Methods for reducing the size of each weight are early stopping (e.g., Prechelt, 1998), which stops training at the appropriate point to avoid the network learning the high frequency noise, and the regularization (or weight decay) technique (e.g., Buntine and Weigend, 1991; Mackay, 1991; Weigend et al., 1991; Krogh and Hertz, 1992; Neal, 1996), which encourages smoother network mappings by adding a penalty term to the objective function to be minimized. The two methods may be combined to hopefully achieve better generalization. For instance, Tresp et al. (1997) propose Early Brain Damage method, which extends a pruning technique - Optimal Brain Damage (Le Cun et la., 1990) so that it can be used in connection with early stopping. Besides, the addition of noise (e.g., Holmström and Koistinen, 1992), and combining networks may help to reduce generalization errors.

In this chapter, after briefly describe the two techniques, we will first compare the two generalization techniques, i.e., Bayessian regularization or cross-validated early stopping in making one-step ahead forecasts for univariate time series, including several synthetic

nonlinear time series and several real-world observed time series; then we make multi-step forecasts for daily streamflows of the Yellow River at TNH in the case of involving several exogenous variables (upstream discharges, precipitations and temperature).

7.2 Methods of Generalization

7.2.1 Cross-validated Early stopping

Cross-validated early stopping (referred to as CVES hereafter) is the most popular method to achieve generalization through using cross-validation data. Cross-validation method has a long history of being used as a standard tool for selecting models in statistics and improving the reliability of parameter estimation for statistical models (see, e.g., Toussaint, 1974; Golub et al., 1979). In CVES, the available data are usually split into two subsets: training and cross-validation (CV) sets. The training set is used for updating the network weights and biases. The CV set is used to monitor the error variation during the training process. The validation error will normally decrease during the initial phase of training, as does the training set error. If the network begins to overfit the data, the error on the validation set will typically begin to rise. When the validation error increases for a specified number of iterations, the training is stopped. More generally, we can make *k*-fold CV, in which the data are divided into *k* subsets of (approximately) equal size. The net is trained *k* times, each time leaving out one of the subsets from training, but using only the omitted subset to compute whatever error criterion interests you. If *k* equals the sample size, this is called leave-one-out CV.

One important issue with regard to CVES method is in what ratio to split the training samples into training set and CV set. Amari et al. (1997) suggested that the average generalization error would be minimized asymptotically when the ratio of CV set size to training set size is:

$$r_{opt} = \frac{\sqrt{2k-1}-1}{2(k-1)} \tag{7.1}$$

where *k* is the total number of weights in the ANN. When *k* is large enough, $r_{opt} = 1/\sqrt{2k}$.

7.2.2 Bayesian Regularization

Large weights can hurt generalization of networks in two different ways (Sarle, 2002). Excessively large weights leading to hidden units can cause the output function to be too rough, possibly with near discontinuities. Excessively large weights leading to output units can cause wild outputs far beyond the range of the data if the output activation function is not bounded to the same range as the data. To put it another way, large weights can cause excessive variance of the output (Geman et al., 1992). A traditional way of dealing with the negative effect of large weights is regularization. Regularization is a well-known concept in statistical parameter estimation (e.g., Vapnik, 1982). The idea of regularization is to make the network response smoother through modification in the objective function by adding a penalty term that consists of the squares of all network coefficients. This additional term

favours small values of weights and decreases the tendency of a model to overfit noise in the training data.

Let D represent the data set, $\boldsymbol{w}$ represent the vector of network parameters. The typical objective for training feed-forward neural networks is to minimize the sum of squared errors E_D. With regularization, the objective becomes to minimize

$$F(w) = \beta E_D + \alpha E_w \tag{7.2}$$

where E_w is the sum of squares of the network parameters, and α and β are objective function parameters which dictate the emphasis of the training. If $\alpha << \beta$, then the training algorithm will drive the errors small. But if $\alpha >> \beta$, then training will emphasize weight size reduction at the expense of network errors, thus producing a smoother network response. The task of regularization is to find optimal values of α and β so that the data fitted but not overfitted.

Mackay (1991) proposed a technique, called Bayesian regularization, which automatically sets the optimal performance function to achieve the best generalization based on Bayesian inference techniques. In the Bayesian framework the weights $\boldsymbol{w}$ of the network are considered random variables. After data D is taken, the density function for $\boldsymbol{w}$ can be updated according to Bayes' rule

$$P(w \mid D, \alpha, \beta, M) = \frac{P(D \mid w, \beta, M) P(w \mid \alpha, M)}{P(D \mid \alpha, \beta, M)} \tag{7.3}$$

where M is the particular network used; $P(w|\alpha, M)$ is the prior density, which represents our knowledge of the weights before any data is collected; $P(\mathrm{D}|w, \beta, M)$ is the likelihood function, which is the probability of the data occurring given the weights $\boldsymbol{w}$. $P(D|\alpha, \beta, M)$ is a normalization factor, which guarantees that the total probability is 1.

Assuming that the noise and the prior distribution for the weights are both Gaussian, the probability densities can be written as

$$P(D \mid w, \beta, M) = \frac{1}{Z_D(\beta)} \exp(-\beta E_D)$$

and

$$P(w \mid \alpha, M) = \frac{1}{Z_w(\alpha)} \exp(-\alpha E_w) \tag{7.4}$$

where $Z_D(\beta) = (\pi/\beta)^{\mathrm{N}/2}$, $Z_D(\alpha) = (\pi/\alpha)^{k/2}$, and N is the size of training sample, k is the total number of parameters in the network. Substitute (7.4) into (7.3), we get the posterior probability of $\boldsymbol{w}$:

$$\begin{aligned} P(w \mid D, \alpha, \beta, M) &= \frac{\dfrac{1}{Z_w(\alpha)} \dfrac{1}{Z_D(\beta)} \exp(-\beta E_D - \alpha E_w)}{P(D \mid \alpha, \beta, M)} \\ &= \frac{1}{Z_F(\alpha, \beta)} \exp(-F(w)) \end{aligned} \tag{7.5}$$

In the Bayesian framework, the optimal weights should maximize the posterior probability (7.5), which is equivalent to minimizing the regularized objective function given in (7.2). With (7.3), (7.4) and (7.5), $P(D|\alpha, \beta, M)$ can be expressed as

$$P(D \mid \alpha, \beta, M) = \frac{Z_F(\alpha, \beta)}{Z_D(\beta) Z_w(\alpha)} \tag{7.6}$$

The constants $Z_D(\beta)$ and $Z_w(\alpha)$ are known from (7.4). What needs to be estimated is $Z_F(\alpha, \beta)$. Since the objective function has the shape of a quadratic shape in a small area surrounding a minimum point with the parameter vector $\boldsymbol{w}_{\mathrm{MP}}$, we can expand $F(\boldsymbol{w})$ around the minimum point of the posterior density with Taylor series expansion. This yields

$$F(w) = F(w_{\mathrm{MP}}) + \frac{1}{2}(w - w_{\mathrm{MP}})^T H (w - w_{\mathrm{MP}}) . \tag{7.7}$$

where $\mathrm{H} = \beta\nabla^2 E_D + \alpha\nabla^2 E_{\mathrm{w}}$ is the Hessian matrix of the objective function. Then $Z_F(\alpha, \beta)$ is the Gaussian integral:

$$\begin{aligned} Z_F(\alpha, \beta) &= \int d^k w \exp(-F(w, \alpha, \beta)) \\ &\approx (2\pi)^{k/2} \exp(-F(w)) \det{}^{-1/2} \mathrm{H} \end{aligned} . \tag{7.8}$$

To optimize the value of α and β, we apply Bayes' rule

$$P(\alpha, \beta \mid D, M) = \frac{P(D \mid \alpha, \beta, M) P(\alpha, \beta \mid M)}{P(D \mid M)} \tag{7.9}$$

Assuming a uniform prior density $P(\alpha, \beta|M)$ for the regularization parameter α and β, then maximizing the posterior of α and β is achieved by maximizing the likelihood function $P(D|\alpha, \beta, M)$, whose logarithm can be written as

$$P(D \mid \alpha, \beta, M) = -\alpha E_w^{\mathrm{MP}} - \beta E_D^{\mathrm{MP}} - \frac{1}{2}\log\det H - \log Z_w(\alpha) - \log Z_D(\beta) + \frac{k}{2}\log 2\pi \tag{7.10}$$

We can solve for the optimal value of β and α at the minimum point by taking the derivatives of (7.10) with respect to α and β, and set them equal to zero. This yields

$$\alpha_{MP} = \frac{\gamma}{2E_w(w_{MP})} \text{ and } \beta_{MP} = \frac{N - \gamma}{2E_D(w_{MP})} ,$$

where $\gamma = k - 2\alpha_{\mathrm{MP}}\mathrm{tr}(\mathbf{H}^{-1})$ is called the effective number of parameters.

The Bayesian optimization of the regularization parameters requires the computation of the Hessian matrix of at the minimum point $\boldsymbol{w}_{\mathrm{MP}}$. Foresee and Hagan (1997) propose using the Gauss-Newton approximation to Hessian matrix, which is readily available if the Levenberg-Marquardt optimization algorithm is used to locate the minimum point. The additional

computation required of the regularization is thus minimal. In this study, the function *trainbr* built in the Neural Network Toolbox in Matlab (version 7) is used to implement the training of ANN models with the BR technique.

7.3 Data Used

7.3.1 Univariate Time Series Data

Twelve univariate time series data are used in this study, including three synthetic series and nine observed real-world time series.

7.3.1.1 *Synthetic time series*

Two classical chaotic time series and one synthetic time series generated with a ANN model are used in the study.

(1) Henon map chaotic series (Henon, 1976) of the form

$$\begin{cases} x_{n+1} = 1 - ax_n^2 + by_n \\ y_{n+1} = x_n \end{cases}.$$

With a = 1.4 and b= 0.3, for initials such as x = 0, y = 0.9, Henon map series has one strange chaotic attractor.

(2) The discretized Mackey-Glass (Mackey and Glass, 1977) chaotic series of the form:

$$x_{n+1} = \frac{2m - b\tau}{2m + b\tau} x_n + \frac{a\tau}{2m + b\tau} \left(\frac{x_{n-m}}{1 + x_{n-m}^c} + \frac{x_{n-m+1}}{1 + x_{n-m+1}^c} \right).$$

With a = 0.2, b = 0.1 and c = 10, the time series has chaotic attractors of different dimension for $\tau > 16.8$. We choose m = 30, $\tau = 30$.

(3) A time series generated with an ANN model with a structure 5-3-1, namely, five inputs, one hidden layer with three hidden nodes, and one output. It has a standard Gaussian noise term. The size of the ANN series is 3000 points.

Both the synthetic chaotic time series have a size of 10,000 points. Gaussian noises are added to the two chaotic series. The mean of the Gaussian noise is zero, and the standard deviation is 2% that of the chaotic series.

7.3.1.2 *Observed time series*

Six observed streamflow time series are used, including: (1) and (2) daily and monthly streamflow series of the Yellow River at Tangnaihai (TNH), China, from January 1, 1956 to December 31, 2000; (3) and (4) daily and monthly streamflow series of the Rhine River at Lobith, the Netherlands, from January 1, 1901 to December 31, 1996; (5) and (6) daily and

monthly streamflow series of the Danube River at Achleiten, Austria, from January 1, 1901 to December 31, 1990.

In addition, three other observed geophysical time series are used, including: (1) The monthly sunspot number series, starting from January, 1749 and ending up in December, 2004, available on the Belgium Solar Influences Data analysis Center website http://sidc.oma.be; (2) The yearly sunspot number series, starting from 1700, and ending up in 2004, also available on the website of Belgium Solar Influences Data analysis Center; (3) Monthly Southern Oscillation index (SOI) series, from January, 1933 to December, 2004, available on the NOAA website at http://www.cpc.ncep.noaa.gov/data/indices/;

7.3.2 Multivariate Hydrological Time Series Data

So far in this study, we focus on investigating the forecasting of streamflow processes based on univariate streamflow time series. To compare the performances of different generalization techniques in the cases where multivariate time series are involved, besides using the average discharges at TNH in previous several days, we also use several streamflow series and meteorological series that are recorded in the area above TNH gauging station as explanatory variables to forecast multi-day ahead average discharges at TNH. The used hydrological time series include: observed daily average discharges at Maqu (MQ) and at Jimai (JM), daily precipitation data at MQ and at Dari (DR, the same place as the gauging station JM), and daily average temperature data at DR. The locations of all the gauging stations are shown in Figure 7.1. All the data used here start from January 1, 1960, and end on December 31, 2000.

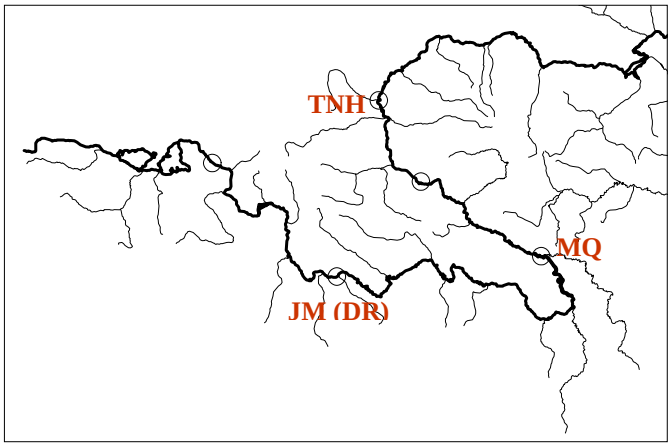


Figure 7.1 Locations of the hydrological gauge stations and meteorological gauge stations

7.4 On the Generalization of ANNs for Univariate Time Series Prediction

7.4.1 Building ANN Models for Univariate Time Series

When building a neural network model, a number of decisions must be made, including the neural network type, network structure, methods of pre- and post-processing of input/output data, training algorithm and training stop criteria. The feed-forward multi-layer perceptron (MLP) ANN is adopted in the present study. The network structure includes the number of

input, the number of hidden layers, the number of hidden nodes in each layer, and the number of outputs. These structure parameters determine the total number of weights in the network. For a univariate time series, the number of inputs is virtually the embedding dimension m. With the embedding dimension m, we convert the scalar time series $\{x_i\}$, i=1, …, N, to state vectors $\{X_t\}$, where $X_t = [x_t, x_{t-1}, ..., x_{t-m+1}]$, and m is chosen depending on the time series. In the present study, the structure of ANNs for the synthetic time series and streamflow series are chosen according to previous studies in the literature or according to the trial and error procedure. De Oliveira et al. (2000) suggest using m-2m-m-1 structure to model chaotic series, i.e., m inputs, 2 hidden layers with 2m and m hidden nodes for each layer, and one output. Follow their suggestion, we use 6:12:6:1 for Henon series as well as the discretized Mackey-Glass series. With trial and error procedure, the chosen structure for all three monthly flow series is 4-3-1. The number of inputs (i.e., values of m) of ANN models fitted to daily flow of the Danube and the Rhine are determined according to their partial autocorrelation functions, and the chosen structure is 23-12-1 for daily flow of Danube, and 16-8-1 for daily flow of Rhine. The number of inputs of the ANN model fitted to daily flow of the Yellow River is 5,which is determined by the false neighbor method (See Section 6.2.1), and the ANN structure is 5-3-1. Although it has been shown (see Section 6.2) that deseasonalization is the best pre-processing procedure for streamflow series, but for simplicity, all the data are standardized by subtracting the mean of the series and dividing by the standard deviation to make the data have a mean of 0 and variance 1.

The ANNs are constructed with Matlab Neural network toolbox. In all ANNs, *tansig* transfer function is used in the hidden layer. The training algorithm for the MLP network with CVES is the backpropagation algorithm that updates weight values according to gradient descent with momentum, whereas the Bayesian regularization takes place within the Levenberg-Marquardt algorithm (Foresee and Hagan, 1997). To avoid the problem of the sensitivity of networks to initial weights, simple ensemble technique is applied. That is, for each case, we fit 10 networks, and then choose five best ones, which have best training performance, and take the average of the outputs of the five selected networks to be the final output.

