2005

Flood Forecasting Using Time Series Data Mining

Chaitanya Damle
University of South Florida

Flood Forecasting Using Time Series Data Mining

by

Chaitanya Damle

A thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Science in Industrial Engineering
Department of Industrial and Management Systems Engineering
College of Engineering
University of South Florida

Major Professor: Ali Yalcin, Ph.D.
Bruce Lindsey, Ph.D.
José L. Zayas-Castro, Ph.D.

Date of Approval:
April 1, 2005

Keywords: phase space reconstruction, flood prediction, nonlinear time series prediction, clustering, genetic algorithm, event prediction

Table of Contents

List of Tables

List of Figures

Flood Forecasting using Time Series Data Mining

Chaitanya Damle

ABSTRACT

Earthquakes, floods, rainfall represent a class of nonlinear systems termed chaotic, in which the relationships between variables in a system are dynamic and disproportionate, however completely deterministic. Classical linear time series models have proved inadequate in analysis and prediction of complex geophysical phenomena. Nonlinear approaches such as Artificial Neural Networks, Hidden Markov Models and Nonlinear Prediction are useful in forecasting of daily discharge values in a river. The focus of these methods is on forecasting magnitudes of future discharge values and not the prediction of floods. Chaos theory provides a structured explanation for irregular behavior and anomalies in systems that are not inherently stochastic. Time Series Data Mining methodology combines chaos theory and data mining to characterize and predict complex, nonperiodic and chaotic time series. Time Series Data Mining focuses on the prediction of events. Floods constitute the events in a river daily discharge time series. This research focuses on application of the Time Series Data Mining to prediction of floods.

Chapter 1. Introduction

This research is an application of Time Series Data Mining methodology to prediction of floods. Chapter 1 is an introduction to effects of floods, nature of geophysical phenomena, existing flood forecasting techniques, the Time Series Data Mining approach and its application to flood forecasting.

1.1 Effects of Floods

According to the United States Geological Survey (USGS) and National Weather Service (NWS), as much as 90 percent of the damage related to natural disasters (excluding droughts) is caused by floods and associated mud and debris flows. Over the last 10 years, floods have cost on average, $3.1 billion annually in damages. USGS and NWS estimate that more than 95 lives are lost, on average, per year and proper detection and prediction of these disasters can save countless lives and over 1 billion dollars a year in damages [34].

1.2 Nature of Geophysical Systems

Traditionally, geophysical systems are viewed as systems that exhibit irregular behavior, essentially due to the large number of variables that govern and dominate the underlying systems [44]. Geophysical phenomena such as earthquakes, floods etc represent nonlinear systems whose occurrences are subject to high levels of uncertainty and unpredictability. The irregularity of these systems is not a transient phenomenon, but an intrinsic property [17]. A few deterministic approaches have been applied, but the stochastic approaches have proved better in the process of

representing important statistical characteristics of the geophysical system and provide reasonably good predictions [44]. Nonlinear dynamics or chaos addresses the set of systems that may be stochastic but also display correlations that are deterministic, and are known as “deterministic chaotic systems”. Deterministic chaos provides a structured explanation for irregular behavior and anomalies in systems which do not seem to be inherently stochastic [25]. Thus, chaotic systems are treated as "slightly predictable" and can be studied in the framework of nonlinear system dynamics. This fact has sparked a surge of interest in nonlinear models among researchers in applied sciences.

1.3 Existing Flood Forecasting Techniques

Flood prediction is a complex process because of the numerous factors that affect river water levels such as the location, rainfall, soil types and size of catchments. The relationship between these factors has not been fully understood. Classical linear Gaussian time series (deterministic) models are inadequate in analysis and prediction of complex geophysical phenomena. Linear methods such as *ARIMA* approach are unable to identify complex characteristics due to the goal of characterizing all time series observations, the necessity of time series stationarity and, the requirement of normality and independence of residuals [41]. Nonlinear time series approaches such as Hidden Markov Models (HMM) [3], Artificial Neural Networks (ANN) [5] and Nonlinear Prediction (NLP) [28], applied to discharge forecasting produce accurate predictions for short prediction periods of up to one day.

Because the world of nonlinear models is so vast, much attention has been devoted to particular families of models, which have been found to perform well in a range of applications. *Time delayed embedding* is one such technique that has been applied to a variety of physical applications in the domains of physiology [53], economics [12, 13, 40], geophysics [9, 24, 36, 44,

50, 51] and engineering applications [4, 42]. The *Time Series Data Mining* methodology [41] is based on a variant of *time delayed embedding* called the *reconstruction of phase space*.

1.4 Time Series Data Mining

The *Time Series Data Mining* (TSDM) methodology [41], proposed by Richard Povinelli follows the time delayed embedding process to predict future occurrences of important events. TSDM framework combines the methods of phase space reconstruction and data mining to reveal hidden patterns predictive of future events in nonlinear, nonstationary time series. TSDM has its theoretical justification in the theory of nonlinear dynamics, specifically the *Takens' Embedding Theorem* and *Sauer's Theorem* [41]. The TSDM methodology is explained in detail in Chapter 3. The general block diagram of TSDM methodology is shown in Figure 1.1.

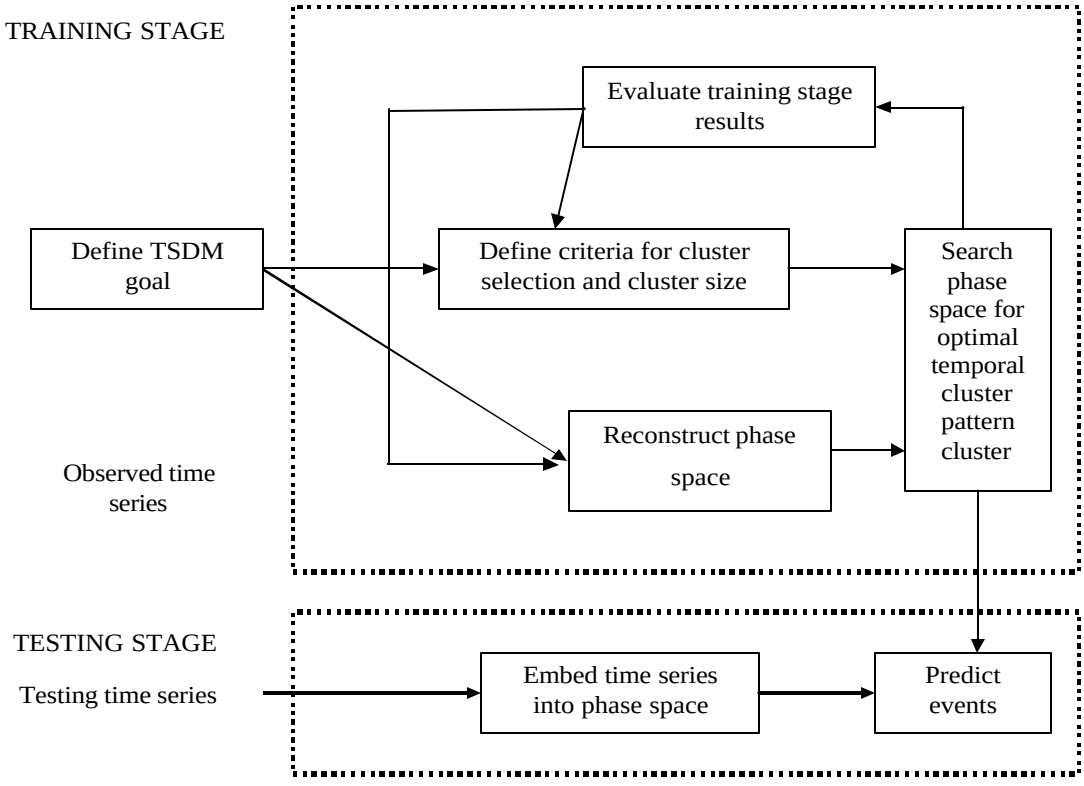

Figure 1.1 Block Diagram of TSDM Methodology [41]

TSDM methodology has been applied to the areas of earthquake prediction based on seismic activity, stock market predictions and to manufacturing applications such as predicting release of welding droplets [41].

Phase space reconstruction provides a simplified, multidimensional representation of a nonlinear time series that facilitates further analysis. An example of nonlinear time series and its reconstructed phase space is shown in Figures 1.2 and 1.3 respectively. The details of phase space reconstruction and the calculation of embedding parameters are explained later.

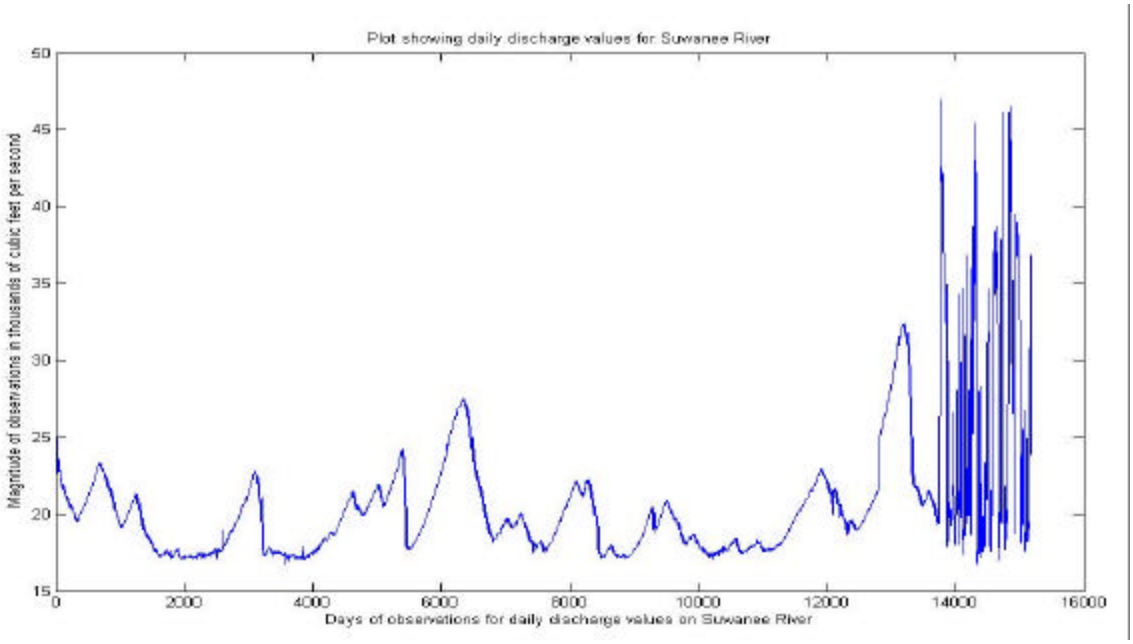

Figure 1.2 Time Series Representing Daily Discharge of Suwanee River

1.5 Flood Forecasting Using Time Series Data Mining

Motivation

Nonlinear approaches such as the HMM, ANN and NLP have been applied to the area of flood forecasting, with the prediction results decreasing in accuracy as the prediction periods exceed a day. Moreover, HMM, ANN and NLP are useful in the forecasting of all future values, and their accuracy is measured over all forecasted values and not for the accuracy of predicting

events such as floods. The prediction of floods requires a technique that can predict events (floods) in particular.

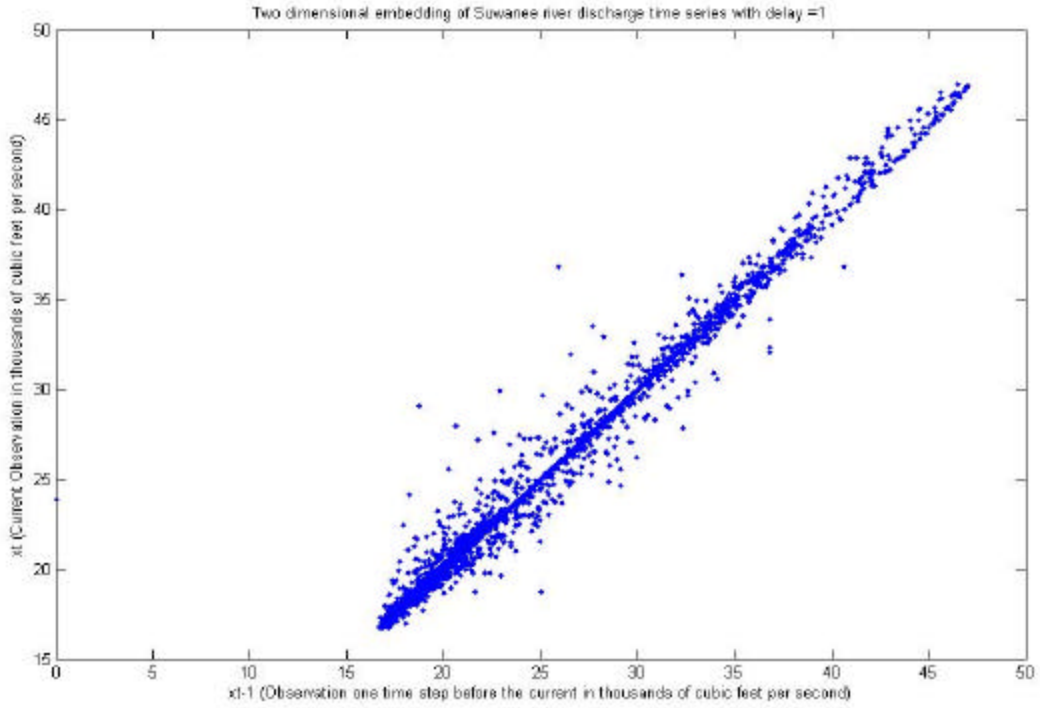

Figure 1.3 Reconstructed Phase Space with t = 1 in Two Dimensions

This is where the event based data mining approach of TSDM is useful because it focuses on the prediction of events (floods). The prediction accuracy can then be measured by how successful the methodology is in predicting an event (flood) rather than the whole series of forecasted values. The TSDM methodology will be applied in this research to flood prediction at the three gauging stations and, the accuracy in predicting floods over short and long prediction periods will be measured.

Chapter 2 presents a literature review enlisting previous work done in discharge forecasting, flood prediction, time series predictions, and nonlinear prediction techniques. In Chapter 3 the TSDM methodology is reviewed. In Chapter 4 TSDM is applied to flood prediction problem. Three different gauging stations are considered and the results are presented. Chapter 5 includes the conclusion of this research and the directions for future research.

Chapter 2. Literature Review

This chapter includes an introduction to presence of chaos in geophysical systems, followed by the classification and a brief review of existing flood prediction techniques along with their advantages and disadvantages. This is followed by a review of application of Non Linear Time Series Analysis to flood forecasting.

2.1 Presence of Chaos in Geophysical Systems

It is necessary to confirm the presence of chaos in any system before applying techniques based on chaos theory. It has been observed that the application of chaos theory based methods to systems that are not chaotic may produce wrong results [25]. Additionally, if there is no proof of existence of chaos, other methods targeted towards deterministic or stochastic time series analysis can be applied with greater success. Sivakumar [44] presents a detailed literature review on application of Chaos Theory in geophysics with applications to rainfall, river flow, rainfall runoff, sediment transport, temperature, pressure, wind velocity, wave amplitudes, sunshine duration and tree rings. M.N. Islam and B. Sivakumar [23] have performed a comparative study of existing techniques to determine the presence of chaos in hydrological systems. Each technique has its own advantages and disadvantages and no single method guarantees a correct classification of a system as being chaotic or non chaotic. Although proof for existence of chaos in different river flow time series has been illustrated in [23, 25, 36, 50, 51], its presence in the river discharge time series needs to be confirmed for the three examples considered.

2.2 Existing Flood Forecasting Techniques

In recent years, numerous studies from varied fields of hydrodynamics, civil engineering, statistics and data mining have contributed to the area of flood prediction. Some of the existing techniques used in flood prediction are:

1. Stream and Rain-Gauge Networks and Hydrograph Analysis
2. Radar and Information Systems
3. Linear Statistical Models and
4. Nonlinear Time Series Analysis and Prediction.

The first three techniques mentioned above use more than one input parameter (multivariate) for characterization and prediction of floods. For example one of the linear statistical models uses the flood discharge, weighted flood discharge, precipitation intensity, elevations, stream length, and main channel slope for flood prediction.

Some of the nonlinear time series approaches such as Hidden Markov Models [3] and Artificial Neural Networks [5] are also based on multiple time series. Nonlinear Prediction (NLP) [28] method developed by Farmer and Sidorowich [14] has been used in river discharge forecasting by Porporato et al [36-38]. Researchers have experimented with the application of NLP to discharge forecasting based on both single variable time series [45-46] and, multiple variable time series [8, 37]. Laio et al [28] have performed a comparison of ANN and NLP approaches in daily discharge forecasting. The results have shown that the NLP method provides accurate forecasts over a shorter prediction period (1-6 hours), but over prediction periods exceeding 24 hours, the ANN approach is more accurate. However, for periods exceeding a day, the HMM, ANN and NLP methods lose their accuracy. Moreover, the HMM, ANN and NLP techniques are not adequate for event predictions for reasons explained in Section 1.5. These approaches are described in the following sections.

2.2.1 Stream and Rain-Gauge Networks and Hydrograph Analysis

River flows and precipitation volumes are measured and monitored by more than 7,300 gauge stations operated mainly by USGS out of which about 4,200 are telemetries by an earth satellite based communication system. There also exist more than 14,000 rain gauges operated by NWS [34]. Data from these two recording systems acts as an input to statistical hydraulic models that estimate the possible river stage and discharge that may result. The models often turn out to be inaccurate because they are built using historic data that hasn't been recorded for more than the past 25 years and changes in topography as a result of rapid urbanization. Since hydraulic models cannot predict exactly what will happen to the river, rating curves or hydrographs that show the relationship between water flows and water levels are used simultaneously [34]. The disadvantage of using a hydrograph is that it does not consider the changes in river cross sections that result from changes in channel bed and as a result, the stage and discharge relationship is altered [34].

2.2.2 Radar and Information Systems

With the development of Advanced Hydrological Prediction Services (AHPS) and NEXRAD by the NWS, the Doppler Radar and the Geographic Information System (GIS) is being used along with the traditional hydraulic models for improved flood forecasting [34]. This is complex system comprising of simulation programs and uses data from various sources such as telemetry, automated gauges to calculate runoff, infiltration and precipitation volumes using land use and elevation information. AHPS has been found to result in flood forecasts that are 20% more accurate than the stream and rain gauge analysis. NWS has implemented 478 AHPS forecast points by the year 2004 with a one-time fee of $2.1 million and $300,000 annually for maintenance [34]. The major disadvantages of AHPS are the complexity and the high cost of implementation and maintenance.