7.4.2 How Many Data are Demanded to Avoid Overfitting?

Amari et al. show that (1997), when the ratio (referred to as R hereafter) of the training sample size to the number of weights is larger than 30, no overtraining is observed. This view is accepted by many researchers as a guideline for training ANNs (e.g., Mair and Dandy, 2000; Sarle, 2002). But, is there such a clear cut-off value of R? Here, we test the impact of R on root mean squared error (RMSE) of training data and testing data with three synthetic series. To avoid the possible impact of nonstationarity, real world data are not applied here.

We make experiments with different values of R range from 5 to 50. We use the last 1000 points of each synthetic series as the test data, while the training data vary according to the value of R. Networks are trained with Levenberg-Marquardt backpropagation algorithm and the training epoch is 1000. The variation in RMSE of training data and test data with different values of R for the three synthetic series are plotted in Figure 7.2.

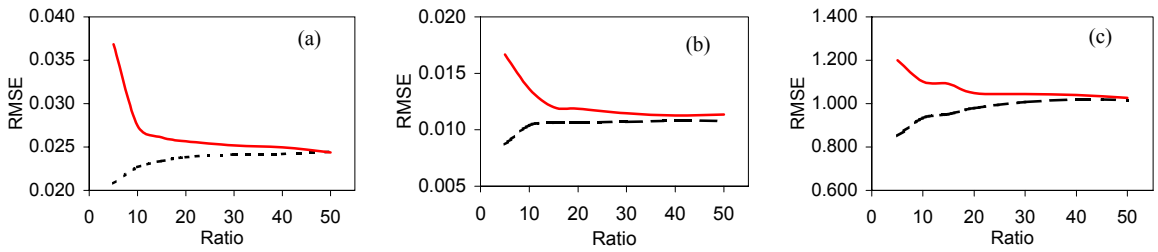


Figure 7.2 RMSE of training data and test data with different values of *R* for (a) Henon series; (b) Mackey-Glass series and (c) simulated ANN series.

From Figure 7.2, we see that with the increase of *R*, the training error grows, whereas, the testing error decreases with the increase of *R*, which indicates that the intensity of overfitting decrease. However, there is no clear cut-off value of *R*, above which overfitting vanishes. When *R*=30, suggested by Amari et al. (1997), overfitting is still observed for all three fitted networks. When the ratio *R* is as high as 50, the overfitting basically disappears. However, the test error is still slightly higher than, albeit very close to, the training error, for the Mackay-Glass series and simulated ANN series, which indicates the existence of slight overfitting.

7.4.3 How to Split the Training Samples in Cross-Validated Early Stopping?

Now we investigate whether there is an optimal value r_{opt} of the CV to training ratio *r*, with which the generalization error is minimized with limited training samples. We calculate the training error and testing error with different values of *r*, ranging from 0.02 to 0.3, for different size of training data with different values of training sample size to number of weight ratio *R*. The results for *R* = 10 are plotted in Figure 7.3.

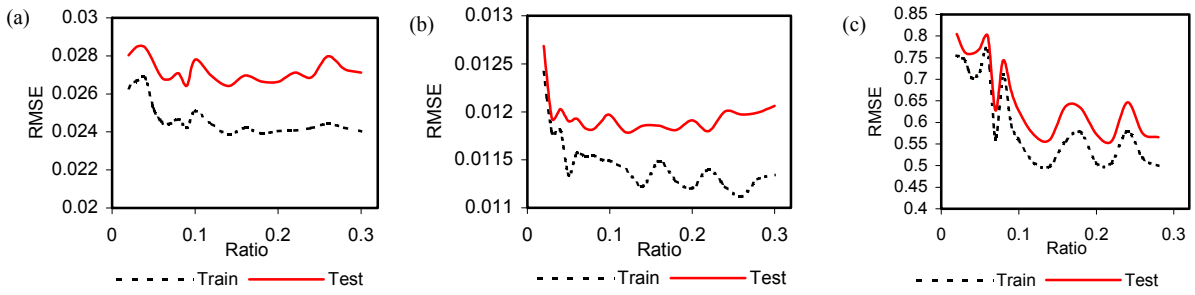


Figure 7.3 RMSE of training set and test data with different CV to training ratio *r* for (a) Henon series; (b) Mackey-Glass series and (c) simulated ANN series when *R* = 10.

The networks for Henon series and Mackey-Glass series have the structure of 6-12-6-1 structure, with total number of weights *k* = 169. The network for the synthetic ANN data set have a 5-3-1 structure, *k* = 22. Therefore, according to the optimal ratio proposed by Amari et al. (1997), as shown in Equation (1), for the former two networks, $r_{opt} \cong 0.077$; for the later one, $r_{opt} \cong 0.132$. However, there is no clear evidence of the existence of such optimal ratios from the visual inspection of Figure 2 as well as the results for other experimental results when *R* is 5, 15 and 20.

Sarle (2002) comments that the results of Amari et al. (1997) contain serious errors that completely invalidate. From the experiments in this study, it seems that this comment is substantiated. In practice, many research use a large part, such as 1/3 (e.g., Prechelt, 1998), of total available data as CV set. However, according to this experiment, the ratio of CV set to training set seems to be not very important for early stopping. 10% could be a practical choice when the data series is stationary.

7.4.4 Which Technique Works Better, Bayesian Regularization Or Cross-Validated Early Stopping?

To compare the performances of the Bayesian regularization or cross-validated early stopping, we make one-step ahead forecasts for the seven time series described above with the two techniques with varying size of training data, which is reflected by varying values of R (the ratio of the training data size to the number of weights in a network). As a benchmark, the no-stop training (NST) is also applied, in which the training is stopped after 1000 epochs. The performances of ANNs with these three approaches are measured with *RMSE*. The results are presented in Table 7.1. The comparison of the performances shows that:

(1) The BR method outperforms CVES in most cases, except for the cases of Mackay-Glass series and monthly sunspot series.

(2) When training sample size is small ($R < 20$), the BR method and CVES method generally may help to improve the generalization of ANN models. But these techniques do not always work. For several cases (e.g., the case of daily streamflow of the Danube) they fail even when $R \leq 10$, especially with the CVES method.

(3) The NST outperforms the CVES in most cases when $R \geq 20$. That means, albeit overfitting still may be observed when $R \geq 20$ or even when R is as high as 50, cross-validated early stopping does not improve the generalization when $R \geq 20$ for most cases. In contrast, the BR method still outperforms the NST in the most cases even when $R \geq 30$.

(4) An advantage of CVES is its fastness compared with the BR method and the NST, especially when the network is complicated. In contrast, the time-costness is a major disadvantage with the BR method, especially when the training data size is big and the network is complicated.

(5) The performance of CVES is highly variable. Namely, when we run the training process for a network with different initial weights, the networks we obtain in these different runs may give significant forecasting performance, which indicating that CVES technique is not robust. This is because it often ends up the training process too early due to lots of local minima of error function for the CV data set. Therefore, care must be taken when using CVES in making streamflow forecasts despite of its fastness, unless the speed is of the most importance. It's better to check the error function surface of CV data to see if there are local minima that may stop training too early before we use the CVES method.

(6) The performance of Bayesian generalization is highly stable, especially compared with CVES. In some cases (e.g., the simulated ANN series), the 10 runs with different initial weights give almost the same result.

Table 7.1 Compare the *RMSE*s of one-step ahead forecasts with Bayesian regularization (BR), cross-validated early stopping (CVES) and no-stop training (NST)

Series	*R*	BR	CVES	NST	Series	*R*	BR	CVES	NST
Henon	5	0.02778	0.02790	0.03684	Rhine	4.9	227	291	230
	10	0.02642	0.02655	0.02745	daily	10	185	277	220
	20	0.02508	0.02575	0.02565		15	188	508	218
	30	0.02491	0.02545	0.02520		25.1	168	276	217
	40	0.02468	0.02531	0.02497		37.7	185	280	211
	50	0.02468	0.02482	0.02439		50.3	186	281	217
Mackay-	5	0.01300	0.01253	0.01666	Danube	6.3	380	395	387
Glass	10	0.01198	0.01186	0.01362	monthly	12.6	377	396	376
	20	0.01144	0.01142	0.01186		18.9	369	386	369
	30	0.01140	0.01131	0.01146		25.3	369	384	364
	40	0.01130	0.01130	0.01128		31.6	367	388	361
	50	0.01133	0.01129	0.01134		37.9	367	380	362
ANN	5	1.080	1.098	1.199	Danube	4.8	218	174	169
	10	1.055	1.072	1.102	daily	9.6	184	178	162
	20	1.045	1.055	1.048		14.5	177	175	163
	30	1.038	1.062	1.044		24.2	172	175	164
	40	1.026	1.044	1.040		36.3	163	170	159
	50	1.017	1.046	1.026		48.5	162	171	159
Yellow	3.8	303	271	317	Sunspot	5	20.9	20.9	20.5
monthly	7.6	289	282	297	Yearly	10	18.9	21.4	19.6
	11.4	253	271	268		15	18.8	21.0	19.4
	15.2	248	259	274	Sunspot	5	20.9	19.3	19.4
	18.9	245	268	265	monthly	10	19.7	18.1	18.2
	22.7	244	265	257		15	19.5	18.0	17.8
Yellow	16.4	50.6	113.8	48.1		20	18.4	17.6	17.6
daily	33.0	56.4	68.7	47.8	SOI	5	1.327	1.409	1.468
	82.8	45.8	86.2	46.3	monthly	10	1.357	1.559	1.461
	165.8	45.2	68.0	46.5		15	1.315	1.373	1.354
	331.8	45.1	63.6	46.2		20	1.333	1.436	1.448
Rhine	6.3	840	867	886		30	1.313	1.393	1.363
monthly	12.6	839	935	858					
	18.9	848	862	867					
	25.3	850	897	853					
	31.6	850	898	854					
	37.9	855	876	855					

Note: *R* refers to the ratio of the training data size to the number of weights.

7.5 Streamflow Forecasting with the Inclusion of Exogenous Variables

7.5.1 Building ANN Models for Streamflow Forecasting with the Inclusion of Exogenous Variable

As in the case of univariate time series, the MLP ANN is also used for the case of multivariate streamflow time series, and we adopt the one-hidden-layer structure. The major concern is the selection of inputs. According to the result of false-neighborhood method (see Section 6.2.1), we include the average discharges in the past five days as part of the inputs. According to the cross-correlation analysis, the average discharge Q_t at TNH has the highest correlation with one-day past discharge at MQ, three-day past discharge at JM, seven-day past precipitations at both MQ and DR, about 20-day past temperature at DR. Therefore, the following 32 inputs are used in building the ANN for making 1-day ahead forecast Q(t):

- daily discharges $Q_{TNH}(t-1)$, …, $Q_{TNH}(t-4)$ at TNH;
- daily discharges $Q_{MQ}(t-1)$ at MQ;
- daily discharges $Q_{JM}(t-1)$, …, $Q_{JM}(t-2)$ at JM;
- daily precipitation $R_{MQ}(t-1)$, …, $R_{MQ}(t-8)$ at MQ;
- daily precipitation $R_{DR}(t-1)$, …, $R_{DR}(t-8)$ at DR;
- daily temperature $T_{DR}(i_1)$, …, $T_{DR}(i_7)$ at DR, where $T_{DR}(i_1) = [T_{DR}(t-1) + T_{DR}(t-2) + T_{DR}(t-3)]/3$, …, $T_{DR}(i_7) = [T_{DR}(t-19) + T_{DR}(t-20) + T_{DR}(t-21)]/3$.

Note that, first, we include the precipitations in the past 8 days rather than 7 days because of the importance of the precipitation. Second, in fact, if we remove the seasonal means of the discharges and those of the temperatures, the correlation between the daily average discharges and the daily average temperatures is very low and the lag time is very short. The reason that we keep the information of the past 21 days in the inputs is on one hand, the temperature is related to the evaporation and the amount of snowmelt; on the other hand, the temperatures may act as an indicator of seasonal variation.

The same inputs are used to make 2- to 5-day ahead forecasts, i.e., Q(t+1), Q(t+2), Q(t+3), Q(t+4). The data in year 1960-1995 will be used for training the network (but data in 1991 are not used because the discharge data at JM in 1991 are missing), and we make 1- to 5-day ahead forecasts for year 1996 to 2000. To train a network with 32 inputs using 34 years' data is very time consuming if we determine the number of hidden nodes with the trial and error procedure. Ebberhart and Dobbins (1990) suggested starting with hidden nodes equal to half of the input nodes. Following their suggestion, we simply take a half of the number of inputs as the number of hidden nodes. Namely, the structure we adopt for the case of multivariate hydrological time series is 32-16-1.

7.5.2 Comparing Normal MLP-ANN with BR-ANN

As it has been shown that the CVES technique is not as reliable as the NST method and the BR method, we do not consider the CVES technique when making the 1- to 5-day ahead forecasts with multivariate hydrological time series. We use the data from 1960 to 1995 to train the network (data in 1990 are not used, because the streamflow data at JM in 1990 are missing), about 12,740 points in total. Therefore, for a network with a 32-16-1 structure, the ratio *R* of training sample size to the number of weights in the network is about 23.4. The performance of non-stop training and the BR method for making 1- to 5-step ahead daily discharge forecasts in year 1996-2000 are shown in Table 2. Model performance measures used here include RMSE, *CE* and *SACE*.

The result shows that for making 1- to 4-day ahead forecasts, the BR method outperforms the NST method. But the advantage of the BR decreases with the increase of lead times, and it disappears for making 5-day ahead forecasts. The scatter plot of observed discharges versus forecasted values are shown in Figure 7.4, and the hydrographs of observed discharges and one-day ahead and five-day ahead forecasted values are shown in Figure 7.5.

Table 7.2 Compare the performance of non-stop training (NST) and Bayesian regularization (BR) for forecasting 1- to 5-step ahead daily discharges in year 1996-2000.

Lead time	NST			BR		
(day)	RMSE	CE	SACE	RMSE	CE	SACE
1	39.82	0.992	0.985	37.12	0.993	0.987
2	67.34	0.976	0.956	63.43	0.979	0.961
3	85.10	0.962	0.93	83.07	0.964	0.933
4	100.02	0.948	0.903	99.43	0.949	0.905
5	114.27	0.932	0.874	114.65	0.932	0.873

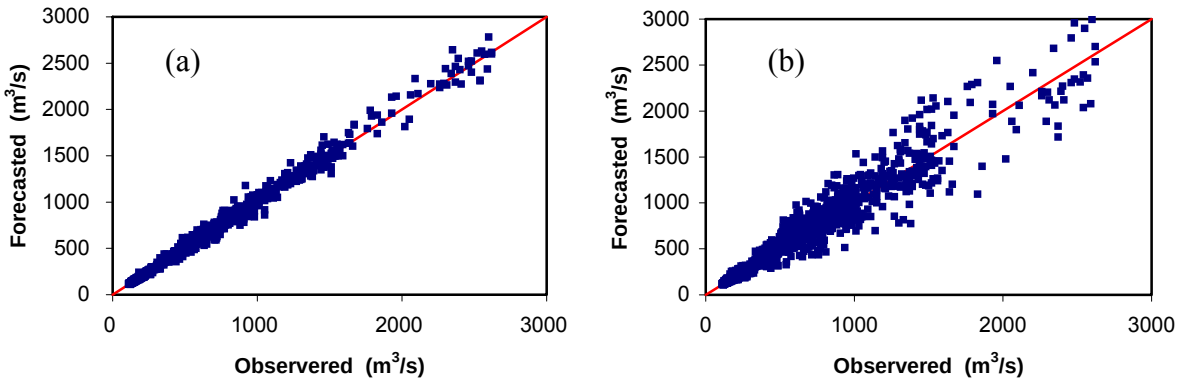


Figure 7.4 Scatter plot of observed discharges versus forecasted values. (a) 1-day ahead forecasts; (b) 5-day ahead forecasts

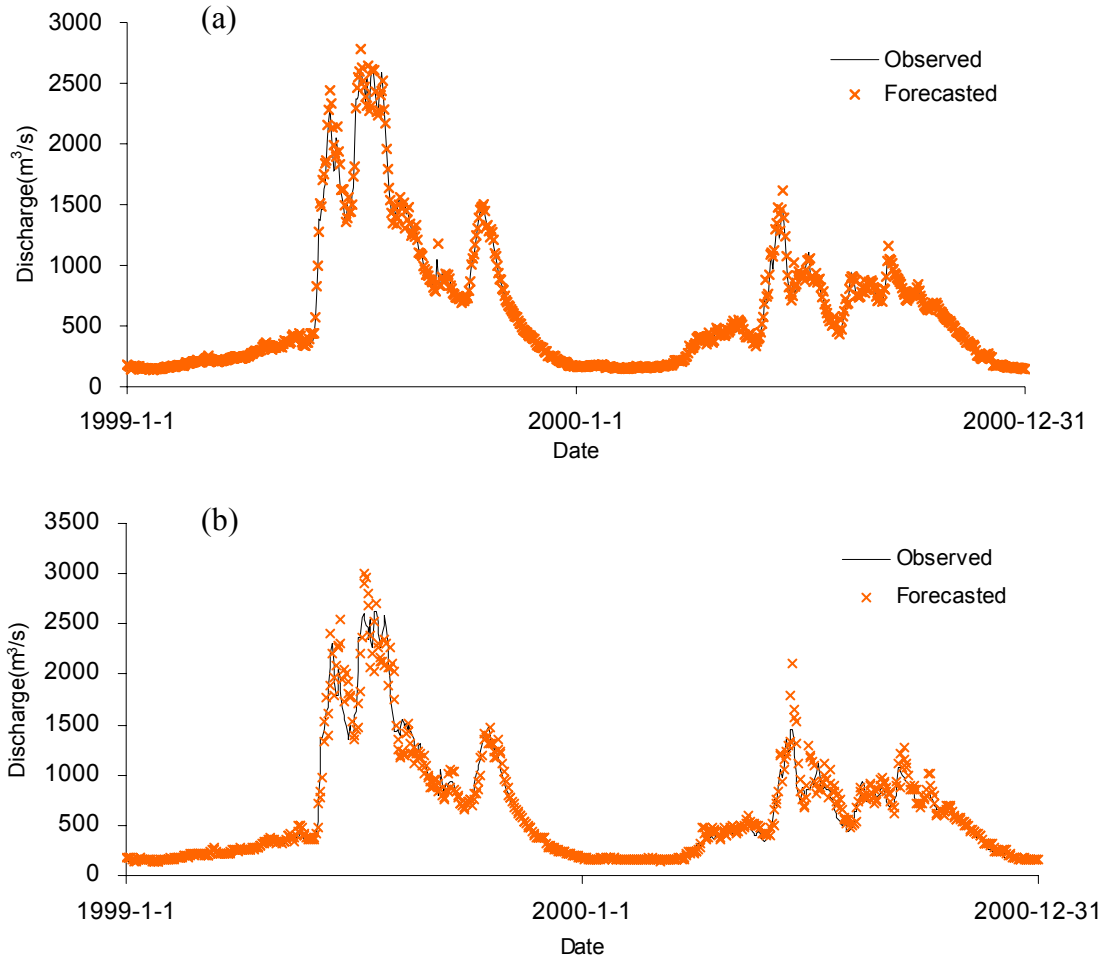


Figure 7.5 Hydrographs of the observed discharges and forecasted values. (a) 1-day ahead forecasts; (b) 5-day ahead forecasts

7.6 Conclusions

Generalization is a crucial issue in constructing a neural network when the size of training data is not big enough. Two commonly used techniques to achieve generalization are the Bayesian regularization (BR) method and the cross-validated early stopping (CVES) method. In this present study, some experiments are made to compare the performances of these two techniques and the no-stop training (NST, in which the training is stopped after 1000 epochs) for making forecasts in the case of univariate time series and multivariate hydrological time series. The results show that, for making one-step ahead forecast, both the BR method and the CVES method generally outperforms the NST method when the ratio R of training sample size to the number of weights in the network is less than 20. But the advantage of the two techniques over the NST method is not guaranteed. In several cases (e.g., the case of daily streamflow of Danube) they fail even when the ratio $R \leq 10$, especially the CVES method. Furthermore, the performance of the CVES method is highly variable because it may stop the training process prematurely due to many local minima.

Comparing the BR method outperforms the CVES method for making one-step ahead forecasts for univariate processes, the BR method outperforms the CVES method in most cases, and it still outperforms the NST method in some cases even when the ratio R is larger than 30. The BR method is further used to make 1- to 5-day ahead streamflow forecasts for the headwater of the Yellow River, and compared with the NST method. The result shows that for making 1- to 4-day ahead forecasts, BR outperforms NST. But the advantage of the BR method decreases with the increase of lead times, and the advantage disappears for making 5-day ahead forecasts. The major disadvantage of the BR method is its time-costliness. Therefore, when the speed of training is not a major concern, the BR method is recommended for making streamflow forecasts.

Chapter 8 Conclusions and Recommendations

Streamflow forecasting is of great importance to water resources management and flood defence. Significant progresses had been seen in the last several decades on this issue. The methods for forecasting streamflows may fall into two general classes: process-driven methods and data-driven methods. Good streamflow forecasting models are built on the basis of the good understanding of the streamflow process. Equivalent to the classification of forecasting methods, methods for understanding streamflow processes may also be broken into two categories: physically-based methods and mathematically-based methods. With the former group of methods, we go directly into the underlying mechanisms of the hydrological processes, then try to describe the mechanisms with mathematical formula, whereas with the latter group of methods, we give mathematical descriptions of the hydrological processes first, then try to find physical explanations behind the mathematical descriptions.

This thesis focuses on using mathematically-based methods to analyze stochasticity and nonlinearity of streamflow processes based on univariate historic streamflow records, and presents data-driven models that are also mainly based on univariate streamflow time series for forecasting the streamflows of the upper Yellow River in northern China.

In this study, six streamflow processes of five rivers in different geological regions (the Yellow River in China, the Rhine and the Danube in Europe, the Umpqua and Ocmulgee in the United States) are investigated for stochasticity and nonlinearity at several characteristic timescales (i.e., one day, one month, 1/3 month, and one year). Several important aspects of the stochasticity are examined in this study, including trend, seasonality, stationarity, and long-memory. As for the study of nonlinearity, besides the general nonlinearity, two special types of nonlinearity are examined, i.e., conditional heteroskedasticity and chaos. To do so, some statistical test methods, which originate from the econometrics but have not been used in testing hydrological time series, are introduced and successfully applied to the streamflow time series analysis.

Data-driven models based on the univariate streamflow time series are fitted to the daily streamflows and monthly streamflows of the upper Yellow River at Tangnaihai (TNH) for making 1 to 10 day ahead, and one-month ahead forecasts. Due to the strong persistence of the daily streamflow series of the daily streamflow at TNH, in terms of the measure of seasonally-adjusted coefficient of efficiency (*SACE*), a satisfying accuracy ($SACE > 0.8$) can be achieved for five-day ahead forecasts with a linear autoregressive moving average (ARMA) model, a fractionally integrated ARMA model (i.e., ARFIMA model), or an artificial neural network (ANN) model. Taking an individual model into account, periodic ANN (PANN) model performs best for one-day ahead forecasts. The forecasts for short lead times can be

improved by combining the forecasts of several individual models with a simple average combination. But for the forecasts of long lead times (> 4 days), ARFIMA model performs best. With the inclusion of several explanatory variables (upstream discharges, precipitations and temperature), a very satisfying accuracy ($SACE > 0.9$) can be achieved for four-day ahead forecasts with the ANN models.

8.1 On Stochasticity

In the context of global warming, how hydrological processes are impacted is an issue of wide concern. The analyses of two streamflow processes of the Yellow River show that there is no obvious trend in the average annual flow process of the upper reach at TNH from 1956 to 2000, whereas the streamflow process recorded at Tongguan in the middle reaches exhibits significant declining trend. No significant decline is found in the precipitation processes (Fu et al., 2004), on the other hand, it is found that the lower the reaches of the Yellow River, the more significant the downward trend. This indicates that the impact of climate warming on the streamflow processes of the Yellow River are far less significant than anthropogenic influences.

Stationarity is required for the construction of many types of models and for the application of many data analysis techniques. Many methods are available to check whether the data of interest are stationary. In the present study, ADF unit root test (Dickey and Fuller, 1979; Said and Dickey, 1984) and KPSS test (Kwiatkowski et al., 1992), which originate from the econometrics, are introduced to test for the nonstationarity in hydrological time series. It is found that the smaller the timescale of the streamflow process is, the more likely it tends to be nonstationary.

Seasonality is a common feature in hydrological time series. Before fitting a time series model, it is popular to deseasonalize the hydrological time series by subtracting the seasonal means and dividing by seasonal standard deviations. But it is shown in the study that seasonal variations still exist more or less in the autocorrelation structures of all the deseasonalized daily, 1/3-monthly and monthly streamflow processes. This indicates that, the deseasonalization procedure can remove the seasonality in the mean and variance, but not the seasonality in the autocorrelation structure.

Many studies have shown that, many hydrological processes, especially streamflow processes, have long-memory property. This is also confirmed by the present study. Furthermore, the investigation of the long-memory phenomenon in streamflow processes at different timescales shows that, with the increase of timescale, the intensity of long-memory decreases. Generally speaking, according to the test results, all daily flow series exhibit strong long-memory; 1/3-monthly flow series may be considered as weak long-memory processes; monthly series may be considered as short memory processes or at most processes of very weak long-memory.

8.2 On Nonlinearity

It is well accepted that watershed systems are nonlinear. Correspondingly, the output of watershed systems, streamflow processes, may also exhibit nonlinearity. But there is no well-accepted methods to quantify the degree of nonlinearity. What is less-accepted or even controversial is the sources or the nature of the nonlineairty, such as whether or not hydrological processes are deterministic chaotic processes.

In this study, the BDS test method (Brock et al., 1996) is introduced to test for the existence of nonlinearity in general in streamflow processes. It is found that the shorter the timescale, the stronger the nonlinearity. All annual series are linear, whereas all daily streamflow processes are strongly nonlinear. Although after removing seasonal effects some 1/3-monthly and monthly streamflow series are linear according to BDS test results, we cannot conclude that there is no nonlinearity present in these 1/3-monthly and monthly streamflow series, because the power analysis of BDS test shows that BDS test is not powerful enough for detecting weak nonlinearity.

Besides those well-recognized physical sources, such as the mechanisms involved in the rainfall–runoff transformation, some other sources can be identified as viewed from the streamflow time series itself. Asymmetric seasonality in the mean and variance of raw (or log-transformed) streamflow processes and the seasonality in the variance of the pre-whitened streamflow processes (i.e., residual series) play a role in the exhibition of nonlinearity. The conditional heteroskedasticity is an important source of the nonlinearity of streamflow processes, so is long-memory if it is also viewed as a type of nonlinearity. The degree of nonstationarity has a significant impact on the test for nonlinearity. Nonstationarity or the stationarity of low significance level may give rise to positive results (i.e., the existence of nonlinearity) in nonlinearity test.

The nonlinear mechanism autoregressive conditional heteroskedasticity (ARCH) in hydrological processes has not received much attention by the hydrology community so far. The existence of ARCH effect is detected in the residual series from linear models fitted to the daily and monthly streamflow processes of the upper Yellow River. It is shown that the ARCH effect is fully caused by seasonal variation in the variance for monthly flows, but seasonal variation in variance only partly explains the ARCH effect for daily streamflow. To capture the ARCH effect in the daily streamflow processes so as to improve the estimate of forecast uncertainty, the ARMA-GARCH error model with seasonal standard deviations is proposed, in which an ARMA model is used to model the mean behaviour and a GARCH model to model the ARCH effect in the residuals from the ARMA model.

Whether or not hydrological processes are deterministic chaotic processes is a widely concerned and quite controversial issue in the last decade. No finite correlation dimension is found for all the streamflow series in the present study with correlation exponent method. Because the existence of finite correlation dimension is crucial for verifying the existence of chaos, therefore, while nonlinear behaviour seemed to be present with different intensity at

various timescales, the dynamics would not seem to be associable to the presence of low dimensional chaos. On the other hand, even if we found the evidence of the existence of chaos in a time series, it does not necessarily mean determinism. Experiments with three well-known chaotic systems (i.e., Henon map, Ikeda map, discretized Mackey-Glass flow) show that chaos could be stochastic. A chaotic system with stochastic components behaves similar to a noise-free system when the stochastic disturbances are not strong.

8.3 On Forecasting

When forecasting streamflow processes, both the forecasting model (or method) and the model performance measure are needed. Coefficient of efficiency (*CE*) (Nash and Sutcliffe, 1974) is a popular measure for evaluating the performances of hydrological models. However this measure could be misleading about the model performance when being applied to seasonal processes. Therefore, a seasonally-adjusted *CE* (*SACE*) is proposed to measure how good forecasts from a model are better than seasonal mean values, instead of the overall mean value as with *CE*. The measure *SACE* is more suitable than *CE* for evaluating seasonal processes. For non-seasonal stationary processes, *SACE* is reduced to *CE*.

Two groups of data-driven models are used in the study: (i) ARMA-type models, including the ARMA(20,1) model, the ARFIMA(7,d,0) model, and the periodic AR (PAR) model; and (ii) ANN-type models, including the normal multi-layer perceptron (MLP) ANN model, the cluster-based ANN model, the threshold-based ANN model, the period-based ANN model (or, periodic ANN model, PANN) and its variations (i.e., soft PANN and hard PANN). Among these models, PAR and PANN are proposed for modeling daily streamflow processes in this study. They are fundamentally a group of AR models or ANN models. Each AR model or ANN model are fitted to a specific season partition, so that the seasonality in daily streamflow processes is better captured. In addition, to combine the strength of these dynamically different models, four forecast combination methods are adopted in this study: simple average method (SAM), rollingly-updated weighted average method (RWA), semi-fixed weighted average method (SFWA), and modular semi-fixed weighted average method (MSFWA).

The comparison of forecast performances of various models and methods shows that, despite of the limitation of univariate streamflow time series, in terms of the *SACE* measure, satisfactory forecasts can be made for lead times of up to 5 days (*SACE* > 0.8). The overall performance of PANN that is based on soft seasonal partitions performs the best for one-day ahead forecasts. SAM generally performs best among the four competitive combination methods, and it outperforms all the five individual models for forecasts of up to 4 days ahead (except for the PANN model for one-day ahead forecasts). The ARFIMA model performs the best for long lead times (longer than four days).

Generalization is a crucial issue in constructing a neural network when the size of training data is not big enough. Two commonly used techniques to achieve generalization are the Bayesian regularization (BR) method and the cross-validated early stopping (CVES) method.

It is shown that the BR method outperforms the CVES method in general for making one step ahead forecasts for univariate time series, and it still outperforms the no-stop training (NST) method sometimes even when the ratio of the training data size to the number of weights in the ANN model is larger than 30. The BR method is further used to make 1- to 5-day ahead streamflow forecasts for the headwater of the Yellow River at TNH, and compared with the NST method. The result shows that for making 1- to 4-day ahead forecasts, BR outperforms NST. The major disadvantage of the BR method is its time-costliness. Therefore, when the speed of training is not a major concern, the BR method is recommended for making streamflow forecasts.

Interval forecasts are important to supplement point forecasts, especially for medium- and long-range forecasting, so as to define the predictive uncertainty. The residual based empirical approach and bootstrap approach are applied to construct prediction interval (PI) for monthly streamflow forecasts. The results show that both empirical approach and bootstrap method work reasonably well, and the empirical approach gives results comparable to or even better than bootstrap method. Because of the simplicity and calculation-effectiveness, empirical method is preferable to the bootstrap method. When there is significant seasonal variation in the variance of the residuals, to improve the PI construction, it is necessary to use seasonal empirical distribution functions which are defined by residuals in different seasons.

Predictability is an important aspect of the dynamics of hydrological processes. However, the predictability of hydrological processes has not attracted much attention by the hydrology community until recent several years. A univariate time series based approach is proposed in the present study, in which the predictability of a process is defined as the predictable horizon for which the prediction is no better than the mean value for a stationary process or the seasonal mean value for a seasonal process. At the same time, for practical purposes, we define the predictability at a given *CE* or *SACE* level as the predictable horizon for which the *CE* or *SACE* of predictions is larger than a given value. With such a definition, the predictability is easily comparable among different streamflow processes. Investigation of the predictabilities of a number of streamflow series at different basin scales shows that, in general, the larger the basin scale, the better the predictability.