2.2.3 Linear Statistical Models

Linear Statistical Models such as *Autocorrelation functions*, *Spectral Analysis*, *Analysis of cross correlations*, *Linear Regression* and *Autoregressive Integrated Moving Average (ARIMA)* have been studied for the applicability to flood forecasting. Solomatine et al. [46] have found in their study that the use of stationary *(ARMA*) as well as non stationary *(ARIMA)* versions of linear prediction techniques does not provide accurate predictions. Application of other linear stochastic methods has also resulted in inaccurate predictions, clearly indicating that linear statistical models do not accurately represent historical data and hence are not acceptable methods for a non-linear application such as flood forecasting [46].

2.3 Nonlinear Time Series Analysis and Pre diction Applied to Flood Forecasting

2.3.1 Hidden Markov Models

The concept of the state of a system is powerful even for nondeterministic systems. A Markov chain consists of a continuous range of flow values and, given the transition probability of moving from one state to another, will predict the most probable future state based on the current state. The objective of using *Hidden Markov Models* to predict floods is to provide a simpler, generalized data mining model that could be reused for various geographical areas, in which independence of predictions could be obtained with minimal consideration of past events. Most of the applications of Hidden Markov Models in flood forecasting have used the spatio-temporal approach, whereas the time series prediction used in this research is a purely temporal approach [3, 29]. The drawbacks of this approach are that the initial structure of the Markov model may not be certain at the time of model construction and it is very difficult to change the transition probabilities as the model itself changes with time. It was also observed that the Hidden Markov Models have a higher error for longer prediction periods as well as for

prediction of events with sudden occurrences such as flash floods, leading to the conclusion that Hidden Markov Models do not perform better than other data mining techniques [3].

2.3.2 Artificial Neural Networks

Artificial Neural Networks (ANN) are widely accepted as a potentially useful way of modeling complex nonlinear and dynamic systems. They are particularly useful in situations where the underlying physical process relationships are not fully understood or where the nature of the event being modeled (i.e. a flood) may display chaotic properties. Though neural networks do not remove the need for knowledge or prior information about the systems of interest, they reduce the model's reliance on this prior information. This removes the need for an exact specification of the precise functional form of the relationship that the model seeks to represent. Artificial Neural Networks represent input output "connectionist models" where different factors such as temperature, precipitation, flow rate, depth etc are provided as input to the model. This technique has been used by a number of researchers in a variety of geophysical phenomena from predicting currents in sea to flood prediction [5, 10-11, 20, 22, 52-53]. There is however, no set algorithm that can be applied to ensure that the network will always yield an optimal solution as opposed to a local minimum value, and the nonlinear nature of the ANN often results in multiple predicted values [27].

2.3.3 Non Linear Prediction Method

Non Linear Prediction (NLP) method was developed by J. Doyne Farmer and John J. Sidorowich [14] and was subsequently used in flood prediction by Porporato et al [36-38]. Researchers have experimented with flood prediction based on single parameter (river flow) time series [45-46] and multiple variable time series as well [8, 37]. The first steps of NLP are same as that of Time Series Data Mining (TSDM) methodology used in this research, starting with the

reconstruction of phase space from the measured time series of the variable to be forecast. The phase space reconstruction and Takens theorem which provides theoretical justification for phase space reconstruction are explained next.

Phase Space Reconstruction

Attractors are the states towards which a system evolves when starting from certain initial conditions. Since the dynamic of the system is unknown, the original theoretical attractor that gives rise to the observed time series cannot be constructed. Instead, a phase space is created where the attractor is reconstructed from the scalar observed data that preserves the invariant characteristics of the original unknown attractor described by the time delay method to approximate the state space from a single time series data.

The *reconstructed phase space* is a Q-dimensional metric space into which a time series is embedded. It is a vector space for the system such that specifying a point in this space specifies the state of the system and vice versa. Time-delayed embedding maps a set of Q time series observations taken from X onto $x_{t,}$ where x_t is a vector or point in phase space. A time series is represented by $\{x_{t-(Q-1)t},\ldots, x_{t-2t}, x_{t-t}, x_t\}$ where x_t represents the current observation, and $(x_{t-(Q-1)t},\ldots, x_{t-2},, x_{t-t})$ are the past observations. If t is the current time index, then t-t is a time index in the past, and t+t is a time index in the future. The embedding delay (t) is the time difference in number of time units between adjacent components of delay vectors. The embedding dimension (m) is the number of dimensions of reconstructed phase space that are required to achieve an embedding. Any further analysis of deterministic properties of a nonlinear time series depends on the precondition of a successful reconstruction of a state space of the underlying process [17].

There are many theorems based on time delayed embedding used in reconstruction of phase space such as the *Takens Embedding Theorem* [44] and *Whitney's embedding Theorem* [46]. Other methodologies that are not based on the method of time delays are Filtered Embedding [40] that comprise of Principal Components [17] and Derivatives and Legendre coordinates [17].

Takens Embedding Theorem

Takens theorem gives the theoretical justification for the reconstruction of phase space. It states that when a system has a state space M that is Q dimensional, $j : M \to M$ be a map that describes the dynamics of the system, and $y : M \to R$ its twice-differentiable function (function that is differentiable and has a differentiable derivative), which represents the observation of a single state variable. Then the map $?_{(f,y):} M? R^{2Q+1}$ defined by

$$\Phi_{(j,y)}(x) = \left(y(x), y(j(x)),..., y\left(j^{2Q}(x)\right)\right) \tag{2.1}$$

is an embedding which is a mapping that retains the structure of the original attractor from one topological space to another. Takens Theorem guarantees that the reconstructed dynamics are topologically identical to the true dynamics of the system.

According to the Takens Embedding Theorem, the selection of any value for the delay (t) will result in an embedding, given the fact that the data is infinitesimally accurate and does not contain any noise [17, 23]. The data collected from naturally occurring dynamical systems hardly matches these specifications.

In regards to the embedding dimension (*m*), an accurate value of *m* is only required when we want to exploit determinism with minimal computational effort. If an *m*-dimensional embedding yields a faithful representation of state space, every m`-dimensional reconstruction with $m` > m$ does so as well [23]. Selecting a large value of *m* for chaotic data adds redundancy, degrades the performance of prediction algorithms and increases the computation time [23].

Many algorithms have been proposed that provide an estimation of optimal embedding dimension and time delay. Some of the methods for estimation of embedding dimension are the method of *False Nearest Neighbors* [24], *The Fillfactor algorithm* [6, 7], and *The Integral Local Deformation algorithm* [6, 7]. Selection of method for estimation of m and t is largely dependant on the application and the kind of analysis that is to be performed.

Methods of Estimating t (Time Delay)

A variety of methods have been proposed for approximation of ideal time delay to ensure an embedding. Some of these techniques are *autocorrelation, power spectrum functions*, *average mutual information (AMI) function*, *degree of separation function* and *Lyapunov exponents* [1, 17, 25]. In this research, the AMI [16] and the autocorrelation function [25] have been used to approximate the time delay. For the AMI method, first, a histogram is created from the existing time series for determining the probability distribution of the data. p_i is the probability that the signal assumes a value inside the *i*th bin of the histogram, and $p_{ij}(t)$ is the probability that $s(t)$ is in bin I and $s(t + t)$ is in bin j. The AMI function is based on *Shannon's entropy* [16], which is a measure of uncertainty and is represented by the expression

$$I(t) = \sum_{i,j} p_{ij}(t) \ln p_{ij}(t) - 2\sum_{i} p_i \ln p_i \qquad (2.2)$$

The first minimum of $I(t)$ marks the delay where $a(t=t)$ adds maximum information to the knowledge we have from $s(t)$, i.e. the redundancy is minimum. However, the AMI function is only applicable to two-dimensional embeddings [25]. The AMI function generated from the Suwanee daily water level recording time series is shown in Figure 2.1.

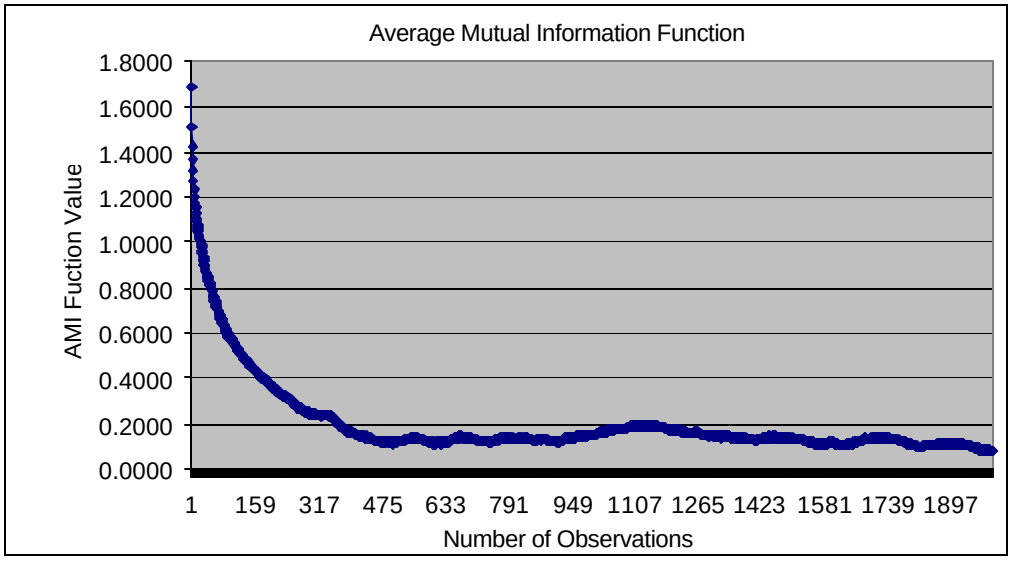

Figure 2.1 Average Mutual Information Function for Suwanee River Time Series

Autocorrelation function is a method used to find delay dimension for higher dimensional embeddings. Given a time series x_t, t = 0, 1, … (N-1), its correlation can be measured by checking whether the product of x_t and x_{t+t} for a given delay t differs significantly from zero on average over the whole time series [17]. The autocorrelation function is given by

$$A(t) = \frac{1}{N - t - 1} \sum_{t=0}^{N-t-1} x_t x_t + t \tag{2.3}$$

In principle the autocorrelation function should be zero at all lags equal or larger than t, however selecting a delay higher than t leads to completely uncorrelated elements in the phase space, which is undesirable. Therefore, a reasonable choice of delay is the time index where the autocorrelation function decays to zero or first minimum [25]. The accurate calculation of time delay is important for the NLP technique since it tries to exploit the predictive structure of the phase space.

Method for Estimating the Embedding Dimension (m)

False nearest neighbors [26], *The Fillfactor Algorithm* [6], *Integral Local Deformation Algorithm* [6] are some of the methods used for estimating the phase space dimension, with the most popular being the method of false nearest neighbors. The basic concept behind the method of false nearest neighbors is that if a reconstructed phase space does not have enough dimensions, it cannot be described as an embedding. Suppose a phase space has been reconstructed with dimension *m*. If this *m* is less than the optimal value, the points which in true state space are separated by large distances will be mapped as neighbors in the reconstructed phase space and hence are called false nearest neighbors. The False Nearest Neighbor algorithm by Kennel et al. [26] identifies the points that move the farthest in terms of Euclidean distance when plotted in a dimension higher than the present one and gives the percentage of false nearest neighbors. As we move into higher dimensions, the attractor is unfolded and, a true embedding is achieved when all

the false nearest neighbors are removed. An example showing the reduction in fraction of false nearest neighbors with the increase in embedding dimension is shown in Figure 2.2.

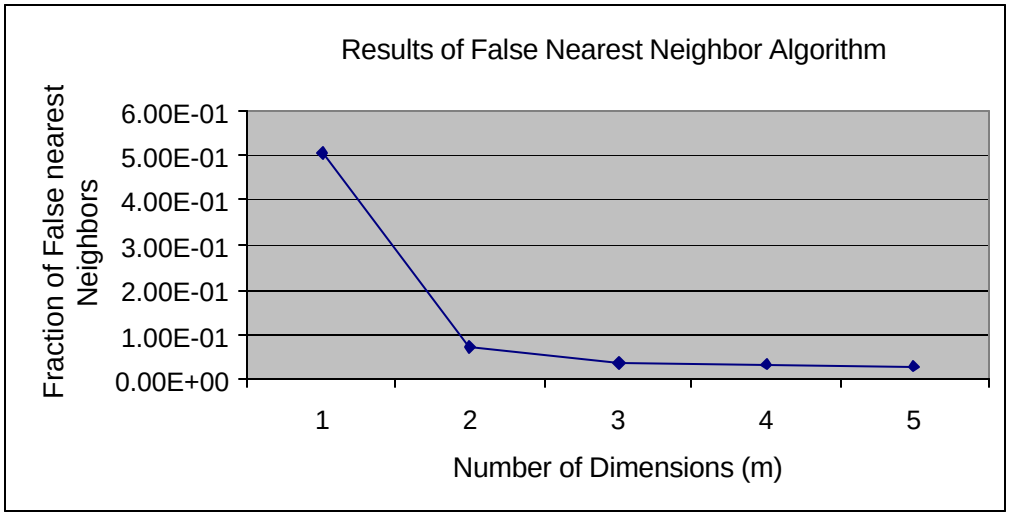

Figure 2.2 Plot Showing Relationship Between *m* and Percentage of False Nearest Neighbors

Once an embedding is achieved, the next step for prediction is to estimate a suitable function that is able to project the present state of the system, $x(t)$, into the forecasted value $x(t + t)$ at future time $t + t$. The function is presented in Equation 2.4.

$$x^F(t + t) = F(x(t)) \tag{2.4}$$

The next step is to identify the sets into which the constructed domain can be subdivided. The number of mutually overlapping sets is taken to be the equal to the total number of points to be forecasted; each set containing number *n* of vectors whose Euclidian distance from $x(t)$ is the minimum. *n* is the number of neighbors and when a suitable number of neighbors are chosen the model can be completely identified [28]. When a linear relationship is chosen for the function *F*, the model assumes the form as given in Equation 2.5

$$x^F(t + t) = \sum_{i=1}^{m_{tot}} (k_L)_i . a_i(t) + k_0 \tag{2.5}$$

where $a_i(t)$ are the components of $x(t)$, k_0 is the bias and $(k_L)_i$ are weights to be found using information from the n nearest neighbors of $x(t)$. The weights are different for each forecasted

point because each point lies in a different neighborhood in which weights are calibrated, thus ensuring the nonlinearity of the predicted times series.

F. Laio et al. [28] have performed a comparison of ANN and NLP approaches in flood prediction. The results have shown that the NLP method provides more accurate forecasts over a shorter prediction period (1-6 hours), however for prediction periods exceeding 24 hours, the ANN approach is more accurate.

Chapter 3 reviews the TSDM methodology.

Chapter 3. Time Series Data Mining Methodology

This chapter reviews the Time Series Data Mining methodology proposed by Richard Povinelli [41].

3.1 Time Series Data Mining

A primary difference between ANN, HMM and NLP and , Richard Povinelli's *Time Series Data Mining (TSDM)* is that the focus of TSDM is to identify and predict the occurrences of events, which in our case are floods whereas, the methods mentioned in the sections above focus on forecasting all future values of a time series. The TSDM methodology in this research is based on a univariate river flow time series. The steps in TSDM methodology are:

1. Training Stage
    1. Reconstruct the phase space from the observed time series by using the method of delays.
    2. Frame TSDM goal in terms of event characterization function, objective function and optimization formulation.
        1. Define the event characterization function g.
        2. Define the objective function f.
        3. Define the optimization formulation, the constraints on the objective function.
    4. Associate eventness with each time index represented by the event characterization function. Create the augmented phase space.

5. Search for optimal temporal pattern cluster in the augmented phase space that best characterizes the events.
6. Evaluate training stage results and repeat training stage as necessary.

2. Testing Stage
    1. Embed the testing time series into phase space.
    2. Use optimal temporal pattern cluster for predicting events.
    3. Evaluate testing results.

These steps are explained in detail in the following sections. The steps in training stage are elaborated in Section 3.2. Testing stage steps are explained in Section 3.3.

3.2 Training Stage

3.2.1 Step 1. Reconstruct the Phase Space

Reconstruct the attractor in a two dimensional phase space using time delay of one. The accurate calculation of time delay and number of embedding dimensions is not a requirement in TSDM. The reason is that unlike the NLP, TSDM does not try to exploit the predictive structure of the reconstructed phase space. The only purpose of a phase space reconstruction is to provide a simplified representation of the nonlinear time series. The identification of predictive patterns is accomplished by the data mining part of the TSDM methodology. Hence, the selection of time delay and embedding dimension is more a matter of choice. As an example, a synthetic seismic time series and its reconstructed phase space is shown in Figure 3.1 and 3.2 respectively.

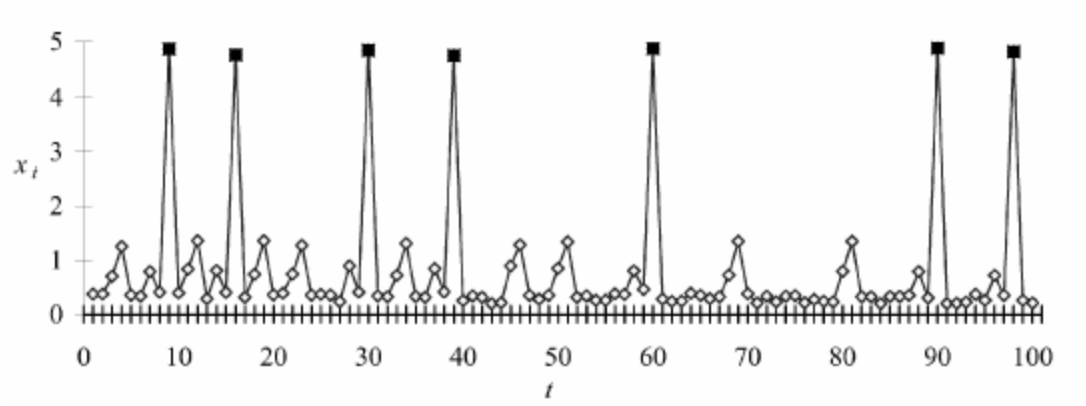

Figure 3.1 Synthetic Seismic Time Series (Training) with Events Identified [41]

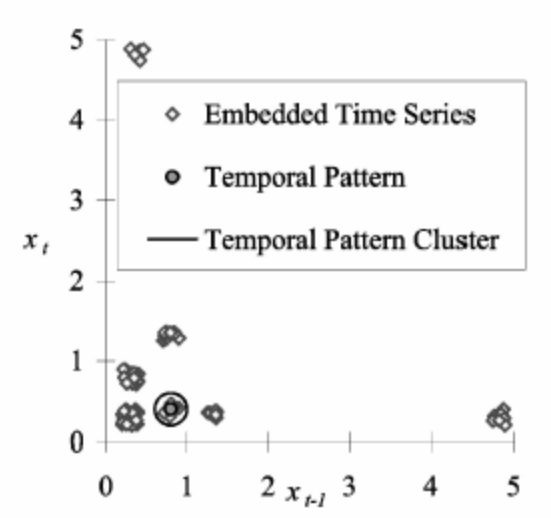

Figure 3.2 Reconstructed Phase Space from Synthetic Seismic Time Series [41]

3.2.2 Step 2. a) Define the Event Characterization Function

After an embedding has been achieved, the next step is to define a function that helps identify the events of interest and projects them in the phase space so as to identify the temporal patterns and temporal pattern clusters. The event characterization function is an application dependent function, defined in such a way that value that represents the equation at time *t* correlates to the value of that event in the future. These types of event characterization functions are classified as *causal* and are useful in prediction type problems. The other type of event characterization

function is the *non-causal* type and is useful for system identification and classification problems. For example, an event characterization function $g(t) = x_{t+1}$ is useful in predicting an event one-step before it actually occurs. Another example is

$$g(t) = \frac{x_{t+1} - x_t}{x_t} \tag{3.1}$$

g (t) representing the percentage change of values for time *t* to time *t+1*. The *g* value is calculated for each phase space point and is called *eventness*. The choice of event characterization function has a significant effect on the prediction results.