8.4 Recommendations

One limitation of the present study in the aspect of forecasting is that only one streamflow process (the daily streamflow series of the Yellow River at the Tangnaihai gauging station) is considered for modelling in detail, and only in one case with the ANN model explanatory variables (such as precipitation data and upstream discharge data) are included for make daily streamflow forecast. To establish generality of the conclusions about forecast modelling, more streamflow processes should be analysed. In addition, for improving the accuracy of long-range streamflow forecasts, the investigation of the linkage between streamflow processes and remote factors (e.g., SSTs and ENSO events) would be inevitable.

A contribution in streamflow process modeling in this study is the introduction of periodic modeling approach, including the PAR model and the PANN model, for the case of daily streamflow forecasting. Both the PAR model and the PANN model may be considered as a hybrid modeling approach. As a matter of fact, besides the cluster-based and the threshold-based approaches which are compared with PANN in this study, there is a variety of other hybrid modeling techniques available nowadays to break the complicated streamflow forecasting problem down before solving the resulting sub-problems with different models (e.g., neural networks). It would be interesting to further compare the season-based PANN approach with other techniques, such as Bayesian-concept based modular ANN (e.g., Zhang et al., 2000), fuzzy-logic based hybrid modeling (e.g., See and Openshaw, 1999) and SOM-cluster based hybrid modeling (Abrahart and See, 2000). In addition, the PAR model and PANN model proposed in this study are based on univariate streamflow time series. It would be interesting to extent the idea of the PAR and PANN model to the cases where exogenous variables are included.

The ARMA-GARCH error model with seasonal standard deviations is proposed to capture the autoregressive conditional heteroskedasticity effects in streamflow processes. The purpose of proposing the model is to improve the estimation of forecasting variance, so as improve our knowledge of forecast uncertainty. It would be interesting to see whether we can get better estimation about the prediction interval with this type of model in the future work.

The characteristics of hydrological time series investigated in the present study, including the seasonality, nonstationarity, long-memory, nonlinearity, and conditional heteroskedasticity, may be closely linked to each other. For example, the test results of stationarity and long-memory for streamflow processes have similar timescale pattern, i.e., the shorter the timescale, the stronger the degree of stationarity or long-memory, and there is a general tendency that the stronger is the nonstationarity, the more intense is the long-memory. In fact, there are some attempts to use KPSS stationarity test to test for the existence of long-memory (e.g., Lee and Schmidt, 1996). It would be worthwhile to investigate such kind of linkage, and go further to see to what extent these characteristics are separable. This will help to give more insightful knowledge of the characteristics of different streamflow processes.

Pre-processing is of great importance in hydrological time series analysis. Because on one hand, the assumption of normality and stationarity by many time series models are often violated by the hydrological time series in their original forms; on the other hand, many hypothesis testing methods (e.g., Mann-Kendall trend test) require the removal of serial dependence from the time series. Therefore, one may pre-white, normalize, standardize, re-scale, deseasonalize the hydrological time series data before using them. In some cases the impacts of pre-processing have been addressed in the present study. For example, when analysing the stationarity, it was noticed that deseasonalization tends to make the KPSS test more likely to reject the hypothesis of stationarity; when testing for nonlinearity, an analysis was conducted about the impacts of different pre-processing procedures on the results; when forecasting daily streamflows with ANN models, several different pre-processing procedures are also compared. However, the side effects of pre-processing procedures worth further investigation.

References

Abraham, N. B. Albano, A.M., Das, B. et al., 1986. Calculating the Dimension of Attractors from Small Data. Phys. Lett. A 114, 217–221.

Abrahart, R.J., See, L., 2000. Comparing neural network and autoregressive moving average techniques for the provision of continuous river flow forecasts in two contrasting catchments. Hydrological Processes, 14, 2157-2172.

Agiakloglou, C., Newbold, P., Wohar, M., 1993. Bias in an estimator of the fractional difference parameter. J. Time Ser. Anal., 14, 235-246.

Akaike, H., 1973. Information theory and an extension of the maximum likelihood principle. In Proceedings of the 2nd International Symposium on Information Theory, B.N. Petrov and F. Csaki (eds.). Akademia Kiado, Budapest, 267-281.

Akaike, H., 1974. A New Look at Statistical Model Identification. IEEE Transactions on Automatic Control, 19(6), 716–722.

Aksu, C. Gunter, S.I., 1992. An empirical analysis of the accuracy of SA, OLS, ERLS and NRLS combination forecasts. International Journal of Forecasting 8(1), 27-43.

Amarasekera, K.N., Lee, R.F., Williams, E.R., Eltahir, E.A.B., 1997. ENSO and the natural variability in the flow of tropical rivers. J. Hydrol., 200, 24-39.

Amari, S.-i., Murata, N., Muller, K.-R., Finke, M., Yang, H.H., 1997. Asymptotic statistical theory of overtraining and cross-validation. IEEE Transactions on Neural Networks, 8 (5), 985–996.

Amorocho, J., 1963. Measures of the linearity of hydrological system. J. Geophys. Res. 68(8), 2237–2249.

Amorocho, J., Brandstetter, A., 1971. Determination of nonlinear functional response functions in rainfall–runoff processes. Water Resour. Res. 7(5), 1087–1101.

Anderson P.L., Meerschaert, M.M., 1998 Modeling river flows with heavy tails. Water Resources Research, 34(9), 2271-2280.

Andrews, Donald WK, 1991. Heteroskedasticity and Autocorrelation Consistent Covariance Matrix Estimation. Econometrica, 59, 817-858.

Armstrong, J.S., 1989. Combining forecasts: the end of the beginning or the beginning of the end? Int. J. Forecasting, 5, 585-588.

Armstrong, J.S., 2001. Combining forecasts. In: Principles of Forecasting: A Handbook for Researchers and Practitioners, Armstrong, J.S. (ed.). Boston: Kluwer Academic, 417-439.

Astatkie, T, Watts, D G, Watt, W E. Nested threshold autoregressive (NeTAR) models. International Journal of Forecasting, 1997, 13(1), 105-116.

Awadallah1 AG, Rousselle J., 2000. Improving Forecasts Of Nile Flood Using SST Inputs In TFN Model. Journal of Hydrological Engineering, 5(4), 371-379.

Ballini, R.; Soares, S.; Andrade, M.G., 2001. Multi-step-ahead monthly streamflow forecasting by a neurofuzzy network model. Proceedings of the Joint 9th IFSA World Congress and 20th NAFIPS International Conference, Vol. 2, Vancouver, Canada, 992-997.

Barnett, T. P., R. Preisendorfer, 1987. Origins and levels of monthly and seasonal forecast skill for United States surface air temperatures determined by Canonical Correlation Analysis, Mon. Weather Rev., 115, 1825–1850.

Barnston, A. G., T. M. Smith, 1996. Specification and prediction of global surface temperature and precipitation from global SST using CCA. J. Climate, 9, 2660– 2697.

Bartholmes J., Todini E., 2005. Coupling meteorological and hydrological models for flood forecasting. Hydrology and Earth System Sciences, 9(4), 333-346.

Bartlett, M.S., 1950. Periodogram Analysis and Continuous Spectra. Biometrika, 37, 1-16.

Bartolini, P., Salas, J.D., Obeysekera, J.T.B., 1988. Multivariate periodic ARMA(1,1) processes. Water Resources Research, 24, 1237-1246.

Basak,G.K., Chan,N.H., Palma, W., 2001. The approximation of long-memory processes by an ARMA model. Journal of Forecasting, 20(6), 367-389.

Batchelor, R., 1990. All forecasters are equal, Journal of Business and Economic Statistics, 8,143-144.

Bates, J.M., Granger, C.W.J., 1969. The combination of forecasts. Operational Research Quarterly 20, 451–468.

Bender M, Simonovic S., 1994. Time-Series Modeling for Long-Range Stream-Flow Forecasting. J. Water Res. Plan.Manag., 120(6), 857-870.

Beran J., 1994. Statistics for Long-Memory Processes. Chapman & Hall, New York.

Beven, K., Binley, A, 1992. The future of distributed models: model calibration and uncertainty estimation, Hydrol. Process., 6, 279-298.

Beven, K., Freer, J., 2001. Equifinality, data assimilation, and uncertainty estimation in mechanistic modeling of complex environmental systems using the GLUE methodology. Journal of Hydrology, 249, 11-29.

Bezdek, J.C., 1981. Pattern recognition with fuzzy objective function algorithms. Plenum, New York.

Bhansali, R.J. , 1992. Autoregressive estimation of the prediction mean squared error and an R2 measure: an application, in Brillinger, D. et al. (eds), New Directions in Time Series Part I, Springer-Verlag, New York, 9-24.

Birikundavyi S, Labib R, Trung HT, Rousselle J., 2002. Performance of neural networks in daily streamflow forecasting. J. Hydrol. Eng., 7 (5), 392-398.

Bloomfield, P., 1992. Trend in global temperature. Climatic Change, 21, 1-16.

Bollerslev, T., 1986. Generalized Autoregressive Conditional Heteroscedasticity. Journal of Econometrics, 31, 307–327.

Bollerslev, T., Engle, R.F., Nelson, D.B., 1994. ARCH models. In: Handbook of Econometrics, Vol. 4, (Eds) R.F. Engle and D.L. McFadden. Amsterdam: North Holland, 2959-3038.

Bordignon S, and Lisi F., 2000. Nonlinear analysis and prediction of river flow time series. Environmetrics, 11, 463-477.

Box, G.E.P., Cox, D.R., 1964. An analysis of transformations. Journal of the Royal Statistical Society, Series B, 26:211-252.

Box, G.E.P., Jenkins, G.M., 1976. Time series analysis: Forecasting and control (2nd edition). San Francisco: Holden-Day.

Brock, W.A., Dechert, W.D., Scheinkman, J.A., LeBaron, B., 1996. A test for independence based on the correlation dimension. Econ. Rev., 15(3), 197–235.

Brock, W.A., Hsieh, D.A and LeBaron, B., 1991. Nonlinear Dynamics, Chaos, and Instability: Statistical Theory and Economic Evidence. The MIT Press, Cambridge, MA.

Brockwell, P. J., Davis, R. A., 1991. Time Series and Forecasting Methods. Second edition. Springer, New York.

Bunn, D.W., 1975. A Bayesian approach to the linear combination of forecasts. Operational Research Quarterly 26, 325–329.

Bunn, D.W., 1989. Forecasting with more than one model. Journal of Forecasting, 8, 161-166.

Buntine, W.L., Weigend, A.S., 1991. Bayesian backpropagation. Complex Systems, 5(6), 603–643.

Burn, D.H., Hag Elnur, M.A., 2002. Detection of hydrological trend and variability. Journal of Hydrology 255, 107–122.

Butts, M.B., Payne, J.T., Overgaard, J., 2004. Improving streamflow simulations and flood forecasts with multimodel ensembles. Proceedings of 6th International Conference on Hydroinformatics, Singapore, 21-24 June, 2004.

Cai Y., Wang D.M., 1996. Correlation analysis between the land surface temperature field and the variation of the runoff within a year. Plateau Meteorology, 15(4), 472-477. (in Chinese)

Carney J.G., Cunningham P., Bhagwan U., 1999. Confidence and prediction intervals for neural network ensembles. In: Proceedings of International Joint Conference on Neural Networks,Vol.2, 1215-1218.

Chang F.J., Chen Y.C., 2001. A counterpropagation fuzzy-neural network modeling approach to real time streamflow prediction. J. Hydrol., 245, 153-164.

Chatfield, C., 1993. Calculating interval forecasts (with discussion), Journal of Business and Economic Statistics, 11, 121-144.

Chatfield, C., 2001. Prediction intervals for time series. in Principles of Forecasting: A Handbook for Practitioners and Researchers, Armstrong, J. S. (ed.), Norwall, MA: Kluwer Academic Publishers, 475-494.

Chatfield, C., 1996. Model uncertainty and forecast accuracy. Journal of Forecasting, 15, 495–508.

Chen, H.L., Rao A.R., 2002. Testing hydrological time series for stastionarity. J Hydrol. Eng., 7(2), 129-136.

Chen, H.L., Rao, A.R., 2003. Linearity analysis on stationary segments of hydrological time series. J. Hydrol., 277, 89-99.

Chernick, M.R., 1999. Bootstrap methods: A Practitioner's Guide. John Wiley & Sons, New York.

Chiew, F.H.S., S.L. Zhou, T.A. McMahon., 2003. Use of seasonal streamflow forecasts in water resources management. Journal of Hydrology, 270, 135–144.

Christoffersen, P.F., 1998. Evaluating interval forecasts, International Economic Review, 39(4), 841-862.

Cigizoglu, HK., 2003. Incorporation of ARMA models into flow forecasting by artificial neural networks. ENVIRONMETRICS, 14(4), 417-427.

Clark, M.P., Hay, L.E., 2004. Use of medium-range numerical weather prediction model output to produce forecasts of streamflow. J.Hydrometeor., 5(1), 15-32.

Clemen, R.T., 1989. Combining forecasts: A review and annotated bibliography. International Journal of Forecasting 5, 559–583.

Coulibaly, P., 2003. Impact of meteorological predictions on real-time spring flow forecasting. Hydrological Processes 17(18):3791-3801.

Coulibaly, P., Hache, M., Fortin, V., Bobee, B., 2005. Improving Daily Reservoir Inflow Forecasts with Model Combination. J. Hydrol. Eng., 10(2), 91-99.

Cover, T.M., P.E. Hart., 1967. Nearest neighbor pattern classification. IEEE Transactions on Information Theory, IT-13, 21-27.

Crane, D.B., Crotty, J.R., 1967. A two-stage forecasting model: Exponential smoothing and multiple regression, Management Science, 13, B501-B507.

Crato, N., Ray, B. K., 1996. Model selection and forecasting for long-range dependent processes. Journal of Forecasting, 15, 107–125.

Curry, K., Bras R.L., 1980. Multi-variate seasonal time series forecast with application to adaptive control. Technical Report 253, Ralph M. Parsons Laboratory, Department of Civil Engineering, Cambridge, MA: MIT.

Dawson, C.W., Wilby, R.L., 2001. Hydrological modeling using artificial neural networks. Progress in Physical Geography, 25(1), 80–108.

Dawson, C.W. Harpham, C. Wilby, R.L., Chen, Y., 2002. Evaluation of Artificial Neural Network Techniques for Flow Forecasting in the River Yangtze, China. Hydrology and Earth System Sciences, 6(4), 619-626.

Day G.N., 1985. Extended steramflow forecasting using NWSRFS. J. Water Resour. Plan. Manage, 111(2), 157-170.

de Menezes, L.M., Bun, D.W., Taylor, J.W., 2000. Theory and methodology review of guidelines for the use of combined forecasts. European J Oper. Res., 120, 190-204.

De Oliveira, K.A., Vannucci, A., da Silva, E.C., 2000. Using artificial neural networks to forecast chaotic time series. Physica A 284,393-404.

De Roo APJ, Bartholmes J, Bates PD, Beven K et al., 2003. Development of a European Flood Forecasting System. Journal of River Basin Management, 1(1), 49-59.

De Villars, J., Barnard, E., 1993. Backpropagation neural nets with one and two hidden layers. IEEE Trans. Neural Networks, 4(1), 136–141.

Dettinger, M.D., Cayan, D.R., McCabe, G.J., Redmond, K.T., 2000. Winter-Spring 2001 United States Streamflow Probabilities based on Anticipated Neutral ENSO Conditions and Recent NPO Status. Experimental Long-Lead Forecast Bulletin, 9(3), http://grads.iges.org/ellfb/Sep00/dettinger.pdf

Dettinger, M.D., Cayan, D.R., Redmond, K.T., 1999. United States Streamflow Probabilities based on Forecasted La Niña, Winter-Spring 2000. Experimental Long-Lead Forecast Bulletin, 8(4), http://grads.iges.org/ellfb/Dec99/dettinger.htm.