3.2.2 Step 2. b) Define the Objective Function

The objective function is used to determine which temporal pattern cluster is efficient in its ability to characterize events and is consistent with the TSDM goal. The objective function calculates a value for temporal pattern cluster *P*, which provides an ordering to temporal pattern clusters according to their ability to characterize events. The number of temporal pattern clusters could be one or more. Before finding the objective function, some basic definitions about *average eventness*, *variances* are necessary. The index set ? is a set of all time indices *t* of phase space points described by

$$? = \{t : t = (Q-1)t + 1, \ldots, N\} \tag{3.2}$$

where (*Q*-1) t is the largest embedding time-delay, and *N* is the number of observations in the time series. The time index set *M* is the set of all time indices *t* when x_t is within the temporal pattern cluster (*P*), i.e. $M = \{t : x_t \in P, t \in \Lambda\}$

The average value of *g*, also called *average eventness*, of the phase space points within the temporal cluster *P* is given by

$$m_M = \frac{1}{c(M)} \sum_{t \in M} g(t) \tag{3.3}$$

where $c(M)$ is the *cardinality* of *M*. Cardinality of *M* represents the number of points lying inside the temporal cluster. The average eventness of the phase space points of phase space points not in *P* is

$$m_{\tilde{M}} = \frac{1}{c(\tilde{M})} \sum_{t \in \tilde{M}} g(t) \tag{3.4}$$

where $c(\tilde{M})$ is the number of points outside the temporal cluster.

The average eventness of all the phase space points is given by

$$m_x = \frac{1}{c(\Lambda)} \sum_{t \in \Lambda} g(t) \tag{3.5}$$

The corresponding variances for respective average eventness functions are given in Equations 3.6, 3.7 and 3.8

$$s_M^2 = \frac{1}{c(M)} \sum_{t \in M} (g(t) - m_M)^2 \tag{3.6}$$

$$s_{\tilde{M}}^2 = \frac{1}{c(\tilde{M})} \sum_{t \in \tilde{M}} (g(t) - m_{\tilde{M}})^2 \tag{3.7}$$

$$s_x^2 = \frac{1}{c(\Lambda)} \sum_{t \in \Lambda} (g(t) - m_x)^2 \tag{3.8}$$

From these definitions, many different objective functions can be created for the purpose of selecting one temporal pattern cluster over the other. One example is the *t* test for the difference between two means of points inside and outside the cluster, shown in Equation 3.9.

$$f(P) = \frac{m_M - m_{\tilde{M}}}{\sqrt{\frac{s_M^2}{c(M)} + \frac{s_{\tilde{M}}^2}{c(\tilde{M})}}} \tag{3.9}$$

where *P* is a temporal pattern cluster. This function can be used for identifying a single statistically significant cluster that has high average eventness. For example, in Figure 3.3, cluster

P_1 is the best collection for identifying events, whereas P_2 is not. The objective function as in Equation 3.9 will map the collection of temporal pattern clusters such that $f(C_1) > f(C_2)$.

For identifying a single temporal pattern cluster where the purpose is to minimize the false predictions, an objective function of the type

$$f(P) = \frac{tp}{tp + fp} \tag{3.10}$$

is useful, where values of *tp*, *tn fp*, and *fn* are shown in Table 3.1. However this type of objective function is only applicable to problems where the time series observations can be classified as *tp*, *tn fp*, and *fn*.

Table 3.1 Event Categorization

	Actually an event	Actually a nonevent
Categorized as an event	True positive, *tp*	False positive, *fp*
Categorized as a nonevent	False negative, *fn*	True negative, *tn*

When the accuracy of prediction is of primary importance, the efficacy of a collection of temporal pattern clusters in total prediction accuracy is given by

$$f(C) = \frac{tp + tn}{tp + tn + fp + fn} \tag{3.11}$$

where *C* is the collection of temporal pattern clusters.

The category of TSDM problems, where the exact count of events and their magnitude are known, an event characterization function of the type

$$f(P) = \begin{cases} m_M & If\ c(M)/c(?) = ß \\ (m_M - g_0)\dfrac{c(M)}{bc(\Lambda)} + g_0 & otherwise \end{cases} \tag{3.12}$$

where ß is the minimum proportion of points inside the cluster. And g_0 is the minimum eventness of the phase space. This objective function orders temporal pattern clusters on the basis of time series observations with high eventness and characterizes at least a minimum number of events. The selection of objective function depends on the goal of TSDM.

3.2.2 Step 2. c) Define Optimization Formulation

It is possible that different temporal pattern clusters within the augmented phase space can contain the same set of phase space points. The optimization formulation becomes important in this case to determine the optimal size of cluster by maximizing the objective function *f (P).* Three types of biases, minimize, maximize or moderate can be possibly placed on *d,* the radius of the temporal pattern cluster *hypersphere.* Formulation by minimizing the *d* subject to *f (P)* remaining constant can be used to minimize the false positive prediction errors, i.e. the error of classifying a non-event as an event. This ensures that the temporal pattern cluster has as small coverage as possible, keeping the value of objective function constant.

3.2.3 Step 3. Create Augmented Phase Space

The augmented phase space is a *Q+1* dimensional phase space that is created by adding one more dimension to the existing phase space. The additional dimension is represented by the event characterization function *g (.)* in a way that every phase space point is a vector, as shown in Figure 3.3. The augmented phase space is expressed in Equation 3.13 given below

$$\langle x_t, g(t) \rangle \in R^{Q+1} \tag{3.13}$$

The next step is to be able to identify the optimal temporal cluster in the augmented phase space and this requires the formulation of an objective function.

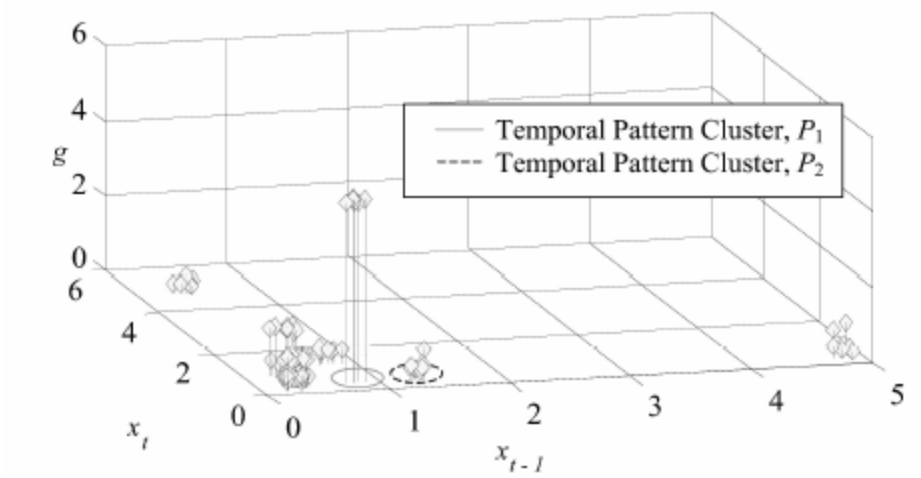

Figure 3.3 Clusters Identified in Phase Space from Synthetic Earthquake Time Series [41]

3.2.4 Step 4. Search for Optimal Temporal Patter Cluster

The search for optimal temporal pattern cluster is performed using the Genetic Algorithm [18]. Figure 3.4 shows the temporal pattern cluster identified by the Genetic Algorithm.

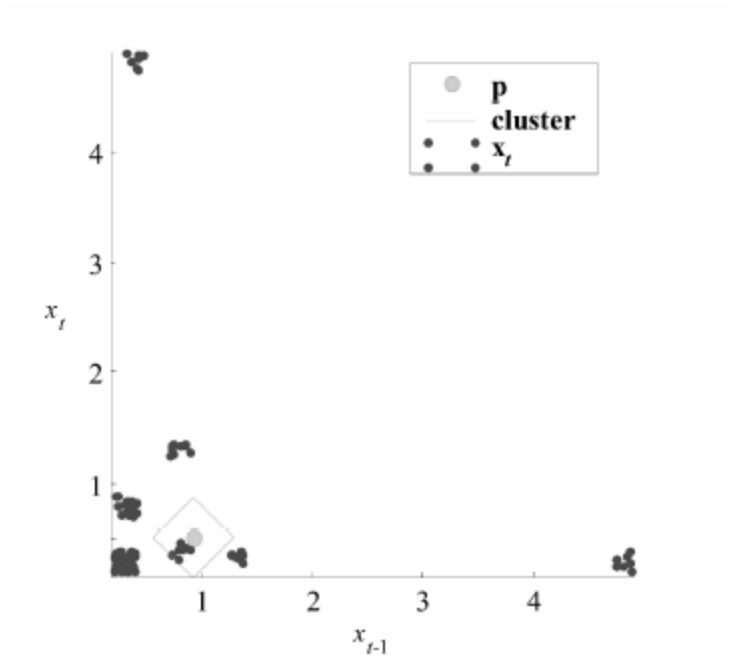

Figure 3.4 Synthetic Seismic Phase Space with Temporal Pattern Cluster Identified [41]

3.2.5 Step 5. Evaluate Training Results

From Figure 3.5 it is clear that all temporal patterns occurring before the events have been identified and thus training stage is successful.

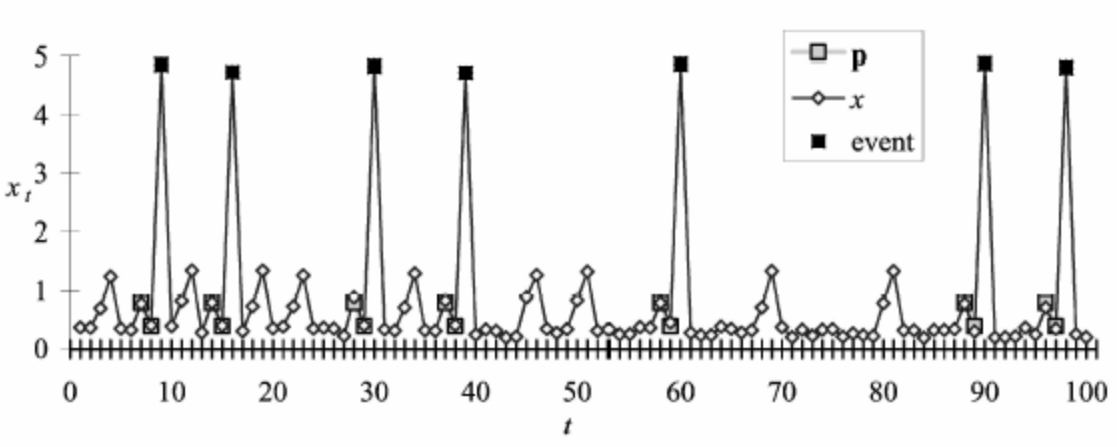

Figure 3.5 Synthetic Seismic Time Series with Temporal Patterns Highlighted [41]

3.3 Testing Stage

3.3.1 Step 1. Embed The Testing Time Series Into Phase Space

Embed the testing time series in phase space using the same embedding parameters as the training time series.

3.3.2 Step 2. Use The Optimal Temporal Pattern Cluster to Predict Events

Whenever a point in the testing time series phase space falls inside the cluster identified in the training stage, an event is predicted.

3.3.3 Step 3. Evaluate Testing Results

Testing results are evaluated by measuring the number of events correctly identified and predicted. Figure 3.6 shows a testing time series and the events predicted.

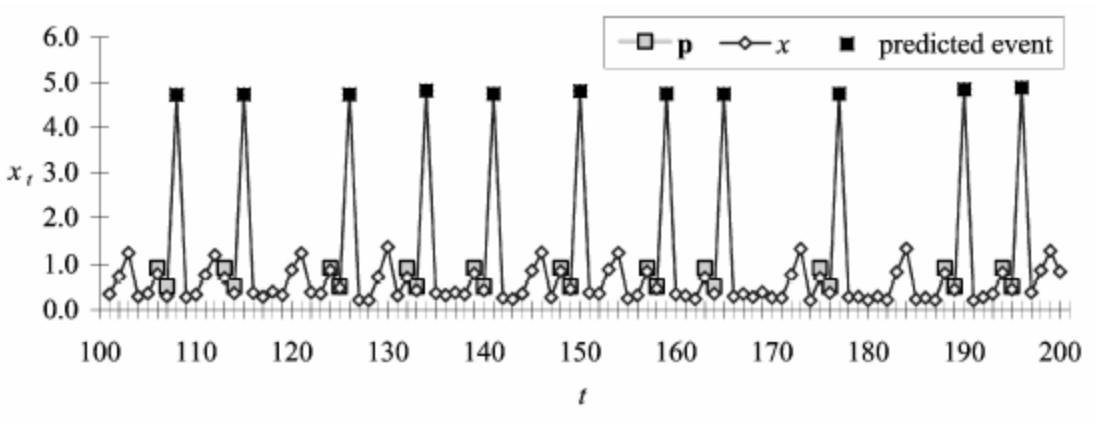

Figure 3.6 Prediction Results in a Testing Time Series [41]

In Chapter 4, the TSDM is applied to flood prediction at the three gauging stations and results are presented.

Chapter 4. Application of TSDM to Flood Forecasting and Results

This chapter describes the application of TSDM to flood prediction at three gauging stations. Section 4.1 describes the Surrogate Data method for detection of chaos in the river discharge time series. In Section 4.2 the parameters for phase space reconstruction are explained. In Section 4.3 the event characterization function, objective function and the optimization formulation are explained. Section 4.4 describes the measures for cluster prediction accuracy. This is followed by description of problem and setup in Section 4.5. Section 4.6 presents the results using two objective functions and the Earliness Prediction Accuracy. A summary of results is presented in Section 4.7.

4.1 Detection of Chaos in the River Daily Discharge Time Series

Before the TSDM methodology is applied to flood forecasting, the presence of nonlinearity in the river discharge time series at the gauging stations needs to be confirmed. For this, the Surrogate data method [25], a technique for detecting the presence of chaos is used. Surrogate data method consists of computing a nonlinear statistic from the data being analyzed and from an ensemble of realizations of a linear stochastic process, which mimics linear properties of the data under study. If the computed statistic for the original data is significantly different from the values obtained for the surrogate data set, it can be inferred that the data was not generated by a linear process; otherwise the null hypothesis, that a linear model fully explains the data, cannot be rejected and the data can be analyzed, characterized and predicted using well-developed linear methods [31]. The TISEAN 2.1 software for Nonlinear Time Series Analysis [25] is used here for

detecting chaos in river daily discharge time series. The null hypothesis and alternate hypothesis are:

H_o: Daily discharge time Series is linear and stationary

Root Mean Square Error for daily discharge time series = Root Mean Square Prediction Error for its surrogate data

H_a: Daily Discharge Time Series is non linear and non-stationary

Root Mean Square Prediction Error for daily discharge time series > Root Mean Square Prediction Error for its surrogate data

Level of Significance: $a = 0.05$

Test Statistic: Difference between Root Mean Square Prediction Error for Daily Discharge Time Series and its Surrogate Data

All three discharge time series were tested for presence of chaos. Using the surrogate data testing, the null hypothesis of a stationary, linear Gaussian random process was rejected at the 95% level of significance, since the prediction error of the river discharge data is found to be significantly higher than that of the surrogates. This proves that the daily discharge time series observed at all three gauging stations are chaotic and justify the use of TSDM in their analyses.

4.2 Reconstruction of Phase Space from Training Time Series

As explained in Section 3.2, the TSDM methodology does not require accurate estimates of time delay and embedding dimension. The choice of time delay and number of dimensions does not have any effect on the prediction results of TSDM methodology. The phase space is reconstructed in two dimensions with a time delay of one for all three examples considered.

4.3 Event Characterization Function, Objective Function and Optimization

4.3.1 Event Characterization Function

Event characterization function represents the value of future eventness for the present time index. Different event characterization functions can be used to model the prediction of floods. The event characterization functions considered are:

1. $g(x_t) = \frac{x_{t+1} - x_t}{x_t}$: percentage change in discharge values between two consecutive days
2. $g(x_t) = x_{t+1}$:captures the goal of characterizing a flood one time step in future
3. $g(x_t) = x_{t+i}$:captures the goal of characterizing a flood i time steps in future

It is found that the percentage change function could not highlight the events of interest in the augmented phase space. During the evaluation of training stage results, the percentage change function was able to identify only 4 out of 15 events in the training time series from St. Louis gauging station. This indicates that a high rise in the discharge values over two successive days does not necessarily result in a flood. For example, if a heavy rainfall follows a period of draught, the discharge will increase rapidly over a period of days. These rapid increases in discharge will be highlighted in the augmented phase space although they may not lead to a flood. On the other hand, if the discharge increases gradually and results in a flood, it would not be highlighted by the percentage change function, since the changes are not drastic.

The event characterization function is chosen depending on how many days early the flood is to be predicted. The one step-ahead event characterization function is useful for prediction of floods one day early. Different step-ahead event characterization functions are applied and the effect of earliness of prediction on the prediction accuracy is measured.

4.3.2 Objective Function

The difference of means objective function,

$$f(P) = \frac{m_M - m_{\tilde{M}}}{\sqrt{\frac{s_M^2}{c(M)} + \frac{s_{\tilde{M}}^2}{c(\tilde{M})}}},$$

explained in Chapter 3, is not applicable to the flood prediction problem because of the linearly increasing nature of the phase space as shown in Figure 4.1. The cluster, in order to maximize the difference between the g values of points inside and outside the cluster, includes all the points in the phase space, resulting in a large number of false alarms in the testing time series. There is no upper limit on the size of the cluster. The difference of means objective function is only applicable to phase spaces that are disjoint or have multiple clusters, where the objective function selects one cluster over the other based on the $f(P)$ value.