Dettinger, M.D., Cayan, D.R., Redmond, K.T., 2002. United States Streamflow Probabilities and Uncertainties based on Anticipated El Niño, Water Year 2003. Experimental Long-Lead Forecast Bulletin, 11(3), http://grads.iges.org/ellfb/Sep02/Dettinger/dettinger.pdf.

Deutsch, M., Granger, C.W.J., Terasvirta, T., 1994. The combination of forecasts using changing weights. International Journal of Forecasting 10, 47–57.

Dibike Y.B., Solomatine D.P., 2001. River Flow Forecasting Using Artificial Neural Networks, Journal of Physics and Chemistry of the Earth, Part B: Hydrology, Oceans and Atmosphere, 26(1), 1-8.

Dickey, D.A., Fuller,W.A., 1979. Distribution of the estimators for autoregressive time series with a unit root. J. Am. Stat. Assoc. 74, 423–431.

Dickinson, J.P., 1975. SOme comments on the combination of forecasts. Oper. Res. Q., 26, 205-210.

Diebold, F.X., Kilian, L., 2001. Measuring predictability: theory and macroeconomic applications. Jouranl of Applied Econometrics, 2001, 16(6), 657-669.

Dietz, E.J., T.J. Killeen., 1981. A Nonparametric Multivariate Test for Monotonic Trend With Pharmaceutical Applications. Journal of the American Statistical Association 76(373), 169-174.

Donaldson, G., Kamstra, M., 1996. Forecast combining with neural networks. Journal of Forecasting 15, 49–61.

Douglas, E. M., R. M. Vogel, and C. N. Kroll, 2000. Trends in floods and low flows in the United States: Impact of spatial correlation, J. Hydrol., 240, 90– 105.

Druce, D.J., 2001. Insights from a history of seasonal inflow forecasting with a conceptual hydrological model. J. Hydrol., 249, 102-112 .

Eckmann, J. P. Ruelle, D., 1992. Fundamental limitations for estimating dimensions and Lyapunov exponents in dynamical systems. Physica D 56, 185-187.

Eldaw, A.K., Salas, J.D., Garcia, L.A., 2003. Long-range forecasting of the Nile River flows using climatic forcing. J. Applied Meteo., 42 (7), 890-904.

Elshorbagy, A., Simonovic, S.P., Panu, U.S., 2002. Estimation of missing stream flow data using principles of chaos theory. J. Hydrol., 255, 125-133.

Eltahia, E.A.B., 1996. El Nino and the natural variability in the flow of the Nile River. Water Resour. Res., 32(1), 131-137.

Engle, R., 1982. Autoregressive conditional heteroscedasticity with estimates of the variance of UK inflation. Econometrica, 50, 987-1008.

Farmer, D.J., Sidorowich, J.J., 1987. Predicting chaotic time series. Phys. Rev. Lett. 59, 845–848.

Fast, B., 1990. Analysis of seasonal inflow volume forevasts produced using regression equations. Operations Control Department, British Columbia Hydro. Canada.

Fernando D.A.K., Jayawardena A.W., 1998. Runoff Forecasting Using RBF Networks with OLS Algorithm. Journal of Hydrological Engineering, 3(3), 203-209.

Fiordaliso, A., 1998. A nonlinear forecasts combination method based on Takagi–Sugeno fuzzy systems. International Journal of Forecasting, 14, 367–379.

Foresee, F. D., M. T. Hagan, 1997. Gauss-Newton approximation to Bayesian regularization. In: Proceedings of the 1997 International Joint Conference on Neural Networks, pp. 1930–1935.

Franz, K.J., Hartmann, H.C., Sorooshian, S., Bales, R., 2003. Verification of national weather service ensemble streamflow predictions for water supply forecasting in the Colorado River basin. J. Hydrometeor., 4(6), 1105-1118.

Fraser, A. M., Swinney, H. L., 1986. Independent coordinates for strange attractors from mutual information, Phys. Rev. A 33(2), 1134-1140.

Freer, J., Beven, K., Ambroise, B., 1996. Bayesian estimation of uncertainty in runoff prediction and the value of data: An application of the GLUE approach, Water Resour. Res., 32, 2161– 2173.

Fu, G., Chen, S., Liu C., Shepard D., 2004. Hydro-climatic trends of the Yellowriver basin for the last 50 years. Climatic Change 65, 149–178.

Garen, D.C., 1992. Improved techniques in regression-based streamflow volume forecasting. J Water Res. Plan. Manage., 118 (6), 654-670

Geman, S., Bienenstock, E., Doursat, R., 1992. Neural Networks and the Bias/Variance Dilemma, Neural Computation, 4, 1-58.

Geweke, J., Porter-Hudak S., 1983. The estimation and application of long memory time series models. J. Time Series Anal., 4:221-238.

Ghilardi, P., Rosso, R., 1990. Comment on chaos in rainfall. Water Resour. Res 26 (8), 1837–1839.

Gimeno, R., Manchado, B., Minguez, R., 1999. Stationarity tests for financial time series. Physica A 269, 72-78.

Giraitis, L., Robinson, P.M., Samarov, A., 1997. Rate optimal semiparametric estimation of the memory parameter of the Gaussian time series with long range dependence. J. Time Ser. Anal., 18 49-60.

Gladysev, E.G., 1961. Periodically correlated random sequences. Soviet Math. Dokl., 2, 385–388.

Goldfeld. S.M., Quandt, R.E., 1972. Nonlinear Methods in Econometrics. Amsterdam: North-Holland, 258-277.

Golub, G. H., Heath, M., Wahba, G., 1979. Generalized cross-validation as a method for choosing a good ridge parameter, Technometrics, 21(2), 215–223

Goswami, M., O'Connor, K.M., Shamseldin, A.Y., 2002. Structures and Performances of Five Rainfall-Runoff Models for Continuous River-Flow Simulation. In: Rizzoli, A.E., Jakeman, A.J. (eds), Integrated Assessment and Decision Support, Proceedings of the First Biennial Meeting of the International Environmental Modelling and Software Society, iEMSs: Manno, Switzerland, 2002, vol.1, 476-481.

Gouweleeuw B.T., Thielen J., Franchello G., De Roo A.P.J., Buizza R., 2005. Flood forecasting using medium-range probabilistic weather prediction. Hydrology and Earth System Sciences, 9(4), 365-380.

Grabs, W., 1997. Impact of climate change on hydrological regimes and water resources management in the Rhine basin. CHR report no. I-16, CHR, Lelystad, the Netherlands, pp. 17.

Granger C. W. J., Joyeux R., 1980. An Introduction to Long-Memory Time Series Models and Fractional Differencing, J. Time Series Anal., 1, 15-29.

Granger, C.W.J., Andersen, A.P., 1978. An introduction to bilinear time series models. Göttingen: Vandenhoeck and Ruprecht.

Granger, C.W.J., Ramanathan, R., 1984. Improved methods of combining forecasts. Journal of Forecasting, 3, 197–204.

Grassberger, P., 1990. An optimized box-assisted algorithm for fractal dimensions. Phys. Lett. A, 148, 63-68.

Grassberger, P., Procaccia, I., 1983a. Measuring the strangeness of strange attractors. Physica D 9, 189–208.

Grassberger, P., Procaccia, I., 1983b. Estimation of the Kolmogorov entropy from a chaotic signal. Phys. Rev. A, 28, 2591–2593.

Gunter, S.I., 1992, NonnegativityNonnegativity restricted least squares combinations. International Journal of Forecasting 8(1), 45-59.

Gutierrez F., Dracup, J.A. An analysis of the feasibility of long-range streamflow forecasting for Colombia using El Nino-Sourthern Oscillation indicators. J. Hydrology, 246, 181-196.

Hall F.R., 1968. Base flow recessions—a review. Water Resources Research 4, 973–983.

Hamilton, J.D., 1994. Time series analysis, Princeton University Press, Princeton.

Hamlet, A.F., Huppert, D., Lettenmaier, D.P., 2002, Economic Value of Long-Lead Streamflow Forecasts for Columbia River Hydropower. Journal of Water Resources Planning and Management, 128(2), 91-101.

Hamlet, A.F., Lettenmaier, D.P., 1999. Columbia River streamflow forecasting based on ENSO and PDO climate signals. J. Water Resour. Plan. Manage., 125(6), 333-341.

Han, D., I. D. Cluckie, D. Karbassioun, J. Lawry And B., 2002. Krauskopf. River Flow Modelling Using Fuzzy Decision Trees. Water Resources Management 16, 431–445.

Harvey A.C., 1993. Time series models (2nd edition). Massachusetts: MIT Press.

Haslett, J., Raftery, A. E., 1989. Space-time modeling with long-memory dependence: assessing Ireland's wind power resource (with discussion). Applied Statistics 38, 1-50.

Hastenrath S., 1990. Diagnosis and prediction of anomalous river discharge in northern South America. J. Climate, 3, 1080-1096.

Hauser, M.A., Kunst, R.M., 1998. Fractionally Integrated Models With ARCH Errors: With an Application to the Swiss 1-Month Euromarket Interest Rate. Review of Quantitative Finance and Accounting, 10, 95–113.

He, C.-Z, Xu X.-Z., 2005, Combination of forecasts using self-organizing algorithms. Journal of Forecasting, 24, 269–278.

Hegger, R., Kantz, H. Schreiber, T., 1999. Practical implementation of nonlinear time series methods: The TISEAN package. Chaos, 9, 413- 435.

Helsel, D.R., 1987. Advantages of nonparametric procedures for analysis of water quality data. J Hydrol. Sci., 32(2), 179-190.

Helsel, D.R., Hirsch R.M., 1992. Statistical methods in water resources. Amsterdam: Elsevier.

Henon, M., 1976. A two-dimensional mapping with a strange attractor. Commun. Math. Phys., 50(1), 69-77.

Henry, M., 2001. Robust automatic bandwidth for long-memory, J. Time Series Anal., 22(3), 293-316.

Hinich, M.J., 1982. Testing for Gaussianity and linearity of a stationary time series. J. Time Series Anal., 3 (3), 169–176.

Hipel, K.W., McLeod, A.I., 1994. Time series modeling of water resources and environmental systems. Elsvier, Amsterdam, 1994

Hirsch, R. M., 1981. Stochastic hydrological model for drought management. J. Water Resour. Plan. Manage., 107(2), 303-313

Hirsch, R.M., Slack J.R., 1984. A Nonparametric Trend Test for Seasonal Data with Serial Dependence. Water Resources Research, 20(6), 727-732.

Hirsch, R.M., Slack J.R., Smith, R.A., 1982. Techniques of Trend Analysis for Monthly Water Quality Data. Water Resources Research, 18(1), 107-121.

Hirsch, R.M., 1981. Stochastic hydrological model for drought management. J. Water Resour. Plan. Manage., 107(2): 303-313.

Holmström, L., Koistinen, P., 1992. Using additive noise in back-propagation training, IEEE Transaction on Neural Networks, 3, 24-38.

Hong, Shi-Zhong, Hong, Shi-Ming., 1994. An Amendment to the Fundamental Limits on Dimension Calculations. Fractals 2(1), 123-125.

Hornik, K., Stinchcombe, M., White, H., 1989. Multilayer feedforward networks are universal approximators. Neural Networks, 2(5), 359–366.

Hosking, J.R.M., 1981. Fractional Differencing, Biometrika, 68, 165-176.

Hsu, K., Gupta, H.V., Sorooshian, S., 1995. Artificial neural network modeling of the rainfall–runoff process. Water Resources Research, 31, 2517–30.

Hu, T.S., Lam, K.C., Ng, S.T., 2001. River flow time series prediction with a range-dependent neural network. Hydrol. Sci. J., 46(5), 729-745.

Hu, W., 2002. Flood Forecasting in the Wuding River using artificial neural networks and the large scale precipitation forecast. Msc thesis, UNESCO-IHE, Netherlands

Hughes, D.A., 2004. Incorporating groundwater recharge and discharge functions into an existing monthly rainfall–runoff model. Hydrological Sciences–Journal–des Sciences Hydrologiques, 49(2), 297-311

Huo, S.Q., Rao S.Q., Xue J.G., 2001. Inflow forecast for the Sanmen Gorge reservoir of the Yellow River during non-flooding period. People's Yellow River, 23(12), 14-16 (in Chinese) .

Hurst H.E., 1951. Long-term storage capacity of reservoirs. Transactions of the American Society of Civil Engineers, 116:770-808.

Hurvich, C.M., Bellrao, K., 1993. Asymptotics for the low-frequency ordinates of the periodogram of a long-memory time series. J. Time Series Anal., 14, 455-472.

Hurvich C.M., Deo, R.S., 1999. Plug-in Selection of the Number of Frequencies in Regression Estimates of the Memory Parameter of a Long-memory Time Series. J. Time Series Anal., 20(3), 331-341.

Hussain S., Elbergali A., 1999. Fractional order estimation and testing, application to Swedish temperature data. Environmetrics, 10, 339-349.

Ikeda, K., 1979. Multiple-valued stationary state and its instability of the transmitted light by a ring cavity system. Opt. Commun. 30, 257-261.

Imrie, C. E., Durucan S., Korre, A., 2000. River flow prediction using artificial neural networks: generalisation beyond the calibration range, J. Hydrol., 233, 138-153.

Insightful Corpration, 2001. Chapter 27 " Analyzing time series and signals" of the S-PLUS 6 for Windows Guide to Statistics, Volume 2, Insightful Corporation, Seattle, WA.

Islam, M.N., Sivakumar, B., 2002. Characterization and prediction of runoff. dynamics: a nonlinear dynamical view. Adv. Water Res., 25, 179–190.

Jacobs, R. A., Jordan, M.I., 1993. Learning piecewise control strategies in a modular neural network architecture. IEEE Trans. Syst. Man Cybern., 23(2), 337–345.

Jacobs, R.A., Jordan, M.I., Nowlan, S.J., Hinton, G.E., 1991. Adaptive mixtures of local experts. Neural Computation, 3, 79-87.

Jain S., Lall U., 2001. Floods in a changing climate: Does the past represent the future? Water Resources Research, 37(12), 3193-3205.

Jain, S.K., Das, A., Srivastava, D.K., 1999. Application of ANN for reservoir inflow prediction and operation, J. Water Res. Plann. Manage., 125(5), 263-271.

Jayawardena, A. W., Lai, F., 1994. Analysis and prediction of chaos in rainfall and stream flow time series. J. Hydrol., 153, 23–52.

Jeffrey, P. H., Salas., J.D., 1988. Development and testing of a multivariate, seasonal ARMA(1,1) model. Journal of Hydrology, 104, 247-272.

Johnson, O. W., Waples, R. S., Wainwright, T. C., Neely, K. G., Waknitz, F. W., Parker, L. T., 1994. Status review for Oregon's Umpqua River sea-run cutthroat trout. U.S. Dep. Commer., NOAA Tech. Memo. NMFS-NWFSC-15.

Jones, R.H., Brelsford, W.M., 1967. Time series with periodic structure, Biometrika, 54, 403-408.

Jordan, M. I., Jacobs, R.A., 1994. Hierarchical mixture of experts and the EM algorithm, Neural Comput., 6, 181–214.

Kahya, E., Dracup, J., 1993. US streamflow patterns in relation to the El Nino/Southern Osillation. Water Resour. Res., 29(8), 2491-2503.

Kahya, E., Dracup, J., 1994. The influence of Type 1 El Nino and La Nino events on streamflows in the Pacific Southwest of the United States. J. Climate, 7(6), 965-976.

Kalman, B.L., Kwasny, S.C., 1992. Why Tanh: Choosing a sigmoidal function. Proceedings of the International Joint Conference on Neural Networks, Baltimore. Vol. 4, pp. 578-581.

Kang, K. W., Kim, J. H., Park, C. Y., Ham, K. J., 1993. Evaluation of hydrological forecasting system based on neural network model. In: Proc., 25th Congress of Int. Assoc. for Hydr. Res., IAHR, Delft, The Netherlands, 257–264.

Kantz, H., Schreiber, T., 2004. Nonlinear Time Series Analysis (2nd). Cambridge University Press, Cambridge.

Karagiannis T., Molle M., Faloutsos M., 2004. Long-range dependence: ten years of internet traffic modeling. IEEE Internet Computing, 8(5): 57-64.

Karanasos, M., 2001. Prediction in ARMA models with GARCH-in-mean effects. J. Time Series Anal., 22, 555–78.

Karlsson, M.S., 1985. Nearest neighbor regression estimators in rainfall-runoff forecasting. Ph.D thesis, University of Arizona.

Kember, G., Flower, A.C., Holubeshen, J., 1993. Forecasting river flow using nonlinear dynamics, Stocastic Hydrol. Hydraul. 7, 205–212.

Kendall, M.G., 1938. A New Measure of Rank Correlation. Biometrika, 30, 81-93.

Kendall, M.G., 1975. Rank Correlation Methods, Charles Griffin, London.

Kendziorskia C.M., Bassingthwaighteb J.B., Tonellato, P.J., 1999. Evaluating maximum likelihood estimation methods to determine the Hurst coefficient. Physica A, 273, 439-451.

Kennel, M.B., Brown, R., Abarbanel., H.D., 1992. Determining embedding dimension for phase-space reconstruction using geometrical construction. Phy. Rev. A, 45, 3403-3411.

Khu, S.T., Werner, M.G.F., 2003. Reduction of Monte Carlo simulation runs for uncertainty estimation in hydrological models, Hydrol. Earth System Sc., 7(5), 680-692

Kim, H.S., Kang, D.S., Kim, J.H., 2003. The BDS statistic and residual test. Stochast. Environ. Res. Risk Assess., 17, 104–115.

Kim, Y.O., Palmer, R.N., 1997. The value of seasonal flow forecasts in Bayesian stochastic programming. Journal of Water Resources Planning and Management, 123(6), 327-335.

Kisi O., 2004. River Flow Modeling Using Artificial Neural Networks. J. Hydrol. Eng., 9(1), 60-63 .

Kitanidis P.K., Bras R.L., 1980. Real-tie forecasting with a conceptual hydrological model: 1., Analysis of uncertainty. Water Resources Research, 16(6), 1025-1033.

Klemes, V., 1974. The Hurst phenomenon: A Puzzle?, Water Resources Research 10, 675-688.

Koutsoyiannis, D., Pachakis, D., 1996. Deterministic chaos versus stochasticity in analysis and modeling of point rainfall series. J. Geophys. Res., 101 (D21), 26 441–26 451.

Krishnamurti, T.N, Kishtawal, C.M., LaRow, T. T. E., Bachiochi, D. R., Zhang, Z., Williford, C. E., Gadgil, S., Surendran, S., 1999. Improved Weather and Seasonal Climate Forecasts from Multimodel Superensemble, Science, 285, 1548-1550.

Krogh, A., Hertz., J.A., 1992. A simple weight decay can improve generalization. In J. E. Moody, S. J. Hanson, and R. P. Lippmann, editors. Advances in Neural Information Processing Systems 4. San Mateo, CA: Morgan Kaufmann, 950-957.

Krzysztofowicz R., 1999. Bayesian theory of probabilistic forecasting via deterministic hydrological model. Water Resour. Res., 35(9), 2739-2750.

Krzysztofowicz R., 2001. The case for probabilistic forecasting in hydrology. J. Hydrol., 249, 2-9 .

Krzysztofowicz, R., 1999. Bayesian theory of probabilistic forecasting via deterministic hydrological model. Water Resour. Res., 35(9), 2739-2750

Kuczera, G., Parent E., 1998. Monte Carlo assessment of parameter uncertainty in conceptual catchment models: The Metropolis algorithm. J. Hydrol., 211, 69– 85.

Kundzewicz, Z.W., Graczyk, D., Maurer, T., Przymusinska, I., Radziejewski, M., Svensson, C., Szwed, M., 2004. Detection of change in world-wide hydrological time series of maximum annual flow. GRDC Report 32, GRDC, Koblenz, Germany.

Kuo I.T., Sun Y.H., 1993. An intervention model for average 10 day streamflow forecast and synthesis, Journal of Hydrology, 151(1), 35-56.

Kwiatkowski, D., Phillips, P.C.B., Schmidt, P., Shin, Y., 1992. Testing the null of stationarity against the alternative of a unit root: How sure are we that economic time series have a unit root? Journal of Econometrics 54, 159–178.

Kwok, T.Y., Yeung, D.Y., 1997. Constructive algorithms for structure learning in feedforward neural networks for regression problems. IEEE Transactions on Neural Networks 8, 630–45.

Kyriakidis P.C., Miller N.L., Kim J., 2001. Uncertainty propagation of regional climate model precipitation forecasts to hydrological impact assessment. Journal of Hydrometeorology, 2(2), 140–160.

Lachtermacher, G., Fuller, J.D. Back propagation in time!series forecasting. Journal of Forecasting 14, 381-393.

Lall, U., 1995. Recent advance in nonparametric function estimation - hydrological application. Reviews of Geophysics, 33 Suppl: 1093-1102.

Lawson, C.L. Hanson, R.J., 1974. Solving Least-Squares Problems (Chapter 23), Prentice-Hall, Englewood Cliffs, N.J., pp.161.

Le Cun, Y., Denker, J. S., Solla, S. A., 1990. Optimal brain damage. In: D. S. Touretzky (Ed.) Advances in Neural Information Processing Systems 2. San Mateo, CA: Morgan Kaufmann, 598-605.

Lee, D., Schmidt, P., 1996. On the power of the KPSS test of stationarity against fractionally-integrated alternatives. J. Econometrics, 73(1), 285-302.

Lettenmaier, D.P., Wood, E.F. Chapter 26 Hydrological forecasting. In: D.R. Maidment (ed.), Handbook of Hydrology. McGraw-Hill, New York, 199326.1-26.30.

Li, J. Q., 1998. Comparison of monthly rainfall-runoff models for catchments in northern China. Advances in Water Sciences. 9(3), 282-288 (in Chinese).

Linsley, R. K., Kohler, M. A., Paulhus, J.L.H., 1958. Hydrology for Engineers (2nd ed.). New York: McGraw-Hill.

Liu, Q., Islam, S., Rodriguez-Iturbe, I., Le, Y., 1998. Phase-space analysis of daily streamflow: characterization and prediction. Adv. Water Resour., 21, 463–475.

Liu, X. R., 1997. Sequential hydrological models. J. Hohai Univer., 25(3), 7-14 (in Chinese).

Livina, V., Ashkenazy, Y., Kizner, Z., Strygin, V., Bunde, A., Havlin, S., 2003. A stochastic model of river discharge fluctuations, Physica A, 330, 283-290.

Lo A.L., 1991. Long-term memory in stock market prices. Econometrica, 59 (5), 1279–1313.

Lorenz, E. N., 1991. Dimension of weather and climatic attractors. Nature, 353, 241–244.

Lu, H.Y., Shao, D.G., Guo, Y.Y., 1996. Research on long-term runoff forecasting for the Danjiangkou Reserviour. Journal of Wuhan hydraulics and hydropower University, 29(6), 6-10 (in Chinese).

Mackay, D., 1991. Bayesian methods for adaptive models. PhD Thesis, California Institute of Technology.

Mackey, M. C., L. Glass., 1977. Oscillations and chaos in physiological control systems. Science, 197, 287-289,

Mahabir, C., Hicks, F. E., Robinson Fayek. A., 2003. Application of fuzzy logic to forecast seasonal runoff. Hydrol. Process., 17, 3749–3762.

Maier, H.R., Dandy, G.C., 1998. The effect of internal parameters and geometry on the performance of back-propagation neural networks: An empirical study. Environmental Modelling and Software, 13 (2), 193–209.

Maier, H.R., Dandy, G.C., 2000. Neural networks for the prediction and forecasting of water resources variables: a review of modeling issues and applications. Environmental Modelling and Software 15, 101–124.

Makridakis, S., Anderson, A., Carbone, R., Fildes, R., Hibon, M., Lewandowski, R., Newton, J., Parzen, E., Winkler, R. (1982), The accuracy of extrapolation (time series) methods: results of a forecasting competition. Journal of Forecasting 1, 111-153.

Makridakis, S., Winkler, R.L., 1983. Average of forecasts: some empirical results. Manage. Sci., 29(9), 987-996.

Man, K.S., 2003. Long memory time series and short term forecasts. International Journal of Forecasting 19, 477–491.

Mandelbrot, B.B., Wallis, J.R., 1969a. Computer experiments with fractional Gaussian noises. Part 1, 2 and 3. Water Resources Research 5, 228–267.

Mandelbrot, B.B., Wallis, J.R., 1969b. Some long-run properties of geophysical records, Water Resour. Res., 5, 321– 340.

Mandelbrot, B.B., Wallis, J.R., 1969c. Robustness of the rescaled range R/S in the measurement of noncyclic long-run statistical dependence. Water Resour. Res., 5, 967-988.

Mann, H.B., 1945. Nonparametric test against trend. Econometrica, 13, 245-259.

Marengo, J.A., 1995. Variations and change in South American streamflow. Climate Change, 31, 99-117.

Markus, M., 1997. Application of neural networks in streamflow forecasting. Ph.D dissertation, Department of Civil Engineering, Colorado State University, Fort Collins, Colorado

Marquardt, D.W., 1963. An algorithm for least-squares estimation of nonlinear parameters. J. Soc. Indust. Appl. Math., 11(2), 431–441.

Marsden, M.A., Davis., R.T., 1968. Regression on principle components as a tool in water supply forecasting. Proceedings of the 36th Western Snow Conference, 33-40.

Maurer, E.P., Lettenmaier, D. P., 2003. Predictability of seasonal runoff in the Mississippi River basin. J. Geophys. Res., 108(D16), 8607, doi: 10.1029/2002JD002555.

Maurer, E.P., Lettenmaier, D.P., Mantua N.J., 2004. Variability and potential sources of predictability of North American runoff. Water Resources Research, 40, W09306, doi:10.1029/2003WR002789.

McKerchar, A.I., Delleur, J.W., 1974. Application of seasonal parametric linear stochastic models to monthly flow data. Water Resources Research, 10, 246-255.

McLeod A.I., Hipel K.W., 1978. Preservation of the Rescaled Adjusted Range, I. A Reassessment of the Hurst Phenomenon. Water Resources Research, 14, 491-519.

McLeod, A.I., 1994. Diagnostic Checking of Periodic Autoregression. J. Time Series Anal., 15(2), 221–233.

McLeod, A.I., 1975. Derivation of the theoretical autocovariance function of autoregressive-moving average time series. Applied Statistics, 24(2), 255-256.

McLeod, A.I., Hipel, K.W., Lennox, W.C., 1977. Advances in Box-Jenkins Modeling 2. Applications. Water Resources Research, 13, 577-586.

McLeod, A.I., Li, W.K., 1983. Diagnostic checking ARMA time series models using squared residual autocorrelations. J. Time Series Anal., 4, 269–273.

Mcleod, A.I., Noakes, D. J., Hipel, K. W., Thompstone R.M., 1987. Combining hydrological forecasts. J. Water Res. Plan. Manage., 113(1), 29-41.

Meade, N., Maier, M.R., 2003. Evidence of long memory in short-term interest rates. J. Forecast., 22, 553–568.

Mechoso, C., Perez-Iribarren, G., 1992, Streamflow in southeast South America and the Southern Osillation. J. Climate, 5, 1535-1539.

Milanese, M., Novara, C., 2004. Nonlinear Set Membership prediction of river flow. Systems & Control Letters, 53, 31–39.

Miller, R.B., 1979. Book review on "An Introduction to Bilinear Time Series Models" by C.W. Granger and A.P. Andersen. J. Amer. Statist. Ass., 74, 927.

Minns, A.W., Hall, M.J., 1996. Artificial neural networks as rainfall–runoff models. Hydrological Sciences Journal, 41, 399– 417.

Minshall, N.E., 1960. Predicting storm runoff on small experimental watershed. J. Hydraul. Div. Am. Soc. Eng. 86(HYB), 17-38.

Mishra, A., Hata, T., Abdelhadi, A. W., Tada, A., Tanakamaru1, H., 2003. Recession flow analysis of the Blue Nile River. Hydrol. Process., 17, 2825–2835.

Mizumura, K., 1995. Application of fuzzy theory to snowmelt-runoff. In: Kundzewicz, Z.W.(Ed.), New uncertainty concepts in hydrology and water resources, Cambridge University Press, New York.

Montanari A., Brath A., 2004. A stochastic approach for assessing the uncertainty of rainfall-runoff simulations. Water Resources Research, 40, doi:10.1029/2003WR002540.

Montanari, A., Rosso, R., Taqqu, M. S., 1997. Fractionally differenced ARIMA models applied to hydrological time series: Identification, estimation, and simulation. Water Resources Research, 33(5), 1035–1044.

Montanari, A., Rosso, R., Taqqu, M. S., 2000. A seasonal fractional ARIMA model applied to the Nile River monthly at Aswan. Water Resources Research, 36:1249-1259.

Moore, R.J., Bell, V.A., Carrington, D.S., 2000. Intercomparison of rainfall-runoff models for flood forecasting. In: Lees, M., Walsh, P. (Eds.), Flood forecasting: what does current research offer the practitioner?, Occasional Paper No. 12, British Hydrological Society, pp. 69–76.

Moradkhani, H., Hsu, K.L., Gupta, H.V., Sorooshian, S., 2004. Improved streamflow forecasting using self-organizing radial basis function artificial neural networks. J Hydrol., 295, 246–262.

Nash, J. E., Sutcliffe, J.V., 1970. River flow forecasting through conceptual models, I, A discussion of principles, J. Hydrol., 10, 282–290.

Nash, J.E., Barsi, B.I., 1983. A hybrid model for flow forecasting on large catchments. Journal of Hydrology, 65, 125-137.

Nash, J.E., Foley, J.J., 1982. Linear models of rainfall-runoff systems. In: Proceedings of the International Symposium on Rainfall-Runoff modelling. Mississippi State University, V.P. Singh (Ed.), Water Resources Publications, 51-66.

Nayak, P. C., Sudheer, K. P., Ramasastri, K. S., 2005. Fuzzy computing based rainfall–runoff model for real time flood forecasting. Hydrol. Process., 19, 955–968.

Nayak, P.C., Sudheerb, K.P., Ranganc, D.M., Ramasastri, K.S., 2004. A neuro-fuzzy computing technique for modeling hydrological time series. J. Hydrol., 291(1-2), 52-66.

Neal, R.M., 1996. Bayesian Learning for Neural Networks, New York: Springer-Verlag

Nerenberg, M.A.H., Essex, C., 1990. Correlation dimension and systematic geometric effects. Phys. Rev. A 42(12), 7065–7074.

Newey, W.K., West, K.D., 1987. A simple, positive semi-definite, heteroskedasticity and autocorrelation consistent covariance matrix, Econometrica 55, 703-708.

Noakes, D.J., McLeod, A.I., Hipel, K.W., 1985. Forecasting monthly riverflow time series. International Journal of Forecasting, 1, 179-190.

Novara, C., Milanese, M., 2001. Set Membership prediction of nonlinear time series. In: Proceedings of the 40th IEEE Conference on Decision and Control, Orlando, U.S., 2131–2136.

Nowlan, S. J. and G. E. Hinton., 1992. Simplifying neural networks by soft weight-sharing. Neural Computation, 4(4), 473-493.

NRC, 2002. Report of a workshop on predictability & limits-to-prediction in hydrological systems. National Academy Press, Washington, D.C.

NRC, 1999. Hydrological Science Priorities for the U.S. Global Change Research Program: An Initial Assessment. National Academy Press, Washington, D.C., 6-10.

O'Connor K.M., Goswami M., Bhattarai K.P., Shamseldin A.Y., 2004. A comparison of the lead-time discharge forecasts of the 'Perfect' and 'Naïve-AR' quantitative precipitation forecast (QPF) input scenarios, to assess the value of having good QPFs. In: Hydrological Risk; Recent advances in peak river flow modelling, prediction and real-time forecasting – Assessment of impacts of land use and climate change, Brath, A., Montanari, A. and Toth, E. (Eds.). CNR-GNDCI Publ. No. 2858, Editoriale Bios, Via A. Rendano, 25-87040 Castrolibero (CS), Italy, pp. 187-217.

Ooms, M., Franses, P.H., 2001. A seasonal periodic long memory model for monthly river flows. Environmental Modelling & Software, 16, 559-569.