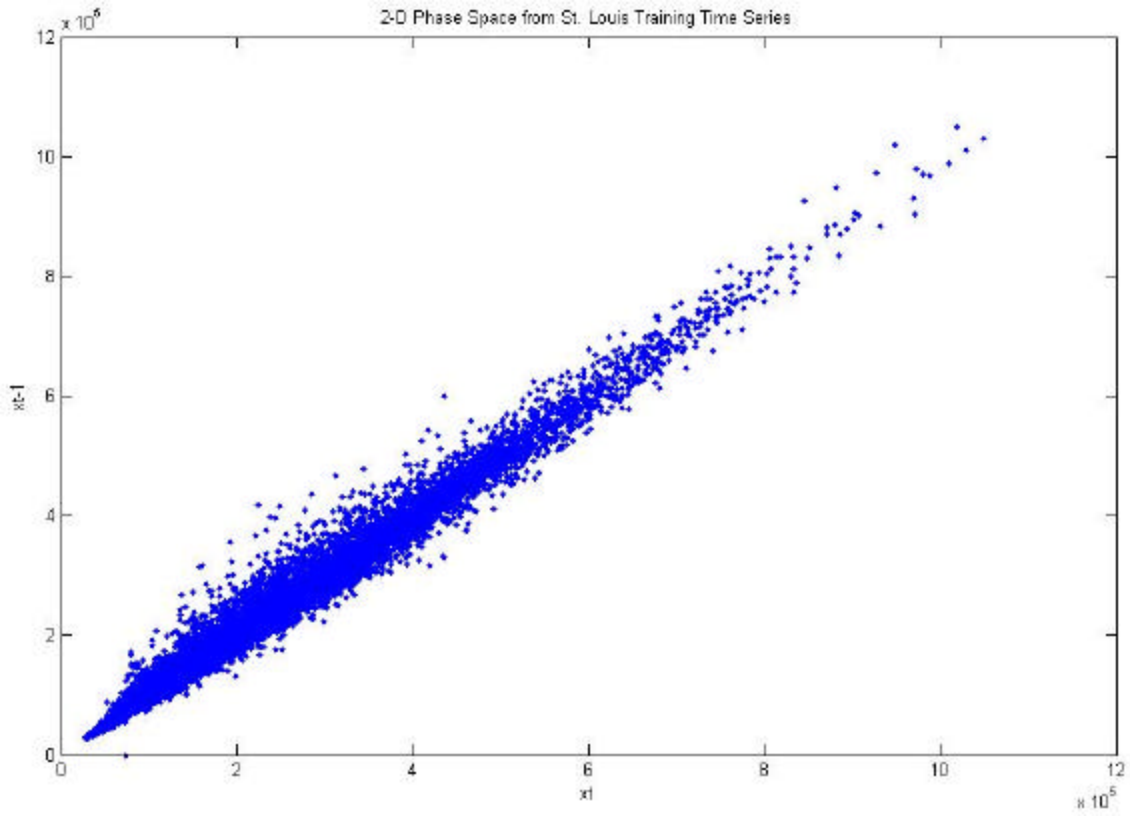

Figure 4.1 2-D Embedding of Training Time Series from St. Louis Gauging Station

Two objective functions, both of which are variations of the objective function that selects at least a minimum proportion of points in the phase space, are considered. The objective function that selects a minimum proportion of points in phase space is

$$f(P) = \begin{cases} m_M & \text{If } c(M)/c(?) = ß \\ (m_M - g_0)\dfrac{c(M)}{bc(\Lambda)} + g_0 & \text{otherwise} \end{cases} \quad (4.1)$$

where *c(M)* is the cardinality of the cluster and *c(?)* is the total number of points in the phase space. *ß* specifies the minimum proportion of points that the cluster can include. The objective function value is m_M, which is the mean of g values of points inside the cluster, if the cluster includes the minimum proportion of points. Otherwise the objective function value is $(m_M - g_0)\frac{c(M)}{bc(\Lambda)} + g_0$ which is lesser than m_M.

This objective function tries to maximize the mean of *g* values of points inside the cluster. As in case of the difference of means function, this event characterization function cannot be applied to the flood prediction problem because of the linearly increasing nature of phase space . In order to maximize the mean of *g* values of points inside the cluster, the GA selects a cluster with a single point having the highest *g* value in the phase space. Every other point added to the cluster reduces the cluster mean and hence the GA terminates with only one point in the cluster and a radius of zero. There is no lower limit on the size of the cluster. In order to address this problem, two variations of the objective function for minimum proportion of points are proposed.

Objective Function I

Objective Function I is applicable to flood prediction problems where the history of floods is known. The minimum discharge value that results in a flood can be identified and, based on this value, each event can be classified as a true positive or a false positive according to the following definitions:

1. *tp* (True Positives): If the event identified by the cluster as a flood is actually a flood it is called a true positive.
2. *fp* (False Positive): If an event identified by the cluster is not a flood it constitutes a false positive or a false alarm.

Objective Function I is formally stated as

$$f(P) = \begin{cases} tp \times \sum_{i=1}^{M} g_M & If \ \dfrac{f_p}{f_p + t_p} \le b \\ -\left(\sum_{i=1}^{M} g_M \right) & otherwise \end{cases} \qquad (4.2)$$

The Objective Function I uses the knowledge learnt from historical data. Instead of specifying the minimum proportion of points to be included in the cluster as in Equation 4.1, this objective function specifies the maximum proportion of false positives allowed in the cluster. Another difference is that the objective function tries to maximize the product of sum of *g* values and the number of *tp*'s. For every *tp* included, the $f(P)$ function is rewarded by multiplying the summation of *g* values of points inside the cluster by the number of *tp*'s inside the cluster. As a result, the cluster tries to include all true positives along with other phase space points with high *g* values, however, its size is restricted by the maximum proportion of *fp*'s (ß).

Objective Function II

Objective Function II is applicable to the flood forecasting problems where no information is available regarding previous floods. The lack of information means that there is no way to classify an event as a *tp* or a *fp*. In this case, another criteria which is based on flood zoning is used to train the cluster.

Areas susceptible to floods are classified into 10-year, 50-year, 100-year flood zones. Suppose an area has been classified as 100-year flood zone. It means that there is a 1% probability of a flood occurring in that area in any given year. From this probability, the expected

number of floods for the period under study can be calculated. The Objective Function II uses the ratio of expected number of days of flooding to the total number of observations in the training time series for limiting the cluster size. The Objective Function II is presented below

$$f(P) = \begin{cases} \sum_{i=1}^{M} g_M & \text{If } c(M)/c(?) = ? \\ -\left(\sum_{i=1}^{M} g_M \right) & \text{otherwise} \end{cases}$$

where, ? is the maximum proportion of phase space points inside the cluster. If the proportion of phase space points inside the cluster is less than ?, the objective function value is the positive summation of g values inside the cluster. If the proportion of phase space points inside the cluster exceeds ? , the objective function value is negative summation of g values inside the cluster.

The goal of this objective function is to select the points with high g values to be a part of cluster in order to maximize the summation of g values. The objective function is maximum when all the phase space points lie inside the cluster, however the cluster size is restricted by ?, which the proportion of points based on the flood zone and the length (in number of days or observations) of the training time series.

4.3.3 Optimization Formulation

The selection of optimal cluster size is accomplished with the use of optimization formulation. The optimization formulation is modeled as a multiobjective formulation with the two objectives as:

1. Maximize the value of objective function and
2. Minimize the radius of the cluster.

An unsupervised clustering technique, the Genetic Algorithm [18] is used in the search process for optimal temporal pattern cluster. The Genetic Algorithm (GA) searches for a global maxima and, identifies the optimal cluster. As explained in Section 3.2.2, priorities are assigned

to the two objectives, with the maximization of objective function being the first priority and minimization of radius is the secondary priority. Thus, the GA will search for a temporal pattern cluster that maximizes the objective function value and if there are multiple clusters with equally high objective function value, it selects the cluster with minimum radius. The minimization objective is required to select the crispest cluster in order to minimize the number of false positives in the testing phase. The Genetic Algorithm Toolbox in Matlab, version 7.0.1 (Release 14) is used for the search and the output is the cluster center and its radius.

4.4 Cluster Prediction Accuracy

In order to determine the prediction accuracy of the cluster and measure its ability to identify and characterize the floods in the training and testing phases, the following set of parameters is defined.

1. *tp* (True Positives): If the event identified by the cluster as a flood is actually a flood it is called a true positive.
2. *fp* (False Positive): If an event identified by the cluster is not a flood it constitutes a false positive or a false alarm.
3. *a* (Type I Error): In hypothesis testing, *a* is defined as the probability of rejecting a null hypothesis when its true. Analogically, applied to cluster prediction accuracy, *a* is the probability of missing a true positive.
4. *ß* (Type II Error): *ß* is defined as the probability of failing to reject the null hypothesis even though it's false. Applied to cluster prediction accuracy, *ß* is the probability of selecting a false positive.
5. *Positive Prediction Accuracy*: Positive Prediction Accuracy (PPA) [41] is the percentage of true positives in the cluster. Since the events are classified either as true positives or

false positives, the positive prediction accuracy of a cluster is calculated as $\frac{tp}{tp + fp} \times 100$.

PPA can also be calculated as $(1 - b) \times 100$, since ß is $\frac{fp}{tp + fp}$. $(1 - b)$ is the probability of selecting a true positive.

6. *Correct Prediction Percentage*: Correct Prediction Percentage (CPP) is defined as the percentage of true positives predicted. Correct Prediction Percentage = $\frac{tp(predicted)}{tp(actual)}$.

   The cluster having a high PPA as well as high CPP is the crispest cluster with maximum prediction accuracy.

7. *Number of Starts Missed*: Since the goal is to predict occurrences of floods, it is more important to predict the first instance when the discharge exceeded the flood threshold, causing the river to overflow, than to predict all events (*tp*'s) when the discharge exceeds the threshold. Hence, along with CPP measured with respect to number of *tp*'s predicted, the number of starts of floods missed are also measured.

4.5 Problem Description and Setup

TSDM is applied to flood forecasting at three gauging stations. The Kansas City gauging station time series contains 3 floods, the St. Louis gauging station time series contains 6 floods and the Harrisburg gauging station time series contains 15 floods during the periods under study. These three gauging stations are selected as examples because they represent low, medium and high number of flood occurrences.

St. Louis Gauging Station, Mississippi River

The daily discharge time series for St. Louis gauging station is obtained from the United States Geological Survey website [21], covering the time period from April 1933 to September 2003, consisting of 25750 data points. The daily discharge time series is shown in Figure 4.2.

Historical records show that floods have occurred during the years 1943, 1944, 1947, 1973, 1993 and 1995. The events of interest are the peak discharge values during the flooding periods.

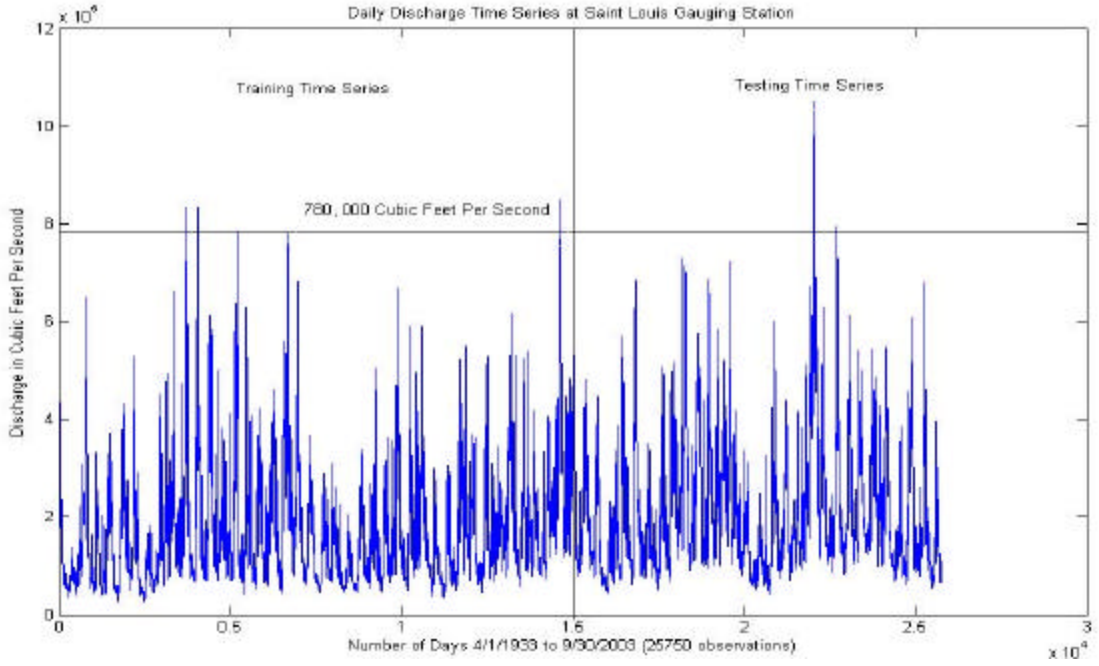

Figure 4.2 Time Series Plot of Daily Discharge Values at St. Louis Gauging Station

The time series is split into two parts, the first 15000 data points making up the training time series and, the rest will be used in testing. TSDM is applied to the training data set and, the prediction accuracy is calculated by its ability to predict actual floods occurring in the testing time series. The minimum value of discharge, during the reported flooding periods is 780,000 cubic feet per second. This metric has been chosen as an identifier for floods, i.e. a value of discharge exceeding 780,000 cu. feet/sec results in a flood and hence constitutes an event. The 2-D phase space and the augmented phase space from the training time series are shown in Figures 4.3 and 4.4 below. The temporal pattern clusters are identified from this time series and consequently used in the prediction of floods in the testing part.

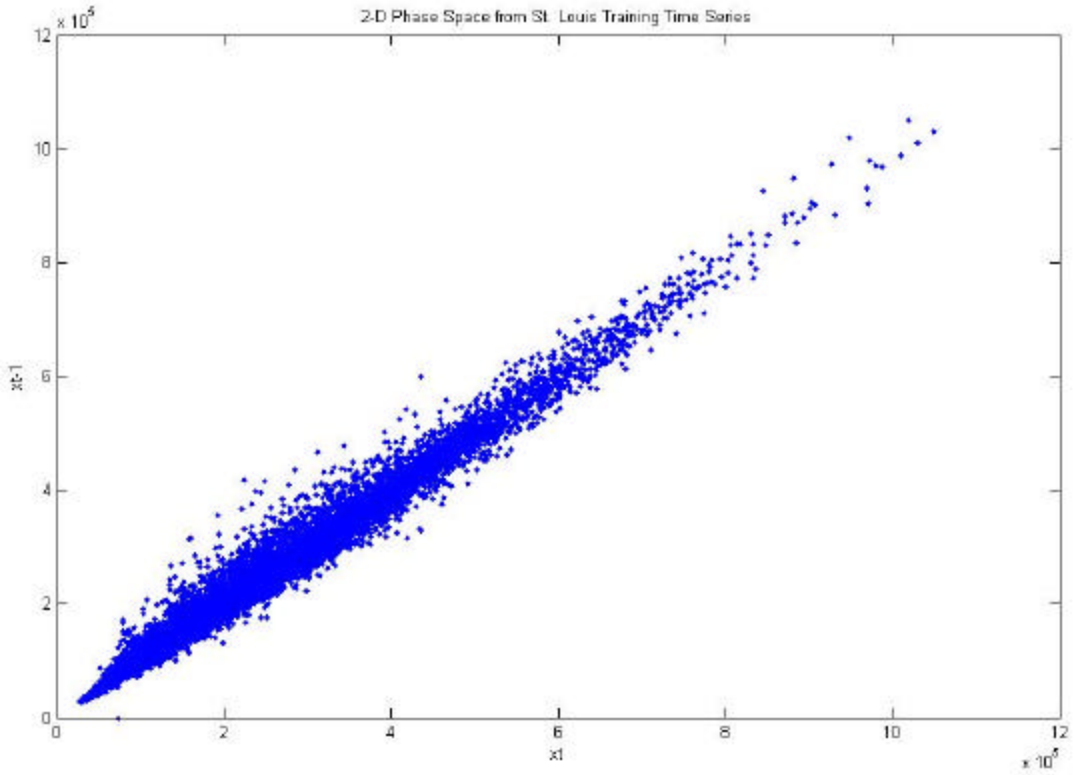

Figure 4.3 2-D Embedding of St. Louis Training Time Series

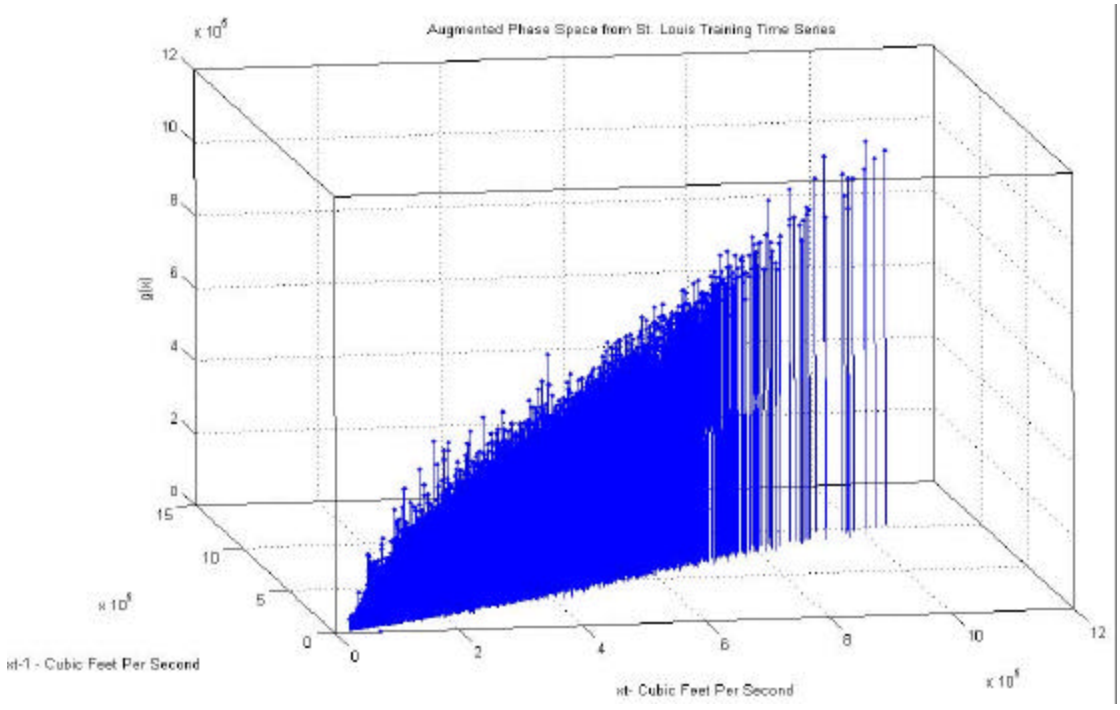

Figure 4.4 Augmented Phase Space For Training Series for St. Louis

Kansas City Gauging Station, Missouri River

The Missouri River that passes through Kansas City, MO has flooded three times during the recorded history. These floods occurred during the years of 1951, 1952 and 1993. The daily discharge time series consists of 27393 observations [21], recorded between 10/1/1928 and 9/30/2003 as shown in Figure 4.5. The minimum discharge observed during the floods is 380,000 cubic feet per second and is chosen as the event identifier. The time series is split into the training and testing time series. The training time series has 13500 data points, contains two floods (1951 and 1952) and six events when the discharge exceeded 380,000 cubic feet per second .

The testing time series is comprised of 13893 data points, representing the time period between 16/9/1965 and 9/30/2003. One flood occurred during this time period, during the year of 1993. The testing time series contains six events.

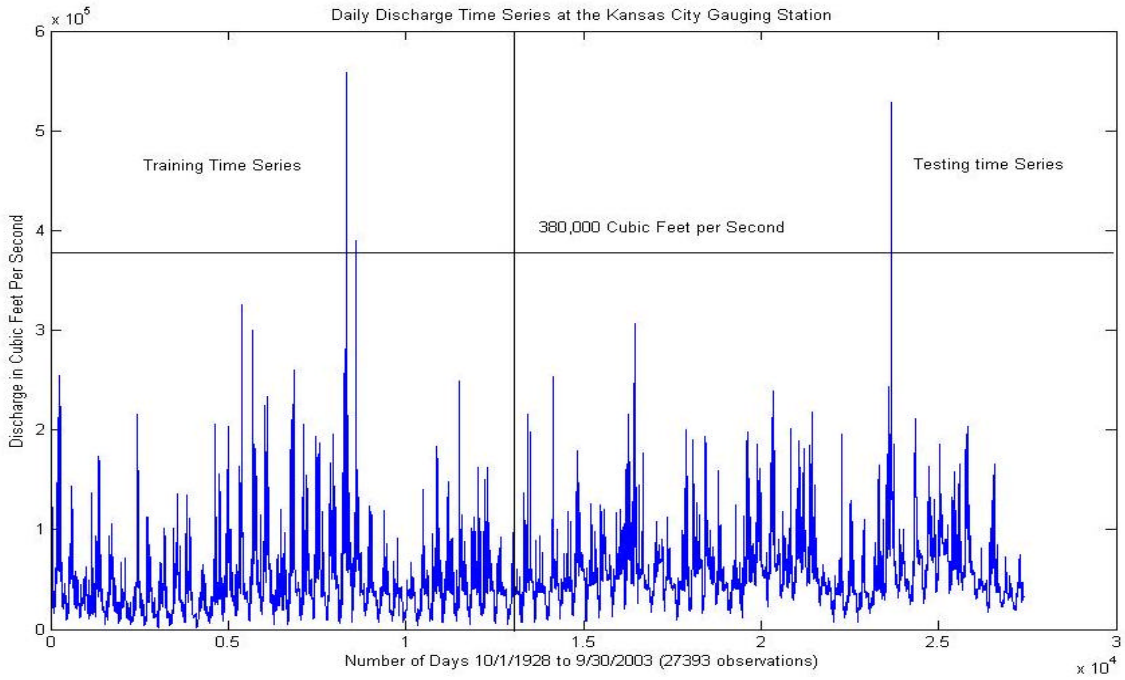

Figure 4.5 Daily Discharge Time Series at Kansas City Gauging Station, Missouri River

The 2-D phase space from the training time series and the augmented phase space are presented in Figures 4.6 and 4.7 below.

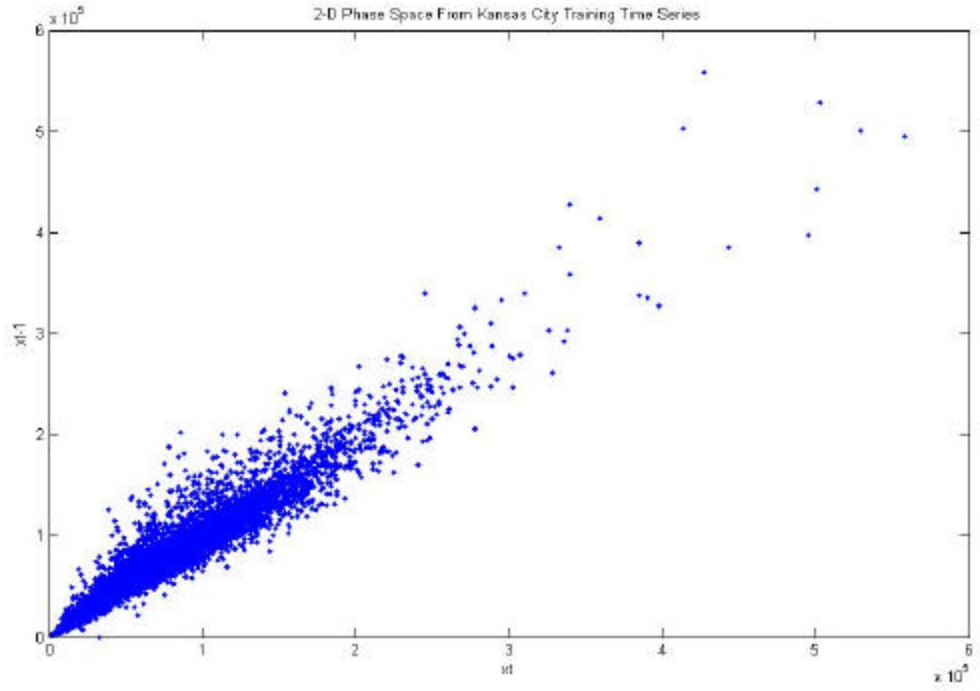

Figure 4.6 2-D Embedding of Kansas City Training Time Series

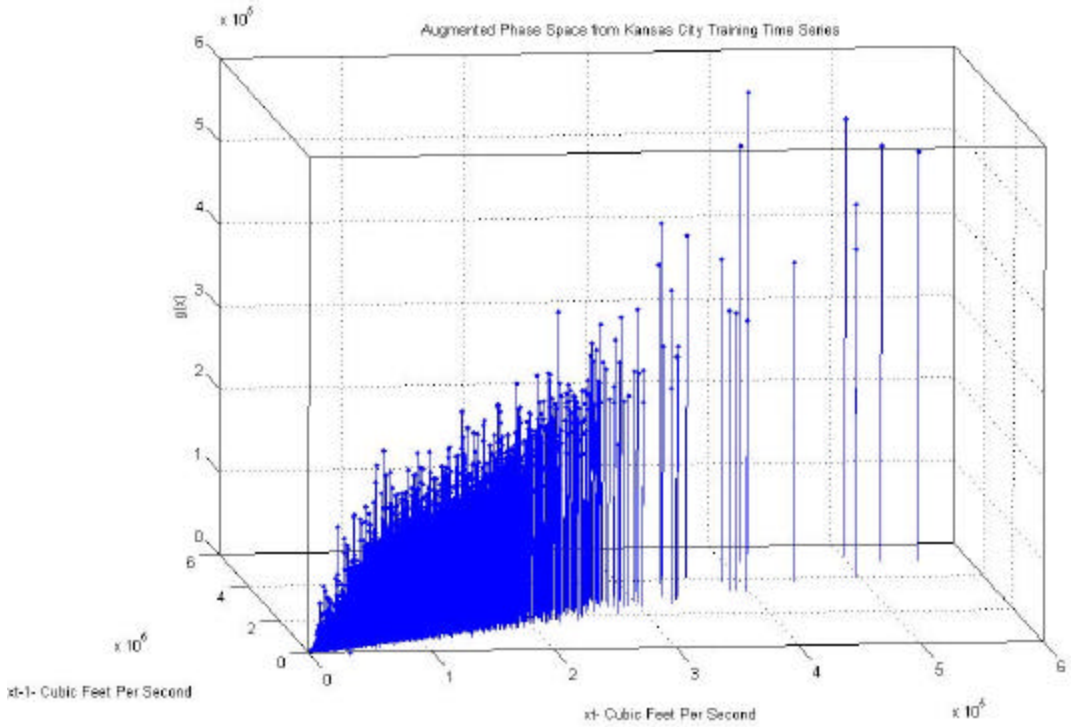

Figure 4.7 Augmented Phase Space For Training Series for Kansas City

Harrisburg Gauging Station, Susquehanna River

The daily discharge time series at Harrisburg gauging station contains observations recorded between 10/01/1890 and 09/30/2003 (41272 data points) [21]. Harrisburg is one of the highest flood hit areas of United States. Floods have occurred during the years of 1894, 1901, 1902, 1920, 1936, 1940, 1943, 1950, 1964, 1972, 1975, 1980, 1985, 1994 and 1997. The time series is bisected into training and testing time series, each comprising of 20636 data points. The minimum value of discharge during flooding is 390,000 cubic feet per second. This value is chosen as the identifier for flood events. The training time series covers time between 10/01/1890 to 04/01/1947, contains seven floods and 17 *tp*'s. The testing time series contains eight floods and 14 *tp*'s. The goal is to predict the eight floods in the testing time series accurately and as early as possible. The daily discharge time series is presented in Figure 4.8.

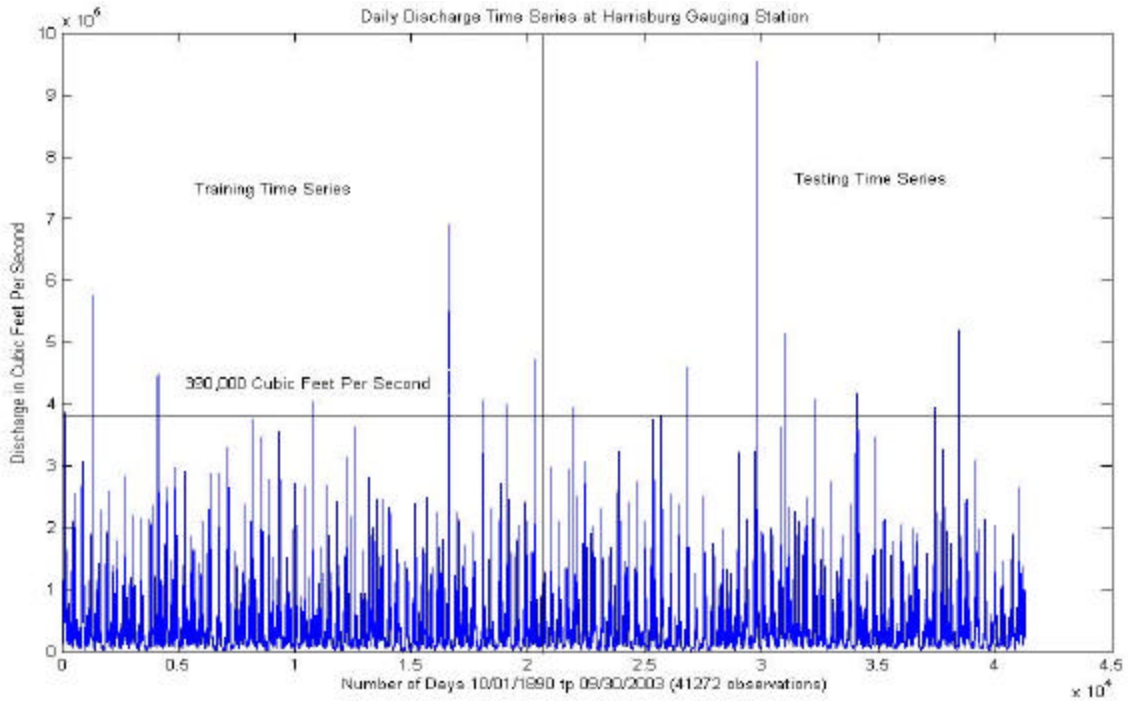

Figure 4.8 Daily Discharge Time Series at Harrisburg Gauging Station, Susquehanna River

The 2-D phase space and the augmented phase space for the Harrisburg gauging station training time series are shown in Figure 4.9 and 4.10.

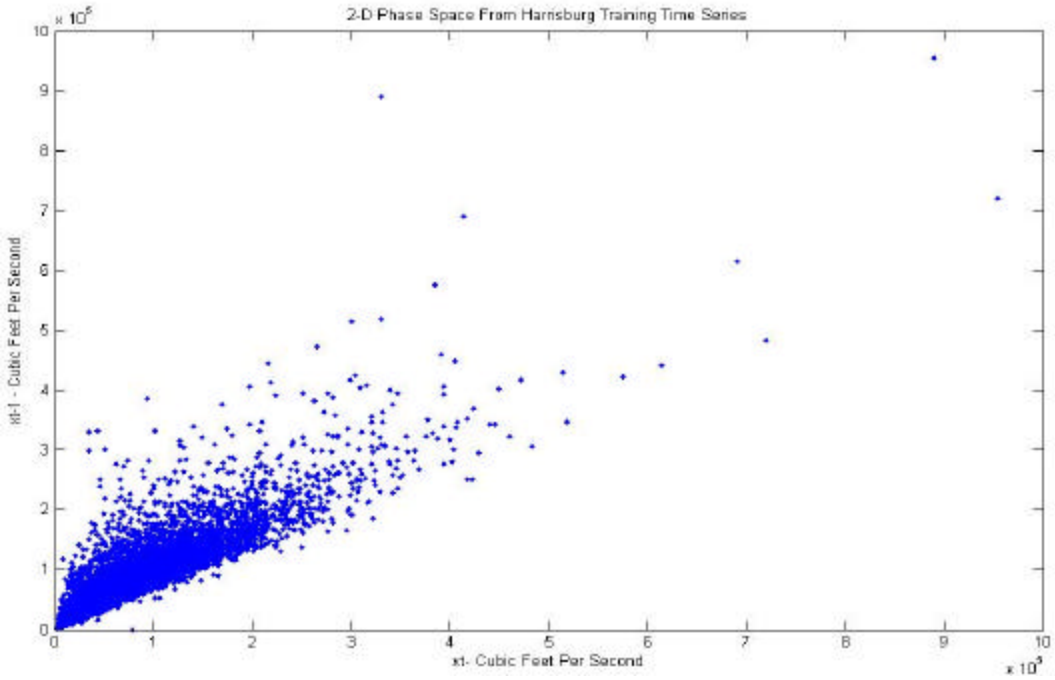

Figure 4.9 2-D Phase Embedding of Harrisburg Training Time Series

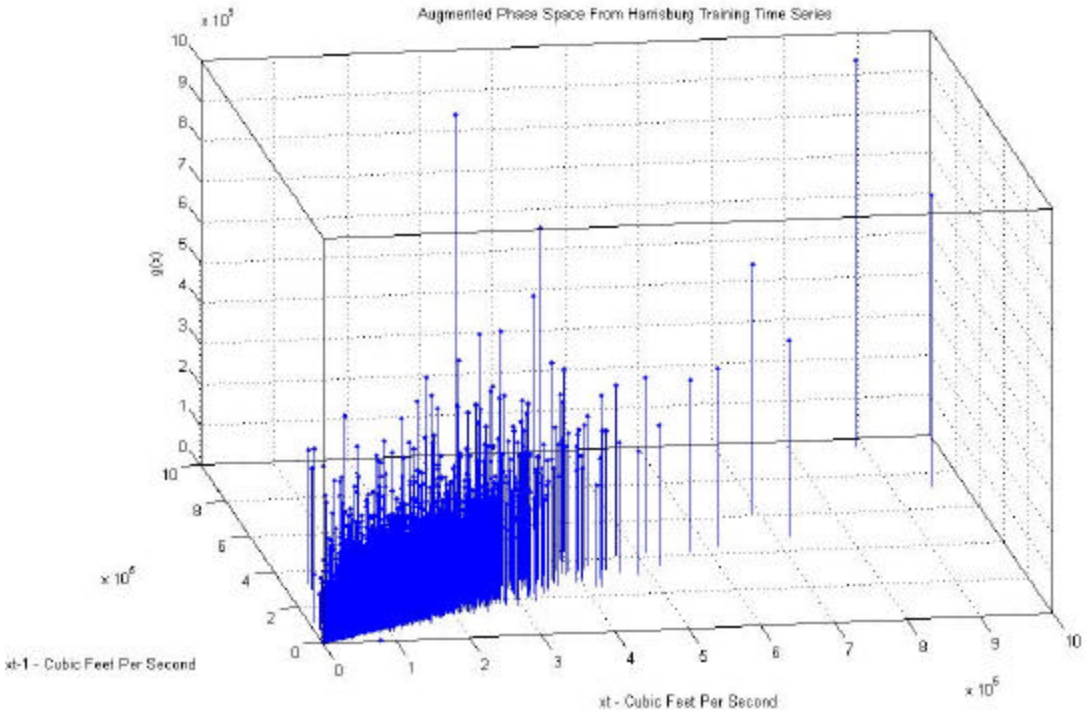

Figure 4.10 Augmented Phase Space for Harrisburg Training Series

4.6 Results

This section presents the results from application of TSDM to flood forecasting. The goal is to predict floods accurately and as early as possible. In Section 4.6.1, the prediction accuracy is calculated for all three gauging stations using Objective Function I for different values of *ß*. In Section 4.6.2, the Earliness Prediction Accuracy is evaluated for all three gauging stations using different step-ahead event characterization functions. This is followed by application of Objective Function II to flood forecasting at St. Louis gauging station.

4.6.1 Prediction Accuracy with Objective Function I

This section presents the prediction accuracy for the three gauging stations using Objective Function I for different *ß* values. The following set of parameters is used :

Event Characterization Function : One step-ahead function – $g(x_t) = x_{t+1}$

Objective Function : Objective Function I

$$f(P) = \begin{cases} tp \times \sum_{i=1}^{M} g_M & If \ \dfrac{f_p}{f_p + t_p} \le b \\ -\left(\sum_{i=1}^{M} g_M \right) & otherwise \end{cases}$$

Optimization Formulation : 1. Maximize the value of Objective Function (Priority I)

2. Minimize the radius of cluster (Priority II)

First, the training and testing results for St. Louis gauging station are presented followed by the results for Kansas City and Harrisburg gauging stations.