Packard, N. H., Crutchfield, J. P., Farmer, J. D., Shaw, R. S., 1980. Geometry from a time series, Phys. Rev. Lett. 45(9), 712–716.

Pagano, M., 1978. On periodic and multiple autoregressions, The Annals of Statistics, 6, 1310-1317.

Pal, N.R., Pal, S., Das, J., Majumdar, K., 2003. SOFM–MLP: A Hybrid Neural Network for Atmospheric Temperature Prediction. IEEE Transactions on Geoscience and Remote Sensing, 41(12), 2783 – 2791.

Park Y.R., Murray T.J., Chen,C., 1996. Predicting sun spots using a layered perceptron neural network. IEEE Trans. Neural Networks, 7, 501–505.

Parker D., Tunstall S., Wilson T., 2005. Socio-economic benefits of flood forecasting and warning. In: Proceedings of the International Conference on Innovation, advances and implementation of flood forecasting technology, October 17-19, 2005, Norway, Tromso.

Pascual, L., Romo, J., Ruiz, E., 2004. Bootstrap predictive inference for arima processes. J. Time Series Analysis, 25(4), 449-465.

Pasternack, G.B., 1999. Does the river run wild? Assessing chaos in hydrological systems. Advances in Water Resources 23, 253-260.

Patterson, D.M., Ashley, R.A., 2000. A nonlinear time series workshop: a toolkit for detecting and identifying nonlinear serial dependence. Kluwer Academic, Boston

Pencavel, John H., 1971. A note on the predictive performance of wage inflation models of the British economy, Economic Journal, 81, 113-119.

Peng, M.X., Ge Z.X., Wang H.B., 2000. The relationship between the runoff of the upstream Yellow River and the sea surface temperature of the Pacific and its application for prediction. Advance in Water Sciences, 1(3), 272-276 (in Chinese)

Perica S., 1998. Integration of Meteorological Forecasts/Climate Outlooks into an Ensemble Streamflow Prediction System. 78th AMS Annual Meeting, Phoenix, Arizona, 130-133.

Piechota, T.C., Dracup, J.A., 1999. Long-Range Streamflow Forecasting Using El–Nino Southern Oscillation Indicators, J. Hydrol. Eng., 4(2), 144-151

Piechota, T.C., Dracup, J.A., Chiew, F.H.S., McMahon, T.A., 1998. Seasonal streamflow forecasting in Eastern Australia and the El Nino–southern oscillation. Water Resour. Res., 34(11), 3035–3044.

Pitman, W. V., 1973. A mathematical model for generating river flows from meteorological data in South Africa. Report no. 2/73, Hydrological Research Unit, University of the Witwatersrand, Johannesburg, South Africa.

Porporato, A., Ridolfi, L., 1997. Nonlinear analysis of river flow time sequences, Water Resour. Res., 33(6), 1353–1367.

Porporato, A., Ridolfi. L., 2001. Multivariate nonlinear prediction of river flows. J. Hydrol., 248, 109-122.

Prechelt, L., 1997. Connection pruning with static and adaptive pruning schedules. Neurocomputing, 16(1), 49-61

Prechelt, L., 1998. Early stopping – But when? In: Orr, G.B. and Mueller, K.-R., eds., Neural Networks: Tricks of the Trade. Berlin: Springer, 55-69.

Priestley, M.B., 1988. Non-linear and non-stationary time series analysis. London: Academic Press.

Procaccia, I., 1988. Complex or just complicated? Nature 333, 498-499.

Rajurkar, M.P., Kothyari, U.C., Chaube., U.C., 2004. Modeling of the daily rainfall-runoff relationship with artificial neural network. J. Hydrol., 285, 96-113.

Raman, H., Sunilkumar, N., 1995. Multivariate modeling of water resources time series using artificial neural network. Hydrol. Sci. J., 40(2), 145–163.

Rao, A.R. Bhattacharya, D., 1999. Hypothesis testing for long-term memory in hydrological series. Journal of Hydrology, 216(3-4), 183-196.

Rao, A.R., Yu, G.-H., 1990. Gaussianity and linearity tests of hydrological time series. Stochast. Hydrol. Hydraul. 4, 121–134.

Reed, R., 1993. Pruning algorithms—A survey. IEEE Trans. Neural Networks, 4,740–747.

Reed, S., Koren, V., Smith, M., Zhang, Z., Moreda, F., Seo, D.-J., DMIP participants, 2004. Overall distributed model intercomparison project results. Journal of Hydrology 298(1–4), 27–60.

Refsgaard J. C., Storm B., 1996. Construction, calibration and validation of hydrological models. In: Distributed Hydrological Modelling (ed. by M. B. Abbott & J. C. Refsgaard), Dordrecht: Kluwer Academic, 41–54.

Refsgaard, J.C., Knudsen J., 1996. Operational validation and intercomparison of different types of hydrological models. Water Resources Research, 32(7), 2189-2202.

Ribeiro J., Lauzon N., Rousselle J., Trung H.T., Salas J.D., 1998. Comparaison de deux modèles pour la prévision journalière en temps réel des apports naturels. Can. J. Civ. Eng., 25, 291-304

Robinson, J.S., Sivapalan, M., Snell, J.D., 1995. On the relative roles of hillslop processes, channel routing and network geomorphology in the hydrological response of natural catchments. Water Resour. Res., 31(12), 3089-3101.

Rogers, W.F., 1980. A practical method for linear and nonlinear runoff. J. Hydrol., 46, 51-78.

Rogers, W.F., 1982. Some characteristics and implications of drainage basin linearity and nonlinearity. J. Hydrol., 55, 247-265.

Rogers, W.F., Zia, H.A., 1982. Linear and nonlinear runoff from large drain basins. J. Hydrol., 55, 267-278.

Said, S.E., Dickey, D., 1984. Testing for unit roots in autoregressive moving-average models with unknown order. Biometrika, 71, 599-607.

Sajikumar, N., Thandaveswara, B.S., 1999 A nonlinear rainfall runoff model using an artificial neural network. J. Hydrol., 216, 32–55.

Salas, J.D., 1993. Analysis and modeling of hydrological time series. In: Handbook of Hydrology, edited by David R. Maidment. McGraw-Hill, New York, 19.1-19.72.

Salas, J.D., Abdelmohsen, M.W., 1993. Initialization for generating single-site and multisite low-order periodic autoregressive and moving average processes, Water Resources Research, 29(6), 1771-1776.

Salas, J.D., Boes, D.C., Smith, R.A., 1982. Estimation of ARMA models with seasonal parameters. Water Resources Research, 18, 1006-1010.

Saltzman, B., 1983. Climatic system analysis. Advances in Geophysics, 25, 173–233.

Sankarasubramanian, A., Lall, U., 2003. Flood quantiles in a changing climate: Seasonal forecasts and causal relations. Water Resources Research, 39(5), Art. No. 1134.

Sarle, W.S., 2002. Neural Network FAQ, part 3 of 7: Generalization. URL: ftp://ftp.sas.com/pub/neural/FAQ3.html

Sauer, T., Yorke, J.A., 1993. How many delay coordinates do you need? Int., J. Bif. Chaos 3, 737-744.

Schaake, J.C., Welles, E., Graziano, T., 2001. Comment on "Bayesian theory of probabilistic forecasting via deterministic hydrological model". Water Resour. Res., 37(2), 439.

Schertzer D, Tchiguirinskaia I, Lovejoy S, Hubert P, Bendjoudi H., 2002. Discussion on "Evidence of chaos in the rainfall-runoff process": Which chaos in the rainfall-runoff process? Hydrol Sci J. 47(1), 139-147.

Schreiber, T., Schmitz, A., 1996. Improved surrogate data for nonlinearity tests, Phys. Rev. Lett. 77(4), 635-638.

Schwert, G.W., 1989. Tests for unit roots: A Monte Carlo investigation. J. Business Eco. Stat. 7,147-159.

See, L., Abrahart, R.J., 2001. Multi-model data fusion for hydrological forecasting. Computers & Geosciences, 27, 987–994.

See, L., Openshaw, S., 1999. Applying soft computing approaches to river level forecasting. Hydrol. Sci. J., 44 (5), 763 –778.

See, L., Openshaw, S., 2000. A hybrid multi-model approach to river level forecasting. Hydrol. Sci. J. 45 (4), 523–536.

Shamseldin, A. Y., O'Connor, K. M., 1996. A nearest neighbor linear perturbation model for river flow forecasting. J. Hydrol., 179, 353–375.

Shamseldin, A. Y., O'Connor, K. M., Liang, G. C., 1997. Methods for combining the output of different rainfall-runoff models. J. Hydrol., 197, 203–229.

Sharkey, A.J.C., 1996. On Combining Artificial Neural Nets. Connection Science, 8 (3 & 4), 299-313.

Shin, Y., Schmidt, P., 1992. The KPSS stationarity test as a unit root test. Economics Letters, 38, 387-392.

Shrestha, A., Kostaschuk, R., 2005. El Niño/Southern Oscillation (ENSO)-related variablity in mean-monthly streamflow in Nepal. J. Hydrol., 308, 33-49.

Silverman, B. W., 1986. Density Estimation, Chapman and Hall, London.

Simpson, H.J., Cane, M.A., 1993. Annual river discharge in south-eastern Australia related to El Nino-Southern Oscilation forecasts of sea surface temperatures. Water Resour. Res., 29(11), 3671-3680.

Sivakumar B, Berndtsson R, Persson M., 2001. Monthly runoff prediction using phase space reconstruction. Hydrol. Sci. J., 46(3), 377-387

Sivakumar, B., 2000. Chaos theory in hydrology: important issues and interpretations. J. Hydrol., 227, 1-20.

Sivakumar, B., Liong, S.-Y., Liaw, C.-Y., Phoon, K.-K., 1999. Singapore rainfall behavior: chaotic? J. Hydrol. Eng. 4(1), 38–48.

Smith, J. A., Day, G.N., Kane, M.D., 1992. Nonparametric framework for long-range streamflow forecasting. J. Water Resour. Plan. Manage., 118(1), 82-92.

Smith, J.A., 1991. Long-Range Streamflow Forecasting Using Nonparametric Regression. Water Resources Bulletin, 27(1), 39-46.

Smith, M.B., Seo, D.-J., Koren, V.I., Reed, S., Zhang, Z., Duan, Q.-Y., Cong, S., Moreda, F., Anderson, R., 2004. The Distributed Model Intercomparison Project (DMIP): Motivation and Experiment Design. Journal of Hydrology, 298(1–4), 4–26.

Solomatine, D. P., Dulal, K.N., 2003. Model trees as an alternative to neural networks in rainfall-runoff modeling. Hydrol. Sci. J., 48(3), 399-411.

Solomatine, D.P., 2002. Data-driven modelling: paradigm, methods, experiences. Proc. 5th Int. Conference on Hydroinformatics. Cardiff, UK, 757-763.

Solomatine, D.P., Dulal., K.N., 2003. Model trees as an alternative to neural networks in rainfall–runoff modelling. Hydrological Sciences Journal, 48(3), 399–411.

Solomatine, D.P., Xue, Y., 2004. M5 model trees compared to neural networks: application to flood forecasting in the upper reach of the Huai River in China. Journal of Hydrological Engineering, 9(6), 491-501.

Sowell, F., 1992. Modeling Long-Run Behavior with the Fractional ARIMA Model, Journal of Monetary Economics, 29, 277-302.

Stedinger, J. R., Grygier, J., Yin, H., 1989. Seasonal streamflow forecasts based upon regression. In: Labadie JW, Brazil LE, Corbu I and Johnson LE (eds.), Computerized decision support systems for water managers. New York: ASCE, pp. 266-279.

Stine, R. A., 1987. Estimating properties of autoregressive forecasts. J. Am. Statist. Assoc., 82, 1072–8.

Sugihara, G., May, R.M., 1990. Nonlinear forecasting as a way of distinguishing chaos from measurement error in time series. Nature, 344, 734–741.

Takens, F., 1981. Detecting strange attractors in turbulence. Lecture Notes in Mathematics 898, 366–381.

Tallaksen, L.M., 1995. A review of baseflow recession analysis. Journal of Hydrology, 165, 349–370.

Tang, C.W., Zhang, J., Yang, L., 1992. Correlation analysis between the runoff volume and the earthquake in China. Chinese Sciences B, (8), 889-896 (in Chinese).

Tangborn, W.V., Rasmussen, L.A., 1976. Hydrology of the North Cascades region, Washington - Part 2: A propsed hydrometeorological streamflow prediction method. Water Resources Res., 12(2), 203-216.

Tawfik, Maha., 2003. Linearity versus non-linearity in forecasting Nile River flows. Advances in Engineering Software, 34, 515–524.

Taylor J.W., Numm D.W., 1999. Investigating improvements in the accuracy of prediction intervals for combinations of forecasts: A simulation study. International Journal of Forecasting, 15(3), 325-339.

Teverovsky, V., Taqqu, M.S., Willinger, W., 1999. A critical look at Lo's modified R/S statistic. Journal of Statistical Planning and Inference, 80,211-227.

Theiler, J., 1986. Spurious dimension from correlation algorithms applied to limited time-series data, Phys. Rev. A 34(3), 2427–2432.

Theiler, J., 1990. Estimating fractal dimension. J. Opt. Soc. Am. A 7, 1055-1073.

Theiler, J., Eubank, S., 1993. Don't bleach chaotic data. Chaos 3, 771-782.

Theiler, J., Eubank, S., Longtin, A., Galdrikian, B., Farmer, J. D., 1992 Testing for nonlinearity in time series: the method of surrogate data, Physica D 58, 77–94.

Thomas, H.A., Fiering, M.B., 1962. Mathematical synthesis of stream flow sequences for the analysis of river basins by simulation. In: Design of Water Resources, Ed. Maass, A., Hufschmidt, M.M., Dorfman, R.,Thomas, H.A., Marglin, S.A., Fair, G.M. Cambridge, MA: Harvard University Press.

Thomas, L. A., Schucany, W. R., 1990. Bootstrap prediction intervals for autoregression. J. Am. Statist. Assoc. 85, 486–92.

Thompstone, R.M., Hipel, K.W., Mcleod, A.I., 1985. Forecasting quarter-monthly riverflow. Water Resources Bulletin, 21(5), 731-741.

Tiao, G. C., Tsay, R. S., 1994. Some advances in non-linear and adaptive modeling in time-series. J. Forecasting, 13, 109-131.

Tingsanchali, T., Gautam, M. R., 2000. Application of tank, NAM, ARMA and neural network models to flood forecasting, Hydrol. Process, 14, 1362_1376.

Tokar, A. S., Markus, M., 2000. Precipitation-runoff modeling using artificial neural network and conceptual models. J. Hydrological Eng., 5(2), 156–161.

Tol, R.J.S., 1996. Autoregressive conditional heteroscedasticity in daily temperature measurements. Environmetrics, 7, 67–75.

Tomasino, M., Zanchettin D., Traverso, P., 2004. Long-range forecasts of River Po discharges based on predictable solar activity and a fuzzy neural network model. Hydro. Sci. J., 49(4), 673-684.

Tong, H., Lim, K. S., 1980. Threshold autoregression, limit cycles and cyclical data. Journal of the Royal Statistical Society (Series B), 42(3), 245-292.

Toussaint, G. (1974) Bibliography on estimation of misclassification. IEEE Transactions on Information Theory, 20(4), 472–479.

Tresp, V., Neuneier, R., Zimmermann, H. G. (1997) Early brain damage. In M. Mozer, M. I. Jordan, and T. Petsche, editors, Advances in Neural Information Processing Systems 9. Cambridge, MA: MIT Press, 669 - 675

Tsonis, A. A., 2001. Probing the linearity and nonlinearity in the transitions of the atmospheric circulation. Nonlinear Processes in Geophysics 8, 341–345.

Tsonis, A. A., Elsner, J. B., Georgakakos, K. P., 1993. Estimating the dimension of weather and climate attractors: Important issues about the procedure and interpretation, J. Atmos. Sci. 50(15), 2549–2555.