Training and Testing Results for St. Louis Gauging Station

Training stage results for St. Louis gauging station are presented in Table 4.1. Objective Function I rewards the GA for including *tp*'s in the cluster. Since the *tp*'s are rewarded, the GA tries to include maximum possible *tp*'s in the cluster.

Table 4.1 Training Stage Results for Different ß Values at St. Louis Gauging Station

ß Specified	Points in Cluster	*tp*	*fp*	Events Missed	*a*	PPA (*tp* /*tp*+*fp*) or (1- ß)	CPP *tp(predicted)* / *tp(actual)*
0.95	295	15	280	0	0	5.08	100
0.85	98	15	83	0	0	15.31	100
0.75	60	15	45	0	0	25.00	100
0.65	42	15	27	0	0	35.71	100
0.55	33	15	18	0	0	45.45	100
0.45	27	15	12	0	0	55.56	100
0.35	23	15	8	0	0	65.22	100
0.25	20	15	5	0	0	75.00	100
0.15	17	15	2	0	0	88.24	100
0.05	No Cluster Identified						

For ß value until 0.15, the GA is able to identify clusters which include all *tp*'s and hence have a 100% CPP and the *a* value is zero. No cluster was identified for ß value of 0.05 and below. The clusters identified in the training stage are used to predict floods in the testing stage. Clusters for different values of ß from training stage are applied testing phase space and the resulting *a*, ß, PPA and CPP values are shown in Table 4.2

Table 4.2 Testing Stage Results for Different ß Values at St. Louis Gauging Station

Number of Floods in Testing Time Series= 2, Number of Events in Testing Time Series= 33									
ß Spec. In Training	Points in Cluster	*tp*	*fp*	Events Missed	*a*	Resultant ß	PPA *tp* /*tp*+*fp* or (1- ß)	CPP *tp(pred.)* / *tp(actual)*	Number of Starts Missed
0.95	272	12	260	21	0.64	0.96	4.41	36.36	0
0.85	90	20	70	13	0.39	0.78	22.22	60.61	0
0.75	53	17	36	16	0.48	0.68	32.08	51.52	0
0.65	26	12	14	21	0.64	0.54	46.15	36.36	0
0.55	18	14	4	19	0.58	0.22	77.78	42.42	0
0.45	18	15	3	18	0.55	0.17	83.33	45.45	0
0.35	22	20	2	13	0.39	0.09	90.91	60.61	0
0.25	11	10	1	23	0.70	0.09	90.91	30.30	0
0.15	11	10	1	23	0.70	0.09	90.91	30.30	0
0.05	No Cluster Identified								

Although the *a* value increases with a decrease in ß, indicating that more events are being missed, none of the clusters miss the starts of floods. The Correct Prediction Percentage is highest

for ß values of 0.85 and 0.35. From the testing time series results, it can be seen that there is no clear relationship between decrease in ß value and the CPP. This happens because of the non-uniform density of points in the phase space. If the density of points in phase space would have been uniform, a decrease in the ß value would have meant a smaller cluster with less number of *fp*'s and a lesser number of *tp*'s as well. However, this is not the case as prediction accuracy also depends on the location of the cluster in the phase space as much as it does on the cluster size (controlled by ß).

Training and Testing Results for Kansas City Gauging Station

Using the same set of parameters as in St. Louis gauging station, the training stage results for Kansas City gauging station are presented in Table 4.3.

Table 4.3 Training Time Series Results for Different ß Values at Kansas City Gauging Station

ß Specified	Points in Cluster	*tp*	*fp*	Events Missed	*a*	PPA (*tp /tp+fp)* or (1- ß)	CPP *tp(predicted)* / *tp(actual)*
0.95	97	5	92	1	0.17	5.15	83.33
0.85	24	5	19	1	0.17	20.83	83.33
0.75	11	5	6	1	0.17	45.45	83.33
0.65	8	5	3	1	0.17	62.50	83.33
0.55	7	5	2	1	0.17	71.43	83.33
0.45	7	5	2	1	0.17	71.43	83.33
0.35	7	5	2	1	0.17	71.43	83.33
0.25	6	5	1	1	0.17	83.33	83.33
0.15	4	4	0	2	0.33	100.00	66.67
0.05	4	4	0	2	0.33	100.00	66.67

The testing time series results are presented in Table 4.4

Table 4.4 Testing Time Series Results for Different ß Values at Kansas City Gauging Station

Number of Floods in Testing Time Series= 1, Number of Events in Testing Time Series= 6									
ß Spec. In Training	Points in Cluster	*tp*	*fp*	Events Missed	*a*	Resultant ß	PPA *tp /tp+fp* or (1- ß)	CPP *tp(pred.)* / *tp(actual)*	Number of Starts Missed
0.95	60	6	54	0	0.00	0.90	10.00	100.00	0
0.85	15	4	11	2	0.33	0.73	26.67	66.67	0
0.75	8	4	4	2	0.33	0.50	50.00	66.67	0
0.65	8	4	4	2	0.33	0.50	50.00	66.67	0
0.55	7	4	3	2	0.33	0.43	57.14	66.67	0
0.45	7	4	3	2	0.33	0.43	57.14	66.67	0
0.35	8	6	2	0	0.00	0.25	75.00	100.00	0
0.25	7	6	1	0	0.00	0.14	85.71	100.00	0
0.15	3	3	0	3	0.50	0.00	100.00	50.00	0
0.05	4	3	1	3	0.50	0.25	75.00	50.00	0

As in training time series, there is no clear relationship between the ß and CPP because of the non-uniform density of phase space points. The highest prediction accuracy is 100% for ß = 0.95, 0.35 and 0.25. The clusters do not miss the starts of floods for any value of ß .

Training and Testing Results for Harrisburg Gauging Station

The training stage results for Harrisburg gauging station are presented in Table 4.5.

Table 4.5 Training Time Series Results for Different ß Values at Harrisburg Gauging Station

ß Specified	Points in Cluster	*tp*	*fp*	Events Missed	*a*	PPA (*tp /tp+fp)* or (1- ß)	CPP *tp(predicted)* / *tp(actual)*
0.95	229	13	216	4	0.24	5.68	76.47
0.85	60	13	47	4	0.24	21.67	76.47
0.75	45	13	32	4	0.24	28.89	76.47
0.65	24	12	12	5	0.29	50.00	70.59
0.55	21	10	11	7	0.41	47.62	58.82
0.45	18	11	7	6	0.35	61.11	64.71
0.35	13	9	4	8	0.47	69.23	52.94
0.25	14	11	3	6	0.35	78.57	64.71
0.15	9	8	1	9	0.53	88.89	47.06
0.05	6	6	0	11	0.65	100.00	35.29

The testing stage results are presented in Table 4.6

Table 4.6 Testing Time Series Results for Different ß Values at Harrisburg Gauging Station

Number of Floods in Testing Time Series= 8, Number of Events in Testing Time Series= 14									
ß Spec. In Training	Points in Cluster	*tp*	*fp*	Events Missed	*a*	Resultant ß	PPA *tp /tp+fp* or (1- ß)	CPP *tp(pred.)* / *tp(actual)*	Number of Starts Missed
0.95	226	10	216	4	0.29	0.96	4.42	71.43	0
0.85	59	10	49	4	0.29	0.83	16.95	71.43	0
0.75	51	10	41	4	0.29	0.80	19.61	71.43	0
0.65	19	8	11	6	0.43	0.58	42.11	57.14	2
0.55	18	9	9	5	0.36	0.50	50.00	64.29	1
0.45	17	8	9	6	0.43	0.53	47.06	57.14	2
0.35	14	9	5	5	0.36	0.36	64.29	64.29	1
0.25	12	8	4	6	0.43	0.33	66.67	57.14	1
0.15	8	7	1	7	0.50	0.13	87.50	50.00	3
0.05	7	7	0	7	0.50	0.00	100.00	50.00	3

Starting at ß = 0.65, the clusters determined from the training stage start missing floods. Again, as in the St. Louis and Kansas City examples, there is no clear relationship between ß and CPP in both the training and testing time series. For a ß of 0.15 and below, the clusters start missing out 3 floods out of the total 8 floods that occurred in the testing time series.

4.6.2 Earliness Prediction Accuracy

An event is predicted when a phase space point from the testing time series falls inside the cluster from the training time series. Since the goal is to predict the floods accurately and as early as possible, one performance measure is the Earliness Prediction Accuracy of TSDM using different step-ahead functions. For this, ß is kept constant and the prediction accuracy is measured in terms of *tp*, *fp*, *a*, ß, PPA and CPP for different step-ahead event characterization functions.

The ß with highest CPP value in the testing stage using the Objective Function I is selected. The objective function and optimization formulation are described on the next page.

Objective Function : Objective Function Type I

$$f(P) = \begin{cases} tp \times \sum_{i=1}^{M} g_M & If \ \dfrac{f_p}{f_p + t_p} \le b \\ -\left(\sum_{i=1}^{M} g_M \right) & otherwise \end{cases}$$

Optimization Formulation : 1. Maximize the value of Objective Function (Priority I)

2. Minimize the radius of cluster (Priority II)

Earliness Prediction Accuracy for St. Louis Gauging Station

The CPP in the testing time series is maximum for ß = 0.85 (Table 4.2). Using this ß value and the objective function and optimization formulation mentioned above, different event characterization functions are applied starting from the one step-ahead function. The training stage results for St. Louis gauging station are presented in Table 4.7 and Figure 4.11. From Table 4.7 and Figure 4.11 it is clear that the both the PPA and the CPP decrease as the prediction time horizon increases. Other indication of degradation of prediction accuracy with increasing prediction time horizon is the increase in *a* values.

Table 4.7 Training Earliness Prediction Accuracy for ß = 0.85 at St. Louis Gauging Station

Event Char. Func.	Points in Cluster	*tp*	*fp*	Events Missed	*a*	ß	PPA (*tp* /*tp*+*fp*) or (1- ß)	CPP *tp(predicted)* / *tp(actual)*
x_{t+1}	98	15	83	0	0.00	0.85	15.31	100.00
x_{t+2}	90	14	76	1	0.07	0.84	15.56	93.33
x_{t+3}	81	14	67	1	0.07	0.83	17.28	93.33
x_{t+4}	66	10	56	5	0.33	0.85	15.15	66.67
x_{t+5}	43	7	36	8	0.53	0.84	16.28	46.67
x_{t+6}	30	5	25	10	0.67	0.83	16.67	33.33
x_{t+7}	13	3	10	12	0.80	0.77	23.08	20.00
x_{t+8}	No Cluster Identified							

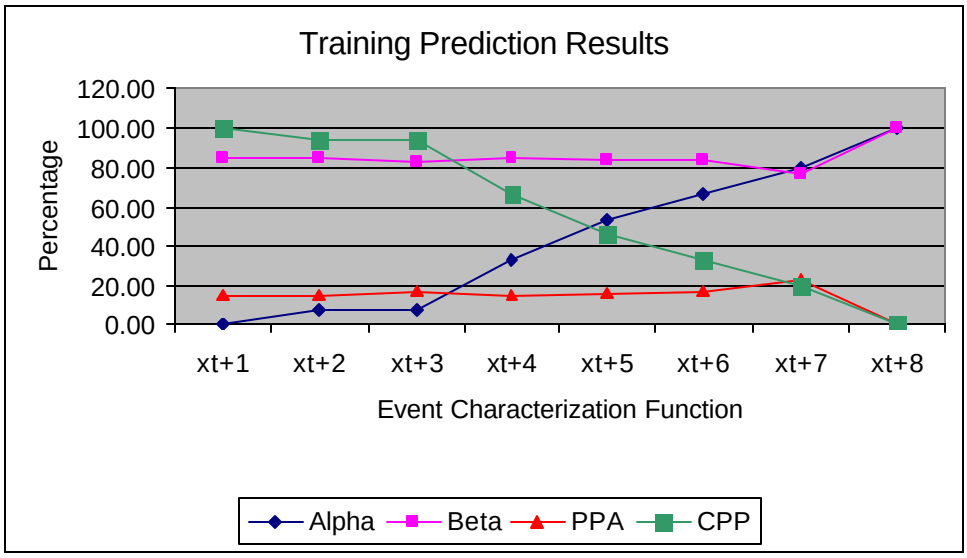

Figure 4.11 Training Earliness Prediction Accuracy for *ß* = 0.85 at St. Louis Gauging Station

The clusters found in the training phase are used to predict the floods in the testing phase. The prediction results from testing phase are displayed in Table 4.8 and Figure 4.12. As in the training phase, the prediction accuracy decreases with the increase in prediction horizon. For prediction horizon of seven days and more, the cluster misses the starts of floods.

Table 4.8 Testing Earliness Prediction Accuracy for *ß* = 0.85 at St. Louis Gauging Station

Number of Floods in Testing Time Series=2, Number of Events in Testing Time Series=33									
Event Char. Func.	Points in Cluster	*tp*	*fp*	Events Missed	*a*	Resultant *ß*	PPA *tp /tp+fp* or (1- *ß)*	CPP *tp(pred.)* / *tp(actual)*	Number of Starts Missed
x_{t+1}	90	20	70	13	0.39	0.78	22.22	60.61	0
x_{t+2}	75	10	65	23	0.70	0.87	13.33	30.30	0
x_{t+3}	73	10	63	23	0.70	0.86	13.70	30.30	0
x_{t+4}	43	7	36	26	0.79	0.84	16.28	21.21	0
x_{t+5}	36	4	32	29	0.88	0.89	11.11	12.12	0
x_{t+6}	24	3	21	30	0.91	0.88	12.50	9.09	0
x_{t+7}	16	1	15	32	0.97	0.94	6.25	3.03	1
x_{t+8}	10	0	10	33	1.00	1.00	0.00	0.00	2

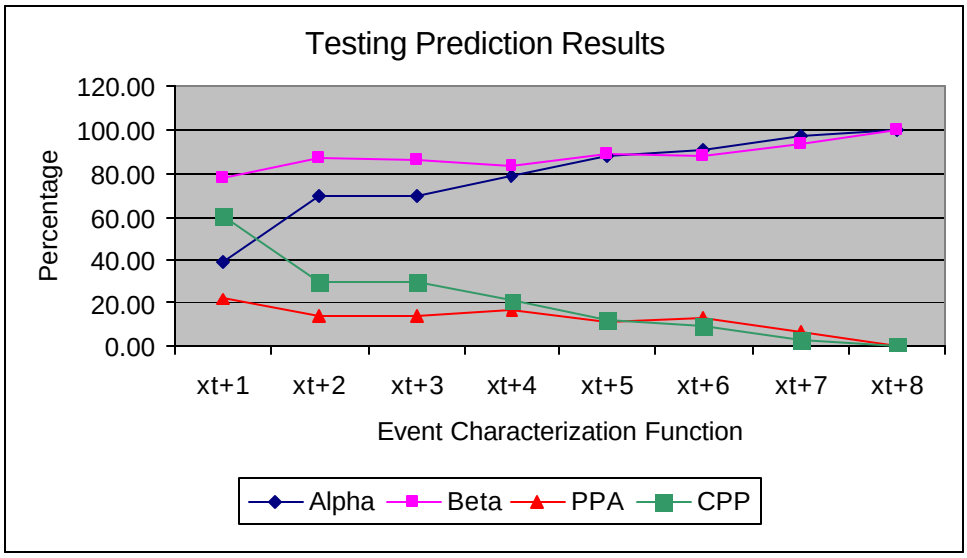

Figure 4.12 Testing Earliness Prediction Accuracy for β = 0.85 at St. Louis Gauging Station

Figure 4.13 displays the events predicted in the testing time series using some examples of step-ahead functions. The first time series represents the testing time series and, the following time series display the predicted events for different step-ahead event characterization functions. It can be seen from Figure 4.13 that as the prediction horizon increases from one day to seven days, the number of false alarms as well as the number of events predicted goes on decreasing. For a seven day prediction horizon, the cluster misses the start of one flood out of two.

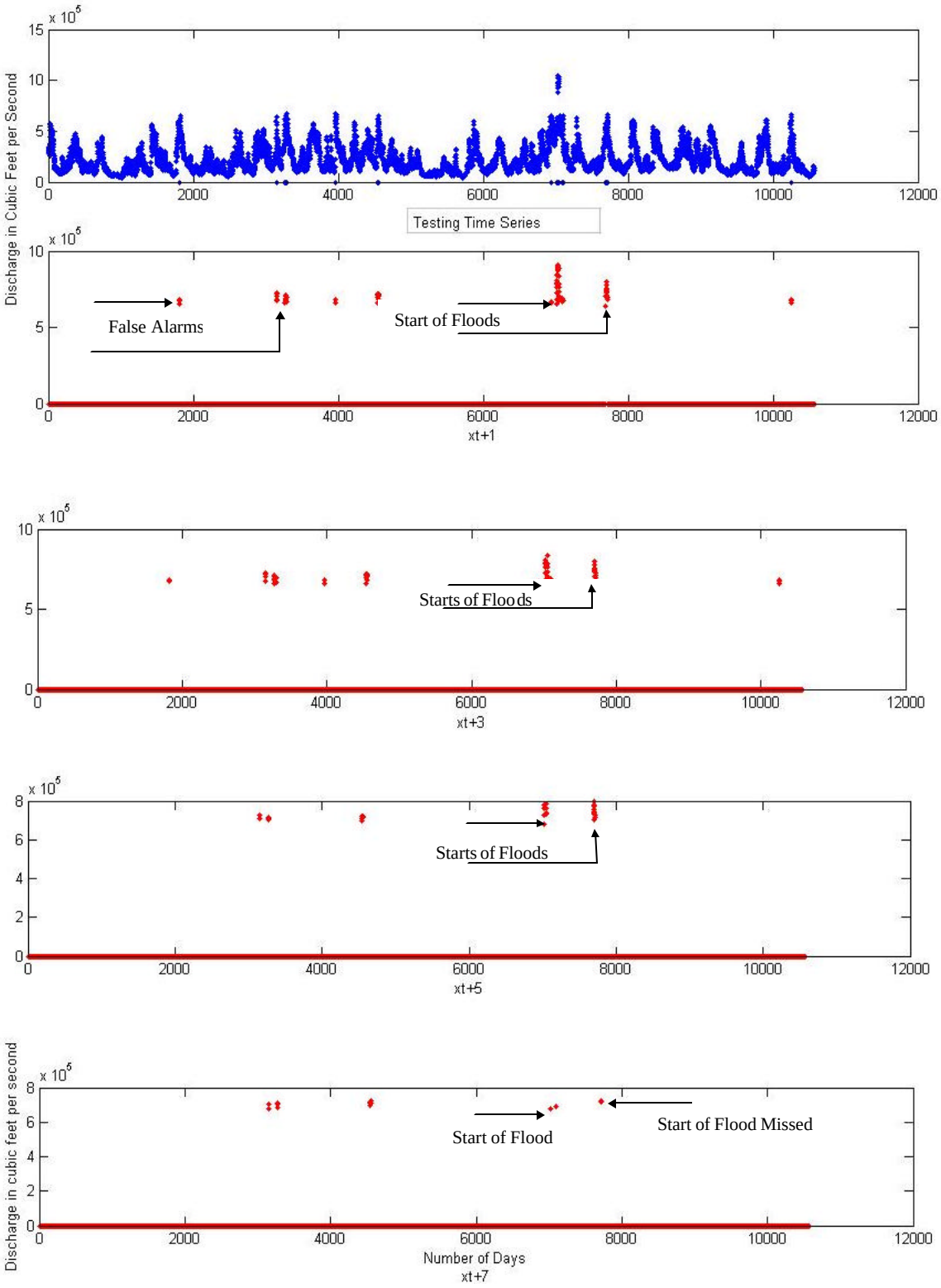

Figure 4.13 Earliness Prediction Results for β = 0.85 at St. Louis Gauging Station

Earliness Prediction Accuracy for Kansas City Gauging Station

The results for earliness prediction accuracy are presented in Table 4.9 and Figure 4.14. *ß* value of 0.25 is used for calculating the earliness prediction accuracy using different step-ahead event characterization functions.

Table 4.9 Training Earliness Prediction Accuracy for *ß* = 0.25 at Kansas Ct Gauging Station

Event Char. Func	Points in Cluster	*tp*	*fp*	Events Missed	*a*	*ß*	PPA (*tp* /*tp*+*fp*) or (1- *ß*)	CPP *tp(predicted)* / *tp(actual)*
x_{t+1}	6	5	1	1	0.17	0.17	83.33	83.33
x_{t+2}	5	4	1	2	0.33	0.20	80.00	66.67
x_{t+3}	2	2	0	4	0.67	0.00	100.00	33.33
x_{t+4}	0	0	0	6	1.00	0.00	0.00	0.00

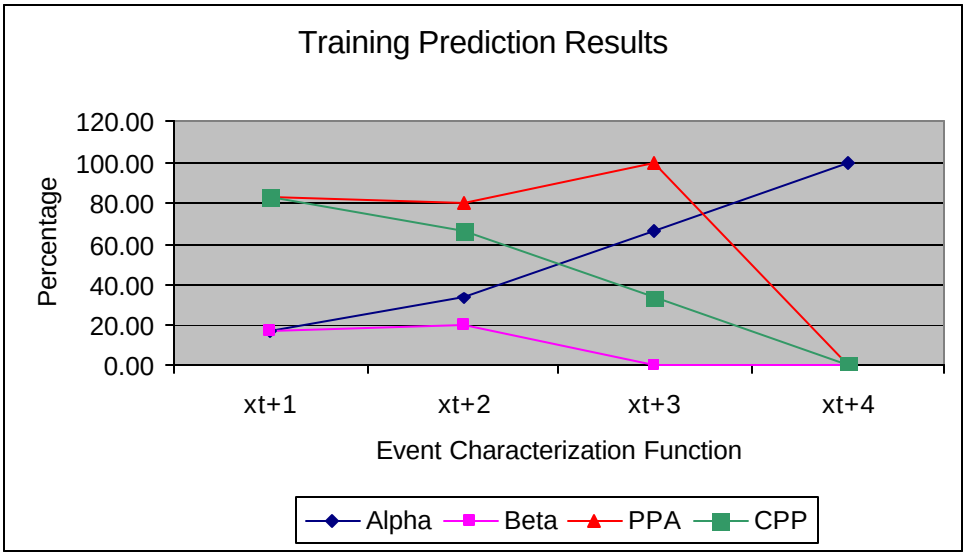

Figure 4.14 Training Earliness Prediction Accuracy for *ß* = 0.25 at Kansas Ct Gauging Station

Testing time series results are presented in Table 4.10 and Figure 4.15.

Table 4.10 Testing Earliness Prediction Accuracy for ß = 0.25 at Kansas Ct Gauging Station

Number of Floods in Testing Time Series= 1, Number of Events in Testing Time Series= 6									
Event Char Func	Points in Cluster	*tp*	*fp*	Events Missed	*a*	Resultant ß	PPA *tp /tp+fp* or (1- ß)	CPP *tp(pred.)* / *tp(actual)*	Number of Starts Missed
x_{t+1}	7	6	1	0	0.00	0.14	85.71	100.00	0
x_{t+2}	3	2	1	4	0.67	0.33	66.67	33.33	0
x_{t+3}	2	2	0	4	0.67	0.00	100.00	33.33	1

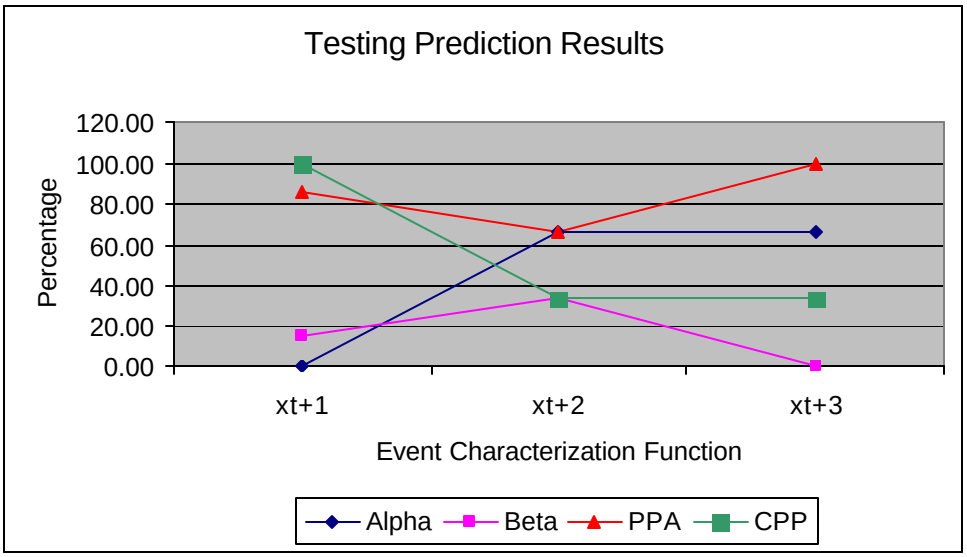

Figure 4.15 Testing Earliness Prediction Accuracy for ß = 0.25 at Kansas Ct Gauging Station

The CPP decreases as the prediction horizon is increased. The highest CPP is with the use of one step-ahead Event Characterization Function and it decreases drastically as the prediction horizon is increased to two days and more. The clusters start missing the flood for a prediction horizon of 3 days and more.

.

Earliness Prediction Accuracy for Harrisburg Gauging Station

The CPP is maximum at ß of 0.75. Using this ß value the earliness prediction accuracy is calculated using different step-ahead functions. The training stage results for Earliness Prediction Accuracy at Harrisburg gauging station are presented in Table 4.11 and Figure 4.16.

Table 4.11 Training Earliness Prediction Accuracy for ß = 0.75 at Harrisburg Gauging Station

Event Char. Func.	Points in Cluster	*tp*	*fp*	Events Missed	*a*	ß	PPA (*tp* /*tp*+*fp*) or (1- ß)	CPP *tp(predicted)* / *tp(actual)*
x_{t+1}	45	13	32	4	0.31	0.71	28.89	76.47
x_{t+2}	25	8	17	9	0.53	0.68	32.00	47.06
x_{t+3}	3	2	1	15	0.88	0.33	66.67	11.76
x_{t+4}	No Cluster Identified							

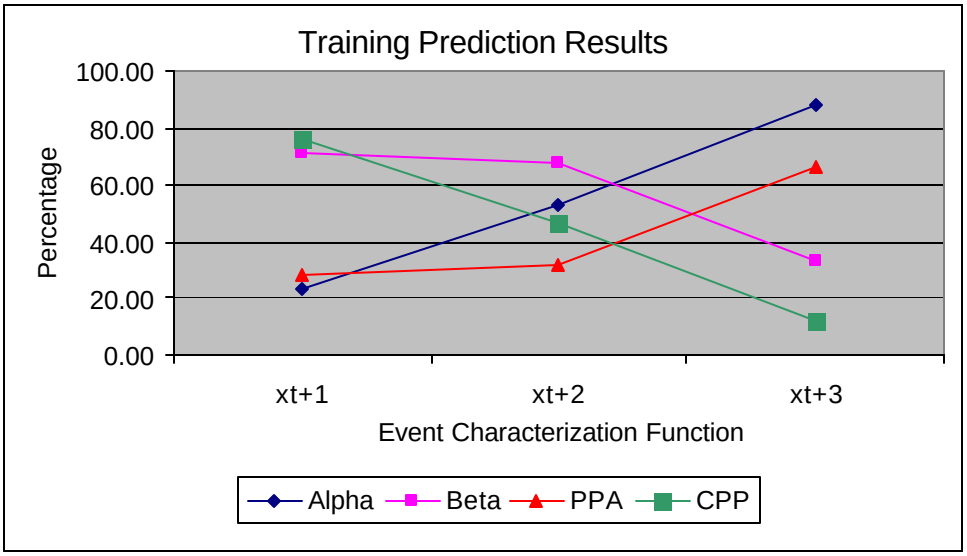

Figure 4.16 Training Earliness Prediction Accuracy at Harrisburg Gauging Station

The results show that the CPP degrades for longer prediction horizon and for a prediction horizon of 4 days, no cluster can be identified. The results from testing phase are presented in Table 4.12 and Figure 4.17.

Table 4.12 Testing Earliness Prediction Accuracy for ß = 0.75 at Harrisburg Gauging Station

Number of Floods in Testing Time Series=8, Number of Events in Testing Time Series=14									
Event Char. Func.	Points in Cluster	*tp*	*fp*	Events Missed	*a*	Resultant ß	PPA *tp /tp+fp* or (1- ß)	CPP *tp(pred.)* / *tp(actual)*	Number of Starts Missed
x_{t+1}	51	10	41	4	0.29	0.80	0.20	0.71	0
x_{t+2}	25	3	22	11	0.79	0.88	0.12	0.21	6
x_{t+3}	1	0	1	14	1.00	1.00	0.00	0.00	8
x_{t+4}	No Cluster Identified								

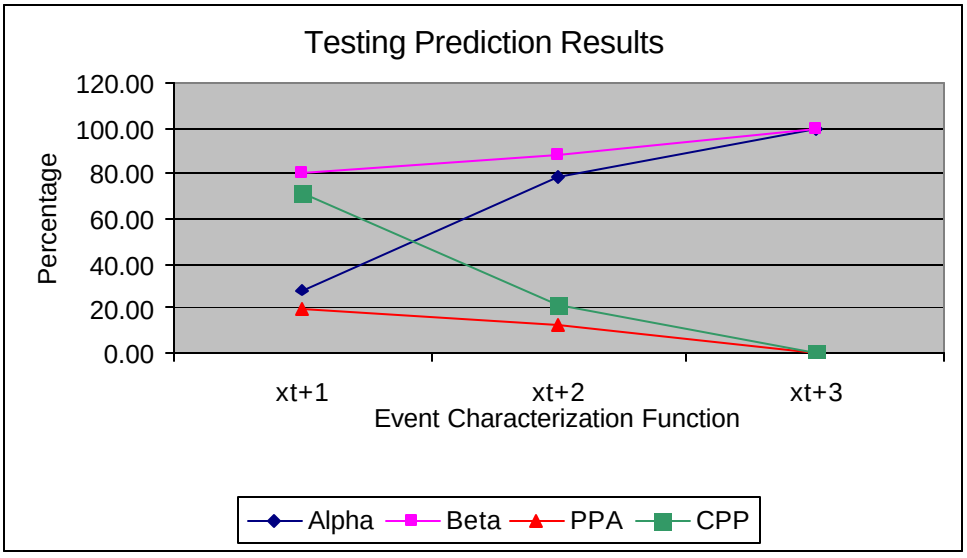

Figure 4.17 Testing Earliness Prediction Accuracy at Harrisburg Gauging Station

For prediction periods longer than a day, the cluster misses the starts of floods. Out of the 8 floods in the testing time series, 6 floods are missed for a prediction horizon of 2 days and for a 3 day prediction horizon all 8 floods are missed.

4.6.3 Prediction Accuracy with Objective Function II

The results presented in this section are for different values of ß using Objective Function II applied to the St. Louis gauging station. The following set of parameters is used :

Event Characterization Function : One step-ahead function – $g(x_t) = x_{t+1}$

Objective Function : Objective Function I

$$f(P) = \begin{cases} \sum_{i=1}^{M} g_M & \text{If } c(M)/c(?) = ? \\ -\left(\sum_{i=1}^{M} g_M \right) & \text{otherwise} \end{cases}$$

Optimization Formulation : 1. Maximize the value of Objective Function (Priority I)

2. Minimize the radius of cluster (Priority II)

The difference between Objective Function II and Objective Function I used in Section 4.6.1 is that the summation of g values inside the cluster is not multiplied by the number of true positives inside the cluster. This means that the GA algorithm will not be rewarded for selecting a cluster with high number of *tp*'s. Objective Function II is useful when the history of floods is not known. The GA will select the points with high g values to be a part of cluster in order to maximize the summation of g values. Since the phase space points representing floods are points with high g values, the GA would try to include them in the cluster to maximize the objective function. The cluster size is restricted by the ? value which is determined from the flood zone specification.

The information regarding flood zone at St. Louis gauging station was not available. Therefore, for the purpose of demonstration, instead of the ? value, the cluster size is restricted using ß values as in Section 4.6.1. The objective function is modified as

$$f(P) = \begin{cases} \sum_{i=1}^{M} g_M & \text{If } \dfrac{f_p}{f_p + t_p} \le b \\ -\left(\sum_{i=1}^{M} g_M \right) & \text{otherwise} \end{cases}$$

The training and testing stage results are presented next.

Table 4.13 displays the training stage results for different values of ß.

Table 4.13 Training Stage Results for Different ß Values Using Objective Function II

ß Specified	Points in Cluster	*tp*	*fp*	Events Missed	*a*	PPA (*tp /tp+fp)* or (1- ß)	CPP *tp(predicted) tp(actual)*
0.95	295	15	280	0	0.00	5	100
0.85	98	15	83	0	0.00	15	100
0.75	60	15	45	0	0.00	25	100
0.65	42	15	27	0	0.00	36	100
0.55	33	15	18	0	0.00	45	100
0.45	27	15	12	0	0.00	56	100
0.35	17	14	3	1	0.07	82	93
0.25	13	10	3	5	0.33	77	67
0.15	7	6	1	9	0.60	86	40
0.05	6	6	0	9	0.60	100	40

As shown in Table 4.13, the Prediction Accuracy is 100% for ß value of 0.45 and after that, the cluster starts missing *tp*'s which decreases the prediction accuracy.

Testing stage results using Objective Function II are presented in Table 4.14. The cluster starts missing starts of floods at ß of 0.05. The *a* and Positive Prediction Accuracy increase with decrease in ß and Prediction Accuracy. This happens because a reduction in ß reduces the number of *fp*'s, making the cluster crisper and increasing its PPA. However, it also starts to miss more *tp*'s thereby reducing CPP.

Table 4.14 Testing Stage Results for Different ß Values Using Objective Function II

Number of Floods in Testing Time Series= 2, Number of Events in Testing Time Series= 33									
ß Spec. In Training	Points in Cluster	*tp*	*fp*	Events Missed	*a*	Resultant ß	PPA *tp /tp+fp* or (1- ß)	CPP *tp(pred.) tp(actual)*	Number of Starts Missed
0.95	272	12	260	21	0.64	0.96	4	36	0
0.85	90	20	70	13	0.39	0.78	22	61	0
0.75	53	17	36	16	0.48	0.68	32	52	0
0.65	26	12	14	21	0.64	0.54	46	36	0
0.55	18	14	4	19	0.58	0.22	78	42	0
0.45	18	15	3	18	0.55	0.17	83	45	0
0.35	10	9	1	24	0.73	0.10	90	27	0
0.25	6	5	1	28	0.85	0.17	83	15	0
0.15	7	7	0	26	0.79	0.00	100	21	0
0.05	1	1	0	32	0.97	0.00	100	3	1

4.7 Summary of Results

1. From the training and testing results for all three gauging stations, it can be seen that no clear relationship exists between the specified ß for a cluster and its Correct Prediction Percentage. As explained earlier, the non-uniform density of phase space is responsible for this. If the phase space would have been uniform in density, a higher value of ß would have led to a cluster with larger radius since more *fp*'s are allowed. The CPP would also be high considering the fact that a cluster with a larger radius would include a high number of *tp*'s. However, the density of phase space points is non-uniform. Consider two clusters in the Harrisburg gauging station phase space as shown in Figure 4.19. Specifying a smaller value for ß (allowable proportion of *fp*'s in the cluster) may lead to Cluster 1 being identified as the optimal cluster. The density of points in Cluster 1 is sparse which allows a larger radius for the cluster. Since it includes more number of *tp*'s, the CPP for Cluster 1 is high. On the other hand, specifying a large ß may lead to Cluster 2 with a smaller radius (since points are dense) and less CPP because it misses some of the *tp*'s. Thus, the CPP not only depends on the ß specified, but also on the location of the cluster in the non-uniform phase space.
2. An inverse relationship exists between the earliness of prediction and the CPP. It can be observed from Tables 4.8, 4.10 and 4.13 that as the prediction horizon increases, the Correct Prediction Accuracy decreases and, the number of starts of floods missed also increases. Ideally, one would like to predict a flood as early as possible, however, associated with the earliness of prediction is the probability of missing the start of floods.
3. Using Objective Function II, the CPP was found to be either to equal or lower than that using Objective Function I, for different values of ß. This result was expected because the Objective Function I uses the historical floods and threshold discharge values to train the GA in the search for optimal temporal pattern clusters. Figure 4.20 presents a graph comparing the CPP values for the St. Louis gauging station example using Objective Function I and II.

The ? value in Objective Function II was replaced by ß since the information on flood zoning was not available. However, using the ?, the CPP would be even lower than the CPP using ß because the flood zoning values are calculated using linear regression models which are inadequate in the analysis of nonlinear time series.