Tucci, C.E.M., Clarke, R.T., Collischonn, W., da Silva Dias, P.L., de Oliveira. G.S., 2003. Long-term flow forecasts based on climate and hydrological modeling: Uruguay River basin. Water Resour. Res., 39(7), SWC3-1-11.

Uvo, C.B., Graham, N.E., 1998. Seasonal runoff forecast for northern South America: A statistical model. Water Resour. Res., 34(12), 3515-3524.

Uvo, C.B., Tolle, U., Berndtsson, R., 2000. Forecasting discharge in Amazonia using artificial neural networks, Int. J. Climatol., 20, 1495– 1507.

van Belle, G., Hughes, J. P., 1984. Nonparametric tests for trend in water quality. Water Resources Research, 20(1), 127-136.

Vapnik, V., 1982. Estimation of dependence based on empirical data. New York: Springer-Verlag

Vecchia, A.V., Ballerini, R., 1991. Testing for periodic autocorrelations in seasonal time series data, Biometrika, 78, 53-63.

Vecchia, A.V., 1985. Maximum Likelihood Estimation for periodic autoregressive moving average models. Technometrics, 27, 375-384.

von Storch, H., 1995. Misuses of statistical analysis in climate research, In: Analysis of Climate Variability: Applications of Statistical Techniques, edited by H. V. Storch and A. Navarra, Springer-Verlag, New York, pp. 11– 26.

Wang, C.T., Gupta, V.K., Waymire, E., 1981. A geomorphologic synthesis of nonlinearity in surface runoff. Water Resour. Res., 19(3), 545-554.

Wang, W., Van Gelder, P.H.A.J.M., Vrijling, J.K., Ma, J., 2004. Predictability of streamflow processes of the Yellow River. Proceedings of the 6th International Conference on Hydroinformatics. Singapore : World Scientific, 1261-1268.

Wang, W., van Gelder, P.H.A.J.M., Vrijling, J.K., 2005a. Constructing prediction interval for monthly streamflow forecasts. In: Proceedings of the International Symposium on Stochastic Hydraulics, May 23-24, 2005, Nijmegen, Netherlands.

Wang, W., Van Gelder, P.H.A.J.M., Vrijling, J.K., 2005b. Some Issues About the Generalization of Neural Networks for Time Series Prediction, In: Artificial Neural Networks: Formal Models And Their Applications, W. Duch et al. (Eds.), Lecture Notes in Computer Science, vol.3697, 559-564.

Water Cycle Study Group (WCSG)., 2001. A Plan for a New Science Initiative on the Global Water Cycle. Report to the U.S. Global Change Research Program, Washington, D.C.

Water Master Plan Project, 1983. Multi-lead forecasting of the River Nile streamflows. Technical Report 21 (UNDP/EGY/81/031), Ministry of Irrigation, Cairo, Egypt.

Weigend, A. S., Rumelhart,D. E., Huberman, B. A. (1991) Generalization by weight-elimination with application to forecasting, In: Advances in Neural Information Processing Systems 3, R. Lippmann, J. Moody, and D. Touretzky, Eds. San Mateo, CA: Morgan Kaufmann, 875–882.

Weinberg, C.B., 1986. Arts plan: implementation, evolution, and usage. Marketing Science, 5, 143-58.

Weiss, A.A., 1984. ARMA models with ARCH errors. J. Time Series Anal., 5, 129-143.

Werner, K., Brandon D., Clark M.P., Gangopadhyay S., 2005. Incorporating Medium-Range Numerical Weather Model Output into the Ensemble Streamflow Prediction System of the National Weather Service. Journal of Hydrometeorology, 6, 101-114.

Whitaker, D.W., Wasimi, S.A., Islam, S., 2001. The El Nino-Southern Oscillation and long-range forecasting of flows in the Ganges. Int. J. Climatol., 21(1), 77-87.

Wilcox, B. P., Rawls, W. J., Brakensiek, D. L., Wight J. R., 1990. Predicting runoff from rangeland catchments: A comparison of two models, Water Resources Research, 26, 2401–2410.

Wilcox, B.P., Seyfried, M.S., Matison, T.H., 1991. Searching for chaotic dynamics in snowmelt runoff. Water Resour Res., 27(6), 1005-1010.

Winkler, R.L., 1981, Combining probability distributions from dependent information sources. Management Science 27, 479–488.

Winkler, R.L., Makridakis, S., 1983, The combination of forecasts, Journal of the Royal Statistical Society, Series A, 146, 150-157.

WMO, 1975. Intercomparison of conceptual models used in operational hydrological forecasting. Operational Hydrology Report No. 7, WMO, Geneva, Switzerland.

WMO, 1986. Intercomparison of Models of Snowmelt Runoff," Operational Hydrology Report No. 23, WMO, Geneva, Switzerland.

WMO, 1992, Simulated real-time intercomparison of hydrological models. Operational Hydrology Report No. 38, WMO, Geneva, Switzerland.

Wolf, A., Swift, J. B., Swinney, H. L., Vastano, A., 1985. Determining Lyapunov exponents from a time series, Physica D, 16, 285–317.

Wood A. W., E. P. Maurer, A. Kumar, and D. Lettenmaier, 2002. Long-range experimental hydrological forecasting for the eastern United States, J. Geophys. Res., 107 (D20), 4429, doi:10.1029/2001JD000659.

Xiong, L., Shamseldin, A., O'Connor, K.M. (2001), A non-linear combination of the forecasts of rainfall–runoff models by the first-order Takagi–Sugeno fuzzy system. J. Hydrol., 245, 196–217.

Yakowitz, S., Karlsson, M., 1987. Neareast neighbour methods for time series with application to rainfall-runoff prediction. In: J.B. MacNeil and G.J. Umphrey (eds.), Stochastic Hydrology, D. Reidel, Dordrecht, 149-160.

Yang, T.C., Yu P.S., Chen C.C., 2005. Long-term runoff forecasting by combining hydrological models and meteorological records. Hydrol. Process., 19, 1967-1981.

Yawson D.K., Kongo V.M., Kachroo R.K., 2005. Application of linear and nonlinear techniques in river flow forecasting in the Kilombero River basin, Tanzania. Hydrological Sciences Journal, 50(5), 783-796.

Yeh, W.W-G., Becker, L., Zettlemoyer, R., 1982. Worth of Inflow Forecast for Reservoir Operation. Journal of Water Resources Planning and Management Division, 108-WR3, 257-269.

Yurekli K., Kurung A., Ozturk F., 2005. Testing the Residuals of an ARIMA Model on the Cekerek Stream Watershed in Turkey. Turkish J. Eng. Env. Sci., 29, 61-74.

Zadeh, L.A., 1965. Fuzzy sets, Inform. Control Systems. 8, 338–353.

Zealand, C.M., Burn, D.H., Simonovic, S.P., 1999. Short term streamflow forecasting using artificial neural networks. J. Hydrol., 214, 32–48.

Zhang, B., Govindaraju, R.S., 2000. Prediction of watershed runoff using Bayesian concepts and modular neural networks. Water Res. Res., 36(3), 753-762.

Zhang, X., Harvey, K. D., Hogg, W. D., Yuzyk, T. R., 2001. Trends in Canadian streamflow, Water Resour. Res., 37(4), 987–998.

Zhao, R. J., Liu X. R., 1995. The Xinanjiang model. In: Singh, V.P. (ed.) Computer Models of Watershed Hydrology. Water Resources Publications, 215-232.

Zhu, M-L., Fujita, M., Hashimoto, N., Kudo, M., 1994. Long lead time forecast of runoff using fuzzy reasoning method. Journal of the Japanese Society of Hydrology and Water Resources, 7(2), 83–89.

Zivot, E., Wang, J., 2002. Modelling Financial Time Series with S-Plus. New York: Springer Verlag.

Zuzel, J.F., Robertson, D.L., Rawls, W.J., 1975. Optimizing long-term streamflow forecasts. Journal of soil and water conservation, 30(2), 76-78.

Acknowledgements

This thesis is the result of the work that was carried out mainly at the Section of Hydraulic Engineering of Delft University of Technology (TU Delft), where I have benefited a lot from the inspiring and friendly atmospheres.

First of all, I would like to thank my promoter Professor J.K. Vrijling for giving me the opportunity to do this PhD-research, giving me the freedom to try out new ways, and giving me continuous support and stimulating advises. I am also very grateful to Dr. Pieter H.A.J.M. van Gelder who as my supervisor provided constructive comments and constant support during my PhD research.

I would especially like to thank my Chinese supervisor Prof. Liu Xinren from Hohai University whose guidance and encouragement helped me a lot for doing the study. Unfortunately, she cannot attend my defence ceremony due to health consideration.

Many thanks go to the examination committee memmbers for reviewing my thesis, especially to Prof. Z.W. Kundzewicz and Prof. Y.F. Chen for giving many constructive comments.

I extend my sincere gratitude and appreciation to many people who made this thesis possible. I am indebted to the late Professor M.J. Hall from UNESCO-IHE, who gave me many suggestions during the initial phase of my PhD research. Special thanks are due to Dr. D.P. Solomatine from UNESCO-IHE for his stimulating courses on data-driven modeling. I am grateful to Mr. Ma Jun for his constant encouragement and his generous provision of data. The gratitude also goes to Prof. Liu Xuegong for providing a nice picture for the cover of my thesis. I am especially grateful to Ms. M.A. Burgmeijer for translating the summary.

The PhD program is a part of a China-Dutch cooperation project on "Strengthening of education and applied research in water resources engineering and water resources management at a selected number of institutes in the P. R. of China (China-DC/WRE)", funded by the Netherlands Directorate General for International Cooperation through the Royal Netherlands Embassy in Beijing.The financial supports from the project and from the Section of Hydraulic Engineering of TU Delft are greatly appreciated. Many thanks go to CICAT of TU Delft for the management of the project, especially to Ms. Theda Olsder for her kind help during my PhD period.

So much fun and good times I have shared with my friends in Delft. I am thankful to all of them and I wish them all the best.

Especially, I am very grateful to my wife Li Hong, for her continuous support and encouragement during the PhD period. I would say sorry to my lovely son, Ziyi, for having not spent much time staying with him.

Finally, I would like to express my special gratitude to my parents who take care of my son during my absence.

Wang Wen

April 10, 2006

About the Author

Wen Wang was born in Jiangyan, Jiangsu, China, on December 16, 1967. He started his high education at Nanjing University in China in 1985 and got the Bachelor of Science degree in geomorphology and Quaternary geology in 1989. In 1991, He started his MSc-study at Nanjing University. After he obtained the Master of Science degree in geomorphology and Quaternary environment in 1994, he got a job in Hohai University, working there and starting his academic research career from then on. He is mainly engaged in the research on physical geography, and the application of geographic information system techniques and remote sensing techniques in the field of hydrology. In 1998, he started his PhD-study in the Nanjing Institute of Geography & Limnology, Chinese Academy of Sciences (NIGLAS), and got his first PhD title in physical geography in 2003. In February 2001 he was selected by Hohai University to start his sandwich PhD-research at the Section of Hydraulic Engineering of Delft University of Technology on streamflow modeling mainly based on time series analysis techniques. Presently, he is an associate professor in the State Key Laboratory of Hydrology-Water Resources and Hydraulic Engineering of Hohai University.

Publications (first-author papers in the last 5 years):

1. Wang W., Xie Z.R., Sea-Level Change in China based on Chinese historical documents. Advance in Earth Sciences, 2001, 16(2), 272-278. (in Chinese, with English abstract)
2. Wang W., van Gelder P.H.A.J.M., Vrijling J.K., Stochastic properties and preliminary forecasting of daily flow of the Yellow River at Tangnaihai and Tongguan. In: Proceedings of the Second International Symposium On Flood Defence. Beijing: Science Press, 2002, pp. 922-929.
3. Wang W., Xu W.C., Analysis of predictable time length of chaotic time series. Journal of Hohai University, 2004, 32(4), 367-371. (in Chinese, with English abstract)
4. Wang W., van Gelder P.H.A.J.M., Vrijling J.K., Periodic autoregressive models applied to daily streamflow. In: Proceedings of the 6th International Conference on Hydroinformatics, Singapore: World Scientific, 2004, pp. 1334-1341.
5. Wang W., van Gelder P.H.A.J.M., Vrijling J.K., Ma J., Predictability of streamflow processes of the Yellow River. In: Proceedings of the 6th International Conference on Hydroinformatics. Singapore: World Scientific, 2004, pp. 1261-1268.
6. Wang W., van Gelder P.H.A.J.M., Vrijling J.K., Applying periodic ANN model to daily streamflow forecasting. In: Proceedings of the 6th International Conference on Recent Advances in Soft Computing, Nottingham Trent University, 2004, pp. 624-629. (on CD-ROM)

7. Wang W., Ma J., Review on some methods for hydrologic forecasting. Advances in Science and Technique of Water Resources, 2005, 25(1), 56-60 (in Chinese, with English abstract).
8. Wang W., Xu W.C., Some issues in the characterization of chaotic properties of hydrologic time series. Advances in Water Sciences, 2005, 16(4), 609-616. (in Chinese, with English abstract).
9. Wang W., van Gelder P.H.A.J.M., Vrijling J.K., Ma J., Testing and modelling autoregressive conditional heteroskedasticity of streamflow processes. Nonlinear Processes in Geophysics, 2005, 12, 55-66.
10. Wang W., van Gelder P.H.A.J.M., Vrijling J.K., Long-memory properties of streamflow processes of the Yellow River. Proceedings of the International Conference on Water Economics, Statistics and Finance, vol. 1, Rethymno-Crete, Greece, 8-10 July, 2005, pp. 481-490.
11. Wang W., van Gelder P.H.A.J.M., Vrijling J.K., Trend and stationarity analysis for streamflow processes of rivers in the western Europe. Proceedings of the the International Conference on Water Economics, Statistics and Finance, vol. 1, Rethymno-Crete, Greece, 8-10 July, 2005, pp. 451-161.
12. Wang W., van Gelder P.H.A.J.M., Vrijling J.K., Constructing prediction interval for monthly streamflow forecasts. Proceedings of the International Symposium on Stochastic Hydraulics, Nijmegen, the Netherlands, 25-27 May, 2005. (on CD-ROM)
13. Wang W., van Gelder P.H.A.J.M., Vrijling J.K., Is the streamflow process chaotic? Proceedings of the International Symposium on Stochastic Hydraulics, Nijmegen, the Netherlands, 25-27 May, 2005. (on CD-ROM)
14. Wang W., van Gelder P.H.A.J.M., Vrijling J.K., Some Issues About the Generalization of Neural Networks for Time Series Prediction, In: Artificial Neural Networks: Formal Models And Their Applications, W. Duch, J. Kacprzyk, E. Oja and S. Zadrozny, (Eds.), Lecture Notes in Computer Science, 2005, vol.3697, pp. 559-564.
15. Wang W., van Gelder P.H.A.J.M., Vrijling J.K., Improve daily streamflow forecasts by combining ARMA and ANN Models, In: Proceedings of the Innovation, Advances and Implementation of Flood Forecasting technology conference, 17-19 October 2005, Tromso, Norway. (on CD-ROM)
16. Wang W., van Gelder P.H.A.J.M., Vrijling J.K., Detection of changes in streamflow series in western Europe over 1901–2000, Water Science & Technology: Water Supply, 2005, 5(6), 289-299.
17. Wang W., Vrijling J.K., van Gelder P.H.A.J.M., Ma J., Testing for nonlinearity of streamflow processes at different timescales. Journal of Hydrology. (in press)
18. Wang W., van Gelder P.H.A.J.M., Vrijling J.K., Ma J., Forecasting Daily Streamflow Using Hybrid ANN Models. Journal of Hydrology. (in press)
19. Wang W., van Gelder P.H.A.J.M., Vrijling J.K., Ma J., Comparing Bayesian regularization and cross-validated early-stopping for streamflow forecasting with ANN models. In: Proceedings of the International Symposium on the Methodology in hydrology, Nanjing, China, 30 October – 1 November, 2005, IAHS Red Book. (in press)
20. Wang W., Liu X.G., Environment evolution of Lake Wuliangsuhai since 1980 and the measures of ecological restoration. In Proceedings of the 9th Inter-Regional Conference on Environment-Water, Delft, the Netherlands, 17-19 May 2006. (in press)