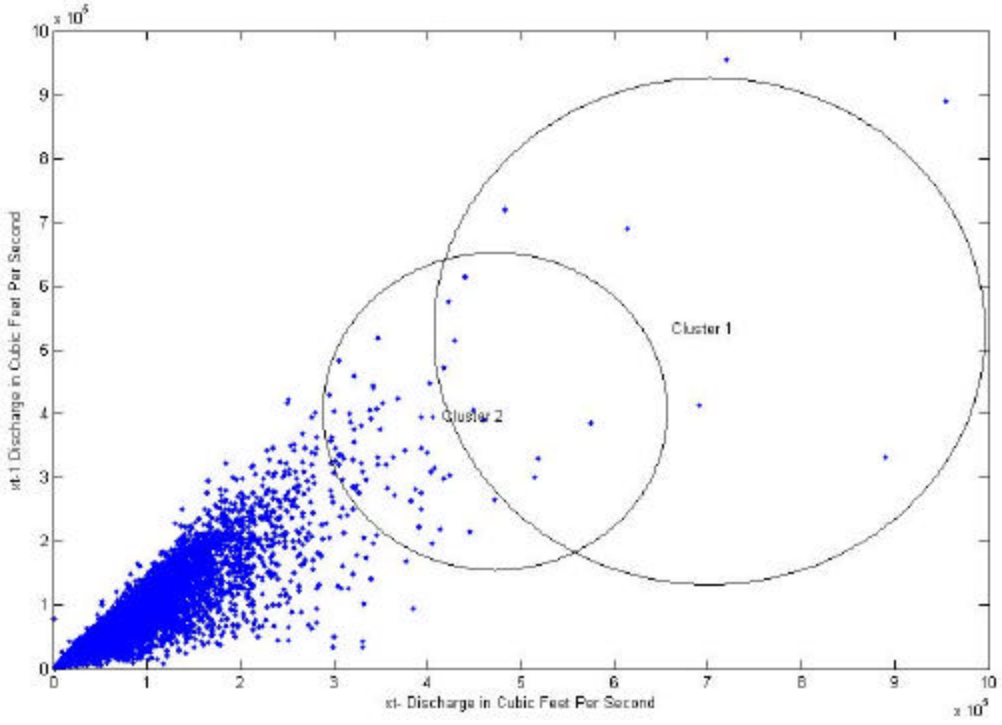

Figure 4.19 Comparison of Two Clusters in Phase Space

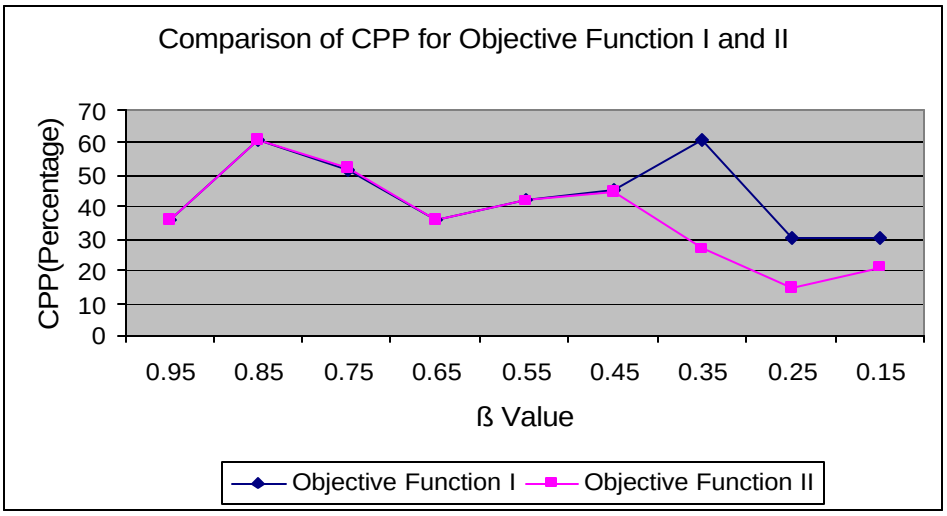

Figure 4.20 Comparison of CPP for Objective Function I and II at St. Louis Gauging Station

In this Chapter the results were presented for flood prediction problem applied to three gauging stations. Chapter 5 presents the conclusion and the directions for future research.

Chapter 5. Conclusions and Future Work

This chapter presents the conclusions and directions for further research. Section 5.1 summarizes this research and presents its application as a decision making tool. Section 5.2 enlists the directions for future research in this area.

5.1 Conclusions

Time Series Data Mining is applied to the area of flood forecasting with the goal of predicting floods accurately and as early as possible. Three examples of gauging stations, representing high, medium and low flood occurrences are considered. The prediction accuracy is evaluated in terms of a, $ß$, Positive Prediction Accuracy (PPA) and Correct Prediction Accuracy (CPP). Two variations of objective function are presented. Objective Function I can be used in flood forecasting problems where the information about history of floods is available. Where this information is not available, the flood zoning information can be used along with Objective Function II. The effect of earliness of prediction on the prediction accuracy is also presented.

Earlier approaches have dealt with forecasting magnitudes of future discharge values. This research focuses on the early prediction of floods (events). It is the first application of an event based data mining technique to flood forecasting.

The predictions are specific to the location of the gauging station and its catchment area. Variations in factors such as river depth, cross section, rainfall runoff , snowmelt affect the flood characteristics. For example, for the St. Louis gauging station, a discharge of 780,000 cubic feet per second causes the river to overflow. On the other hand, for the Kansas City gauging station, a

discharge of 380,000 cubic feet per second is enough to make the river overflow. Hence, the predictions are not generic and for a flood prediction problem at another location, the GA needs to be trained on the discharge time series from that location.

The prediction results from TSDM can be used as a decision making tool by city planners. The decision variables in this approach are ß (Proportion of False Positives) and, the Event Characterization Function (Step-ahead function). Depending on the location of the gauging station and its catchment area, the impact of flood on surrounding human population and economy, the city planners can decide the values for the above mentioned variables that would provide them enough time to plan for flood mitigation and evacuation procedures. For example, the flooding on Hillsborough River passing through the City of Tampa will affect a huge population and have large scale economic repercussions. Whereas, a flood on the Suwanee River that passes through large unpopulated areas in Florida would have a lesser impact. Thus, for an impact area such as the City of Tampa, the planners would not want to miss predicting a flood and would also accept a certain number of false alarms as long as the no flood is missed. This criterion determines the value of ß (proportion of false positives) for the cluster. From the results it can be seen that a higher ß leads to high number of false alarms in the testing time series. However, associated with ß, there is an *a* (proportion of points missed). Hence, when choosing the value of ß, the decision must also consider the *a* value for that cluster. There is no identifiable relationship between *a* and ß of a cluster, due to the non-uniform nature of the phase space. In most cases, the decision makers would want to select parameters such that both *a* and ß are minimized and PPA and CPP are maximized.

Another factor that the city planners have to account for is the earliness of prediction. This factor affects the choice of event characterization function. The planners would want to predict the flood as early as possible; however, a tradeoff exists between the earliness of prediction and

the Correct Prediction Percentage. From the results it can be seen that the earlier the TSDM methodology tries to predict the flood, less is the and Correct Prediction Percentage.

Concluding, this research leads to the development of a decision making tool for planning flood mitigation and evacuation procedures for use by planning authorities. Based on location and the possible impact , the planners can make a choice in selecting the parameters for TSDM. The tradeoffs involved in selecting different values for these parameters are also presented. This is a general approach and can be applied to any gauging station and its catchment area for flood prediction.

5.2 Future Work

1. Decision Support Software Development

A Decision Support Software can be developed for a general purpose use by city planners and emergency operations agencies. The proposed software would have a graphical user interface with input screens to allow users to input the training time series for any gauging station. Depending on whether the history of floods is available the software will train the GA using one of the two objective functions.

In cases where the history of floods is available, the users are allowed to choose the value for ß. As mentioned in Section 5.1, this selection would depend on the impact the flood can have on the affected area. For cases where the historical information on floods is not available, the GA can be trained using the Objective Function II. Generally, all residential areas that have a probability of being affected by a flood are classified into flood zones for insurance and mortgage calculations by government or private agencies. The Objective Function II makes use of this zoning information to search for optimal clusters indicative of floods. The software would also allow the decision makers to choose how many days early they want the prediction to be.

The possible customers for this software would be insurance agencies and city planning authorities.

2. Use of Multiple Parameters

In this research the daily discharge time series is used for flood prediction. A future study could use other data sets such as rainfall runoff, the height of river (water level), snowmelt. The use of combination of multiple parameters such as a discharge and rainfall runoff time series may lead to higher prediction accuracy. Another approach would be to modify the Event Characterization Function itself to include multiple variables.

3. Changing Embedding Time Delay

It was observed that the efficiency of the GA in finding a cluster depends on the spread of points in the phase space. For St. Louis gauging station, the spread of phase space points is minimum and the points are concentrated around the diagonal. The spread of points can be controlled by specifying different values for delay. A higher delay would possibly lead to lower spread in the phase space points, however it also results in loss of information. An interesting direction for future research would be to experiment with different time delays to observe the effect on the prediction accuracy of the GA.

4. TSDM in Conjunction with NLP

TSDM and NLP can be used in conjunction to predict both the time and magnitude of floods. This will help in real time alerting and evacuation planning. If the time of floods and magnitude is known the area of impact can be estimated more accurately. For example, discharge over the threshold would cause the river to overflow, however the magnitude of discharge will determine what area is actually affected by the flood.

References

[1] Abarbanel H.D.I. *Analysis of Observed Chaotic Data*. Institute for Nonlinear Science, 1996.

[2] Albano A.M, Passamante A., and Farell M.E. *Using higher-order correlations to define an embedding window.* Physica D, 54:85-97, 1991.

[3] Ayewah N. *Prediction of Spatial Temporal Events Using a Hidden Markov Model.* Department of Computer Science and Engineering, Southern Methodist University, Dallas, TX, 2003.

[4] Bangura J.F., Povinelli R.J., Demerdash N.A.O., Brown R.H. *Diagnostics of Eccentricities and Bar/End-Ring Connector Breakages in Polyphase Induction Motors through a Combination of Time-Series Data Mining and Time-Stepping Coupled FE-State Space Techniques*. IEEE Transactions On Industry Applications, vol. 39, no. 4, 1005-1013, 2003.

[5] Boogard H. F. P. van den, Gautam, D. K. and Mynett, A. E. *Auto-regressive neural networks for the modeling of time series.* Hydrodinamics98, Babovic & Larsen (eds), Balkema, Rotterdam. 741-748, 1998. Publishers, Dordrecht, 23-51, 2000.

[6] Buzug T., Pfister G. *Comparison of algorithms calculating optimal parameters for delay time coordinates*. Physica D, 58:127, 1992.

[7] Buzug T., Reamers T., and Pfister G. *Optimal reconstruction of strange attractors from purely geometrical arguments*. Europhysics Letters, 13:605-610, 1990.

[8] Cao L., Mees A., Judd K. *Dynamics from multivariate time series.* Physica D, 121:75-88, 1998.

[9] Clemins P., Povinelli R.J. *Detecting Regimes in Temperature Time Series.* Artificial Neural Networks in Engineering, Proceedings, 727-732, 2001.

[10] Coulibaly, P., Anctil, F. and Bobée, B. *Daily reservoir inflow forecasting using artificial neural networks with stopped training approach*. Journal of Hydrology 230, 244-257 2000.

[11] Deo, M. C. and Thirumalaiah, K. *Real time forecasting using neural networks*. Artificial neural networks in Hydrology, R. S. Govindaraju and A. Ramachandra Rao (eds), Kluwer Academic Publishers, Dordrecht, 53-71, 2000.

[12] Diggs D.H., Povinelli. R.J. *A Temporal Pattern Approach for Predicting Weekly Financial Time Series*. Artificial Neural Networks in Engineering, St. Louis, Missouri, 707-712, 2003.

[13] Duan M., Povinelli R.J. *Estimating Stock Price Predictability Using Genetic Programming*. Proceedings of the Genetic and Evolutionary Computation Conference (GECCO2001), 174, 2001.

[14] Farmer and Sidorowich. *Predicting chaotic time series*. Physics Review Letters 59, 845–848, 1987.

[15] Fraser A. M., and Swinney H. L. *Independent coordinates for strange attractors from mutual information.* Physics Review A, 33:1134-1140, 1986.

[16] Fraser A.M. and Swiney, H.L. *Independent coordinates for strange attractors from mutual information.* Physics Review A, 33, 1134, 1986.

[17] Galka A. *Topics in nonlinear Time Series Analysis-With Implications for EEG Analysis.* World Scientific Publishing Company, 2000.

[18] Goldberg D.E. *Genetic Algorithms in Search, Optimization and Machine Learning.* Reading, Mass.: Addison-Wesley, 1989.

[19] Hilborn R. C. and Ding M. *Optimal reconstruction space for estimating correlation dimension*. International Journal of Bifurcation and Chaos, 6:377-381, 1996.

[20] Hsu K., Gupta H.V. and Sorooshian S. *Artificial neural network modeling of the rainfall runoff process*. Water Resources Research, 31(10), 2517-2530, 1995.

[21] Internet Website URL: http://www.usgs.gov/index.html , United States Geological Survey, 2004.

[22] Imrie C.E., Durucan S. and Korre A. *River flow prediction using artificial neural networks: generalization beyond the calibration range*. Journal of Hydrology 233, 138-153, 2000.

[23] Islam M.N., Shivkumar B. *Characterization and Prediction of runoff dynamics: a nonlinear dynamical view*. Advances in Water Resources, 25:179-190, 1996.

[24] Jayawardena A.W., Lai F. *Analysis and prediction of chaos in rainfall and stream flow time series*. Journal of Hydrology, 153:23-52, 1994.

[25] Kantz H. and Schrieber T. *Nonlinear Time Series Analysis.* Cambridge Nonlinear Science Series, No. 7, 1999.

[26] Kennel M.B., Brown R., Abarbanel H.D.I. *Determining embedding dimension for phase space reconstruction using geometric method*. Physics Review A; 45:3403-11, 1992.

[27] Kingston, G., Lambert, M., Maier, H. *Development of Stochastic Artificial Neural Networks for Hydrological Prediction*. Centre for Applied Modeling in Water Engineering, University of Adelaide, Australia, 2003.

[28] Laio F., Porporato A., Revelli R. and Ridolfi L. *A comparison of nonlinear forecasting methods*. Water Resources Research, Vol 39, No.5, 1129, 2003.

[29] Li Z. and Dunham M.H. STIFF: *A Forecasting Framework for Spatio-Temporal Data*. Proceedings of the KDMCD Workshop, vol., p. 1, 2002.

[30] Maulik U. and Bandyopadhyay S. *Genetic Algorithm Based Clustering Technique*, Pattern Recognition, vol. 33, no. 9, pp. 1455-1465, 2000.

[31] Mees A.I. (Editor). *Nonlinear Dynamics and Statistics*. Birkhäuser Boston, 2001.

[32] Montgomery D. C., Runger G. C., and Hubele N. F. *Engineering Statistics*. John Wiley and Sons, 1997.

[33] Olbrich E. and Kantz H. *Inferring chaotic dynamics from time series: on which length scale determinism becomes visible*. Physics Letters A, 232:63-69, 1997.

[34] Organ D. and Yalcin A., *Flood Forecasting using Nonlinear Time Series Analysis.* REU Report, submitted to the University of South Florida, College of Engineering, 2004.

[35] Paulus M. *Testing for nonlinearity using redundancies: Quantitative and Qualitative aspects*. Physica D, 80:186-205, 1995.

[36] Porporato A., Ridolfi L. *Clues to existence of deterministic chaos in river flow.* International Journal of Modern Physics B; 10:1821-62, 1996.

[37] Porporato A., Ridolfi L. *Multivariate nonlinear prediction of river flows*. Journal of Hydrology 248 (1-4): 109-22, 2001.

[38] Porporato A., Ridolfi L. *Nonlinear analysis of river flow time sequences.* Water Resources Research, 33(6), 1353-1367, 1997.

[39] Povinelli R. *A New Temporal Pattern Identification Method for Characterization and Prediction of Complex Time Series Events*. IEEE Transactions on Knowledge and Data Engineering, Vol. 15, No.2, 2003.

[40] Povinelli R.J. *Identifying Temporal Patterns for Characterization and Prediction of Financial Time Series Events*. Temporal, Spatial and Spatio-Temporal Data Mining: First International Workshop; revised papers / TSDM2000, 46-61, 2000.

[41] Povinelli R.J. Time Series Data Mining: *Identifying Temporal Patterns for Characterization and Prediction of Time Series Events*. Ph.D. Dissertation, Marquette University, Milwaukee, WI, 1999.

[42] Povinelli R.J., Feng X. *Characterization And Prediction Of Welding Droplet Release Using Time Series Data Mining*. Artificial Neural Networks in Engineering, Proceedings, 857-862, 2000.

[43] Sauer T., Yorke J.A. and Casdagli M. *Embedology*. Journal of Statistical Physics, Vol 65, pp. 579-616, 1991.

[44] Sivakumar B. *Chaos theory in geophysics: past, present and future*. Chaos, Solutions & Fractals, 19:441-462, 2004.

[45] Sivakumar B., A.W. Jayawardena, T.M., and Fernando K.G. *River flow forecasting: use of phase-space reconstruction and artificial neural networks approaches*. Journal of Hydrology 265: 225–245, 2002.

[46] Solomatine D.P., Rojas C., Velickov S., Wust H. *Chaos theory in predicting surge water levels in the North Sea*. Proceedings 4th International Conference on Hydro informatics. Iowa, USA, July 2000.

[47] Takens F. *Detecting strange attractors in fluid turbulence*. Dynamical systems and turbulence, pages 366-381. Springer, Berlin 1981.

[48] Theiler J., Eubank S., Longtin A., Galdrikian B., and Farmer J.D. *Testing for non linearity in time series: the method of surrogate data*. Physica D, 58:77-94, 1992.

[49] Whitney H. *Differentiable manifolds*. Annals of Mathematics, 37:645, 1936.

[50] Wilcox B.P., Seyfried M.S., Blackburn W.H., Matison T.H. *Chaotic characteristics of snowmelt runoff: a preliminary study*. Symposium on Watershed Management. Durango, Co: American Society of Civil engineering; 1990.

[51] Wilcox B.P., Seyfried M.S., Matison T.H. *Searching for chaotic dynamics in snowmelt runoff*. Water Resources Research; 27(6): 1005-10, 1991.

[52] Zaldivar, J.M., Gutierrez, E., Galvan, I.M., Strozzi, F. and Tomasin, A. *Forecasting high waters at Venice Lagoon using chaotic time series analysis and non-linear neural networks*. Journal of Hydroinformatics, vol. 2, No.1, pp. 61-84, 2000.

[53] Zealand C. M., Burn D. H. and Simonovic S. P. *Short term streamflow forecasting using artificial neural networks*. Journal of Hydrology, 214, 32-48, 1999.

[54] Zimmerman M.W., Povinelli R.J. *On Improving the Classification of Myocardial Ischemia Using Holter ECG Data*. Computers in Cardiology, Chicago, Illinois, September 19-22, 2004.